DISSERTATION

MODERNIZING AUTOMATION IN INDUSTRIAL CONTROL/CYBER PHYSICAL SYSTEMS THROUGH THE SYSTEM ENGINEERING LIFECYCLE

Submitted by

Trevor J Ault

Department of Systems Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Fall 2021

Doctoral Committee:

Advisor: Thomas Bradley

Susan Golicic
Bret Windom
Edwin Chong

ABSTRACT

MODERNIZING AUTOMATION IN INDUSTRIAL CONTROL/CYBER PHYSICAL SYSTEMS THROUGH THE SYSTEM ENGINEERING LIFECYCLE

The systems engineering process seeks to develop systems beginning from a need and ending with an operational system. The systems engineering framework is acknowledged as an effective tool for building complex systems, but this research seeks an expansion in scope and emphasis to include more detailed methods for managing, operating, and upgrading existing subsystems when they are challenged by obsolescence, functional degradation, and upgrades/commissioning.

System development from a blank slate is often the default for the systems engineering field, but often an individual subsystem (in this case studied here, the automation system) must undergo upgrades much sooner than the rest of the system because it can no longer meet its functional requirements due to obsolescence. Partial system upgrades can be difficult to conceive and execute for a complex industrial system, but the fundamentals of the system engineering process can be adapted to meet the requirements for maintenance of an industrial control/cyber physical system in practice. Cyber physical systems are defined as systems that are enabled by interactions between computers and physical systems. Computers and other automation components that control the physical processes are considered part of this system. This dissertation seeks to engineer industrial automation systems to enable identification of obsolescence in cyber physical systems, simulation testing of the automation subsystems before/during upgrade, and integrity testing of alarms and automation after completion.

By integrating some key aspects of the systems engineering approach into operations and maintenance activities for large-scale industrial cyber physical systems, this research develops and applies 1) novel risk-based approaches for managing obsolescence, 2) novel techniques for simulation of automation controls for fast commissioning in the field, and 3) an automatic alarm configuration engineering and management tool. These systems engineering developments are applied over the course of 5 years of continuous operation and 14 large upgrades to automation systems in the process industry (gas processing, chemical, power generation). The results of this application illustrate consistent improvement in the management, upgrading, and engineering of industrial automation systems. Metrics of system performance used to quantify the value of the proposed methodological innovations include commonly used metrics such as number of alarms, cost, and schedule improvement.

For the research contribution which develops novel obsolescence identification and replacement strategies, the results show that using a modified risk management approach for automation and cyber physical systems that can quickly identify components that required upgrade. The results indicate a reduction of roughly 70% of reactive replacements due to obsolescence after the major upgrade and a 24% reduction in unplanned downtime due to part failure during normal operations.

For the research contribution illustrating that automation system simulation can confirm that the upgraded subsystems meet functional requirements during upgrade on continuously running sites, results are similarly positive. A new metric is developed to normalize the cost of simulations per system which measures the amount of simulation inputs (I/O) divided by cost. Results show that using the proposed simulation tools can reduce the cost of simulation by 40%

on a normalized basis and reduce alarms for a system by 55% during system startup and early operations.

Lastly, an audit system was developed for the automation systems to ensure that the subsystem continued to meet functional requirements after the upgrade. Deploying the audit system for alarm configuration was successful in that it resulted in no unauthorized alarm changes after the subsystem upgrade. It also resulted in improved alarm performance at sites since causes of alarm deterioration were eliminated. Results show that these added controls resulted in 52% fewer alarms (post implementation) and the elimination of alarm flooding (periods where more than 10 alarms occur in under 10 minutes).

The goal of this dissertation is to document innovative means to develop systems engineering towards operational and maintenance upgrades for industrial automation systems and to provide examples of ways this process can be applied. The values of the proposed engineering methods were validated through its application to over a dozen industrial sites of varying processes and complexity. While this research focused on heavy process industries, the process for identifying obsolete components and making major subsystems upgrades can also be applied to a broad set of industries and systems and provide research contributions to both the fields of industrial automation and system engineering.

ACKNOWLEDGEMENTS

I would like to first thank my advisor, Dr. Thomas Bradley, who has helped me achieve the goals of the Ph.D. program and helped me focus during the COVID pandemic. I would also like to thank my previous advisor Dr. Liuqing Yang. Both have encouraged me to be a life-long learner. After I'm done with the program, I hope to continually contribute to the field of systems engineering by documenting how automation interacts with systems. I would also like to thank my committee members, Dr. Susan Golicic, Dr. Bret Windom, and Dr. Edwin Chong who similarly encouraged and challenged me. As a professional automation/system engineer who has worked full-time during this program, I felt very welcome by my committee and the systems engineering department. I was able to pursue a graduate degree while continuing my career as a professional engineer. I personally believe the trend of professionals continuing to learn throughout their careers will increase. My committee has encouraged me and given me a positive experience.

I would also like to acknowledge my employer, Chevron, and my supervisor, Ghassan Sammour. I made it a goal to never overlap my Ph.D. with my job but my supervisor made me include my school program status in our career development plan and included conference attendance and publications in my training plan. After I used my GI Bill from service in the Navy, my employer financially supported me in the program.

DEDICATION

I dedicate this work to my wife Sawitree. She only made me feel slightly guilty if I worked longer hours than normal about something I am passionate about.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1 – Challenges with upgrades to automation subsystems

Systems Engineering in Heavy Industry

Systems engineering places a large emphasis in developing a system from concept to final system validation and testing. The process of following the systems engineering lifecycle is a key concept of the field of study (Sage). The issue is that often a system requires a large functional improvement due to aging facilities and components reaching obsolescence. An example of this is seen in chemical, energy, refining, and other heavy industries where the cost to build a new site is often prohibitive (either due to large capital investment or regulatory concerns). When this occurs, a large sub-system upgrade must be undertaken to either meet the new customers' system requirements or meet the original design requirements of the system if it has degraded. This can occur due to lifecycle degradation or obsolescence of components and subsystems. More often for industrial sites the system undergoes major expansion and renovation to upgrade the system. Roughly 70% of industrial investment goes to upgrading subsystems compared to building new systems (Labs), (Smith and Brelsford).

In addition to degradation of performance being a driver to do major subsystem upgrades, many refineries and power plants are upgrading to produce alternative and greener fuels to meet emissions goals and transfer to a greener (less carbon intensive) energy infrastructure (Othman). Similarly, it is often cheaper to upgrade existing sites (termed brown-field projects due to having existing infrastructure) compared to building new sites (termed green-field projects). Brown field projects for heavy industries are the more common systems engineering challenge for these industries. This dilemma stretches to other industries and development projects as well where analysis must be made to invest in a new system development or upgrade an existing one (Adams). The global chip shortage during 2020-2021 was blamed for the heavy capital investment to build new manufacturing sites. These are cost prohibitive, so few sites existed and no new sites were being built. Many factors led to the global chip shortage such as a major fire disabling one of the two main foundries and a surge in demand. But many researchers pointed to the large capital investment to build a new chip manufacturing site causing the main sites undergoing years of expansions via subsystem upgrades to increase their production only incrementally. The continual expansion of only a few production centers lead to a shortage of chips when demand spikes and it was difficult to keep expanding the sites. (Wu)

Systems engineering origins are in building aerospace systems from concept to operation (Shishko). In addition to supporting traditional systems' engineering industries, there needs to be an added emphasis on undergoing large system upgrades to existing sites since many times that is the case for systems' development. Figure 1 outlines the traditional system engineering lifecycle that this work hopes to propose variations on and tools to aid the major upgrades of subsystems. The purpose of this diagram is to outline the development of systems from concept to system validation. The left side of the "V" emphasizes the design from high-level down to detailed engineering tasks. The right side of the figure emphasizes testing and verification of the system from parts to overall system validation, ensuring that the system meets operational and functional requirements as first defined in concept development. The relationship between left and right of the systems "V" is meant to highlight those stages (seemingly far apart in development and system timeline) that are linked in engineering details and testing requirements as parts, components, and integration layers come together to form a complex system. Due to the unique challenges identified within the application of industrial controls, the typical process of starting from concept to development to operations must be adapted to identify obsolescence and perform major subsystem upgrades.

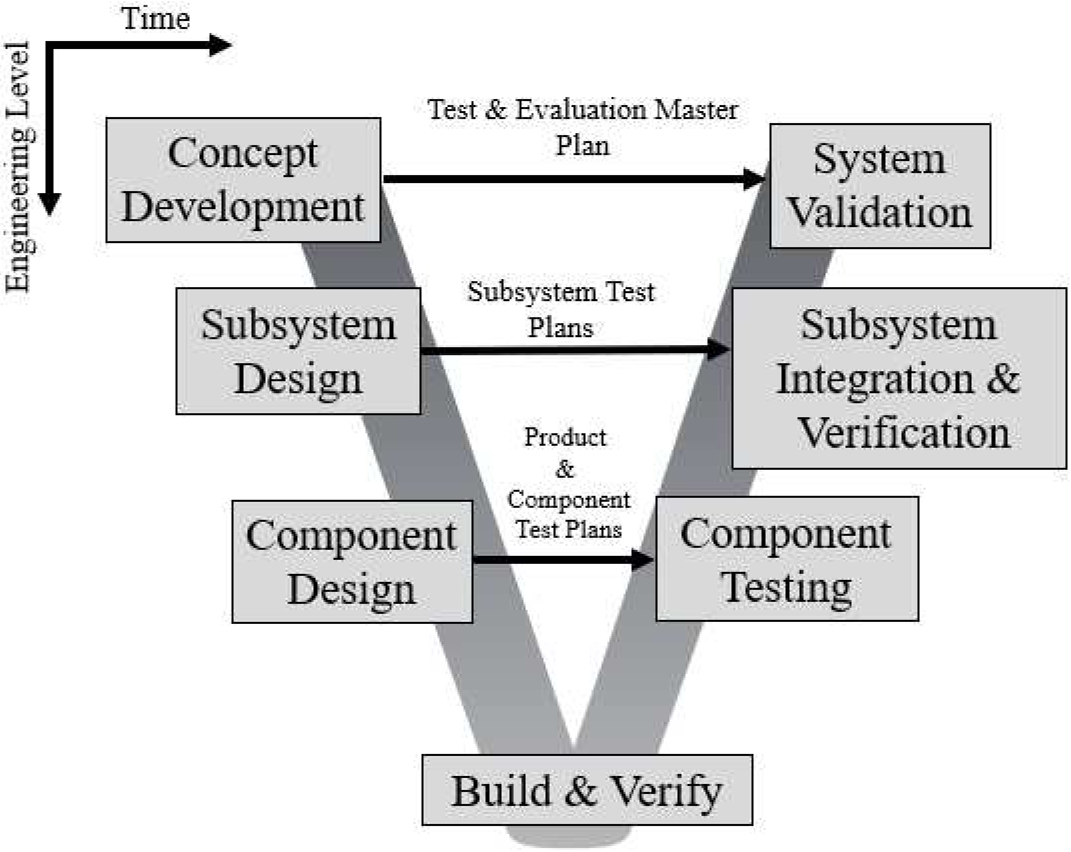


Figure 1: Simplified version of the canonical systems engineering "V" model

For many industries such as refining, chemicals, energy production, and other heavy process industries, the automation systems become obsolete much faster than the rest of the systems. Because of the end of life and product discontinuation by manufacturers, industries cannot to meet the same functional requirements, or they lack local resources to maintain their functional requirements. Because of these factors, automation systems are currently going through a period where continual upgrades are required much sooner than the rest of the systems that make up the site. The rapid development of automation technology has the unintended downside of making many COTS (commercial off the shelf) automation systems obsolete.

Industries are unable to interchange to new systems due to different programming languages and few people with skills to operate. This issue poses a threat to heavy industries (chemical, power, refining) to continually operate (Cesar). This poses a dilemma for the industries where many systems can last decades (like tanks, rotating equipment, electrical distribution) but the automation systems become obsolete and requires upgrades much sooner. Many complex systems such as refineries, chemical plants, nuclear sites, and other larger facilities are built once but must undergo multiple upgrades to subsystems throughout their lifespan. Another challenge is that long downtimes for the upgrades need to be avoided due to their facility's continual operation and any prolonged downtime has a large negative impact on revenue. Taking a chemical plant down for several months to upgrade the automation system is cost prohibitive (Torres).

Classical system engineering framework needs improvement for these specific cases and needs adaptation to account for these scenarios. This research focuses on extending lifespan of systems connected to cyber physical systems where normative stages of system engineering cannot be followed. It proposes a methodical way to identify obsolescence for automation, determining an appropriate level of simulation and testing prior to cutover to the new system, and maintaining an audit system for tracking the system to maintain integrity. The audit system ensures future upgrades are easier to identify all components in an automation system that can include thousands of instruments, controllers, and other automation equipment. Figure 2 below is shown as a typical site that was upgraded during this research. These sites have traditional equipment (non-electrical or non-automated) with unit operations, interconnected piping, control rooms that often undergo brownfield upgrades of system and most recently the automation system of this site was recently upgraded. By applying the techniques outlined in this research, the downtime and lost revenue can be minimized.

Figure 2: A typical heavy industry site illustrating a broad mix of process components

This work was done over several locations in southern California. Figure 3 below shows all the sites upgraded and if multiple upgrades occurred at one location. For example, some sites had major automation upgrades done for the gas plant, water plant, or other major facilities. Typically, these locations contained multiple major systems and, in many cases, more than one needed to be upgraded.

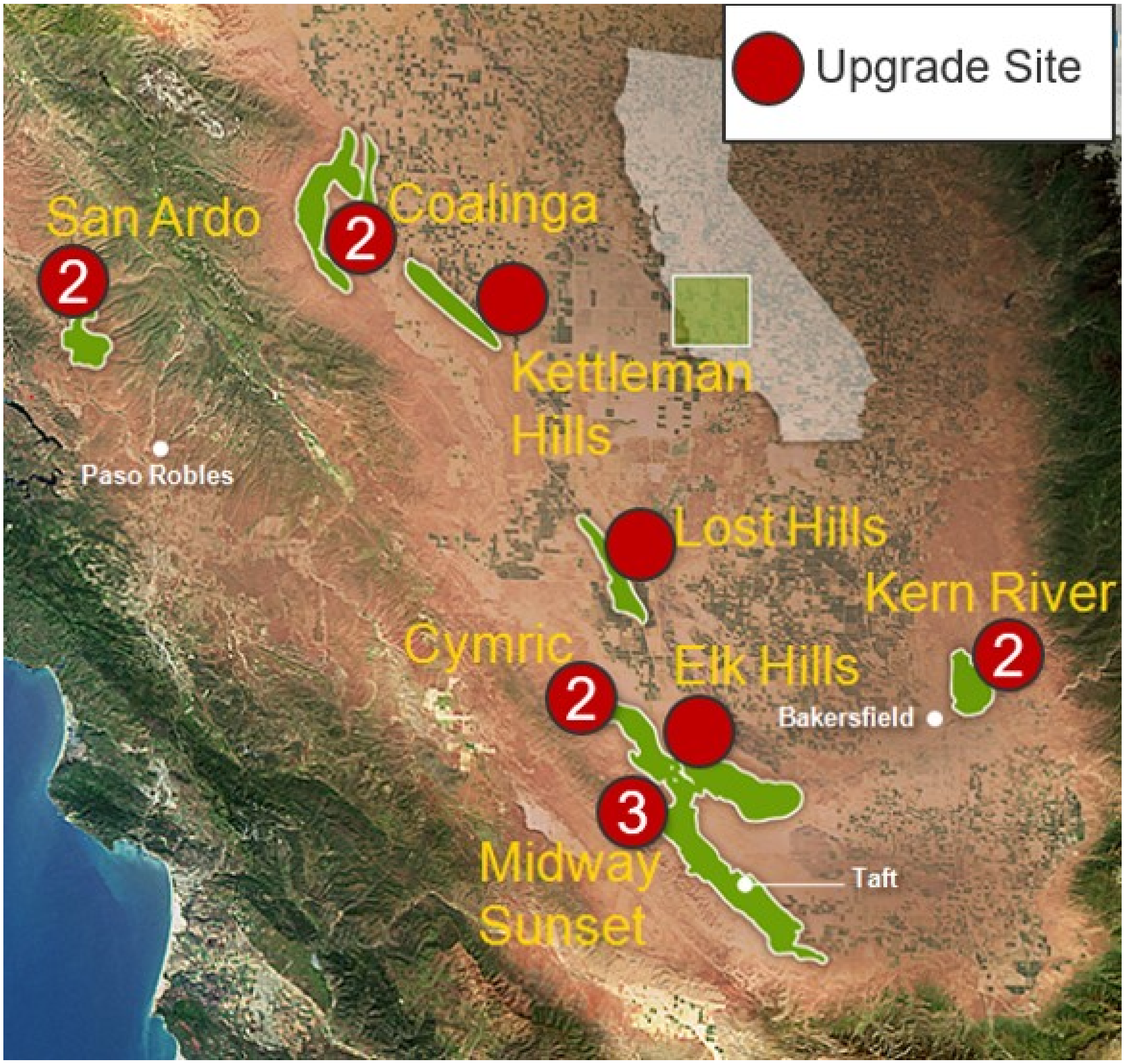


Figure 3: Map of locations and number of upgrades per site

Review of Obsolescence

The focus of this research is the series of projects completed to upgrade control systems for power, chemical, refining, and other heavy process industries. This challenge of methodically upgrading automation systems, testing, and confirming integrity extends into space, defense, avionics, and other sectors (Rojo). This research started with a risk analysis of several southern California process sites where it was identified the automation systems had reached obsolescence roughly 5 years ago. Since then, a program was developed and roughly 14 sites have been upgraded with work continuing today. This research proposes a way to solve the challenge of identifying when to do an obsolescence upgrade and what components to include in the scope of the obsolescence upgraded while fitting it into a turnaround cycle (planned shutdown). Figure 4 gives a high-level overview of how this proposed method fits into overall systems framework. This research's motivation and goal is to make special adaptations to systems engineering framework by developing tools and variations to methodology to aid in major subsystem upgrades.

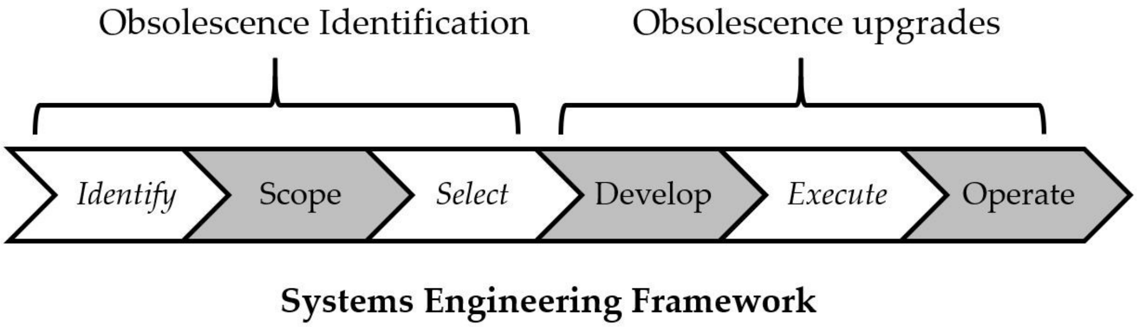


Figure 4: Obsolescence management in the context of systems engineering processes

The unique challenges for industrial sites made following traditional stages of systems engineering difficult. Subsystem upgrades are common in system engineering. A case study that was examined was the upgrade of aerospace systems (such as F-15 fighters) where they had major automation upgrades due to obsolescence. In fact, many aerospace systems undergo upgrades comparable to industrial sites when the system lasts longer, and owners wish to undergo upgrades compared to building a new system (Gee). In this case individual upgrades to major subsystems could occur in a hanger with no real time constraint as planes would be upgraded when not in operation, completely retested, and eventually returned to service. Industrial sites do not have this luxury. Each site is unique, and not being fully operational represents a large impact to revenue (Zurlo). Taking a site down and not returning it to service after a major upgrade within a day was considered unacceptable. This forced the development of strategies to mitigate downtime. This unique challenge may extend to other industries where complex systems cannot be taken down for long periods of time, but major subsystem upgrades must occur regularly.

Figure 5 shows a traditional bathtub curve that highlights that there can be rapid degradation of automation & electrical equipment at end-of-life. This figure emphasizes that it is often better to replace components in proactive phase before failure rates increase in reactive phase. This is done to limit unplanned downtime of the system. The purpose of showing the bathtub curve is that replacing parts during the proactive phase is difficult to assess. In reactive phase there is typically an economic justification for replacement. The failure rate of a device can justify its replacement from obsoleteness due to lost revenue or cost for repair. However, how should components be selected to for replacement during the proactive phase when the devices are not failing at a point where it becomes economically justified? To replace components during this proactive phase, different criteria must be used other than lost revenue or costly repairs. To do this, project databases were assessed to determine what parts were typically highlighted as obsolete in the reactive phase and why. Reactive obsolescence means the parts failure was causing lost revenue and therefore the upgrade to a modern component was solely justified based on economics. An example of reactive obsolescence would be an instrument that keeps failing and causing a loss of revenue. In those cases, the part or component was upgraded to a newer type solely based on the reliability and economics associated with it.

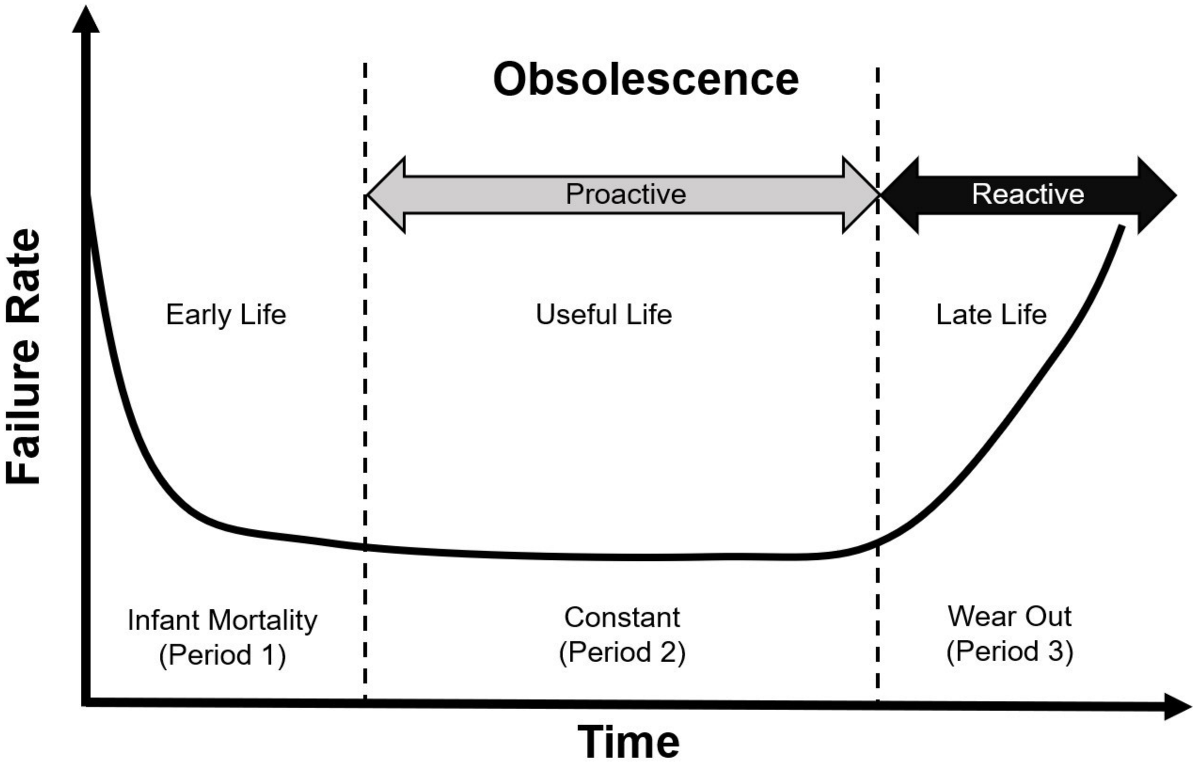


Figure 5: A canonical reliability "bathtub curve" illustrating the stages of failure rate

Lastly, another unique challenge during this research was the finite time frame to perform the upgrades. During a turnaround of the facility, the upgrade must be done efficiently to prevent large downtimes and therefore lost revenue. Figure 6 emphasizes this point showing a template for a system's upgrade roadmap showing upgrade dependencies between system upgrades, the timing for design work, and varying themes for the upgrade justification (compliance, functionality improvements, obsolescence management). This figure highlights another challenge faced by this work. This challenge is that major systems upgrades must occur often within windows of a one day to a few hours to perform the upgrade. For industrial sites, it is common to only have certain finite opportunities to do major upgrades due to continual operation of the system. This introduces several complications such as planning years in advance, considerations to long lead equipment, and inter system upgrade dependencies. This figure shows typical dependencies, multi-year timing, and varying requirements to be executed (compliance, new capabilities, base business). The key message is that for the certain special characteristics of industrial sites, traditional methods for systems engineering must be modified or considerations given to account for legacy subsystems reaching obsolescence and limited opportunities to do upgrades of those subsystems.

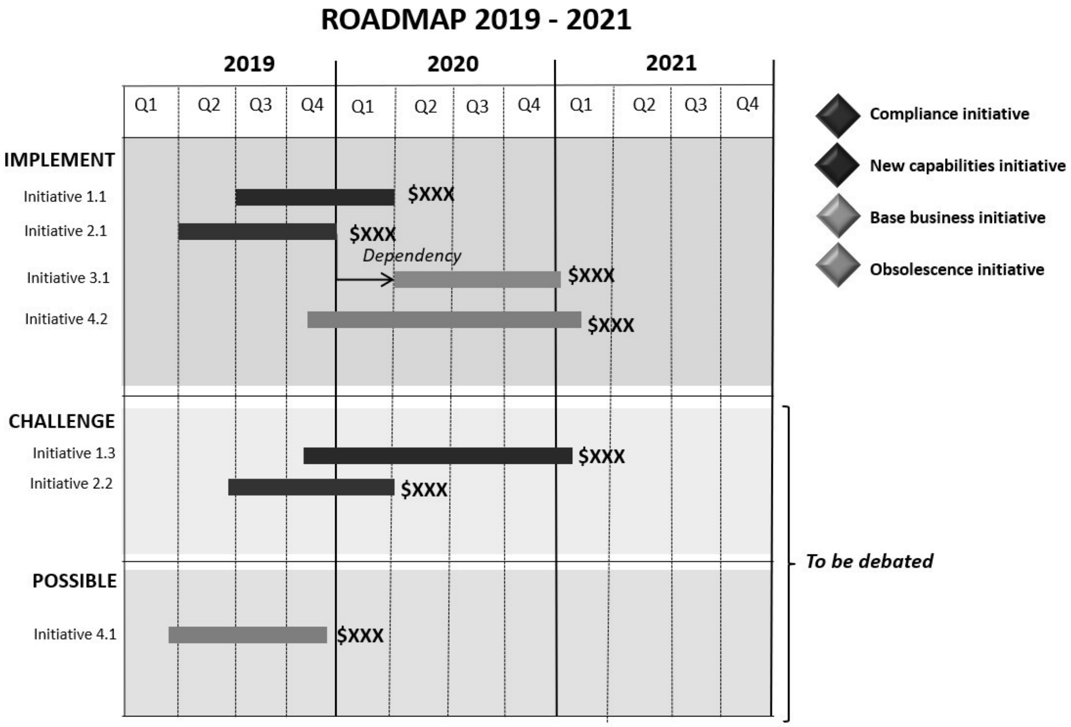


Figure 6: A sample systems upgrade roadmap for an industrial automation system

Lost revenue is directly tied to a system's shutdowns/turnarounds that occur in heavy industries. It is infeasible to do months of testing with the newly built system before restoring normal operations. So, another challenge with regards to this research and systems engineering framework is how to compress or do in parallel, the traditional systems engineering steps of component selection, testing, and system's validation and performance verification. In summary we can identify a problem statement to motivate this research:

- Problem statement: The Systems engineering lifecycle management framework is an excellent tool it must be adapted to systems where normative stages of the system lifecycle must be altered to the unique characteristics of this industrial controls application. Specifically, we can ask the following motivating research questions.
- How to quickly assess obsolete components to identify their upgrade and replacement for next turnaround cycle?
- How to test the subsystem upgrades efficiently without causing delays to system restart?
- What tools can we put into place after the upgrade to allow for easier subsystem and system upgrades for complex sites in the future?

This chapter continues by giving a brief overview of industrial automation to illustrate how its unique characteristics may hamper systems engineering development. Chapter 2 introduces industrial automation to explain the background of why this work is important in the context of the field of application. Chapter 3 provides a literature review of current work in this field and each of the subtopics, methods, and tools developed. Chapter 4 documents the development of an efficient method for identifying obsolete electrical/automation equipment by first developing a catalog of components, determining which factors lead to reliability failures, and proposing a risk management method for obsolescence. The results are documented after deployment at several industry sites. Chapter 5 outlines a method for controls checkout and simulation testing which is to test major automation upgrades prior to upgrade to ensure no errors in programming. Chapter 6 describes the development of an audit system to ensure the alarm system's integrity is maintained after the upgrade. Chapter 7 defines how these topics tie into the growing and broader field of digital twinning. Proposed future work which plans to extend the audit system to the rest of the process control network is also discussed. By continually adding features of the process control network, it shows how digital twinning can be achievable where many times it is seen as cost prohibitive. Chapter 8 delineates the contributes this research provided to modernize automation in industrial control/cyber physical systems. Also, this chapter includes some final thoughts on why subsystem upgrades are important for the energy infrastructure as legacy, conventional systems become more adaptable to lower-carbon, more sustainable systems.

Research Question Summary

This research addresses this obsolescence challenge by advancing systems engineering methods towards a methodical set of strategies to upgrade automation by identifying obsolescence and defining functional specifications, simulating results, and auditing after cutover. The summary of the background and research areas of this report are given below in Table 1.

In summary, with multiple industrial sites requiring a complex and unique subsystem upgrade, my research work's main objective was to find a simple way to identify obsolescence, ensure programming was able to transition to new systems quickly and seamlessly, and develop tools that could be left in-place after the upgrade to ensure that the integrity of the system could be maintained. The next chapter gives an introduction to industrial automation to detail further why these subsystem upgrades are uniquely challenging.

Table 1: Summary of report background and research areas

Research Topic	Modernizing Automation In Industrial Control/Cyber Physical Systems via System Engineering Lifecycle: Research into extending lifespan of systems connected to cyber physical systems		
Background and research area	Identified roughly 14 sites with major automation subsystems at end of life. How to upgrade complex systems (instruments, logic controllers, graphics, programming language) where traditional stages of systems engineering could not be followed?		
Literature Review	Review of obsolescence management of automation systems	Review INCOSE for single subsystem upgrades. I.e., what gaps are in current body of knowledge for extending life of systems but major upgrades for individual systems.	Review of similar major automation projects and techniques used to ensure specifications are met prior to cutover
Synthesis of knowledge	Trends show that many systems upgrades are done with existing sites. Systems engineering methodology is excellent for new projects but lacks techniques for systems upgrades to extend life and account for fields where technological improvement is rapid (automation, PCN security, etc.). These sites pose a unique challenge for system engineering and further research is warranted.		
Research Questions	With inability to know end of life for electrical equipment, can there be methods to risk rank obsolete equipment to identify for next upgrades?	Integrity testing of code needs improvement. What techniques can be used to simulate prior to cutover to improve safety and commissioning times?	How can we establish monitoring to improve identification of obsolete and unapproved changes to automaton systems?

Chapter 2 – Industrial Automation Subsystems

Industrial Automation Overview

Automation systems can be intimidating for people who experience them for the first time. Seeing the amount of communication cables and wires being pulled throughout a facility to connect from an instrument to an enclosure can make it difficult for people to connect the physical system to the automaton system clearly. The connection from the thousands of field devices, pumps, and other rotating equipment to the graphics in the control room can be difficult to understand. However, this complexity can be simplified. This section makes this complexity easier to understand. Automation and systems engineering share a future together due to interface with humans (control room), ability to link components and subsystems together, and both disciplines being necessary for complex and modern systems.

Other analogies are probably better, but when I give tours of facilities to people, I compare automation/cyber physical systems to a sprinkler system. There are field devices that do what you need them to do (sprinklers) with wires that connect back to a logic controller (main sprinkler valves) and a programmable logic controller (your sprinkler controller). Every field facility has an automation system. This is also true with air conditioning and refrigerators. If an instrument measuring a variable records a drop below a specified limit, the internal computer controls the process. But I typically start describing automation from the perspective of the sprinkler systems since many people have had to troubleshoot them by either replacing parts or reprogramming the controller. There is not a huge difference between configuring an industrial logic controller and setting up sprinkler system. For me personally, I would rather reprogram a compressor than deal with my sprinklers to give some perspective that automation is just a matter of familiarity. Over the decades, practitioners have tried to simplify automation by updating key concepts, hardware, and software.

Going from a sprinkler system to a large chemical plant is just a matter of complexity. After that, you can get convoluted with variable frequency drives, distributed controls systems, wireless technology, and other features. But even the most complex industrial site is just a glorified sprinkler system. The same goes for the computer system that makes up the plant and its software. Our phones have as much computing power as many refineries from decades ago (Nordhaus). What makes automation systems different from a laptop is only that they are designed to be stable (run continually for many years without crashing) and that is a key reason why they are developed independently from business/personal computers (Rullan). Typically for commercial off the shelf systems (COTS) they are designed to have their software work with their own hardware and not have any major faults (shutdowns) that might cause unplanned downtime at a site with the expectation they run for decades continually. Theoretically, you could run a refinery from a personally computer off a traditional programming language (such as FORTRAN or C++), commercial automation systems address many problems with this approach.

Firstly, with programming the system, COTS systems offer many templates and functionality to prevent complicated programming. In other words, many basic functions can be done with pre-built templates in the industrial automation software which allows for continuity and consistency in programming over decades of development and personnel turnover. Expandability of automation system also led to it being a separate field from personnel computer development. Modules were built to be interchangeable and expandable to assist with industrial applications. The number of wires being pulled to logic controller can change from supporting dozens of devices to thousands if one continually expands the network and the upgrades the programming/processor. And as previously mentioned, the operating system must be incredibly stable to ensure that running calculations, solving logic, and recording data can be done continuously for decades. If the logic controller froze or crashed as much as a cell phone or a personal computer, it would pose a serious risk to the operation of the site (Freeman).

In summary, industrial automation systems, typically where programming logic controllers (PLCs) are the main component, are similar to personal computers. But logic controllers and personal computers have to be developed separately due to:

- Continual scanning of field devices and real time determinism for process execution. PLCs scan devices continually whereas PCs are event driven.
- Cost vs. Performance: Industrial PLC cost rise exponentially with memory or programming power due to stability concerns
- PLCs are designed around robustness and reliability in execution and continued operation
- Standardized programming for commercial systems to prevent faults in system
- Functional isolation to ensure a dedicated control response. For example, PLCs have programming dedicated ensuring if an event occurs the PLC will address it in milliseconds (Kleines)

A Process Control Network (PCN) is an interconnected system used to monitor, control and automate a physical process. These systems are used nearly everywhere in the modern world, including utilities and transport. Similar to our nervous systems in our bodies, the PCN is a complex system of sensors (nerves) and actuators or valves (muscles). Actions are controlled by sensor inputs and an automated logic decision. The basic loop of an industrial process control system is to monitor (via sensors that measure flow, level, pressure, and temperature among other things), evaluate (via logic solvers and programming), and then control the system via control elements (Liptak).

The basic steps of an automation system are:

- The control system monitors input information from process, machine, or device.
- Controller evaluates input information based on given sets of rules and generates output information.
- Output information used to control process, machine, or devices.

This is performed by the following common PCN components (Dahm):

- Sensors and instruments which determine the current state of a variable, for example a temperature sensor.
- Programmable logic devices control actuators using logic, measurements and configured 'set points'. For example, an air-conditioner maintains a room's temperature using a sensor, a configured 'set-point' and a logic calculation.
- Distributed Control Systems (DCS) provide integrated interconnectivity throughout the PCN to enable centralized management. Supervisory Control and Data Acquisition (SCADA) systems provide localized visibility and control (of DCS subcomponents).
- A Safety Instrumented System (SIS) which is designed to safely halt the process if an unsafe state is identified. For example, a fire and gas detection system is designed to detect to a gas leak and alert operators.
- A process historian records 'point in time' instrument measurements for production accounting, regulatory reporting, fault diagnosis and many other tasks.

- A Manufacturing Execution System (MES) connects, monitors and tightly controls engineering and production operations (Almada-Lobo).

The basic components of a process control network are shown below (Figure 7). The most complicated automation system can be simplified into these 3 basic parts and components. A sensor records variables, the logic solver then does computations and sends signal to final element, and the final element controls the process.

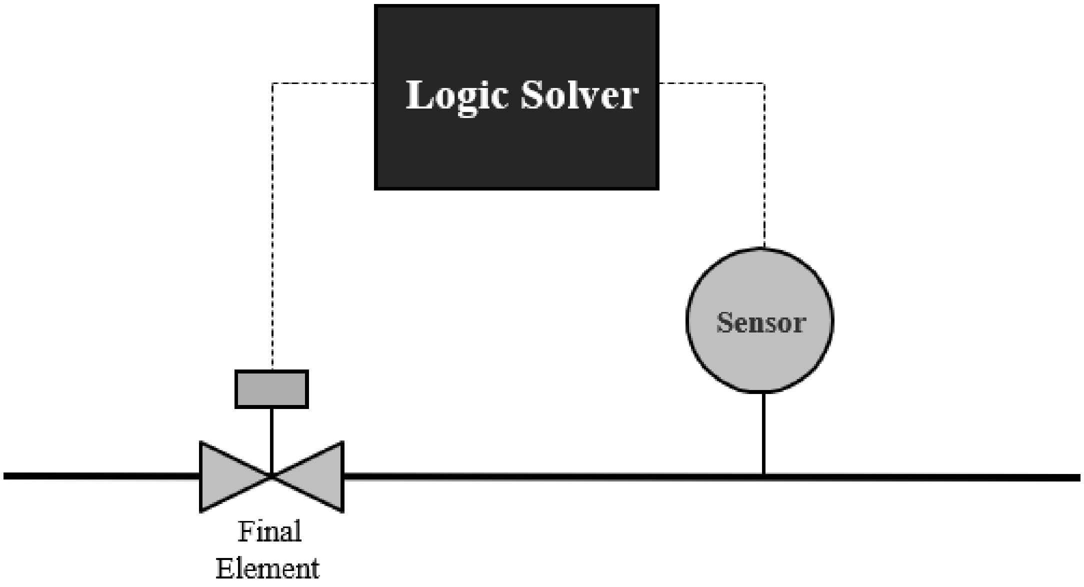


Figure 7: The basic components of a process control network.

The unit operations that make up the industrial site (distillation columns, vessel, tanks, etc.) all have instruments connected to devices that measure variables. The typical sensors and final control elements for industrial systems measure various types of variables that make up the system and are listed below (Liptak):

- Flow: For gas applications, differential pressure flow measuring techniques such as orifice plates, pitot tube, and Venturi tubes, or ultrasonic flowmeters (for flare gas) are common types. For liquid measurement, orifice meter, turbine meter, Coriolis meter, ultrasonic, and magnetic are common types.

- Level: Various types of level measurement methods are suitable for process (fluid) condition such as displacement, float, radar, and differentials.
- Pressure: Various types of pressure measurement methods are available such as pressure gage, pressure transmitter, and differentials pressure.
- Temperature: Various types of temperature measurement methods are suitable for process (fluid) condition such as RTD (resistance temperature detector), thermocouples, and liquid fills.
- Final Control Elements: Various types of level measurement methods are suitable for process (fluid) condition such as control valves, shutdown valves, blowdown valves, choke valves, deluge valve
- Miscellaneous: Various type of instrumentation and electrical components are suitable for process (fluid) condition and control functions such as heaters, solenoids valves, pilot valve, various type of indicators, various type of analyzer such as ORP (oxidation reduction potential), GC (gas chromograph), density, specific gravity, heat detector, UV (ultraviolet light) detector, IR (infrared light) detector, and gas detector.

For most industrial sites, production halts without each of the components listed above functioning. In addition to controlling the physical process, access to real-time production information is helping the industrial site by:

- Automating production accounting and regulatory reporting.
- Using remote support, which has access to

- Improving production performance and reliability through end-to-end process engineering (Spitzer).

The next major component of an automation system is the programmable logic controller (PLC) or programmable controller. It is a digital computer used for automation of electromechanical processes, such as control of machinery on factory assembly lines, amusement rides, or light fixtures. A PLC is an industrial digital computer which has been ruggedized and adapted for the control of manufacturing processes, or any activity the requires high reliability control. Typically, they are made up of modules for customization based on the size and design of the system they are controlling. PLCs are built to provide "real-time" interaction with the processes of the system and can take inputs and provide outputs based on the programmed logic to the process in very short duration (milliseconds) (Harris). Major components of programmable logical controllers are listed below (Netto):

- Power Supply
- Controller: brain of programmable control system
- Solid-state device, similar to a computer
- User-programmable memory and a central processor
- I/O (input/output) system: Electronic plug-in units used to interface with the input and output devices in the machine or process being controlled
- Input modules that receive data from input devices and send it to the processor.
- Output modules that receive data from the processor and send it to output devices.
- Programming system
- Communications network

PLCs can be programmed with a variety of functionality. Some of the typical functionality contained in a programming logic controller is voting for control. An example of this is two instruments can be 1 out of 2 voting to initiate a shutdown if a variable goes above a safe limit. Another is storing information with specific time data to alert which components tripped first, referred to as “First-out” logic. In a system containing “First-Out” logic, the first status change (alarm) is a sequence of events that is “tagged” or displayed first, making it possible to identify what device caused the initial alarm. When a PLC transmits its current status to the server, the "First-Out" annunciator allows the operator to discern which device tripped first. One abnormal condition can result in a sequence of shutdowns or status changes at the facility, "First-Out" can be a valuable troubleshooting aid. As an example, if a pipeline transmitter and a pipeline pump are monitored, it would be advantageous to know if the pump shutdown resulted from the pipeline transmitter tripping, or vice versa. This aids in isolating the problem; that is, whether there is a problem with the pump or the pipeline. Digital or discrete signals behave as binary switches, so called On/Off signal (1 or 0, True or False, respectively). Push buttons, limit switches, relay contacts, and photoelectric sensors are examples of devices providing a discrete signal. Discrete signals are sent using either voltage or current, where a specific range is designated as “On” and another as “Off”. For example, a PLC might use 24 V DC I/O, with values above 22 V DC representing “On”, values below 2VDC representing “Off”, and intermediate values undefined. Zero and 24VDC is an indication of instrument failure, broke wires, or other issues. That is why that the maximum range for current or voltage is not used because the extremes provide troubleshooting help. Initially, PLCs had only discrete I/O that only sent on/off signals to a logic controller. Analog signals can send a range of data to a

PLC. Analog devices are becoming the main devices at industrial sites due to their expanded functionality and ability to send internal diagnostics much easier (Wilson).

Analog signals are like volume controls, with a range of values between zero and full-scale. These are typically interpreted as integer values (counts) by the PLC, with various ranges of accuracy depending on the device and the number of bits available to store the data. As PLCs typically use 16-bit signed binary processors, the integer values are limited between -32,768 and +32,767. Pressure, temperature, flow, and level are often represented by analog signals. Analog signals can use voltage or current with a magnitude proportional to the value of the process signal. For example, an analog 0 - 10 V input or 4-20 mA would be converted into an integer value of 0 – 32767 (Liñán).

The below Figure 8 is an example of this (showing both discrete and analog sensors). This is an expansion of the simple control loop shown previously showing expanded functionality and methods to combine things like voting between instruments and fail-safe design. A pressure switch is an example of discrete or digital signal and pressure transmitter is an example of an analog signal to the logic controller. The discrete switch sends a signal (1 or 0) to logic controller if the process causes the switch to trip. The transmitter sends a 4-20mA signal which can be converted into a value in logic solver for the exact process condition. The two can work together to (in a voting a system) to open or close a valve to shut down the process when a safety threshold is violated. If the example in Figure 8, a 1 out of 2 voting (one instrument or switch signal causes valve to activate) then the “first out” signal described previously can help identify which system may be malfunctioning if a spurious trip occurs (D. X. Wang).

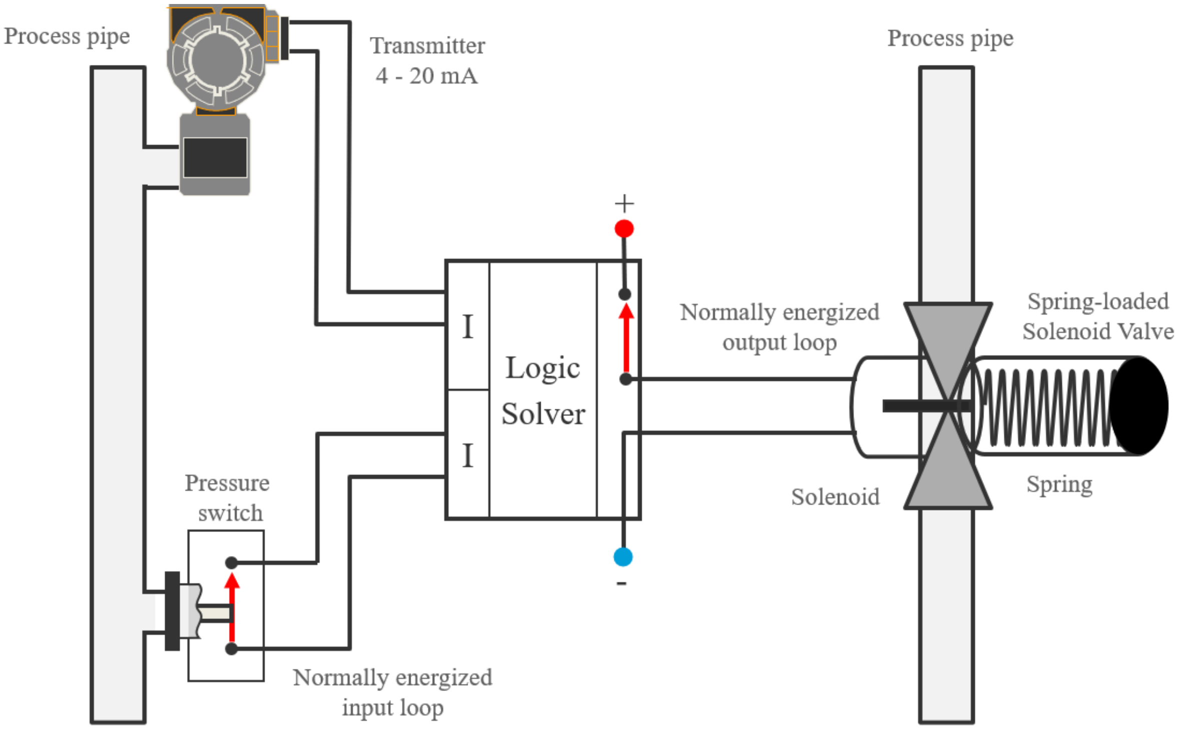


Figure 8: The basic components of a more complicated simple control loop

A PLC program is generally executed repeatedly as long as the controlled system is running. The status of physical input points is copied to an area of memory accessible to the processor, sometimes called the “I/O Image Table“ or “Tag Name Database”. The program is then run from its first instruction rung down to the last rung (Nedvěd). It takes some time for the processor of the PLC to evaluate all the rungs and update the I/O image table with the status of outputs. This scan time may be a few milliseconds for a small program or on a fast processor, but older PLCs running very large programs could take much longer (say, up to 100 ms) to execute the program. If the scan time was too long, the response of the PLC to process conditions would be too slow to be useful.

As PLCs became more advanced, methods were developed to change the sequence of ladder execution, and subroutines were implemented. This simplified programming and could also be used to save scan time for high-speed processes; for example, parts of the program used only for setting up the machine could be segregated from those parts required to operate at higher speed. PLCs may need to interact with people for the purpose of configuration, alarm reporting or everyday control. A human-machine interface (HMI) is employed for this purpose. HMIs are also referred to as man-machine interface (MMI) and graphical user interface (GUI) (M. a. Metzger).

A simple system may use buttons and lights to interact with the user. Text displays are available as well as graphical touch screens. More complex systems use programming and monitoring software installed on a computer, with the PLC connected via a communication interface. PLC programs are typically written in a special application on a personal computer, then downloaded by a direct-connection cable or over a network to the PLC. The program is stored in the PLC either in battery backed up RAM or some other non-volatile flash memory. Often, a single PLC can be programmed to replace thousands of relays. The IEC 61131-3 standard currently defines 5 programming languages for programmable control systems: function block diagram (FBD), ladder diagram (LD), structured text (ST; similar to the Pascal programming language), instruction list (IL; similar to assembly language) and sequential function chart (SFC) (Wareham). These techniques emphasize logical organization of operations.

While the fundamental concepts of PLC programming are common to all manufacturers, differences in I/O addressing, memory organization and instruction sets mean that PLC programs are never perfectly interchangeable between different makers. Even within the same product line of a single manufacturer, different models are often not be directly compatible.

From the PLC to the control room, the data typically goes through IO (input/output) drivers. IO drivers are responsible for the software connection between complex devices, PLCs, and control room operating system. In short, IO drivers are a way to standardize and communicate data between varying devices in field, logic controllers, and control room (P.-q. a.-w. Wang). Many times, beyond PLCs, there are smart meters and other complex devices at field level and the IO drivers do the job of connecting them together and converting to a common language for the control room screen to display as data, graphics, and other information to system operators. IO drivers and HMI displays is typically what constitutes the beginning of the SCADA network (Supervisory Control and Data Acquisition). SCADA, as the name implies, has a supervisor (often the operator) at a control room collecting data and acting on information that is received.

The following are major components that make up the SCADA system:

- Human Machine Interface (HMIs) aim at a better Human-machine interface. Any automation system is said to be blind without HMI. HMI gives the ability to the operator, and the management to view the plant in real time. Add to that the ability to have alarm management that can warn the operator of a problem. It can even log and print all the alarms in real time, which can help the management to improve the production and efficiency. User interfaces exist for various systems, and provide a means of:
    - Input, allowing the users to manipulate a system
    - Output, allowing the system to indicate the effects of the users' manipulation
- HMI Application (the software and programming of control screens)
    - Machine monitoring and control of valves, pumps, and other equipment that the operator can manipulate to better control the system
    - Motor control center monitoring, tracking, and control

- Building Automation and Security (control of security gates, truck loading, and other interfaces with outside world and the system)
- Electrical substation monitoring

Human-Machine Interface (or HMI) is the apparatus which presents process data to a human operator, and through which the human operator controls the process. The size and scope of an HMI can vary as the size and complexity of the system in monitors. The HMI is usually linked to the SCADA system's databases and software programs, to provide trending, diagnostic data, and management information such as scheduled maintenance procedures, logistic information, detailed schematics for a particular sensor or machine, and expert-system troubleshooting guides (Malcolm). Figure 9 and 10 below shows the basic layout and a real example of a soon to be commissioned control room screen HMI and how various systems can be brought together to be monitored by a single individual. Figure 9 emphasizes that often complex systems can be combined into one control room. This graphic that combines two distinct systems into a combined HMI with a consolidated alarm screen for two systems separated by hundreds of miles. Integrated operations support centers are growing in use. Figure 10 shows a real-world example of an HMI soon to be commissioned control HMI that was completed as part of this research work. The terms "level" in graphic design for control rooms refers to varying level of details. The typical design of control room screens is level 1 is a high-level overview of the system. Level 2 would be an overview of a major subsystems or a level of increased detail but less of an overview. Continuing to level 3 and 4 you get increased detail (such as compressors and rotating equipment). So, in this way HMI design and systems engineering is very much interconnected since both fields are designed to organize the system in a way to enable the operator to get a precise component or part while being able to get back to the overall system.

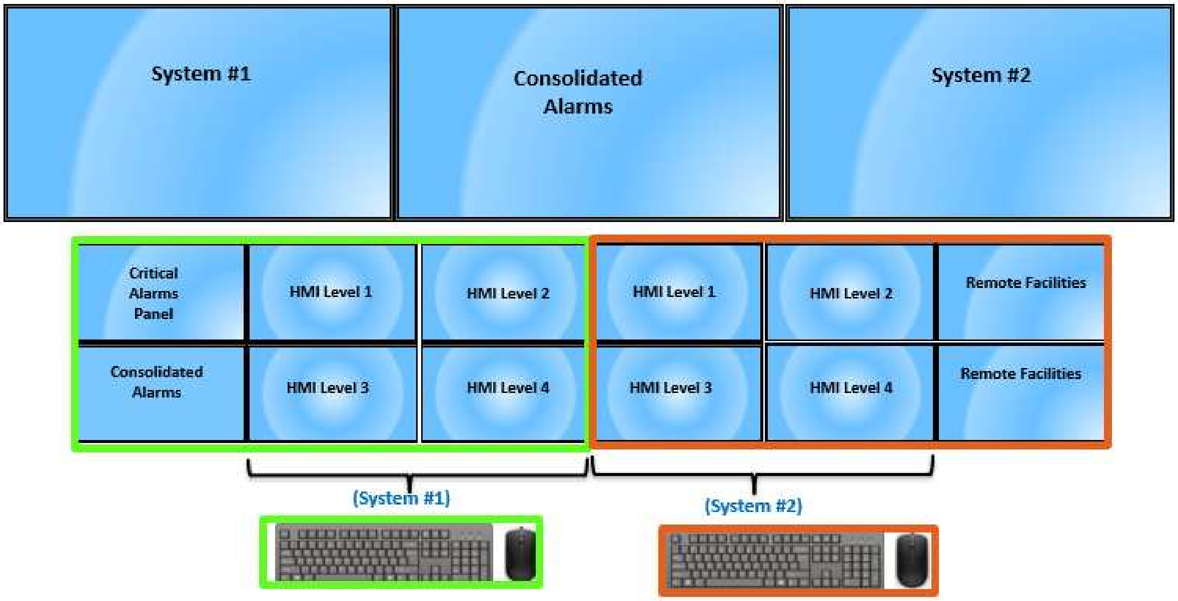


Figure 9: An example human machine interface (HMI) layout for industrial systems

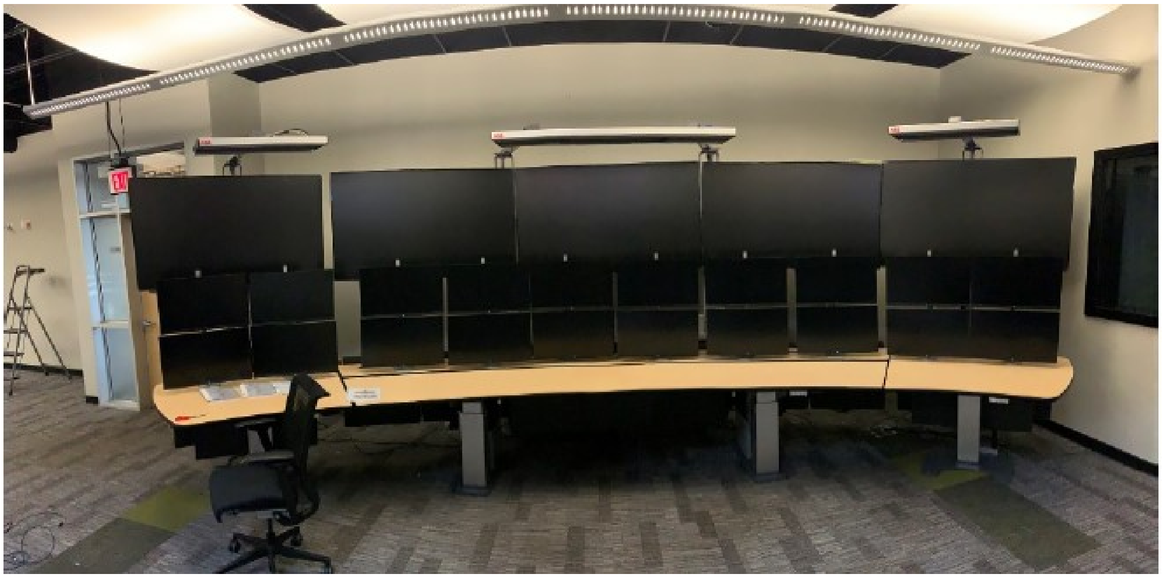


Figure 10: A photograph of an example HMI layout for industrial automation systems

After the control room/HMI, the control network extends to other layers and eventually to the business network where data is collected. A control network, alternately called industrial automation and control systems (IACS), that defines all systems (personnel, hardware, and software) that can affect the safe, secure, and reliable operation of industrial processes. They include, but are not limited to (Feldkirchner):

- Industrial control systems and associated instrumentation, which include the devices, systems, networks, and controls used to manage, operate, monitor and/or automate industrial processes.
- Associated systems at level 3 or 3.5 (DMZ level) or below of the reference model shown in Figure 11. Examples include advanced or multivariable control, online optimizers, dedicated equipment monitors, graphical interfaces, process historians, production management systems, pipeline leak detection systems, and power management systems.
- Associated internal, human, network, wireless, or machine interfaces used to provide control, safety, manufacturing, or remote operations functionality to continuous, batch, discrete, and other processes.
- An information system used to control industrial processes such as manufacturing, product handling, production, and distribution. Industrial control systems include supervisory control and data acquisition (SCADA) systems used to control geographically dispersed assets, as well as distributed control systems (DCSs) and smaller control systems using programmable logic controllers to control localized processes.

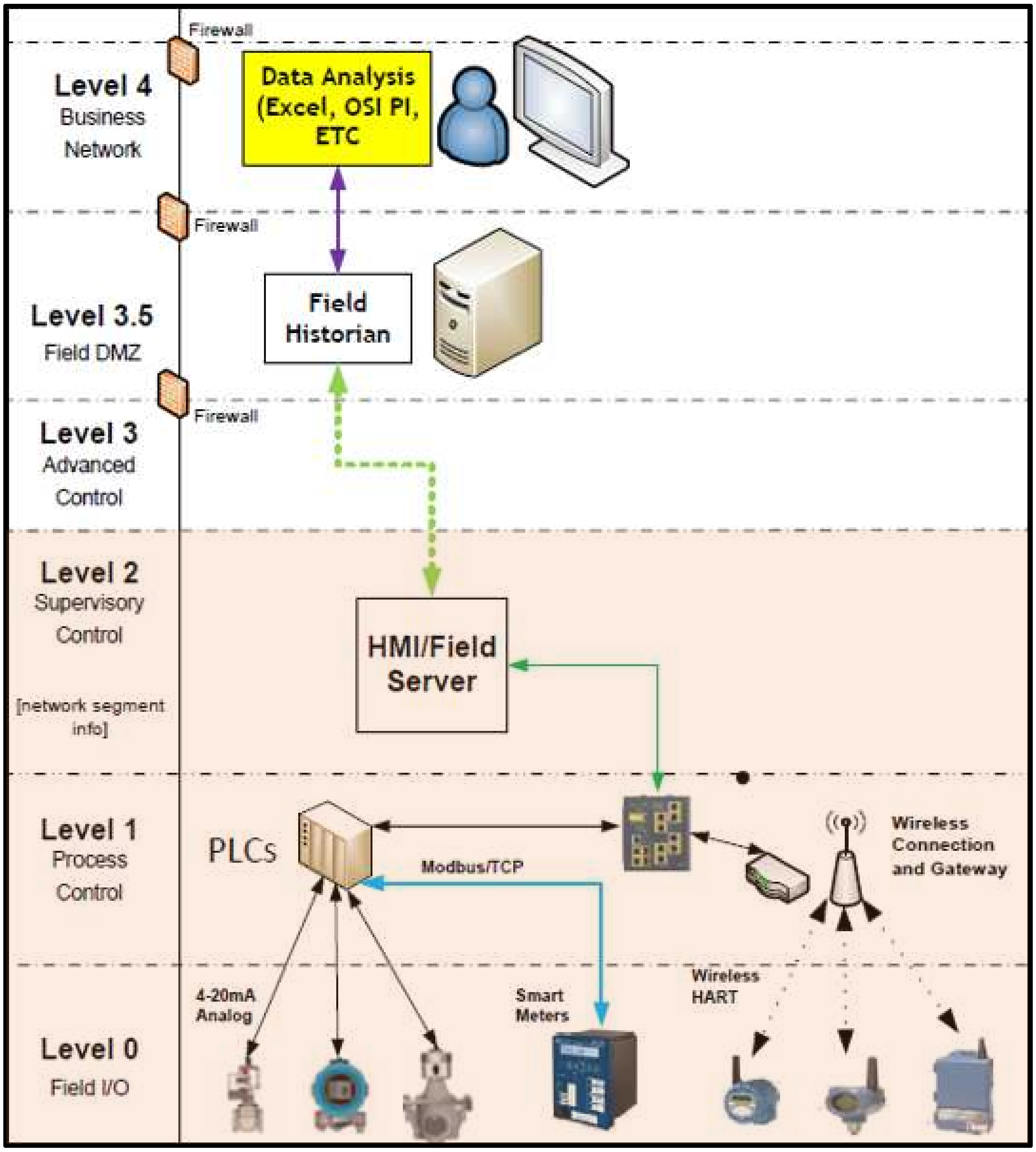


Figure 11: A simplified Purdue model, highlighting the scope of this dissertation (orange)

Figure 11 represents the Purdue model of industrial automation systems which is the standard model for representing an industrial automation system network architecture. The Purdue model is the most used architecture for automating industrial systems (Boyes) and it is used as a reference to modify network architectures to new designs. The general architecture of most industrial sites follows the Purdue model to transfer control within the levels of the network

(Li). Its key message is to keep like components segregated in different levels that are isolated by either physical protection (locked enclosures, security gates) or cyber protection (firewalls, password protection). Researchers at Purdue University developed this model as a standard for heavy industries and manufacturing (Williams). Its main purpose was to define different levels of the critical infrastructure used in continually operating facilities such as energy, chemical, and other manufacturing systems. In commercial practice, it is used as a reference and starting point when designing automation and cyber physical systems. The levels are described as:

- Level 0 – The physical process level defines the actual physical processes. Sensing and manipulating physical processes occur at this level with sensors, analyzers, actuators, and related instrumentation,
- Level 1 – Logic controllers (PLCs) and field data collection such as industrial wireless gateways and field communication switches,
- Level 2 – Supervisory control and data acquisition (SCADA),
- Level 3 – Optimization and other software in control network: Such as program to run optimization of existing sites but have operator override,
- Level 3.5 – Demilitarized zone (DMZ): A newer addition to model, this level includes security systems, such as firewalls, used to separate the business network from the process control and SCADA network,
- Level 4 – Enterprise or business network layer is where IT networking and the primary business functions occur.

The Purdue model helps establish the scope for the systems upgrades that are the subject of this investigation. Typically, major upgrades of higher levels of the Purdue model do not affect the PCN and SCADA networks (McFeaters). An example of this would be and upgrade to the historian, which can have redundancy and back-ups enabled and its outage would not affect the lower levels. This is because typically at sites information flows in 1-direction only within the model: from lower levels to high levels.

The upgrades and their challenges that are the focus of this dissertation focused on levels 0,1, and 2, where an outage would impact the continual operation of these industrial processes. Because upgrades to these system levels will cause an outage, and therefore a loss of revenue, adaptations of the traditional systems engineering lifecycle and tools must be developed to aid the upgrades to automation subsystems. For example, the obsolescence analysis was only done on Level 0 to Level 2 components due to those higher levels could be replaced without an upset to the manufacturing process (Williams).

Upgrades of automation systems pose a challenge to systems engineering as summarized below:

- Advances to automation occur at a rapid rate, including improved operator control functionality and cyber security improvements, which contrasts to the long-life and incremental advancement of the plant systems.
- New industrial systems (refineries, chemical plants, pharmaceutical manufacturing, microchip manufacturing) take large capital investment and are more likely to expand on an existing site than build a new one to support a automation upgrade.
- Outages to production are highly discouraged and are only done within limited time windows to prevent lost revenue.

As a result, automation system upgrades to large systems in industrial settings are unique, and special systems engineering tools must be built to support execution of the upgrade.

Automation System Upgrades in Practice

This work started several years ago when engineering teams identified 14 sites with major automation systems at or near end of life. This identification occurred simply based on diagnosis and an understanding of the systems. The programming languages were no longer used, no longer supported by automation equipment manufacturer, and not understood locally. Instruments and other automation equipment (such as logic controllers) were at end-of-life and like-in-kind (or identical) replacement was impossible due to discontinuation by the manufacturer. In most cases, the automation platforms were not standardized at the sites. Many COTS automation systems are unable to inter-communicate, several different programming languages are used (and supported by a small support staff), and the age of the equipment was high. As is typical for high value industrial sites, the full shutdown of the sites only occurred on a one-to-three-year basis for only about 24hrs of downtime. To upgrade the programming, instrumentation, and control room screens had to be designed and tested to ensure a seamless transition to the new system within a short window of scheduled downtime.

Within this application, the main challenge of this research project was how to upgrade complex systems (instruments, logic controllers, graphics, programming language) where the canonical stages of systems engineering could not be followed traditionally and where deliverables had to be meet quality assurance and quality control (QA/QC), accomplish design and functional requirements, and retain the previous system's operability. The identification of obsolescence for instruments and other automation/electrical equipment at the industrial sites studied were difficult. The previously shown bathtub curve (showing reactive and proactive obsolescence) highlights the individual parts and components of cyber physical systems that can suddenly start failing as they age. Their lifespan is difficult to assess due to their internal complexity (as a component) and due to external variables such as temperature, voltage, and vibration at their location. The sensitivity of these cyber systems is in contrast to mechanical systems (such as pipelines) where corrosion rate and pipe thickness can be reliably measured and combined to give an estimate of end-of-life. Beyond logic controls and software, the remaining life for the thousands of instruments that make up major portions of possible upgrades are typically hard to assess. This is because their remaining life is made of hundreds of parts that are poorly characterized.   This became a challenge identified early on in my research and a strategy needed to be developed to address this.

Secondly, another challenge arose, the testing of this new system had to be done entirely in a simulation due to the short window to upgrade the automation and did not allow for pilot tests or long commissioning times. The systems engineering methodology is well developed for new projects but is not as easily adapted to systems upgrades to extend life, or to account for technological fields where improvement in both functional features and security is rapid (such as automation, PCN security, etc.). Figure 12 shows an example of the coordination and timing required for these upgrades which can take over a year to plan and prepare. The integration and startup-up must take place quickly. Several multi-discipline teams must be coordinated to install new instruments, update, or completely replace the programming, and ensure new control room screens operate successfully during a specified outage window. These upgraded systems require testing both for extending the life of the facility, but also for the addition of new features. Safety (mostly driven by process hazard analyses and layer of protection analyses) required upgrades to also include major functionality upgrades and safety enhancements ranging from new instruments to adding safety logic controllers.

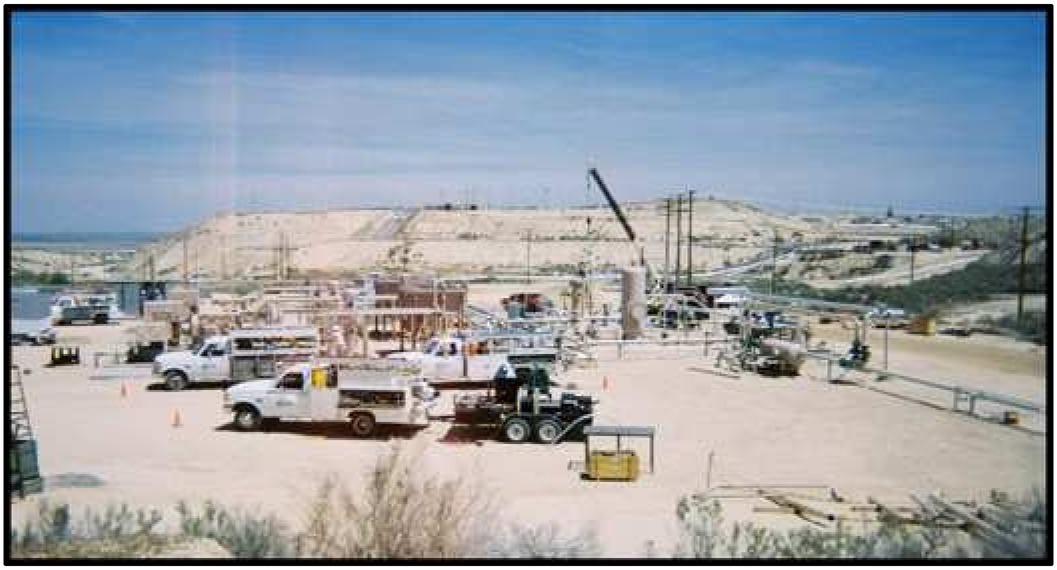


Figure 12: A photo of a rapid systems upgrade illustrating a coordinated upgrade

After the commitment to upgrade these automation systems to extend their life, additional requests would be met to enhance the system's functionality and performance. These could include the addition of new graphics to control room screens and enabling new features to alarm management (like suppression, alarm modes, and pushbuttons to take equipment out of service for maintenance while not interrupting service). Table 2 describes the types of additional system upgrades that were often requested for each major facility. This table serves as a general overview of some of the other reasons to perform automation upgrades beyond just merely extending the life of the system that was encountered frequently. Many reasons can be given to upgrade a control system and perform a similar type of upgrades, but the below table summarizes the common and recurring benefits identified during this research. This table is meant to highlight the many reasons that are often given to perform one of these upgrades at a site. The reasons could range from reliability, enhanced features, safety concerns, or other that would require a major upgrade of the automation system in place. This could be accompanied by economic justification but most of the drivers to upgrade were qualitative.

Table 2: Overview of benefits of upgrading automation subsystems.

Benefits	Comments
• Improve Safety	• Install hardware with redundancy • More fault tolerance • Inclusion of safety instrumented systems (SIS)
• Improve employee Productivity	• Graphics upgrades to control room screens • Include newly developed functionality (alarm suppression, advanced HMI graphics)
• Regulatory Compliance	• Emissions reporting • Accurate custody transfer of product • Electrical distribution (Peacock)
• Reduce Maintenance Costs	• Employ machine learning or complex control loops
• Safety Sustainability	• Better audit of integrity of system • Simpler automation systems
• Greater Flexibility	• Systems easier to expand without going offline • Ring topology versus start topology
• Better Performance	• Quicker response times for controllers • Imbedded diagnostics
• Increase Production Output	• Systems able to operate at best efficiency point (optimal operating window) without operator intervention

After the initial requirement to upgrade the systems and the additional functional and performance requirements were collected, the systems design work began. This was uniquely challenging again for automation of the industrial cyber physical systems considered here due to incompatibility between components. Figures 13-15 show 3 examples of automation systems with different programming that were at many of the sites before consolidation to a common programming system and language were selected and implemented. The programs to operate these industrial sites were developed by line-by-line (by either ladder logic as with SLC-500 and ControlLogix examples or relay logic for Modicon platform shown in the figures) and could not be automatically upgraded, especially with added functionality, without re-creating the program again line by line. The challenges of this painstaking and error-prone process were exacerbated by the fact that these upgrades from old program to new program had to take place within hours and were required to have no errors from converting old to new programming.

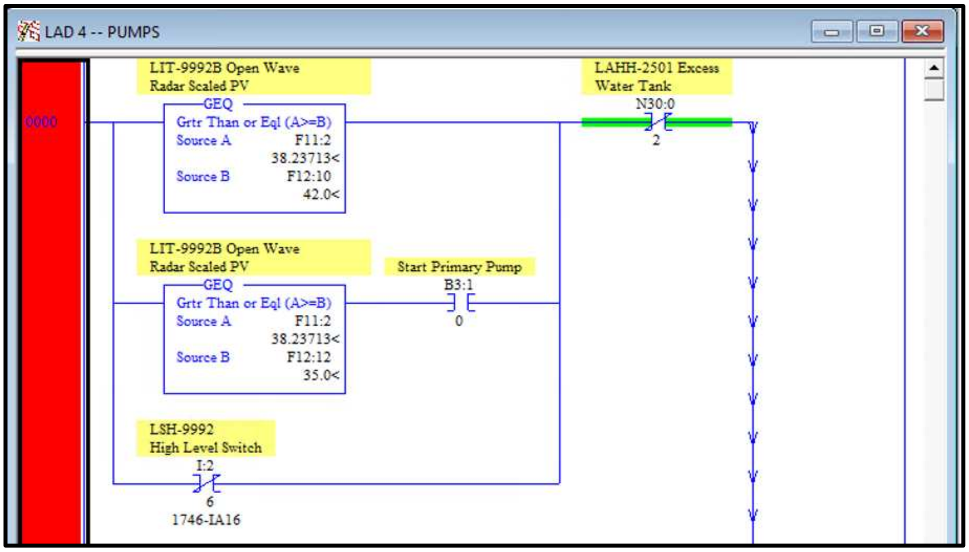


Figure 13: SLC-500 programming, to be contrasted with other programming examples

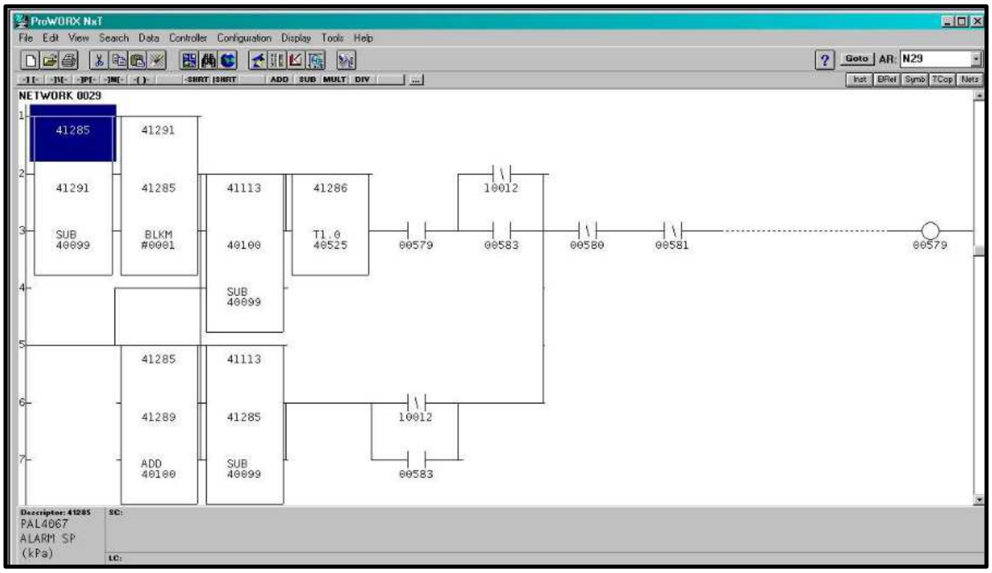
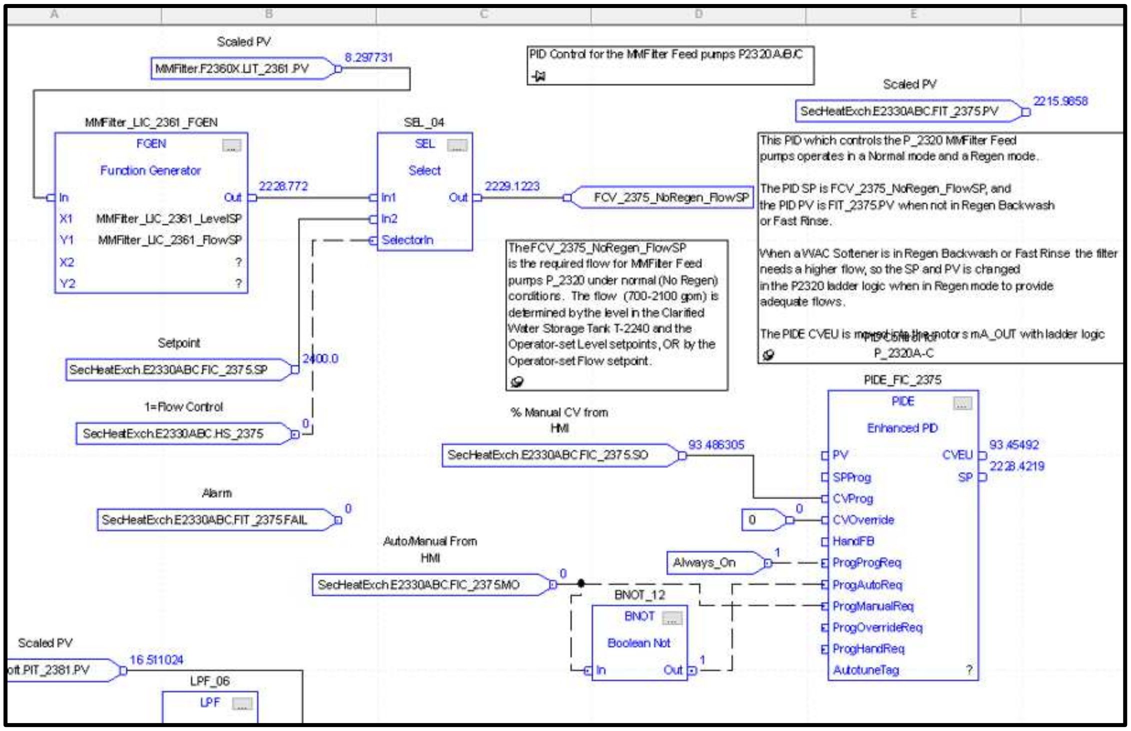


Figure 14: Modicon programming, to be contrasted with the other programming examples


PID Control for the MMFilter Feed pumps P2320A,B,C

This PID which controls the P_2320 MMFilter Feed pumps operates in a Normal mode and a Regen mode.

The PID SP is FCV_2375_NoRegen_FlowSP, and the PID PV is FIT_2375.PV when not in Regen Backwash or Fast Rinse.

When a WAC Softener is in Regen Backwash or Fast Rinse the filter needs a higher flow, so the SP and PV is changed in the P2320 ladder logic when in Regen mode to provide adequate flows.

The PIDE CVEU is moved into the motor s mA_OUT with ladder logic

The FCV_2375_NoRegen_FlowSP is the required flow for MMFilter Feed pumps P_2320 under normal (No Regen) conditions. The flow (700-2100 gpm) is determined by the level in the Clarified Water Storage Tank T-2240 and the Operator-set Level setpoints, OR by the Operator-set Flow setpoint.

Figure 15: ControlLogix programming, to be contrasted with other programming examples

In an ideal situation, for new industrial systems, new instruments could be tested prior to start-up by following these steps:

- "Loop check" each instrument. This ensures that each instrument is landed to the correct termination point inside the logic controller and tested to ensure the correct range and span of the instrument is configured in programming.
- Partially function test each line of the programming. This involves raising and lowering the variable of the instrument to ensure the correct logic was enabled such as starting and stopping of pumps, alarms, and automatic shutdowns
- Fully function test the system. Start up the system with either the process fluid or water depending on the service it was in (due to safety) and run through all the complex control such as emergency shutdowns, advanced process control (cascade, feed forward, etc.), and ensure control from HMI.

Due to these need to perform these tests, the startup of even a small system can be complex and time consuming, in the context of the limited time available for switch over. Studies have shown that correcting programming errors in the field can take twelve times as long as during commissioning (Zimmerman). The correction in a laboratory can be done with no time constraints, more resources, and less time pressure than can be done at the site during commissioning. A rigorous QA/QC and laboratory testing process prior to commissioning was required to avoid making corrections in the field, delaying commissioning. In summary, thousands of lines of code must be transferred to new automation platform, new functionality must be added, and the cutover to new system must be done seamlessly during a short window.

Figure 16 shows the setup in a laboratory of five basic logic controllers and two safety controllers for an upgrade. The testing for a medium sized site would go through months of testing prior to field commissioning. Figure 17 shows the setup of instruments and final control elements (valves, etc.) that were also testing and pre-assembled to minimize system downtime and limit lost revenue that was done during these upgrade projects. The pre-assembly of valves and final elements for testing improved ease of installation prior to field commissioning by ensuring they were pre-configured correctly.

The control systems are the automated parts that control the individual unit operations and rotating equipment. They include instruments, logic controllers, motor control centers, transformers that control things like tanks, distillation columns, and other major systems. Programmable logic controls (PLCs) are the heart of many modern manufacturing processes that solve logic to control automated equipment. PLCs use programming languages as the basis of their logic and control. Input signals from different instruments, pumps, or other parts of a facility are read from the PLC, logic is solved, and outputs from PLC are sent to devices which control the plants operation (Alphonsus).   Logic controllers send data to control room screens (HMIs and operator terminals).

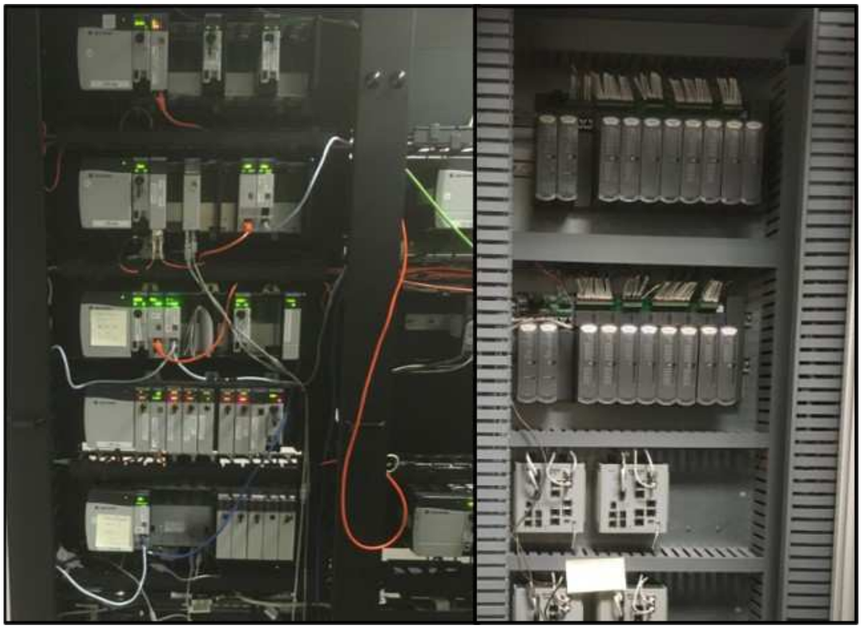


Figure 16: The testing of logic controllers in a laboratory environment prior to installation

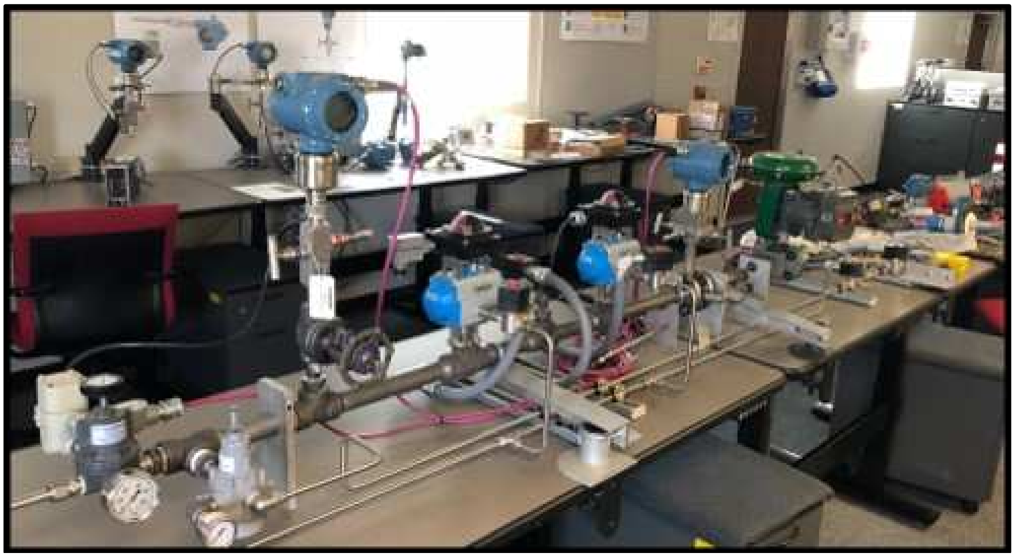


Figure 17: The pre-assembly of valves and final elements, prior to installation

Figure 18 shows how the automation system interfaces with the rest of the system and interacts with the other traditional systems of a facility such as separations vessels (systems “A” and “C”) and pumping systems (System “B”). This graphical representation is to show how the automation systems interface with traditional mechanical process equipment, labeled as System "A", "B", and "C". As prescribed in Purdue model, level "0" contains instruments, drives, and final elements. Level "1" is logic controllers that solve algorithms to control these process control systems and other unit operations. Level "2" is control room screens (human machine interfaces), operator terminals (local screens), and other panels where humans can input changes to setpoints, acknowledge alarms, and other changes.

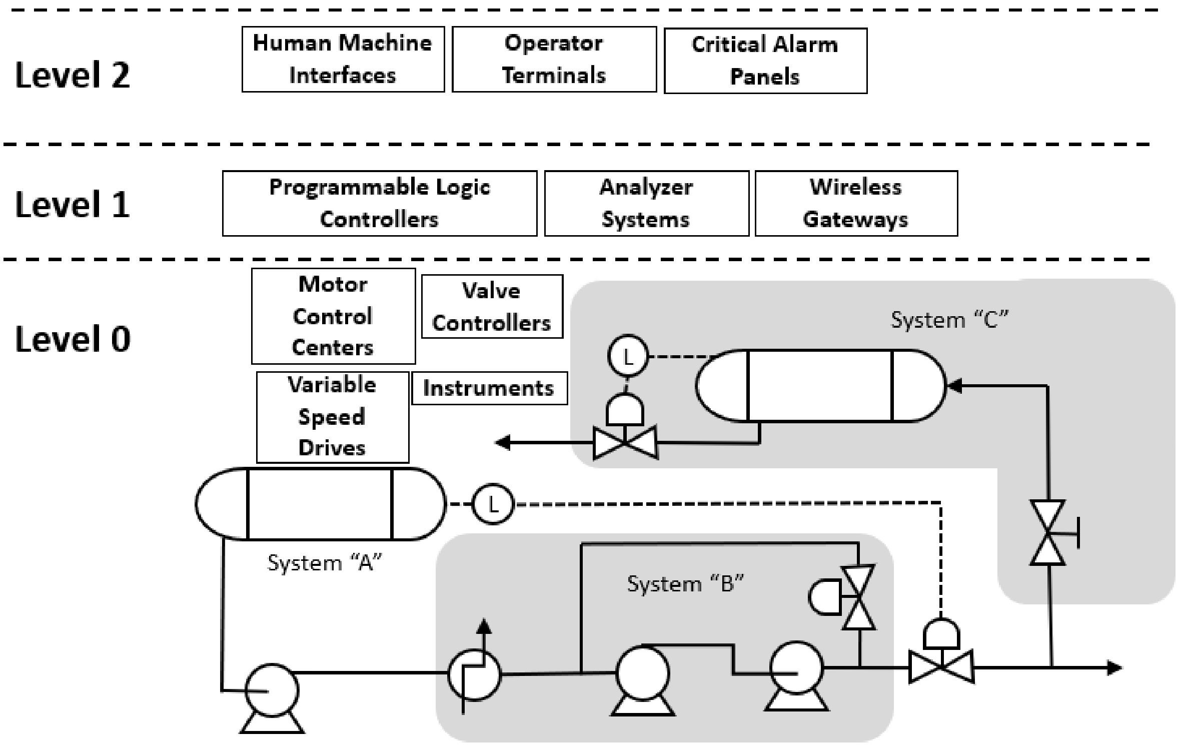


Figure 18: Automation systems interface with mechanical equipment

Installing redundant controllers and control rooms is typically difficult to implement and run into same obsolescence challenges previously mentioned. Higher levels of the Purdue model typically can have redundancy or outages don't result in and outage of the heavy industrial process site.

PLCs and automation equipment are designed to keep the facility running continuously and minimize operator intervention. Creating duplicate logic controllers, with field instruments, and control rooms is often costly and still does not avoid the issue of rapid obsolescence these components face. Often PLCs and other automation equipment are installed with an expectation that they will run continuously for greater than 10 years but the equipment they automate, such as tanks and distillation columns, last much longer (P. V. Sandborn). Therefore, upgrades to automation equipment typically occur throughout the life of a facility. An example of an upgraded control system is seen below (Figure 19) where the logic controller on top of was no longer manufactured, and parts could not be replaced. A failure would have resulted in a prolonged outage at the facility. The logic controller on the top of Figure 19 was commonly manufactured in the 1980s but had been discontinued and each of the components were no longer manufactured or supported. The bottom logic controller, manufactured by same company, was the modern version. Its components were still being manufactured and supported. The upgrade of this obsolete control system involves its replacement with a newer one. The PLC-5, top of Figure 19, (which was, as mentioned, very common in the 1980s) being was replaced with a more modern controller. This replacement would ensure if any part failed it could be replaced. Without the upgrade, any part that failed would be difficult to replace. When performed correctly, obsolescence management for continuous operation ensures that the facility stays running. As PLCs are electrical equipment running with software, they typically require upgrades to avoid obsolescence (Cesar). Beyond the logic controller, often times many of the subcomponents and parts inside the enclosure also require frequent upgrades and replacement.

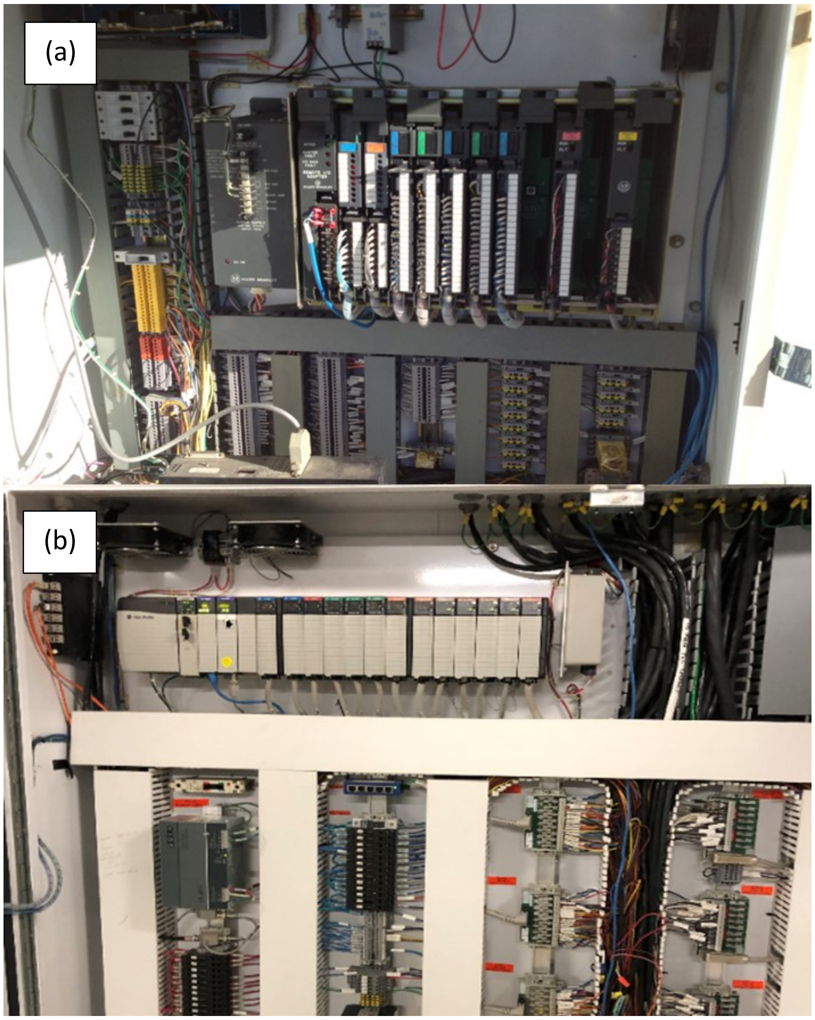


Figure 19: The upgrade of an (a) obsolete control system, to a (b) modern one

In addition to hardware obsolescence, the analysis of industrial sites found problems in that the site no longer had a local resource for programming. There are several standards used for programming languages such as ladder logic, sequential function charts, and text language. Ladder logic is the most common type, both for our sites and in industry (Erickson). While there are standards written for ladder logic and PLC programming, in general (such as IEC 61131 (International Electrotechnical Commission programming standard for programmable logic controllers) (Öhman), there is still the challenge of upgrading from different and older PLC platforms to newer.   Even two models of logic controllers made by the same manufacturer often have programming conversion software to upgrade to newer versions automatically. But this level of automation is still regarded as only a “starting point” for large sites and significant testing must be completed after the automatic conversion of older programs to newer. This testing is performed manually by engineering technicians. This automatic conversion is not available for converting controller programs from one automation manufacturer, as is usually the case in such a multi-year maintenance program. The continual expansion and modification of the industrial system often required consideration of differing programming types, disconnected automation systems, and a variety of technology platforms in each field.

In summary, the analysis of several industrial sites determined that a long-term series of projects should be undertaken to upgrade the control systems, programming, and any associated automation equipment from obsolescence at the next facility planned shutdown. This would require determining which parts and components were near obsolescence, designing both the hardware and software upgrades, and planning to replace during the next turnaround to minimize facility downtime (and thus minimize loss revenue). A multi-year program was undertaken to upgrade these industrial sites. This involved taking aging systems, with obsolete programming, and upgrading to newer systems with different programming language. While the systems were not yet in failure, both the discontinuation of key parts of the control system and the product reaching the end of its documented life span was used as a driver to justify upgrading the control system from obsolescence to a more modern system. These upgrades had to be done during turnarounds (shutdowns for major maintenance) to minimize lost revenue, so attention was given on how to manage upgrading obsolete components based on a turnaround cycle (TC). For additional context, Figure 20 shows the other components of the systems upgrades such as local junction boxes, cable tray, and underground conduit. When this research began, it was difficult to quantify the risk of automation systems failing. Some cases were obvious when the automation system was no longer manufactured, and its failure would either result in a prolonged shutdown due to inability to find parts. The initial investigations identified major components of the system to upgrade based on knowledge of the system and discussion with operators and key instrumentation and electrical personnel. No plan or program was available at the beginning of this project to methodically identify at-risk components.

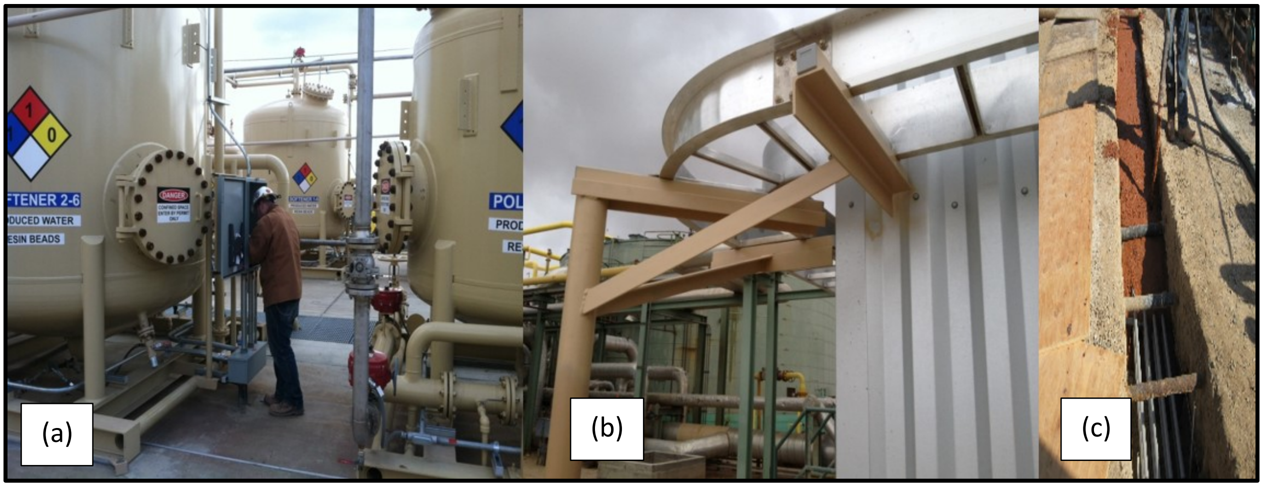


Figure 20: A photo of (a) local junction boxes, (b) cable tray, (c) and underground conduit

Automation system upgrades performed during this research

After the identification of the parts that would be upgraded, an initiative was undertaken that provide a standardized controls platform that would be used upgrading obsolete control systems and degraded supporting infrastructure. The standardized platform included a common logic controller at all sites, similar instruments and electrical equipment, and graphics at control rooms. This standardization across multi-sites, and across a series of complex process systems allowed for future upgrades to be done quickly and efficiently.   The benefits of standardization included a common programming language to ensure adequate and interchangeable personnel so programmers could work at multiple different sites.   The common graphics allowed familiarity between sites for operations. In addition, it allowed for easier upgrades in future by documenting standardized functionality for the logic controller and HMIs to be able to quickly identify when sites lagged others in functionality. So, a multi-year and multi-location effort with individual projects in various stages included fourteen facilities comprising of power plants, process plants, pumping stations, gas plants and other major sites. The upgrade is outlined in Figure 21 below and its stated goals were to:

- Replace aged control systems of each facility,
- Implement separate Safety Instrumented System (SIS), where needed,
- Construct a robust Process Control Network (PCN),
- Implement enhanced situational awareness of operators through improved Human Machine Interface (control screens) and better alarm management via graphics upgrades and console functionality improvements,
- Implement relevant technical codes and standards if sites no longer met regulatory requirements or engineering codes,
- Update automation-related documentation for the entire facility (P&ID, PFD, Control Philosophy, Plot Plan, etc.),
- Enable ongoing operator orientation through dynamic process simulation.

Figure 21 is meant to emphasize that there while there are similarities between each site (such as similar rotating equipment and unit operations), each site is unique with its main function, size, and complexity. The figure shows the timeline, relative upgrade difficulty, and overall cost of each system upgrade.

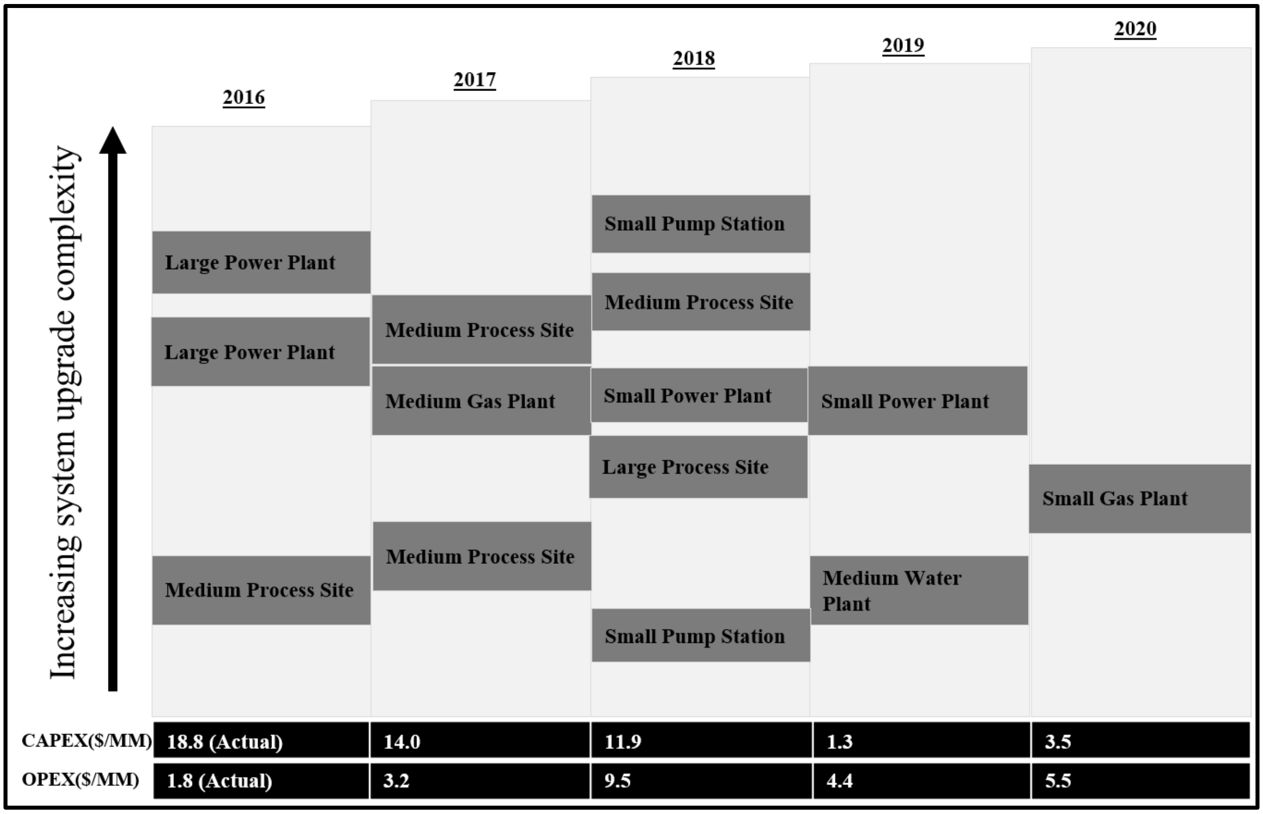


Figure 21: The timespan of upgrade projects at the facilities under investigation

Due to the unique demands of these upgrades, several problems were discovered with traditional systems engineering processes, that we have sought to address. These challenges and resolutions became the basis for this research. Tabulated in Table 3 are the challenges, resolutions and possible solutions being the major focus areas of my research. The research documented in this dissertation demonstrates how to rapidly upgrade automation systems by identifying obsolete components and specifying replacements, simulating results, and auditing system after the upgrade as shown in table below. The research effort focused mainly on the underlined set of solutions (Table 3). These major challenges shown in table 3 are paired with their potential solution. These major challenges identified represent the major challenges faced and the focus of this research. The major chapters are organized around the potential solutions identified.

Table 3: The challenges and proposed solutions for automation upgrades

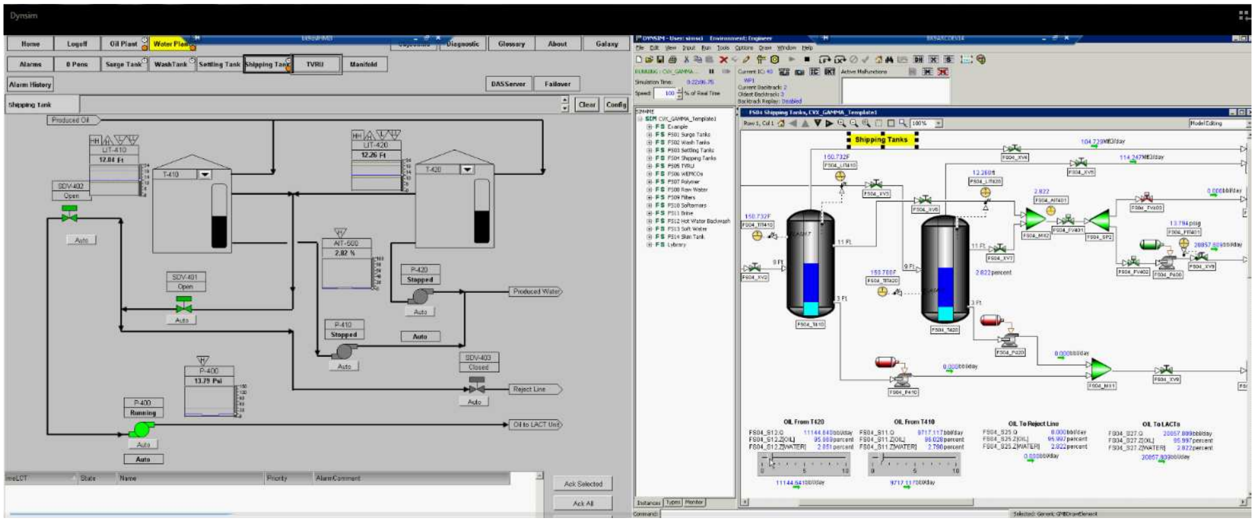
Major Challenge	Specific Problems Faced	Solution
Difficultly in efficiently identifying obsolescence in field	• Old control systems – end-of-life, spares unavailable • Bathtub curve – increasing failure rate • Mix of multiple generations of control systems • Mix of multiple application programming styles	• Replace with uniform, newer, more powerful, standardized platform • Follow rigorous programming standards, templates • **<u>Develop standardized way to screen for obsolescence based on a common risk management practice to aid operations and maintenance during turnarounds</u>**
Re-creating code can introduce hundreds of errors due to human mistakes	• Current practice – checkout code during commissioning in the field – risky tests, longer commissioning time • Inadequate Operator Orientation of new application – risk for Operator Errors during run • No opportunity to test control systems and Operator behavior during upset conditions	• Implement software-based dynamic process simulator for control logic checkout in controlled test environment in Laboratory • Use the same tool for Operator Orientation and Upset Conditions Behavior Monitoring in safe environment • **<u>Standardize the simulation process to ensure the proper fidelity level is reached and all functionality is checked to ensure smooth cutover</u>**
Systems regularly can undergo small changes and documentation is not kept up to date.	• Older facilities – documentation totally out of date • Lots of smaller scale modifications in the field • Several versions of engineering drawings – which is the single version of truth? • Significant mismatch between P&IDs, SCADA Index, Control Philosophy & application program	• Painstakingly update documentation • Resolve discrepancies within documents • Store updated documentation in PSI (Process Safety Information) EDMS (Electronic Data Management system) • **<u>Create audit and enforcement tool to automatically track changes to alarm configuration and automation infrastructure.</u>**

As outlined in Table 3, the upgrades of automation systems for major sites studied posed many challenges. These major issues in automation upgrades, their specific problems, and solutions employed during this study are separated into 3 main challenges. The first major challenge addressed was the identification of obsolete equipment. This was a done by first researching computerized maintenance systems in order to develop a risk management plan that quickly screens for obsolescence. After identifying the major risk factors, a simple method was developed that allowed personnel to quickly find all equipment requiring upgrade for a site containing thousands of automation equipment. An initial list could be prepared that became the basis for the obsolescence upgrades and future design work. This process was repeated for the next site.

The next major challenge was how to ensure the programming was done accurately without having a lengthy commissioning process in the field. A simulation was built to enable testing of the programming, but the challenge was how to build the simulation to address the major concerns for each site, perform proper QA/QC on the functionality, and to ensure the complexity of the simulation did not cause it to be too expensive to build. "Build a simulation" is a request many engineers and scientist have to address risks or unknowns for a system being developed. Sometimes that is all the guidance that is given prior to undertaking the simulation work from stakeholders to address a risk with a system under development. "Build a simulation" was requested of me many times to address the risk of changing from an old automation control system to a new upgraded one. Such a request was made for a large chemical plant to begin my first of the fourteen sites studied in my research. Beyond being an automation engineer for the project, I was given the side job of being the simulation advisor. The request was "build a simulation to test the system prior to the commissioning to remove the risk of a shutdown" was all the guidance I was given as I broke out into a cold sweat was how I began my journey of working with simulations with regards to upgrading automation systems. This project main goal was to replace the controllers with new versions which involved completely re-writing the programming to a new language (think going from FORTRAN to C++ for the industrial programming of a large complex facility). It required a flawless transition including removal of the old programming, and installation of the new device with the new programming with as little downtime as possible. The work had to have zero errors, or it would cause millions of dollars in losses as the facility either had to revert the change or debug the error in the field.   To simplify the project's main objective was to remove the industrial computers at each site and replace with a new one written in different code.

This meant the code had to be replaced manually. The two programming languages were completely different and had to be converted line by line, a process which the team knew would introduce the potential for human errors.   After application of the methods and tools described in this dissertation, it was decided to incorporate dynamic simulation into the project to verify programming code integrity and test functionality during Factory Acceptance Testing (FAT) before deployment of the new control systems to ensure a seamless switchover. This use of dynamic simulation was particularly important in addressing the established practice of using checkout code during PLC commissioning in the field. The traditional checkout practice, however, provided no opportunity to test the response of critical safety systems to the new PLCs. Software-based dynamic process simulator for code checkout addressed this problem by providing a controlled test environment. It could also be used to train operators on the new control system prior to operation.

With the main goal of the simulation identified to QA/QC the new programming and to also ensure downtime was minimized during the upgrade, the underlying challenge was how to efficiently use the simulation to ensure its design meet the requirements and not exceed costs and time constraints. Because the team expected to make many simulations over a period of years, a metric was developed to measure the cost improvements resulting from simulations during the upgrade projects. This metric was Cost per I/O, or a normalized cost per instrument for each system.   Other metrics were developed to benchmark the work, but this metric allowed normalization across facilities of varying complexity. An example of cross checking the simulation with the programming is shown in Figure 22. This shows an example of checking the programming (left side HMI screen) against the simulation (right site) to ensure programming accuracy.



OIL From T420
FS04_S12.Q 11144.640bbl/day
FS04_S12.Z[OIL] 95.069percent
FS04_S12.Z[WATER] 2.851percent
11144.641bbl/day

OIL From T410
FS04_S11.Q 9717.117bbl/day
FS04_S11.Z[OIL] 96.028percent
FS04_S11.Z[WATER] 2.780percent
9717.117bbl/day

OIL To Reject Line
FS04_S25.Q 0.000bbl/day
FS04_S25.Z[OIL] 95.997percent
FS04_S25.Z[WATER] 2.822percent
0.000bbl/day

OIL To LACTs
FS04_S27.Q 20857.809bbl/day
FS04_S27.Z[OIL] 95.997percent
FS04_S27.Z[WATER] 2.822percent
20857.809bbl/day

Figure 22: Example of QA/QC industrial programming tests

Lastly, a concern existed that these sites would continue to undergo small changes after the major upgrade projects are completed. How does one maintain the original integrity of system, design specifications, and functional requirements?   An audit system was developed for automation components that continually scanned the alarm configuration and in field components against a database to ensure that any changes that violated the requirements of the system were quickly identified and addressed. This confidence that the sites maintained their integrity over long periods of time while concurrent project upgrade occurred.

As mentioned previously, the main purpose of my research was upgrading legacy automation systems and programmable logic controllers (PLCs) across several industrial sites with the stated requirement that long or unplanned shutdowns could not occur during the upgrade. Automation systems are typically made up of commercial off the shelf (COTS) systems composed of both hardware and software with lifespans shorter than the overall systems' lifespan (Alelyani). An example of this is in large scale manufacturing when tanks, rotating equipment, piping can last many decades, but the automation systems that support them can become obsolete in 5-10 years (Rajagopal). The differences in lifespans of these systems requires constant upgrading to ensure reliability and operational availability. The focus of this research was the series of projects completed to upgrade control systems for power, chemical, refining, and other process industries with the requirement that upgrading automation systems, testing, and confirming integrity extends into space, defense, avionics, and other sectors (Rojo) happens routinely. This work started with a risk analysis of several southern California process sites where it was identified the automation systems had reached obsolescence roughly 5 years ago. Since then, a program was developed and roughly 14 sites have been upgraded with work continuing today.

The area of concern that led to this method being developed was that automation systems rapidly become obsolete while the rest of the system has a much longer lifespan. This problem is increasing as well across many industries (Bil). While in systems engineering the traditional approach is to approach development of a system holistically, with automation, there is a need often to do major upgrades of subsystems within systems. This has been explored before by other authors where it was determined that obsolescence shifts the baseline of the system's performance during the system engineering lifecycle (Herald).

The first step to perform these upgrades was to research existing tools and research into similar challenges faced by systems engineering discipline and field of study. The focus of my literature review extended from not only industrial systems but covered military and aerospace platforms.   The next chapter provides an overview, but the literature review will continue in detail for each following chapter. Obsolescence management is key focus area of many fields including systems engineering. Thus, past work will be leveraged and incorporated into this work to determine efficient ways to identify obsolete components. While my research and series of systems of upgrades had some unique characteristics, the challenge of extending the life of a system by upgrading a major subsystem has been completed before and its learnings will be leveraged here.   Finally, large scale automation upgrades will be analyzed in the literature review to see what common characteristics existed between these upgrades and other recent major upgrades.

Chapter 3 – Literature Review

Overview of Relevant Research for Systems Engineering

This chapter gives a high-level overview of relevant research in systems engineering and other automation related areas that relate closely to this work. A more comprehensive set of the literature is cited in the chapters that make up the core of this dissertation. The purpose of this chapter is to briefly review a few key works closest in relevance to this study, and to provide a brief introduction to the literature. The review is broken into three parts that relate to different aspects of the work in this report:

- Obsolescence management research
- Subsystem upgrades
- Case studies of major automation upgrades to systems

Many recent publications have addressed increasing challenges on the horizon for development, operations and maintenance of cyber physical systems. This can provide both opportunities and challenges for systems engineering, as a field of study as applied to thes systems. The increasing utilization and integration of digital technologies over several decades has led to many studies on its impact and its interface with systems engineering. "Digital transformation" is a term that describes the changes imposed by information technologies (IT) to (partly) automatize tasks (Legner). Digital transformation is evident in numerous societal areas from commerce to politics.

Automation and systems engineering share many aspects and future work should be aimed at closer linking these two fields. Automation deals with process, people (operators), and business to link all together at different levels much in the same way systems engineering unites fields of engineering and business together. Systems engineering and automation coming together as one engineering field is currently being studied with technology maps that link automation development with trends in systems engineering research (Tafvizi Zavareh). While the focus of this literature review are topics that related directly with automation subsystem upgrades, the development of automaton and system engineering will be linked going forward. Digital transformation, industrial internet of thigs (IIOT), Industry 4.0, and other terms used by systems engineering have a lot of overlap with ongoing developments in automation field (Shin). The next three subsections summarize some of the studies going on that tie automation and systems engineering discipline together.

Research on Obsolescence in Automation Systems

The literature review for this research starts in reviewing the broad topic of obsolescence management for automation systems. Much of the work has centered on development of key performance indicators where the goal of the work is to establish, based on failure rates, when to replace individual components. Most relevant to this work was the studies of Ferreira et al. who built a risk-based system for all equipment in a facility that focused on identification of obsolete components based on several factors including (1) Equipment replacement forecast, (2) Component price, (3) Retrofit difficulty, and (4) Retrofit cost. (Ferreira) Their work acknowledges the restrictions of time and recommended a quarterly look ahead for components that need replacement to allow for sufficient planning. But some of the drawbacks were that the weighting system would be seen as complex and difficult to apply for by personnel typically responsible for daily operations and maintenance. Another limitation of these studies was the focus was mainly on mechanical (non-automation) equipment. In contrast to Ferreira, this work seeks to develop a weighting system that is based on risk-based assessments using a decision matrix. The loading of difficult-to-obtain data was avoided such as taking ratios of proactive and reactive maintenance. This decision matrix also avoided using complicated formulas to assess individual replacement. Their work also did not consider the age of equipment in a way that would be relevant for the industrial systems considered here. Equipment age was determined to be a key factor in assessing the remaining life of automation and electrical equipment based on established run hours by manufacturers and observed time for failure. Leveraging the estimated lifespan of automation and electrical equipment is a key requirement for an obsolescence management system which can be relevant to the oil and gas processing industry, with its long component lifetimes.

Another relevant example of obsolescence management as implemented in the literature established a connection between a continuously ongoing systems engineering lifecycle, and obsolescence management. This study focused on obsolescence management and need to continually consider subsystem upgrades: “a system design is baselined and instantiated, then the challenge during development, production, and utilization life cycle stages is to maintain the currency of the physical system baseline to facilitate affordable system support.” (Herald). Their work established an Obsolescence Management Framework (OMF) which is a proposed method for a system design and evolution. In this method they can connect directly to system engineering principles. The OMF articulates the six integral components necessary to protect a system from operationally ineffective evolution due to obsolescence of the elements that make up the system. The drawback of this framework, from the point of view of industrial automation, was it was primarily focused on military and aerospace systems. These types of systems focus on a “clean sheet” design phase and not on rapid upgrades to existing systems. They detail 6 framework components (technology road mapping, system costing, technology forecasting, technology trade study and product selection, technology surveillance and health assessment, and technology transition) as key tenets to build into systems to make them robust against obsolescence.   This work's main emphasis is to design against obsolescence. This is done by addressing key issues early in system engineering phases, but does not directly address subsystem upgrades or other aspects of industrial automation systems.

Another risk-based approach to managing obsolescence was proposed for off-shore oil platforms. This is a similar industrial setting as the focus of this dissertation research. For this approach, it again focused on all equipment and but only addresses a simple approach to identifying obsolescence in industrial sites (Memuletiwon). It only uses "years to end of life" which implies that the expected lifespan of the equipment is known. The expected lifespan of pipelines and rotating equipment is often known with corrosion data, but this is often not the case for electrical equipment that can be made up of thousands of components and is complex. Their review of other obsolescence management literature focuses on management practices and high-level guidance that can be applied to many industries.   The integration of obsolescence management can be incorporated into systems engineering design from the early stages of concept development to build a system that is robust against obsolescence, but presently in the systems engineering discipline there is little literature which addresses the upgrade of major subsystems to reverse obsolescence once the system is already built and functioning.   Table 4 gives an overview of the literature reviewed for enabling a contrast to the state of the art. A more detailed literature review is included in each of the core chapters of this dissertation. Each chapter details the core research, background information, and leveraged studies to support each developed theory and practice in each chapter.

Table 4: An overview of the literature review of obsolescence management

Identification of Obsolescence in Automation Systems		
Citation	Applicability to automation upgrades	Limitations
Ferreira et al (2019)	• Risk based approach to quickly identifying obsolescence for heavy industries • Long time for planning upgrades a strong factor • Addresses uniqueness of commercial of the shelf (COTS) systems	• Difficultly for operation personnel to apply techniques • No focus on specific challenges with automation systems
Herald et al. (2009)	• Integration of obsolescence into systems engineering lifecycle • Generic framework that allows application to all systems (multiple application in various industries)	• Focus is on building systems using traditional systems engineering lifecycle (i.e. building in robustness to systems from onset but not subsystems upgrades when system is operational
Memuletiwon, D. T., et al., (2017)	• Established obsolescence guidance for managing industrial facilities with regards to risk management and determining the remaining life of the equipment	• The work assumes the remaining life can be determined exactly which is difficult with automation equipment

Literature on subsystem upgrades in System Engineering

This literature review seeks to understand the work in the system engineering field that has been completed to establish guidelines for sub-systems upgrades for existing facilities. The first report analyzed was a recent publication that focused on developing a framework for modular upgrades to existing systems. This work focused on a way to find an optimal approach to upgrade subsystems (Broas). Their technique focuses on identifying key new technologies or innovations and applying them efficiently. Their method incorporates directly into the systems engineering lifecycle. This work did not address specific automation challenges in heavy industries, and is challenged to address problems such as turnaround planning and challenges with programming upgrades.

Another recent publication focused directly on the upgrade of cyber physical systems within the systems engineering framework (Kutscher). Their work established a high-level methodology to upgrade legacy automations systems that consists of several states with iterations: (1) initial state analysis, (2) upgrade options identification, (3) identification of technologies, (4) evaluation, and (5) implementation and validation. Their work accepts the current challenge with legacy automation systems: “there is a demand for flexible production systems that combine the virtual and the physical world within Cyber-Physical Systems (CPS). As a result, the owners of production plants must decide between acquisition of new systems and upgrade the existing ones”. While their work gives a simplified framework and methodology to upgrade automation systems, it does not address the core challenges of automation systems in industrial settings. Their work presents a methodology for the upgrade to CPS, which is adapted from the development of mechanical systems and focuses on the key areas that drive automation upgrades. These key areas are listed in their work as: data processing, autonomy, intersection with humans, environment, and other systems, communication, identification, virtual representation, and security. To support the upgrade methodology, a novel assessment scheme based on identified CPS categories is also proposed. Their supporting tools are especially important for upgrade projects because it can be assumed that interdisciplinary teams are involved with automation upgrades and should be involved with selection of technology.

Another recent publication that focuses on subsystem upgrades looked at establishing an open architecture for aircraft to aid in module upgrades. This work discusses the Modular Open Systems Architecture (MOSA) approach. This approach is a method of employing modular systems for aircraft to increase future upgrades and adaptability. This method also hopes to reduce the cost of upgrades. MOSA, according to the U.S. Department of Defense (DoD), is “a technical and business strategy for designing an affordable and adaptable system. A MOSA is the DoD preferred method for implementation of open systems, and it is required by United States law (DoD).” A key attribute for MOSA is creating a standardized system and list of requirements so that third party manufacturers can design their systems to be compatible not only with existing systems but with each other. This work focuses on aircraft, which is its main downside to applying to the work in my research. However, the system designed for future obsolescence upgrades of automation equipment (albeit for aerospace systems) had relevance to automation system upgrades in my studies. Often it is a request by manufacturing companies to outside automation companies to build in interchangeability and modularity to their automation components to allow for easier upgrades at lower cost in the future (Maier) . This DoD study shows that aerospace and industry share a common challenge of automation upgrades.

A high-level overview of the literature review completed is given in Table 5 which shows rigorous work done to establish a methodology for automation and cyber physical upgrades inside the systems engineering framework. However, addressing specific challenges with industrial automation upgrades were not discussed. This summary of recent publications which focus on automation subsystem upgrades within the traditional systems engineering lifecycle highlighted many unique issues. It was notable that so many independent works referenced digital twinning as an emerging technology for major subsystem upgrades. Many of the works also referenced the challenges unique to each industry such as automotive having specific periods when major upgrades occur and the interaction with operations and maintenance personnel.

Table 5: Summary of recent publications on subsystem upgrades.

Integration of automation and major subsystem upgrades in System Engineering		
Citation	Applicability to automation upgrades	Limitations
Broas et al., (2021)	• Focuses on identifying key new technology or innovations and applying them efficiently.	• Did not address specific challenges such as programming upgrades with short commissioning window
Kutscher et al., (2020)	• Defined methodology to upgrade automation systems using systems engineering lifecycle.	• High level and does not address specific challenges with heavy industrial automation upgrades
Maier et al., (2020)	• Follows Department of Defense MOSA methodology to ensure integration of automation components for ease of future upgrades	• Focus is only on aerospace industry

Case studies of major automation upgrades

Finally, a literature review was completed concerning similar major automation projects and techniques used to ensure specifications were met prior to cutover. One interesting publication was the discussion of DevOps in upgrading automation system for automobile manufacturing where they established three main takeaways from the upgrades and recommended systems development practices that should be followed:

> "(i) software development team attaining ownership and responsibility to deploy software changes in production is crucial in DevOps. (ii) toolchain usage and support in deployment pipeline activities accelerates the delivery of software changes, bug fixes and handling of production incidents. (ii) the delivery speed to production is affected by context factors, such as manual approvals by the product owner (iii) steep learning curve for new skills is experienced by both software developers and operations staff, who also have to cope with working under pressure" (Lwakatare).

Other case studies of major automation upgrades focused on leveraging digital twins to help facilitate the upgrades. The use of digital twinning was utilized to help facilitate the continual upgrade of sites: “the automated production plants are continuously optimized during the detail planning phase as well as after the start of production as a result of improved processes and model upgrading” (Biesinger). This work leverages creating a digital twin (or mirror image of the site in virtual reality) to help facilitate upgrades. Their work conceded that ongoing work needs to occur to refine the process of utilizing this method to continually improve the process for automation upgrades. Further work and publications reviewed leveraged the use of digital twins to enable major upgrades to automation systems. One example showed that (Redelinghuys) using templates for upgrades could have benefits to the execution of the upgrades. This work established a reference architecture for modern automation systems that can serve as a mode for vertical and horizontal integration. They implemented this architecture for a small industrial site to document the ease of upgrades with regard to automation and leveraging a digital twin. Table 6 summarizes some of the initial literature reviewed that recently focused on case studies and the methods employed for industrial settings. Some of the notable features of the recent publications are a focus on mutli-site upgrades and focus on major programming upgrades in a production environment. Both these attributes led to similar challenges of this work such as each site having unique characteristics and inability to have a template design to upgrade each. Every site was unique and had its own challenges for an automation upgrade. Some of the limitations of the upgrades are that planning around shutdowns was not addressed and that programming was not specifically addressed except at a high level. As mentioned previously, digital twinning was often leveraged to assist with the upgrades. Some of the gaps of the recent research is not a heavy emphasis on heavy industrial sites. This literature review showed that a greater emphasis needs to be placed on more diverse industries beyond automotive.

Table 6: A literature summary of recent automaton upgrade projects

Recent case studies of major automation upgrades for industrial sites		
Citation	Applicability to automation upgrades	Limitations
Lwakatare (2019)	• Multi-site upgrades completed • Acknowledges and documents the challenges associated with upgrading systems in operation and manufacturing state of systems engineering lifecycle • Focus is on major programming upgrades in production environment	• Planning around shutdowns is not addressed
Biesinger (2018)	• Utilizes digital twins to plan major automation upgrades to industrial sites	• Focus is on automotive industry • Conceded that the method needed further development
Redelinghuys (2020)	• Leverage of digital twins to make major upgrades to automation systems • Use of template network architectures to make upgrades simplified	• Goal is to improve manufacturing but does not address specific goals of digital twin beyond integration into Industry 4.0

The literature review of the challenges faced with automation upgrades for industrial sites revealed several key research strengths and weaknesses. The strengths of the literature in reference to the problem sets identified for this research are:

- Methodologies exist for incorporating subsystem upgrades for automation in existing systems engineering framework,
- Leverage of digital twin technologies and concepts to aid the upgrades,
- Working with automation manufacturers to ensure future products are designed for simplified upgrades,
- Use of risk-based techniques to upgrade obsolete equipment.

The weaknesses include:

- For risk-based techniques for identifying automaton upgrades, the emphasis is on mechanical parts and equipment with known end-of-life dates based on corrosion and other mechanical factors,
- Overly complex risk-based techniques that may be difficult to implement continually by operations personnel for multiple sites,
- Heavy focus on aerospace and military applications and little focus on manufacturing systems,
- Specific challenges faced by heavy industries with continual operations such as planning around shutdowns for upgrades,
- Continual audit of automation systems to identify deficiencies and automation obsolescence continually.

With both these strengths and weaknesses in the current research and literature addressed, one can extrapolate some key research areas to focus on to aid my research.

- A specific risk-based approach for identification of obsolescence in automation and cyber physical systems that combine both the uniqueness of the systems and easy to implement for operations personnel to quicky identify components to upgrade is needed.
- A defined way to do controls and other programming testing to aid short commissioning windows is needed. This includes simulation testing and methods to ensure efficient testing is done prior to commissioning.
- Establish a continual monitoring system for automation to audit the system to ensure deviations from functional requirements are quickly identified. This can leverage a

digital twin approach to tie the system of record with operational data to aid the upgrades.

These three key weaknesses with the current research that form the basis of this dissertation and research. The first challenge addressed with establishing a risk-based approach to identify obsolescence. After that, a method to do programming and system testing in a lab environment was developed to aid major upgrades prior to short commissioning windows. Finally, a method to audit the automation and cyber physical system was established to aid future upgrades. The first focus though is on obsolescence management for industrial automation subsystems. Chapter 4 explores the research done to address the challenges of doing proactive replacement of parts and components. Each following chapter includes further literature reviewed to aid the research performed, theory, and tools developed.

Chapter 4 – Risk-based approach for managing obsolescence for automation systems

Introduction

Obsolescence is an important consideration in management and engineering of heavy industrial facilities. Examples of these sites include refineries, chemical plants, and other materials processing sites. These facilities typically stay in operation for many decades but operate on software, automation hardware, and other electrical equipment that reaches obsolescence much sooner than the rotating equipment, unit operations, and other major systems that make up the facility. This work proposes a way to manage obsolescence of automation and electrical systems at industrial sites. This risk management approach was developed by first analyzing a project database for parts that were replaced due to obsolescence. The analysis showed that by cataloguing the average lifespans of equipment, their criticality in operations, their manufacturers continued production, and other factors, a prioritization for replacement of obsolescent parts can be created. Based on those results, a taxonomy and risk assessment plan were developed to manage the replacement of parts. The value of this proposed management strategy was validated through its application to several industrial sites. The results indicate a reduction of roughly 70% of reactive replacements due to obsolescence after the major upgrade and a 24% reduction in unplanned downtime due to part failure during normal operations. While this study focused on heavy industries, this process for identifying obsolete components can be applied to other industries and systems.

Industrial control systems are the automated systems that control the individual unit operations and rotating equipment in an industrial operation. They include instruments, logic controllers, motor control centers, transformers and more, which control things like tanks, distillation columns, and other major systems. Most commonly, input signals from instruments, pumps, or other parts of a facility are read by programmable logic controllers (PLC), logic is solved, and outputs from PLC are sent to devices which control the plants operation, including human machine interfaces and operator terminals (Alphonsus). Often controls and automation equipment are installed with an expectation that they will run continuously for greater than 10 years, but the equipment they automate, such as tanks and distillation columns, can last much longer (P. V. Sandborn). This sets up a challenge in heavy industries: there needs to be continual obsolescence management of automation systems since they need to be replaced at a higher frequency than rest of facility's major systems. Maintenance of the industrial control systems ensures that the facility stays running, but their nature as software-intensive cyber-physical systems means that they will require management and upgrades over the lifetime of the industrial system to avoid obsolescence (Cesar).

Obsolescence can be defined as when an entity is becoming outdated or no longer useful and in this way is considered inappropriate (Marc). The definition of obsolescence as applied to this work is when associated equipment no longer has the reliability to meet functional requirements and specifications of the system either from end-of-life or manufacturer discontinuation (which would result of inability to replace upon failure). The research into the obsolescence of equipment at several industrial sites was used to understand that a long-term series of projects should be undertaken to upgrade the control systems, programming, and associated automation equipment during planned facility shutdowns. A multi-year program was undertaken to prioritize and then upgrade 14 industrial sites across Southern California. This involved taking aging systems, with obsolete programming, and upgrading to newer systems, and measuring the costs and benefits of this new obsolescence management program.

Obsolescence management is a key focus of systems engineering and literature discusses KPIs and methods to manage analytically (Akhtyamov). In addition to this method, the intricacies of obsolescence upgrades and how to modify the systems engineering methods should be considered when designing a new system. Previous work has shown that life-time buys (or buying enough spare parts of automation equipment to outlast the lifespan of mechanical parts) can be infeasible (Pecht). and this method outlines a flexible, efficient way to identify part to upgrade.

This obsolescence challenge exists in many industries when it comes to maintenance planning and replacement of obsolete equipment. Some similar studies and work include the US Air Force F-15 program which developed techniques to schedule maintenance to minimize economic loss by grouping large maintenance activities together instead of many smaller ones (Langford). Other recent work includes advanced models to predict when a site needed maintenance to be performed for specific times for each component based on risk ranking equipment (Elwerfalli). Other studies have done excellent work with specific platforms (such as military or space aircraft which designs are controlled) to design in features to prevent obsolescence (Katz) for automation components. Other studies have done this for specific sites and complex systems such as space systems where they analyzed the entire system to address all obsolescence issues (Tobias). Most heavy industrial sites (such as refineries and chemical plants) are unique in their design, so this technique was developed with them in mind and leveraged data from project databases of upgrades at their sites. The goal of this work is to address the challenge of obsolescence planning for complex systems with a generalized approach.

Methodology

The current practice for many industrial sites is to schedule major outages/turnarounds around a particular piece of equipment that needs maintenance and outage will force rest of site to shut down. Then maintenance and engineering personnel arbitrarily select other equipment to replace what they perceive as obsolete based on their knowledge. The current practice has developed to have methodical methods for identifying obsolete equipment (Bertolini) but what these methods lack is an analysis of the manufacturer of the product and lifespan of equipment. Most state-of-the art methods focus on consequence of failure for establishing risk-based inspection. Recent publications focus on assigning grades to automation equipment but again focus on failure consequences (Cesar). This method presented here proposes focusing on turnaround planning and obsolescence from age of equipment, manufacturer of automation and electrical equipment, and other obsolescence factors. Admittedly, obsolescence management in heavy industries has room to improve due to disadvantages it faces. Automotive, aviation, health care does not have the disadvantages faced by chemical plants, refineries, and other large sites: small windows for obsolescence upgrades due to outages and equipment that can stay in service for decades before being replaced. The current practice in heavy industry for obsolescence management is to base obsolescence identification on local knowledge and risk of device failure. This study looked at several other factors to expand on current practices. The gaps for obsolescence management for COTS (commercial of the shelf) in CPS (cyber physical systems) was studied recently and identified several gaps:

> "One of the key findings of our research effort is that there are no studies which focus on existing obsolescence management strategies specifically for COTS-based CPS systems…Offsetting "obsolescence" to reduce performance risk and maintain systems reliability, availability, maintainability, and cost-control is an underpinning competency of systems engineering, with importance across all system life cycle phases." (Alelyani).

Obsolescence, for the purposes of this study, is defined as when a part (material or technology) that is needed to manufacture or support a product or systems is not available from existing manufacturer of the part (material or technology) (Bartels). This can include lack of programming resources for the programming language, which was often a justification for upgrading control systems within the industrial controls application. It can also mean when the reliability of sub-components or subsystems have decreased to the point where replacing the entire system is economically justified (P. Sandborn). A key distinction must be made between upgrading components due to reliability versus obsolescence. For many industrial control components, it is common to replace obsolescent components and their associated systems with newer ones once the poor reliability of the original component can justify the upgrade.

This distinction is salient for this study because reliability considerations are what can justify the shutdown of an industrial site for major maintenance, at which time obsolescence management can take place. So, these two sources of cost and benefit must be considered together, as reliability considerations will dictate when the obsolescence management can be performed, the reliability considerations will determine which systems will be replaced or upgraded as part of obsolescence management.

Reliability Analysis for Determining Site Shutdown and Replacement Priority

For this research, the timing and duration of site shutdown for major maintenance is determined through a risk-based reliability analysis. In addition, the risk of failure is also included in prioritizing the replacement of obsolete equipment, where components at higher risk of failure are prioritized.

The classical model on which a risk-based reliability analysis can be based is the Weibull distribution and its associated "bathtub curve". The Weibull distribution mathematically is given by:

$$f(t) = \left(\frac{\beta}{\eta}\right)\left(\frac{t-\gamma}{\eta}\right)^{(\beta-1)} e^{\left(-\frac{t-\gamma}{\eta}\right)^{\beta}} \quad \text{Eq. 1.}$$

Where the Weibull parameter *β* is the slope which can be modeled as the rate of failure for a component, *η* is the scale parameter or characteristic life (life at which 63.2% of the population will have failed), *γ* is the location parameter, and *t* represents time. For the purposes of this study, we consider only components that are failing due to age related degradation, modeled as *β*>>1).[1] Reliability functions are derivable from Eq. 1 and derivative equations. The survivor function is the probability that an item is functioning at time (*t*):

$$S(t) = e^{-(\eta t)^{\beta}} \quad \text{Eq. 2}$$

The failure rate (failures per unit of time) is then given by:

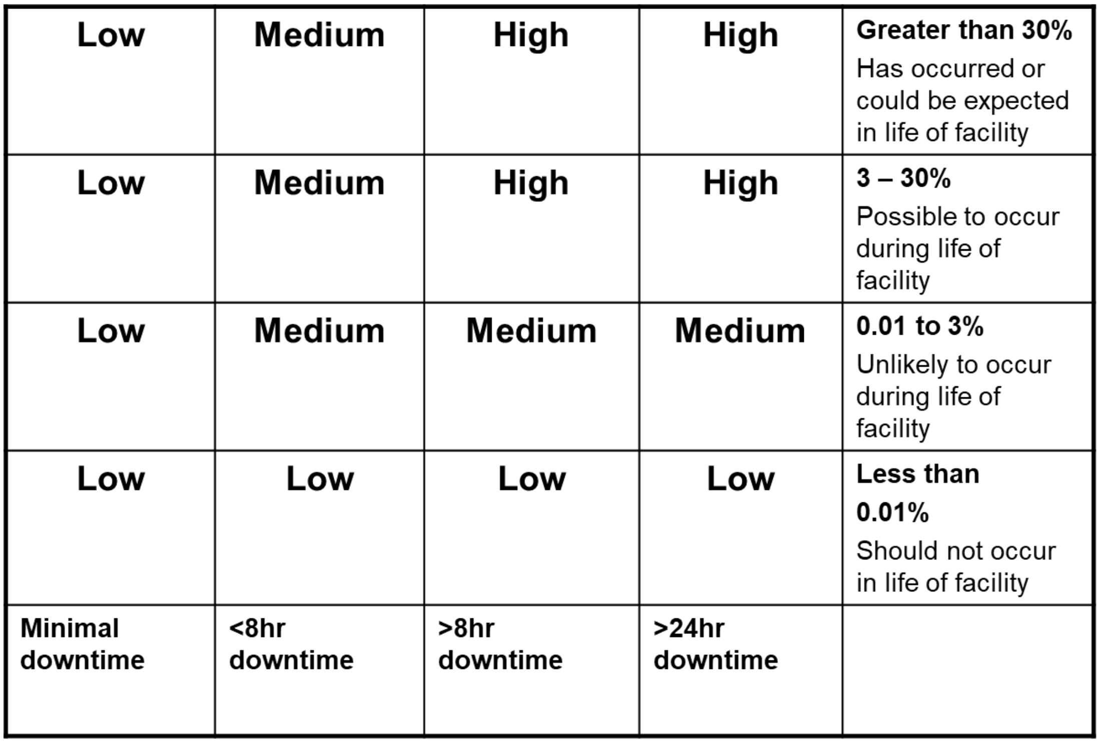
$$\text{Failure rate} = \frac{f(t)}{S(t)} \quad \text{Eq. 3}$$

Based on this model of component failure, we can construct a preventive replacement strategy for scheduling of site shutdown for major maintenance, and for prioritization of the replacement of obsolescent components, as illustrated in the following sections. As failure rates increase, facility down time increases substantially (Melchor-Hernández) so increased risk scores

[1] For component failures that occur during the "infant mortality" or "constant failure rate" regimes of the bathtub curve, these facilities use a reactive maintenance model based on allowing the components to run to failure. Only near end of life (modeled as *β*>>1) do these facilities perform proactive maintenance.

were assigned as parts moved closer and beyond their lifespan and were expected to have higher failure rates.

Typically, industries utilize a risk matrix (example shown below) to determine and communicate the acceptable risk when designing facilities. A simplified version for reliability is shown below (Figure 23). Companies typically design to ensure low-medium risk for downtime, if there is a high risk then the design is changed to lower the risk (Woodruff). The risk matrix and traditional equations for reliability are combined to generate the method outlined in this dissertation by (1) Assuming heavy industry sites are designed to the minimum acceptable risk threshold (2) Analyzing reasons for increased failure. The failure of automation and electrical equipment was surveyed for several industrial sites detailed later to find causes for failure. Then, combined with bathtub curve and failure rate equations, the method to identify equipment to replace during a shutdown was developed since any significant change in age of equipment, service condition, redundancy, and ability to replace equipment would result in the site transitioning from acceptable to unacceptable with regards to risk profile. The risk-based approach was used to tie back to the traditional bathtub curve by showing that significant increases of risk can occur when high components start to age, deteriorate, or an unable to be replaced. This is what moves the risk profile of a site from its original design to unacceptable levels of reliability. The methodology outlined in this chapter is based on determining what caused components to be replaced (age, vulnerability, manufacturer’s discontinuation) and simplifying reliability models for maintenance. In short, failure rates increase exponentially at end of life (eq 1.), which greatly affects acceptable risk profile of a site (fig 23), and operators need a simplified way to identify components to replace during turnaround cycles. When components age or are unable to be replaced, they move the risk profile of a site to an unacceptable risk level and action should be taken.



Low	**Medium**	**High**	**High**	**Greater than 30%** Has occurred or could be expected in life of facility
Low	**Medium**	**High**	**High**	**3 – 30%** Possible to occur during life of facility
Low	**Medium**	**Medium**	**Medium**	**0.01 to 3%** Unlikely to occur during life of facility
Low	**Low**	**Low**	**Low**	**Less than 0.01%** Should not occur in life of facility
Minimal downtime	**<8hr downtime**	**>8hr downtime**	**>24hr downtime**	

Figure 23: A typical risk matrix used for evaluating reliability for an industrial facility

Survey of Obsolescence at Industrial Process Sites

To understand and then prioritize the characteristics of obsolescence in the industrial process facilities that are the subject of this research, historical project databases were analyzed and computerized maintenance system databases to determine which components and systems were highlighted as obsolete and why. For all these projects, only reactive obsolescence management strategies were employed.[2] The down-selected data set that was used for analysis included roughly 4000 projects and $3 billion in capital costs for projects that were replaced solely for reactive obsolesce reasons. These reactive obsolescence projects were looked at for

[2] Reactive obsolescence means that a repeated failure of an obsolescent component was causing system downtime, and lost revenue. The upgrade from an obsolescent component to a more modern component could therefore be justified based on economics.

their age, criticality, and other features to see what common trends there were between the obsolete parts.

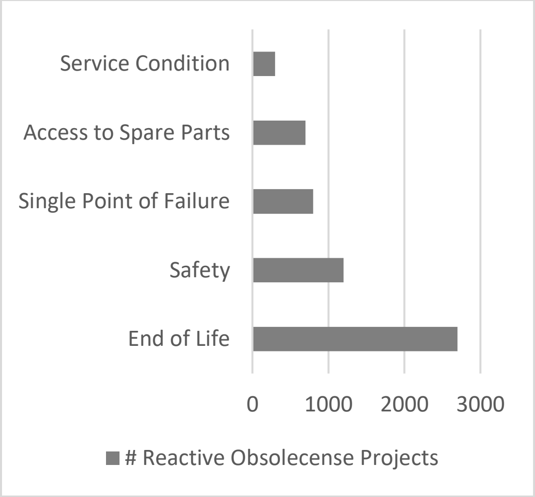


Figure 24: Researched Causes of Reactive Obsolescence

Because these records are historical in nature, they do not represent an impartial or infallible source of data regarding obsolescence in these facilities. In periods of higher margins, there is typically more investment in upgrading obsolete equipment both due to improved returns on investment and fewer capital budget constraints (Alpanda). From the analysis of available data, it was noted that reactive upgrades of obsolete systems were much more common than proactive upgrades. Figure 23 shows the results of analyzing project management databases for parts and components that were replaced due to obsolescence. Figure 23 shows the importance of reliability considerations (safety, end of life, and single points of failure) in enabling obsolescence management, which can be contrasted to a baseline understanding of the value of obsolescence management, which usually focuses on metrics of maintainability or replace-ability. The next section focus on connecting the data to a simple and methodical way to identify which parts and components will required to be upgraded at the next planned shutdown based on these identified characteristics.

Risk-based approach for proactively managing obsolescence

From this dataset, a proactive risk management strategy was developed to proactively assess all parts of a facility so that obsolete parts could be replaced during plant turnarounds. Using the results of analysis of the project database for replacement due to obsolescence, it was determined that asset obsolescence should be prioritized using these categories: Consequence of Failure, Single Point Vulnerability, Spare Parts Availability, Service Condition, and Age of Equipment. A taxonomy of the automation components that are present in the entire facility and their lifetimes, enables that calculation of an obsolescence risk.

Such a taxonomy was developed as well to track the different types of automation equipment (Table 7) which gives the classification of different automation/electrical components and a range of expected lifecycles (LC). This taxonomy was developed for all automated equipment and their expected lifecycle. The purpose of the taxonomy was to be able to group similar electrical and automation equipment together. The resulting catalog was used to provide an integration into computerized maintenance systems. The taxonomy also provided a way to organize work and crews during turnarounds by knowing how many similar components (analyzers, computer systems, power distribution) had to be upgraded.

The lifetime of components throughout the facility was also calculated from the datasets available. The most common reason for reactive replacement of obsolescent parts was "end of life" due to either voltage, temperature, or environmental factors. In the data extracted from the survey, the "end of life" is often much sooner than what the manufacturer's stated number of hours a device can run due to variations from the manufacturer's optimal setup where device is installed (Zhang). Manufacturer's often give an expected number of hours the device is expected to run, but this is based an optimal voltage supply, temperature, and other environmental factors.

The life cycle for each piece of automated equipment was based on how often they have failed in service after running for several years in situ, at the conditions of operation of these facilities.

Table 7: A sample of taxonomy developed

Taxonomy	Asset Type	Life Cycl (years)	0.9 xLC	1.2x LC	1.5 x LC
	Electrical				
637	UPS	15	13.5	18	22.5
610	Micro Processor base protective relays	15	13.5	18	22.5
610	Micro Processor base protective relays	20	18	24	30
630A	LV Adjustable Speed Drives	15	13.5	18	22.5
630B	MV Adjustable Speed Drives (ASD)	20	18	24	30
624	DC Motors	30	27	36	45
621	LV Motors	30	27	36	45
622	MV Motors	40	36	48	60
NA	MV Power Cables (UG)	40	36	48	60
638A	Battery Chargers	20	18	24	30
608	LV Circuit Breakers	30	27	36	45
607	MV Circuit Breakers	40	36	48	60
	Instrumentation				
763A	Control Valves	30	27	36	45
763C	On/Off Valves	30	27	36	45
783	Compressor/Electronic Controls	20	18	24	30
721, 727B/C/I,	Electronic Transmitters	20	18	24	30
720	Ultrasonic Flow Meter	20	18	24	30
786	Vibration System	20	18	24	30
784	Programmable Logic Controllers (PLC)	10	9	12	15
	Analyzers				
702A	Analyzers – Complex GC	15	13.5	18	22.5
702C	Analyzers – Mass Spec	15	13.5	18	22.5
702E	Analyzers - Combustibles	15	13.5	18	22.5
702R	Analyzers - O2	15	13.5	18	22.5
702V	Analyzers - Viscosity	15	13.5	18	22.5

Table 8: Obsolescence score determination

Obsolescence Score	
SINGLE POINT VULNERABILITY	**Weighting**
Asset is a single point vulnerability	1
Asset has some redundancy or work-around available	2
AVAILABILITY / ACCESSIBILITY TO SPARE PARTS AND SOFTWARE/FIRMWARE UPDATES	
Manufacturer no longer supports equipment, spare parts, software/firmware updates AND Spare parts may be in storehouse but cannot be obtained from OEM (original equipment manufacturer) or after market	1
Manufacturer no longer supports equipment, spare parts, software/firmware updates AND Spare parts are in storehouse or can be obtained after market	2
Equipment is still supported by the Manufacturer OR Manufacturer has announced future end of support date (Spare parts still available from OEM)	3
Consequence of Failure (Aligned with RTP Consequences)	
Significant asset loss, damage and/or downtime. Costs >$1,000,000 to $10,000,000 or Fire > $250k damage or Shutdown of a principal unit	1
Some asset loss, damage and/or downtime. Costs >$300,000 to $1,000,000 or Fire > $1k damage or Minor unit shutdown with effects to other units	2
Some asset loss, damage and/or downtime. Costs $100,000 to $300,000 or Fire < $1k damage or Reduced feed rate at principal unit or Product off-spec	3
Minimal damage. Negligible down time or asset loss. Costs $30,000 < $100,000 or Reduced feed rate at minor unit	4
Minimal damage. Negligible down time or asset loss. Costs < $30,000 or Process unit upset	5
Production Likelihood Guidance	
SERVICE CONDITION OF EQUIPMENT	**Weighting**
Severe Service - Severe is defined as subject to high ambient temperature, outdoors and prone to dust/dirt accumulation.	1
Mild Service - Mild is defined as indoor, in a climate-controlled environment with low likelihood of dust/dirt intrusion.	2
AGE OF EQUIPMENT (Reference Expected Life Cycle Table above)	
Equipment is significantly past its expected equipment life (≥1.5x expected Life Cycle)	1
Equipment has exceeded its expected equipment life (≥1.2x to 1.5x expected Life Cycle)	2
Equipment is close to or has exceeded its expected Life Cycle (≥0.9x up to 1.2x expected Life Cycle)	3

Table 8 is used determine obsolescence score based on different characteristics of different electrical and automation equipment. This table was created based on identification of common recurring causes of lost revenue and other consequences. Using this information, we developed an approach to proactively prioritize the replacement of obsolescent parts in these facilities as follows:

- Utilizing Table 8, determine the evaluations for each of the criteria. Use Equipment Typical Life Cycle Data table from Table 7 as required,
- Add evaluations from each category to obtain an overall "Obsolescence Score" ,
- Utilize Table 9 to determine the appropriate required action.

Table 9: Recommended Actions based on Total Obsolescence Score

Obsolescence Score	Recommended Actions
7 or less	Notify Operations Management. If appropriate, add threat to site watch list. Develop a recommendation to mitigate. Procure replacement for obsolete part in case of unplanned failure of part. Add part to replace during next site turnaround.
8-10	Review Obsolescence score annually and plan to upgrade during next turnaround/planned shutdown
11 or greater	Review Obsolescence score annually and document any changes to score

This procedure was used to prioritize specific systems and subsystem for proactive obsolescence management. The risk management method follows typical industry risk management method for reliability where risk tolerance was defined by lost revenue. The obsolescence scores were based on a threshold where replacements would be needed to avoid unplanned shutdowns.

Proactive obsolescence management procedure

After the parts were identified to be upgraded from obsolescence, another challenge that was addressed was when to do these obsolescence upgrades. For this study, all obsolescence management procedures were performed during major shutdowns that are part of the typical turnaround cycle. Re-engineering the programming of a plant controller can take almost a year of work depending on how the engineering is staffed. Due to the importance of automation on the operations of the facilities, the timing of these obsolescence upgrades needed to be either done by shutdown (loss of revenue), some in segments where possible, or coordinated with a typical turnaround cycle where maintenance was done on the rest of the facility.

In the case of this work, the sites identified to have major reliability issues with regards to automation were ranked. The most at-risk sites had their automation system upgrades completed first. This allowed for budgeting and forecasting to ensure that parts and labor are available to coincide with the turnaround cycle. Once obsolete components were identified and the decision was made to upgrade them, the next step was to select the correct new components for the automation system to replace them with. This was done following the systems engineering process for the automated systems. By following the systems design processes and paying attention to key interfaces between IT, electrical distribution and process equipment, the design, acquisition, and startup of the automation system can be performed efficiently (Blanchard). There are things to pay attention to when creating the functional specifications of an automation system (such as room for expansion, redundant modules, and power supplies, etc.) but this can be handled following the systems engineering process to list new system requirements and designing accordingly.

Results and Discussion

Based on this analysis, this proposed procedure was then implemented at 14 industrial process facilities throughout Southern California. At the time of publication 14 sites were completed but only 13 were able to collect data due to the long period of time to assess results (2 years of continuous data was gathered). Tracking of the costs of maintenance, of the amount of unplanned downtime was performed both before and after the implementation of this procedure to allow for the measurement of the baseline and change in these key performance indicators. To determine the effectiveness across several sites we normalized each site based on previous year's budget. In other words, each industrial site is unique, so we took previous year's obsolescence budget and runtime and established that as normal then compared it to after the major upgrade to see improvement. The normalization strategy was to compare the previous 2 years' obsolescence costs (replacement due to end-of-life and manufacturer discontinuation) with 2 years after upgrade. The normalization equations are shown below in equation 4-6:

$$\frac{\text{Obsolesence costs for 2 years after upgrade}}{\text{Obsolesence costs for 2 years prior upgrade}} = \text{normalized cost} \quad \text{Eq. 4}$$

$$\frac{\text{Runtime (hrs) actual}}{\text{design expected runtime (hrs)}} \text{x100} = \text{normalized runtime} \quad \text{Eq. 5}$$

$$\frac{\text{Runtime (hrs)for 2 years after upgrade}}{\text{Runtime (hrs) for 2 years prior upgrade}} \text{x100\%} = \text{\% improvement} \quad \text{Eq. 6}$$

Figure 25 illustrates the measured normalized cost of each site's expenditures for upgrade of obsolete equipment after this method was applied. It shows using the proposed obsolescence management procedure at 13 industrial process facilities (with the 14th and last site still undergoing the upgrade and collection of results), costs are measured for 2 years post implementation. A score of 80 means there was an 20% reduction in the budget of previous year was applied to upgrade obsolete equipment over a 2-year period after the identified obsolete equipment was upgraded using this process. These costs are normalized per site, since the size of each facility varied and therefore the capital spent on obsolescence varied. These results show that for each of the 13 sites where this obsolescence management plan was put into place, the costs of obsolescence management was decreased at every site. Average improvement across the sites was a 30% improvement, and at some sites, the cost of obsolescence management was decreased by 60% as measured in the 2 years after implementation of this program.

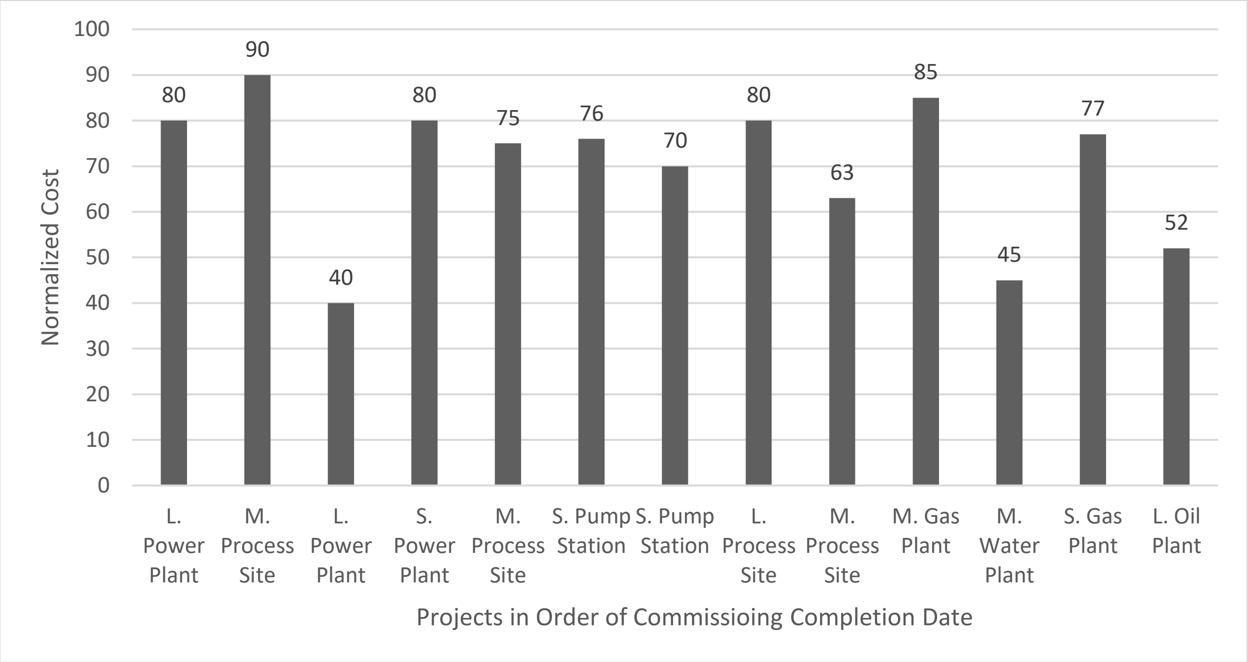


Figure 25: Normalized capital costs using method

Figure 26 illustrates the measured normalized run-time for the facilities that were the subject of this obsolescence management program. The percent improvements represents the normalized improvements for runtime over a 2-year period after upgrades of obsolete equipment. A normalized run time of 90 means that the facility automation was running for 90% of the required operational hours of the two-year measurement period. For each site, Figure 26 presents the normalized run time for the 2 years prior to the implementation of the obsolescence management program at major upgrade, and the normalized run time for the 2 years after. The relative improvement is also graphed for each site. These results show that for each site, the normalized availability of the facility was equal or improved. For some sites which were particularly prone to unplanned maintenance, that improvement is measured as an improvement by up to 7% in availability. This is equivalent to a 24% decrease in unplanned shutdown time due to failed parts and reactive obsolescence.

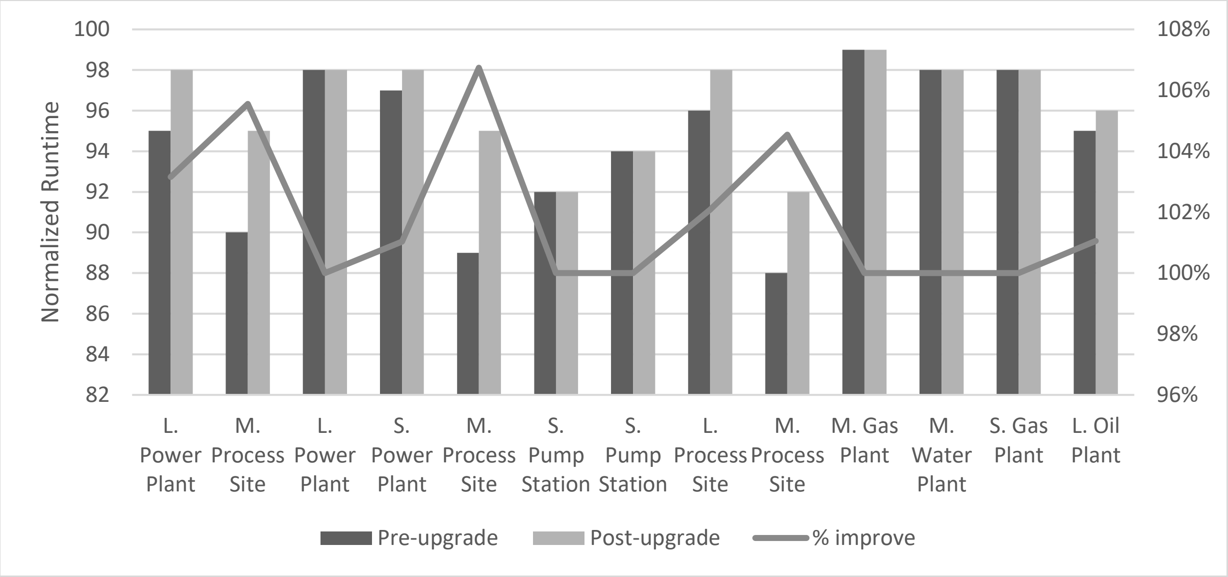


Figure 26: Normalized improvements for runtime

The results of this study have demonstrated the value of an integrated systems approach to proactive maintenance and obsolescence management. For the application of heavy industrial automation systems, this program of active obsolescence management must take into account both the very high value of process time, the high capital costs associated with the process equipment, and the relatively short obsolescence timeframes that affects the electronics and control equipment. Certain major components and systems can be close to a hundred years old such as piping and major mechanical sections of equipment. On the other hand, control systems, information technology, instruments, and other systems and large components must be replaced regularly due to the high cost of repair and replacement as they reach obsolescence. For this proposed procedure, these constraints are recognized and evaluated in obsolescent systems management, where replacement of components and parts are undertaken on specific subsystems, replaced during a planned shutdown, and the bulk of the process system stays in place.

More qualitatively, field engineers and maintenance personnel generally liked the method for assisting with major shutdown planning and efficient decision making. Being able to plan for a planned outage a year in advance then quickly scanning computerized maintenance system for end-of-life equipment and manufacturer's discontinuation to quickly generate a scope of work allowed field personnel a starting point to accomplish a complex task. This work was meant to assist shutdown planning but researchers in field can utilize the results to show that expected lifespan of electrical and automation equipment can drive reliability. Traditional studies on system reliability focuses on positioning of equipment but this work hopes to strengthen the concept that there is a time dependency for reliability on complex automation-based systems. The definition of obsolescence as applied to this work is when systems cannot meet functional requirements (including reliability) and action must be taken to replace them. For heavy industrial sites, the definition of obsolescence includes replacing parts were their reliability (either through age, manufacturer discontinuation, or other risk factors we included in study) significantly deteriorates to point that system no longer meets its functional requirements.

Conclusions

This research has sought to develop a practical program for assessment and implementation of an active obsolescence management system. This work analyzed a project database to prioritize obsolescence projects that could be performed in synchrony with major shutdowns. The weighting criteria to identify and prioritize obsolete systems was based on reasons recorded during historical reactive replacement of parts. By now identifying and prioritizing these obsolete systems actively, this project was able to demonstrate reduced costs, and improved uptime as measured in process facilities across Southern California industrial sites. This investigation has provided information to answer the first research question: With inability to know end of life for electrical equipment, can there be methods to risk rank obsolete equipment to identify for next upgrades?

The research contributions of this study in particular are

- Proposed, developed, implemented, and documented results of a risk-based obsolescence management system for automation and cyber physical systems in an industrial setting
- New theories developed for obsolescent management as a risk-based approach as compared to traditional methods for "design against obsolescence"
- Embeds risk-based obsolescence management into systems engineering lifecycle

This research can be used both as empirical validation of the value of active obsolescence management strategies, and as a demonstration of the simplification and application of these strategies to heavy industry. Future research will focus on the generalization of these methods and results to consideration of obsolescence in these facilities, across the systems engineering lifecycle.

This work was based on both field experience and recent studies that showed grouping maintenance efforts if often cheaper than individual maintenance activities (Do Van). It also was inspired by other attempts to simplify the prediction of replacing components before failure for operations (Laggoune) and computer-based systems obsolescence upgrades (Ahmad). Obsolescence for automation systems can be viewed with long time period (sometimes 10-15 years between upgrades) and this method gives operations engineers an easy and quick way to establish a obsolescence management program and baseline their system with just basic information on age of equipment, redundancy, and ability to replace components. With the components to replaced identified, the next step was to redo the programming for the new automation systems. This had to be done in a laboratory due prolonged downtimes being unacceptable for the upgrades. A method was developed to QA/QC the programming prior to commissioning. This method leveraged dynamic simulation with code checkout.

Chapter 5 – Optimizing the Use of Simulations for Commissioning

Introduction

This work began during a risk assessment of industrial sites. It was discovered that key facility control systems were obsolete and had a high risk of unplanned shutdown. A multi-year program was undertaken to upgrade the industrial sites. This involved taking aging systems, with obsolete programming, and upgrading to newer systems with a newer programming language. The main stakeholder's concern was that a shutdown of any operation for the upgrade would result in lost revenue.

This chapter describes how these system upgrades utilized simulation with defined interfaces to the new logic controllers to aid in the quality checking of programming and startup of new systems. A unique metric of performance was developed for these simulations as well to benchmark results based on the complexity of each facility. A variation to a traditional Delphi method was utilized to evaluate the most efficient way to perform simulation and testing via an objective analysis and results show a 40% improvement in simulation cost efficiency over the span of the multi-year upgrade to ten industrial sites.

As an aside, I have led teams developing simulations for large scale industrial projects such as chemical plants, power generation, and other large-scale systems. My observations are that when simulations are first developed, the simulation team often struggles to control the scope, to ensure we directly addressed design questions, and to keep quality controls on the work to ensure the simulations added value to the system development. Simulations are often charged with reducing the risk to the project, but when simulations were called for to assist with the design and construction of new control systems, many on the team questioned the strategy.

There were questions on the value of developing simulations that might be overly complex and not addressing any real issues with the upgrades. Some stakeholders gave negative feedback to the idea of using simulations for control system design due to past simulation work they experienced did not directly address the design questions. The result of this critical feedback was a simulation process that was particularly responsive to needs. We developed a simulation procedure to QA/QC programming prior to commissioning, which was well received by stakeholders after the upgrade was completed. A methodical process was developed over the course of many large-scale industrial projects to upgrade control systems for chemical, power, and other industrial sites to improve the quality of simulations. When the project began, there was a lot of uncertainty in how to incorporate simulations effectively into existing processes, but after developing process and tools outlined in this chapter, the costs came under control, design questions were resolved, and quality control of the simulations enabled maintained confidence in the programming upgrades.

Trust in simulations can be difficult to achieve for a complex system. It often is a tradeoff between accuracy and cost that a team must consider when undertaking a simulation or model to support the development of a complex system. The ability to combine all the complexity of the real world into a computer model is difficult to accomplish along with the maintaining the team's unbiased influence on the outcome of the simulation results. The use of a methodical process and analytical tools ensure that the simulation adds value and directly addresses the major risks to the system being developed. The following are the highly condensed results of this chapter:

- Performed an objective analysis and functional specifications of simulations for industrial control systems to document the risks needed to be addressed and resolved by the simulation.
- Developed, adopted, and maintained a simulation philosophy to be a living document to track the design basis, risks being assessed, and data throughout systems engineering lifecycle.
- Collected data and benchmarked the simulations of industrial systems to provide continual improvement both to costs and accuracy.

Systems engineering is a field that focuses on the development of complex systems that require management of development from concept, detailed design, production, and lifecycle management. Systems engineering takes conceptual ideas through methodical processes of preliminary design, detailed design, testing and commissioning, and eventually production. A subset of systems engineering discipline is systems test, evaluation, and validation. This area of the field covers the planning of testing to ensure design and performance requirements are met and data analysis of test reports. "Modeling and simulation" are key components for development of a system regarding systems testing since it allows a cost-effective way to test a system virtually without a large investment. It can be utilized at any point during the systems engineering process as an effective means to test to see if the requirements of the system are being met. The INCOSE Systems Engineering Handbook gives guidance on the use of simulations. For example, "Large systems may justify the development of high-level simulations evolved from the simulations architecture. The simulation should contain sufficient functional elements that the interactions can be properly assessed" (Shortell). The goal of the simulation is to give guidance on performance where testing to realistic conditions cannot be achieved or is not cost-effective. This chapter is seeking to give systems engineers guidance on how to apply simulations to their work, how to measure simulation accuracy, and how to gauge if the simulation is cost effective with regards for industrial programming integrity checking with simulation.

No matter how expertly or detailed a simulation is, there is always a level of trust that goes along with utilizing simulations since they are by their very definition a simplification of actual physical system being simulated. This introduces an important topic of both systems engineering and simulations: verification and validation (V&V). Rigorous methods for V&V for simulations have been developed that ensure the range of accuracy of a simulation is understood. These techniques include consideration of both the data used in the simulation and the simulation itself. Methods of V&V attempt to ensure that the simulation meets its operational requirements and stakeholders understand the robustness of its results (R. G. Sargent). The architecture of simulations can vary for systems as well. A defined simulation architecture aids the team knowing what interfaces need to be defined and standardized throughout the project. A reference architecture for simulations can also be utilized (JE Hannay). Human performance can also be a large factor in a simulation and efforts are ongoing to integrate human factors into traditional physics-based models (M. Watson). Another key concept for utilizing simulations in system development is key defining roles of each member and the scope of the simulation. It is also important to document models required to be built that will test the simulation against the defined system requirements. There are methods to developing model-based design verifications by simulation that includes the system requirements (W. Schamai). Beyond simulations used to test system requirements, often a simulation is utilized in training for similar reasons it is used in systems development: cost, flexibility, and risk aversion. Simulation used for training is also being developed for systems engineering. The Systems Engineering Experience Accelerator (SEEA) proposes a taxonomy for experience and tools to help develop systems engineering efficiently (R. Turner).

Simulations (D. K. Pace) and their accuracy is a limiting factor in systems development, for a variety of reasons. One is that the simulation is only a virtual representation of a problem so by its very nature contains assumptions. Another reason is that the simulation scope for systems engineering is typically limited in scope and accuracy to meet the deadlines of a complex system development timeline.

As an aside, one of the biggest challenges I have faced is when an engineering team wishes to perform a simulation on a project in development or an existing system. This is a challenge because for existing systems, the simulation will never meet all the details as the real system. I have always felt better about building a system in concept development to answer some initial design questions versus building a simulation to mirror a system in operation due to all the complexities involved. As we build a large piece of machinery, or complex system, or did a multiyear development of an area we would encounter a challenge or unknown and the initial thought was simulating the challenge or unknown would allow us to address the problem, but it was not always the case a simulation could solve any problem that was encountered.

People will propose "building a simulation" with the hopes of it solving many problems that system development faces. There are a lot of challenges that simulations face when integrating into the system engineering lifecycle. Verification and validation (V&V) are the field that is dedicated to help ensure models and simulations are correct and reliable. In the last 15 years, there has been many improvements to V&V for simulation but still many challenges remain. (Kleinjnen). Also, a detailed explanation of how to efficiently integrate simulation into system engineering needs to be addressing to improve it from a one-off tool. Systems are complex and made up of interrelated components. Simulations are considered a key aspect of systems engineering in that they are a tool to aid development during preliminary system design and detailed design. Simulation is leveraged to test a system via indirect experimentation. The objective during testing is often "optimization of an effectiveness or performance measure. Rarely, if ever, can this be done by direct experimentation with a system…." (Haberfellner). In other words, simulation can test a system where it is cost prohibitive or impossible from a safety perspective to analyze the performance of a system. With systems being complex combinations of thousands of devices, it brings me to my original challenge: How to efficiently create a computer-based simulation to test to performance of a system that is operational and capture all its complexities. Why go through the effort to recreate a complex system, with historical data, in a computer-based model when it is impossible to match the intricacy of the system that is already in operation? The short answer is that any simulation will not be perfect. The longer answer is to apply the tools detailed here: Simulation Objective Analysis, Simulation Functional Specifications, and Simulation Philosophy Simulation processes to get an agreed upon scope and complexity for the simulation that will answer the pressing questions a design team has.

Part of design tools and aids, simulation methods during early stages of preliminary system design and detailed design can help confirm functional requirements, visualize the system configuration and architecture, and test performance of the system in virtual world where it may be unsafe in the physical world. (Kossiakoff and Sweet) Simulation is a way to test a system through indirect experimentation. But the challenge exists: the proper use and selection of simulation strategy and its related simulation technology is a considered more of an art than a science currently. "…there is no available theory by which the best model for a given system simulation can be selected. The choice of an appropriate model is determined as much by the experience of the systems analyst as the system itself." (Blanchard). The focus of this work is to highlight the current challenges faced by using simulations for projects and address key gaps. The results are incorporated into a method into the simulation objective analysis process.

As discussed in the systems engineering discipline, simulations offer many benefits. Most notably, simulation have ability to address questions that cannot be investigated and under controlled conditions. One can run simulations on models that do not exist to gain understanding and help evaluate alternatives. In short, simulations offer these benefits which are aligned with systems engineering field of study:

- Evaluate alternatives
- Define system and subsystem requirements
- Help define the system design
- Study performance in virtual world
- Detect design problems early
- Approximate operational effectiveness
- Determine operational and maintenance support requirements early

The disadvantages of simulations are as numerous as the benefits as well. Simulations are an approximation and never should be considered a real system. Any assumption, correct or incorrect, will cause the simulation to depart from reality. Data is difficult to obtain for simulations or may not be available. Most often, higher the accuracy of the simulation, the more complex they become to create and maintain. Similarly, simulation results often must be interpreted by specialists which can lead to biases and opinions on the accuracy of the results.

The focus of this work is the use of simulations in process industry (chemical, refining, power generation, etc.) but the tools can be applied to any simulation development. The use of simulations is a very wide topic, and simulations can be applied to many fields of study. The results of the work will focus on the process industry for oil treatment (processing, refining), power generation, and gas treatment due to the ability to leverage the dozens of highly complex simulations developed over the last decade. The focus of this work is the use of simulations to aid systems engineering development so the tools will be presented in context to the systems engineering lifecycle. The tools developed here are meant to be applied to any simulation, but the results and analysis of the work will be applied to systems engineering only. Future work may expand the use of the tools developed here to other fields and their application of simulations.

History of Simulation

The history of simulation analysis is long and is briefly reviewed here. History can be written in many different perspectives, but this will cover the rapid growth of simulations, especially the last 50 years and the current areas of research into verification and validation of simulations. Pre-computers, several methods existed to simulate systems. One of the more famous methods was the Monte Carlo method which was inspired by Buffon's needle experiment in 1777 by throwing numerous needles onto a plane with equal spaced parallel lines to estimate the value of π (Badger). This, among other experiments, lead to Monte Carlo methods based on random sampling and a broad set of random numbers to simulate results. A more modern version of the Monte-Carlo method was developed in the 1940s by Stanislaw Ulman in the Los Alamos National Laboratory. This method was programed into early computers to carry out the Monte Carlo simulations. These methods were used in the Manhattan project with limited computational tools in the 1940s to model the behaviors of particles and atoms much more efficiently in their MANIAC (mathematical and numerical and integrator and calculator) and ENIAC (electronic numerical integrator and computer) computers (Benov).

Computer modeling and simulation began to take off in the 1950s. Keith Douglas Tocher created the first simulation package: the General Simulation Program (GSP). It consisted of a set of routines that Tocher believed to be necessary for all simulation programs. These included initialization, time and state advance, and report generation (Tocher). FORTRAN computer language was used to develop SIMSCRIPT in 1963 which was intended for people who were not computer experts to run simulations (Markowitz). This led to SIMSCRIPT II which was utilized by Philip Kiviat who developed GASP (General Activity Simulation Program) which was used for manufacturing models (Nygaard). RAND, IBM, Cornell, and U.S Steel were a major activity area for the development of simulation language development during this time created SIMULA which is argued to be the most influential programming language in computing history that focused on simulation development. (Aebersold).

As computer programming techniques advanced, so did the hardware developed to support simulations. The 1970s-1980s had significant improvement to computer hardware which allowed for wider access to simulations. The decrease of cost and size for physical storage, logic, central processing, graphics devices, and overall performance led to a large increase in simulation software during the early 1980s and beyond for different industries and more accessible to different parties (Nance). This essentially leaves us at our current state: there are many options to complete simulations for our systems engineering work with no large requirements for computing power.

Developed parallel to the advances of simulations, was the development of the field of verification and validation (V&V) of simulation models. Verification and validation of simulations is a field of study that develops methods to ensure simulations present accurate data (R. G. Sargent). Model verification ensures that the computer program of the model and its implementation are correct, while validation is the substantiation that a model has a satisfactory range of accuracy and is consistent with the intended application.

Model verification was a focus of operations research in the 1940s. Verification was not an area of concern due to the early years of operations research due to model reliance not relying on digital computers. In other words, models were not computer based in early days of operations research but were based with accompanying physical models. The 1950s as computer-based simulations began to expand, so did the study of V&V. But it was not until the late 1960s where definitions were properly established (R. G. Sargent). The earliest papers focused on simulation statistics. The Fishman and Kiviat paper is considered one of the earliest papers that established the current definitions of V&V that continue up to today. The paper stated that "verification determines whether a model with a particular mathematical structure and data base actually behaves as an experimenter assumes it does" and "validation tests whether a simulation model reasonably approximates a real system". The paper did not explain how to do V&V but only that it is needed required when completing a simulation (Fishman).

With V&V being defined and an understanding of the need to use in simulations, further work was done to define the techniques used for V&V. The Naylor and Finger article was one of the first articles to propose validation methods and analysis. They based the technique of the three philosophy of science methods for validation: rationalism, empiricism, and positive economics. Rationalism required the model be logically developed from a set of assumptions.

Empiricism requires every model and assumption and outcome to be validated. Positive economics, perhaps to most unfamiliar term, is the confirmation that causal relationships and mechanisms are valid. These three requirements for what was recommend ensuring a simulation was accurate were combined into what was termed “multistate validation” (Naylor).

The 1970s-1980s continued to see accelerated development in definition of V&V for simulation models. The focus during this time was formal definitions for V&V terminology, how V&V relates to model development process, formal processes, and a recognition that a V&V process is necessary for simulation models (R. G. Sargent). The question of how accurate simulation models were was brought up by the U.S. Government as it became a concern for the use in weapons systems. This resulted in the creation of a document entitled “Guidelines for Model Evaluation” (U.S. General Accounting Office). Its goal was to inform the decision maker on how much confidence to place in a model’s results. The report gave the advice models can be helpful in giving an analytical approach but not to view then as “a magical ‘black box’ which automatically gives reliable, valid answers. It warns that mangers can use a model’s results without being informed of the underlying theories, assumptions, and judgements used to create the simulation. In reviewing results and interviewing experts, the GAO listed three areas of emphasis:

1. Model verification
2. Sensitivity testing
3. Model documentation

The report listed criteria for model evaluation and a recommended set of steps to carry out simulations. The steps are paraphrased here and the repost suggests that responsibility of carrying out these steps are for both the people creating the simulation and the simulation sponsors.:

1. Describe the problem, study objectives, and assumptions
2. Isolating the system or process to be modeled
3. Develop a supporting theory and develop a flow or logic diagram
4. Determine available data sources, formulating the mathematical model
5. Collect data
6. Describe the logic of the model with input, processing, and outputs. This leads to development of the computer model
7. Verify the simulation logic/mathematical description of the program is correct (includes debugging).
8. Develop alternative solutions and analyzing using the mode
9. Evaluating the results and output obtained from the mode.
10. Present results with a plan to carry out recommendations
11. Maintain the simulation and the data.

The GAO’s report general guideline for model verification and validation is shown in the below chart (Figure 27). The general motive for many of the governmental studies was a increasing lack of trust around simulations that were used to justify major military systems. During this time period, many military systems were given funding based on the promise of performance that was demonstrated in simulations. A series of reports and guidelines were issued to ensure that simulations followed a methodical method to ensure they were constructed well. These simulations should accurately demonstrate the performance of a system before the large amounts of funding was given to further develop the system.

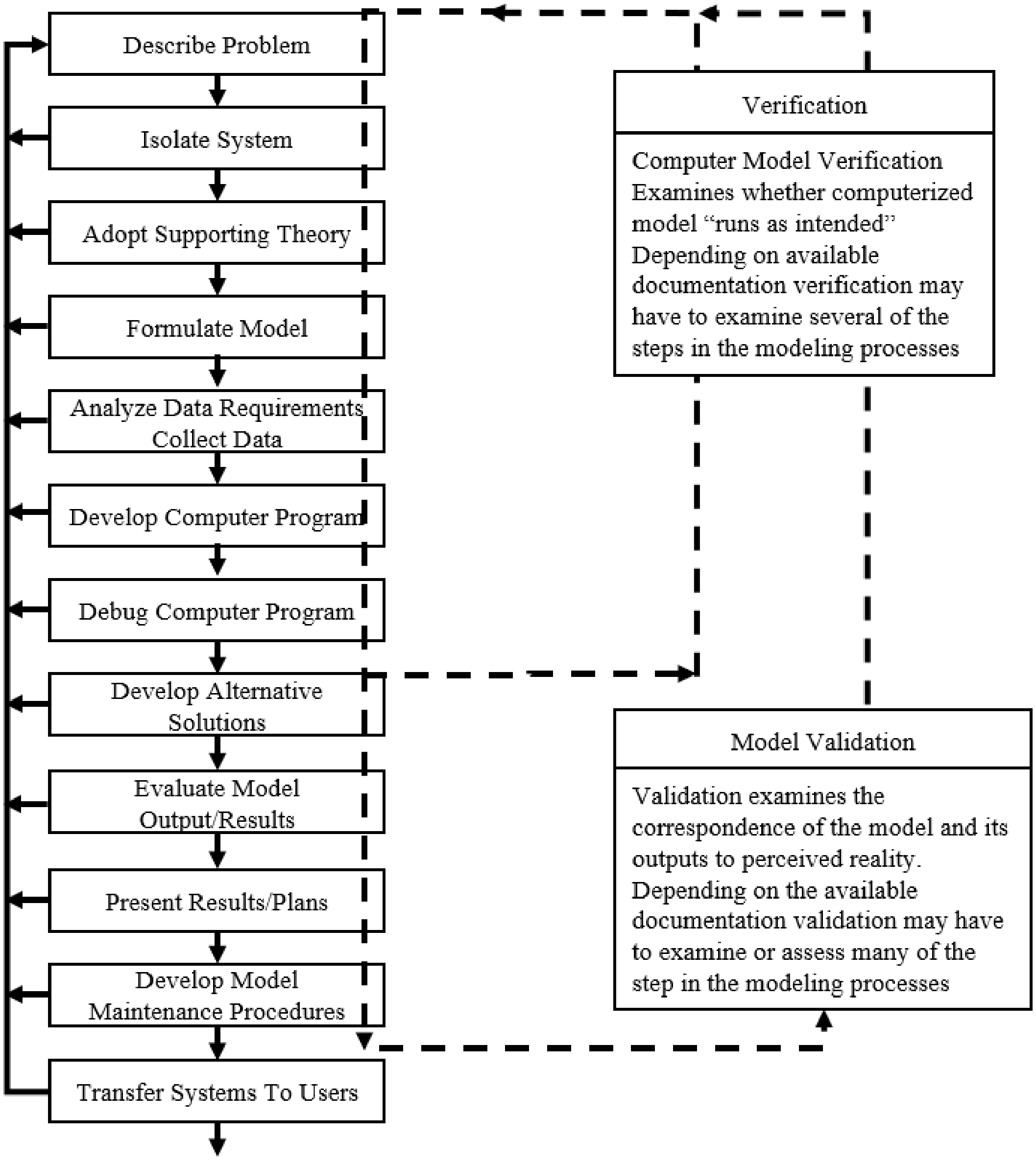

Describe Problem
Isolate System
Adopt Supporting Theory
Formulate Model
Analyze Data Requirements Collect Data
Develop Computer Program
Debug Computer Program
Develop Alternative Solutions
Evaluate Model Output/Results
Present Results/Plans
Develop Model Maintenance Procedures
Transfer Systems To Users
Verification
Computer Model Verification Examines whether computerized model "runs as intended" Depending on available documentation verification may have to examine several of the steps in the modeling processes
Model Validation
Validation examines the correspondence of the model and its outputs to perceived reality. Depending on the available documentation validation may have to examine or assess many of the step in the modeling processes

Figure 27: A graphics representation of the GAO's guidelines

The report concludes with giving criteria for model evaluation along with steps to model. The model also puts into their "Appendix A" a recommended checklist of information that is recommended for a simulation to be evaluated by other parties as well. This was method was incorporated into the process used by this research. Figure 28 shows the recommended relationship between data and the computer simulation. It shows how the computer program/simulation should be continually developed along with data and operational verification. The key message of this figure is to show how the GAO wanted there to be more emphasis on continually validating operational performance with data used in the computer simulation. For the practitioners of simulation development for military systems, the message was to improve their simulations since stakeholders were losing confidence in using simulations to fund expensive projects.

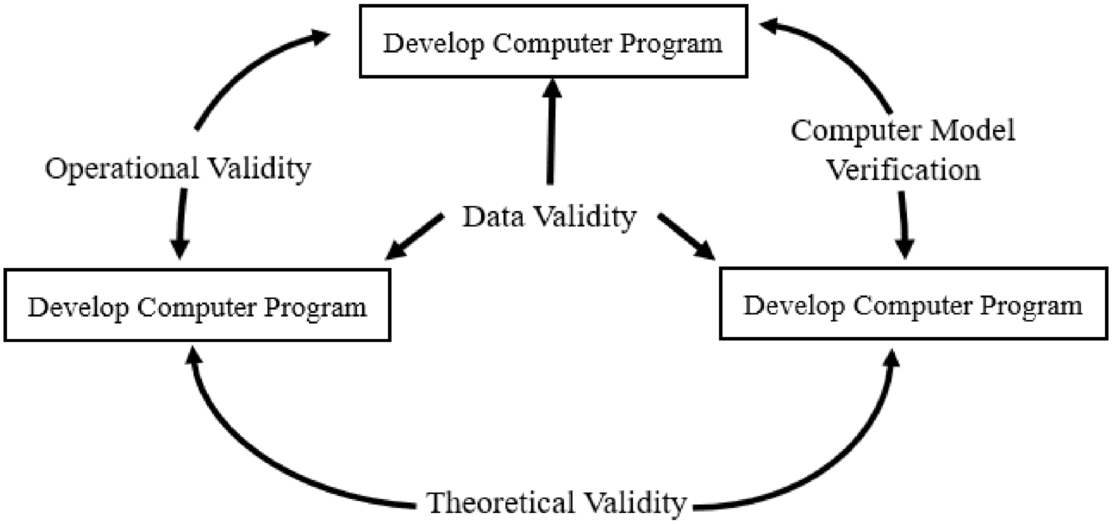


Figure 28: GAO's relationship between verification and validation

In 1987, the US Government General Accounting Office (GAO) published a report highlighting the need for an improved assessment procedure for simulations titled "Improved Assessment Procedures Would Increase the Credibility of Results". This report highlighted the weakness in simulations and how they could even be a threat to assess the capability of weapon systems. The report established the basic framework for simulations challenges that were severe enough to questions the very usefulness of the simulations (Table 10). It summarized simulation challenges that could pose a risk to accurately assess weapons systems:

Table 10: Summarization of GAO challenges for simulations

Area of Concern	Factor
Theory, Model design, and input data	1. Match between the theoretical approach and the real events being simulated 2. Choice of measures of effectiveness 3. Representation of the operational performance 4. Portrayal of weapon's immediate combat environment 5. Depiction of the critical aspects of broad scale battle environment 6. Appropriateness of mathematical and logical representation 7. Selection of input data
The correspondence between the model and the real world	8. Verification effort 9. Attention to statistical quality of results 10. Sensitivity testing effort 11. Validation effort
Management Issues	12. Organizational support 13. Documentation 14. Full disclosure of results

In addition to the above area of concerns for simulations, they also did little effort to validate simulation results by comparing them with operational risks, historical data, or other models. It was recommended by the GAO to develop a guidance on “producing, validating, documenting, managing, maintaining, using, and reporting simulations”. While this work was aimed at simulations based on weapons systems, similar weaknesses have been seen in other simulations for the process industry. (Fossett, Harrison and Weintrob)

The increase of process simulators, starting in the early 1980s, had a large impact in improving process simulations. It has fostered in cross discipline teams that have collaborated for process modeling, optimization, economic analysis, and operator training under a variety of system projects. Some estimate that the increased use of simulation has help lead to energy consumption per unit of production to be decreased by 20% and process capital costs are down 10%. (McMahon)

The use of simulations continued to expand both with practitioners of systems engineering and outside the discipline. Simulation use is a continuous evolution, so the tools presented here are as flexible as possible. Vikas Dhole, Vice President of Engineering Product Management at process-simulation software manufacturer AspenTech, identified six key trends (Dhole):

- Process modeling is morphing into process optimization
- Sequential problem-solving is evolving into simultaneous solutions
- Simulation is edging continually closer to process engineering
- Wireless tables are replacing the PC environment
- Usability is soaring as simulation workflows become more streamlined
- A marketplace for independent software suppliers is developing within the simulation environment

But as simulations continue to be a key point in systems engineering development, their use is still plagued by many problems that have existed when they first started being utilized.

The development of a defined set of challenges for verification and validation has developed alongside the development of simulation technology. While important improvements have been made, especially within the last 20 years, significant challenges remain that slow the full potential of simulation. That was motivation for the work to establish an integrated work process for simulations in systems engineering and establish metrics. In 2002, roughly 200 people, representing government, academia, and industry met to establish the current condition of simulation and verification & validation (D. Pace):

- The primary motivation for modeling and simulation V&V is risk reduction
- Effective communication is a problem because of continuing differences in the details about terminology, concepts, and V&V problems
- Advances in modeling and simulation framework/theory can enhance V&V capabilities and is essential for increasing automated V&V techniques
- Limitations in items required for effective V&V (such as required data and detailed characterization of associated uncertainties and errors, simulations/software, etc.) must be addressed, with many of the management processes for coping them being coming in many areas of simulation application
- Cost and resource requirements for modeling and simulation V&V are not well understood as they need to be because meaningful information about such is not widely shared within modeling and simulation communities, and much more information about cost and resource requirements needs to be collected and made available to facilitate development of more reliable estimates processes.
- Areas of modeling and simulation V&V need to employ more formal (repeatable and rigorous) methods to facilitate better judgements about appropriateness of simulation capabilities for indexed uses

So these problems highlight the current challenges seen by practitioners in the use of simulation.   Furthermore, Foundations '02 (D. Pace) identified three management challenges:

- Qualitative assessment: This involves human judgement in assessment such as peer reviews, subject matter experts, face validations. When these assessments are performed, they are often completed by people with not the proper credentials or a formal process.
- Formal assessment: there is little mathematical methods (statistical in nature or following some rigorous approach) to apply to simulations. The management challenge is to develop easy to use, "lightweight" variants of the processes that can be more easily used in modeling and simulation V&V to enhance the quality of formal assessments. (Eldabi)
- Cost/Resources: Correct estimation of resources needed is a primary challenge in any modeling and simulation project. The challenge is to collect and organize appropriate cost and resource information.

Solutions to each of these challenges will be addressed via the tools developed with simulation objective analysis, simulation functional specification, and simulation philosophy. The simulation objective analysis (the term used for the development of a simulation scope by ongoing interviews and analysis of the system) is outlined here. It would be used to document the simulation functional specification (like functional requirements of a system under development) and would be incorporated into the simulation philosophy. Simulation philosophy is based on two structural principles: stakeholders hold the risk for the project and with integrated project teams on system development. Initially the document is drafted as a simple set of rules and processes to promote desired practices and behaviors, later as the simulation matures it can be a location of fixed procedures and detailed instructions. It serves to document decisions, provide safeguards, and provide ways to document innovations found during the simulation work. Reading all the government reports and recent publications regarding simulation verification and validation all shared specific recommendations in common to developing simulations for complex systems:

- Have an agreement made prior to developing the model between the simulation team, stakeholders and (if possible) the users, specifying the basic requirements, scope, and design of the simulation, taking into account the system being designed, cost & schedule limits, and the fidelity needed for the simulation
- Specify the amount of accuracy required of the model's output variables prior to starting the development of the model or very early in the model development process.
- Document and test, wherever possible, the underlying assumptions and theories for the model.
- Seek expert advice on the validity of the model. I.e., continually check if the simulation is close to reality based on expert experience
- In each simulation update, at least test a range of values to see if simulation accuracy still holds.
- If simulation is being made alongside a prototype or actual system, test the simulation alongside physical system to test accuracy.
- Track changes to simulation and any validation results in documentation for simulation or system development.

These common recommendations discovered when reviewing the literature from government studies, V&V publications, and other simulation case studies became the starting point for developing a methodical process to integrate into systems engineering how to manage simulations for a complex system underdevelopment. Modeling and simulation are core research areas for systems engineering. It works as a method for testing and validation of design requirements. It can be utilized at all phases of development from preliminary system design, to detailed design, and eventually to production. It can be used to test alternative designs or to run tests on a system that are infeasible to run in reality (Blanchard). It was for those reasons that simulation was emphasized in a multi-year program to upgrade power plants, processing sites, and other manufacturing sites in Southern California. This chapter discusses the incorporation of this testing into the systems engineering framework, lessons learned, and innovations used to ensure that when transferring from control system a newer model there is no operational upset.

As previously discussed, a team was tasked with upgrading legacy programmable logic controllers (PLCs) across several industrial sites with the stated requirement that long or unplanned shutdowns could not occur during the upgrade. It was decided to incorporate dynamic simulation into the project to verify programming code integrity and test functionality during Factory Acceptance Testing (FAT) before deployment of the new control systems to ensure a seamless switchover. This use of dynamic simulation was particularly important in addressing the established practice of using checkout code during PLC commissioning in the field. The traditional checkout practice, however, provided no opportunity to test the response of critical safety systems to the new PLCs. Software-based dynamic process simulator for code checkout addressed this problem by providing a controlled test environment. It could also be used to train operators on the new control system prior to operation. With the main goal of the simulation identified to QA/QC the new programming and ensure little or no downtime during the upgrade, the underlying problem became how to efficiently use the simulation to ensure its design meet the requirements and not excessive costs or time were dedicated to it that could jeopardize the project. To address this, a method was developed to do a group analysis of the system and document results. The responses were analyzed, and a proposal was made for a cost-effective simulation that meet all the project requirements (including training of operators, testing of safety systems, and checking programming integrity). A metric was developed to measure the cost improvements made for simulations during the span of the upgrade program. It showed that simulations costs improved 40% after using this method and all simulations meet the design requirements.

This chapter is organized into several sections which serve to give a background into the problem addressed, results, and innovations developed. First, it will go over the basic operation of programmable logic controls and their effect on industrial site operations. PLCs are discussed to highlight how they can easily be interfaced with simulations. Similar current research in systems engineering regarding simulations and credibility assessments will also be discussed. The development of objective analysis portion goes over the details of the systematic method to optimize the use of simulation for each unique project. Finally, the results go over the cost improvements that occurred by utilizing this method.

Logic controllers and their relation to simulations

Programmable logic controls (PLCs) are the heart of many modern manufacturing processes. Its predecessor was hard wired relay panels that would control a facility based on signals from instruments and unique relays. PLCs use programming languages as the basis of their logic and control. Input signals from different instruments, pumps, or other parts of a facility are read from the PLC, logic is solved, and outputs from PLC are sent to devices which control the plants operation (Sages). For example, if a pump is supposed to run if the temperature is above 100 degrees, a temperature transmitter will send a signal for temperature to the PLC, the PLC will read if temperature goes above that setpoint, and then send a signal to start the pump.   PLCs are designed to keep the facility running continuously and minimize operator intervention. Often PLCs are installed with an expectation that they will run continuously for greater than 10 years. Their continuous operation ensures that the facility stays running. The typical architecture of a PLC is the processor, power supply, and input/output (I/O) modules. The I/O modules are then connected to the rest of instrumentation, rotating equipment, and other devices that are controlled by the PLC. The number of I/O is generally proportional to the size of the program and therefore, the size of the automated portion of the facility. It was because of this that I/O was used to normalize the results of the simulation across multiple projects to determine the cost efficiency of each simulation developed.

There are several standards used for programming language such as ladder logic, sequential function charts, and text language with ladder logic being the most common (Erickson). While there are standards written for ladder logic and PLC programming in general such as IEC 1131, there is still the challenge of upgrading from different older PLC platforms to newer.   Different models and the potential human programming error combined with criteria to minimize lost revenue during a shutdown make the instantaneous change of an old control system to a new one challenging. It was determined the best way to error check program, train operators to new graphics associated with it, and test functionality was to develop simulations that would test the accuracy of programming in a virtual world before being deployed to the real world.

Delphi method used for simulation scope definition

A variation to the Delphi method was utilized for this simulation development. The Delphi method is the use of iterative surveys to get expert feedback and predictions. Typically, it involves a questionnaire and two or more rounds and at the end the facilitator provides a summary of the results. The Delphi method is common in systems engineering due to its ability to quickly weigh different expert opinions. Related to this work, a Delphi method was utilized to assess the credibility of modeling and simulation models for NASA. It resulted in efficiently evaluating the credibility of many models and simulation and determined that the tools where at a point between development and production level in their complexity (Ahn). The utilization of a risk management plan also was beneficial during determining the design of the simulation to highlight the key risks to the project and incorporate that into the testing plan. This slight modification of the Delphi method that recycled simulation scope if not approved by stakeholders is shown below in Figure 29.

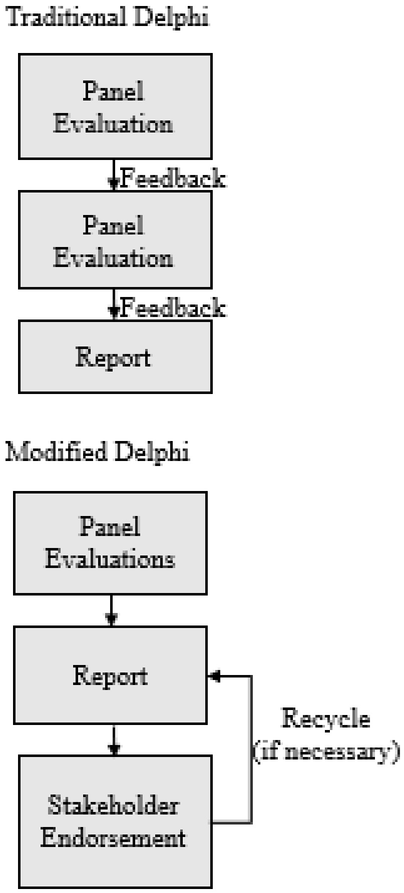


Figure 29: Modified Delphi method utilized

Development of Simulation Objective Analysis

The meeting was structured to survey the panel of local experts on the details of the facility. After introductions and quick review of the facility's design, the questions asked during the meeting were broken up into several groups based on theme. The themes were intellectual property, process design, controls design, training simulation, and project close out. Examples of each theme's questions are given in Table 11. There were roughly fifty questions developed for each team analysis done. After completion the answers were discussed and a detailed simulation scope for the project would be developed. This would be communicated out later to the project team, attendees during analysis, and stakeholders. Major design decisions for the simulation were explained based on information of the facility and answers from the objective analysis. This method differs from the Delphi method discussed previously in several ways. Firstly, team members were experts but in different backgrounds. Other differences to a traditional Delphi included the consensus was reached on the answers as a group and the final recommendation was reported out later for iteration. Very rarely was there a major iteration to the results after the first proposal was made.

Each site was different in complexity, size, and design so a method had to be developed to ensure that the simulation for controls check-out added value to the project and ensured that unique high-risk scenarios were addressed. The team developed a series of questions to ask each project team to document the concerns for each facility where the upgrade occurred. This was done based on industry best practices and stakeholder feedback. For the objective analysis the meeting included the instrument and controls engineer, process engineer, operations, expert consultant on simulations, and moderator. Other personnel could attend as needed such as subject matter experts which could aid a specific area, but this was the core team.

Table 11: Typical screening questions utilized

Question Themes	Examples
Intellectual Property	• Who is the process licensor of process design? • What are the intellectual property issues involved with process designs?? • What software restrictions are there in area and for company?
Process Design	• Is the asset an existing site? Or new construction? • Is there a need to analyze transient behavior to establish safety or design parameters? • Is there a need to assess environmental impact of startup, operation, or shutdown of design alternatives? • Is there a need to test the overall control design for the process operability and integrated control system?
Controls Design	• How complex is the safety system interlocks? • Is it necessary to test the configuration of control loops and automated processes of facility? • Is there a need to test third-party controls such as burners, instrument air systems, or asset management software?
Training Simulation	• Are operators already familiar with the process? • Will operators learn new procedures with this project? • Could low fidelity, less expensive tools be adequate for the training on this project? • What is the overall complexity of the process compared with current operations? • Is it necessary to simulate emergency procedures?
Project close-out	• How often do permanent changes occur at the site with controls systems, unit operations, and instrumentation? • Does the facility have in-place an existing system to maintain simulation documents (i.e., engineering drawings, wiring schematics, operator training)? • Who would be long-term owners of the simulation after completion? • Are there any synergies with other groups maintaining a simulation long-term? I.e., production or safety groups?

Other team members were included at times but those five composed the key team to be in place for each session assessing the simulation design. It was agreed for schedule constraints to limit the meeting to one session only for expert feedback. After this meeting, the incorporation of feedback would go into a detailed report for stakeholder endorsement. If the detailed design of the simulation could not be endorsed by management and panel experts, then it would be recycled back to the report stage to modify. The traditional Delphi method to incorporate expert feedback is based on continual feedback from panel. Due to schedules and time constraints, it was agreed to for objective analysis that the report of the preferred design of the simulation would occur after only one meeting. After this the report would be prepared and submitted for approval. If rejected, the feedback rejected would be incorporated into another report, but the expert panel would not meet again.

Fidelity of a simulation is term used to describe its ability to recreate reality (Hays). High fidelity models refer to 3-D emersion and complex scenario testing, while low fidelity models in this case refer to simulations done with PLC only and only simple individual function testing. Many different fidelity models were considered for each simulation. The term fidelity, regarding simulation, is a measure of the ability of a simulation to exactly match the system. The higher required fidelity of a simulation results in higher costs due to the added detail you must incorporate into the model. Often it is requested to develop simulations for systems already in operation, but it remains cost prohibitive to do this.

That is my motivation for this work based on years of working with simulations and the challenges faced in late-stage systems development. Fidelity (accuracy of the simulation) is limited by effort, but inaccurate results are often not the result of poor effort. In other words, countless hours can be put into a simulation but a single incorrect assumption, missing process detail, or lack of knowledge of human behavior can render a simulation worthless. Simulations can be a powerful tool for late-stage systems development (commissioning, production, and operations) but a care must be given to ensure simulations that are created directly address the problem. This motivated me to develop the simulation objective analysis process. The process's goal is to have a methodical, direct way to develop simulations that directly address the concern of the engineering team and stakeholders without unnecessary scope.

Apart from two simulations developed, the simulation fidelity level selected for each upgrade was to connect the PLC to an isolated network with operations terminal combined with a thermodynamics package. These were termed "human-machine interface tools", the others were simple simulations. Cost estimates were prepared several times for higher fidelity models though at stakeholder request to weigh trade-offs. In addition to often being cost prohibitive, the complexity of the sites did not justify the use of more advanced models. Table 12 below shows several of the technologies evaluated, their description, a high-level impact to organization, and relative costs. The simulation type used for many of the PLC upgrades is highlighted as "selected technology".

This was the simulation design selected for these upgrades. It should be noted that it was considered to utilize this design for operator training simulators after the upgrades were completed. Ultimately this was not done. This tool and method were utilized as a way to test the programming prior to field commissioning, orient the console operators to new graphics and functionality, test complicated control schemes (such as feed forward, cascade) prior to deployment to a new system. It was considered to utilize "immersive 3D simulation" and a formal operator training simulator (OTS) but these were not selected due to be cost prohibitive. The selection of an HMI interface with logic control and dynamic simulation was the lowest cost option that was able to mean all the requirements needed to test the programming. Simple simulation, such as manually testing the programming, was not rigorous enough to ensure all the functionality worked prior to commissioning.

Table 12: Different levels of fidelity for simulation

Example Fidelity Levels	Description	Impact	Relative Costs
Immersive 3D Simulation	Real-life look and feel avatars and total operator interactivity for an immersive learning experience.	Requires significant organizational capability to support ✓ Similar to OTS	40-80
Operator Training Simulators (OTS)	In addition to using HMI coupled with simulation software, OTS contain more functionality. Contain complete interactivity, scenario analysis and scoring capabilities. Provides comprehensive/advanced training of plant operators using actual plant equipment and operating conditions	Requires significant organizational capability to support ✓ Dedicated Plant Trainers ✓ Facility engineering ✓ Information Technology ✓ Operations Personnel ✓ Simulation Engineers ✓ Automation, Programmers ✓ Long term maintenance	20-50
Human-Machine Interface tools (HMI) for Simulation Testing Selected Technology	Use current HMI to build interactive displays of equipment. Integrate programming with thermodynamic simulation to re-create controls of plant and typical responses. Able to give operations a view of graphics and control responses prior to commissioning. Simple tests only.	Requires: ✓ Automation, Programmers ✓ Operations Personnel ✓ Simulation Engineers	10
Simple Simulation	Testing of programming code in system hardware only. Adjust inputs to programming and observe outputs. Little operations interaction with new programming.	Requires time for QA/QC. Typically involves a rigorous checklist to ensure all design requirements are met. ✓ Automation, Programmers	1

Figure 30 shows the developed architecture of the simulation used during the multi-year upgrade program. The simulation software composed of a thermodynamics package to simulate the process conditions and unit operations occurring at the facility. Graphic software was the software used to re-create the operations terminal graphics. The Human Matching Interface (HMI) would be where the person operating the simulation would interact. The PLC hardware would be the controller that would eventually be used in the physical upgrade. A network switch was used to connect the network to each component. OLE (Object Linking and Embedding) for Process Control, or simply OPC is the standard for data transmission and converts data sent from the PLC. With all these components setup the simulation can run with operator manipulating the HMI, PLC solving the logic, and the simulation model responding.

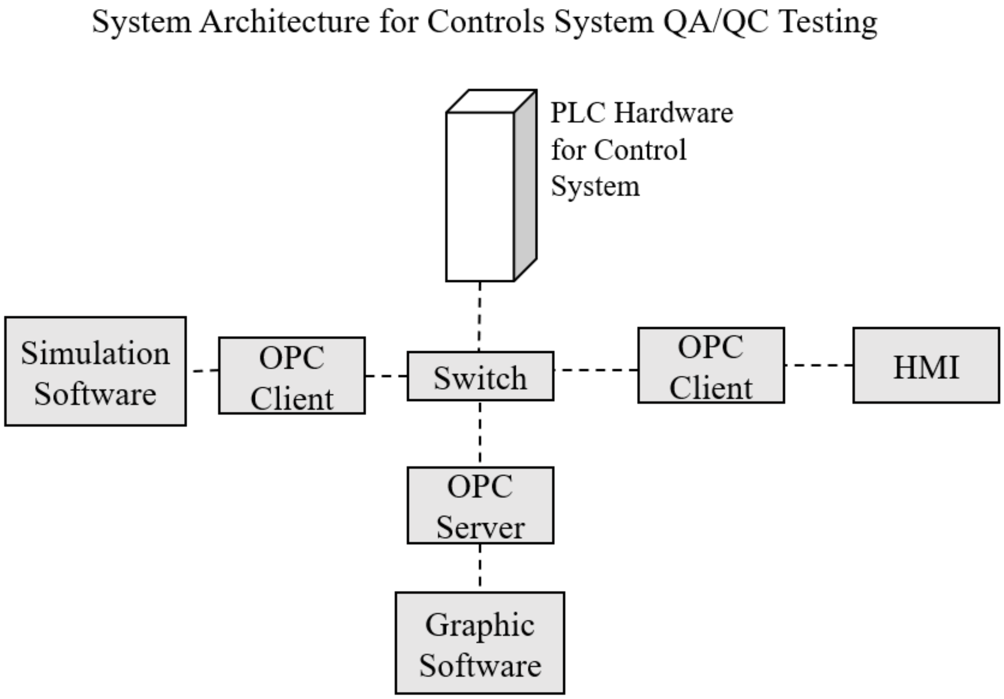

System Architecture for Controls System QA/QC Testing
PLC Hardware for Control System
Simulation Software
OPC Client
Switch
OPC Client
HMI
OPC Server
Graphic Software

Figure 30: Simulation testing network architecture

Experimental Results

The developed check-out and testing procedure are outlined below. After the scope of the simulation was determined, these steps were followed to test the programming against the simulation to ensure no errors were in the programming:

- Formulate Problem:
    - Define scope and complexity of PCS and SIS systems
    - Identify system boundaries
    - Define the goals of the dynamic simulation (PLC cutover, RCA investigation, etc.)
    - Determine what dynamic simulation models are to be developed during detailed engineering to checkout equipment design for transient operations, mode changes, process upsets, startup, or shutdown. \
    - Define other aspects of problem as needed
    - Optional but recommended: Hold kick-off meeting to review objectives of study, specific questions of study to be answered, scope, performance measures, system architecture, and required resources and time frame of study
- Collect Data & Construct Dynamic Simulation
    - Collect Process Safety Information (PSI) such as P&IDs, control philosophy, existing programming, PHA etc.
    - Collect data to specify simulation parameters (e.g., time to failure, response times, high consequence events, conditions of systems)
    - Define assumptions to be used

    - Collect all data in project folder
    - Construct Simulation
        - Detail of simulation should depend on objectives, data availability, SME feedback, time & money constraints
- Validate the dynamic simulation
    - Walkthrough of model should be performed by someone experienced with simulation models
    - Walkthrough of model with project team, operations, and other SMEs
    - If errors or omissions are discovered (which is almost certain), model should be updated and/or more data collected to ensure accuracy. Again, a review should occur to make sure model is accurately updated.
- Program model/interface with logic
    - Verify program (ex. ControlLogix) is accurate and up to date for system being analyzed
    - Interface program with dynamic simulation
        - Ensure tags and other identifiers match up
- Is programming with model accurate?
    - Similar to a QA/QC procedure for programming, ensure the dynamic simulation works with programming by testing responses to model
    - This should be done with a predeveloped checklist to see if specific scenarios are accurate
- Conduct Simulation Tests/Experiments
    - Results should be analyzed to determine if additional testing is required

- Document Results
    - Documentation of simulation should include a description of model, a detailed description of the scenarios tested, and the results
    - The results should include the validation process to give the simulation credibility

The metric used to measure the improvement of the simulation's cost efficiency during project upgrades was the simulation's costs divided by the number of I/O at the site. Figure 31 shows the improvements made after the process was established. There was one notable increase in costs which was attributed to a project where scope increased after the simulation scope was agreed to and other complexities of the project. Costs lowered for almost all simulations from the original one developed. The improvements increased for the most part as we progressed the projects as well. This was attributed to the members of the team learning from past projects and encouraging fewer complex simulations where possible and modifying the objective analysis questions. Figure 32 shows the average costs for each scope of the projects. The average costs to include simulation for controls checkout was roughly 2%. Other results indicated that error checking the programming in lab prior to deployment in field resulted in a decreased amount needed for commissioning. The major requirement, of no unplanned shutdowns or stoppage of production during upgrades, was also achieved. Less tangible results included positive reception by operations about the ability to test the new control system prior to deployment. As mentioned previously, operations were able to test all the graphics prior to start-up of the new system and full operation. Graphic software often was replaced along with these upgrades so it was an opportunity to show the new graphics which were often dramatically different form the legacy systems they were replacing.

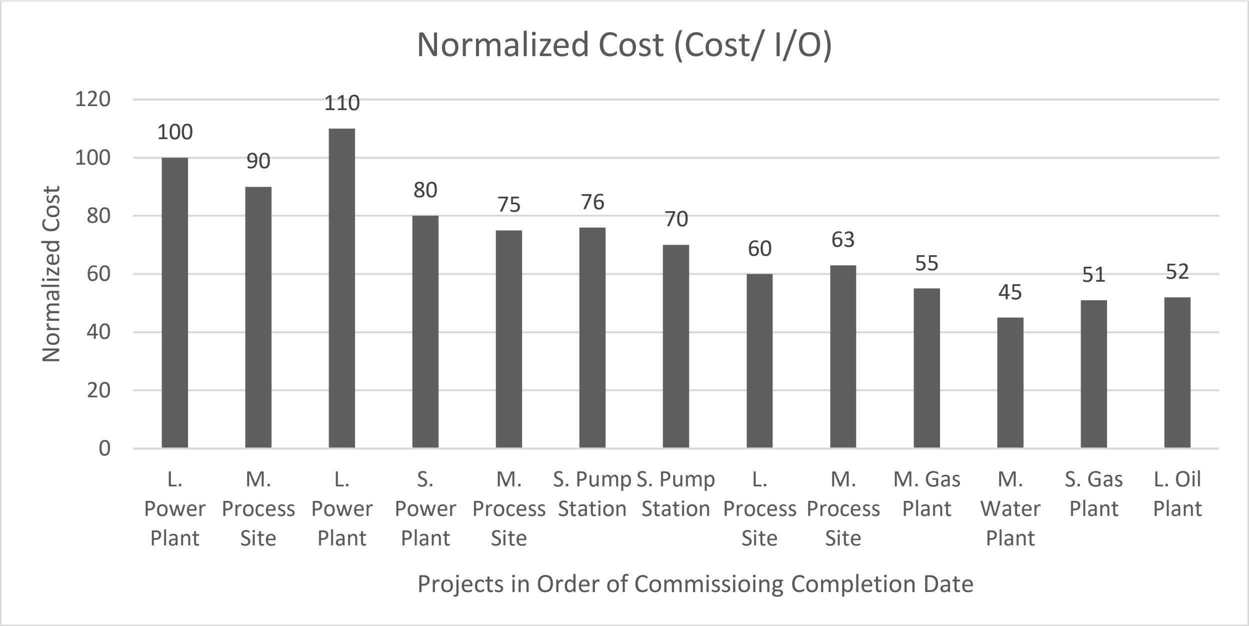


Figure 31: Chart of improved simulation efficiency over time

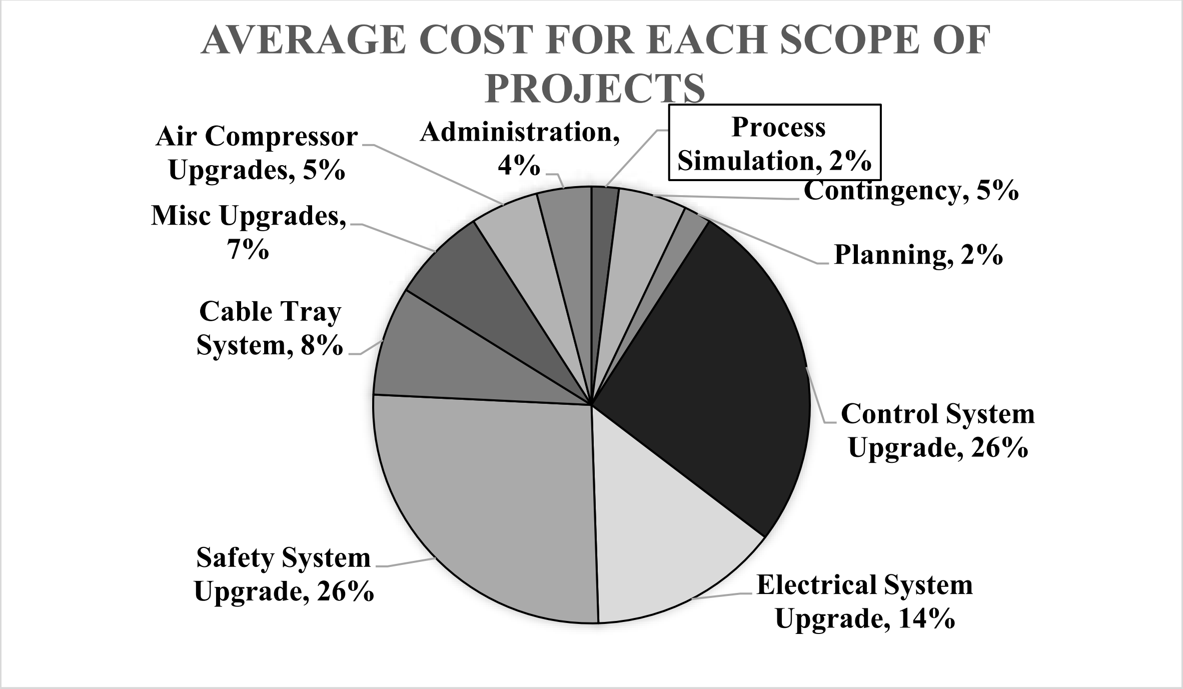


Figure 32: Average relative costs for upgrade scopes

Limitations

One major criticism of the results that utilized benchmarking is that it does not capture the complexity of each site in the efficiency of the simulation work completed. In other words, if a site that was incredibly complex but had a normal amount of I/O and a 3-D model simulation was determined necessary it would not be seen in the results. A counter to that is that the sites where these upgrades took place and simulations created were similar in nature. Power generation, oil processing, and pumping stations, while have their differences, are in a related industry. These sites could not be directly compared to rocket launch pads, space stations, or aircraft using these metrics. The metrics used are only applicable to sites in a similar industry or level of complexity. The process of objective analysis continually led to improvements of the simulation's cost efficiency. Some of the feedback received was that there was little use of the simulation after the project startup completed. This was expected since the agreement typically reached was to use the simulation as quality control for programming and to orient operators to new design. It was discussed with stakeholders about the long-term maintenance of the simulations but the cost and constant design changes at the facility (which would need to be incorporated into simulation) resulted in the simulations not being used after startup. Future work is looking into similar methods shown here to have cost effective simulations in the production phase of a system which will let operations quickly modify their site in a virtual environment that maximizes the value and minimizes the cost. This will allow a system in the production phase to be tested in a simulation quickly prior to making the changes at the facility.

With testing it was discovered that certain process variables had larger impact on simulation results than others. Simulations for process systems have the biggest impact in measuring performance by making assumptions of variables which are fast transients (flow and pressure) as opposed to slow transients (level and temperature) based on calibrated range. In the case of process engineering, the closer and faster the error to key performance indicators had the biggest degradation of simulation accuracy. It is important in designing simulations to assess which error degrades the simulation the most and how to improve its accuracy (in situ tests, prototyping, etc.). Beyond process systems, simulations must assess which data possess the biggest risk and design safeguards to limit the damage to the validity of the simulation. When it was determined that error in certain variables caused large errors to propagate through the simulation, special attention was given to ensure the process variable error was controlled.

Discussion

As mentioned in previous sections the formal development of a simulation objective analysis led to a way to get documented feedback from operations personnel for a site where a large control system upgrade was planned. It led to improvements in cost efficiency by verifying the characteristics of the site, the high-risks aspects of the control system, and goals of the stakeholders to develop a list of design requirements needed for the simulations. This led to a defined scope for the simulation that led to cost improvements and minimized unnecessary work. There are still limitations in using dynamic simulation technology that was encountered during the project and increased the costs of using simulation. One example was the reliance of PLCs on central clocks (often referred to as coordinated system time) to coordinate operations. This led to no ability to speed up simulations to test results for process results or to train operators. Similarly, inefficiencies of the simulation in our case also included inability to load specific states for operator training and forced us to manually setup systems prior to training. Different platforms have varying abilities so care must be taken to assess the limitations of combining simulation software with hardware to improve programming QA/QC and aid training.

Conclusions

The results of doing a defined simulation objective analysis to solidify the specifications of the simulation design for a series of control system upgrade projects led to a significant cost efficiency improvement. Simulations are often viewed as a tool of last resort or an easy budget item to cut. Having a clear way to limit the scope and formal method to document goals led to dynamic simulation being a beneficial tool for PLC cutovers at industrial sites. Further work is needed to improve the performance and flexibility of the simulation to improve its efficiency and be able to imbed the use of simulations for control check-out into long term operation of site after the initial upgrade is completed. Future work is tailored to leveraging dynamic simulations, real-time data, and machine learning to embed simulation long term into a facility long after commissioning. Examples of the benefits of combining these three aspects include rapidly testing variations in production, automatically generating programming and graphics (such as interlock documentation) and predicting complex events such as fuzzy logic or multiple failures over long periods of time. This research addresses the second research question: What techniques can be used to simulate prior to cutover to improve safety and commissioning times? The research contributions of this study in particular are:

- Defined systemic way to test control systems, developed unique architecture for simulation, and test results for subsystem upgrades for constraints in systems engineering development.
- Established and used new metric for simulation modeling for industrial systems which baselines efficiency (Cost per IO)

- Developed a new method which utilized simulation with defined interfaces to the new logic controllers to aid in the quality checking of programming and startup of new systems.
- Outlines a systemic way to test control systems for subsystem upgrades for constraints in systems engineering development

With the risk of programming errors eliminated via dynamic simulation testing, the upgrades proceeded. The final step was how to design the automation subsystem to maintain their integrity over longer periods of time and to identify obsolescence quicker. Chapter 6 discusses a developed method to audit the integrity of the automation system. This is done with creating an audit of the alarm configuration ton ensure unsafe changes are not made to the control system after the upgrades.

Chapter 6 – Automating the audit of alarm configurations for manufacturing sites

Introduction

This chapter details the development of a continuous alarm monitoring system to track the compliance of alarms in a manufacturing environment. In industrial settings, management of alarm systems is often complicated by conditions where simultaneous capital projects and continual process optimization can cause the integrity of the alarm system to lapse. This work details the development of an automated alarm audit system, its application, and the measured results. A risk-based objective analysis method was utilized to establish safety thresholds and prioritization for each alarm. An automatic audit tool was then developed to continually cross-reference an alarm safety server with the process control network to notify management and engineering of any adjustments that violated alarm and safety rules for the site. When deployed to seven facilities, this method resulted in large improvements over prior manual methods. The implementation of the audit tool led both to improved site safety, and to an improvement of alarm metrics due to greater confidence in adjusting alarms. One site saw a monthly reduction in the quantity of alarms by 55% and the elimination of alarm floods. Alarm auditing is a requirement for many regulated industrial sites and this work demonstrates the development of an effective, risk-realized, automated system.

Alarm management fundamentals

An alarm is an audible and/or visible means of indicating to the operator an equipment malfunction, process deviation, or abnormal condition requiring a response (ANSI). Alarms allow for human interaction to avert consequences (personnel safety, environmental, asset loss) during both normal operations and periods of high alarm traffic, such as power failures or process upset (ISA). A well-managed alarm system promotes injury-free and incident-free operation by assisting the operator in determining the causes and proper actions to effectively manage the abnormal situation. The alarm system must be designed and maintained for effective mediation of a single alarm during normal operation, or of many alarms during a major event.

Alarms are employed in many diverse fields of engineering and industry. Each industry which utilizes alarms though has its own unique characteristics for how often they manipulate alarms after the system is built. For example, the setpoints of alarms change very rarely for aviation after the airplane model is built without considerable oversight and controls (Stanton). In medical systems, setpoints can be determined by the individual patient characteristics, and adjustments are made to prevent nuisance (alarms giving false positive notification too often) or chattering alarms (where alarms continually reset due to being set too conservatively) (Horkan). Spacecraft also deal with alarm management such as detecting anomalies for their particular mission, so adjustments are continually made and there is a challenge of minimizing false alarms and ensuring operators of these systems are notified when actual action is needed (Xu).

In industrial process systems (the focus of this study), the design and management of the alarm system is complicated by the fact that large systems are undergoing continuous optimization, process refinement, upgrading, modification, and maintenance. These changes can lead to alarm system entropy, a condition where the alarm system becomes unsynchronized from the engineering rationales and requirements that drive its design (Izadi). If unaddressed, alarm system entropy causes alarm systems to degrade over the operational lifetime of these systems. (Hollifield and Habibi) The general established method for alarm system auditing with the purpose of re-rationalizing the alarm system, is based on American National Standards Institute (ANSI) and International Society of Automation (ISA) standards. ISA 18.2 states: “To maintain the designed alarm attribute settings (e.g., alarm setpoints, and alarm priorities) at authorized values, there should be a regular comparison of the rationalized values with the settings in effect in the control system" (Bransby).

Alarm rationalization is the process to define the attributes and configuration details of an alarm, and their engineering/technical basis. Many alarms are only defined by their setpoint, but the formal rationalization of alarms include definitions of setpoint, criticality, and characteristics of intended operator response. Alarm rationalization is a systematic approach to evaluate every alarm in a facility to document consequences, impact, corrective actions, response time, and priority (Beebe). Inconsistent and undisciplined use of alarms can create a frustrating situation for operations and lead to ignored and ineffective alarms. The goal of alarm rationalization is to review, validate, and justify alarms that meet the criteria of an alarm. There are several means of implementing these standard techniques for alarm rationalization ranging from the very simple to the complex (Noda), but the broader purpose is to design the alarm system with sufficient number of alarms to ensure the process system is safe, that operators have sufficient time to respond, and the system operates in a safe process range (Hollifield and Habibi). Alarm rationalization is a major determinant of whether the effort is successful or fails to manage alarms at a large industrial site (Beebe).

In practice, the alarm management and rationalization techniques that are documented in the literature are challenged to manage alarm systems at the scale of industrial process sites such as oil and gas operations which can contain thousands of alarms for medium-sized sites. With the widespread use of distributed controls systems (DCS), the number of alarms has increased at sites along with the ease to alter them (Koene and Vedam). These conditions often lead to alarm systems that are operating under conditions of continuous change due to simultaneous operations, construction, and maintenance. Managing the alarms within these types of industrial systems has been performed by manually alarm auditing the process control network (PCN) against an alarm master database, through quarterly, manual audits and spot-checks of only a handful of alarms. (Safer and Laplante), (Vasarhelyi and al.). These manual audits become infeasible with reliance on legacy systems, with increasing scale, activity, and number of alarms.

This understanding of the state of the field motivated the development of a continuous and automated framework for alarm rationalization and auditing. A risk-based approach was utilized to evaluate each proposed alarm and risks associated with it to determine key parameters of each alarm. The parameters to determine during the analysis included maximum time to respond (MTR), alarm priority, alarm type, cause, operator actions, and setpoint. After the alarm rationalization was completed, a master alarm database was created with the results. An automated alarm audit and enforcement tool was created to cross reference the actual process control network alarm settings with the alarm rationalization in the master alarm database. This chapter describes the characteristics of this proposed means to perform alarm system rationalization and auditing. It also describes the costs and benefits of such a system as implemented on an industrial scale.

The methods proposed here for this investigation of continuous, automated alarm rationalization and audit are composed of the steps of 1) alarm rationalization and creation of alarm database, 2) Building the alarm audit IT system, 3) Implementation at industrial process sites and experimental procedures. These methods are described in the following sections.

Alarm Rationalization and Creation of Master Alarm Database

Upon receiving management approval for alarm system audit, the first step was to rationalize the alarms to determine the correct parameters. While data existed for some sites for the alarm tag, setting, priority, and other important data, it was requested to refresh the data at each site. It was agreed that for all new systems, modified systems, and systems with known problems, a formal process would be undertaken to document and correct alarm rationalization issues. A systemic approach was developed based on common industry practices to rationalize the alarms based on an objective analysis carried out by facilitator and key personnel. This rationalization method is common in industry with several variations, but the outline is described here. The following participants were considered key participants for the objective analysis: Facilitator, head operator, shift operator, process engineer, programmer, automation engineer. Engineering documentation is also required for the meeting which included (but not limited to) piping & instrumentation diagrams (P&IDs), process hazard analysis studies (PHA), cause and effect matrix, and list of alarm tags. With the key personnel and engineering data collected, the goals of the alarm analysis are:

- Identify, rationalize, and document alarms and alarm setpoints that are required for safe operations
- Select minimum number and proper type of alarms
- Define unique responses to alarms
- Assign alarm priorities based on the severity of the potential abnormal condition and the maximum time to respond (MTR)
- Develop a database the defines the required alarms

Figure 33 summarizes the steps of the alarm objective analysis (AOA). AOA is a formal approach of alarm rationalization seeking agreement between operations and engineering on set points, operator responses, and priorities (D. a. Metzger). Due to either a new project being commissioned, a lapse in time since last analysis, or other reason, an AOA would be conducted, and a spreadsheet developed (Step 1). The engineering data would be fed into the data and validated (Step 2). The meeting would occur and based on alarms and their event consequences priorities would be established (Step 3). The results would be reviewed and approved (Step 4). Finally, the documentation would be collected and uploaded into a master alarm database (Step 5). After this was completed, each site had an approved alarm master database that included the critical data needed for the alarm audit and enforcement tool.

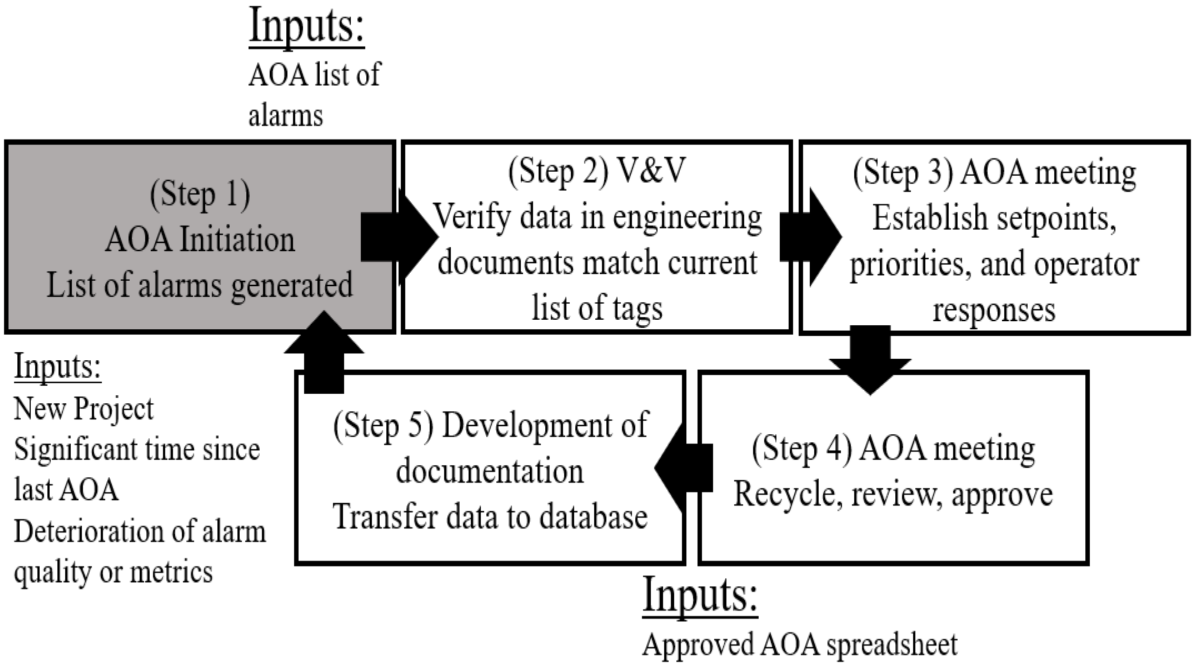
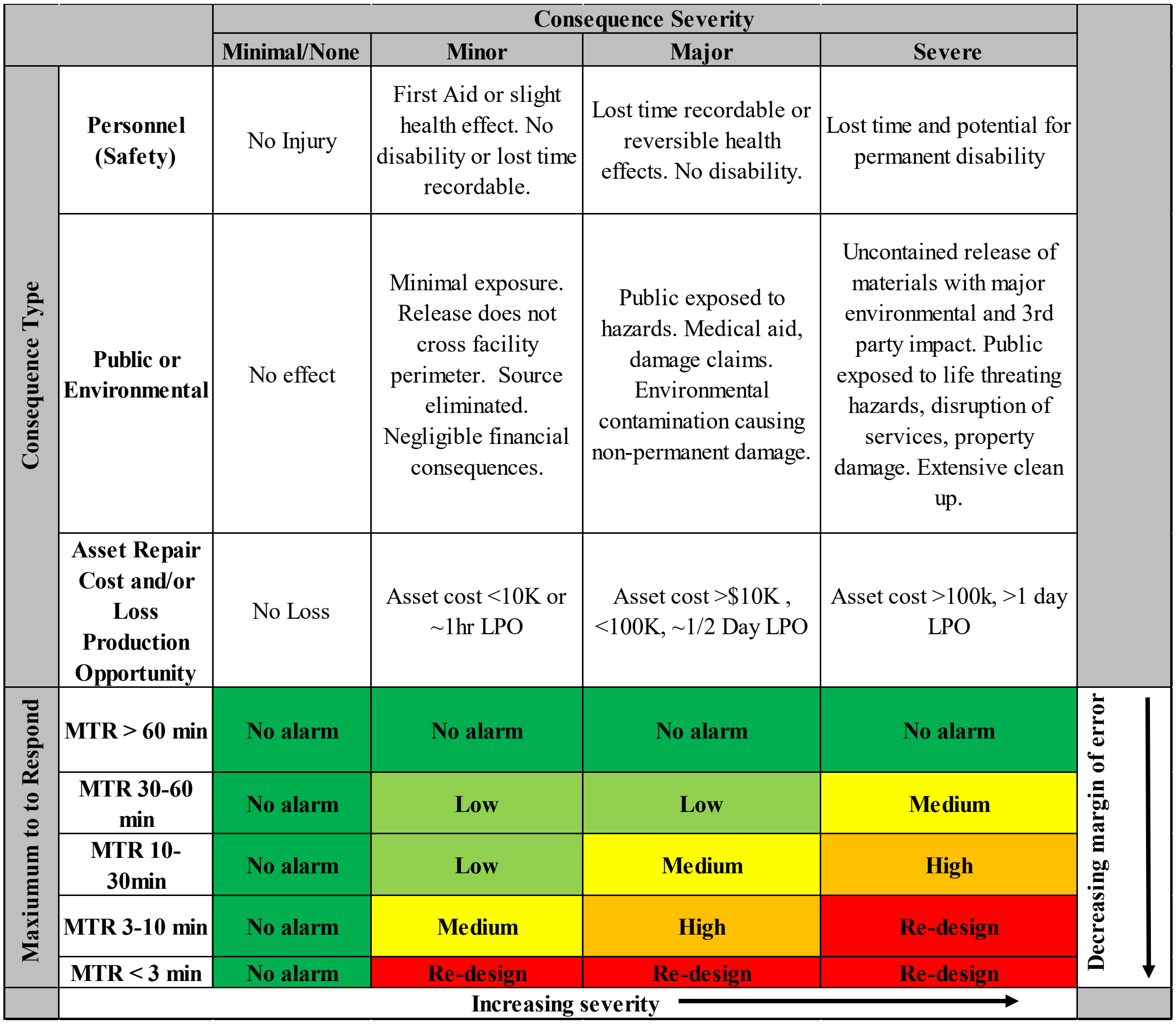

Inputs:
AOA list of alarms
(Step 1) AOA Initiation List of alarms generated
(Step 2) V&V Verify data in engineering documents match current list of tags
(Step 3) AOA meeting Establish setpoints, priorities, and operator responses
Inputs:
New Project
Significant time since last AOA
Deterioration of alarm quality or metrics
(Step 5) Development of documentation Transfer data to database
(Step 4) AOA meeting Recycle, review, approve
Inputs:
Approved AOA spreadsheet

Figure 33: Flow chart of the alarm rationalization done for these sites

Alarm rationalization can be a daunting exercise for large sites with thousands of alarms. Bulk rules, templating, and other techniques were used to ensure that systems with alarm train design, system diagnostic alarms (such as communication errors in network), and established alarm rules for a site were documented and leveraged to make process more efficient. To aid this, a spreadsheet was developed for each alarm where the initiating event could be recorded with the alarm to respond to the event. From there the maximum time to respond could be analyzed to determine the priority of the alarm. The spreadsheet recorded alarm tags, causes to trigger alarm, maximum time to respond (MTR), setpoint, engineering units, and severity of cause.

With this data captured, the team can utilize the simplified matrix shown (Figure 34) to determine the proper alarm priority. This technique allows the users to attribute a consequence severity to alarms in a variety of categories of consequence (consequence to personnel, to the environment, and to costs). This is then cross-referenced with the metric of maximum time to respond (MTR). An outcome of "no alarm" for this example means that the operator is still notified but it does not get elevated to an audible and high alert visual notification. Typically, an incident that has a high MTR and low consequence was re-engineered. This is due to if the incident is a slowly developing process, safety controls could be introduced to control the risk such as control loop or it should simply be a diagnostic alert to notify of maintenance required. Both MTR and severity can be modified in this table, depending on the characteristics of the site or system. Sites with advanced controls (quick shut-off valves) may allow for much shorter maximum times to respond compared to a site with slower controls (manually operated valves) and an operator that must leave a control room to do work inside facilities.

This method is aligned with a "classical" risk management profile and risk management methods where a risk profile is unique for each system, and risk tolerances can vary between industries and locations. Each industry or facility may develop their own unique alarm prioritization matrix due to the uniqueness of their site to include the operating model and how quickly operators can respond to alarms and the typical consequences.



		Consequence Severity				
		Minimal/None	Minor	Major	Severe	
Consequence Type	Personnel (Safety)	No Injury	First Aid or slight health effect. No disability or lost time recordable.	Lost time recordable or reversible health effects. No disability.	Lost time and potential for permanent disability	
	Public or Environmental	No effect	Minimal exposure. Release does not cross facility perimeter. Source eliminated. Negligible financial consequences.	Public exposed to hazards. Medical aid, damage claims. Environmental contamination causing non-permanent damage.	Uncontained release of materials with major environmental and 3rd party impact. Public exposed to life threating hazards, disruption of services, property damage. Extensive clean up.	
	Asset Repair Cost and/or Loss Production Opportunity	No Loss	Asset cost <10K or ~1hr LPO	Asset cost >$10K , <100K, ~1/2 Day LPO	Asset cost >100k, >1 day LPO	
Maxiumum to to Respond	MTR > 60 min	No alarm	No alarm	No alarm	No alarm	Decreasing margin of error
	MTR 30-60 min	No alarm	Low	Low	Medium	
	MTR 10-30min	No alarm	Low	Medium	High	
	MTR 3-10 min	No alarm	Medium	High	Re-design	
	MTR < 3 min	No alarm	Re-design	Re-design	Re-design	
	Increasing severity					

Figure 34: An adapted risk table for alarm prioritization

Based on the results of the alarm rationalization, the following set of process control network priorities were also developed to further assist operations organize and manage alarms (Table 13). These are the priorities that will be used on the human machine interfaces (HMIs) to alert operators and list the current active alarms. The distribution of alarm priorities is reviewed periodically to ensure that the risk profile of a site is not skewed towards too many urgent alarms, or too many low priority alarms. Like other safety analyses, the distribution of alarms from catastrophic, to critical, to high, and then to low should resemble a pyramid (Hu). An example of this is given below in Table 13 which was measured at one of the industrial process sites at which this research was performed. The middle column shows the current percent of each alarm priority along with what the project team thought the recommended alarm priority quantity should be for the sites.

Table 13: An example of the profile of alarm priorities for a sample site

Alarm Priority	Current	Recommended
HIGH	4%	2-10%
MEDIUM	16%	10-25%
LOW	80%	65-80%
Total	3201	N/A

Alarm rationalization and its formalized industry practice of alarm objective analysis is meant to give values to audit. Establishing the agreed upon alarm configurations for each site probably is the most time-consuming part of the process but it gives the data which will be audited by the automated system.

Building of Alarm Audit IT System

The next stage of the project was to construct an alarm audit system to communicate between the process control network and the alarm master database (also termed the alarm safety server) which contains the results of the rationalization exercise. The goal of the development was to provide an automated reconciliation audit and enforcement process to verify that alarm settings implemented in site's process control network (programmable logic controllers, human machine interfaces) are matching with the designed settings stored in the alarm safety server.

The process is designed to be run on a continual basis to identify changes made to alarm set points and priorities. It was also designed to determine if alarms were deleted or added at the process control network without requiring a manual update of the master alarm database. If automated enforcement is enabled, discrepancies are rectified by changing the settings implemented in the process control network to match the master database settings. If automated enforcement is not enabled, discrepancies are rectified by changing the implemented settings to match the master database settings manually. System change audit logs are created by the reconciliation process to document the discrepancies found.

The development to enable the communication between the HMIs, historian, and alarm safety server began with creation of alarm audit tag objects. These alarm audit tages are created by the custom master alarm audit object during initial configuration, and become a static resource once deployed. Plant expansions may require that more objects be created, in which case the master alarm audit object must be reconfigured for the larger number of alarms. A custom master alarm audit object was created using Visual C# and tool kits provided by software used in process control network. These objects scan the system for alarm objects, and create, configure and deploy alarm audit tag objects. The alarm audit tag objects have attributes that will be bound to the set points, priority and mode of the alarm objects in the network. One alarm audit tag object will be created for every 20 alarms. Figure 35 shows an overview of the audit software and hardware for the alarm audit system. Variations of this architecture could be adapted as well. The key quality of this architecture is that it follows the traditional layout of an industrial system in that logic controllers feed information to the control room human machine interface (HMI). From there, a table can be constructed of the alarm attributes and compared to the system of record. The system of record can then be compared to the data from logic controllers and HMI screens to identify discrepancies. Linking control network and business network is the key aspect of performing this audit method.

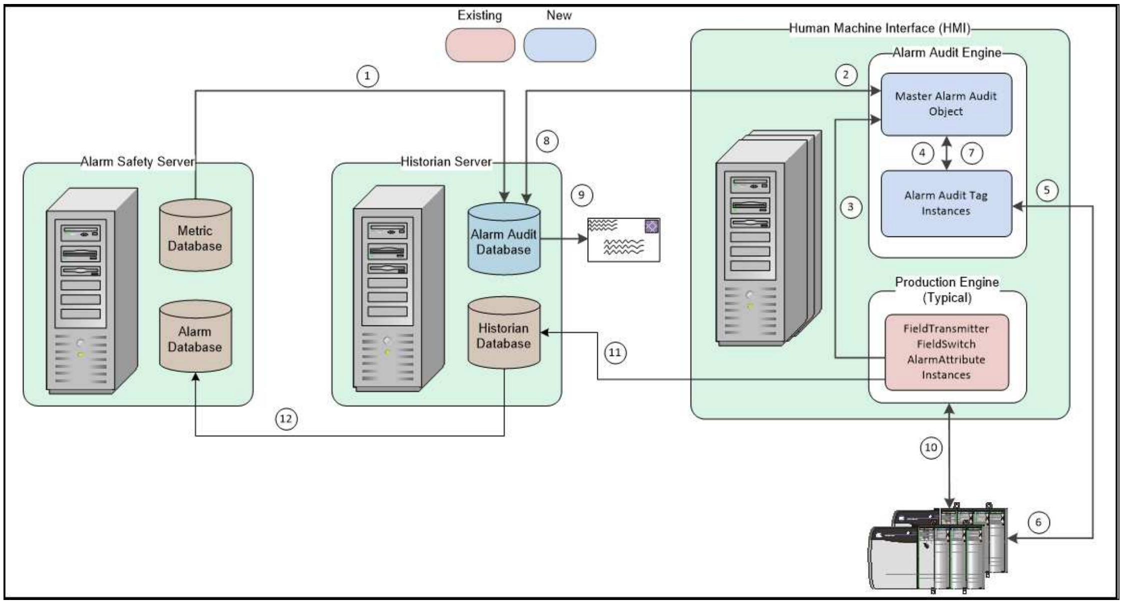


Figure 35: Overview of alarm audit system architecture as implemented

The automated alarm audit process is illustrated with the enumeration in Figure 35 and follows this data flow and architecture:

1. Alarm objective analysis (AOA) info in the Alarm Audit Database is pulled from the metric database by a scheduled job running in the Alarm Audit Database.
2. AOA info is transferred to the Master Alarm Audit Object in the HMI server.
3. The Alarm scanner iterates through production field transmitter, field switch, and alarm attributes objects on a regular basis and generates alarm audit tag instances to monitor alarm configuration.
4. Individual alarm configuration data is applied to alarm audit tag instances for comparison and correction.
5. Alarm audit tag instances monitor alarm configuration in HMIs/PLCs on a regular basis.
6. Alarm configuration discrepancies are corrected in PLC(s) when found.

7. Alarm audit tag instances report alarm configuration enforcement actions to the Master Alarm Audit Object.
8. Alarm configuration audit and enforcement information is transferred back to the Alarm Audit Database.
9. A triggered job in the alarm audit database produces email-able reports.
10. Normal PLC-to-HMI SCADA dataflow between the logic controllers and control room server
11. Normal alarm historization pull data from field to store in historian database
12. Normal alarm history transfer to alarm database from historian

The outcome of this process is a both immediate and regularly scheduled reports, depending on the desires of operations, that notify of changes to field settings for alarm setpoints and prioritization. The timing and information in the report can be customizable. An example of this was some sites experimented with immediate email notification if a HIGH priority alarm was changed and no longer matched safety database. With the previous rationalization completed and the alarm audit IT system designed, the system could be implemented.

Implementation and Experimental Methods

The automated alarm audit system was implemented in a set of oil and gas processing sites throughout Southern California. The industrial process equipment at these sites consists of rotating equipment, tanks, unit operations, and other heavy machinery. This equipment contains instrumentation and other process monitoring devices that are connected to logic controllers. When an alarm condition is met, the logic controller sends a signal to the control room.

As might be typical for large diffuse industrial processing equipment, the project began with little formal documentation for the alarm master database. By following the AOA process, the master alarm database was created. With master alarm database created and containing safety information about each alarm, the automated audit tool was created to cross reference field information with the safety information in the master database. After the system was created and following typical commissioning and start-up checks, the automated audit and enforcement tool went into service.

In the audit stage, periodic reviews are conducted to maintain the integrity of the alarm system and alarm management processes. Audits will likely reveal gaps that routine monitoring is unable to identify. This stage will also include modifications to the alarm philosophy and will identify the need to increase knowledge of the organization regarding the alarm philosophy.

For sites included in this project, alarms are automatically cross referenced with field settings and alarm objective analysis. The results are shown in a dashboard which can be accessed by all personnel via an application. The data can be reviewed at any time, but the data will be reviewed regularly at field meetings for each facility. meeting. Auditing of the control system alarm configuration against the Master Alarm Database was typically performed on a monthly basis for alarms that have been added, changed (e.g., limits, priorities) or deleted (including bypassed, disabled, inhibited, or inoperable alarms) where the auditing capabilities are automated and quarterly for manual auditing systems. Audit discrepancies relative to the alarm rationalization database are resolved with an appropriate Operations and Engineering review and correction.

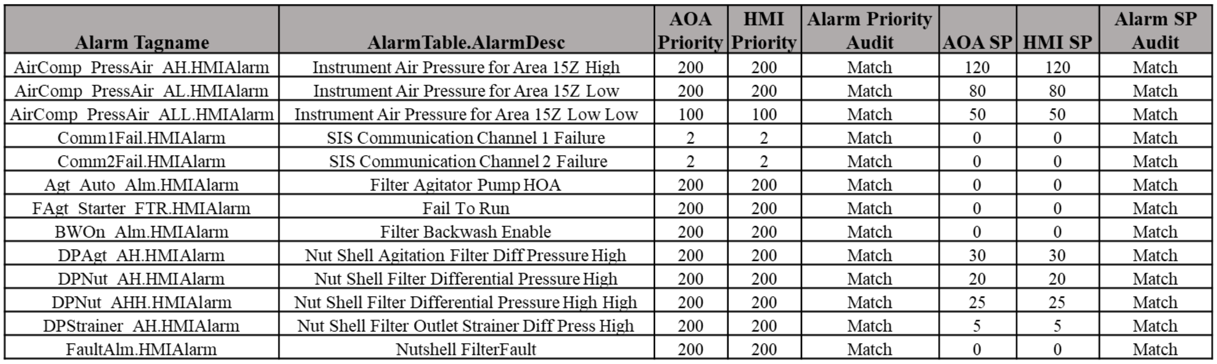
Results and Discussion

Immediately after formal commissioning and correction of technical issues, the system began generating reports to engineering and management. Reports included any discrepancies between the master alarm database and the field. This included new alarms, deleted alarms, changes to setpoint, and changes to priority. With this management oversite, the concurrent engineering work that was making multiple changes to programming by different contractors was more supervised since there was an automated way to check for discrepancies. After the creation of reports began, alarm meetings also occurred to regularly review the alarm information. The meetings focused on any data found by alarm audit tool, suppression of alarms, and “bad actor” alarms (top alarms in terms of frequency). With this data, operations teams could agree on any needed changes to alarm system and initiate management of change which would lead to an update of the master alarm database. At this time, the system has been deployed to seven large industrial sites with five more planned. One of the early adopters has utilized the data to reduce alarms by reclassifying them to a lower priority based on safety analysis. This site has reduced alarms from roughly 55% (decrease in overall alarm count prior to rationalization and system deployment) by utilizing the audit tool with the safety information in the alarm master database. All sites which have had the audit tool deployed have not suffered from any lost production or safety incidents due to multiple programming projects occurring at site.

Results of Alarm Audit

Table 14 shows an example of an audit that was automatically generating showing a series of sites where the alarms matched well with the alarm rationalization done formally. It shows an example of an alarm audit with no discrepancies where the audit matches the field settings. It shows that the formal rationalization (AOA – alarm objective analysis) matches with the HMI (human machine interface which is termed used for the control room screen). The alarm setpoints (SP) also match between rationalization and control system. This table shows satisfactory results that shows a match between control system and system of network.

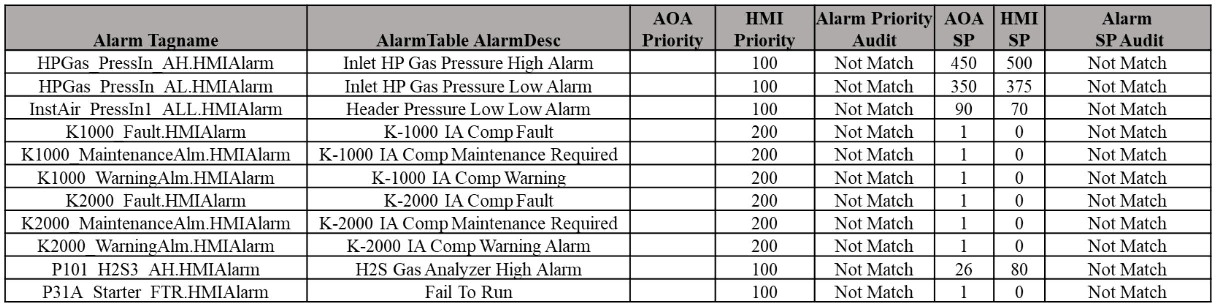
Table 14: An example of a satisfactory alarm audit.



Alarm Tagname	AlarmTable.AlarmDesc	AOA Priority	HMI Priority	Alarm Priority Audit	AOA SP	HMI SP	Alarm SP Audit
AirComp PressAir AH.HMIAlarm	Instrument Air Pressure for Area 15Z High	200	200	Match	120	120	Match
AirComp PressAir AL.HMIAlarm	Instrument Air Pressure for Area 15Z Low	200	200	Match	80	80	Match
AirComp PressAir ALL.HMIAlarm	Instrument Air Pressure for Area 15Z Low Low	100	100	Match	50	50	Match
Comm1Fail.HMIAlarm	SIS Communication Channel 1 Failure	2	2	Match	0	0	Match
Comm2Fail.HMIAlarm	SIS Communication Channel 2 Failure	2	2	Match	0	0	Match
Agt Auto Alm.HMIAlarm	Filter Agitator Pump HOA	200	200	Match	0	0	Match
FAgt Starter FTR.HMIAlarm	Fail To Run	200	200	Match	0	0	Match
BWOn Alm.HMIAlarm	Filter Backwash Enable	200	200	Match	0	0	Match
DPAgt AH.HMIAlarm	Nut Shell Agitation Filter Diff Pressure High	200	200	Match	30	30	Match
DPNut AH.HMIAlarm	Nut Shell Filter Differential Pressure High	200	200	Match	20	20	Match
DPNut AHH.HMIAlarm	Nut Shell Filter Differential Pressure High High	200	200	Match	25	25	Match
DPStrainer AH.HMIAlarm	Nut Shell Filter Outlet Strainer Diff Press High	200	200	Match	5	5	Match
FaultAlm.HMIAlarm	Nutshell FilterFault	200	200	Match	0	0	Match

Table 15 below shows a sample of poorer alarm rationalization that can be uncovered when performing automated alarm audits. This shows an example of poor results from an alarm audit where there is no alarm rationalization data (AOA) for priority and alarm rationalization setpoint do not match field settings. The results show no data for alarm priority and mismatches between setpoints of alarm rationalization and control system configuration. In this case, the need for rationalization of these alarms is now identifiable. The reasons for these alarms to have failed their automated audit ranged across sites. Some were due to sites that were not included in the rationalization effort. Some alarms were not documented in engineering documents (i.e., system diagnostic alarms for control systems such as low battery voltage or communication errors). Some alarms did not account for calculated setpoints that changed based on conditions, etc. The automated alarm auditing system enabled the speedy identification and correction of these problems. One of the main observations of the unsatisfactory audit results was missing alarms for network diagnostics. This includes network switches, communication errors between logic controllers, and induvial module failures. These were easy to fix due to the application of bulk rules across all the internal automation and network diagnostic alarms for all the sites which can typically be applied due to the similar cause and consequence of these system and diagnostic alarms.

Table 15: An example of poor results from an alarm audit



Alarm Tagname	AlarmTable AlarmDesc	AOA Priority	HMI Priority	Alarm Priority Audit	AOA SP	HMI SP	Alarm SP Audit
HPGas_PressIn_AH.HMIAlarm	Inlet HP Gas Pressure High Alarm		100	Not Match	450	500	Not Match
HPGas_PressIn_AL.HMIAlarm	Inlet HP Gas Pressure Low Alarm		100	Not Match	350	375	Not Match
InstAir_PressIn1_ALL.HMIAlarm	Header Pressure Low Low Alarm		100	Not Match	90	70	Not Match
K1000_Fault.HMIAlarm	K-1000 IA Comp Fault		200	Not Match	1	0	Not Match
K1000_MaintenanceAlm.HMIAlarm	K-1000 IA Comp Maintenance Required		200	Not Match	1	0	Not Match
K1000_WarningAlm.HMIAlarm	K-1000 IA Comp Warning		200	Not Match	1	0	Not Match
K2000_Fault.HMIAlarm	K-2000 IA Comp Fault		200	Not Match	1	0	Not Match
K2000_MaintenanceAlm.HMIAlarm	K-2000 IA Comp Maintenance Required		200	Not Match	1	0	Not Match
K2000_WarningAlm.HMIAlarm	K-2000 IA Comp Warning Alarm		200	Not Match	1	0	Not Match
P101_H2S3_AH.HMIAlarm	H2S Gas Analyzer High Alarm		100	Not Match	26	80	Not Match
P31A_Starter_FTR.HMIAlarm	Fail To Run		100	Not Match	1	0	Not Match

Resolution of Alarm Discrepancies in Audit

The resolution of alarm settings and priority mismatches between field PCN settings and AOA was designed to be as simple as possible without long management of change (MOC) procedures for easily explained mismatches. As previously mentioned, the review of alarm audit reports will highlight discrepancies. The automation engineer will cross reference with recent MOCs and if a MOC was completed fully they will update the AOA since proper change management was followed. If an MOC was not completed, they will review, with process engineering, process safety information (PSI) such as P&IDs and control philosophy to determine if new setting is consistent with the correct settings in field (i.e. the setpoint is correct in field and the same in PSI). If this is seen, they can hold a “fit-for-purpose” (e.g. a very informal agreement between operations and engineering) AOA to ensure the settings are correct. For a few alarms and a few discrepancies, this was done via email to make sure field settings are correct with confirmation of operations and engineering. If the new settings are satisfactory, they can update the AOA to reflect the field settings. If the PSI and AOA both do not reflect the field settings, the project engineering team should review to make sure it was not a recent project with MOC still in progress or project related issue to cause a discrepancy. If a project was not closed out properly, then project team should close-out MOC and update the AOA. In the last scenario of an unsafe condition that could arise: if the field PCN alarm configuration is not reflected in PSI and AOA it may be an unsafe setting. In this case, an MOC should be conducted, and alarm configuration corrected to change the setting in the field. The continual review and resolution of alarms audits described here is outlined in Figure 36, which shows that often alarm audit discrepancies were due to improper project close-out or poor communication between teams. It shows the general outline to correcting discrepancies between alarm audit and control network (PLC/HMI) configuration. Very rarely did we encounter changes done that put site in danger, but any examples were quickly identified and resolved.

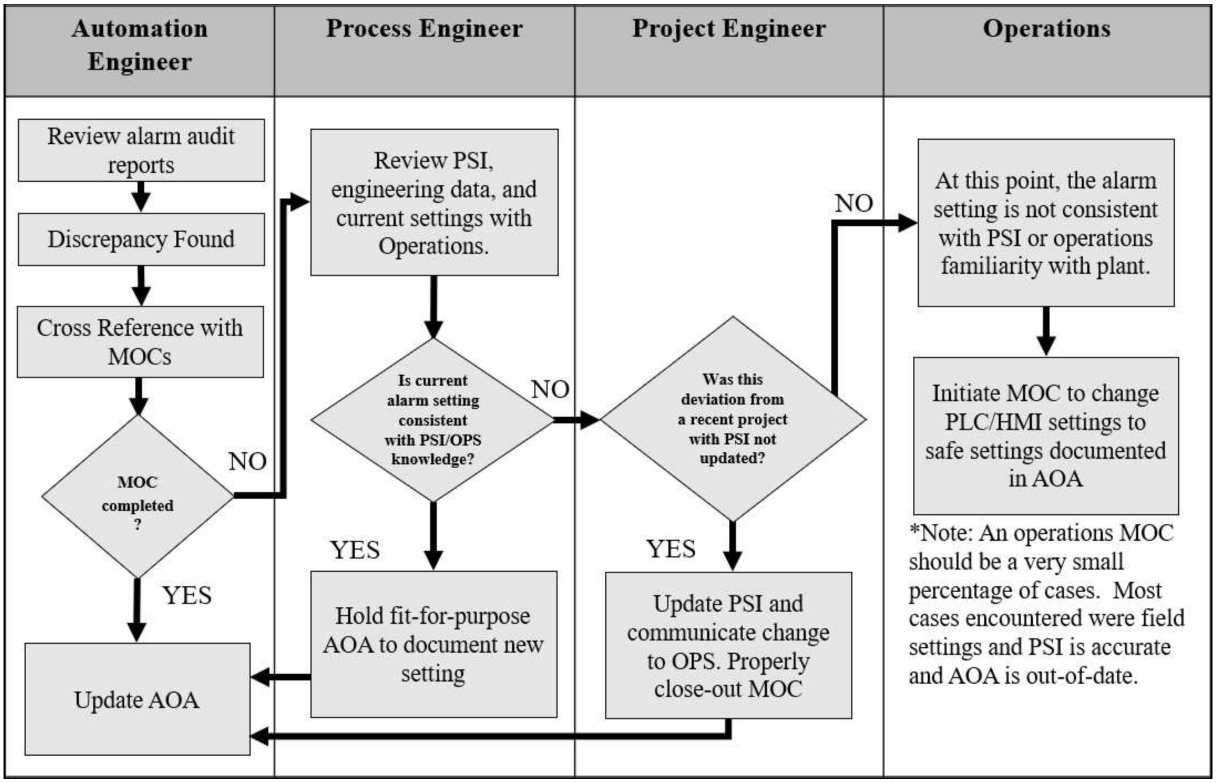

Automation Engineer
Process Engineer
Project Engineer
Operations
Review alarm audit reports
Discrepancy Found
Cross Reference with MOCs
MOC completed ?
NO
YES
Update AOA
Review PSI, engineering data, and current settings with Operations.
Is current alarm setting consistent with PSI/OPS knowledge?
NO
YES
Hold fit-for-purpose AOA to document new setting
Was this deviation from a recent project with PSI not updated?
NO
YES
Update PSI and communicate change to OPS. Properly close-out MOC
At this point, the alarm setting is not consistent with PSI or operations familiarity with plant.
Initiate MOC to change PLC/HMI settings to safe settings documented in AOA
*Note: An operations MOC should be a very small percentage of cases. Most cases encountered were field settings and PSI is accurate and AOA is out-of-date.

Figure 36: Workflow to correct control network alarm misconfiguration.

As a reminder: alarms change over time due to projects, small field settings, and other normal situations. Most audit mismatches were due to a recent project. Very rarely was an unsafe condition (where alarm configuration was unsafe and incorrect) encountered but the continual changes to sites and parallel ongoing projects made alarms settings continually mismatch the AOA. This highlights that after the audit system was put into place, discrepancies did occur as part of the on-going changes to the site, but they would most of the time be handled easily and most could be explained. This may not be possible with manual or yearly spot-checks of alarm configuration versus the AOA where hundreds of differences between safety database and field configuration could be overwhelming or difficult to explain.

Continuous Improvement in Alarm System Performance

These types of alarm audits were performed automatically and continuously for the 21,812 alarms present in this process control systems at all sites. Approximately 6 months of continuous automated auditing and alarm rationalization were required to reach a steady-state of alarm rationalization. Several months of clean-up we got the results shown in Figure 37 and Figure 38. Figure 37 shows the average results after roughly 9 months at sites for setpoint audit which cross references the field settings with alarm safety information and thresholds. Figure 38 shows the average results after roughly 9 months at sites for alarm priority audit which cross references the field settings with alarm safety information. Priority mismatches usually meant no alarm rationalization was performed and no data exists. Still not matching 100% for both setpoint and priority was attributed to the fact that more projects had occurred, and constant changes were occurring at plants. Part of the "entropy of alarms systems" described previously is that alarms systems never are truly static. Many things such as changing process conditions, additions of projects, and other factors make 100% alarm audit and enforcement an unreasonable goal. Our goal was to review audit results regularly and adjust. Often new projects simply forgot to update the alarm safety database after completion. A risk-based approach was also used to set metrics for higher urgency alarms to be fixed sooner than less urgent alarms. For example, often the alarm priorities used were High, Medium, and Low based on consequence severity and maximum time to respond.   High urgency alarms had to be fixed within two weeks. Medium was 1-month and low was 3 months to fix discrepancies between audit and field settings. Considering Table 13 previously, where we aimed to have a pyramid style for alarm priority distribution, and that that most changes were for low priority alarms, these metrics were achievable.

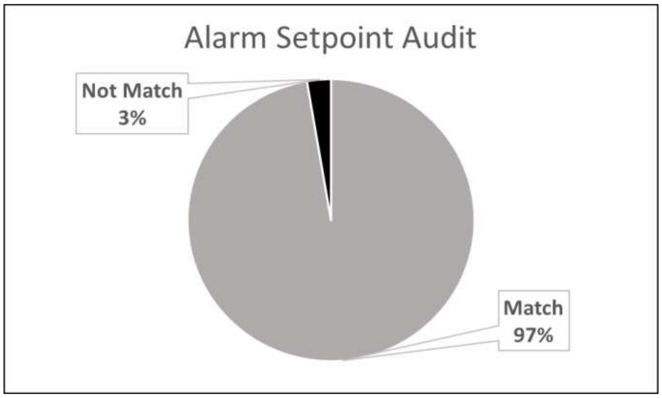


Figure 37: Setpoint audit 9 months after implementation

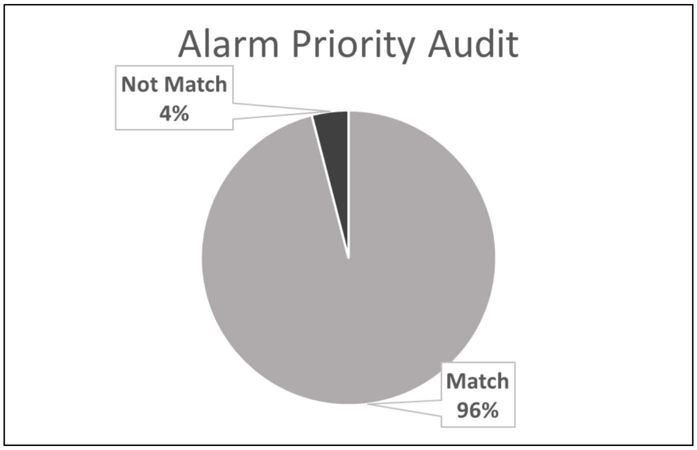


Figure 38: Priority audit after 9 months after implementation

An added benefit of rationalizing alarms and building the audit tool was ability to reverse the degradation of the alarm system. With work completed, it was easier to adjust alarms to meet new condition of the facility. The 9-month improvement since each audit tool was deployed per site is shown in Figure 39. This shows the decrease of alarms over time as alarm adjustments were made to align better with alarm database. One of the biggest improvements was applying bulk rules for process control network alarms across all sites. Alarms such as low voltage in logic controllers, communication errors, uninterruptible power supply (UPS) fault alarms, and many other alarms for equipment that can be the same across sites were able to have standardized/bulk rule alarms.

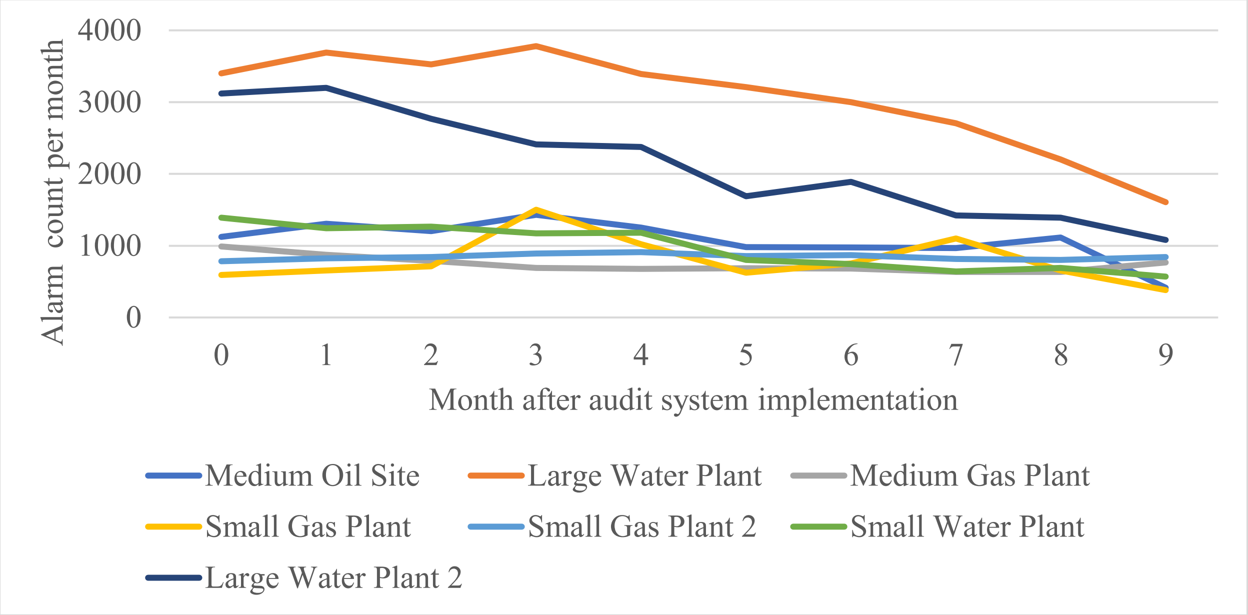


Figure 39: The long-term decrease in alarm frequency after audit system deployed

Similarly, standardization of alarms for other common equipment such as instrument air compressors, equipment in train configuration, and custody transfer equipment led to decreases of alarms. It was interesting to see identical equipment with different alarm setpoints and priorities but understandable if they were installed years apart. Table 16 summarizes the decrease after a 9-month period after rationalization and audit system for alarms was put into place. This shows alarm reductions attributable to the alarm audit project after implementation and roughly nine months of alarm management. Alarm reductions attributable to the alarm audit project after implementation and roughly nine months of alarm management resulted in several significant drops at individual sites. One site ("small gas plant") had a slight increase due to ongoing expansion projects and being close to capacity. This site saw a small increase in alarms but due to activity and being close to capacity, the increase in alarms for the site could be explained as a condition of the site rather than a defect in the audit tool.

Table 16: Summary of alarm reductions across Southern California

Site	Reduction in Alarms 9mo after Implementation
Medium Oil Site	-63.1%
Large Water Plant	-52.8%
Medium Gas Plant	-22.6%
Small Gas Plant	-35.4%
Small Gas Plant 2	7.8%
Small Water Plant	-59.1%
Large Water Plant	-65.4%
Total	-52.0%

Conclusions

Alarm audit is continually identified as a challenge for alarm management (Goel). It is defined by ISA 18.2 in simple terms but depending on the specific site it needs to be applied to, it can lead to confusion and not be implemented. Software exists to enable alarm audit, but it can often be industry specific (Chu), (B. Hollifield). The goal of this work is to document the building of an alarm audit and enforcement system that can be applied to other industries. Many industries face this challenge such as health care where the uniqueness of each patient require different setpoints to alert health care professionals. Nurses and control room operators can suffer the same challenges of alarm overload and fatigue due to misconfigured alarm setpoints

(Maine). This method shows it can be as simple as (1) setting setpoints via rationalization and (2) comparing two tables: rationalized alarm setpoints and current settings. This follows ANSI/ISA guidance and can be applied to any situation.

Prior to project, a series of industrial production sites did not have an automated mechanism of reconciling and verifying if implemented alarm settings matched designed settings. Verification and validation are important safety concepts, and they should include consideration of alarming. Implementing this audit tool allowed sites to continue to mature validation and verification of appropriate barriers and safeguards. Alarm configuration changes occur in alarm generating systems sometimes without corresponding changes in the master alarm database. Prior to work, there was no means of validating alarm master database with field alarm configuration systems on a periodic basis. This audit tool bridged this gap.

This work led to development of an audit tool that checks for alarm setpoint, alarm priority and alarm HMI tag mismatch but also led to improvements in safety by checking for unauthorized changes to alarm settings. It allowed operations to continually verify if alarms were acting as in their intended role as a protection against hazardous events. Giving the operator a reliable alarm configuration information helps the operator maintain the plant within safe operating limits and recognize a potentially hazardous condition early and take proper action. Some industrial sites may not have a similar need for rigorous, automated alarm auditing due to less work activity or less frequent changes to alarms. The need for a alarm audit took can vary depending on industry, capital investment, and management controls already in place.

Research question #3 (How can we establish monitoring to improve identification of obsolete components and unapproved changes to automaton systems?) is answered with this work and the results are summarized below:

- Expanded on high level industry standards to propose and execute a risk-based and automated method to alarm audit. This proven method can be applied to other industries and professional practices.
- Utilized a modified alarm objective analysis (AOA) to document safety thresholds across several industrial sites.
- Developed and published a reference network architecture for audit of alarms.
- Expands on industry standards to develop an automated (vs. conventional manual) method way to audit alarms.
- This method can be applied to other industries and professional practices.

The result is a system that allows sites to continually verify and validate all facility alarm configurations are safe by use of an automated tool. Future work is looking at ways to further study high frequency alarms against the master alarm database to better rank alarms which need attention. The next steps after establishing this link between system of record and physical system is to theorize ways to generalize this process so that it can be applied to other information and different industries. Alarm attributes are only a small part of the total information for a system. This is one tool to help but this link and audit method could be applied to other network attributes and other systems. A rationalization of an attribute and then linking to physical system to allow for an automated audit could be expanded beyond alarms.

Chapter 7 – Relation to Digital Twinning and Future Work

Digital twin overview

"Digital twin" is a concept of linking physical systems with digital copies to aid in development, testing, maintenance, and a variety of other purposes. The basic architecture of a digital twin links physical space to virtual space through sharing of information. The physical space sends data to the virtual space where it can be analyzed, then information can be sent back to physical space to aid the design or operations of the physical system. The main goal is facilitating work activity and limiting physical resources required. The goal is to substitute information for wasted physical resources. Digital twins are formally defined as:

> "A Digital Twin is a set of virtual information constructs that fully describes a potential or actual physical manufactured product's form and behavior from the micro atomic level to the macro geometric level. At its optimum, any information that could be obtained from inspecting a physical manufactured product can be obtained from this Digital Twin" (M. a. Grieves)

The three tools developed over the course of this dissertation all shared a common theme in that they are versions of a low fidelity digital twin. The obsolescence management techniques leveraged maintenance information and other system of record data to provide information on which components needed proactive replacement. The simulation objective analysis built a virtual twin to test the programming quality prior to commissioning. The alarm audit system directly connected the cyber-physical system to the system of record to ensure the integrity of the system. After many of the upgrades were completed, the expansion of the digital twin continued after these three initial successes were achieved.

The digital twin field, like the automation field, has potentially many interfaces with systems engineering in that it combines many cross-disciplinary fields and aims to make the operations of the system more efficient. One of the challenges though is quantifying the costs and benefits of digital twin uses. It remains very difficult to assess the benefit of a digital twin prior to undertaking the development of one. This is shown graphically below in Figure 40 which is a reproduced graphic from 2021 ASME Digital Twin conference "Digital Twins Today - Advancement & Opportunities" (M. Grieves).

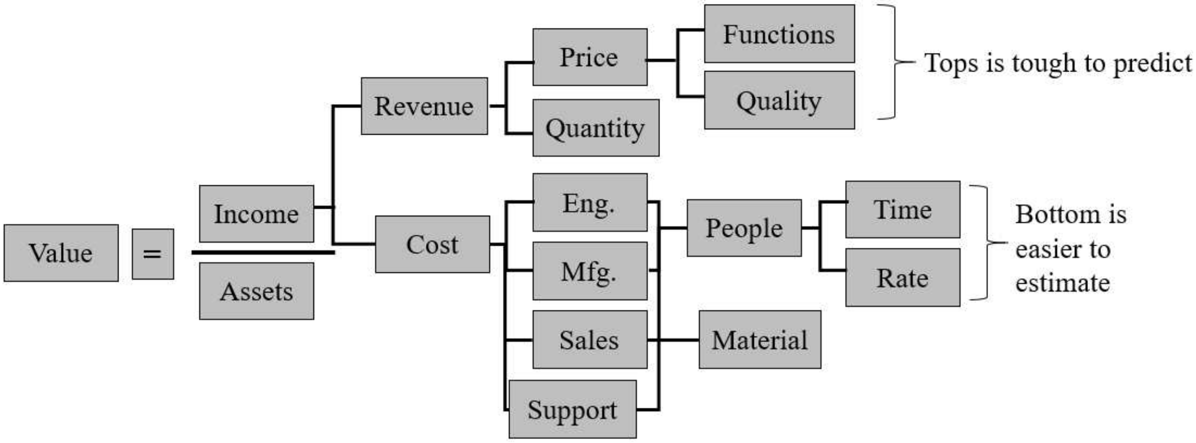


Figure 40: Digital twin value map from Michael W. Grieves LLC 2003-2021

This digital twin value map is meant to highlight some of the challenges currently faced in the digital twin field. It shows that a value of digital twin is quantified by the monetary income it generates. The income is a trade-off between the revenue it generates and the cost. The cost is easier to estimate, as estimates of engineering, design, and interfaces for digital twin can be had. It is much more difficult to estimate the revenue a digital twin will generate. This is because there are often many intangible benefits of a digital twin that will not be measurable until after its completion. This was seen in this dissertation work through the example of developing an alarm audit tool which also had added benefits of assisting alarm management and overall reduction of alarms at many sites. The future work of automation subsystem upgrades and management will focus on continual development of digital twin models to aid their subsystem management. To prevent the implementation of overly complex and therefore under-used digital twins, a strategy to focus on low cost, low fidelity methods and grow systematically as additional functionality can be considered. Legacy systems can have digital twins built slowly overtime as previous tools are proved out, to avoid any risks associated with deployment of overly complex and expensive digital twins. The remaining sections of this chapter detail ongoing work and a theoretical digital twin maturity model to develop higher value, lower cost digital twins.

Future work for automation subsystems and digital twins

After the development of the methods and tools to upgrade automation systems, future work continued to establish a real-time monitoring system of the automation systems of industrial sites. The goal was to extend the auditing of the alarm configurations previously discussed to all automation equipment. The next development underway is a PLC reliability scanner that is solution intended to further short-term and long-term automation reliability and integrity goals. Short-term goals of relevant-time monitoring of the automation ecosystem are accomplished by providing an continuously updated list of online devices and an indication of their configuration and health (for example, current faults and running program name). Long-term goals of inventory and life cycle management are accomplished by generating the traffic needed for systems to build a picture of the network, traffic that normal operations may not produce. Maintaining the short-term and long-term reliability of the control systems infrastructure depends on having accurate inventory and accurate alerting to exceptional conditions. The features of this future work include:

- Asset Lifecycle Management – knowing what devices are installed and in what quantity for upgrade/end of life/compatibility planning
- Cyber security recommendations and requirements - to maintain asset information in the PCN (process control network) enterprise inventory to aid cybersecurity concerns,
- Reliability - Status of processor state (major or minor faults, key switch position – ability for PLC to be remotely changed, etc.) that may not be identifiable through "ping" or that depend on logic executing,
- Status of non-controller devices - Redundancy modules, communication modules.

But this future work has many challenges which include:

- Large-scale systems with frequent additions controllers or modules, changes of version, replacement of failed parts – not always with wide notification of changes,
- Different revisions of programming standards across sites. Identifying and building monitoring tags for controllers and modules individually would be costly and difficult,
- This monitoring system requires a new method to determine inventory by listening for specific traffic that has module information in it, and normal operations rarely generate this type of comprehensive traffic.

But some of advantages of considering digital twinning with current automation equipment is:

- Almost all information needed to meet short-term and long-term reliability goals are available through normal device messaging,
- All modern automation products support inventory messaging and conform to the parts to the standard required for reliability/inventory,
- Devices of interest can be reached from the PCN and through backplane/network routing,

- Each device can be individually addressed without any modification to the processor program or before-hand knowledge of the system.

The core functionality of the solution (network browsing and device interrogation) is accomplished with a typical automation processor in the process control network environment. Logic controls the execution of state machines that browses the network, drilling through the different network connections found, and exploring each connected device. As new devices of interest are found (such as controllers and redundancy modules), the path to each is added to a fast-polling list. A separate state machine iterates through these lists, polling each device in succession for different status attributes. The results of these polls are stored in a set of tags that is read and stored by a data historian.

Visualization of collected data is accomplished with a typical dashboard. Data is collected from the scanner as time-series tags, with each time value representing the equivalent of a “record”. Time-series values are retrieved by the dashboard and then re-arranged through a series of transformations to a table that is keyed on unique devices. Summaries of this table are presented as a dashboard (such as number of major faults and a summary of key switch positions). A visual example of the PLC reliability scanner is shown in Figure 41. This shows a visual representation of polling automation equipment in an industrial setting to find minor and major faults. Colors indicate relative severity of issues ranging from components no longer supported by manufacturer to minor and major faults of equipment. Red is a major fault (inability to operate or be fixed) for an automation/cyber physical part or component. Yellow is a minor fault (able to be fixed in most cases). Purple is a part or component that is no longer being manufactured my manufacturer so special considerations need to be replace the hardware. Green is the firmware for the equipment is not latest approved version in-use for field.

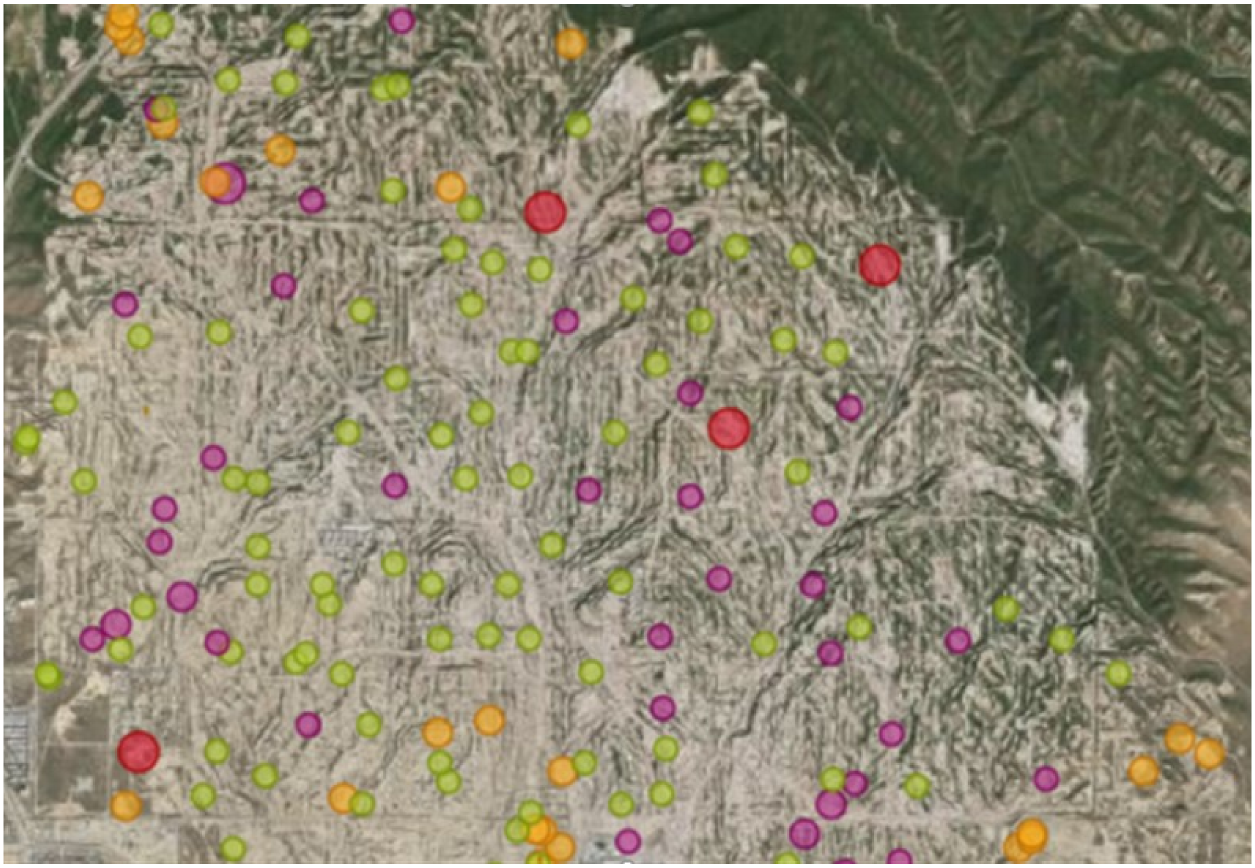


Figure 41: Large scale polling automation equipment overlayed on a map

The action behind the PLC Reliability Scanner is that almost all automation assets of interest can be accessed from the PCN. Those that are not directly connected to the network can be accessed through backplane routing. The backplane is the piece of equipment at the logic controller that integrates all the major automation components at that site. As backplane routing addresses can be generated without prior knowledge of what is expected to be at the address (if anything), it is possible to explore the entire address space for an unknown network, at will. The application of this process is performed through a set of rules that can allow the PCN network range to be interrogated iteratively. Importantly, keeping track of the devices already identified will prevent the system from discovering parallel paths and looping through these endlessly. This process is referred to as “subnet crawling” in the PLC reliability scanner. Network crawling is already established for many telecommunication applications (Areekijseree) but “subnet crawling" a novel technique when applied to scan industrial networks for inventory assessment. In short, each IP address is scanned on the network, the details of each device can be collected (including model, firmware version, and any other important data) to be built into a digital twin. This digital twin would then be used to assess the current state of the automation subsystem and both short term and long-term concerns previously outlined could be addressed.

The continual development of digital twinning of the automation systems is a key focus area for future work due to the current integration of automation with real-time monitoring. Automation systems already collect large amounts of data and have functionality already established to collect system of record information. Combined with the current challenges already discussed on how automation systems have unique characteristics with complex industrial systems, they are an area of high potential to leverage digital twinning. The basic process going forward to expand the automation infrastructure to digital twin technology is to collect real-time data on the existing components. The existing component in the field is considered the "System of Record" (or the database of the existing condition of the system). This PLC reliability scanner method, in addition to alarm audit tool, can serve as a baseline to extend the fidelity of the digital twin for the system. By collecting alarm configuration and a real-time catalog of automation equipment, a system can be upgraded to have added functionality. Examples of this include adding maintenance records and engineering data to the digital twin over time. This is outlined in Figure 42, starting with the automation system of sensors and logic controllers, the digital twin can be extended to transactional data from field (maintenance, production data, and operator routine duties) and static data (such as engineering documentation). This figure shows an overview of the digital twin architecture for industrial systems.

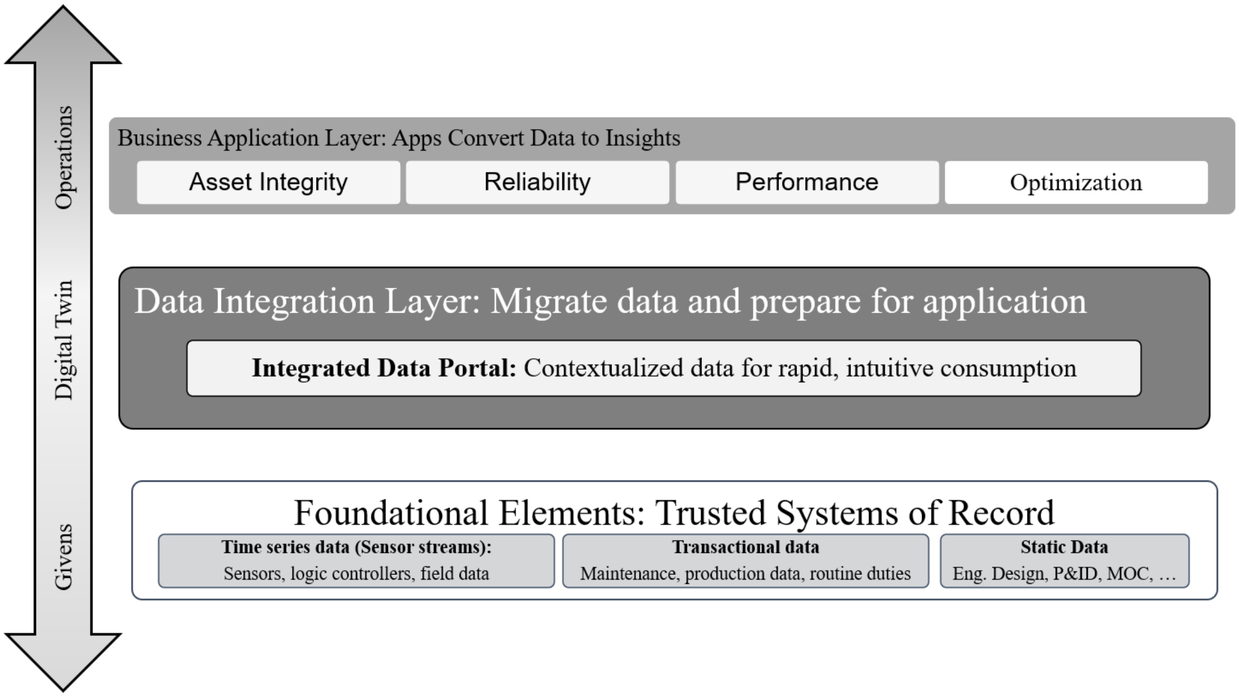

Operations
Digital Twin
Givens
Business Application Layer: Apps Convert Data to Insights
Asset Integrity
Reliability
Performance
Optimization
Data Integration Layer: Migrate data and prepare for application
Integrated Data Portal: Contextualized data for rapid, intuitive consumption
Foundational Elements: Trusted Systems of Record
Time series data (Sensor streams):
Sensors, logic controllers, field data
Transactional data
Maintenance, production data, routine duties
Static Data
Eng. Design, P&ID, MOC, …

Figure 42: Simplified digital twin architecture for industrial systems

The conclusions of this report show the results of establishing a digital twin for alarm systems and the advantages it achieved. The current focus of the research going forward is applying the PLC reliability scanner to facilities. These have themes of digital twinning that can be continually applied to expand the functionality of the digital twin overtime. The gradual implementation of digital twinning is recommended. Often highly complex digital twin models can be employed, however their functionality does not improve system performance due to difficulty in complexity and maintenance. Principles need to be established to slowly grow the fidelity of a digital twin to ensure adoption and continual improvement based on the value they provide. They key message is that attempting to deploy a high fidelity, overly complex digital twin can be cost prohibitive and difficult to implement. Lower fidelity methods, such as outlined in the audit method discussed in previous chapter, can be easier to achieve.

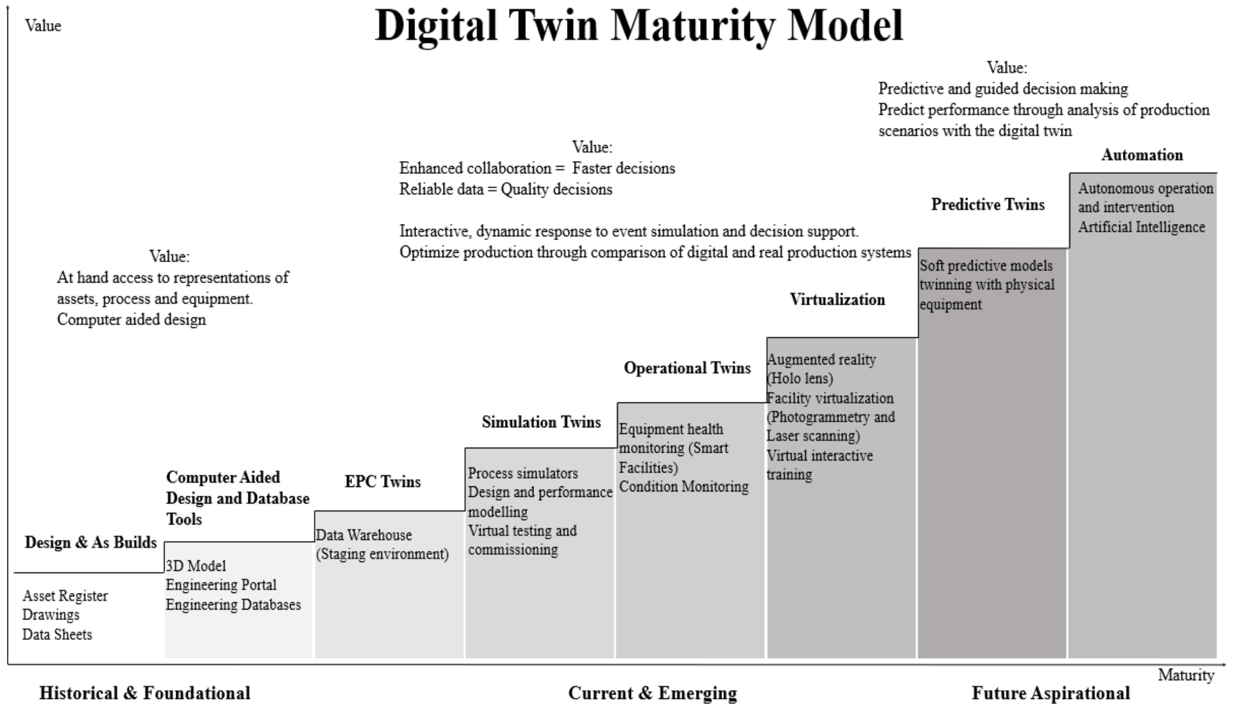
The following principles for digital twin architecture was developed to help users develop architectures for any system in development and represent what was the basis for developing the digital twin architecture described.

- Minimize Regretted Spending: When delivering tactical solutions, their design must consider the future transition to a strategic architecture, and must avoid lock-in
- Reduce Manual Handling: Whenever possible, data should be streamed and directed in through an established pipeline, with key business rules and metrics applied automatically.
- Adaptable & Future Proof: The solution architecture should be extensible and scalable to support continue growth in the number of users and the volumes of data processed.
- Speed, Timeliness and Scalability: The platform should support high volume, velocity, variety, and veracity of data – structured, unstructured, and sensory. It should be scalable, and real-time, where needed.
- End User Experience is the Key: When using the visualization applications from different locations, user experience is to be considered wherever feasible.
    - Local instances of data or hosting may be required where network connectivity or performance issues are identified.
    - Edge-like instances of applications/servers so as to best serve specific user groups.
- Vendor-agnostic Architecture: Core architecture components (layers) should not be driven by vendor-specific requirements where feasible. The vendor products should be consumers of the architecture not controllers.

    - All use cases may not elect to use the same vendor for digital twin applications, the architecture should support this and not require fundamental changes.
- Duplication of data sets within the platform is to be minimized, where applicable.
    - Performance constraints and maximizing end user experience may require holding the same data in multiple places.
- Data should aim to be centralized for shared access, avoiding the creation of silos for bespoke purposes.
    - Local disk storage of data should be kept to a minimum where feasible.
    - If multiple applications require use of a shared data set (e.g. photos, 3D models) then a centralized storage option should be considered first and only when performance or other requirements dictate storing a copy on local disk. This also minimizes having to manage synchronization of data.
- System of record data should not be modified when initially accepted.
    - Loading data into the platform should ensure a level of trust (data has not changed) and minimize the drift between the source and the platform (where feasible, exceptions would be bound by the frequency of data ingestion into the platform).

Digital twin maturity model

A digital twin maturity model is described here which captures several concepts that were covered in this discussion. Often, very complicated digital twins are proposed that are expensive to develop and deploy. This digital twin maturity model highlights a cost-effective way that many can use the concepts of digital twin, no matter how small their system is. Most systems start with foundational information such as drawings and data sheets. This work shows that using individual use cases and available data from the system of record, one can deploy a simple digital twin with a foundational architecture.   From there, the functionality can be incrementally expanded towards more mature digital twins. One can develop a digital twin with foundational information and gradually expand it to higher levels of maturity. Deploying digital twins requires a multifaceted technology strategy which is a precise combination of leadership vision, technology investment, and prudent execution.



Value
Digital Twin Maturity Model
Value:
Predictive and guided decision making
Predict performance through analysis of production scenarios with the digital twin
Value:
Enhanced collaboration = Faster decisions
Reliable data = Quality decisions
Interactive, dynamic response to event simulation and decision support.
Optimize production through comparison of digital and real production systems
Value:
At hand access to representations of assets, process and equipment.
Computer aided design
Automation
Autonomous operation and intervention
Artificial Intelligence
Predictive Twins
Soft predictive models twinning with physical equipment
Virtualization
Augmented reality (Holo lens)
Facility virtualization (Photogrammetry and Laser scanning)
Virtual interactive training
Operational Twins
Equipment health monitoring (Smart Facilities)
Condition Monitoring
Simulation Twins
Process simulators
Design and performance modelling
Virtual testing and commissioning
EPC Twins
Data Warehouse (Staging environment)
Computer Aided Design and Database Tools
3D Model
Engineering Portal
Engineering Databases
Design & As Builds
Asset Register
Drawings
Data Sheets
Maturity
Historical & Foundational
Current & Emerging
Future Aspirational

Figure 43: Digital twin maturity model

This “Digital twin maturity model” was developed to illustrate that digital twins can be categorized and described in a similar way to simulations. Simulations have different levels of fidelity where increasing levels of accuracy and detail has increasing levels of value and cost to develop. This model shows that often the path to a high maturity digital twin can be done incrementally.   Many times, digital twin development is not pursued due to the proposed twin being too mature and is too expensive to achieve.   On the other hand, this work details three versions of low fidelity digital twin models that were successfully executed. The obsolescence management system would be considered utilizing a data warehouse of computerized maintenance system to assess remaining life of components. The simulation testing is an example of simulation twins to aid virtual testing and commissioning. The alarm audit tool and PLC reliability scanner is an example of equipment health and monitoring.   This shows that often low fidelity digital twins can be utilized as a starting point to continually digital twins at legacy systems. Future work is encouraged to develop digital twins for automation and cyber physical systems along this digital twin maturity model.

Chapter 8 – Research Contributions and Conclusions

This research addresses the unique systems engineering challenges that are present with automation upgrades in industrial settings. The traditional systems engineering framework establishes a proven method for building systems form concept development to operations and manufacturing. It does not go in depth on how to upgrade individual subsystems for a complex system such as a continually running industrial site that undergoes subsystem upgrades. The main body of this research includes methods to identify obsolescence in automation systems, simulate the upgrade, and tools developed to audit the performance. This represents an advancement for the systems engineering field. Systems engineering typically emphasizes the development of a system from concept to detailed design to manufacturing and operations. But often, legacy systems need to be upgraded due to aging subsystems or customer demands changing over time. A proposed mapping of this work to the systems engineering "V" is shown in Figure 44. This shows proposed modifications to the systems engineering lifecycle to address obsolescence and upgrading of subsystems. Chapter 3 discussed the development of obsolescence (either through product discontinuation or age of parts) and the development of a way to quickly identify subsystem obsolescence and the parts to upgrade. The arrow is reversed to show how an operational system can revert back to design easily with the developed theory and tools shown. Chapter 4 strengthens the ability to test programming upgrades wit controls testing with simulation. This aids the "build and verify" portion of the systems engineering framework. Finally, chapter 5 showed a way to improve the system validation step of the framework. It does this by showing how to develop and deploy a low fidelity digital twin to aid audit of a system.

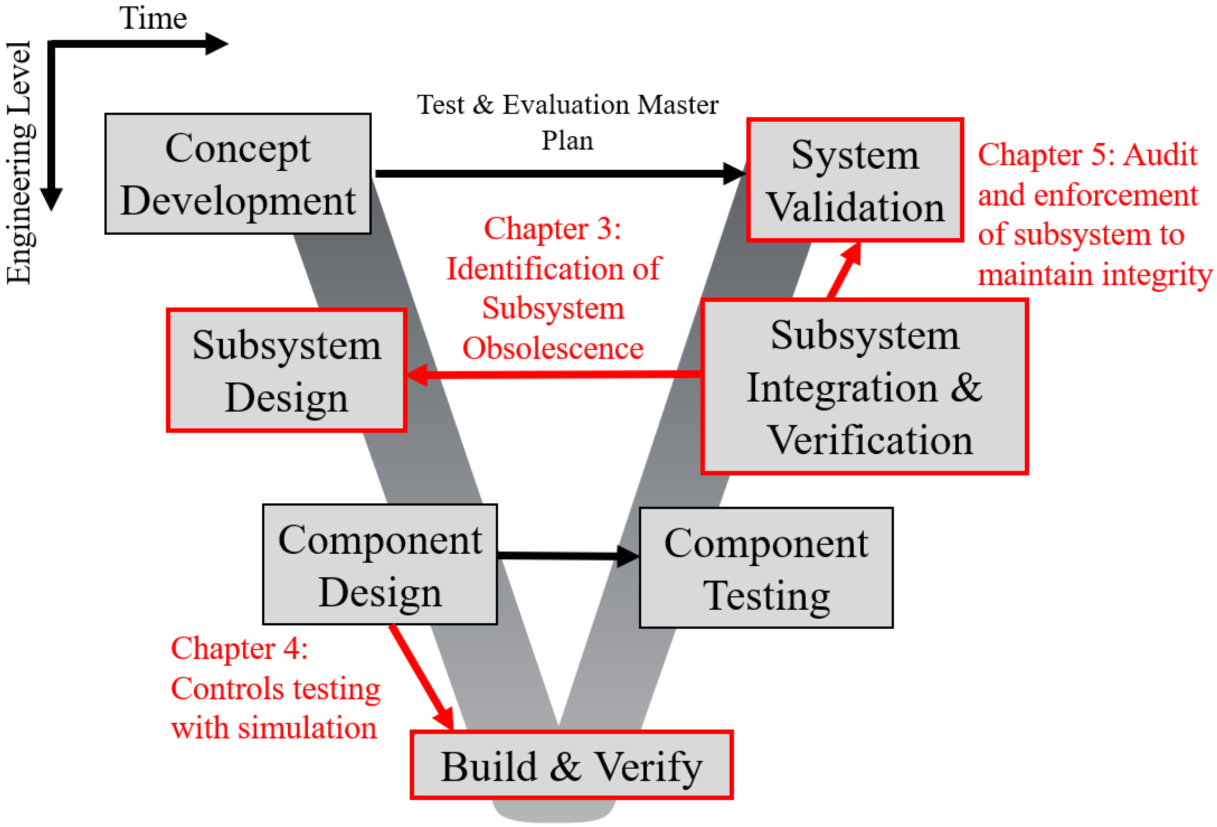

Time
Engineering Level
Concept Development
Test & Evaluation Master Plan
System Validation
Chapter 5: Audit and enforcement of subsystem to maintain integrity
Chapter 3: Identification of Subsystem Obsolescence
Subsystem Design
Subsystem Integration & Verification
Component Design
Component Testing
Chapter 4: Controls testing with simulation
Build & Verify

Figure 44: Research contributions mapped to the systems engineering "V" framework

This mapping illustrates that over long periods of time, an operational system may no longer meet its previous functional requirements. Examples of functional requirements that may lapse overtime and are automation related may include cyber security requirements, requirements to easily make modifications to programming, requirements to replace components manufactured by an outside party, and requirements to apply controls and advanced modes of operation to sufficiently achieve production goals. All these examples of functional requirements may be stated early in the concept development but can lapse over time due to reasons including obsolescence in automation systems, alarm entropy, and discontinuation of components.   The updated proposed systems engineering lifecycle illustrates that if a subsystem is unable to meet functional requirements the "V" process can be repeated to upgrade a major subsystem. This is done by identification of a subsystem and any associated components which can be aided by continual audit of the system. Secondly, once a scope of upgrades to the subsystem is identified, an efficient method is needed to test the proposed upgrades with an efficient simulation and testing method. Finally, an ongoing audit of the configuration of the system (either done with the audit tool described or future high fidelity digital twin method) must continually monitor the system to make sure the functional requirements are maintained. My research discusses the uniqueness of automaton upgrades within the systems engineering framework. Future work will continue to apply digital twinning strategies to extend the monitoring of automation systems with the aim of aiding easier upgrades.

Table 17 defines the summarized research contributions from this work. It is universally understood that petroleum and other energy systems are undergoing a transformation where legacy systems are being upgraded to lower carbon intensity systems (Zou). The use of the methods and tools discussed here will aid the identification of subsystem upgrades to aid the transition to a lower carbon energy economy. Industrial sites are unique in that the personnel that operate and design them are continually making modifications to expand their capacity or make major modifications to their functionality. These industrial energy systems are designed to be continuously modified and upgraded, and legacy energy systems will be able to take advantage of the advancements proposed in this dissertation to quickly integrate “greener” technologies as society transitions to a greener energy economy.

Table 17: Summary of research areas and contributions

Research areas and contributions	
Proposed, developed, implemented, and tested a risk-based obsolescence management system for automation and cyberphysical systems in industrial settings	• New theories developed for obsolescent management as a risk-based approach as compared to traditional method for "design against obsolescence" • Able to extend method to other industries • Quantitative results: • Results indicate a reduction of roughly 70% in reactive replacements due to obsolescence after the major upgrade • Results also show a 24% reduction in unplanned downtime due to part failure during normal operations. • Improvement of runtimes by 3+% per system Embeds risk-based obsolescence management into systems engineering lifecycle
Established novel simulation method for a multi-year program that utilized simulation as a tool for commissioning subsystem upgrades	• Established and applied a new metric for simulation modeling for industrial systems which baselines efficiency (Cost per IO) • Developed a new method which utilized simulation with defined interfaces to the new logic controllers to aid in the quality checking of programming and startup of new systems. • Quantitative Results: • The results show a 40% improvement in simulation cost efficiency over the span of the multi-year upgrade to ten industrial sites. • No unplanned downtime due to programming errors were achieved using this method Defines systemic way to test control systems for subsystem upgrades for constraints in systems engineering development
Expanded on high level industry standards to propose and execute a risk based and automated method to alarm audit. This proven method can be applied to other industries and professional practices	• Utilized a modified alarm objective analysis (AOA) to document safety thresholds across several industrial sites • Developed and published a reference network architecture for audit of alarms • Extended the work to emerging digital twin field to aid system engineering by constructing a virtual twin to help facilitate work activity • Quantitative results: • Sites saw an average reduction of around 52% of alarm frequency • Defined a method to extend industry standards to automated solution • Automated system deployed with substantial improvements over manual methods Theoretical development of linking digital twinning to systems engineering via "digital twin maturity model" and subsystem audit

References

U.S. General Accounting Office. "Guidelines for Model Evaluation." *PAD-79-17* (1979).

Adams, David, Christopher De Sousa, and Steven Tiesdell. "Brownfield development: A comparison of North American and British approaches." *Urban Studies* (2010): 75-104.

Aebersold, Michelle. "The history of simulation and its impact on the future." *AACN advanced critical care* 26.1 (2016): 56-61.

Ahmad, Rosmaini, and Shahrul Kamaruddin. "An overview of time-based and condition-based maintenance in industrial application." *Computers & industrial engineering* (2012): 135-149.

Ahn, Jaemyung, Olivier L. Weck, and Martin Steele. "Credibility assessment of models and simulations based on NASA's models and simulation standard using the Delphi method." *Systems Engineering* (2014): 237-248.

Akhtyamov, Rustam. et al. "Measures and approach for modemization of existing systems." *2018 IEEE International Systems Engineering Symposium (ISSE).* IEEE, 2018.

Alelyani, Turki, et al. "A literature review on obsolescence management in COTS-centric cyber physical systems." *Procedia Computer Science* (2019): 135-145.

Almada-Lobo, Francisco. "The Industry 4.0 revolution and the future of Manufacturing Execution Systems (MES)." *Journal of innovation management* (2015): 16-21.

Alpanda, Sami, and Adrian Peralta-Alva. "Oil crisis, energy-saving technological change and the stock market crash of 1973–74." *Review of Economic Dynamics* (2010): 824-842.

Alphonsus, Ephrem Ryan, and Mohammad Omar Abdullah. "A review on the applications of programmable logic controllers (PLCs)." *Renewable and Sustainable Energy Reviews* (2016): 1185-1205.

ANSI. *American National Standards Institute ANSI/ISA - 18.2 Management of Alarm Systems for Process Industries*. New York, 2009.

Areekijseree, Katchaguy, Ricky Laishram, and Sucheta Soundarajan. "Guidelines for online network crawling: A study of data collection approaches and network properties." *Proceedings of the 10th ACM Conference on Web Science*. 2018.

Badger, Lee. "Lazzarini's Lucky Approximation of π." *Mathematics Magazine* 67.2 (1994): 83-91.

Baheti, Radhakisan, and Helen Gill. "Cyber-physical systems." *The impact of control technology* (2011): 161-166.

Bartels, Bjoern, et al. *Strategies to the Prediction, Mitigation and Management of Product Obsolescence*. John Wiley & Sons, Incorporated, 2012.

Beebe, Dustin et al. "The connection of peak alarm rates to plant incidents and what you can do to minimize." *Process Safety Progress* 32.1 (2013): 72-77.

Benov, Dobriyan M. "The Manhattan Project, the first electronic computer and the Monte Carlo method." *Monte Carlo Methods and Applications* 22.1 (2016): 73-79.

Bertolini, Massimo, et al. "Development of risk-based inspection and maintenance procedures for an oil refinery." *.Journal of Loss Prevention in the Process Industries* (2009): 244-253.

Biesinger, Florian, et al. "A case study for a digital twin of body-in-white production systems general concept for automated updating of planning projects in the digital factory." *2018 IEEE 23rd international conference on emerging technologies and factory automation (ETFA).* . IEEE, 2018.

Bil, Cees, and John Mo. "Obsolescence Management of Commercial-off-the-shelf (COTS) in Defence Systems." *Concurrent Engineering Approaches for Sustainable Product Development in a Multi-Disciplinary Environment* (2013): 621-632.

Blanchard, Benjamin S., Wolter J. Fabrycky, a. *Systems engineering and analysis*. Vol. 4. Englewood Cliffs, NJ: Prentice Hall, 1990.

Boyes, Hugh, et al. "The industrial internet of things (IIoT): An analysis framework." *Computers in Industry* 101 (2018): 1-12.

Bransby, M. L., and James Jenkinson. *The management of alarm systems*. Sudbury: HSE Books, 1998.

Broas, Romulo F. Jimenez, and Mo Mansouri. "Search Optimization Applied to a Modular System Upgrade: A Preliminary Model." *16th International Conference of System of Systems Engineering (SoSE)*. IEEE, 2021.

Cesar, Eduardo L., et al. "Technological Obsolescence Management of Electrical Equipment and Automation Systems." *2019 IEEE Petroleum and Chemical Industry Committee Conference (PCIC).* Ed. IEEE. 2019.

Chu, Ng Kok, and Koji Ueda. "Alarm Rationalization for Improving Safety." Technical Report . 2011.

Dahm, Michele, and Anoop Mathur. "Automation in the food processing industry: distributed control systems." *Food Control* (1990): 32-35.

Dhole, Vikas. "Transformation of Process Engineering - Best Practices and Innovation for the MIddle East." *1st MEPEC Conference*. Manama, Bahrain, 2011.

Do Van, Phuc, et al. "Dynamic grouping maintenance with time limited opportunities." *Reliability Engineering & System Safety* (2013): 51-59.

DoD. "Modular Open Systems Architecture (MOSA)." 2020.

Eldabi, Tillal, et al. "Hybrid simulation: Historical lessons, present challenges and futures." *2016 Winter Simulation Conference (WSC)*. IEEE, 2016.

Elwerfalli, Abdelnaser, et al. "Developing turnaround maintenance (TAM) model to optimize TAM performance based on the critical static equipment (CSE) of GAS plants." *International Journal of Industrial Engineering* (2019): 12-31.

Erickson, Kelvin T. "Programmable Logic Controllers." *IEEE Potentials* 15.1 (14-17): 1996.

Feldkirchner, Michael, and Christina Lutkus. "Process Control Networking-Bridging Between Process Control and Information Technology." *IEEE IAS Pulp, Paper and Forest Industries Conference (PPFIC)*. IEEE, 2019.

Ferreira, S., et al. "KPI development and obsolescence management in industrial maintenance." *Procedia Manufacturing* (2019): 1427-1435.

Fishman, George S., and Philip J. Kiviat. "The statistics of discrete-event simulation." *Simulation* (1968): 185-195.

Fossett, Christine A, et al. "As Assessment Procedure for Simulation Models: A Case Study." *Institue for Operations Research and the Management Sciences* 39.5 (1991): 710-723. <https://doi.org/10.1287/opre.39.5.710>.

Freeman, Raymond A. "Process hazard analyses of control and instrument systems." *Process Safety Progress* (2001): 189-195.

Gee, Wesley A. "Systems Engineering requirements for legacy DoD hardware upgrade and sustainment requirements definition, analysis, and validation." *3rd Annual IEEE Systems Conference*. IEEE, 2009.

Goel, Pankaj, Aniruddha Datta, and M. Sam Mannan. "Industrial alarm systems: Challenges and opportunities." *Journal of Loss Prevention in the Process Industries* (2017): 23-36.

Grieves, Michael. "Digital Twins Today - Advancement & Opportunies." *ASME Digital Twin Conference*. 2021.

Grieves, Michael, and John Vickers. "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems." *Transdisciplinary perspectives on complex systems*. Springer, Cham, 2017. 85-113.

Haberfellner, R., Nagel, P., Becker, M., Büchel, A. and von Massow, H. *Systems engineering.* Springer International Publishing., 2019.

Harris, J. J., J. D. Broesch, and R. M. Coon. "A combined PLC and CPU approach to multiprocessor control." *Proceedings of 16th International Symposium on Fusion Engineering.* . IEEE, 1995. Vol. 2.

Hays, Robert T., and Singer, Michael J. *Simulation fidelity in training system and design: Bridging the gap between reality and training.* Springer & Business Media, 2012.

Herald, Tom, et al. "An obsolescence management framework for system baseline evolution—Perspectives through the system life cycle." *Systems Engineering* (2009): 1-20.

Hollifield, Bill R. and Eddie Habibi. *Alarm Management: A Comprehensive Guide: Practical and Proven Methods to Optimize the Performance of Alarm MangementSystems*. International Society of Automation, 2010.

Hollifield, Bill. "Understand and Cure High Alarm Rates." *Chemical Engineering* (2016): 123.3.

Horkan, Alicia M. "Alarm fatigue and patient safety." *Nephrology Nursing Journal* (2014): 83-86.

Hu, Wenkai, et al. "An application of advanced alarm management tools to an oil sand extraction plant." *International Federation of Automatic Control (IFAC)* (2015): 641-646.

ISA. *ISA, 2009, ANSI/ISA-18.2: Management of Alarm Systems for the Process Industries.* Durham, NC: International Society of Automation. Durham, n.d.

Izadi, Iman, et al. "An introduction to alarm analysis and design." *IFAC Proceedings* (2009): 645-650.

JE Hannay, K Brathen, OM Mevassik. "Hybrid Architecure Framwork for Simulations in a Service-Oriented Enviornment." *Systems Engineering* 20.3 (2017): 235-256.

Katz, Tami. "Evaluation of COTS Hardware Assemblies for use in Risk Averse, Cost Constrained Space-based Systems." *INCOSE International Symposium* (2019): Vol. 29. No. 1.

Kleines, Harald, et al. "Measurement of real-time aspects of Simatic/spl reg/PLC operation in the context of physics experiments." *IEEE Transactions on Nuclear Science* (2004): 489-494.

Kleinjnen, J.P.C. "Verification and Validation of Simulation Models." *European Journal of Operational research* 82.1 (1995): 145-162.

Koene, Johannes and Hiranmayee Vedam. "Alarm Management and Rationalization." *Thrid International Conference on Loss Prevention* (2000).

Kossiakoff, Alexander and William Sweet. *Systems Engineering principles and practice*. Wiley and Sons, 2011.

Kutscher, Vladimir, et al. "Upgrading of legacy systems to cyber-physical systems." *Proceedings of TMCE 2020*. 2020.

Labs, Wayne. "Food Engineering's 43rd Annual Plant Construction Survey" 2020." *Food Engineering* 2020. <https://www.foodengineeringmag.com/articles/98904-food-engineerings-43rd-annual-plant-construction-survey>.

Laggoune, Radouane, Alaa Chateauneuf, and Djamil Aissani. "Impact of few failure data on the opportunistic replacement policy for multi-component systems." *Reliability Engineering & System Safety* (2010): 108-119.

Langford, Gary. "Maintenance Scheduling Using Systems Engineering Integration." *INCOSE International Symposium.* . 2016. Vol. 26. No. 1.

Larsen, Peter H., Charles A. Goldman, and Andrew Satchwell. "Evolution of the US energy service company industry: Market size and project performance from 1990–2008." *Energy Policy* (2012): 802-820.

Legner, Christine, et al. "Digitalization: opportunity and challenge for the business and information systems engineering community." *Business & information systems engineering* (2017): 301-308.

Li, Hong, and Theodore J. Williams. "Interface design for the Purdue enterprise reference architecture (PERA) and methodology in e-Work." *Production Planning & Control* (2003): 704-719.

Liñán, Gustavo, et al. "ACE4k: an analog I/O 64× 64 visual microprocessor chip with 7-bit analog accuracy." *International Journal of Circuit Theory and Applications* (2002): 89-116.

Liptak, Bela G. *Instrument engineers' handbook, volume two: Process control and optimization.* CRC press, 2018.

Lwakatare, Lucy Ellen, et al. "DevOps in practice: A multiple case study of five companies." *Information and Software Technology* (2019): 217-230.

M. Watson, C. Rusnock, M. Miller, J. Colombi. "Informing System Design Using Human Performance Modeling." *System Engineering* 20.2 (2017): 173-187.

Maier, Andrew, et al. "Integrating Open Systems Architecture Models for Next Generation Cockpits: A Systems Engineering Approach." *IEW World Conference*. 2020.

Maine, Jeanine. *Get the Beep Out: A QI Proejct to Decrease Nusance Physiological Alarms in a Medical ICU*. Touro University Nevada, 2018.

Malcolm, J. Scott, and Daryl L. Harmon. "Advanced control room design." *IEEE Power and Energy Magazine* (2006): 43-48.

Marc, Zolghadri, et al. " "Analysing the impact of system obsolescence based on system architecture models." ." *International Symposium on Tools and Methods of Competitive Engineering (TMCE)*. 2020.

Markowitz, H. M. "Past, present, and some thoughts about the future." *Current Issues in Computer Simulation* (1979): 27-60.

McFeaters, Kevin A., Dick Ciammaichella, and Joe Waters. "Legacy process control system migrations." *61st IEEE Pulp and Paper Industry Conference (PPIC)*. IEEE, 2015.

McMahon, Terry. "Process Simulation and Process Control." *Chemical Engineering Progress* 109.9 (2013): 19-19.

Melchor-Hernández, César L., et al. "An analytical method to estimate the Weibull parameters for assessing the mean life of power equipment." *International Journal of Electrical Power & Energy Systems* (2015): 1081-1087.

Memuletiwon, D. T., et al. "Obsolescence Management: Adopting a Risk-Based Approach to Optimizing Equipment Availability and Asset Lifecycle Extension ." *Offshore Technology Conference*. 2017.

Metzger, Doug, and Ron Crowe. "Technology Enables New Alarm Management Approaches." *ISA Technical Conference*. Houston, TX, 2001.

Metzger, Mieczyslaw, and Grzegorz Polaków. "A study on appropriate plant diagram synthesis for user-suited HMI in operating control." *Engineering Interactive Systems*. Berlin: Springer, 2008. 246-254.

Nance, Richard E., and C. Michael Overstreet. "History of computer simulation software: An initial perspective." *Winter Simulation Conference (WSC)* (2017): 243-261.

Naylor, Thomas H., et al. "Verification of Computer Simulation Models." *Management Science* (1967): B92-B106.

Nedvěd, Miroslav, Pavel Vrba, and Marek Obitko. " Tool for visual difference display of programs in IEC 61131-3 ladder diagrams." *2015 IEEE International Conference on Industrial Technology (ICIT).* IEEE, 2015.

Netto, Richa, and Aditya Bagri. "Programmable logic controllers." *International Journal of Computer Applications* (2013).

Noda, Masaru, et al. "Event Correlation Analysis for Alarm System Rationalization." *Asia Pacific Journal of Chemical Engineering* 6.3 (2011): 497-502.

Nordhaus, William D. "The progress of computing." *Cowles Foundation for Research in Economics (Available at SSRN 285168)* (2001).

Nygaard, Kristen and Dahl, Ole-Johan. "The development of the SIMULA languages." *History of programming languages* (1978): 439-480.

Öhman, Martin, Stefan Johansson, and Karl-Erik Årzén. "Implementation aspects of the PLC standard IEC 1131-3." *Control Engineering Practice* (1998): 547-555.

Othman, Siti Fauzuna, et al. "Process Design Challenges In Producing Clean Fuels Projects in Brownfield Refinery." *Platform: A Journal of Science and Technology* (2021): 12-18.

Pace, D.K. "V&V State of the Art: Proc. Foundations '02, a Workship on Model SImulation and Validation for the 21st Century." Laurel, MD: The Society for Modeling and Simulation, 2002. CD-ROM. <http://www.dmso.mil/public/transition/vva/foundations>.

Pace, Dale K. "Modeling and Simulation Verification and Validation Challenges." *Johns Hopkins APL Technical Digest* 25.2 (2004).

Peacock, Ian, and Kevin Mahoney. "The ABCs of small hydro upgrade and automation." *Industry Application* (2011).

Pecht, Michael, et al. "Uprating of electronic parts to address obsolescence." *Microelectronics international* (2006).

R. Turner, D. Bodner, D. Kemp, Y. Rodriguez, J. Wade, P. Zhang. "Systems Engineering Simulation Experiance Design: Infrastructure, Process, and Application." *INCOSE International Symposium* 27.3 (2017): 296-308.

Rajagopal, S., J. A. Erkoyuncu, and R. Roy. "Impact of software obsolescence in defence manufacturing sectors." *Procedia CIRP 28* (2015): 197-201.

Redelinghuys, A. J. H., Anton Herman Basson, and Karel Kruger. "A six-layer architecture for the digital twin: a manufacturing case study implementation." *Journal of Intelligent Manufacturing* (2020): 1382-1402.

Rojo, Francisco Javier Romero, Rajkumar Roy, and Essam Shehab. "Obsolescence management for long-life contracts: state of the art and future trends." *The International Journal of Advanced Manufacturing Technology* (2010): 1235-1250.

Rullan, Agustin. "Programmable logic controllers versus personal computers for process control." *Computers & industrial engineering* (1997): 421-424.

Safer, Don and Phillip A. Laplante. ""The BP Oil Spill: Could Software be a Culprit?"." *IT Professional* (2010): 6-9.

Sage, Andrew P. ""Systems engineering education."." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* (2000): 164-174.

Sages, TM. "Programmable Logic Controllers." *Citrus Engineering Conference*. American Society of Mechanical Engineers, 1991. 67-81.

Sandborn, Peter. "Managing obsolescence risk." *Through-life Engineering Services* (2015): 341-357.

Sandborn, Peter, Varun Prabhakar, and Omar Ahmad. "Forecasting electronic part procurement lifetimes to enable the management of DMSMS obsolescence." *Microelectronics Reliability* 51.2 (2011): 392-399.

Sargent, R. G. "Verification and Validation of simulation models." *Journal of Simulation* 7 (2013): 12-24.

Sargent, R.G. "Verification, validation and accreditation of simulation models." *2000 Winter Simulation Conference Proceedings*. IEEE, 2000. Vol. 1.

Sargent, Robert G., and Osman Balci. "History of Verification and Validation of Simulation models." *Proceedings of the 2017 Winter Simulation Conference* (2017).

Shenhar, Aaron J., and Dov Dvir. *Shenhar, Aaron J., and Dov Dvir. Reinventing project management: the diamond approach to successful growth and innovation*. Harvard Business Review Press, 2007.

Shin, Kee-Young, and Hyun-Chul Park. "Smart manufacturing systems engineering for designing smart product-quality monitoring system in the industry 4.0."." *19th International Conference on Control, Automation and Systems (ICCAS)*. IEEE, 2019.

Shishko, Robert, and Robert Aster. *NASA systems engineering handbook*. NASA Special Publication 6105, 1995.

Shoaib, M. W., Walter Wukovits, and Saeed Gul. "Review of process simulation and simulation softwareopen source software development." *2nd Conference on Sustainability in Process Industry (SPI)*. 2014.

Shortell, Thomas M., ed. *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. John Wiley & Sons, 2015.

Smith, Chris and Robert Brelsford. "OGJ Worldwide Construction Survey." *Oil and Gas Journal* (2021).

Spitzer, David W. *Advanced regulatory control: Applications and Techniques.* Momentum Press, 2009.

Stanton, Neville. *Human Factors in Alarm Design.* . CRC Press, 1994.

Tafvizi Zavareh, Mona, et al. "A Study on the socio-technical aspects of digitization technologies for future integrated engineering work systems." *DS 91: Proceedings of NordDesign 2018*. Linköping, Sweden, 2018.

Tobias, Mark E., et al. "Booster Obsolescence and Life Extension (BOLE) for Space Launch System (SLS)." *2020 IEEE Aerospace Conference*. IEEE, 2020.

Tocher, K. D. and Owen D.G. "The automatic programming of simulations." *Proceedings of the Second International Conference on Operational Research* (1960): 50-68.

Torres, Raymund J., and Brian C. Boggan. *IEEE IAS Electrical Safety Workshop (ESW)*. IEEE, 2017.

Turner, R., et al. "Systems Engineering Simulation Experience Design: Infrastructure, Process, and Application." *INCOSE International Symposium*. 2017. 296-308.

Vasarhelyi, Miklos and et al. "The continuous audit of online systems." *Auditing: A Journal of Practice and Theory* (1991).

W. Schamai, N. Alarello, H. Philipp, L. Buffoni, P. Fritzson. "Towards the Automation of Model-Based Design Verification." *INSIGHT* 20.2 (2017): 173-187.

Wang, Ding, Xiaowei Wang, and Xiaodong Xu. "Redundant design of control station in digital safety I and C system for nuclear power plant." *Atomic Energy Science and Technology* (2013): 105-108.

Wang, Pei-qing, and Wei-wen Lu. "Configurable Design of I/O Drive System." *Journal of System Simulation* (2013).

Wang, Wenguang, et al. "Service-oriented simulation framework: An overview and unifying methodology." *Simulation* (2011): 221-252.

Wareham, R. "Ladder diagram and sequential function chart languages in programmable controllers." *Fourth Annual Canadian Conference Proceedings: Programmable Control and Automation Technology Conference and Exhibition.* IEEE, 1988.

Williams, Theodore J. "A reference model for computer integrated manufacturing from the viewpoint of industrial automation." *IFAC Proceedings Volumes* 23.8 (1990): 281-291.

—. "The Purdue enterprise reference architecture." *Computers in industry* (1994): 141-158.

Wilson, Robert L. "Smart transmitters-digital vs. analog." *Forty-First Annual Conference of Electrical Engineering Problems in the Rubber and Plastics Industries.* IEEE, 1989.

Woodruff, Jason Michael. "Consequence and likelihood in risk estimation: A matter of balance in UK health and safety risk assessment practice." *Safety Science* (2005): 345-353.

Wu, Xiling, Caihua Zhang, and Wei Du. " "An Analysis on the Crisis of "Chips shortage" in Automobile Industry——Based on the Double Influence of COVID-19 and Trade Friction." *Journal of Physics: Conference Series*. IOP Publishing, 2021.

Xu, Zhaoyi, and Joseph Homer Saleh. ""Machine learning for reliability engineering and safety applications: Review of current status and future opportunities." *Reliability Engineering & System Safety* (2021): 107530.

Zhang, Xiang, and Ernst Gockenbach. "Assessment of the actual condition of the electrical components in medium-voltage networks." *IEEE Transactions on reliability* 55.2 (2006): 361-368.

Zimmerman, Karl, and David Costello. "Lessons learned from commissioning protective relaying systems. IEEE, 2009." *2009 62nd Annual Conference for Protective Relay Engineers*. Ed. IEEE. 2009.

Zou, Caineng, et al. "Energy revolution: From a fossil energy era to a new energy era." *Natural Gas Industry* (2016): 1-11.

Zurlo, J.A. "Refining: High-impact challenges in today's global refining market." *Hyrdocarbon Processing* November 2016.