ORNL/SPR-2015/209

Taxonomy for Common-Cause Failure Vulnerability and Mitigation

Richard Wood
Kofi Korsah
James Mullens
Laura Pullum

September 2015

Approved for public release. Distribution is unlimited.

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE US DEPARTMENT OF ENERGY

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ORNL/SPR-2015/209

Reactor and Nuclear Systems Division
Electrical and Electronics Systems Research Division
Computational Sciences & Engineering Division

Taxonomy for Common-Cause Failure Vulnerability and Mitigation

Richard Wood
Kofi Korsah
James Mullens
Laura Pullum

Date Published: September 2015

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6283
managed by
UT-BATTELLE, LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

Page intentionally blank

CONTENTS

Page

Page intentionally blank

LIST OF FIGURES

Figure **Page**

Page intentionally blank

LIST OF TABLES

Table **Page**

Page intentionally blank

ACRONYMS

CFR	Code of Federal Regulations
AIChE	American Institute of Chemical Engineers
ALWR	Advanced Light-Water Reactor
ARP	Aerospace Recommended Practice
ASI	Advanced Sensors and Instrumentation
BNL	Brookhaven National Laboratory
BTP	Branch Technical Position
CCF	common-cause failure
CCPS	Center for Chemical Process Safety
CFR	Code of Federal Regulations
CMF	common-mode failure
COSS	computerized operator support system
CPU	central processing unit
D3	diversity and defense-in-depth
DAS	Database and Analysis System
DFWCS	digital feedwater control system
DISPO	DIverse Software PrOject
DOE	U.S. Department of Energy
DSD	diversity-seeking decision
EPRI	Electric Power Research Institute
FPGA	field programmable logic device
GDC	General Design Criterion
HSI	human-system interface
I&C	instrumentation and control
IAEA	International Atomic Energy Agency
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
INL	Idaho National Laboratory
ISO	International Organization for Standardization
ISTec	Institut für Sicherheitstechnologie (Institute for Safety Technology)
LER	licensee event report
NC	nonclassified
NE	Office of Nuclear Energy
NEET	Nuclear Energy Enabling Technologies
NEI	Nuclear Energy Institute
NPP	nuclear power plant
NRC	U.S. Nuclear Regulatory Commission
ORNL	Oak Ridge National Laboratory
PIE	postulated initiating event
PRA	probabilistic risk assessment
RISC	risk-informed safety class
RPS	reactor protection system
SAE	Society of Automotive Engineers
SCSS	Sequence Coding and Search System
SIS	safety instrumented system
SRM	staff requirements memorandum
SSC	structures, systems, and component
STAMP	Systems-Theoretic Accident Model and Processes

Std	standard
VME	Versa Module Europa (computer bus)

ACKNOWLEDGMENTS

The research described in this report was sponsored by the Nuclear Energy Enabling Technologies (NEET) Advanced Sensors and Instrumentation (ASI) Program of the U.S. Department of Energy (DOE) Office of Nuclear Energy (NE). The findings documented in this report leverage information from prior investigations to develop the basis for a taxonomy for common-cause failure (CCF) vulnerabilities and mitigation. In particular, the authors incorporated technical information generated through their own prior work under the sponsorship of the U.S. Nuclear Regulatory Commission (NRC) Office of Nuclear Regulatory Research.

The authors wish to thank the technical peer reviewers and editorial staff at Oak Ridge National Laboratory for their feedback and assistance in improving this report.

Page intentionally blank

ABSTRACT

Applying current guidance and practices for common-cause failure (CCF) mitigation to digital instrumentation and control (I&C) systems has proven problematic, and the regulatory environment has been unpredictable. The potential for CCF vulnerability inhibits I&C modernization, thereby challenging the long-term sustainability of existing plants. For new plants and advanced reactor concepts, concern about CCF vulnerability in highly integrated digital I&C systems imposes a design burden that results in higher costs and increased complexity. The regulatory uncertainty in determining which mitigation strategies will be acceptable (e.g., what diversity is needed and how much is sufficient) drives designers to adopt complicated, costly solutions devised for existing plants.

To address the conditions that constrain the transition to digital I&C technology by the US nuclear industry, crosscutting research is needed to resolve uncertainty, demonstrate necessary characteristics, and establish an objective basis for qualification of digital technology for nuclear power plant (NPP) I&C applications. To fulfill this research need, Oak Ridge National Laboratory is investigating mitigation of CCF vulnerability for nuclear-qualified applications. The outcome of this research is expected to contribute to a fundamentally sound, comprehensive basis to qualify digital technology for nuclear power applications.

This report documents the development of a CCF taxonomy. The basis for the CCF taxonomy was generated by determining consistent terminology and establishing a classification approach. The terminology is based on definitions from standards, guides, and relevant nuclear power industry technical reports. The classification approach is derived from identified classification schemes focused on I&C systems and key characteristics, including failure modes. The CCF taxonomy provides the basis for a systematic organization of key systems aspects relevant to analyzing the potential for CCF vulnerability and the suitability of mitigation techniques. Development of an effective CCF taxonomy will help to provide a framework for establishing the objective analysis and assessment capabilities desired to facilitate rigorous identification of fault types and triggers that are the fundamental elements of CCF.

Page intentionally blank

1. INTRODUCTION

The U.S. Department of Energy (DOE) Office of Nuclear Energy (NE) established the Advanced Sensors and Instrumentation (ASI) technology area under the Nuclear Energy Enabling Technologies (NEET) Program to coordinate instrumentation and control (I&C) technology research across DOE-NE and to identify and lead efforts to address common needs. As part of the NEET ASI research program, the Digital Technology Qualification project was established based on collaboration between Oak Ridge National Laboratory (ORNL) and Idaho National Laboratory (INL). ORNL is investigating mitigation of digital common-cause failure (CCF) vulnerability for nuclear-qualified applications, and INL is investigating the suitability of digital alternatives to analog sensors, control loops, and actuators. ORNL is responsible for integrating the technical findings and research products of this collaborative effort.

This report documents recent findings from ORNL's research activities. Specifically, the report describes the basis for a taxonomy for CCF vulnerability and mitigation through key terminology and a classification approach to support system analysis and technical evaluation.

1.1 Technical Issue

Experience in other industries shows that digital technology can provide substantial benefits in performance and reliability. However, the US nuclear power industry has been slow to adopt digital technology extensively in its I&C applications because of inhibiting factors such as regulatory uncertainty, insufficient technological experience base, implementation complexity, limited availability of nuclear-qualified products and vendors, and inadequate definition of modernization cost recapture. Obsolescence of replacement analog components and development of *de facto* standard approaches based on subjective criteria have enabled modest movement toward increasing the use of digital electronics for some command functions (e.g., control or protection algorithms/logic). However, issues such as software quality and mitigation of CCF vulnerability have led to the imposition of complex, costly design conventions and implementation practices that challenge the qualification of digital technology for high-integrity nuclear power plant (NPP) applications and constrain the benefits that can be achieved through the transition to digital.

Design criteria for safety-related I&C systems embody principles such as quality, integrity, reliability, independence, and qualification to ensure that safe conditions are maintained under all operational conditions. Separation and redundancy, physical barriers, and electrical isolation are commonly applied as design measures within a defense-in-depth concept to address potential vulnerabilities arising from single failures of equipment and propagation of failure effects. However, errors, deficiencies, or defects at any stage of a system's life cycle can result in systematic faults that may remain undetected until operational conditions activate the faulted state, resulting in failure of a critical function. The potential for CCF of multiple systems (or redundancies within a system) constitutes the principal credible threat to defeating the defense-in-depth provisions within NPP I&C system architectures. The unique characteristics and inherent complexity of digital I&C systems can exacerbate this vulnerability.

Diversity and defensive design measures are the primary means employed to address CCF vulnerability. However, the benefits of various strategic approaches for design, implementation, and architecture are not well understood. The lack of technical certainty results in the imposition of complex, costly, expedient solutions that inhibit the use of digital technology and complicate its regulatory acceptance. Consequently, the provision of adequate diversity and defense-in-depth (D3) has been identified as a high-priority technical issue for the nuclear power industry by both the Digital I&C Steering Committee of the U.S. Nuclear Regulatory Commission (NRC) and the Industry Digital I&C and Human Factors Working Group of the Nuclear Energy Institute (NEI) [1].

Applying current guidance and practices on CCF mitigation to digital I&C systems has proven problematic, and the regulatory environment has been unpredictable. In a recent license amendment in the United States involving digital modernization of plant safety systems, regulatory concerns about CCF vulnerability, which was identified through a D3 analysis, proved difficult to resolve. As a consequence, licensing was delayed considerably, resulting in a substantial time and cost impact. Ultimately, it was determined that additional diverse systems must be implemented to mitigate potential CCF vulnerability, resulting in even greater complexity and cost. The potential for CCF vulnerability inhibits I&C modernization, thereby challenging the long-term sustainability of existing plants. For new plants and advanced reactor concepts, concern about CCF vulnerability for highly integrated digital I&C systems imposes a design burden that results in higher costs and increased complexity. International regulators in Finland, the United Kingdom, and France have expressed concern about the treatment of CCF vulnerability in highly digital I&C architecture for advanced light-water reactor designs. Specifically, the regulatory review of the safety systems for the new third unit under construction at the Olkiluoto NPP in Finland has been complicated due to concerns about potential susceptibility of the I&C architecture to CCF. The regulatory uncertainty in determining which mitigation strategies will be acceptable (e.g., what diversity is needed and how much is sufficient) drives designers to adopt complicated, costly solutions devised for existing plants. Consequently, unnecessarily complex I&C architectures are imposed that are clearly not optimal solutions and may be inappropriate for advanced reactor designs. Thus, mitigation of CCF vulnerability is an issue of concern for existing plants, new plants, and advanced reactor concepts.

1.2 Problem Statement and Research Approach

The high-integrity design approaches and systematic quality assurance processes employed in the nuclear power industry provide many techniques to reduce the likelihood of residual faults and to mitigate those failure vulnerabilities that may exist. The nuclear power industry applies rigorous quality process control to avoid faults, errors, and deficiencies. However, the potential for latent faults persists. Thus, diversity and defensive design measures are employed to mitigate residual CCF vulnerability. The concern, as indicated above, is that great uncertainty remains as to the efficacy of the mitigation strategies employed and the value each provides. The central question is, "If diversity is required in a safety system to mitigate the consequences of potential CCFs, how much diversity is enough?" Further research is needed to develop comprehensive mitigation strategies to effectively address CCF vulnerability without introducing unnecessary complexity and significant cost while also providing a sound scientific basis for establishing an acceptable safety justification.

Because of the complexity of digital I&C system technology and the necessary reliance on process-driven approaches to software development and quality assurance, definitive quantitative measures for key digital I&C system characteristics are as yet unavailable. As a result, it has not been feasible to develop a comprehensive measure of diversity (particularly for software-based systems) that could be used to establish wholly objective acceptance criteria to support review of CCF mitigation.

The foremost deficiency in knowledge relates to a fundamental understanding of the nature of CCF vulnerability in the context of the nuclear power application domain. In particular, the sources of systematic faults and the triggers that impact safety-related functions in an NPP must be comprehensively identified. These fault-trigger combinations should be mapped to functions and architectural elements (e.g., I&C system blocks) and then related to hazards that could compromise plant safety. In addition, the various diversities and design measures that can mitigate CCF need to be related to the particular kinds of faults, triggers or fault-trigger combinations and to the corresponding failures that can result. This would allow for a better understanding of the impact of each diversity, the value of other defensive design measures, and the synergistic effect of combined mitigation techniques.

To resolve this knowledge gap, a more thorough definition of CCF vulnerability and mitigation techniques must be developed. Various application domains have different characterizations of diversity.

In addition, there are no readily available models and metrics to develop systematic methods, quantifiable measures, and objective criteria for evaluating CCF mitigation approaches.

This research began with an analysis of diversity approaches and experience from the international nuclear power industry, as well as other industries and organizations, capturing expert knowledge and lessons learned, determining best practices, and evaluating the remaining knowledge gaps [2]. The primary goal of the current stage of research is to establish a taxonomy to provide the basis for characterizing and analyzing CCF vulnerability and for assessing the suitability for various mitigation strategies. An effective CCF taxonomy can facilitate a rigorous identification of fault types and triggers to enable a thorough, systematic evaluation of CCF vulnerability, and it can also allow for comprehensive determination of effective mitigation techniques with the goal of substantially reducing the potential for CCF in NPP I&C systems. Without a science-based quantitative assessment capability, the nuclear power industry is limited to the more-subjective assessments and best-practice remediation currently used to provide reasonable assurance that adequate CCF mitigation is provided.

To define the research approach for this stage, it is necessary to understand the product being developed. A *taxonomy* is a scheme of classification that partitions a body of knowledge into taxonomic units and defines the relationships among these units. It encompasses the description, identification, nomenclature, and classification of a subject. It also provides a standardized set of terminologies. Development of a CCF taxonomy will help provide a framework for establishing the objective analysis and assessment capabilities desired.

To develop a CCF taxonomy, various sources were investigated to identify related taxonomies [3–8], capture prevailing terminology, and determine an effective classification scheme. Several taxonomies for failure modes and system dependability were identified. However, most were focused on characterizing contributions to reliability and impacts of risk. Specifically, many involved schemes to capture digital I&C system reliability and failure characteristics for incorporation into probabilistic risk assessments (PRAs). The objective of this CCF taxonomy is to contribute to the basis for a design analysis approach directly focused on safety as opposed to reliability. Consequently, this work involved development of a consistent terminology relevant to CCF and identification of a suitable classification approach.

1.3 Report Organization

The report is divided into four major sections that address CCF vulnerability terminology, CCF mitigation terminology, classification approaches, and CCF taxonomy basis. Fundamental terminology regarding errors, faults, failures and CCF is presented in Chapter 2. Chapter 3 describes terminology associated with fault management and prevailing CCF mitigation strategies. Chapter 4 summarizes and investigation into classification approaches suitable for organizing system and failure information in terms of key features and characteristics. Chapter 5 documents the basis for a taxonomy for CCF vulnerability and mitigation that is suitable to facilitate system analysis and evaluation to help ensure that coping strategies are adequate to ensure safety.

Page intentionally blank

2. TERMINOLOGY FOR CCF VULNERABILITY

Vulnerability is the quality of being open to attack or damage. In terms more specific to CCF, a vulnerability involves the presence of an internal fault that enables an external condition or event to harm a system (e.g., cause a failure). This section identifies terminology relevant to characterizing the CCF vulnerability of an I&C system. The fundamental terminology for system or functional failure addresses potential sources of a vulnerability, the causal effect that is the vulnerability itself, and the event that arises when the vulnerability is activated. Once the general terms are defined, the relationship between the vulnerability and the failure is described. Finally, the nature of CCF is discussed.

2.1 Defects, Errors, and Mistakes

In the general vernacular, defects, errors and mistakes are frequent causes of failure. Defects relate to flaws, imperfections, and inadequacies. Errors involve something that is unintentionally wrong in terms of implementation, execution, or result. Mistakes arise through misunderstandings, erroneous decisions, or incorrect actions.

However, these terms have specialized meanings in the systems engineering domain. The Institute of Electrical and Electronics Engineers (IEEE) refers to a *defect* in software as a “product anomaly” [9]. Examples include omissions and imperfections during the software life cycle phases. The International Electrotechnical Commission (IEC) defines a *mistake* or *human error* as an “action that produces an unintended result” [10].

The IEC defines an *error* as a “discrepancy between a computed, observed or measured value or condition and the true, specified or theoretically correct value or condition” [10]. The International Atomic Energy Agency (IAEA) extends this definition by adding that the discrepancy may be “due to a nonconformity or interference” [11]. The IEEE [9] provides a similar definition for *error* as the IEC, citing “difference” rather than “discrepancy.” However, IEEE also defines an *error* as an “incorrect step, process or data definition,” an “incorrect result” and a “human action that produces an incorrect result.” It is noted that the four cited definitions can be distinguished by attributing their meanings to be more closely representative of error, fault, failure, and mistake, respectively. Finally, in IAEA technical documents on NPP I&C systems [12, 13], *error* is defined as a “human action or process that produces an unintended result.” Thus, in the nuclear power application domain, IAEA equates errors to mistakes or deficiencies while extending the coverage of error sources to include processes as well as actions.

Based on the above definitions, defects and mistakes clearly correspond to sources of CCF vulnerability. In normal usage for system dependability, error more closely corresponds to deviations of an external or observed state of the system from its correct performance. Thus, in this sense, error equates to the consequence of an activated defect. When the deviation from correct performance exceeds a threshold or range, the error is characterized as a failure. However, the system design context in the above definitions of *error* includes consideration of erroneous action (e.g., programming) and process (e.g., software design life cycle stages). In this sense, error does relate to sources of CCF vulnerability. Consequently, clarification of the terminology is necessary to avoid confusion in usage.

The IEC defines *failure cause* as “set of circumstances that leads to failure” [10]. Furthermore, it is noted that a “failure cause may originate during specification, design, manufacture, installation, operation or maintenance of an item.” Specifically, the cause or source of a failure vulnerability can be attributed to a situation or occurrence during design, manufacture or use. As shown below, the outcome of the causal circumstance is a faulted state for an I&C system. Generally, any fault that is not observed and corrected in the design process remains latent or undetected until the system is in service and specific conditions arise. For the purposes of this taxonomy, defects, mistakes, or errors (in the broader sense identified above) are considered to be sources of CCF vulnerability. In the definitions to follow, the term *error* is frequently used to refer to a source of failure vulnerability.

2.2 Faults

Faults are the immediate consequence or manifestation of deficiencies (e.g., defects, errors, mistakes) in the specification, design, manufacture, implementation, installation, operation or maintenance of an I&C system. When activated by specific conditions, a fault can lead to a system failure during operation. Essentially, faults are the states of an I&C system that characterize CCF vulnerability.

In the vocabulary for system dependability [10], the IEC defines a *fault* as "the state of an item characterized by inability to perform a required function, excluding the inability during preventive maintenance or other planned actions, or due to lack of external resources." However, in standards addressing safety-related I&C systems, IEC defines a *fault* as an "abnormal condition that may cause a reduction in, or loss of, the capability of a functional unit to perform a required function" [14] and as a "defect in a hardware, software or system component" [15].

In IEC standards addressing systems and software engineering [16], a *fault* is defined as an "incorrect step, process, or data definition in a computer program." This is also referred to as a software development or implementation error. As noted above, one of the general IEEE definitions of *error* is similar and can be used to characterize a fault. In fact, the common glossary of software engineering terms for IEC, IEEE and the International Organization for Standardization (ISO) [17] gives this definition of *fault*. An alternate definition is also given in which a *fault* is described as a "manifestation of an error in software." It is noted that a "fault, if encountered, may cause a failure." The term *bug* is identified as a synonym for *fault* in the context of software.

In IEC 61513, "Nuclear power plants—Instrumentation and control systems important to safety—General requirements for systems" [15], it is noted that faults "may be subdivided into random faults, that result e.g. from hardware degradation due to ageing, and systematic faults, e.g. software faults, which result from design errors." Furthermore, IAEA concurs with this characterization by stating that faults in a system "may be the result of design errors of hardware or software or may be caused by aging or wear or by environmental stress on the hardware of a system" [13]. For NPP I&C systems [15], *random faults* are defined as "non-systematic fault of hardware components" where the fault is "a consequence of physical or chemical effects, which may occur at any time." Conversely, a *systematic fault* for NPP I&C systems is defined as a fault in "the hardware or software which concerns systematically some or all components of a specific type." It is noted that a systematic fault "may result from errors in the specification or design, from manufacturing defects or from errors which are introduced during maintenance activities." *Software faults* are considered to be systematic faults and are defined as a "design fault located in a software component."

Since high-integrity system/software design life cycles include verification and validation measures to detect and correct faults, it is presumed that remaining faults in installed safety-related I&C systems are dormant or latent. IEC 61513 defines latent faults as "undetected faults in an I&C system." A note to the *fault* definition in the standard states that a "fault (notably a design fault) may remain undetected in a system until specific conditions are such that the result produced does not conform to the intended function, i.e. a failure occurs." In the definition of *systematic fault*, it is also noted that components "containing a systematic latent fault may fail randomly or coincidentally, depending on the kind of fault and the mechanisms that trigger the fault." Finally, the standard also notes that latent faults implemented in redundant subsystems could be triggered by specific conditions to cause a CCF.

2.3 Failures

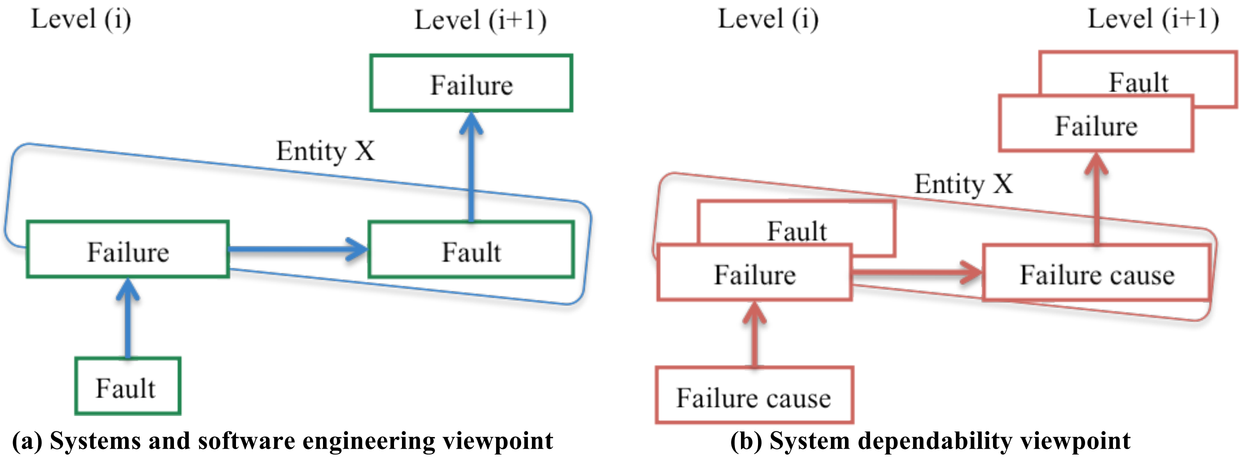
The IAEA Safety Glossary [18] defines *failure* as the “inability of a structure, system or component to function within acceptance criteria.” It further notes that an item fails “when it becomes incapable of functioning, whether or not it is needed at that time.” The IEC standards for NPP I&C systems adopt the IAEA definition. The IEC vocabulary for system dependability [10] and the ISO/IEC/IEEE vocabulary for systems and software engineering [17] define *failure* as the “termination of the ability of a product to perform a required function or its inability to perform within previously specified limits.” It is noted that a “failure may be produced when a fault is encountered.” An alternate definition given in the ISO/IEC/IEEE vocabulary describes a *failure* as an “event in which a system or system component does not perform a required function within specified limits.” In the nuclear power context, a failure results when “either an intended function is not performed on demand or an unintended function is initiated,” as noted in IAEA-TECDOC-952, “Advanced Control Systems to Improve Nuclear Power Plant Reliability and Efficiency” [13].

A *systematic failure* is defined by IEC [14, 17] as “failure related in a deterministic way to a certain cause, which can only be eliminated by a modification of the design or of the manufacturing process, operational procedures, documentation or other relevant factors.” A software failure is a type of systematic failure. The IEC vocabulary for system dependability defines a *software failure* as a “manifestation of a dormant software fault.” However, the IEC standards for NPP I&C systems [15] define a *software failure* as a system failure “due to activation of a design fault in a software component.” It is noted that software failures are “due to design faults, since software does not wear out or suffer from physical failure.” It is further observed that, since the conditions or triggers that can “activate software faults are encountered at random during system operation, software failures also occur randomly.”

Finally, it is important to distinguish between failure mechanisms, failure modes, and failure effects. The IEC [10] and IEEE [9] define a *failure mechanism* as “the physical, chemical or other process which has led to a failure.” Basically, it captures the relationship of a failure to its causes. The ISO/IEC/IEEE systems and software engineering vocabulary defines a *failure mode* as the “physical or functional manifestation of a failure.” The IEEE [9] defines a *failure mode* as the “effect by which a failure is observed to occur” or the “manner in which failure occurs.” Essentially, failure modes are the observable modes in which a system (or its components, software, and processes) can fail. Finally, IEC [10] defines *failure effect* as the “consequence of a failure, within or beyond the boundary of the failed item.” The consequences of a failure (i.e., its effect) can be observed in terms of the operation, function or status of the system. Failure effects indicate the significance of a failure.

2.4 Fault–Failure Relationship

The definitions above indicate two distinct fault-failure relationships. First, the systems and software engineering viewpoint establishes a causal relationship in which a failure results when a fault is activated or triggered. Second, the system dependability viewpoint identifies a failure as an event with a fault being the resulting state of the system. These relationships are illustrated in Fig. 2.1, which is derived from IEC 61508-4, “Functional safety of electrical/electronic/ programmable electronic safety-related systems – Part 4: Definitions and abbreviations” [14]. In each illustration, “entity” corresponds to the failure condition (or failed state) for items of interest within an I&C system (e.g., component, software object, module, subsystem, system). These items of interest can be described as functional units that are interrelated through a nested hierarchical construct of multiple levels (i.e., the higher-level functional units are composed of lower-level functional units). For example, a functional unit may be a sub-element of a complex system or may be an element of an overall architecture of interconnected systems. Thus, the illustrations show two levels (i and i+1) in which an effect in the lower-level functional unit can cause an effect in the higher-level functional unit. In Fig. 2.1(a), the relationship is shown such that a fault results in a failure in the lower-level functional unit, which then becomes a fault in the higher-level functional unit. In effect, the Entity X (i.e., condition or state of failure) results as faults are activated and become failures. In Fig. 2.1(b), the relationship is shown such that a failure cause leads to a failure, which then results in a fault. An event (failure) results in a state (fault) in the lower-level functional unit that in turn serves as the failure cause for an event in the higher-level functional unit. As before, the consequence of these interrelated effects is the failure condition, Entity X.



(a) Systems and software engineering viewpoint **(b) System dependability viewpoint**

Fig. 2.1. Fault-Failure relationship.

The first viewpoint, as shown in Fig. 2.1(a), corresponds to the treatment of functional safety in standards specific to I&C systems within the nuclear power application domain (i.e., IEC 61513 and its daughter standards). Consequently, the concept of a fault as a precursor to failure is appropriate for characterizing CCF vulnerability in I&C systems at NPPs. In fact, this viewpoint corresponds directly to the expressed conditions for CCF in IEC 62340, "Instrumentation and Control Systems Important to Safety—Requirements to Cope with Common Cause Failure (CCF)" [19].

As discussed above, encountering a fault may result in a failure, depending on the presence of a triggering event or condition. Several of the standards and technical reports referenced herein promote the understanding that events or conditions are what trigger or activate a fault. Figure 2.2 illustrates the process by which a fault is activated by a trigger to become a failure. Specifically, the activation of a fault can be seen as an event or phenomenon under which a fault becomes a failure. Thus, evaluating failure vulnerabilities must consider fault-trigger combinations. Essentially, the failure vulnerability (i.e., fault) cannot be properly assessed without consideration of the conditions that activate the vulnerability to become a failure.

IAEA Nuclear Energy Series Report No. NP-T-1.5, *Protecting Against Common Cause Failures in Digital I&C Systems of Nuclear Power Plants*, states that a "failure is the result of the activation of a fault by a triggering event." The IAEA report defines a triggering mechanism as a "[s]pecific event or operating condition that causes structures, systems or components to fail due to a latent fault." Furthermore, the report notes that triggering mechanisms can "cause failure of two or more separate . . . components in a time-correlated way." The ISO/IEC/IEEE vocabulary for systems and software engineering defines a triggering event as an "event that occurs outside the boundary of the measured software and initiates one or more functional processes." However, IEC 62340 identifies triggering events as including "natural phenomenon, plant process operation or an action caused by man or by any internal event in the I&C system." In addition, the IEC vocabulary for system dependability defines a trigger as a "combination of operating circumstances that activates a dormant fault." It is noted that the trigger "may be the internal state of the system, the data being processed, operator action, environmental conditions, or any combinations thereof."

Triggering events or activation conditions can be plant transients and initiating events, external conditions, interactions among systems, human interaction, internal states (e.g., execution profile, exception handling), and so forth. IEC 61513 notes in its definition of failure that a "failure is the result of a hardware fault, software fault, system fault, or human error, and the associated signal trajectory which triggers the failure." Thus, signal trajectory is considered a primary activation condition. Specifically, the IEC [19, 20] defines signal trajectory as the "[t]ime histories of all equipment conditions, internal states, input signals and operator inputs which determine the outputs of a system." Thus, an understanding of fault triggers requires knowledge of the operation of the system as well as its context of use (e.g., plant state and external conditions).

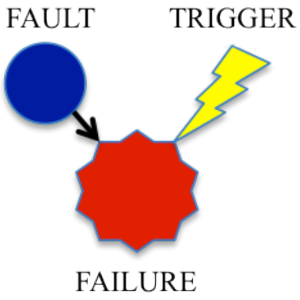


Fig. 2.2. Failure resulting from activation of a fault by a triggering event.

2.5 Common-Cause Failure

The IEC vocabulary for system dependability defines *CCF* as "failures of multiple items, which would otherwise be considered independent of one another, resulting from a single cause." It also defines *common-mode failure* (CMF) as "failures of different items characterized by the same failure mode." CCF and CMF are dependent failures. In effect, they are failure events in which their probability cannot be expressed as a simple product of the unconditional failure probabilities of the individual events. According to definitions established in the late 1980s by the United Kingdom Atomic Energy Authority [21], *CCF* is a "specific type of dependent failure that arises in redundant components where simultaneous (or near simultaneous) multiple failures result in different channels from a single shared cause." *CMF* is defined as a "subset of common-cause failures in which multiple items fail in the same mode."

IEEE [9] defines *CCF* as "[t]wo or more redundant component failures due to a single cause" or more simply as "multiple failures attributable to a common cause." It notes that "common-cause events that cause multiple failures are usually . . . events that exceed the design envelope of the component." For I&C systems, IEC 61508-4 states that *CCF* is "failure, which is the result of one or more events, causing coincident failures of two or more separate channels in a multiple channel system, leading to system failure"

CCF is defined by the IAEA as a "failure of two or more structures, systems or components due to a single specific event or cause" [22]. IAEA adds examples of causes such as "a design deficiency, a manufacturing deficiency, operation and maintenance errors, a natural phenomenon, a human induced event, saturation of signals, or an unintended cascading effect from any other operation or failure within the plant or from a change in ambient conditions" [18]. IAEA also defines *CMF* as a subset of CCF in which systems fail "in the same manner or mode due to a single event or cause." IEC 62340 further adds to the *CCF* definition by noting that the "coincidental failure of two or more structures, systems or components is caused by any latent deficiency from design or manufacturing, from operation or maintenance errors, and which is triggered by any event induced by natural phenomenon, plant process operation, or action caused by man or by any internal event in the I&C system."

The NRC defines *CMF* as "causally related failures of redundant or separate equipment" [23, 24]. It further notes that CMF involve "all causal relations, including severe environments, design errors, calibration and maintenance errors, and consequential failures." In recent years, NRC has adapted its terminology to address CCF as the wider set of dependent failures. For example, Branch Technical Position 7-19, "Guidance for Evaluation of Diversity and Defense-in-Depth in Digital Computer-Based Instrumentation and Control Systems" [25], refers to CCF rather than CMF. A more current NRC definition for *CCF* is as "a dependent failure in which two or more component fault states exist simultaneously or within a short period of time, and are a direct result of a shared cause" [26].

IEC 62340 notes that CCF is a subtype of systematic failure in which "failures of separate systems, redundancies or components can be triggered coincidentally." Coincidental failure is interpreted to cover "a sequence of system or component failures when the time interval between the failures is too short to set up repair measures." The IAEA Nuclear Energy Series Report No. NP-T-1.5 also states that failures are "concurrent when the time interval between the failures is too short for repair measures." The basis for a CCF occurrence, as described in the standard and IAEA report, corresponds to the systematic incorporation of a latent fault in multiple systems or redundancies followed by the triggering of that common fault to cause a coincidental failure of some or all of the systems or redundancies. Figure 2.3 illustrates the concept for a CCF affecting multiple systems.

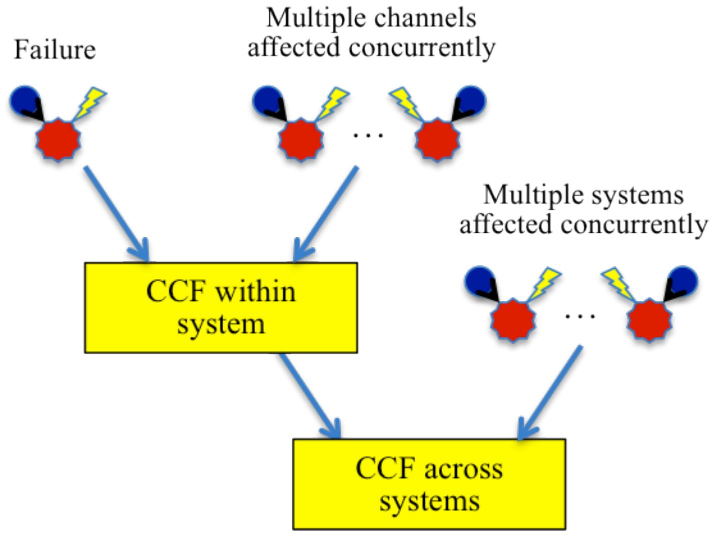

Failure
Multiple channels affected concurrently
. . .
Multiple systems affected concurrently
. . .
CCF within system
CCF across systems

Fig. 2.3. Basis for CCF of multiple systems.

Latent faults can originate at any phase of the digital I&C system life cycle, are typically human induced or technology related, and involve design flaws, performance limitations, or implementation complexity. At a high level, three prominent sources of latent systematic faults are (1) errors in the requirement specification, (2) inadequate provisions to account for design limits (e.g., environmental stress), and (3) technical faults incorporated in the internal system (or architectural) design or implementation.

Quality processes detect and correct many implementation errors. However, as design complexity increases, the feasibility of exhaustive testing or comprehensive formal proof diminishes considerably. Therefore, some residual faults may remain undetected and persist as latent faults within the system. Design errors arising from flawed, incomplete, ambiguous, or misinterpreted requirements are systematic in nature and are resulting faults significantly more difficult to detect and correct as the system life-cycle phases progress. These faults do not, in and of themselves, necessarily constitute a hazard unless conditions (e.g., operational, environmental, relational, or temporal) activate the faulted state and result in failure of a critical function.

Triggers that can activate faults and result in failure arise primarily from signal trajectory, human actions, external events, and temporal effects. The signal trajectory for a digital I&C system involves not only current input values but also past input values, the internal state of the system, and the sequence of transitions among internal states. Failures arising from latent faults activated by triggering mechanisms associated with signal trajectories clearly correspond to conditions that either were not anticipated or properly addressed during system development and that were not exposed through testing.

Human actions that can induce a CCF include maintenance errors, input mistakes, out-of-sequence commands, and ill-timed or conflicting actions. External events that can pose common cause triggers include transient effects such as anomalies or failures propagating from other systems or components within the I&C system architecture, and environmental stress such as seismic, vibratory, electromagnetic and electrical surge. Temporal effects that can initiate failures include dependence on calendar-date or time-of-day information, synchronization with a common clock, synchronization of processes or systems, and runtime effects dependent on execution cycle histories (e.g., runtime overflows of buffers or stacks).

Page intentionally blank

3. TERMINOLOGY FOR CCF MITIGATION

Mitigation is an approach to make an outcome less severe or intense. It consists of measures taken to moderate or alleviate the effect of an event or to reduce the probability of the occurrence of the event. For both system dependability and software engineering, many of the mitigating design, analysis, and implementation practices are generally label as fault tolerance. Specifically, *fault tolerance* is the "ability of a functional unit to continue to perform a required function in the presence of faults" [14]. Mitigation of CCF implies fault tolerance at a safety function level rather than at a particular system level. Therefore, to properly characterize CCF mitigation within this taxonomy, it is necessary to consider the full range of fault management approaches and understand the context for the prominent techniques employed in the nuclear power industry. With the context established, the relevant terminology for CCF mitigation can be described. In particular, the various forms of diversity must be considered.

3.1 Fault Management Approaches

There are many techniques for managing faults, especially for digital I&C systems, that have been employed for high-integrity functions within various application domains. A hierarchy of these techniques is shown in Fig. 3.1. They are generally grouped in terms of design evaluation (including fault forecasting) and fault removal, fault tolerance (i.e., detection/masking and recovery), and fault avoidance and mitigation. The techniques indicated involve design approaches, life-cycle actions, technology choices, architectural configurations, and so forth. *Fault prevention* is another means of fault management and is often equated to quality assurance practices for high-integrity applications. This technique will not be covered in this report, as it constitutes a general engineering practice.

Design evaluation and *fault removal* involve detailed analyses to identify (or predict) and eliminate threats to the extent practical. They generally apply to high-quality processes employed to minimize the potential for faults and remove vulnerabilities as they are discovered. These techniques promote fault avoidance at a high level and are primarily oriented toward design approaches and evaluation processes.

Fault tolerance in this hierarchy represents specific techniques for accommodating the presence of faults and avoiding consequent failure. Failsafe designs are enabled by these techniques. *Detection* and *masking* relate to identifying the presence of a fault or masking its potential effect (i.e., avoiding failure due to the fault). Diagnostics (e.g., fault identification and isolation) and voted redundancies are common techniques. *Recovery* relates to the response to an activated fault (i.e., failure) and enables continued execution with recapture of the pre-failure state.

Fault avoidance and *fault mitigation* include design strategies to impede the propagation of the effects of faults (i.e., failures). *Separation*, *independence*, and *fault containment* are techniques for constraining the potential effects of activated faults, while *dissimilarity/diversity* and *checked redundancy* are means for mitigating the effect of activated faults by either precluding common faults (in the first case) or detecting and compensating for activated faults (in the second case).

The fault management techniques described above generally relate to the faults themselves, and to some degree, they relate to the triggers that activate the faults to cause failures. These fault management techniques embody supporting technical and life-cycle methods and approaches on which strategies to cope with CCF vulnerability can be based.

At the outset of I&C system architecture development, design principles are invoked to minimize the use of common elements and to limit failure propagation paths. These design considerations are effective in reducing the potential for CCF vulnerability, but their absolute, across-the-board use can result in extremely complicated, inefficient, and potentially unreliable I&C system architectures. As a result, two principal coping strategies are typically employed in responding to CCF susceptibility: (1) CCF avoidance and (2) CCF mitigation.

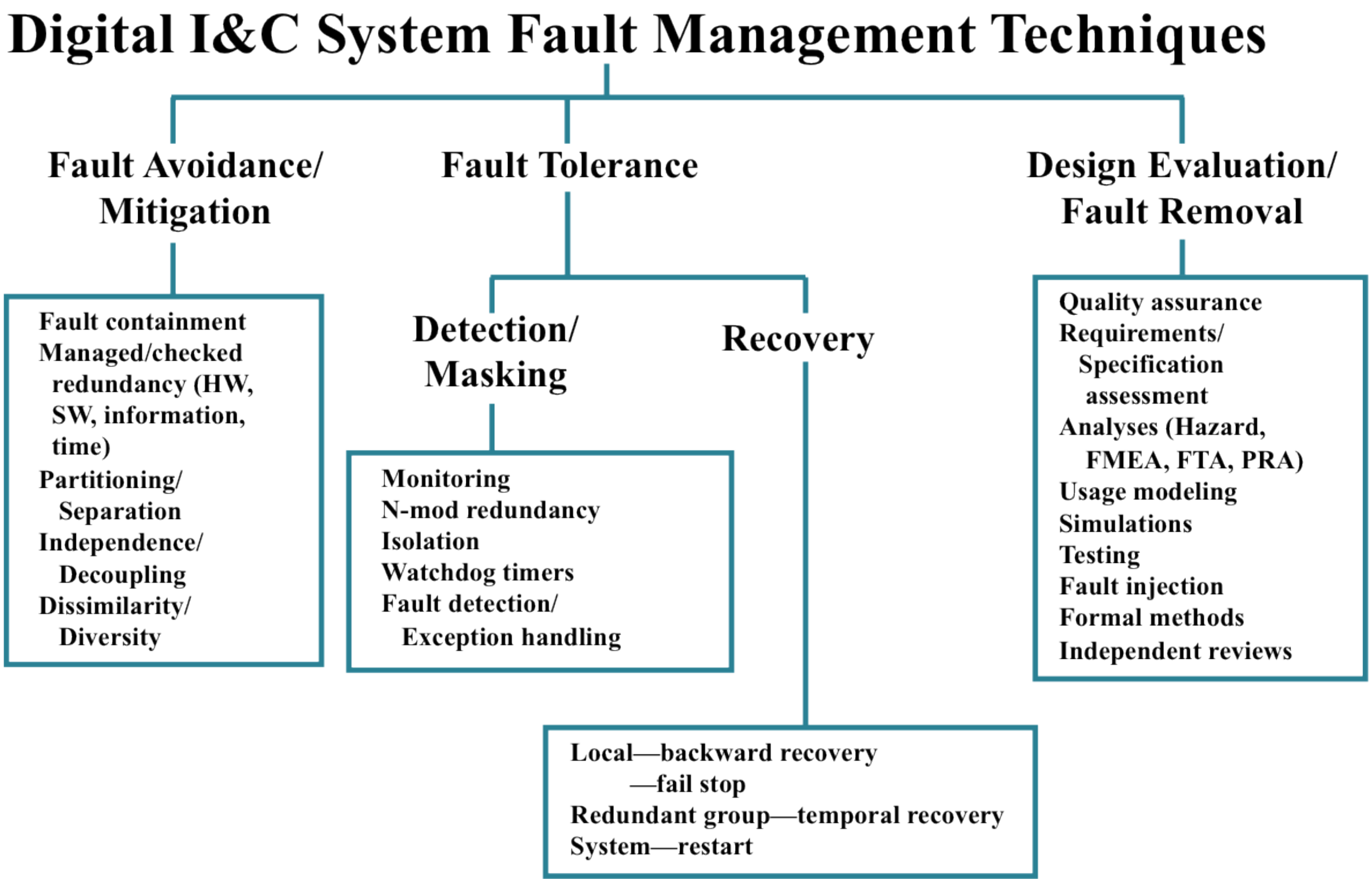


Fig. 3.1. Fault management techniques for digital I&C systems.

The objective of the first strategy, CCF avoidance, is to avoid fault introduction and eliminate conditions for potential common triggers to the degree feasible. Systematic life-cycle processes with comprehensive hazard identification and extensive verification and validation activities are employed to yield high-quality systems with the goal of approaching error-free software (e.g., fault prevention). Nevertheless, experience confirms that undetected faults can progress through even the most rigorous design process. As an additional aspect of the avoidance approach, design measures can be used to reduce the exposure to anticipated triggers or their concurrent application to multiple systems that may have common faults. Application of such design measures depends upon a well-founded understanding of the types of fault-trigger combinations that may be present and the design conventions that are most effective in preventing concurrent triggering of any common faults that may be present. Examples of these design measures are invariant cyclic execution of code, self-monitoring and self-testing, signal validation and command checking, and physical separation by barriers into different environmental control zones [11, 27]. However, since there is no assurance that unanticipated common triggers do not exist, use of these measures cannot guarantee sufficient CCF robustness. Thus, the primary goal of this strategy is to minimize the occurrence of common faults and reduce the likelihood of triggered failures.

The objective of the second strategy, CCF mitigation, is to mitigate any vulnerability to CCF. The first step in pursing this goal involves architectural provisions for the I&C systems at an NPP to defend against failure. Defense-in-depth is employed to compensate for failures in other systems or functions. The IAEA defines *defense-in-depth* as “the application of more than one protective measure for a given safety objective, such that the objective is achieved even if one of the protective measures fails” [18]. In practice, several independent systems are implemented to serve as successive barriers to prevent unsafe consequences from occurring. This aspect of the mitigation approach is especially effective against single failures. However, CCF can potentially disable multiple barriers, resulting in unsafe conditions. Thus, in the nuclear power industry, diversity is employed to provide alternate equivalent functionality or systems that are not susceptible to the same CCF as their counterpart(s) within the I&C system architecture. The difficulty occurs in identifying the full range of fault-trigger combinations that may be present and then selecting the appropriate compensating diversities. Consequently, the primary realistic objective of this approach is to mitigate the vulnerability to CCF by providing alternate or backup functions that are unaffected.

3.2 Diversity

In the ISO/IEC/IEEE vocabulary for systems and software engineering, *diversity* is defined as the “realization of the same function by different means.” Examples of diversity include “use of different processors, storage media, programming languages, algorithms, or development teams.” In IEC 61508-4, *diversity* is defined as “different means of performing a required function.” The standard identifies different physical methods or different design approaches as means of achieving diversity.

In the context of nuclear power, the IAEA [18] defines *diversity* as the “presence of two or more redundant systems or components to perform an identified function, where the different systems or components have different attributes so as to reduce the possibility of common cause failure, including common mode failure.” Examples of the different attributes include “different operating conditions, different working principles or different design teams (which provide functional diversity), and different sizes of equipment, different manufacturers, and types of equipment that use different physical methods (which provide physical diversity).” For software-based I&C systems, IEC 60880, “Nuclear Power Plants—Instrumentation and Control Systems Important to Safety—Software Aspects for Computer-Based Systems Performing Category A Functions” [20], defines *diversity* as “existence of two or more different ways or means of achieving a specified objective.” Diversity is specified as a defense against CCF, with physical or functional differences cited as typical means of achieving it.

As shown in the above definitions, there several forms of diversity, which include physical, functional, and design. Various diversity types have been defined in the nuclear power industry and by other organizations. These diversities must be considered in establishing CCF mitigation terminology for this taxonomy.

3.2.1 Nuclear Power Industry Guidance

General Design Criterion (GDC) 22, "Protection system independence," in Appendix A of Title 10, Part 50 of the Code of Federal Regulations (CFR) [28], requires that "functional diversity or diversity in component design and principles of operation . . . be used to the extent practical to prevent loss of the protection function." NUREG-0493, *A Defense-in-Depth and Diversity Assessment of the RESAR-414 Integrated Protection System* [29], provides some of the earliest guidance on applying diversity to mitigate CCF (or CMF, as it known in the 1970s). The report defines *diversity* as the "design approach for achieving a reduced probability of functional failure as a result of postulated CMF, by providing different equipment as redundant backup." The report identifies four forms of diversity: signal, equipment (i.e., physical), aspect and people. *Signal diversity* involves the use of different signals to initiate a protective action. *Equipment diversity* arises from the use of different equipment to perform safety functions. In the report, it is stated that "different" means the equipment is "sufficiently unlike as to decrease significantly the vulnerability" to CCF. Three examples are given: "relay vs. solid-state logic, transistor vs. magnetic amplifiers, and electrical vs. pneumatic signal transmission." *Aspect diversity* is the use of different logic levels. The cited example is the "use of relays that pick-up to scram vs. drop-out to scram." Finally, *people diversity* involves the use of different groups of people to design or maintain different equipment.

Guidance on addressing CCF vulnerability was formalized in the staff requirements memorandum (SRM), dated July 21, 1993 [30], on SECY 93-087, "Policy, Technical, and Licensing Issues Pertaining to Evolutionary and Advanced Light-Water Reactor (ALWR) Designs" [31]. In that guidance, NRC documented a four-point position on D3 that was incorporated into Chapter 7, "Instrumentation and Controls," of NUREG-0800, *Standard Review Plan for Review of Safety Analysis Reports for Nuclear Power Plants* [32], as Branch Technical Position (BTP) 7-19, "Guidance on Evaluation of Defense-in-Depth and Diversity in Digital Computer-Based Instrumentation and Control Systems" [33].

NUREG/CR-6303, *Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems* [34], provides guidance on performing a D3 assessment to determine the CCF vulnerability of an NPP I&C system architecture. Where the D3 assessment determines that CCF vulnerability of one or more safety functions exists, additional diversity is required as mitigation. The application of that diversity can be achieved by providing a separate automatic system to back up the affected safety function(s) or through the introduction of intentional diversity and compensating design measures at the appropriate lower level(s) of the I&C system architecture (e.g., system, divisional redundancies, subsystems, modules, or components).

NUREG/CR-6303 separated the forms of diversity into the following six attributes to facilitate assessments of adequate diversity in safety systems:

- design diversity,
- equipment diversity,
- functional diversity,
- human diversity,
- software diversity, and
- signal diversity.

The guidance in NUREG/CR-6303 provides a set of recommended criteria for each of the diversity attributes with several diversity criteria within each attribute.

To better reflect the nature of specific diversities, the attributes were expanded and clarified in NUREG/CR-7007, *Diversity Strategies for Nuclear Power Plant Instrumentation and Control Systems* [35]. The *human diversity* attribute is designated as the *life-cycle diversity* attribute to account for its true nature and to avoid the erroneous inference that this attribute involves plant operator diversity or human-versus-machine diversity. In fact, the human (i.e., life-cycle) diversity attribute relates to addressing human-induced faults throughout the system development life-cycle process (e.g., mistakes, misinterpretations, errors, configuration failures) and is characterized by dissimilarity in the execution of life-cycle processes. Additionally, the *equipment diversity* attribute is subdivided into two new attributes to reflect the differences related to the manufactured equipment source (i.e., the manufacturer or supplier) and the differences related to logic processing components (e.g., computational or processing elements such as CPU, printed circuit board, bus architecture for microprocessor-based equipment). Thus, the single *equipment diversity* attribute is treated as two diversity attributes: *equipment manufacturer* and *logic processing equipment*. Finally, the *software diversity* attribute is designated as the *logic diversity* attribute to account for the different means of representing and executing functions that diverse technologies provide (e.g., software for microprocessors, hardwired logic in programmable devices, electronic circuitry for analog modules).

The subsequent descriptions of the attributes and criteria are based on the NRC reports. In presenting each diversity attribute, associated criteria are given in order of diminishing impact. Where warranted, original language from NUREG/CR-6303 is indicated in quotes to illustrate the evolution of the criteria.

3.2.1.1 Design diversity

The NRC reports define *design diversity* as the use of different approaches, including both software and hardware, to solve the same problem or a similar problem. The focus for this diversity is on technology, approach, and architectural differences. Essentially, the design diversity attribute relates to technology choice and usage. For this attribute, three diversity criteria are identified that contribute to diversity between two designs that meet the same or similar requirements:

- different technologies (e.g., analog versus digital),
- different approaches within the same technology (e.g., transformer-coupled AC instrumentation versus DC-coupled instrumentation), and
- different architecture (i.e., arrangement and connection of components).

NUREG/CR-7007 further clarifies the criteria for design diversity. The criterion involving different technologies is equated to the use of fundamentally diverse technologies such as analog versus digital. The concept is illustrated through the example of analog modules such as square-root extractor, summing, and comparator/bistable circuits contrasted against digital modules based on a printed circuit board with a field programmable logic device (FPGA) or central processing unit (CPU) chip (i.e., microprocessor) providing the necessary computational capabilities.

The criterion involving different approaches within the same technology is described in terms of the use of distinctly different technology approaches. Distinct approaches within a broad technology class (i.e., digital) generally provide some intrinsic dissimilarity in the mechanisms by which functions are executed, as well as notable differences in the methods and tools for system implementation. Implementations based on either FPGAs or microprocessors are cited as examples of this type of diversity.

The criterion involving different architectures is clarified as corresponding to adoption of architectural variations within a particular technology (e.g., digital microprocessor based). Application of architectural variations within digital technology is achieved primarily through the use of different microprocessors. This type of technology difference at the system level is not often readily discernible (e.g., few obvious differences between two computer-based systems of similar composition and configuration providing similar functionality). The nature of design diversity within this strategy classification arises primarily at the microarchitecture level (i.e., CPU or processing element), although macro-architectural differences at the board or module level may also have an effect. Essentially, the these differences at the core of the logic processing equipment provide the principal design diversity in this case and result in some degree of hardware and software dissimilarity.

3.2.1.2 Equipment manufacturer diversity

The NRC reports identify four diversity attribute criteria that contribute to diversity between two groups or items of equipment that perform the same or similar function(s). The focus for these criteria, which were originally grouped under the general *equipment diversity* attribute in NUREG/CR-6303, is on the source of the hardware components or aggregate system. These criteria are as follows:

- different manufacturers of fundamentally different designs,
- same manufacturer of fundamentally different designs,
- different manufacturers making the same design, and
- different versions of the same design.

BTP 7-19 states that "[c]laims for diversity on the basis of the difference in manufacturer name" (i.e., "nameplate" diversity) are generally insufficient as the sole means to establish actual equipment diversity.

3.2.1.3 Logic processing equipment diversity

The NRC reports identify four diversity attribute criteria that contribute diversity in the equipment essential to providing logic processing of functions. These criteria were originally grouped under the general *equipment diversity* attribute in NUREG/CR-6303, and they are focused on the type of logic processing equipment employed. These criteria are as follows:

- different logic processing architecture, such as "different CPU architecture (e.g., Intel 80X86 architecture versus Motorola 68000)"
- different logic processing version in the same architecture, such as "different CPU chip versions (e.g., Intel 80386 versus Intel 80486)";
- different component integration architecture, such as "different printed circuit board designs"; and
- different data-flow architecture, such as "different bus structure (e.g., VME versus Multibus II)".

3.2.1.4 Functional diversity

NUREG/CR-6303 characterizes two systems as being functionally diverse if they perform different physical functions. The IEC defines *functional diversity* as "application of the diversity at the functional level (for example, to have trip activation on both pressure and temperature limit)" [19]. Therefore, there is a significant emphasis on the means of achieving a function and the nature of the function itself. The NRC reports identify three diversity attribute criteria that contribute to diversity of function between two independent systems:

- different underlying mechanisms (e.g., gravity convection versus pumped flow, rod insertion versus boron poisoning);
- different purpose, function (e.g., normal rod control versus reactor trip rod insertion), control logic, or actuation means; and
- different response time scale (e.g., a secondary system may react if accident conditions persist for a time)."

Aspect diversity, as identified in NUREG-0493, is a type of functional diversity. However, NUREG/CR-6303 notes that experience has shown this type of functional diversity to be less effective than was originally presumed.

3.2.1.5 Life-cycle diversity

NUREG/CR-6303 notes that the effect of human beings on the design, development, installation, operation, and maintenance of safety systems can be profound. The focus for the diversity criteria attributed to human influence, which was originally identified as "human" or "people" diversity, is on life-cycle resources that constitute potential sources of systematic faults. NUREG/CR-6303 also notes that management can significantly affect diversity through resource allocation and cultural effects. Four diversity attribute criteria that contribute to the diversity achieved throughout the life cycle of different designs are identified in the NRC reports:

- different design organizations/companies,
- different engineering management teams within the same company,
- different design and development teams (e.g., "designers, engineers, or programmers"), and
- different implementation and testing teams (e.g., "testers, installers, or certification personnel").*

3.2.1.6 Logic diversity

NUREG/CR-6303 defines *software diversity* as "the use of different programs designed and implemented by different development groups with different key personnel to accomplish the same safety goals." In keeping with the more general consideration of different means for processing functions that is available through different technologies, the diversity attribute is extended to address all forms of logic processing including software program execution. The NRC reports identify four diversity attribute criteria that contribute to diversity between logic processing approaches adhering to the same requirements. The basis for these criteria excludes the effects of human diversity, which is encompassed in the life-cycle diversity attribute. The logic diversity criteria are as follows:

- different algorithms, logic, and program architecture (e.g., computation structure or execution flow),
- different timing and/or order of execution,
- different runtime environment (e.g., "different operating system"), and
- different functional representation (e.g., "different computer languages").

3.2.1.7 Signal diversity

NUREG/CR-6303 defines *signal diversity* as the "use of different sensed parameters to initiate protective action, in which any of the parameters may independently indicate an abnormal condition, even if the other parameters fail to be sensed correctly." In this sense, signal diversity is related to functional

*This criterion can also include different maintenance technicians.

diversity, with one providing diverse indication and the other capturing the different functional relationships between indication and event. The NRC reports identify three diversity attribute criteria that contribute to diversity between measurement sources:

- different reactor or process parameters sensed by different physical effects (e.g., pressure or neutron flux),
- different reactor or process parameters sensed by the same physical effect (e.g., pressure versus water level or flow sensed by differential pressure sensors), and
- the same reactor or process parameter sensed by a different redundant set of similar sensors (e.g., a set of four redundant temperature sensors backed up by an additional set of four redundant temperature sensors driving a diverse design of protective equipment).

3.2.2 Research and Non-Nuclear Industries Terminology

The nuclear power industry provides the most formalized, definitive guidance on diversity as a strategy to mitigate CCF vulnerability. However, there are other sources of information on terminology relevant to failure mitigation. For example, some other industries provide guidance on treating CCF. The two primary examples arise from the commercial aviation industry and the chemical process industry. In addition, international research organizations have reported findings that add to the body of knowledge on CCF mitigation.

For the commercial aviation industry, the Society of Automotive Engineers (SAE) publishes the Aerospace Recommended Practice (ARP) standard 4754, "Certification Considerations for Highly-Integrated or Complex Aircraft Systems" [36]. SAE ARP 4754 addresses certification aspects of highly integrated or complex systems intended for installation on aircraft while accounting for the overall aircraft operating environment and functions.

In guidance on resolving identified failure vulnerabilities, SAE ARP 4754 identifies the use of system architectural features such as redundancy, partitioning, or dissimilarity to eliminate or contain the degree to which an item contributes to a failure condition. However, SAE ARP 4754 does not use the same definitions of key terms as the nuclear industry. The aviation industry terms of redundancy, partitioning, and dissimilarity are comparable to the nuclear industry concepts of redundancy, isolation, and diversity.

The concept of dissimilarity as used in aircraft design is similar to the concept of diversity as used in the nuclear power industry. The following excerpts from SAE ARP 4754 shows that the use of dissimilarity or diversity is encouraged:

"For all but the simplest systems, it is practically impossible to guarantee the correctness and completeness of requirements or the correctness of all necessary assumptions. An architectural strategy incorporating dissimilarity can be a powerful means of reducing the potential for errors in requirements or in design implementation to cause serious effects . . ." Additionally, the standard states that "[w]hen dissimilarity is used as a means of design error containment, the degree of credit should be related to the type and scope of design errors shown to be covered by the dissimilarity… . Assuming adequate independence can be shown, dissimilar design implementations of dissimilar functions can provide containment coverage for both implementation and function requirements errors." However, SAE ARP 4754 does not provide specific guidance on the types of dissimilarity required. Instead, it requires that substantial dissimilarity be demonstrated to protect an aircraft-level function when needed. Specifically, it identifies differences in the "designs in terms of the means of preventing the top level failure condition(s), the methodology by which the designs are created, the technology through which the designs are implemented, and the operations through which the functions are used."

For the chemical process industry, the American Institute of Chemical Engineers (AIChE) established the Center for Chemical Process Safety (CCPS) to develop and disseminate voluntary guidance for use in the prevention of chemical accidents. The CCPS *Guidelines for Safe Automation of Chemical Processes* [37] and *Guidelines for Safe and Reliable Instrumented Protective Systems* [38] provide extensive guidance on design practices for safety instrumented systems (SISs), which provide protective functions in chemical process plants.

For the highest integrity level functions and especially for hazardous processes, the CCPS recommends that diversity be considered and used where appropriate. *Diversity* is identified as "factors that make two components (e.g., devices, subsystems, systems, software systems, communications systems, sensors, or final control elements) different in a way that minimizes common mode fault" [37]. The CCPS further states that diversity "may include the use of different physical methods, technology, manufacturers, installation, maintenance personnel and/or environment" [38].

Additional CCPS recommendations established for diverse SISs include the use of different technologies, different manufacturers (or products from different vendors), and different application programming teams. For hardware diversity, different sensors and logic equipment are identified as options. For system software diversity, different controller/logic platforms and smart sensor devices are recommended. For application software diversity, development of different programs is recommended.

The treatment of diversity in commercial standards is consistent with the definitions of diversity established in the nuclear power and chemical process industries. The one unique diversity identified in the standards review as part of this research is test diversity. An annex of IEC 60880 includes diversity in testing as a diversity feature that can be considered for resolving software CCF.

The subject of CCF vulnerability has been an issue of interest within the research community for some time. Consequently, there are various forms of diversity that have been identified and investigated in recent research. For example, the British nuclear power industry funded a multiyear study of software diversity. The investigation was conducted by a team of university researchers under a program titled the DIverse Software PrOject (DISPO) [39].

As part of their research, the DISPO team investigated means for enhancing dependability. In particular, the application of diversity in digital I&C systems can be encouraged by invoking decisions in the management of the system design process. These choices are described as diversity-seeking decisions (DSDs). The effect of such decisions is to promote a high degree of fault diversity. The remaining challenge arises because the effect of these decisions on failure diversity (i.e., achieving reduced correlation between failure behaviors of different versions) is indirect. The research found that there is insufficient knowledge to definitively guide the choice among DSDs to effectively produce the desired failure diversity, and in turn, quantify improvement in system dependability. However, there is clear qualitative evidence of the benefit of applying these DSDs individually.

Two forms of "forced" diversity are discussed extensively in the DISPO research: (1) "normal" forced diversity and (2) functional diversity. In normal forced diversity, differences are imposed on development activities, resulting in different design versions that are based on the same underlying physical relationships that correspond to use of the same or similar inputs to indicate each specific event. Functional forced diversity employs alternate underlying physical relationships and results in different design versions using different inputs to provide indication of each event.

Diversity can be forced by imposing constraints on the software-based system development process to introduce development differences between two versions of a diverse redundant architecture. The desired benefit is that the difficulties presented to each development team will differ, so common faults would be unlikely to occur in the two versions. Based on the research, examples of DSDs include the following: "using different development environments, different tools and languages at every level of specification, design and coding, implementing each function with different algorithms, applying different V&V methods, etc." [39]. The identified DSDs are grouped according to data diversity, design diversity, and functional diversity. In this context, *data diversity* refers primarily to input differences achieved by measurement dissimilarity, stochastic signal behavior, analytical variation (re-expression), and loose coupling between functional instantiations (e.g., asynchronous execution of function in separate systems).

The *design diversity* discussed in the DISPO research has a more expansive scope than it does in NRC guidance. Design diversity embodies all of the design options that can engender diversity in the development of parallel systems that provide the same or similar input-to-output function. It includes differences in the system life-cycle process (e.g., resources, methods, tools) as well as different implementations of the functionality. Essentially, the DISPO concept of design diversity incorporates several NRC diversities (design, equipment manufacturer, logic processing equipment, life cycle and logic) under a single type. Effective selection among DSDs can establish "cognitive" diversity for the designers, implementers, testers, and so forth, thereby minimizing the potential for common mistakes, errors, and misunderstandings that can lead to systematic faults. *Functional diversity* involves the establishment of different functional relationships (e.g., diverse parameter and initiation criteria to protect against the same postulated initiating events) as the basis for diversity. Signal diversity, as discussed in the DISPO research, is necessary to enable functional diversity.

In other international research, Lindner [40] and his colleagues at the Institut für Sicherheitstechnologie (Institute for Safety Technology, or ISTec) conducted several theoretical case studies on CCF mitigation. One key finding involved establishing the concept of an additional diversity (i.e., different internal time, τ). This diversity is invoked through staggered restarts/reboots of individual redundancies, while in a bypassed condition, within a multichannel system. This enforced internal time diversity is essentially a form of temporal or platform operational cycle (i.e., execution history) diversity. In the test cases, this "history" diversity was employed in addition to the other seven diversity attributes (design in the form of different architectures, equipment manufacturer, logic processing, equipment, functional, life cycle, logic, and signal) typically used in nuclear power applications. The introduction of history diversity through different internal times helps to resolve concerns about execution history dependence among redundant channels that are executing the same software on the same platform. Specifically, it was noted that the use of different internal times between redundancies reduces commonality of signal trajectory in terms of internal states (and state transitions). Essentially, any faults triggered by time (or execution) dependence would affect only one redundancy given the staggered restarts. The primary impact of the history diversity approach is to diversify the execution profiles of software-based systems by reducing the potential impact of platform usage deficiencies (e.g., buffer overwrites, stack overflows, pointer errors, race conditions).

Finally, Hawthorne and Perry [41] treat the methods of addressing CCF in dependable systems by defining types of diversity and establishing an architectural framework to capture top-to-bottom design diversity. The architectural framework of Hawthorne and Perry incorporates the whole system by representing hardware, software, and infrastructure. A physical representation is adopted for a high-level model of hardware elements (processor, memory, etc.), software elements (applications, layered software components such as utilities and system services, operating system, etc.), and architectural infrastructure (networks, power, etc.). Diversity-enhancing properties of the system design are identified as modal diversity, geographical diversity, and ecological diversity. Modal diversity provides for diverse modes of accomplishing system functions (e.g., functional diversity). Geographical diversity involves distributing hardware/software components to avoid localized failures due to environment and other external influence factors. Ecological diversity can be achieved by employing dissimilar hardware, software, networks, and other infrastructure components to protect against platform or technology specific vulnerabilities. Other diversity properties, such as temporal, control, and combinational diversity, were also identified. Temporal diversity described by these authors involves adaptability of the system to adjust to temporal variability (e.g., variable timing of events). In this sense, the diversity relates to communication delays and lags in data. Control diversity involves differences within and among automatic and manual control capabilities. Examples include distributed versus centralized control or autonomous versus cooperative (e.g., human-machine teaming) control. Combinational diversity involves using unique, but not necessarily mutually exclusive, sets of diversities across multiple parallel systems (e.g., diverse redundant channels). Essentially, the combination of diversity-enhancing properties would vary from channel to channel to achieve this form of system-level diversity.

Page intentionally blank

4. CLASSIFICATION APPROACHES SUITABLE FOR I&C SYSTEMS AND THEIR CHARACTERISTICS

A *classification approach* is a necessary element of a taxonomy. In order to systematically represent and organize CCF vulnerability of I&C systems and the mitigation techniques to resolve those vulnerabilities, classification approaches were investigated to identify those suitable for characterizing I&C systems, functions, and failures. Classification and system representation schemes from nuclear power applications were considered as well as approaches from relevant technical domains such as digital technology, systems analysis, and cognitive engineering. The identified classification approaches can be characterized as four primary types based on their means of organizing information. These approaches to representing system information are:

- safety significance,
- physical representation,
- functional representation, and
- critical characteristics.

This section summarizes the identified classification approaches that address characteristics of I&C systems relevant to establishing the system context for CCF vulnerability and mitigation.

4.1 Safety Significance

The nuclear power industry has historically classified I&C systems according to safety significance. This classification approach is based on a deterministic assessment of the role in assuring safety assigned to the functions accomplished by those systems. Safety classification is well established and is primarily relevant in terms of establishing the safety significance of CCF vulnerability and mitigation, as well that of as the affected systems. What follows is an overview of the prevailing safety classification approaches employed by the international nuclear power industry.

The Code of Federal Regulations [28] establishes a classification approach for structures, systems, and components (SSCs) in a nuclear power facility. Section 2 of 10 CFR 50 defines safety-related SSCs in terms of reliance on those SSCs to remain functional during and after design basis events to assure (1) the integrity of the reactor coolant pressure boundary, (2) the capability to shut down the reactor and maintain a safe shutdown condition, and (3) the capability to prevent or mitigate the consequences of accidents that could result in unacceptable offsite exposures.

For electrical and I&C equipment, 10 CFR 50.49 identifies classification of systems as important to safety. The scope of systems included in the important-to-safety class include safety-related systems, those nonsafety-related systems whose failure under postulated environmental conditions could prevent satisfactory accomplishment of safety functions by safety-related systems, and certain post-accident monitoring systems.

IEEE further identifies I&C systems important to safety as Class 1E equipment. In IEEE 323-2003, “IEEE Standard for Qualifying Class 1E Equipment for Nuclear Power Generating Stations” [42], Class 1E is defined as the “safety classification of the electric equipment and systems that are essential to emergency reactor shutdown, containment isolation, reactor core cooling, and containment and reactor heat removal, or are otherwise essential in preventing significant release of radioactive material to the environment.”

In addition to the traditional deterministic classification approach, a risk-informed approach to safety classification has been established in 10 CFR 50.69. Specifically, SSCs are divided into risk-informed safety classes (RISCs) based on both deterministic safety classification and probabilistic significance to plant safety. In this classification approach, insight from a PRA on the safety significance of the function performed by a system is captured based on its contribution toward reducing the risk of release of radioactive material to the environment. A *safety-significant function* is defined as "a function whose degradation or loss could result in a significant adverse effect on defense-in-depth, safety margin, or risk." The four RSICs are identified as follows:

- *RISC-1* – safety-related systems that perform safety significant functions,
- *RISC-2* – nonsafety-related systems that perform safety significant functions,
- *RISC-3* – safety-related systems that perform low safety significant functions, and
- *RISC-4* – nonsafety-related systems that perform low safety significant functions.

IAEA defines a deterministic safety classification for I&C systems in IAEA Safety Guide NS-G-1.3, *Instrumentation and Control Systems Important to Safety in Nuclear Power Plants* [43]. The classification approach involves assigning a safety class based on the importance to safety of the function performed by the I&C system. Thus, the safety guide divides I&C systems into "systems important to safety" and "systems not important to safety." An I&C system important to safety is one whose malfunction or failure could lead to unacceptable radiation exposure of the site personnel or members of the public. Systems important to safety are further subdivided into "safety systems" and "safety-related systems." Safety systems perform protective functions, while safety-related systems are those I&C systems that perform important functions other than the main protective functions.

The nuclear power standards issued by the IEC adhere to the safety principles established by the IAEA. However, the IEC refines the safety classification approach established by the IAEA by resolving the important-to-safety class based on a three-tiered approach to identifying both I&C systems and the functions they perform. The IEC safety classification approach is based on the IAEA safety philosophy and the plant design base. All SSCs that are items important to safety, including software for digital I&C systems, are classified on the basis of their function and significance with regard to safety. Basically, I&C systems that provide functions to cope with postulated initiating events (PIEs) are classed in the highest safety class while less important functions and equipment are assigned to lower safety classes.

IEC standards provide criteria for assignment of functions to safety categories and establish design requirements for the corresponding I&C systems and equipment. In IEC 61513, the general requirements for I&C systems important to safety are established in terms of safety classes. I&C systems are assigned to one of three safety classes (i.e., Classes 1, 2, and 3), or are unclassified based on their main safety function. The determination of classification for safety functions is established in IEC 61226, "Nuclear Power Plants—Instrumentation and Control Systems Important for Safety—Classification" [44]. This standard classifies functions into three categories (i.e., Categories A, B, and C). Category A functions play a principal role in achieving or maintaining safety by preventing design basis events from leading to unacceptable consequences. Category B covers functions that complement Category A functions in assuring safety, especially those functions required to operate after a nonhazardous stable state has been achieved. Category B also includes functions whose failure could initiate a design basis event or worsen the severity of an event. Category C addresses functions that play an auxiliary or indirect role in the achievement or maintenance of NPP safety. Other functions that do not meet the criteria of the three categories are identified as "nonclassified" (NC).

In addition to the international classification approach defined in safety guides and standards, there are various national approaches that reflect domestic practice and conventions. Table 4.1, which was adapted from Ref. 12, illustrates several safety classification approaches that have been reviewed. The table does not constitute a high-fidelity mapping of the classes and categories employed by the international nuclear power industry but, it does provide a general indication of the approximate relationship among the classification approaches.

The identified safety classification approaches are generally based on a deterministic assessment of the safety significance of the main function performed by an I&C system. As seen in the establishment of the RISC classes, risk insights can be incorporated into the safety classification structure. Identification of safety significance is a fundamental element of any classification approach adopted for characterizing CCF vulnerability and mitigation within an I&C system context. Thus, treatment of safety significance within a CCF taxonomy must be compatible with existing safety classes.

Table 4.1. Comparison of international safety classification structures

National or international standard	Classification of the importance to safety			
USA	Systems important to safety			Nonsafety
	Class 1E, safety, or safety-related		Systems whose failure can inhibit safety functions	
	RISC-1, RISC-3			RISC-2, RISC-4
IAEA	Systems important to safety			Systems not important to safety
	Safety	Safety related		
IEC 61226	Systems important to safety			Unclassified
	Category A	Category B	Category C	
IEC 61513	Systems important to safety			Unclassified
	Class 1	Class 2	Class 3	
European Utility Requirements	F1A (automatic)	F1B (automatic and manual)	F2	Unclassified
France N4	1E	2E	IFC/NC	
Japan[a]	PS1/MS1	PS2/MS2	PS3/MS3	Nonnuclear safety
Korea	IC-1	IC-2	IC-3	Non-IC
Russia	Class 2 (safety system, design basis accident)	Class 3		Class 4 (systems not important to safety)
Switzerland	Category A	Category B	Category C	Not important to safety
UK	Category 1		Category 2	Unclassified

[a] PS = prevention system, MS = mitigation system

4.2 Physical Representation

Information on I&C systems can be most directly captured and organized in terms of the physical elements of the implemented system. Block diagrams and schematics provide an amenable structure for an abstract representation of the detailed physical layout of a system, module, or circuit. For example, Fig. 4.1 shows a physical representation of a typical NPP instrument channel from sensor to actuator. A physical representation is an especially convenient approach to capture information on analog I&C systems, in which discrete elements typically provide dedicated, hardwired functionality. For I&C systems, multiple functions can be realized in a single module via software implementation, and the complexity of the computational element makes it difficult to isolate specific functions or capabilities. Consequently, it is more difficult to represent an I&C system solely by a physical model.

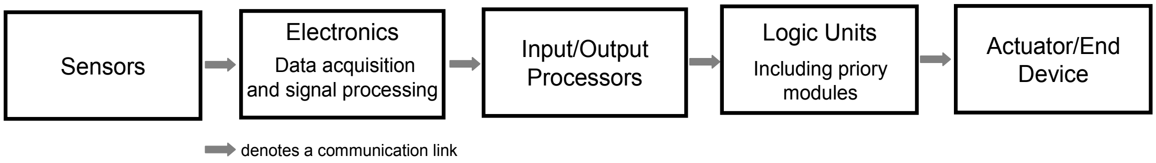


Fig. 4.1. Block diagram of the main elements of an instrument channel.

A physical representation of a plant I&C architecture, system, subsystem, module, or component provides a direct link with actual implementation of I&C systems within an NPP. Detailed element-by-element capture of information given the large number of constituent physical components in an overall plant I&C architecture is an imposing task. Thus, an abstraction of the physical implementations of the many I&C systems is necessary to enable the system context for CCF vulnerability and mitigation to be represented. A common practice to manage complexity is to employ a whole-part abstraction to represent the physical architecture. In a *whole-part abstraction*, a system is modeled as a group of related components at several levels of physical aggregation. Effectively, the approach establishes a hierarchical decomposition of higher-level physical systems into subelements in which the relationship of parts to the whole is represented. Figure 4.2 illustrates a whole-part abstraction approach applied to a generic system.

Adopting a physical representation approach to classification begins with identifying the I&C systems that constitute the overall plant I&C architecture. An example of a whole-part abstraction of I&C systems within the nuclear power industry is found in the technology roadmap on instrumentation, control, and human-machine interface to support the DOE Advanced Reactor programs [45]. Specifically, the roadmap characterized a generic I&C and human-system interface (HSI) system in terms of subsystems involving sensors, monitoring, automation and control, communication, and human-system interfaces.

This model for representing a system was expanded and used to establish a framework to support classification of I&C system degradation as part of a study by Brookhaven National Laboratory (BNL) on the impact of such degradation on human performance [46]. In the BNL framework, the HSI subsystem was decomposed into six subsystems:

- alarms,
- information systems,
- computerized operator support systems (COSSs),
- controls,
- communication systems, and
- workstations.

By coupling the physical representation of an I&C and HSI system to human performance factors, the BNL study established a framework to support top-down and bottom-up analyses of the impact of I&C degradation on operator tasks. The interrelationships established between I&C subsystems, HSI subsystems, and functional tasks corresponding to key human performance factors are shown in Fig. 4.3. Although examples are documented in the report, the comprehensive set of direct associations between the levels of the framework remains to be developed through detailed system and task analyses.

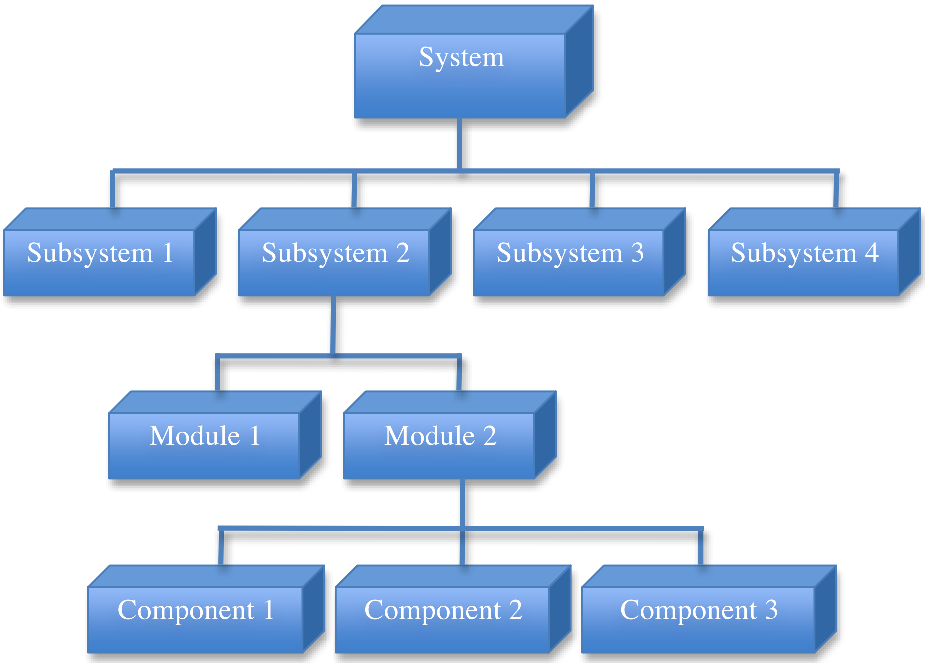


Fig. 4.2. Whole-part representation of a system.

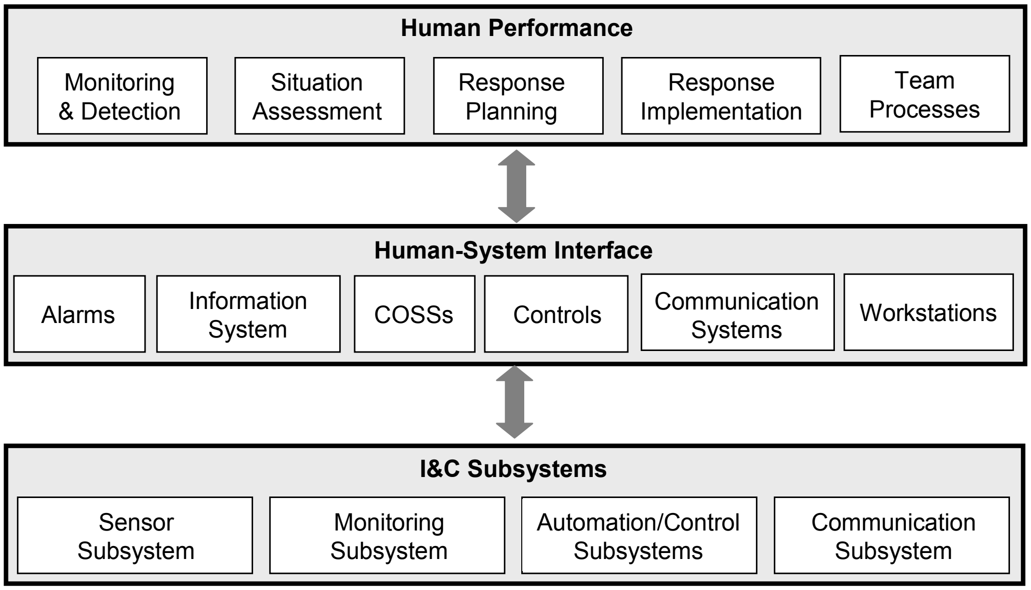


Fig. 4.3. Subsystem representation of an I&C and HSI system.

This classification approach is intended to provide a framework for analysis, demonstrating how human interaction elements of I&C systems can be incorporated into a physical representation. In addition, an approach to linking key characteristics of an application, human performance in this case, with an architectural representation of systems as a means to support analyses is also identified. The key to employing this approach is to establish the linkage or association between system elements and characteristics of interest (e.g., performance, design, failure).

An example of a database within the nuclear power industry that uses a physical representation of plant systems to organize data is found in the coder's manual for the Sequence Coding and Search System (SCSS) [47]. The SCSS provides a searchable database of licensee event reports (LERs). As part of the SCSS, system codes are provided for types of I&C systems. There are 18 codes identified in the coder's manual. These system categories identified for the SCSS are the following:

- alarms/annunciators,
- (plant) computer,
- fire detection,
- environmental monitoring,
- emergency generator I&C,
- turbogenerator I&C,
- plant monitoring/post-accident monitoring,
- in-core/ex-core neutron monitoring,
- pressure boundary leak monitoring,
- radiation monitoring,
- reactor power control,
- recirculation flow control,
- feedwater control,
- reactor protection,
- engineered safety features actuation,
- solid state protection and control system,
- anticipated transient without scram, and
- nonnuclear instrumentation.

Within the SCSS structure, a system can be further identified through a manufacturer's code, an interface code (identifying those plant systems to which the specific system has an interface), and a code identifying the type of component involved in the initiation of the reportable event. However, categorization of systems for the SCSS does not serve as the basis for its primary classification of data. The event is the fundamental data object in the classification approach employed for the SCSS rather than the system or component. In particular, codes are provided to characterize the event and its consequences. These codes include proximate cause, effect on plant, shutdown method, facility status, method of discovery/detection, form and content of any activity release, and type of personnel exposure. Thus, while the SCSS employs a high-level physical representation of plant systems in extracting data from LERs, the physical instance of a system or component is captured as a data attribute corresponding to an event rather than primary data object of interest. In particular, the coupling of system data with event data illustrates a means of linking or associating data collections of one type (e.g., a system architecture) with data collections of another type (e.g., event or failure databases) through common data items (i.e., attributes or properties for data objects).

Another I&C system information collection that involved physical representations in collating data was developed under an Electric Power Research Institute (EPRI) project. The objective was to develop qualification guidelines for application of programmable logic controllers [48, 49]. Part of the effort included generation of technical descriptions for I&C systems at NPPs. The information capture involved identification of safety-related and nonsafety-related I&C systems at NPPs. In addition, the systems were decomposed according to five attributes: system architecture, input signals, output signals, operator interface, and system functions. The first four attributes primarily correspond to physical elements of the

I&C system, while the fifth attribute involves a functional representation of the system. In addition, the physical attributes also include subattributes that arise from critical characteristics such as modes of operation, testability and self-diagnosis, access control, and so forth. While dated in terms of the system decomposition, EPRI's system identification associates physical, functional, and critical characteristic attributes within a comprehensive structure.

Some comclusions can be drawn regarding classification approaches based on physical representations. First, a physical representation provides a basis for organizing information about I&C systems that can be readily associated with the actual implementations and specific configurations, thus preserving a relationship with the installed I&C systems. In particular, a whole-part abstraction approach permits I&C system architectures to be represented while supporting capture of more detailed configuration information through successive decompositions through the whole-part hierarchy. Second, features and capabilities, such as software and human interfaces, strongly depend on recognition of function and critical characteristics that cannot be completely represented solely on the basis of a physical model. Thus, a CCF taxonomy should involve a classification structure that enables physical, functional, and critical characteristic information to be captured and interrelated.

4.3 Functional Representation

As previously noted, function relates to purpose and serves as a primary basis for establishing safety classification. A functional representation of I&C systems is frequently illustrated in block diagram form with the primary functions that are assigned to different elements of a system identified as blocks arranged to correspond with architectural (i.e., physical) structure of the I&C system. A functional block diagram that is comparable to the instrument channel diagram of Fig. 4.1 is shown in Fig. 4.4. The unidirectional arrows represent data flow, and the bidirectional arrow indicates the functional interconnections between the channel functions and the interface functions for human interaction.

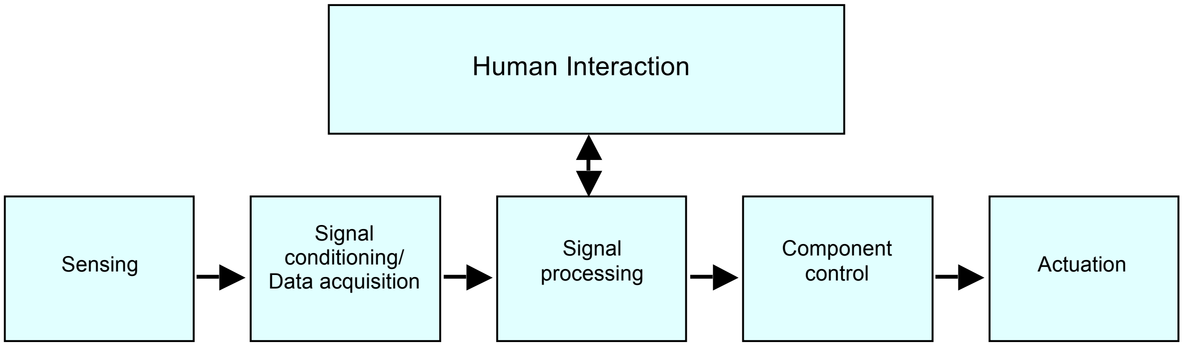


Fig. 4.4. Block diagram of a typical I&C function for an instrument channel.

IAEA Nuclear Energy Series Report No. NP-T-3.12, *Core Knowledge on Instrumentation and Control Systems in Nuclear Power Plants*, provides a basic introductory description of I&C systems and notes that the variety of technological elements that constitute an NPP's I&C system architecture can be difficult to address as a whole because of the depth and breadth of the discipline. Consequently, the report characterizes the I&C systems in terms of three viewpoints representing the full scope of the technical discipline: functional, physical, and life cycle. These viewpoints illustrate the purpose of the I&C systems (functional), the embodiment of those systems (physical), and the means by which those systems are realized and maintained (life cycle).

The functional approach employed in the IAEA report to characterize a generic I&C system architecture focuses on plant-wide system objectives and the means of achieving those objectives. This function-based representation addresses the sensory, communications, monitoring, display, control, and command systems interposed between the process (i.e., the reactor, heat transport, and energy conversion systems), and the plant personnel (i.e., operations and maintenance staff). A whole-part decomposition of function, ranging from plant function to system functionality, is illustrated through basic examples and a high-level overview of a generic functional architecture. The functional architecture representation adopts a coupled functional–physical approach in which the distribution of function within the I&C architecture of a plant is addressed at a high level of aggregation.

In the field of cognitive systems engineering, Rasmussen [50] established a functional abstraction hierarchy in which the structure embodies the relationship between goals and the method of accomplishing those goals (i.e., means to achieve an end). A means-ends abstraction hierarchy captures the functional properties of a system by relating ends (goals) at higher levels to means (methods) at lower levels. At each level of abstraction within the hierarchy, those functional properties are represented by distinct concepts such that each level describes the system using a different set of attributes. Functional properties include information about the purpose of a system, the approach employed by a system (e.g., processes and phenomena), and the physical realization of a system. Thus, the organizational approach employed for capturing system function relates *what* (i.e., functional approach and modes of operation) at one level of the hierarchy to *how* (i.e., specific functions and functionalities) at the level below, and *why* (i.e., goal) at the level above.

The conceptual framework established by Rasmussen involves the following levels of abstraction:

- functional purpose,
- abstract function,
- generalized function,
- physical function, and
- physical form.

At the highest level of abstraction, the overall purpose (i.e., the intended functional effect) of a system is represented. The abstract function level involves fundamental concepts and causal relationships to represent the overall proper function of a system. The generalized function level addresses basic functional relationships and natural processes that are independent of specific physical implementations. The functional states of a system, which are tightly related to its physical form, are represented at the physical function layer. Finally, the representation of physical form (i.e., components and configuration) provides the lowest, most concrete level of abstraction.

A top-down progression through the hierarchy establishes a “purpose” basis for the functional representation of a system to capture information on a specific purpose that can be realized by several physical implementations. A bottom-up progression through the hierarchy establishes a “physical” basis for the functional representation of a system to capture information on a specific implementation that can serve several purposes. The hierarchical structure associated with a means-ends abstraction suggests an approach to organizing functional information for I&C systems. The close connection between physical function and physical form at the hierarchy’s lower levels is consistent with common representations of function, as discussed above. In addition, the tie between purpose and functional implementation supports capturing the significance of functions.

Rasmussen *et al.* [51] employed coupled means-ends abstraction and whole-part decomposition to establish a framework to support cognitive work analyses. The abstraction and decomposition hierarchies were applied to application areas or domains (e.g., the domain of potential risk and the domain of mitigation resources for analysis of decision making). In an unpublished white paper in 1995, Leo Beltracchi of the NRC Office of Nuclear Regulatory Research extended the Rasmussen analysis framework to address design and safety in the requirements for I&C systems at NPPs. In the Beltracchi approach, a means-ends abstraction hierarchy is applied to the hazards and risk domain and the mitigation and defense domain. These domains represent principle considerations in system design. The analysis framework also involves whole-part decomposition of I&C system design, plant functions, and hazards. Figure 4.5 illustrates the Beltracchi analysis framework in which function groupings are established and hazard classes are indicated. A high-level breakdown of a generic I&C system is provided in terms of hardware, software, and human interfaces. Input, output, and human-system interface functions are also treated. Within each functional grouping, critical characteristics (e.g., redundancy, diversity, independence, defense-in-depth) are identified to guide design assessment of the safety properties provided by the I&C system requirements. The purpose of the Beltracchi framework is to support analysis of I&C system requirements to assess safety characteristics (e.g., integrity). Thus, the identification of critical characteristics within a functional-physical framework can serve to enable representation of the consequences of CCF.

As seen in the various approaches investigated, a clear understanding of the relationship between abstracted data and the actual I&C system is facilitated by maintaining close correlation between function and plant I&C architecture (i.e., coupled functional and physical representations) in any analysis of CCF vulnerability and mitigation. The association of critical characteristics with functional (or physical) groupings (e.g., classes), such as for the Beltracchi framework, can support evaluation of system features or failure behavior based on importance relative to safety or performance.

4.4 Critical Characteristics

Systems can be represented in terms of their properties, features, and qualities. A classification approach based on critical characteristics is common in many fields, such as systems engineering, biology, physics, sociology, etc. Weinberg [52] described systems according to three categories based on characteristics of randomness and complexity. The categories are described as *organized simplicity*, *unorganized complexity*, and *organized complexity*. Systems with low degrees of complexity and randomness fall within the organized simplicity category. These systems are characterized by simplicity and organization such that they can be readily decomposed by analytic reduction into noninteracting subsystems. Systems with a high degree of randomness are grouped within the unorganized complexity category. These systems typically do not display an easily identifiable underlying structure and are not readily subject to decomposition. Instead, these systems may be represented as aggregates of interchangeable elements whose behavior can be treated in terms of average properties through statistical analysis. The systems within the organized complexity category are too organized to be treated solely in terms of statistical properties and too complex to be reduced and thoroughly analyzed. Complex software is an example that fits within the organized complexity category, in which both statistical and reduction analyses are insufficient to solely address the full range of design and assessment issues posed by software-based systems.

Systems within the organized complexity category are treated using methods developed under systems theory. In this approach, complex systems are represented hierarchically with control processes establishing the primary interfaces between levels [51]. Effectively, control laws impose constraints from higher levels of the hierarchy onto system/component behavior at lower levels. In the Systems-Theoretic Accident Model and Processes (STAMP) framework for analyzing accidents, Leveson [53] established an approach to classifying accident factors in terms of control loop and process models. These models enable constraints and levels of control to be represented in terms of a typical control loop structure. In addition to a controller, actuators and sensors, and the controlled process, the human supervisor is also represented. Displays and controls provide the interfaces between the controller and the human supervisor while measured and controlled variables provide the interfaces between the controller and controlled process, through the sensors and actuators, respectively. Capturing the causal factors that lead to accidents in terms of flaws incorporated into the components of the control loop at various life-cycle phases (e.g., design, development, manufacturing, operations), Leveson classified control flaws leading to hazard in three categories: (1) inadequate enforcement of constraints (i.e., insufficient control), (2) inadequate execution of control action, and (3) inadequate or missing feedback. Each category is further decomposed into more specific sources of flaws. The STAMP framework is intended to facilitate accident analysis by including representation of the control structure itself in the assessment of what flaws are present and why an event occurred.

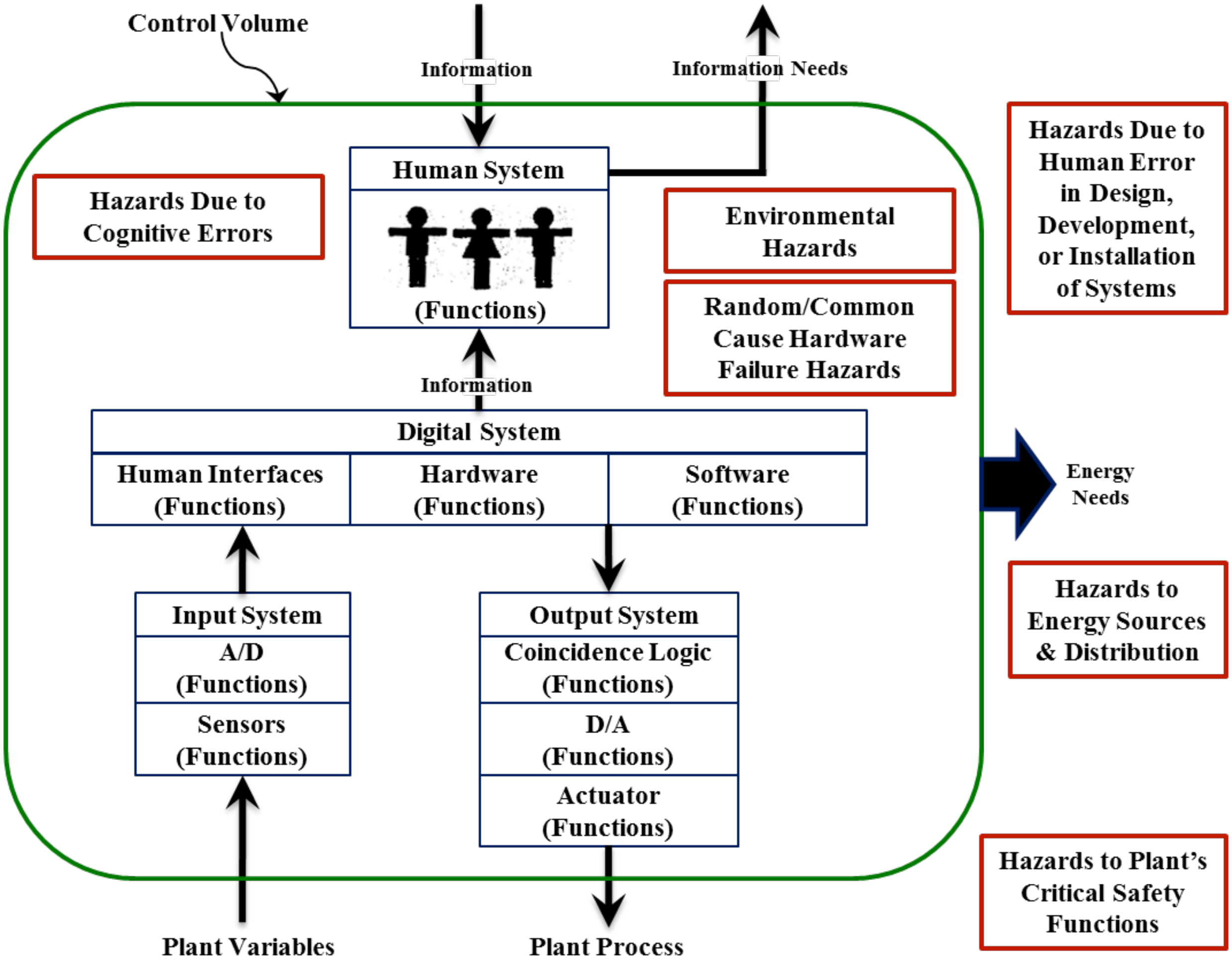

Control Volume
Information
Information Needs
Human System
(Functions)
Hazards Due to Cognitive Errors
Environmental Hazards
Random/Common Cause Hardware Failure Hazards
Hazards Due to Human Error in Design, Development, or Installation of Systems
Information
Digital System
Human Interfaces (Functions)
Hardware (Functions)
Software (Functions)
Energy Needs
Hazards to Energy Sources & Distribution
Input System
A/D (Functions)
Sensors (Functions)
Output System
Coincidence Logic (Functions)
D/A (Functions)
Actuator (Functions)
Hazards to Plant's Critical Safety Functions
Plant Variables
Plant Process

Fig. 4.5. Beltracchi analysis framework.

Failure analysis frequently involves classification of systems and faults in terms of critical characteristics. Laprie [54] established a tree structure to define dependability characteristics for a system. The three dependability branches are *attributes*, *means*, and *threats*. Attributes of dependability include properties such as availability, reliability, safety, confidentiality, integrity, and maintainability. Means for achieving dependability include fault prevention, fault tolerance, fault removal, and fault forecasting. Threats to dependability are characterized as faults, errors and failures. Laprie further classified faults and failures according to tree structures. Faults are classified according to five categories, which are phenomenological cause (physical and human-made faults), nature (accidental, intentional nonmalicious, and intentional malicious faults), phase of creation or occurrence (developmental and operational faults), system boundaries (internal and external faults), and persistence (permanent and temporary faults). Failures are classified according to three categories, which are domain (value and timing failures), user perception (consistent and inconsistent or "Byzantine" failures), and consequences on environment (benign through catastrophic failures).

Lala and Harper [55] adapted Laprie's fault classification approach to address CMF. Lala and Harper grouped the phenomenological cause, phase of creation, and system boundaries categories as subelements of a higher-level classification by origin. Thus, common-mode faults are classified according to nature, origin, and persistence. Treatment of CMF follows by relating the fault classification to a classification of CMF sources (i.e., common-mode faults that lead to CMF). In their taxonomy for CMF, Lala and Harper exclude consideration of intentional faults based on the prevailing condition that security had not often been treated as a requirement for ultra-reliable real-time applications. Five classes of CMF were defined in terms of transient externally induced faults, permanent externally induced faults, intermittent design-related (life-cycle) faults, permanent design-related faults, and interaction (human-system interaction) faults. Lala and Harper note that methods to address common-mode faults typically rely on one of the primary means of achieving a dependable system. Treatment of CMF typically involves fault-avoidance techniques during early life-cycle phases (specification, design and implementation), fault-removal techniques during later life-cycle phases (test and validation), and fault-tolerance techniques during the operational life-cycle phase.

In an assessment of operational experience insights into CCF for digital I&C systems, EPRI 1016731, "Operating Experience Insights on Common-Cause Failures in Digital Instrumentation and Control Systems" [56], identifies groupings of cause for common defects. These groupings include incorrect parameter value, single point vulnerability, hardware failure, manufacturing defect, inadequate requirements definition, inadequate hardware design, inadequate software design, inadequate software verification and validation, inadequate testing, inadequate operating procedures, inadequate maintenance procedures, inadequate vendor information, inadequate training, operator error, maintenance error, human performance, ineffective configuration management, ineffective change management, and ineffective vendor oversight. In addition, based on an analysis of failure for both Class 1E and non-Class 1E systems, the EPRI investigation identified key design attributes that can affect CCF mitigation. These design characteristics include redundancy, shared resources, signal diversity, functional diversity, use of formal software quality assurance methods, functional complexity, and system interactions.

The NRC Common-Cause Failure Database and Analysis System (CCF DAS) at Idaho National Laboratory established a classification structure to capture the main elements of CCF events. [57] Nuclear power plant I&C systems are classified into four categories—system, components, subcomponents, and piece parts—based on a whole-part decomposition. Events are classified in terms of component fault state (available or unavailable), cause, and coupling factor. The coding system for component fault state decomposes into "no failure" and "potentially unavailable" (expanding into "potentially failed" and "potentially functionally unavailable") for the "available class" and "failed" and "functionally unavailable" for the "unavailable" class. Failure causes are grouped as proximate cause or root cause. Associated conditions and triggering events can also be identified. Major categories established within the cause class include the state of other components, design/manufacturing/construction inadequacy, abnormal environmental stress, human actions/plant staff error, internal, procedure inadequacy, and unknown. Coupling factors are classified as being hardware based, operation based, or environment based. Defense mechanisms to address CCF are identified according to their prominent characteristic, such as functional barrier, physical barrier, monitoring and awareness, maintenance staffing and scheduling, component identification, diversity, unknown, and no practical defense. Assignment to these classes is based on analysis of events extracted from failure and event databases.

NRC RES staff [58] documented an initial investigation of classification approaches that could serve to support evaluations of operational experience and investigations of strategies to address CCF vulnerability. As part of this effort, an approach was proposed for classifying digital I&C systems based on critical characteristics relevant to interpreting operational experience. Arndt [59] further described this classification approach and discussed how it could be employed to determine the level of detail in reliability modeling necessary to support performance-based regulatory treatment of I&C systems. As described below, the three-attribute taxonomy for classifying I&C systems in terms of critical characteristics incorporates concepts from other classification approaches. In particular, the NRC approach, documented by Arndt, builds on concepts reported by John Rushby, Charles Perrow, and Tunc Aldemir.

Rushby [60] documented an approach to classifying critical properties of systems in a systematic taxonomy, analyzing representations of critical system properties from the standpoint of dependable, safe, secure, and real-time systems. Rushby noted that the taxonomy for critical systems must address the condition that "modern systems are often required to satisfy two or more critical system properties simultaneously" so design aspects such as security, fault tolerance, real-time performance, and safety must be considered.

Rushby adopted two attributes identified by Perrow [61] as the basis for an organization of critical system properties. These attributes are *interaction* and *coupling*. The interaction attribute characterizes the degree to which the behavior of a component in a system can impact the behavior of other components. Interaction is expressed in terms of complexity, ranging from linear to complex behavior. The coupling attribute characterizes the extent to which system(s) behavior is rigid (tightly coupled) or flexible (loosely coupled) for factors such as input order, timing, and execution sequence. Although the assessment of system properties according to these two principal attributes is subjective, the Rushby approach does enable systems to be classified in terms of the degree of interaction and coupling exhibited through their critical design characteristics.

In NUREG/CR-6901, Aldemir [62] developed a taxonomy for failure modes of digital I&C systems that more fully resolves the interaction attribute while clarifying the coupling attribute. Basically, the determination of what constitutes loosely coupled and tightly coupled is more completely defined by establishing subattributes for interaction. Aldemir defines *Type I interactions* as being those among digital I&C systems and the plant processes that are controlled by the systems while *Type II interactions* address activities within digital systems such as communication, multitasking, and multiplexing. Type I interactions can produce statistically interdependent failure modes due to the coupling of system performance through plant processes. Type II interactions can result in failure modes that originate from the coupling among systems and components (e.g., hardware, software, firmware).

The NRC classification approach extends the concepts of Rushby, Perrow, and Aldemir by establishing an I&C system representation in terms of system complexity, system interaction/ interconductivity, and system importance. System complexity is based on Type II interactions and other indicators of system size and complexity (e.g., function point or cyclomatic complexity metric). The intention is to capture intrinsic interactions between digital system elements such as hardware, software, and firmware and the overall complexity of an I&C system. System interaction is based on Type I interactions and system coupling. This attribute deals with extrinsic interactions for a system with the plant and with other systems. It addresses functional and physical interconnections and dependencies among systems and differentiates between tightly and loosely coupled systems. System importance is based on importance measures that include traditional safety and risk metrics, as well as indicators of significance toward maintaining key design and operational concepts (e.g., defense-in-depth). The application of the NRC classification approach involves characterization of an I&C system in terms of each of the three attributes, which range from simple to complex, loosely coupled to tightly coupled, and low importance to high importance. This approach was demonstrated in a conceptual example reported in Ref. 59 in which a generic reactor protection system (RPS) and generic digital feedwater control system (DFWCS) are classified. The RPS is expected to exhibit relatively high-risk importance but low complexity. The DFWCS is expected to show relatively low importance but much higher complexity and interaction. The approach to classification is illustrated in Fig. 4.6, which is drawn from Ref. 59.

The investigation of classification structures that involve categorization based on critical characteristics suggests an approach that can provide for grouping I&C systems by properties or qualities of primary interest in operational experience analyses. However, many of the characteristics identified in the classification approaches studied are highly subjective in nature (e.g., complexity, diversity) and do not have well-defined classes or comprehensive measures. Consequently, a subjective assessment of I&C systems is necessary that may often be limited to coarse categorization based on yes/no, high/low, or other similarly rudimentary value judgments.

Furthermore the critical characteristics approach to classification depends on categorization that is abstract in nature. In many cases, the relationship between class structure and the physical system is highly obscured by the degree of abstraction. While sorting and portrayal according to observed or inferred characteristics is reasonable for organizing system information, the inverse process of reconstituting concrete details of an I&C system from a depiction based on intangible features can be a severe challenge.

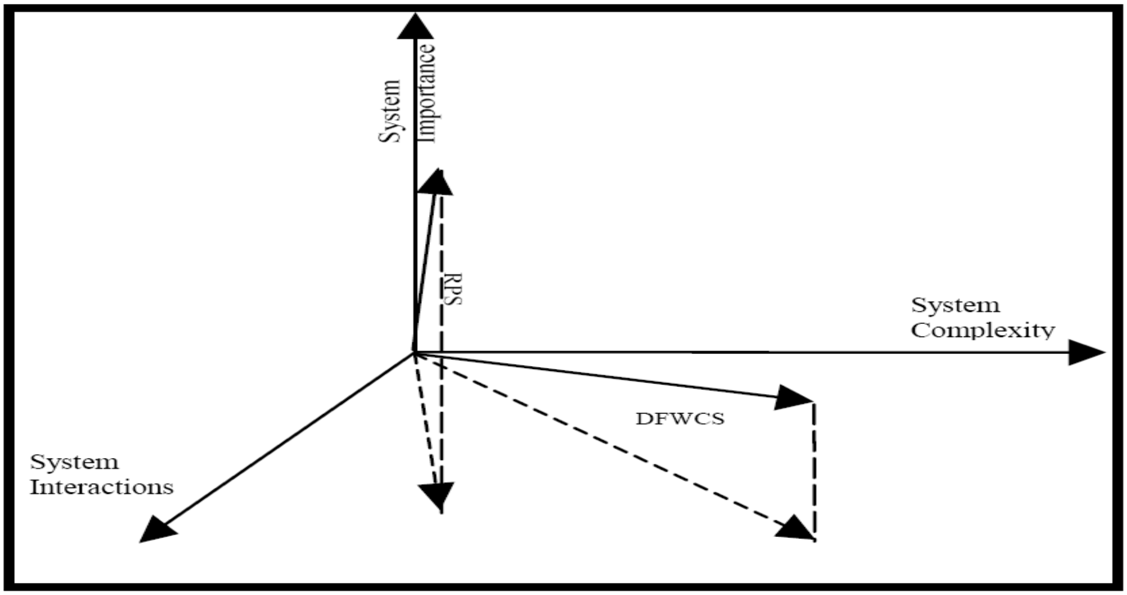


Fig. 4.6. Conceptual classification of digital I&C systems.

Page intentionally blank

5. TAXONOMY FOR CCF VULNERABILITY AND MITIGATION

The basis for a CCF taxonomy has been developed through determination of consistent terminology and establishment of a classification approach. The terminology is based on definitions from standards, guides, and relevant nuclear power industry technical reports. The classification approach is derived from identified classification schemes focused on I&C systems and key characteristics, including failure modes. The CCF taxonomy provides the basis for a systematic organization of key systems aspects relevant to analyzing the potential for CCF vulnerability and the suitability of mitigation techniques. Development of an effective CCF taxonomy will help to provide a framework for establishing the objective analysis and assessment capabilities desired to facilitate a rigorous identification of fault types and triggers that are the fundamental elements of CCF.

The fundamental terminology for characterizing failure vulnerabilities of an I&C system* involves the source of the vulnerability, the instantiation of the vulnerability, the means of activating the vulnerability, and the effect of an activated vulnerability.

The cause or source of failure vulnerability can be attributed to a situation or occurrence during design, manufacture, or use. These sources correspond to defects, mistakes and errors. For this CCF taxonomy, the term “error” is used to represent the sources of failure vulnerability as a class and it is taken to mean a defect, mistake, or deficiency associated with the specification, design, manufacture, implementation, installation, operation, or maintenance of an I&C system.

The instantiation of failure vulnerability is a fault. For this CCF taxonomy, a fault is the manifestation of an error that results in an I&C system state that is characterized by inability to perform a required function. It is noted that not all faults result in failure. However, for the purposes of this taxonomy, faults that lead to failure are the main consideration. With the work being focused on CCF, systematic faults are the primary concern. A systematic fault is generally a design fault but may result from any common design, process, or human error occurring throughout the life cycle of the I&C system. Since nuclear power I&C systems are subject to rigorous quality assurance controls throughout their life cycles, the concern here is for latent faults that remain undetected in a system until specific conditions are such that the result produced does not conform to the intended function.

The means of activating a failure vulnerability is a triggering condition or event. For this CCF taxonomy, a trigger is a specific event or operating condition that causes an I&C system to fail due to activation of a latent fault. Triggers include plant transients and initiating events, external conditions (e.g., environment, natural phenomena), interactions among systems, human interaction, and internal states (e.g., execution profile, exception handling). The signal trajectory of an in-service I&C system is considered a primary trigger of failure vulnerabilities. The signal trajectory is the time histories of all equipment conditions, internal states, input signals, and operator inputs which determine the outputs of an I&C system.

The result of an activated fault is a failure. For this CCF taxonomy, a failure is the inability of an I&C system to execute a required function or to continue to perform within previously specified limits. Failures include events that occur when an intended function is not executed on demand or when an unintended function is initiated. A systematic failure is related in a deterministic way to a certain cause, which can only be eliminated by a modification of the design or of the manufacturing process, operational procedures, documentation, or other relevant factors.

The relationship between errors, faults, triggers, and failures is illustrated in Fig. 5.1. The error represents the cause of the failure vulnerability, which is instantiated into the system as a fault. The latent

* In this discussion, *I&C system* is used to refer to the system itself or any subelement (e.g., channel or division, subsystem, module, board, component, software object or service, part, etc.) of the system.

fault, which is undetected through the design life cycle processes, is activated by a condition or event during service that serves as a trigger. The result of the activated fault is a failure of the I&C system. Consequently, evaluation of the potential for failure must consider fault-trigger combinations that lead to failure. Subsequently, a failure may propagate to cause consequential failures of other elements within the system or other interconnected systems. A cascade failure of this type is another form of dependent failure, along with CCF and CMF. Cascade failures are typically treated as part of the single failure analysis for safety-related I&C systems. However, consequential failures are also within the scope of D3 analyses [33].

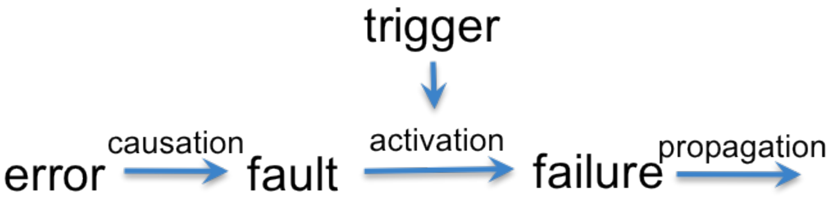


Fig. 5.1. Relationship between vulnerabilities and failures.

In this CCF taxonomy, CCF is the coincidental failure of two or more independent I&C systems due to a single specific cause (i.e., fault-trigger combination). Common latent faults from errors during specification, design, manufacture, implementation, installation, operation or maintenance of I&C systems are the manifestation of CCF vulnerability. Triggers arising from natural phenomenon, plant process operation, human interaction or any internal condition in the I&C system are the specific events that activate CCF. Coincident failures occur when the time interval between the failures is too short for repair measures.

To summarize, a CCF arises when a common fault is present in multiple elements of an I&C system architecture and the occurrence of a trigger (i.e., event or condition) activates that fault in more than one instance of a CCF-susceptible element to result in a coincident failure of a critical function.

Where vulnerability to a CCF is determined or suspected based on a systems analysis (e.g., D3 analysis), the principal responsive approaches are *avoidance* and *mitigation*. In the nuclear power industry, defense-in-depth is employed through provision of independent systems that serve as successive layers or "echelons" of protection to compensate for failures in other systems or functions. In addition, design measures are employed to reduce exposure to anticipated triggers or their concurrent application to multiple systems that may have common faults. Examples of design measures that contribute to defending against CCF vulnerability are discussed in more detail in IAEA Nuclear Energy Series Report No. NP-T-1.5 [11]. However, since absolute avoidance is not generally provable and comprehensive defense-in-depth could be compromised by CCF across echelons, compensating approaches are necessary to address the residual vulnerability. Diversity is the general approach used for addressing perceived vulnerabilities to CCF of I&C system architectures because dissimilarities in technology, function, implementation, and so forth can mitigate the potential for common faults.

Based on the investigation of diversity terminology, this CCF taxonomy adopts the diversity attributes and associated criteria defined in NUREG/CR-6303 as modified in NUREG/CR-7007. Thus, diversity is characterized in terms of the following seven attributes:

- design diversity,
- equipment manufacturer diversity,

- logic processing equipment diversity,
- functional diversity,
- life-cycle diversity,
- logic diversity, and
- signal diversity.

An assessment of the other diversity types identified in this study determined that most were easily covered by the seven primary diversity attributes. However, *history diversity* can be considered as an eighth diversity attribute since it represents a temporal effect not readily subsumed into one of the other diversity attributes. As noted, the research reported by Lindner established that enforced internal time differences through staggered restarts can serve to diversify the execution profiles of software-based systems, thereby reducing the potential impact of platform usage deficiencies (e.g., buffer overwrites, stack overflows, pointer errors, race conditions). Experience in other industries shows that periodic re-initialization of systems and software processes (i.e., software rejuvenation) can address software "aging" [63,64] by refreshing execution cycle history and mitigating accumulated errors to help avoid internal states that could activate faults. This approach is becoming common for high-dependability applications in the financial industry for elements such as online transaction processing systems [65], the telecommunications industry [66], and the internet provider industry for services such as web servers [67].

Nevertheless, it is possible to capture this diversification using two time-related criteria under different diversity attributes. History diversity can be considered as an application of the criteria for a different time scale under functional diversity and for different timing under logic diversity. Consequently, this CCF taxonomy employs the seven primary diversity attributes.

The investigation of classification structures identified several classification frameworks ranging from high-level abstractions to basic descriptive representations. As noted, the primary classification approaches used in the international nuclear industry involve categorization of systems and functions based on safety importance. Any classification approach developed for treating I&C system information and the safety significance of failures (e.g., CCF) must be compatible with the existing safety classes.

As part of this investigation, it was observed that, as the degree of abstraction increased, the tie between constituent classes or categories and the physical aspects of a system becomes less direct. Thus, the classification structures tended to become less intuitive as the level of abstraction increased. Consequently, it was concluded that the classification framework adopted for this taxonomy must include linkage to a more tangible, less abstract structure to be most effective and usable for design analysis and technical review.

It was also observed that a physical representation provides a basis for organizing information about I&C systems while preserving a relationship with the installed systems. However, features and capabilities such as software and human interfaces strongly depend on recognition of function and critical characteristics that cannot be fully represented solely on the basis of a physical model. As seen in the various approaches investigated, a clear understanding of the relationship between abstracted data and the actual I&C system is facilitated by maintaining close correlation between function and the NPP I&C architecture in the organization of information. Additionally, the association of critical characteristics with functional (or physical) groupings can support analysis of system information or operational experience for features important to safety or performance.

The investigation of classification approaches showed that a classification structure that enables physical, functional, and critical characteristic information to be captured and interrelated is most suitable for characterizing CCF vulnerability and mitigation. Accordingly, the adopted classification framework for this CCF taxonomy incorporates a coupling of high-level characteristics related to vulnerability and mitigation with a physical and functional representation of NPP I&C systems within a hierarchical abstraction as a basic structural element.

The classification structure for this CCF taxonomy begins with a whole-part abstraction in which an I&C system is modeled as a group of related components at several levels of physical aggregation. For high-order (i.e., coarse granularity) representations of systems, the approach defined in NUREG/CR-6303 for a D3 analysis provides a suitable basis. Basically, the NPP I&C system architecture is decomposed into a block representation. As defined in NUREG/CR-6303, a *block* "is the smallest portion of the system under analysis for which it can be credibly assumed that internal failures, including the effects of software errors, will not propagate to other equipment." Examples of typical blocks provided in NUREG/CR-6303 are "computers, local area networks, multiplexers, or PLCs." If finer granularity is needed for analysis, the architecture can be decomposed successively into systems, divisions or channels, cabinets or units, modules, and basic components. Decomposition of a digital system into hierarchical layers (e.g., CPU, operating system, basic service software, application software) serves to focus consideration of diversity and other design measures by relating relevant types of CCF (i.e., latent systematic faults and failure-triggering conditions) with specific elements susceptible to the occurrence or propagation of a failure. While software represents an abstraction of functions and tasks, its instantiation can be decomposed into application, runtime, and support services elements, which can be further decomposed into blocks, objects, specific services, etc. However, there should be some tie between the hardware and the software elements (e.g., host properties capture in what hardware element the software resides).

The second element of the classification structure involves an organizing principle for latent faults and triggers that can help to focus analysis of vulnerabilities and compensating design features. This CCF taxonomy adapts the approach to fault classification by Lala and Harper. In their approach, CMF (i.e., CCF) are classified according to nature, origin, and persistence. For this CCF taxonomy, these classes are extended for definition in terms of system aspects to promote a tie to the physical whole-part representation. Basically, the classes of characteristics associated with vulnerability established for this CCF taxonomy are purpose, process, product, and performance (i.e., "4P" class structure). *Purpose* is embodied in the functional requirements satisfied by a specific system. *Process* involves the life-cycle activities at each phase of the system lifetime (e.g., design, development, implementation, installation, operation and maintenance). *Product* consists of the implemented system, including the platform, support services, application software (or complex hardwired logic), interconnections, and distributed elements (e.g., communication nodes, power supplies, sensors, data acquisition and signal conditioning modules, logic elements, actuators). *Performance* includes the behavior of the system and its response to inputs and external factors or events.

The system aspects of CCF mitigation related to purpose and process concern sources by which systematic faults (e.g., flaws, deficiencies, misunderstandings, mistakes, errors, defects) are introduced. These fault sources include requirements, design concepts/system specifications, components and parts, and manufacturing lines, as well as human contributors and tool sets at various life-cycle phases. The product aspect of CCF mitigation is exemplified by the realized systems, including the platforms and applications in which latent faults reside until activated to cause a failure. The location of any common faults may involve the hardware, system software or basic processing elements, application software or logic, integrated hardware/software environment, and/or interconnections (e.g., communication, power, structure). The behavioral aspect of CCF mitigation that concerns performance includes execution of functions and responses to external influences. Execution primarily relates to demands (i.e., inputs) and processing mechanisms (e.g., internal states and state transitions) that can trigger activation of systematic faults or introduce commonalities of condition. Similar response to external influences (e.g., environment or human action) may also serve as triggering mechanisms for common failure.

The impact and benefits of diversity attributes and their associated criteria can be identified in terms of common fault sources (purpose and process), location of vulnerabilities (product), and common triggers (performance). Essentially, the effect of each diversity attribute is characterized according to the resultant capability to minimize the introduction of common faults, mitigate the presence of corresponding vulnerabilities, manage commonality in usage (i.e., execution), and reduce similarity in susceptibility to external factors. Analysis of diversity effects can then be expressed in terms of minimized prospects for common systematic faults, reduced occurrence of concurrent execution profiles, and/or lessened likelihood of similar responses to external influences

Design diversity can impact the process, product, and performance aspects of mitigating CCF vulnerability. The impact on process can be attributed to the prospective effect of technology differences on the sources of systematic faults (e.g., errors) that may arise during the design and implementation of systems. The impact on product can relate to technology-driven differences in the structure and constituent components of systems that may reduce the likelihood of similar architectural locations for vulnerabilities. The impact on performance can involve action, timing, and dynamic response differences that may lead to different execution of function and dissimilar effects from external stress.

Equipment manufacturer diversity primarily impacts the process and product aspects of mitigating CCF vulnerability. The impact on process relates to the prospective effect from use of different resources (e.g., components, manufacturing lines, humans) on the sources of systematic faults (e.g., defects) in the manufacture and supply of systems. The impact on product involves the differences that may arise from the use of different equipment, which may also provide a performance impact via different responses to external influences. At a minimum, equipment manufacturer diversity can reduce potential CCF vulnerability resulting from common or identical equipment.

Logic processing equipment diversity impacts the process, product, and performance aspects of mitigating CCF vulnerability. The impact on process can be attributed to the prospective effect of architectural differences for logic processing on the sources of systematic faults (e.g., errors) that may arise during the design and implementation of systems. The impact on product involves susceptibility differences that may arise from the use of different processing elements or components and from the platform difference that may be present at the macro-architectural level. The impact on performance is related to dissimilarity in the mechanisms of processing that can lead to differences in execution profiles.

Functional diversity impacts the purpose, process, and performance aspects of mitigating CCF vulnerability. The impact on purpose clearly relates to differences in objectives, functional relationships, and computational interactions associated with different functions and can help address the potential for common CCF vulnerabilities resulting from flawed requirements. The impact on process can be attributed to the prospective effect of functional requirement differences on the sources of systematic faults (e.g., misunderstandings, mistakes, or errors) that may arise during the design and implementation of systems. The impact on performance can arise from differences in execution profile that can result from the application of different functionality.

Life-cycle diversity impacts the process, product, and performance aspects of mitigating CCF vulnerability. The impact on process involves the prospective effect on potential sources of systematic error due to variations in cognition and action by different personnel engaged in design, implementation, and installation activities. The impact on product may result from the different development approaches, tool and skill sets, and resource (e.g., personnel and/or capabilities) availability that can differentiate each implementation. The impact on performance can be attributed to human actions that may act as possible triggers (i.e., unanticipated actions) or potential common in-situ fault sources (e.g., maintenance errors).

Logic diversity impacts the process, product, and performance aspects of mitigating CCF vulnerability. The impact on process can be attributed to the prospective effect of differences in the means and form of functional instantiation on the sources of systematic faults (e.g., mistakes or errors) that may arise during the design and implementation of systems. The impact on product relates to prospective differences in the realization of logic (e.g., program) for each application and in the support services provided by each platform. These differences can reduce the potential for latent faults in common elements that may result in CCF vulnerability. The impact on performance includes differences in logic processing mechanisms and functional interactions that can minimize the potential for faulted states to be triggered concurrently due to commonalities in execution profile.

Signal diversity impacts the purpose and performance aspects of mitigating CCF vulnerability. The impact on purpose can arise from the availability of diverse indicators for initiation of protective action, coupled with the associated diverse underlying functional relationships, and can help address the potential for common CCF vulnerabilities due to flawed requirements. The impact on performance relates to differences in execution profile that can result from the presentation of different signal trajectories to diverse systems. An impact on product is also provided in the sense that different sensors are generally involved in achieving signal diversity.

6. REFERENCES

1. U.S. Nuclear Regulatory Commission, “Summary Of March 28-29, 2006, EPRI and NEI Workshop on Digital Instrumentation and Controls (I&C) and Control Room Licensing Issues,” Washington, D.C., March 7, 2007 (Agencywide Documents Access and Management System [ADAMS] Accession Number ML070590059).

2. R. T. Wood *et al*., “Update on Common-Cause Failure Experience and Mitigation Practices,” ORNL/TM-2013/563, December 2013.

3. A. Avizienis *et al*., “Basic Concepts and Taxonomy of Dependable and Secure Computing,” *IEEE Transactions on Dependable and Secure Computing*, Vol. 1(1), 2004, pp. 11-33.

4. A. Avizienis, J.-C. Laprie and B. Randell, “Dependability and Its Threats: A Taxonomy,” *Building the Information Society*, International Federation for Information Processing, Vol. 156, Toulouse, France, August 2004, pp. 91-120.

5. B. Beizer, “Bug Taxonomy and Statistics,” Appendix, *Software Testing Techniques*, 2nd ed., Van Nostrand Reinhold, New York, New York, 1990.

6. T.-L. Chu and M. Yue, “A Comparison of Taxonomies of Digital System Failure Modes,” *Proc. 11th International Probabilistic Safety Assessment & Management Conference (PSAM 11)*, Helsinki, Finland, June 2012.

7. B. Li *et al*., “Integrating Software into PRA: A Software-Related Failure Mode Taxonomy,” *Risk Analysis*, Vol. 26, No. 4, 2006.

8. Nuclear Energy Agency, Committee on the Safety of Nuclear Installations, “Failure Modes Taxonomy for Reliability Assessment of Digital I&C,” NEA/CSNI/R(2014)16, February 2015.

9. Institute of Electrical and Electronics Engineers, “The Authoritative Dictionary of IEEE Standards Terms: 7th ed.,” IEEE Std. 100, Piscataway, New Jersey, 2000.

10. International Electrotechnical Commission, “International electrotechnical vocabulary – Part 192: Dependability,” IEC 60050-192, Geneva, Switzerland, 2015.

11. International Atomic Energy Agency, *Protecting Against Common Cause Failures in Digital I&C Systems of Nuclear Power Plants*, IAEA Nuclear Energy Series No. NP-T-1.5, Vienna, Austria, 2009.

12. International Atomic Energy Agency, *Core Knowledge on Instrumentation and Control Systems in Nuclear Power Plants*, IAEA Nuclear Energy Series No. NP-T-3.12, Vienna, Austria, 2011.

13. International Atomic Energy Agency, “Advanced Control Systems to Improve Nuclear Power Plant Reliability and Efficiency,” IAEA-TECDOC-392, Vienna, Austria, 2011.

14. International Electrotechnical Commission, “Functional safety of electrical/electronic/ programmable electronic safety-related systems – Part 4: Definitions and abbreviations,” IEC 61508-4, Geneva, Switzerland, 2010.

15. International Electrotechnical Commission, “Nuclear power plants—Instrumentation and control for systems important to safety—General requirements for systems,” IEC 61513, Geneva, Switzerland, 2011.

16. International Organization for Standardization, “Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation process,” ISO/IEC 25040, Geneva, Switzerland, 2011.

17. International Organization for Standardization, "Systems and software engineering — Vocabulary," ISO/IEC/IEEE 24765, Generva, Switzerland, 2010.

18. International Atomic Energy Agency, *IAEA Safety Glossary*, 2007 Edition, Vienna, Austria, 2007.

19. International Electrotechnical Commission, "Nuclear Power Plants—Instrumentation and Control Systems Important to Safety—Requirements to Cope with Common Cause Failure (CCF)," IEC 62340, Geneva, Switzerland, 2007.

20. International Electrotechnical Commission, "Nuclear Power Plants—Instrumentation and Control Systems Important to Safety—Software Aspects for Computer-Based Systems Performing Category A Functions," IEC 60880, Ed. 2.0, Geneva, Switzerland, 2006.

21. P. Humphreyes and B. Johnston, *Dependent Failure Procedure Guide*, SRD-R-418, Atomic Energy Authority, Safety and Reliability Directorate, United Kingdom, March 1987.

22. International Atomic Energy Agency, *Radiation Aspects of Design for Nuclear Power Plants*, IAEA S-G-1.3, Vienna, Austria, 2005.

23. U.S. Nuclear Regulatory Commission, *A Defense-in-Depth and Diversity Assessment of the RESAR-414 Integrated Protection System*, NUREG-0493, March 1979.

24. U.S. Nuclear Regulatory Commission, *Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems*, NUREG/CR-6303, December 1994.

25. U.S. Nuclear Regulatory Commission, "Guidance for Evaluation of Diversity and Defense-in-Depth in Digital Computer-Based Instrumentation and Control Systems," Branch Technical Position 7-19, Washington, D.C., 2012 (ADAMS Accession No. ML110550791).

26. U.S. Nuclear Regulatory Commission, *Common Cause Failure Data Collection and Analysis System, vol. 1*, NUREG/CR-6268, Rev. 1, September 2007.

27. Electric Power Research Institute, "Guideline for Performing Defense-in-Depth and Diversity Assessments for Digital Upgrades: Applying Risk-Informed and Deterministic Methods," EPRI 1002835, Palo Alto, California, December 2004.

28. U.S. Nuclear Regulatory Commission, *U.S. Code of Federal Regulations*, Title 10, Part 50, "Domestic Licensing of Production and Utilization Facilities," Washington, DC.

29. U.S. Nuclear Regulatory Commission, *A Defense-in-Depth and Diversity Assessment of the RESAR-414 Integrated Protection System*, NUREG-0493, March 1979.

30. U.S. Nuclear Regulatory Commission, "Policy, Technical, and Licensing Issues Pertaining to Evolutionary and Advanced Light-Water Reactor (ALWR) Designs," Staff Requirements Memorandum on SECY-93-087, Washington, D.C., July 21, 1993 (ADAMS Accession No. ML003708056).

31. U.S. Nuclear Regulatory Commission, "Policy, Technical, and Licensing Issues Pertaining to Evolutionary and Advanced Light-Water Reactor (ALWR) Designs," SECY-93-087, Washington, D.C., April 2, 1993 (ADAMS Accession No. ML003708021).

32. U.S. Nuclear Regulatory Commission, *Standard Review Plan for the Review of Safety Analysis Reports for Nuclear Power Plants, Instrumentation and Controls*, NUREG-0800, Chapter 7, rev. 5, Washington, D.C., 2007.

33. U.S. Nuclear Regulatory Commission, "Guidance for Evaluation of Diversity and Defense-in-Depth in Digital Computer-Based Instrumentation and Control Systems," Branch Technical Position 7-19, rev. 6, Washington, D.C., 2012 (ADAMS Accession No. ML110550791).

34. U.S. Nuclear Regulatory Commission, *Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems*, NUREG/CR-6303, December 1994.

35. U.S. Nuclear Regulatory Commission, *Diversity Strategies for Nuclear Power Plant Instrumentation and Control Systems*, NUREG/CR-7007, February 2010.

36. Society of Automotive Engineers, "Certification Considerations for Highly-Integrated or Complex Aircraft Systems," SAE ARP 4754, SAE International, Warrendale, Pennsylvania, 1996.

37. Center for Chemical Process Safety, *Guidelines for Safe Automation of Chemical Processes*, American Institute of Chemical Engineers, New York, New York, 1993.

38. Center for Chemical Process Safety, *Guidelines for Safe and Reliable Instrumented Protective Systems*, American Institute of Chemical Engineers, New York, New York, 2007.

39. B. Littlewood *et al*., "DISPO Project at City University," Centre for Software Reliability, City University, London, United Kingdom, 2006.

40. A. Lindner, "CCF due to Software – A Contribution to the Actual Discussion," *Meeting Record for IAEA Technical Meeting on Avoiding Common-Cause Failures in Digital Instrumentation and Control Systems of Nuclear Power Plants*, Bethesda, Maryland, June 2007.

41. M. J. Hawthorne and D. E. Perry, "Applying Design Diversity to Aspects of System Architectures and Deployment Configurations to Enhance System Dependability," in *Proceedings of the International Conference on Dependable Systems and Networks*, Institute of Electrical and Electronics Engineers, Florence, Italy, June 2004.

42. Institute of Electrical and Electronics Engineers, "IEEE Standard for Qualifying Class 1E Equipment for Nuclear Power Generating Stations," IEEE 323-2003, Piscataway, New Jersey, 2003.

43. International Atomic Energy Agency, *Instrumentation and Control Systems Important to Safety in Nuclear Power Plants*, IAEA NS-G-1.3, Vienna, Austria, 2002.

44. International Electrotechnical Commission, "Nuclear Power Plants—Instrumentation and Control Systems Important for Safety—Classification," ed. 2.0, IEC 61226, Geneva, Switzerland, 2005.

45. Don Dudenhoeffer *et al*., *Technology Roadmap: Instrumentation, Control, and Human Machine Interface to Support DOE Advanced Nuclear Power Plant Programs*, INL/EXT-06-11862, Idaho National Laboratory, Idaho Falls, Idaho, March 2007.

46. John O'Hara, Bill Gunther, and Gerardo Martinez-Guridi, *The Effects of Degraded Digital Instrumentation and Control Systems on Human-System Interfaces and Operator Performance: HFE Review Guidance and Technical Basis*, BNL-91047-2010, Brookhaven National Laboratory, Upton, New York, February 2010.

47. W. P. Poore III *et al.*, *Sequence Coding and Search System Quality Assurance Program*, ORNL/NOAC-225, Rev. 3, Oak Ridge National Laboratory, Oak Ridge, Tennessee, September 1991.

48. Electric Power Research Institute, "Programmable Logic Controller Qualification Guidelines for Nuclear Applications, Volume 1," EPRI TR-103699, Vol. 1, October 1994.

49. Electric Power Research Institute, "Programmable Logic Controller Qualification Guidelines for Nuclear Applications, Volume 2," EPRI TR-103699, Vol. 2, October 1994.

50. J. Rasmussen, *Information Processing and Human-Machine Interaction: An Approach to Cognitive Engineering*, Elsevier Science Inc., New York, 1986.

51. J. Rasmussen *et al.*, *Cognitive Systems Engineering*, John Wiley & Sons, New York, 1994.

52. G. Weinberg, *An Introduction to General Systems Thinking*, John Wiley & Sons, New York, 1975.

53. N. Leveson, "A New Accident Model for Engineering Safer Systems," *Safety Science*, **42**(4), pp. 237–270 (2004).

54. J. C. Laprie, "Dependable Computing: Concepts, Limits Challenges," *Proceedings of the 25th IEEE International Symposium on Fault-Tolerant Computing—Special Issue*, IEEE, Pasadena, California, pp. 42–54 (1995).

55. J. H. Lala and R. E. Harper, "Architectural Principles for Safety-Critical Real-Time Applications," *Proceedings of the IEEE*, **82**(1), pp. 25–40 (1994).

56. Electric Power Research Institute, "Operating Experience Insights on Common-Cause Failures in Digital Instrumentation and Control Systems," EPRI 1016731, Palo Alto, California, December 2008.

57. U.S. Nuclear Regulatory Commission, *Common-Cause Failure Database and Analysis System: Event Definition and Classification, Vol. 2*, NUREG/CR-6268, Washington, D.C., June 1998.

58. U.S. Nuclear Regulatory Commission, "Assessment of Digital System Operating Experience Data and System Inventory and Classification Structure," unnumbered memorandum, Washington, D.C., March 2008 (ADAMS Accession Number ML0805903832).

59. S. A. Arndt, "Development of Regulatory Guidance for Risk-Informing Digital System Reviews," *5th International Topical Meeting on Nuclear Plant Instrumentation Control and Human Machine Interface Technology (NPIC & HMIT 2006)*, American Nuclear Society, Albuquerque, New Mexico, November 2006.

60. J. Rushby, "Critical System Properties: Survey and Taxonomy," *Reliability Engineering and System Safety,* **43**, pp. 189–219 (1994).

61. C. Perrow, *Normal Accidents: Living with High Risk Technologies*, Basic Books, New York, 1984.

62. U.S. Nuclear Regulatory Commission, *Current State of Reliability Modeling Methodologies for Digital Systems and Their Acceptance Criteria for Nuclear Power Plant Assessments*, NUREG/CR-6901, Washington, D.C., February 2006.

63. Y. Huang, C. Kintala, N. Kolettis, and N. D. Fulton, "Software Rejuvenation: Analysis, Module and Applications," *Twenty-Fifth International Symposium on Fault-Tolerant Computing (FTCS-25)*, Pasadena, California, June 1995, p. 381.

64. V. Castelli *et al.*, "Proactive management of software aging," *IBM Journal of Research and Development*, **45**(2), March 2001.

65. K. J. Cassidy, K. C. Gross, and A. Malekpour, "Advanced pattern recognition for detection of complex software aging phenomena in online transaction processing servers," *Proc. of Int'l Conf. on Dependable Systems and Networks (DSN 2002)*, June 2002.

66. A. Avritzer and E. J. Weyuker, "Monitoring Smoothly Degrading Systems for Increased Dependability," *Empirical Software Engineering*, **2**(1), March 1997.

67. M. Grottke *et al.*, "Analysis of Software Aging in a Web Server," *IEEE Transactions on Reliability*, 55(3), September 2006.