INL/EXT-14-33957

# Light Water Reactor Sustainability Program

# A Distributed Control System Prototyping Environment to Support Control Room Modernization

December 2014

U.S. Department of Energy

Office of Nuclear Energy

INL/EXT-14-33957

# A Distributed Control System Prototyping Environment to Support Control Room Modernization

Roger Lew, Ronald Boring, Thomas Ulrich

December 2014

Idaho National Laboratory
Idaho Falls, Idaho 83415

http://www.inl.gov

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

(This page intentionally left blank)

# ABSTRACT

Operators of critical processes, such as nuclear power production, must contend with highly complex systems, procedures, and regulations. Developing human-machine interfaces (HMIs) that better support operators is a high priority for ensuring the safe and reliable operation of critical processes. Human factors engineering (HFE) provides a rich and mature set of tools for evaluating the performance of HMIs, however the set of tools for developing and designing HMIs is still in its infancy. Here we propose a rapid prototyping approach for integrating proposed HMIs into their native environments before a design is finalized. This approach allows researchers and developers to test design ideas and eliminate design flaws prior to fully developing the new system.

We illustrate this approach with four prototype designs developed using Microsoft's Windows Presentation Foundation (WPF). One example is integrated into a microworld environment to test the functionality of the design and identify the optimal level of automation for a new system in a nuclear power plant. The other three examples are integrated into a full-scale, glasstop digital simulator of a nuclear power plant. One example demonstrates the capabilities of next generation control concepts; another aims to expand the current state of the art; lastly, an HMI prototype was developed as a test platform for a new control system currently in development at U.S. nuclear power plants.

WPF possesses several characteristics that make it well suited to HMI design. It provides a tremendous amount of flexibility, agility, robustness, and extensibility. Distributed control system (DCS) specific environments tend to focus on the safety and reliability requirements for real-world interfaces and consequently have less emphasis on providing functionality to support novel interaction paradigms. Because of WPF's large user-base, Microsoft can provide an extremely mature tool. Within process control applications,WPF is platform independent and can communicate with popular full-scope process control simulator vendor plant models and DCS platforms.

(This page intentionally left blank)

# ACKNOWLEDGMENTS

The original idea for using Microsoft WPF as a prototyping tool in a full-scope simulator stemmed from conversations with John McLerran of Nuscale Power. We wish to express our sincere gratitude and give appropriate credit to Mr. McLerran’s inventiveness. This report was made possible through funding by the U.S. Department of Energy (DOE) Light Water Reactor Sustainability (LWRS) program. We are grateful to Richard Reister of the DOE and Bruce Hallbert, Kenneth Thomas, and Kathryn McCarthy of the Idaho National Laboratory (INL) for championing this effort.

(This page intentionally left blank)

# CONTENTS

# FIGURES

# ACRONYMS

| | |
|---|---|
| ADC | Analog to Digital Converter |
| CBP | Computer Based Procedure |
| COSS | Computerized Operator Support System |
| CSE | Cognitive Systems Engineering |
| CVCS | Chemical and Volume Control System |
| DCS | Distributed Control System |
| DLL | Dynamic Link Library |
| DOE | Department of Energy |
| EID | Ecological Interface Design |
| gPWR | Generic Pressurized Water Reactor |
| HFE | Human Factors Engineering |
| HMI | Human Machine Interface |
| HSSL | Human Systems Simulation Laboratory |
| HTML | HyperText Markup Language |
| I&C | Instrumentation and Controls |
| IDE | Integrated Development Environment |
| INL | Idaho National Laboratory |
| IRD | Information Rich Design |
| JADE | Java Application Development Environment |
| LWRS | Light Water Reactor Sustainability |
| MCB | Main Control Board |
| NEET | Nuclear Energy Enabling Technologies |
| NGNP | Next Generation Nuclear Plants |
| NUREG | Nuclear Regulatory Committee Regulatory Guidelines |
| OPC | Object Linking and Embedding for Process Control |
| P&ID | Piping and Instrumentation Diagram |
| PLC | Programable Logic Controller |
| PWR | Pressurized Water Reactor |
| TCS | Turbine Control System |
| SCADA | Supervisory Control and Data Acquisition System |
| SMR | Small Modular Reactor |
| U.S. | United States |
| WPF | Windows Presentation Foundation |
| WSC | Western Services Corporation |
| XAML | Extensible Application Markup Language |
| XML | Extensible Markup Language |

(This page intentionally left blank)

# 1. INTRODUCTION

In support of nuclear power plant control room modernization, we have developed a prototyping environment that allows for rapid and agile application development, precise control of human-machine interface (HMI) designs, and precise testing of candidate HMIs. A flexible prototyping environment will allow:

- **Theoretical research of human perception and cognition pertaining to human machine interaction.** A microworld, which often manifests as a simplified process control HMI, allows research into perceptual phenomena such as the optimal placement of an indicator or the display of information, as well as cognitive phenomena such as the working memory ability of operators, or the cognition involved with diagnosing the state of a plant process. A microworld also allows testing of these phenomena in a non-specialized environment such as a university psychology laboratory. Because the process control is typically simplified, it's often possible to use non-specialists for testing. For example, a study to investigate alarms might use student operators interacting with a microworld. The findings from such studies should be validated with additional studies using actual operators.

- **Applied research of advanced HMI concepts.** Because the prototyping environment is flexible, it is possible to develop proof-of-concept interfaces such as those employing intelligent control systems, increased automation, or even advanced automation. Such capabilities may be beyond the current feature sets of commercial distributed control system (DCS) platforms. Having a prototyping environment to implement and evaluate advanced concepts can serve as a precursor to eventual implementation in qualified systems.

- **Early stage evaluation of design concepts with licensed operators.** The prototyping environment is capable of mimicking the functionality as well as the look and feel of commercial DCS platforms. For early stage work, the process control is often not fully implemented; however, the prototyping environment can emulate the process control logic, thus allowing more extensive early-stage iteration and refinement of DCS designs prior to final implementation in the actual DCS.

This report chronicles the prototyping development environment we adopted to support a variety of projects. In the next chapter, we discuss the need for careful and thoughtful HMI design as the nuclear industry transitions from analog to digital control systems. In Chapter 3, we introduce the two testing environments in which we have deployed prototype HMIs: microworlds and full-scope simulators. We then introduce the development environment we used to create the prototypes in Chapter 4, and finally provide examples of the applications themselves in Chapter 5. Ultimately, despite the numerous applications presented, the prototyping environment primarily serves as the development platform for DCS prototypes in support of the United States (U.S.) Department of Energy's (DOE's) Light Water Reactor Sustainability (LWRS) Program. These DCS prototypes are the backbone for designing HMIs for control room modernization in the current U.S. commercial nuclear power fleet.

(This page intentionally left blank)

# 2. BACKGROUND

Monitoring and controlling industrial processes, such as nuclear power production, is complex, and human performance represents the single largest contributing factor to the overall reliability of a system [1]. The dynamics of such processes are often high-order with cross-coupled key process variables and long time lags. Further complexity is added through the incorporation of setpoint controllers, safety interlock systems, and other additional forms of automation. Consequently, operators must understand how to anticipate, monitor, and occasionally override automation in addition to understanding the processes being controlled by the automation. Most often, the actual system processes cannot be directly observed. Operators must rely on "keyhole" views of processes provided by electro-mechanical sensors and digital HMIs [2].

The proliferation of digital systems offers many new capabilities. System processes estimates can be made more reliable by incorporating redundant sensors, redundant analog-to-digital converters (ADC), and applying digital signal filtering. Signals can be recorded to data historians and displayed at a moment's notice. In many instances, indications do not directly convey the information that may be of interest to an operator. For example, in a two-state thermo-hydraulic system (i.e., liquid and vapor) operators may need to know the mass in a particular location. Indications might tell the operator the pressure and temperature at a location, but do not directly convey the mass. Systems that perform real-time thermo-hydraulic balancing can approximate and represent such information to operators and may aid in the reliability, safety, and efficiency of industrial process control.

However, despite these advances, having more data is not always beneficial. DCSs and supervisory control and data acquisition systems (SCADAs) have greatly increased access to available data and potential alarms, but the information processing capabilities of operators have not changed. Consider configured alarms, which have increased exponentially from the 1960s to the 2000s. The result is that even when processes are in steady state alarms will sound, providing little to no informative value to operators and potentially hindering their ability to attend to the process itself [3]. Interfaces need to capitalize on advances in digital control by providing operators with data in context instead of simply providing greater quantities of data. Rasmussen and Vincente [4] describe data in context as "the goal to make the invisible, abstract properties of the process (those that should be taken into account for deep control of the process) visible to the operator."

Knowing how to effectively accomplish this goal of presenting information to operators is part art and part science. Current human factors engineering (HFE) is often more diagnostic than prescriptive. It offers normative approaches to evaluating interfaces (e.g., NUREG-0700 [5]) and guidelines specifying what interfaces should do (e.g., they should support an operator's mental model of a physical system or they should provide information relevant to users), but falls short when it comes to explaining *how* such feats can be accomplished. There are many contributors to this problem. Real-world industrial processes are often one-of-a-kind and require extensive training to understand and operate. Cognitive systems engineering (CSE, [6]) proposes a framework for understanding and evaluating the rules, skills, and knowledge operators use to process information. In conjunction with Ecological Interface Design (EID, [4, 7, 8]) a "triadic" approach to interface design can be established as one tailored to specific work demands (domain/ecology), leveraging perception-action skills of the human (human/awareness), and using interface technologies wisely (interface representation, [9]). All three of these components are necessary conditions for the success of an interface. Unlike traditional user-centered design, EID focuses on making present the constraints of the physical system.

Pragmatic approaches to design such as Tufte's [10] aesthetic approach or Hollifield, Oliver, Nimmo, and Habibi's [11] experience-driven *High Performance HMI Handbook*, or Halden Reactor Project's perceptually oriented Information Rich Design [12] also offer significant value to perspective practitioners, although they may forgo much of the theoretic and empirical rigor of approaches like CSE and EID. Pragmatic guidelines will likely lead to resilient interfaces when applied in well-defined contexts; however, when context changes, pragmatic approaches may fall short as they provide knowledge on *how* but not necessarily *why* a solution works. The primary downside to CSE and EID is that the approach is viewed as academic and overly theoretical. A potential interface designer might think they need to first obtain a PhD in Cognitive Psychology and then ponder the meaning of “meaning” before venturing into actual design work.

# 3. TESTING ENVIRONMENTS

The goal here is not to present a treatise between basic and applied perspectives. Bennett & Flach [9] have already provided keen analysis on these issues. For HMIs to progress, a theoretical understanding of how operators perceive, understand, and act on information is needed. This understanding must capture the full complexity provided by real-world work domains. Unfortunately, conducting field studies is not always possible or practical: system and organizational complexities hinder experimental control, using a full-scope simulator with fully qualified operators provides external validity but at a significant financial expense, and lastly the entire population of qualified operators for a real-world process may be insufficient to draw statistically significant inferences. The alternative is to use *microworld* simulation environments to capture the essential characteristics of these real-world problems. By examining *in vitro*, confounding variables can be eliminated and underlying scientific principles can be discovered [13-15]. Then findings can be applied and tested with full-scope simulators, and eventually validated with real-world systems.

In this context, a prototyping environment was envisioned that would enable development of HMIs ranging a full spectrum of applications—from standalone microworld simulators used in a workstation environment to an HMI that uses the picture-in-picture functionality of a full-scope glasstop simulator like the U.S. DOE's Human Systems Simulation Laboratory (HSSL) at the Idaho National Laboratory (INL). Figure 1 shows an example of how the prototype DCS integrates into the glasstop panels in the HSSL, appearing as a virtual DCS in the form of a digital screen amid virtual analog instrumentation and controls (I&C).


**Figure 1. Example of a prototype DCS embedded in the soft panel control boards of the HSSL.**

Figure 2 illustrates the architecture of two prototype configurations—one for a standalone microworld and one for an integrated DCS. Two characteristics differentiate a standalone microworld from an integrated DCS. First, a standalone microworld typically represents a fictitious and simplified control process, whereas an integrated DCS is a simulation of an actual control process. Second, typically the DCS provided by a standalone microworld represents a simplified process in its entirety, whereas an integrated DCS is one of many interfaces an operator must use to interact with the system.

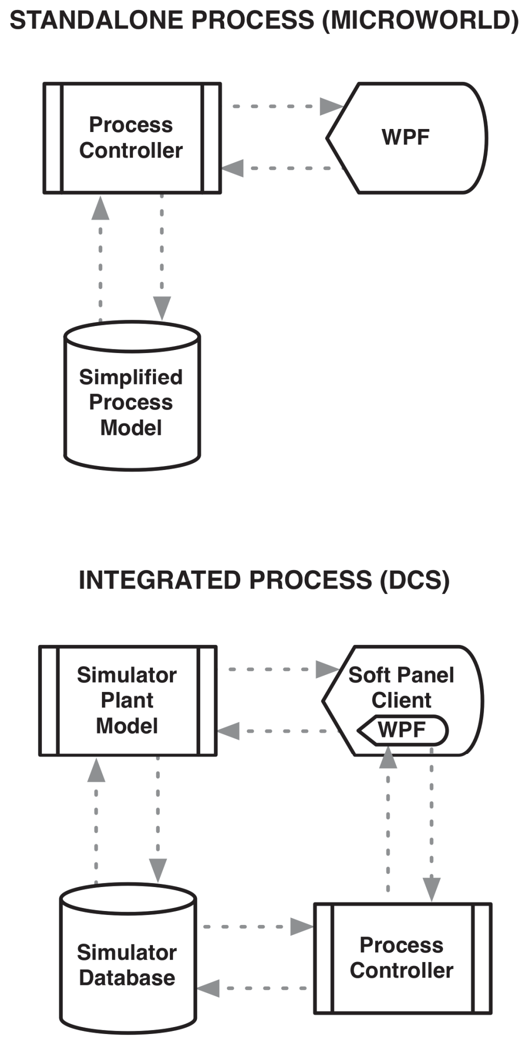


**Figure 2. Framework for a prototype that functions as a standalone microworld (top) or as an integrated DCS (bottom).**

# 4. WINDOWS PRESENTATION FOUNDATION

We use Microsoft's Windows Presentation Foundation (WPF) to develop prototype HMIs. We believe WPF is an overlooked and underused tool in HMI research. WPF provides a tremendous amount of value, reliability, graphical sophistication, and scalability to the implementation of HMIs for process control. With WPF, we quickly develop dynamic DCS prototypes that are independent of any particular DCS vendor or simulator software. These prototypes can be integrated into larger systems for more robust testing and assessment. The benefits of this approach are enumerated below.

1. **WPF is a mature, off-the-shelf tool that is easily configurable with both microworlds and full-scope simulators.** WPF is built on Microsoft's .NET framework and utilizes their Visual Studio integrated development environment (IDE). The large user base and broad deployment has allowed Microsoft to devote a tremendous amount of resources to developing an extensive and reliable suite of controls and programming libraries. WPF inherently separates the presentation or *view* of an application from the underlying model. This analogy is easily adopted by both microworlds and full-scope prototypes. Microworld models can be implemented in any .NET language (C++, C#, Visual Basic, IronPython, F#). WPF applications can also communicate with full-scope simulation models using standard protocols such as TCP/IP sockets, Object Linking and Embedding for Process Control (OPC), or through vendor provided .NET or network interfaces.

2. **WPF does not require expertise in vendor DCS development platforms.** Implementing novel displays and controls in vendor DCS solutions is not always possible or may entail specialized knowledge of and skills with proprietary software. Obtaining the requisite skills and knowledge may even pose difficulty or significant expense. In contrast, WPF is Microsoft's de facto development environment for building native Windows applications. Unlike proprietary process control environments, development pertinent information is readily available as is a cadre of skilled individuals. In practice, this translates to a common tool set for developing microworlds, EIDs, and full-scope prototypes. Elements developed and tested in microworlds can be up-cycled into full-scale process control prototypes.

3. **WPF is accessible to non-programmers.** Within the WPF architecture, the visuals can be defined by extensible application markup language (XAML). The markup is XML compliant (like HTML) and can be coded or produced using graphical design tools like Microsoft Blend or Visual Studio. Custom user controls and objects can be developed and reused across applications. This inherent flexibility allows for non-programmers to design and mockup interfaces that can be used by real applications. Custom user controls can also be developed and tested independent of larger applications. This would allow less experienced software developers, junior engineers, or students to work on components independent of the complexities involved with communicating with full-scope process simulators.

4. **WPF allows the same design to be reused in multiple environments.** The use of the graphical HMI front-end of WPF also translates into reconfigurability of objects and controls. As such, the look and feel of one proprietary environment (e.g., Emerson/Westinghouse Ovation) can be exchanged with another (e.g., Honeywell Experion) simply by changing the style properties of the objects. This allows easy prototyping flexibility for different needs and applications. Changing the look and feel of the objects and controls facilitates HMI research, as aspects of the design can be readily and easily modified to validate and optimize visual elements.

5. **We are developing a library of reusable objects that makes subsequent prototypes simpler to build.** WPF essentially becomes a library of objects and controls (e.g., valves) that can be readily reused or adapted for future rapid prototyping purposes. It is important to note that the objects defined

as part of WPF include not only graphical properties but also behaviors. Simple logic can be built into each object as parameters. For example, a valve may be defined as having an open or closed position. The valve position may be driven by external data sources such as the OPC database used by most full-scope simulator vendors. This relationship may be bidirectional—a change in the valve object in the HMI may change the value in the plant model via OPC, or change in the environment (e.g., automatic valve closing in response to abnormal conditions) may change the status of the object. As our library of objects and controls has grown with use, we have successfully created a standardized set of HMI elements that can interface with internal simplified prototype simulations to full-scope nuclear power plant models.

Note that we are not advocating the adoption of WPF specifically, as other software development platforms would provide many if not all of the same benefits. The key to this approach is developing prototypes outside of any particular vendor platform so that prototypes are reconfigurable and modifications can be implemented with ease. We have found that WPF is well-suited to this purpose.

# 5. PROTOTYPE EXAMPLES

Using WPF, we have implemented four prototypes. These prototypes are integrated into larger systems—either a microworld or a full-scale digital simulator—to provide realistic context and feedback on the design as it will be used. These prototypes were developed for a variety of purposes: for HFE research in automation, for evaluating a new Computer Operator Support System (COSS), for testing a real-world DCS design, and for demonstrating new DCS displays that incorporate best practices from human factors research.

## 5.1 Microworld Small Modular Water Heater

Figure 3 depicts a microworld process control application intended primarily for academic research. The process simulates a resistive electric water heater. Downstream of the water heater is a pump in series with a diverter valve leading to separate accumulators that feed two process control supplies. To the top left is an alarm annunciator panel. The alarm setpoints correspond to the ticks on the right axes of the trend indicator displays. When a setpoint is reached, the corresponding alarm tile flashes, and an auditory alert is sounded.

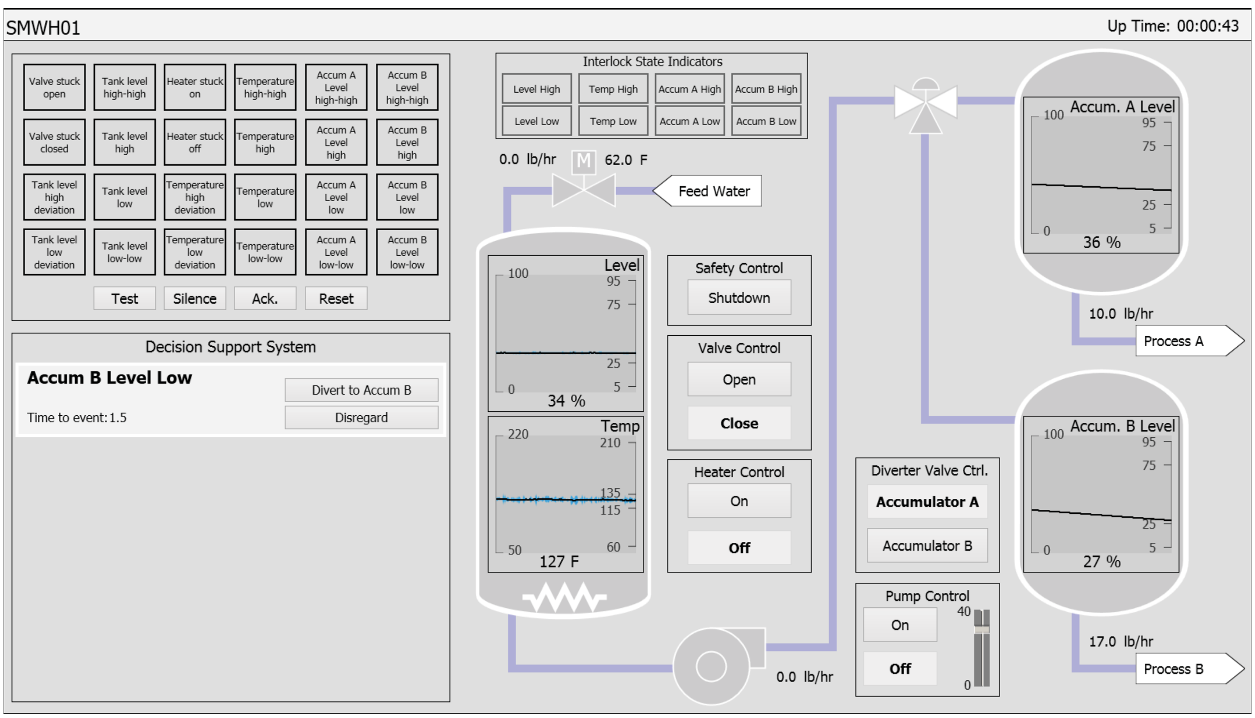


**Figure 3. Small Modular Water Heater microworld with Decision Support System. The SMWH is available under the GNU General Public License from the primary author.**

The microworld is a tool for cognitive research related to the implications of automation. It also serves as a testbed for examining and characterizing the perceptual properties of various presentations of information. As technology improves, automation is able to perform many tasks once assigned to human operators. Instead of actively engaging in control actions, operators take on a passive supervisory role by intervening only when necessary. A potential negative consequence of automation is reduced situational awareness. The microworld small modular water heater has been used to examine automation's impacts on human performance, workload, and situation awareness [16].

The microworld can run with three levels of automation [17]. At the lowest level of automation, the operator must monitor the process variables and control the feedwater valve, pump, and diverter valve to maintain the system within operating parameters. At the highest level of automation, the system controls the valve and pump states. The operator is tasked with ensuring the system stays within bounds and shutting down the system if any of the low-low or high-high thresholds are reached. An intermediate level of automation monitors the value and rate of change of each process variable. When it detects that a variable is about to fall out of bounds, it will prompt the operator with a decision support dialog and a soft control to initiate the appropriate action. All levels of automation also have a safety interlock system that intervenes at the low-low and high-high alarm thresholds to ensure the operator (participant) completes the trial.

Future versions will require a single operator to simultaneously monitor and control several water heater displays, a feature that lends itself to research related to the multi-unit small modular reactors (SMRs) currently under development.

## 5.2 Computerized Operator Support System Prototype

A COSS is a collection of technologies to assist operators in monitoring overall plant performance and making timely, informed decisions on appropriate control actions for the projected plant condition. A prototype COSS for monitoring and controlling the Chemical and Volume Control System (CVCS) of the 3-loop Generic Pressurized Water Reactor (gPWR) was constructed in support of the DOE Nuclear Energy Enabling Technologies (NEET) program [18]. The gPWR full-scope nuclear plant simulator was licensed from GSE Systems, and the control displays have been tailored to fit the bays using GSE Systems' JADE (Java Application Development Environment) software toolkit.

The prototype incorporates four underlying elements consisting of a digital alarm system, a computer-based procedure (CBP) system, piping and instrumentation diagram (P&ID) system representations, and a recommender module for mitigation actions (see Figure 4). The COSS CBP mimics the functionality of the Westinghouse two-column procedures and incorporates soft indicators and controls (see Figure 5). Unlike a purely language based description, the WPF prototype conveys concepts in a fully interactive and graphical manner. Concepts in the COSS could apply not only to control room modernization of existing light water reactors, but also to SMRs or even Next Generation Nuclear Plants (NGNPs).

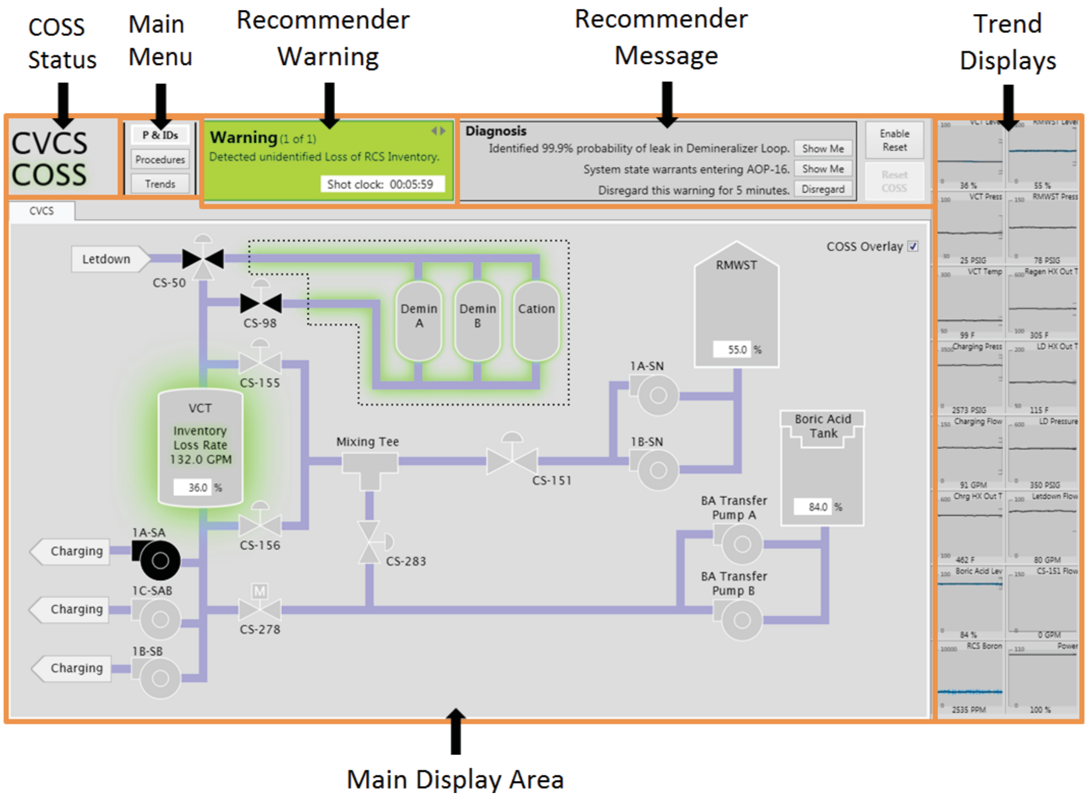


**Figure 4. Annotated COSS display featuring areas of concern highlighted on the P&ID, a recommender warning and suggested mitigation action messages.**

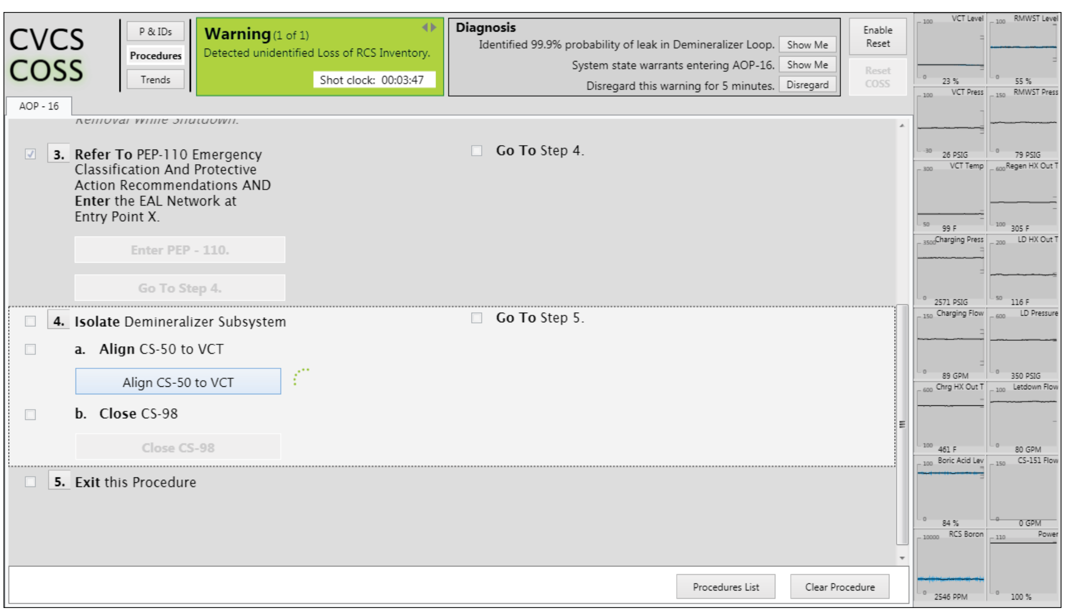


**Figure 5. Computerized Operator Support System (COSS) with Computer Based Procedure (CBP) prototype implemented in WPF.**

## 5.3 Emulating Honeywell Experion For Formative User Evaluation

Rapid prototyping with WPF may also have a niche in formative evaluation of HMI prototypes intended for real-world settings. The flexibility of WPF allows researchers to emulate existing vendor DCS solutions for early stage prototyping and user testing. Real DCS solutions are designed to operate with physical systems, and additional layers of complexity may be required to integrate them into simulated environments. Implementing early stage prototypes in WPF enables formative expert evaluation and user testing well in advance of when logistics might allow a DCS system to interface with a virtual programmable logic controller (PLC), plant model, and full-scope simulator. Furthermore, WPF is divorced from vendor and controller constraints. When potential issues arise, alternatives can be evaluated before proceeding with tedious change orders. Under contract with a major U.S. utility, the INL's HSSL has deployed a WPF prototype emulating the look-and-feel of Honeywell Experion [19]. The prototype (see Figure 6) has two-way communication with the simulator and integrates into a full-scale, full-scope glasstop control room simulator (see Figure 7).

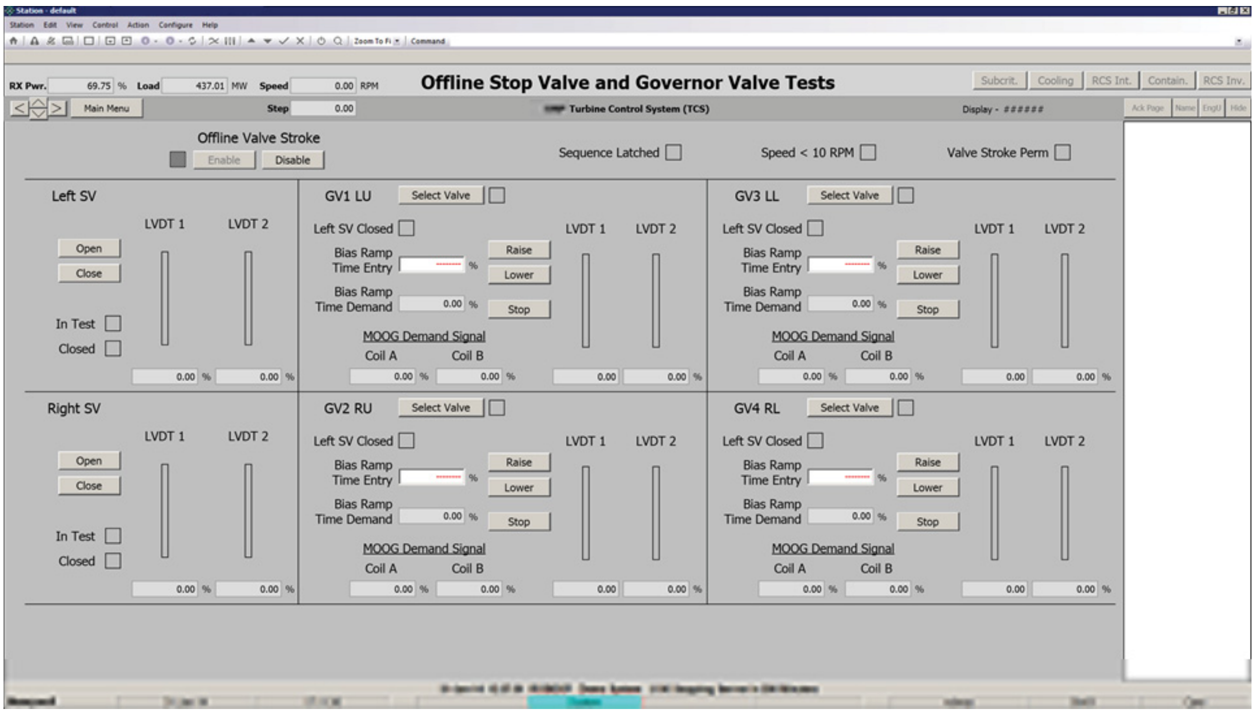


**Figure 6. WPF emulating the "look and feel" of a Honeywell Experion system.**

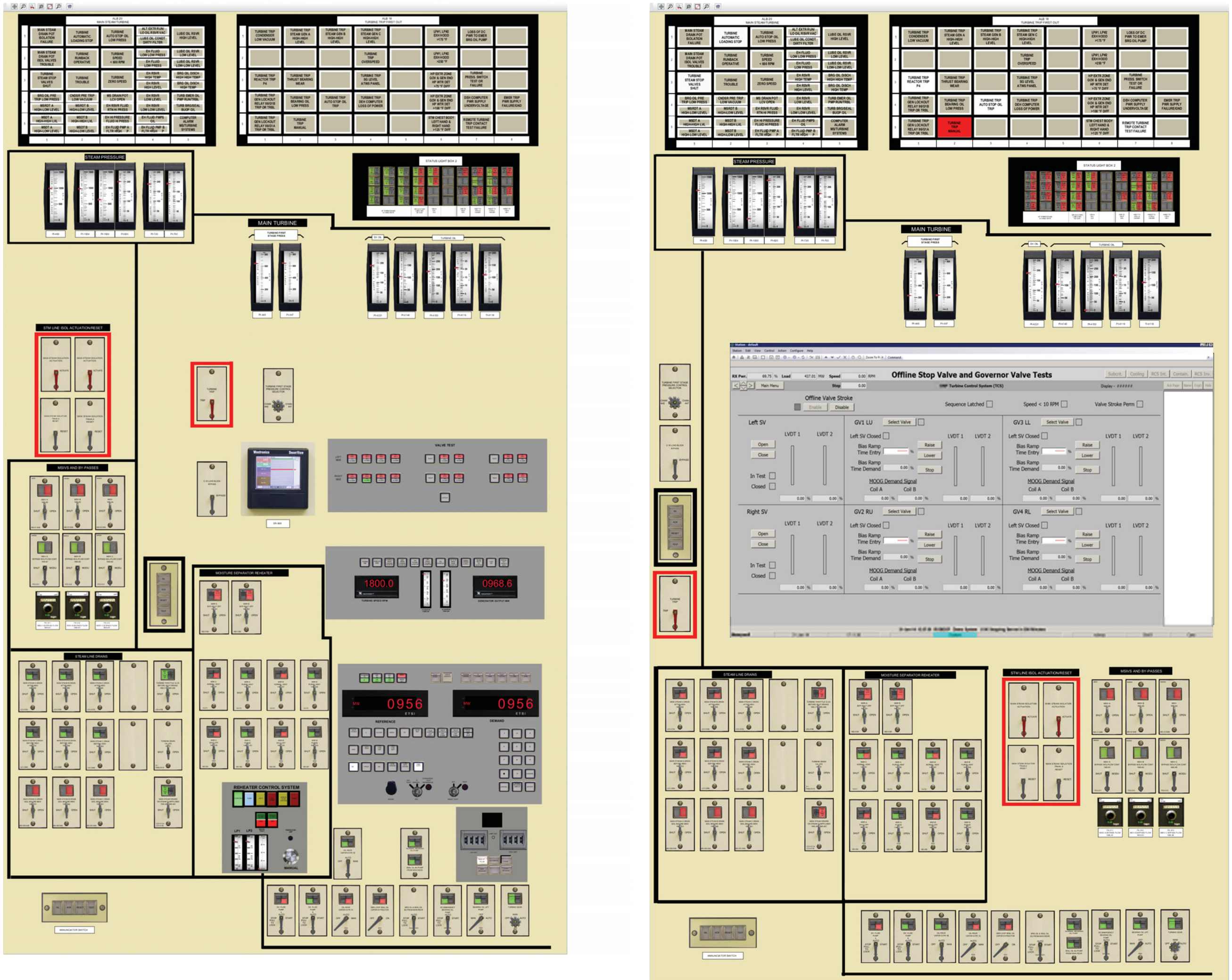


**Figure 7. Example of legacy turbine control system computer (left) updated with at turbine DCS (right) on the glasstop panels of the gPWR.**

## 5.4 Advanced Turbine Control System Interface Prototype

In support of the U.S. DOE's LWRS Program, a prototype turbine control system (TCS) for a multistage turbine and 3-loop pressurized water reactor (PWR) is currently in development (see Figure 8). The turbine control system is implemented in WPF and communicates with a full-scope simulator of a nuclear power plant (NPP) developed by Western Services Corporation (WSC) using a DLL interface provided by WSC. The prototype serves as a test-platform for evaluating "advanced" display formats against typically available presentation schemes. The Advanced TCS prototype presents features and displays that go beyond what would be considered current practice in DCS control and visualization. With sufficient time and diligence, any advanced HMI features and visualizations one would reasonably expect to accomplish can be implemented.

The TCS prototype is envisioned as a retrofit to an existing PWR. The monitor would be mounted at the board and is intended for a high resolution display with a 16:10 aspect ratio. Because the DCS would potentially be at the Main Control Board (MCB), critical values (such as reactor power, turbine speed, and load control variables) are intended to be visible at a distance of 20 feet. The top row of the display contains an alarm annunciator panel. Centrally located is a Level One overview. This overview presents the operator with the most critical information pertaining to the current state of the turbine/turbine protection system/turbine control system, the primary system supplying the turbine, and the electrical grid. The bottom third of the display contains a tab controlled interface to allow the operator to select information appropriate to particular operational modes such as "Load Control" as depicted in Figure 8.

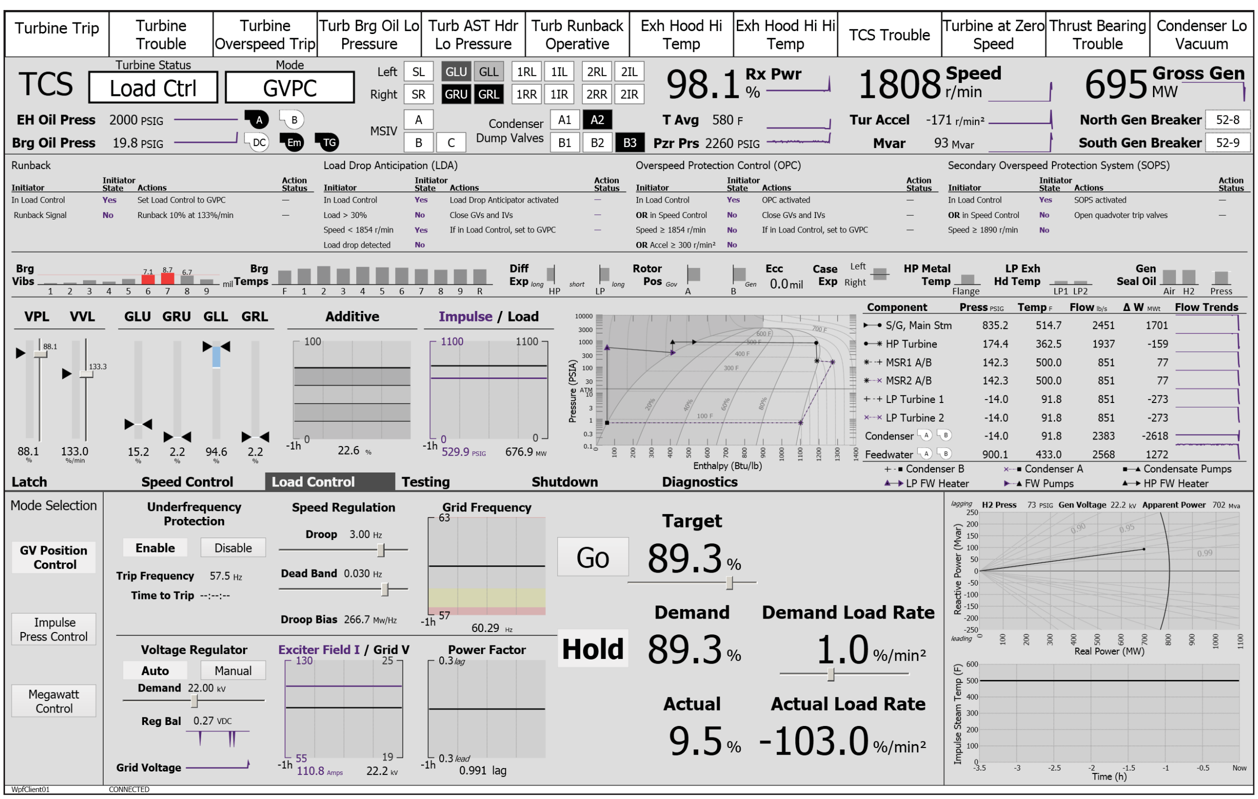


Figure 8. Turbine Control System Prototype with full-scope NPP backend.

The design encapsulates a variety of influences. The overview layout of the display is intended to provide a graphical Information Rich Design (IRD) overview of the most pertinent operating parameters, allowing operator-at-a-glance diagnosis and maintenance of overall situation awareness [12]. Hollifield et al. [11] emphasize that HMIs should be designed from P&IDs but not look like P&IDs. A review of TCS interfaces suggests that the vast majority of existing designs fall into this design scheme. The arrangement of valves in relation to the high and low pressure turbines is well ingrained in any licensed turbine operator and does not need to be explicitly displayed. What would normally be an entire display can be condensed into the digital "light box" below the annunciators. Tufte [10] advocates the benefits of high density information graphics as "intense, simple, word-sized graphics." Tufte's influence is revealed in the information dense row of supervisory bar graphs, and the auto-scaling mini trend lines (sparklines). CSE provides insights to human cognitive processing and decision making, and EID provides the rationale for why displays should present the engineering constraints rather than the task constraints. Essentially, a good interface should be akin to directly observing the movement of a tourbillon watch through the eyes of a master watchmaker; no essential truths pertinent to the operation of the timepiece are concealed. Similarly, no key indicators are obscured from the operators in the design. Several EID displays are implemented throughout the interface, including a dynamic pressure enthalpy display of the entire Rankine heat cycle of the turbine system, and the dynamic generator capability curve located in the load control tab.

(This page intentionally left blank)

# 6. SUMMARY AND CONCLUSIONS

We have found the WPF environment to be a powerful tool for design and development of digital HMIs and are actively using it in conjunction with operator-in-the-loop studies to validate design concepts and verify operator performance when using new HMIs [18]. Note that our purpose is not to endorse a particular software development platform (e.g., WPF), as we believe other development environments will likely prove equally suitable for HMI prototyping.

We have presented four examples of how WPF can be used for process control as an illustration of the many possible applications of rapid prototyping for HMIs. Our prototypes have been used to assess the effectiveness of different approaches to control, to provide a dynamic and integrated example of novel approaches to process control, to test DCS designs that will be implemented in currently operating NPPs, and to illustrate the broader possibilities of digital control HMIs.

The ability to demonstrate functional prototype designs for control room modernization has proven particularly useful as industry works to modernize control rooms with new technologies that will allow for continued safe and reliable operation. Evaluating DCS systems prior to implementation is a powerful way to reduce the need for do-overs and to ensure that replacement HMIs function as expected.

We claim that HMI design is both art and science. One does not become an artist by reading about art but by creating art. Our prototype environment allows us to manifest our ideas in pixels. Developing HMIs for process control has transformed us from mere HFE theoreticians to a team of prescriptive HMI designers.

We are admittedly newcomers to process control HMI research and to WPF, yet we gauchely suggest that our meager experience—just shy of one year's work in this field—is a testament to the value of WPF for process control. WPF is a tool that has allowed us to make mistakes faster. Design is inherently iterative. Many ideas that seemed good on paper have turned out to be less than stellar once implemented. Being able to see displays change in a real-time response to transient events is much more powerful than relying on one's imagination to assess the dynamic characteristics and utility of a display. Being able to test fully functional prototypes using actual operators has ensured that our designs are both practable and usable.

# 7. REFERENCES


[1] F.T. Chandler, Y.H.J. Chang, A. Mosleh, J.L. Marble, R.L. Boring, & D.I. Gertman, D.I. Human Reliability Analysis Methods: Selection Guidance for NASA. Washington, DC: NASA Office of Safety and Mission Assurance Technical Report, 2006.

[2] J. Watts-Perotti & D.D. Woods, "How experienced users avoid getting lost in large display networks," International Journal of Human-Computer Interaction, 11, 26-299, 1999.

[3] B. Hollifield & E. Habibi. The Alarm Management Handbook. Houston: PAS, Inc., 2010.

[4] J. Rasmussen & K.J. Vicente. "Coping with human errors through system design: Implications for ecological interface design," International Journal of Man-Machine Studies, 31, pp. 517-534, 1989.

[5] J.M. O'Hara, W.S. Brown, P.M. Lewis, & J.J. Persensky. Human-System Interface Design Review Guidelines, NUREG-0700, Rev. 2. Washington, DC: U.S. Nuclear Regulatory Commission, 2002.

[6] J. Rasmussen, A.M. Pejtersen, & L.P. Goodstein. Cognitive Systems Engineering. New York: Wiley, 1994.

[7] K.J. Vicente & J. Rasmussen. "The ecology of human-machine systems II: Mediating 'direct perception' in complex work domains," Ecological Psychology, 2, pp. 207-249, 1990.

[8] K.J. Vicente & J. Rasmussen. "Ecological interface design: Theoretical foundations," IEEE Transactions on Systems, Man and Cybernetics, 22, pp. 589-606, 1992.

[9] K.B. Bennett & J.M. Flach. Display and Interface Design: Subtle Science, Exact Art. Boca Raton: CRC Press, 2011.

[10] E.R. Tufte. The Visual Display of Quantitative Information. Cheshire, CT, USA: Graphics Press, 2001.

[11] B. Hollifield, D. Oliver, I. Nimmo, & E. Habibi. The High Performance HMI Handbook. Houston: PAS, Inc., 2008.

[12] A.O. Braseth, T.A. Øritsland "Visualizing complex processes on large screen displays: Design principles based on the Information Rich Design concept" Displays, 34, pp. 215–222, 2013.

[13] D.E. Howie & K.J. Vicente "Making the most of ecological interface design: the role of self-explanation," International Journal of Human-Computer Studies, 49(5), pp. 651-674, 1998.

[14] B.P. Dyre, E.J. Adamic, S. Werner, R. Lew, D.I. Gertman, D.I., & R.L. Boring. "A microworld simulator for process control research and training," Proceedings of the Human Factors and Ergonomics Society 57th Annual Meeting, pp. 1367-1371, 2013.

[15] R. Boring, D. Kelly, C. Smidts, A. Mosleh, & B. Dyre, "Microworlds, simulators, and simulation: Framework for a benchmark of human reliability data sources," Joint Probabilistic Safety Assessment and Management and European Safety and Reliability Conference, 16B-Tu5-5, 2012.

[16] J. Oxstrand, & K. L. Le Blanc. Effects of Levels of Automation for Advanced Small Modular Reactors: Impacts on Performance, Workload, and Situation Awareness. INL/EXT-14-32639. Idaho Falls: Idaho National Laboratory, 2014.

[17] D.B. Kaber & M.R. Endsley. “The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task,” Theoretical Issues in Ergonomics Science, 5, pp. 113-153, 2004.

[18] K. Thomas, R. Boring, R. Lew, T. Ulrich, & R. Vilim. A Computerized Operator Support System Prototype, INL/EXT-13-29651. Idaho Falls: Idaho National Laboratory, 2013.

[19] R. Boring, R. Lew, T. Ulrich, T., & J. Joe. Operator Performance Metrics for Control Room Modernization: A Practical Guide for Early Design Evaluation, INL/EXT-14-31511. Idaho Falls: Idaho National Laboratory, 2014.