CIFE CENTER FOR INTEGRATED FACILITY ENGINEERING

Quantitative Method for Analyzing Engineering Defect Risks in Novel Projects

By

John Chachere

CIFE Working Paper #WP119
April 2009

STANFORD UNIVERSITY

If you would like to contact the authors, please write to:

c/o CIFE, Civil and Environmental Engineering Dept.,
Stanford University
The Jerry Yang & Akiko Yamazaki Environment & Energy Building
473 Via Ortega, Room 292, Mail Code: 4020
Stanford, CA 94305-4020

QUANTITATIVE METHOD FOR ANALYZING ENGINEERING DEFECT RISKS IN NOVEL PROJECTS

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF

MANAGEMENT SCIENCE AND ENGINEERING

AND THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

John Marvin Chachere

November 2007

Abstract

Novel projects, such as those producing manned space missions, new cancer drugs, and unique civil facilities, require difficult decisions that tradeoff the costs of development reliability against the risks of operations failure. Yet, no social science or engineering method is both precise and holistic enough to estimate quantitatively the risks of engineering defects for specific projects where product, organization, process, and context strongly interact. To address this gap, the thesis provides a model of engineering defects as a source of critical dependencies between novel projects' upstream development and downstream operations stages. The thesis method elicits quantitative judgments from project experts regarding different engineering defects' causes during knowledge-based development and those defects' consequences during physical operations. With those data, the thesis models development-stage shortcomings as a function of failures to complete necessary rework, interprets those shortcomings to assess distributions of engineering defect severities, and estimates those defects' potential to reduce the developed product's capacities during operations. The thesis uses a project analysis framework, PRA-VDT, which integrates the model of engineering defects with the existing Virtual Design Team (VDT) simulation of development organizations and processes, the Probabilistic Risk Analysis (PRA) model of product functions and operating contexts, and the Decision Analysis (DA) method of rational decision support. In PRA-VDT, the Defect Model translates VDT output (defects' causes) into PRA input (defects' consequences), thus enabling the framework to formally explain relationships between diverse project features (such as component redundancy, engineering defects, and developer backlog) typically addressed by separate theories. The thesis finally presents PRA-VDT analyses of a hypothetical satellite project and of the Stanford Green Dorm Project as evidence that, compared with standalone models, the new framework can more holistically evaluate a broad range of alternative plans for novel projects.

Acknowledgments

I would foremost like to thank my wife, Yvonne Chachere, for the endless understanding, support, love, and inspiration that made this research possible. As well as the pleasure of life itself, I owe my academic interest and aptitude to my parents Joan and Marvin Chachere. I owe my appreciation of learning also to many great teachers at the Berkeley Unified School District (Alan Lura), UC Santa Cruz (David Haussler), and Stanford University (Clifford Nass).

I am deeply grateful for the rare privilege of conducting this research under the advising of Professor M. Elisabeth Paté-Cornell. Within the Engineering Risk Research Group, I owe thanks also to Lea Deleris, Jimmy Benjamin, Russ Garber, Jan Benjamin Pietzsch, and Pauk Kucik III. Professor Ron Howard's wise Socratic teaching and mentorship have deeply inspired my personal, professional, and academic lives. Within the communities of Professor Howard's Decision Analysis Working Group and Decision and Ethics Center, I am particularly grateful to Thomas Seyller, Somik Raja, Christopher Han, and John Cavallaro. At the Department of Management Science and Engineering, outstanding work by several individuals made my scholarship possible: Lori Cottle, Robert Carlson, John Weyant, and Tina Seelig.

I cannot overstate my gratitude for the consistent academic and personal support of Dr. John Kunz and his wife Lynn, and of Professor Ray Levitt and his wife Kathleen. I am also especially thankful to Professor John Haymaker for his friendship and deep collaboration regarding the dissertation process, industry, and Green Dorm Project. Also from the Center for Integrated Facility Engineering community, I appreciate the collaboration and support of Teddie Guenzer, Martin Fischer, Yan Jin, Ryan Orr, Renate Fruchter, Jack Chang, Marc Ramsey, Claudio Mourgues, Peggy Ho, Mauricio Toledo, Kathleen Liston, Timo Hartmann, Douglas MacKinnon, Tobias Maile, Tore Christiansen, Gaye Oralkan, Jolin Salazar, and Geoff Cohen, among many others.

From the Center for Design Research, I appreciate the Defense Chairmanship of Larry Leifer and many inspiring talks with Ade Mabogunje.

The Green Dorm illustration in this thesis would not have been possible without help from the project team and other Civil Engineering experts, particularly Laura Goldstein, Allan Daly, Brad Jacobson, Forest Flager, Caroline Clevenger, Jennifer Tobias, Ken Azzollini, Frank Azzollini, and Mauricio Toledo. Several aerospace experts, notably David Bergner, Judith Orasanu, Brandon Owens, Douglas Osheroff, Rebecca Wheeer, Bob Oberto, and Ted Sweetser, inspired the Satellite illustration.

I would finally like to acknowledge the sponsors of my Stanford research and teaching: the Kozmetsky Research Fellowship Program and the Stanford Media-X Center, the Management Science and Engineering Department, the CIFE Seed Research Program, and the NASA Ames Engineering for Complex Systems program.

Table of Contents

List of Tables

List of Illustrations

Chapter 1
Purpose and Scope

The thesis provides a quantitative model to support managerial decisions for projects in which engineering defects created during a development stage can cause failure during an operations stage.

> *Attempting to manage high-risk technologies while minimizing failures is an extraordinary challenge. By their nature, these complex technologies are intricate, with many interrelated parts. Standing alone, the components may be well understood and have failure modes that can be anticipated. Yet when these components are integrated into a larger system, unanticipated interactions can occur that lead to catastrophic outcomes.* ***The risk of these complex systems is increased when they are produced and operated by complex organizations that also break down in unanticipated ways.***
>
> *–NASA Columbia Accident Investigation Board, Final Report (NASA 2003)*
> *bold face added*

Novel development efforts require decisions that trade off the costs of development reliability against the risks of operations failure. As examples of costly, strategic risk-reduction investments in industry, space exploration strengthens mission assurance using component redundancies; pharmaceutical development verifies safety and efficacy using repeated trials; and novel civil construction builds physical and virtual prototypes. At a tactical level, these industries' managers and engineers must routinely choose whether to sacrifice product reliability by ignoring design rework, to overrun schedule by slipping deadlines, or to break the budget by working overtime.

The decisions can be exceptionally difficult because they affect the uncertain and dynamic development organization and process as well as the operating product and

context. Failed space missions, pharmaceutical toxicities, and premature building failures have brought shortcomings in these decision processes into the public eye.

These decision processes benefit from a diverse range of applicable social science insights, management methods, and engineering tools (see Chapter 2). Yet, no social science or engineering method is both precise and holistic enough to estimate the significance of potential engineering defect risks for specific projects where the design and development organization and process strongly interact with the operational product and context. The lack of adequate engineering defect risk assessment methods can affect strategic choices (for example) between reducing risk by investing in developer skill or physical system reinforcement.

The thesis addresses the need for better engineering defect risk assessment methods in three ways. The thesis identifies mechanisms by which management choices influence engineering process quality, specifically the failure to complete necessary rework, thereby creating engineering defects that can jeopardize downstream operational success. The thesis provides a quantitative model for assessing the degree to which these phenomena are likely to manifest for a given project. The thesis also presents that defect model within a framework of decision support methods to compare alternative plans for projects having engineering defect risks.

The thesis contribution (to the field of project modeling and optimization) is a quantitative model that views engineering defects as a source of critical dependencies between novel projects' upstream development stages and downstream operations stages. The thesis method elicits quantitative judgments from project experts regarding different engineering defects' causes in knowledge-based development and those defects' consequences in physical operations. Using those data, the thesis models development-stage shortcomings as a function of failures to complete necessary rework, interprets those shortcomings to assess distributions of engineering defects, and estimates those defects' potential to reduce the developed product's reliability during operations.

Consider Figure 1.1 Macro-Level Decision Diagram of PRA-VDT Project Model (below). The PRA-VDT framework uses the existing Virtual Design Team (VDT) method to assess the distribution of development-stage outcomes based on the organization and process of design. This illustration shows that the Defect Model (introduced in the thesis) estimates the distributions of various kinds of defects in building features that could result from ignored design exceptions. The Defect Model then uses those defect distributions to inform a probabilistic risk analysis (PRA) of electrical systems' failure rates during operations. The rates of failures in different parts of the electrical system compose the project utility, which Decision Analysis (DA) uses to compare different alternatives' merits. In PRA-VDT, the Defect Model translates the nuanced VDT output (defects' causes) into PRA input (defects' consequences), thus enabling the framework to formally explain relationships between diverse project features (such as component redundancy, engineering defects, and developer backlog) typically addressed by separate theories. By translating the complex data faithfully from VDT output to PRA input, the Defect Model enables analysis and decision making based on interdependent product, organization, process, and context factors.

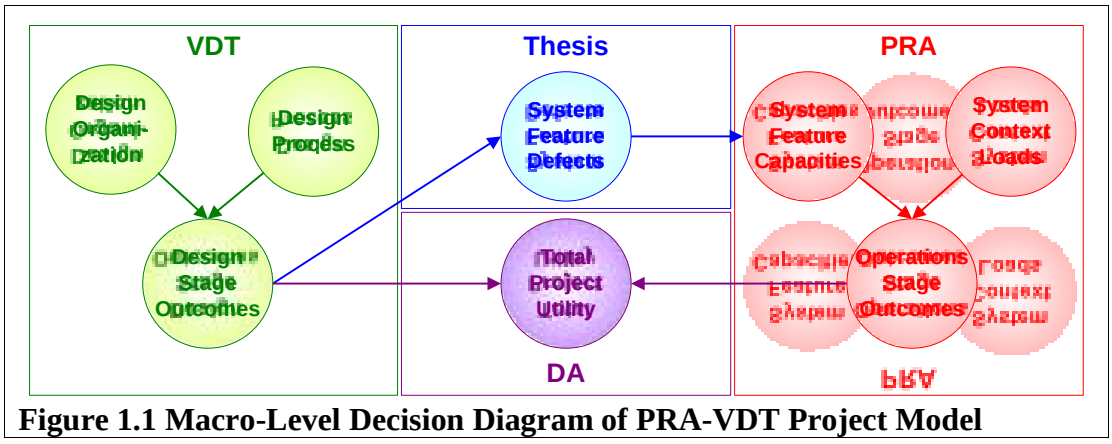
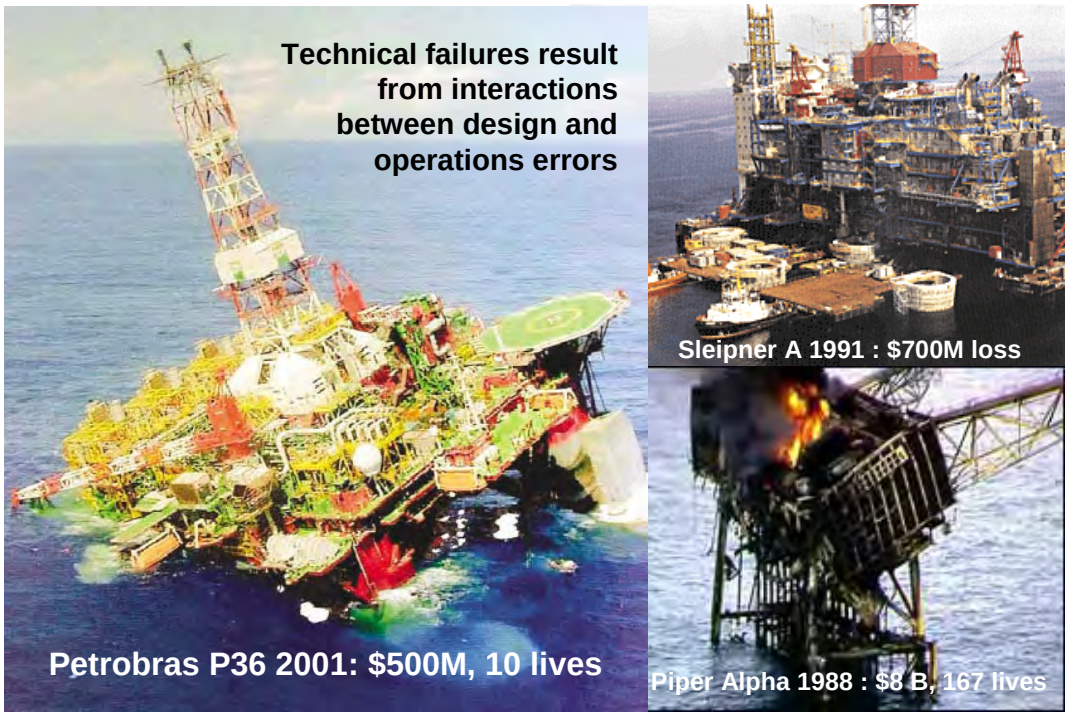


Figure 1.1 Macro-Level Decision Diagram of PRA-VDT Project Model

The thesis presents illustrative PRA-VDT analyses, of a satellite project and of the Stanford Green Dorm Project, that answer difficult questions using transparent, theory- and field data- based reasoning.

1.1 Practical Problem

Novel projects often develop defective products that unexpectedly contribute to the partial or even total failure of operations.



Petrobras has established new global benchmarks for the generation of exceptional shareholder wealth through an aggressive and innovative program of cost cutting on its P36 production facility. Conventional constraints have been successfully challenged and replaced with new paradigms appropriate to the globalized corporate market place. Through an integrated network of facilitated workshops, the project successfully rejected the established constricting and negative influences of prescriptive engineering, onerous quality requirements, and outdated concepts of inspection and client control. Elimination of these unnecessary straitjackets has empowered the project's suppliers and contractors to propose highly economical solutions, with the win-win bonus of enhanced profitability margins for themselves. The P36 platform shows the shape of things to come in unregulated global market economy of the 21st Century. *-Petrobras Official [1]*

Figure 1.2 Engineering Defects Contributed to Several Unexpected, Catastrophic Oil Platform Failures

Many of humanity's most ambitious endeavors, such as space missions and pharmaceutical development, require complex and interdependent design and development efforts culminating in operations that are at risk of failure. In many industries, these projects result in operational failure far more frequently than competent and careful human planners assess.

Catastrophic failure often occurs during operations because of engineering defects introduced during design and development. This thesis expands upon the intuition that in novel projects, the probability of failure during operations depends on the planning and management of engineering activities early in the project. The web of interdependencies among early phase engineering activities and operations failures is complex, dynamic, and uncertain, and is the subject of this thesis.

For example, Figure 1.2 (on page 4) illustrates three offshore oil platforms that failed catastrophically when an alternate design would have withstood the accident. In 1982, all 84 crew members of the Ocean Ranger platform perished due to (in part, a design allowing) the unchecked escalation of failure in a single port light [USCG 1982]. Many design flaws contributed to the 1988 Piper Alpha disaster [Paté-Cornell 1993], in which 167 lives were lost, and for which an $8B cost estimate accounts for new industry design requirements. In 1991, Sleipner A caused a $700M loss and a 3.0 magnitude seismic event when a concrete base structure "failed as a result of a combination of a serious error in the finite element analysis and insufficient anchorage of the reinforcement in a critical zone" [Selby et al 1997]. As recently as 2001, "Many of the same factors that resulted in the NASA Challenger and Columbia accidents were present in the Petrobras P36 accident" [Bea 2003], which cost $500M and 11 lives. The next section details the critical features of those NASA accidents that this thesis addresses.

1.2 Management-Ignored Warnings from Engineering in Aerospace

The Columbia and Challenger disasters illustrate dynamics between engineers and their managers that critically affect operations risks.

Thorough investigations into the 1986 loss of Space Shuttle Challenger and 2003 loss of Columbia revealed how disasters can result from interplay between engineering work and management decision making [Rogers Commission 1986, NASA 1995,

NASA 2003, Vaughn 1996]. This section explains how an organizational and procedural context that persists today shaped critical decisions leading to catastrophic shuttle failures. Based on this motivation, the thesis provides a model to help managers assess the risk of defect-caused accidents and weigh the prospect of minimizing them by focusing resources on unmet engineering rework demands.

[Former NASA Administrator] Goldin was also instrumental in gaining acceptance of the "faster, better, cheaper" approach to the planning of robotic missions and downsizing … In 1994 he told an audience at the Jet Propulsion Laboratory, "When I ask for the budget to be cut, I'm told it's going to impact safety on the Space Shuttle … I think that's a bunch of crap"

- NASA Columbia Accident Investigation Board, Final Report (NASA 2003)

Figure 1.3 Engineering Defects Unexpectedly Induced Two Catastrophic Space Shuttle Failures

Figure 1.3 (above) illustrates the subjects of this section: two well-studied space shuttle disasters in which engineers under time pressure believed (but could not prove) defects might cause a critical failure, and in which management decided to continue status quo operations in spite of that uncertainty. Specifically, Morton-Thiokol's failure during a gate decision to properly interpret known limitations to the range of design analysis (specifically, the temperature at launch) led directly to the Challenger Disaster [Rogers Commission 1986]. Just before the Columbia Disaster, NASA and Boeing engineers' uncertainty regarding the significance of assumptions fundamental to a foam strike analysis (specifically, the size of the striking body) led to management's decision to ignore the need for further investigation. In both cases, management decided to ignore engineers' concerns when reworking the engineering

task might have prevented the disasters. Additional examples of critical decisions to ignore rework demands in the Columbia case include the declassification of foam strikes as a "safety of flight" issue, and management's denial of engineers' requests for imagery of the shuttle [NASA 2003].

Conflicting Institutional Goals

This decision making process requires tradeoffs between conflicting goals in the face of engineering uncertainty:

> *When a program agrees to spend less money or accelerate a schedule beyond what the engineers and program managers think is reasonable, a small amount of overall risk is added…. Little pieces of risk add up until managers are no longer aware of the total program risk, and are, in fact, gambling.*
>
> *–NASA 2003*

The National Aeronautics and Space Administration (NASA) workforce lives by the credo "Failure is not an option" [Kranz 2000] and strives also to be "Faster, better, cheaper" [NASA 2003]. These noble goals frequently conflict, however, and NASA has traced many of its technical failures to honorable choices in the service of one goal, to the detriment of another. As the following quote shows, the shuttle disasters highlight the risks of failing to adequately integrate these goals into effective organizational decision making:

> *NASA managers believed that the agency had a strong safety culture, but the Board found that the agency had the same conflicting goals that it did before Challenger … goals of cost, schedule, and safety.*
>
> *– NASA 2003*

The conflict between cost, schedule, and safety goals manifests routinely in management decision making. In many engineering organizations, managers review any requests to extend schedule in order to conduct rework. The decision to rework an item generally costs money and/or time, while the decision to ignore warranted rework increases the risk that an engineered subsystem will fail.

> *At seven different times in the Shuttle Program, NASA and Thiokol managers made poor technical decisions that ultimately permitted continued flight of an unsafe Solid Rocket Motor design [including] failure to accept John Miller's recommendations to redesign the clevis joint.*
>
> *–NASA 2003*

Organizational Effects on Decision-Making

Engineers often have immediate incentives to keep within cost and schedule limits, but have few incentives regarding risk because failures typically manifest long after the work finishes [Daly 2006]. The following quotations provide examples of engineering sources of risk in the Columbia and Challenger projects:

> *Instead of conducting [rework], NASA engineers qualified the flight design configuration … using extrapolated test data and redesign specifications … due to these testing deficiencies, the board recognized that bolt catchers could have played a role in damaging Columbia's left wing.*
>
> *–NASA 2003*
>
> *The faulty solid rocket motor joint and seal must be changed. This could be a new design eliminating the joint or a redesign of the current joint and seal.*
>
> *–Rogers Commission 1986*

Managers typically have limited engineering knowledge to bear on the problem and rely instead upon engineers' reports and on assessments of schedule and cost. The following quotations provide examples of these information flows at NASA:

> *Communication did not flow effectively up to or down from Program managers.*
>
> *Managers at the top were dependent on engineers at the bottom for their engineering analysis and risk assessments. Information was lost as engineering risk analyses moved through the process. At succeeding stages, management awareness of anomalies, and therefore risks, was reduced either because of the need to be increasingly brief and concise as all the parts of the system came together, or because of the need to produce consensus decisions at each level.*

> *In perhaps the ultimate example of engineering concerns not making their way upstream, Challenger astronauts were told that the cold temperature was not a problem, and Columbia astronauts were told that the foam strike was not a problem.*
>
> *–NASA 2003*

As information travels through the hierarchy, uncertainty such as that bearing upon risk tends to become "absorbed" so that managers can make more straightforward and timely decisions [March and Simon 1958]. The loss of information regarding uncertainties can harm any decision's quality, but particularly harms risk analyses that fundamentally address rare events. The following quotes further explain how uncertainty absorption and the "Normalization of deviance" [Vaughn 1996] contributed to managers' decisions to ignore critical rework needs at NASA.

> *NASA's blind spot is it believes it has a strong safety culture.*
>
> *A pattern of acceptance prevailed throughout the organization that tolerated foam problems without sufficient engineering justification for doing so.*
>
> *Ignored by management was the qualitative data that the engineering teams did have: both instances were outside the experience base.*
>
> *Their presentation included the Crater analysis, which they reported as incomplete and uncertain. However, the Mission Evaluation Room manager perceived the Boeing analysis as rigorous and quantitative.*
>
> *Management focused on the answer – that analysis proved there was no safety-of-flight issue – rather than concerns about the large uncertainties that may have undermined the analysis that provided that answer … "The analysis is not complete. There is one case yet that they wish to run, but kind of just jumping to the conclusion of all that… thermal analysis does not indicate that there is potential for a burn-through."*
>
> *NASA's views of its safety culture in those briefings did not reflect reality.*
>
> *–NASA 2003*

Challenges Posed by Necessary Change

The following quotes regard NASA engineers' historic inability to override critical management decisions:

Worried engineers in 1986 and again in 2003 found it impossible to reverse the Flight Readiness Review risk assessments that foam and O-rings did not pose safety-of-flight concerns.

[In 1986] When [a senior vice president who seldom participated in these engineering discussions] told the managers present to "Take off your engineering hat and put on your management hat," they reversed the position their own engineers had taken

[In 2003] Rocha … did not want to jump the chain of command. Having already raised the need to have the Orbiter imaged … he would defer to management's judgment on obtaining imagery.

Allegiance to hierarchy and procedure had replaced deference to NASA engineers' technical expertise.

–NASA 2003

These examples make clear that the twin challenges of uncertainty absorption and normalization of deviance require an outside assessment and adjustment of the engineering defect-linked development and operations. The following quotes show that NASA investigations determined that reducing the risk of further disasters requires adjusting the organization to support engineering challenges with appropriate management decision making:

The foam debris hit was not the single cause of the Columbia accident, just as the failure of the joint seal that permitted O-ring erosion was not the single cause of Challenger. Both Columbia and Challenger were lost also because of the failure of NASA's organizational system.

Flawed practices embedded in NASA's organizational system continued for 20 years and made substantial contributions to both accidents ... For all its cutting-edge technologies, "diving-catch" rescues, and imaginative plans for the technology and the future of space exploration, NASA has shown very little understanding of the inner workings of its own organization…

Changes in organizational structure should be made only with careful consideration of their effect on the system and their possible unintended consequences.

NASA's challenge is to design systems that maximize the clarity of signals, amplify weak signals so they can be tracked, and account for missing signals.

–NASA 2003

This thesis addresses the needs of organizations like NASA to become not only "faster, better, cheaper," but also reliable, sustainable, and more, without "operating too close to too many margins" [NASA 2003]. The thesis research directly focuses on those dynamics that NASA investigations revealed as contributing to the shuttle disasters. The thesis models the dynamics this section introduced by contributing a model of engineering defects that links an existing model of engineers' performance and management decision making (VDT) to another existing model of risks during the developed system's operations (PRA).

1.3 Existing Methods

To make decisions affecting the design and operations stages of novel projects, managers typically rely on limited experience bases, qualitative theories, and quantitative models that simplistically address the stages independently.

Contemporary project management practice includes important tools for analyzing difficult practical decisions involving specific project risks, costs, schedule, and other objectives. These tools typically address a limited range of project elements, however, such as product design quality or schedule tracking, and often lack methods to address the most difficult decisions systematically. Intuitively, research in decision and risk analysis, and in social psychology and organization theory, suggests that walking the surface of Mars will not require the *complete* mastery of *separate* organization, process, product, and context factors. Instead, taking humanity's next steps will require the identification, assessment, and management of *uncertainties* regarding *interacting* strengths and weaknesses in those four factors.

NASA has devoted tremendous resources to risk management and accident investigations, and they have consistently traced downstream errors in operations to precursors in upstream development and design [Bergner 2005] as well as operations and maintenance. The fraction of major system failures that can be traced to human

and organizational shortcomings is estimated to range from fifty to ninety percent [Paté-Cornell 1990, Murphy and Paté-Cornell 1996].

Table 1.1 Intuition Behind Contributions to Engineering Defect Risks

Table 1.1a

Development Stage		**Organization (Capacity)** Strong	Weak
Process (Load)	Easy	Low Risk	Medium Risk
	Difficult	Medium Risk	High Risk

Table 1.1b

Operations Stage		**Product (Capacity)** Robust	Fragile
Context (Load)	Safe	Low Risk	Medium Risk
	Hazardous	Medium Risk	High Risk

Table 1.1c

Total Project Failure Risk		**Operations Stage** Low Risk	Medium Risk	High Risk
Development Stage	Low Risk	Lowest Risk	Lower Risk	Medium Risk
	Medium Risk	Lower Risk	Medium Risk	Higher Risk
	High Risk	Medium Risk	Higher Risk	Highest Risk

Consider Table 1.1 Intuition Behind Contributions to Engineering Defect Risks (above). Total failure risk often depends on risks introduced during development stages and on risks from operations. Development risks loosely depend in magnitude on the ability of the organization to execute its assigned process, and operations risks generally result from the robustness of a product relative to its operating environment. Qualitatively, the greatest risks most often (but not always) occur where weaknesses of organization, process, product, and context confound one another. This thesis

presents methods to analyze these dynamics quantitatively and in detail for real-world projects.

The thesis builds upon the intuition, presented in Table 1.1 (a, b, and c), that effective planners must assess four factors: how robust their operational processes are under different circumstances; the actual performance and properties of the products they are building (such as a spacecraft and its support systems); the organizations that conduct the mission during its operational phase; and the context in which the product and project operate. Of equal importance, planners must assess the possible interactions among the four factors and must translate these assessments into action.

The thesis refines the intuition of Table 1.1c by providing a quantitative model of engineering defects, a key component of the relationship between operations and development risks. In response to these phenomena, social science and engineering researchers have developed rich theories of collaborative activities and their relationships to risk (Most notably Bem et al 1965, Perrow 1986, Roberts 1990; For literature reviews see Ciaverelli 2003 and Cooke et al 2003). Often, these theories offer only limited benefit to difficult practical decisions, however, because the literature defines them qualitatively; Many of these theories offer insufficient detail to real-world decision makers who need to precisely evaluate such a theory's: range of valid application, potential interactions across disciplines and over time in complex projects, and likelihood of accuracy when in apparent contradiction with other theories.

Lacking this quantitative definition, modern planners today find little research providing precise analysis of common but difficult practical decisions involving specific project risks, costs, schedules, and other objectives. Exceptions include quantitative programmatic and risk models developed by engineers (Most notably Paté-Cornell 1990, Paté-Cornell and Fischbeck 1993.1 and 1993.2, Murphy and Paté-Cornell 1996, Paté-Cornell et al 1996, and Dillon and Paté-Cornell 2001), but the

thesis (in Chapter 2) argues that increasing these tools' levels of integration with organizational concerns can further improve decision making.

NASA has used Probabilistic Risk Analysis (PRA) to quantitatively estimate the failure probabilities of complex engineered systems [Paté-Cornell and Fischbeck 1993.1, 1993.2], and is continuing to apply the technology on the International Space Station. Because PRA does not provide a specific model of the project's upstream engineering organizations or processes, however, it cannot estimate the influence these factors will have on risk.

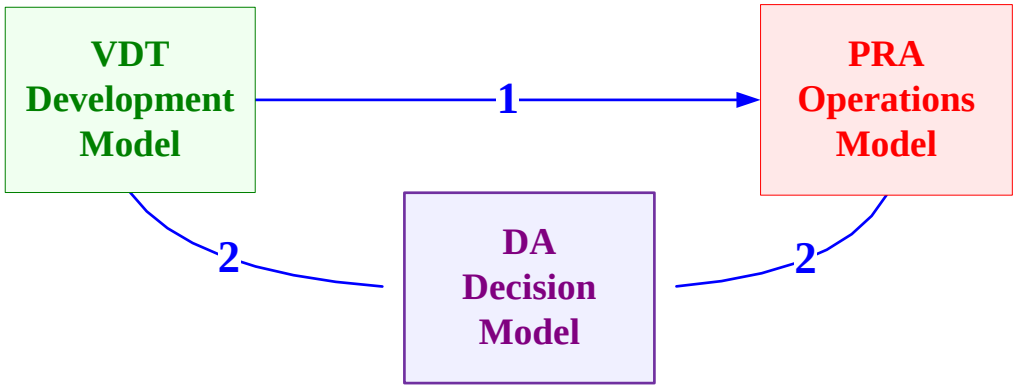
NASA has also used the Virtual Design Team (VDT) simulation to quantitatively assess the behaviors of engineering organizations and processes, including many behaviors that are associated with risk [Kunz et al., 1998; Levitt et al 1999]. Because VDT does not provide an explicit model of products or their operating contexts, however, it cannot assess the impacts these behaviors will have on the probability of failure in operations.

Chapter 2 further explains the degree to which existing methods already address the practical problems outlined in this section. The next section provides the intuition that a new model of engineering defects, which is the thesis contribution, can enable integrating several existing methods and improving their support to project modeling and optimization.

1.4 Research Scope

The thesis contributes a new, quantitative model of engineering defects that enables the integration of existing Virtual Design Team, Probabilistic Risk Analysis, and Decision Analysis models into a project planning decision support tool.

Research Question 1: What is a model that quantitatively describes the dependencies that engineering defects create between a project's development-stage knowledge work and its operations-stage physical outcomes?



Research Question 2: What is a method of using the engineering defect model to compare and choose between alternative projects that have both a development stage (consisting of knowledge work with the capacity to produce defects) and an operations stage (consisting of physical processes)?

Figure 1.4 Research Questions

Figure 1.4 illustrates how the observed problem and existing models motivate two research questions:

1. *What is a model that quantitatively describes the dependencies that engineering defects create between a project's development-stage knowledge work and its operations-stage physical outcomes?*

The thesis addresses the first research question by providing a computational model of defects having enabling causes in early project stage engineering process quality and resultant consequences in later stage operations risks. The Defect Model defines the

modeling assumptions and reasoning steps and relies upon field inquiry to provide data for a specific project.

2. *What is a method of using the engineering defect model to compare and choose between alternative projects that have both a development stage (consisting of knowledge work with the capacity to produce defects) and an operations stage (including physical processes)?*

The thesis addresses the second research question by defining interfaces between the model of engineering defects and several existing models (VDT, PRA, and DA) to form a multi-stage project modeling framework called PRA-VDT.

The thesis model shows how development organizations lacking the resources to meet process demands tend to create more defective product elements. During operations, these defective elements reduce the capacities of engineered systems, and make both partial and total failures more likely.

In the context of PRA-VDT, the thesis model operationalizes the following intuition. Highly competent engineering organizations (those with adequate information processing capacity) both recognize the need for rework to remediate design risks and dedicate the time that is necessary to perform it. The additional time and attention spent on engineering processes tends to increase the fraction of engineering flaws that are discovered and rectified, thus improving the probability that the resulting product will conform to the specification. The more engineering tasks are conducted in this way, the fewer the defects and the greater the probability that the final product will meet the project objectives during operations. In contrast, less competent engineering organizations will have more mishandled and ignored exceptions, which elevates the risk of introducing engineering flaws into the product, and which probabilistically leads to a less conforming and more error-prone result.

Real project dynamics, and management decision making, are more complex than the preceding intuition suggests. Every decision cascades to affect behavior through time and across the organization. For example, a plumbing design team's decision to shift the location of a water line (rework) can require that electrical cable be rerouted (rework), thus burdening the electrical team. In addition, delaying task completion to rework a component can increase schedule pressure on later, dependent tasks, which will then have less time available to complete their own work. This "systems view" of interactions among products, organizations, and processes recognizes that decisions regarding one development task affect other development tasks as well as downstream development and operations. Analyzing development and operations according to the systems view is difficult, but using a quantitative model can help.

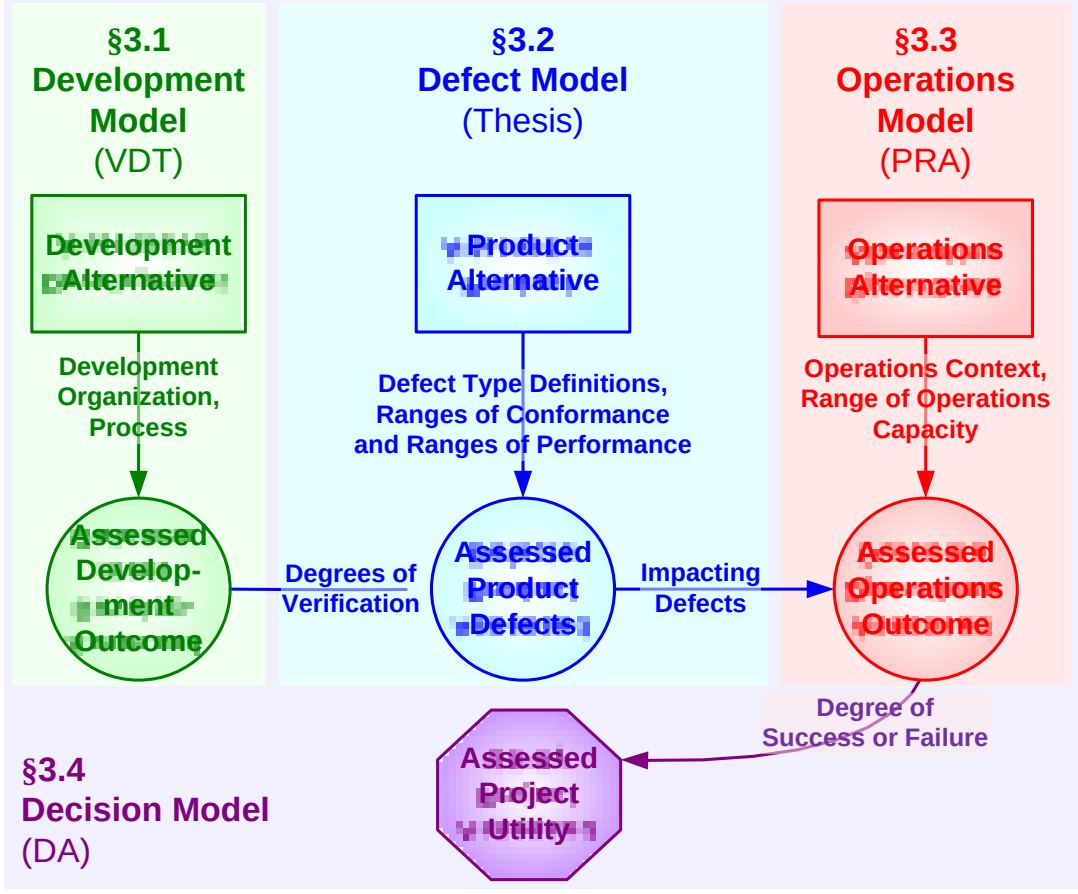


Figure 1.5 Decision Diagram Overview of the PRA-VDT Framework

The thesis representation's principal entities and relationships appear in Figure 1.5 Decision Diagram Overview of the PRA-VDT Framework (above). The PRA-VDT

Framework provides a theoretically founded method to help project managers synthesize the assessments of four models that each focus on one aspect of project behavior. The integrated model includes a VDT model and simulation of product development (e.g., apartment building design), the thesis definition of conformance probability and defects (e.g., structural flaw severities), a PRA analysis of impacted operations (e.g., distribution of earthquake losses), and a DA assessment of benefits to the decision maker (e.g., expected lifetime of the building). The figure shows how the thesis quantitatively represents, and interrelates, both upstream engineering organization- and process-contingencies and possible downstream product and context factors. The PRA-VDT framework connects a Virtual Design Team model of development-stage knowledge work (§3.1), through the Defect Model of engineering defects (§3.2), to a Probabilistic Risk Analysis (§3.3) model of operations involving physical processes, and uses a Decision Analysis formulation of decision making (defined in §3.4) to assess the results. Project Managers aim to maximize project utility (purple) by choosing the best possible combination of development, product, and operations alternatives (green, blue, and red respectively). Their decision is difficult because the choices affect diverse project behaviors that interact in complex ways. Chapter 3 formally defines and intuitively illustrates the PRA-VDT method.

Quantitative analysis can support decision-making in the face of difficult tradeoffs. For example, consider the decision of whether to employ a redundant telecommunications array on the Huygens probe [JPL 2000]. Figure 2.3 provides a PRA fault tree and functional block diagram that the analysis could use. If PRA alone were to assess the two antennae failing in a probabilistically independent manner, redundancy would improve the project failure risk (note that in a detailed real-world analysis, PRA likely would identify sources of engineering or contextual failure dependency). VDT, on the other hand, might assess that designing the redundant system would require a more complex and uncertain processes than designing the simple system; therefore it would have greater process quality risk than designing a simple system.

PRA-VDT can analyze the Huygens example by using the change in degree of verification that engineering difficulty produces, and then estimating the updated failure risk based on the component redundancy. In the model, redundancy increases the spacecraft's design complexity, which raises the probability of engineering flaws causing components to fail. However, the model assesses whether the payload's functional redundancy more than compensates, raising the total success probability above that of the "single string" case; the alternative with redundancy could be more likely to complete the mission, even though it is more likely to suffer component failures.

Chapter 4 and Chapter 5 provide detailed illustrations of engineering defect risks in more complex, satellite and dormitory projects.

1.5 Thesis Outline

The remainder of the thesis reviews existing theory and practice, formally defines the new model and framework, describes two illustrative applications, and draws explicit conclusions.

The remainder of the thesis has the following outline:

Chapter 2 Existing Practice and Theory (starting on page 21) provides the background of literature and state of the art. The chapter explains that to make decisions affecting both development and operations stages, managers typically rely on limited experience bases and quantitative models that simplistically address the stages independently. The chapter describes strengths and weaknesses in current project planning and management practices in the fields this research addresses. The chapter reviews the slate of academic results available to address practical needs, and to explain how each only partially addresses the targeted project risks.

Chapter 3 Model Definition (starting on page 56) defines the thesis contribution, a model of engineering defects linking explicit causes, namely

unverified development-stage knowledge work, to corresponding consequences, namely elevated risks of operations-stage failure.

Chapter 4 Satellite Illustration (starting on page 110) illustrates the thesis contributions by using PRA-VDT to model a hypothetical satellite project and to weigh trade-offs between redundancy and complexity.

Chapter 5 Green Dorm Field Study (starting on page 138) illustrates the thesis contributions by using PRA-VDT to model risks of electrical system failure in a proposed Stanford dormitory that is traditional, one that demonstrates sustainability technologies, and one that both demonstrates and tests sustainability technologies.

Chapter 6 Conclusion (starting on page 174) discusses the contributions to theory and practice, the model's justification, and avenues for related research.

Chapter 2
Existing Practice and Theory

To make decisions affecting both development and operations stages, managers typically rely on limited experience bases, informal qualitative theories, and quantitative models. The existing methods independently address limited parts of the posed problem.

This chapter describes five fields of existing research and practical methods that partially address the questions Chapter 1 raised. Those fields are Project Management, decision and Risk Analysis, Organizational Theory, Computational Organizational Modeling, and Project Modeling and Optimization. Within each field, the chapter briefly describes several specific methods' capacities and limitations regarding the stated problem. The following paragraphs introduce the five fields, notably leaving literature citations for the referenced sections that follow.

Project Management (§2.1 on page 24) commonly addresses the observed problem with important but limited methods. In particular, Stage-Gate Project Management, which reduces project shaping complexity by decomposing projects into sequences of smaller, more manageable parts, serves as a fundamental practical point of departure for the thesis. The Critical Path Method assists task scheduling, and Multi-Attribute Collective Decision Assistance assists product configuration, but neither method formally or adequately considers their subjects' interdependence with other critical project features such as organizational design and failure risk.

Decision and Risk Analysis (§2.2 on page 32) can help reduce the chance of project failures through the formal identification, assessment, and mitigation of risks. Failure Mode Effects Analysis provides a standard for rapidly communicating project and program risks' approximate likelihood, consequences, and resulting severity. §2.2 drills down on Probabilistic Risk Analysis, a method of quantifying risks that serves a critical role in the thesis by providing a model of project operations behaviors. Decision Analysis facilitates the comparison of projects with diverse merits and detractions. The System-Action-Management method describes how management policy decisions influence human actions that, in turn, influence engineered systems' performance (this important principle is a foundation of the thesis method). WPAM, designed for nuclear plant safety assessment, adjusts failure probability formulae to account for probabilistic dependencies between component failures resulting from shared organizational and procedural elements. However, WPAM lacks a theory-based model of organizational behavior, and the method relies solely on human experts to assess the extent of those dependencies. Decision and Risk Analysis provides no specific organization theory-based method of determining the impacts that knowledge work might have on operations failure probability.

Organizational Theory and Social Psychology (§2.3 on page 40) offers many insights that inform decision-makers' views of risk to the products of knowledge work. Safety Culture identifies human decision makers' attitudes as critical in determining risk severity, and that attitude depends upon organizational context. Normal Accident Theory views the most complex and interdependent products as beyond human understanding and therefore naturally prone to unanticipated failure modes. High Reliability Organizations Theory suggests that strategies such as organizational and procedural redundancy can reduce these risks. This chapter reviews in particular detail the Information Processing View, which assesses specific dynamics outlined in Chapter 1's problem statement, because the thesis builds upon that model of organizational decision making as a point of departure. None of these organizational theories, as yet, provides a measure of risk that is precise enough to support difficult real-world project shaping decisions.

Computational Organizational Models (§2.4 on page 44) formally operationalizes organizational theories in ways that often can support organizational design decisions. The Organizational Consultant provides a method of determining the strength of theoretic recommendations about an organization, but applies to a firm that operates over a long time, rather than to aerospace and construction projects that involve a collaboration of limited scope. Interaction Value Analysis distinguishes certain organizations as naturally able to perform efficiently, but its results apply over long time periods rather than over an individual project's span. The section provides detail about the Virtual Design Team, a quantitative model of routine design work that operationalizes the information processing view. None of these models explicitly considers physical processes or product risks, so they can not conclusively address project shaping decisions where these risks are central.

Project Optimization Models (§2.5 on page 51) assist decisions that shape projects having both a significant possibility of cost and duration overruns during a development phase, as well as a significant risk of failure during an operations stage. ARPAM determines the optimal allocation of budget resources between physically reinforceable product components and a contingency available to hedge against development setbacks. APRAM does not consider the direct influence that development process quality can have on operations failure risk. The Exception-Detection Model, the most advanced of Carlo Pugnetti's 1997 Ph.D. thesis models, explicitly models the role of development process in that relationship. However, the Exception-Detection model does not consider the impacts that development organizations' limitations have on the execution of those development processes, which is critical factor in projects having complexity near the limits of human ability.

2.1 Project Management

The addressed projects' managers typically make decisions using a stage-gate project structure to manage complexity (sometimes simplistically) and using analysis methods focused on product or process (only).

2.1.1 Stage-Gate Project Management

Stage-gate projects consist of a sequence of short segments called stages that each involve qualitatively different activities [NASA 1995]. Between each pair of stages lies a gate milestone in which the results from one stage inform subsequent major decisions. For example, gates typically include a "Go/no-go" decision between committing resources for the next stage and aborting the project. This thesis addresses large projects with stage-gate processes (including construction, aerospace, consumer products, and software development). The thesis does not address projects that use other strategies, such as the spiral method (which is common in small to medium-sized software and consumer product development).

Stages

Table 2.1 describes a typical sequence of stages. The project plan divides work into a sequence of segments called stages that each aims to achieve a well-defined subgoal. A typical project begins with a specification of project goals and overall strategy. Design stage activities elaborate and instantiate this specification into a blueprint for an organization and process that can achieve those goals. Development stage activities translate the design blueprint into one or more physical artifacts that will enter an operations stage.

Table 2.1 Example Project Stages

Stage	Description
Specification	A creative, senior team translates an initial concept into a set of goals the project will aim to satisfy.
Design	A collaborative team or engineers and stakeholders translates

Table 2.1 Example Project Stages	
Stage	**Description**
	project goals into specific recommendations for the organizations, processes, and products of development and operations.
Development	A physical artifact is built based on the design product. In the space industry, a physical spacecraft, ground systems, and set of formal procedures are created, tested and packaged for operations. In the Architecture, Engineering, and Construction industry, this is the construction stage.
Operations	The developed product is operated (space mission executed, or the building occupied) during this stage, and each of its functions may be called upon to ensure success. The robustness of the design and development efforts, along with the operating context, will determine the extent of their successes.

Each stage culminates in the delivery of a product, which consists of all the work that can significantly influence behavior later in the project, and which excludes intermediate results. For example, the product of a conceptual design stage is a specification that guides the project's next stage, detailed design. Most "conceptual design" products exclude work generated for the evaluation of design alternatives that were determined to be unworkable.

Gates

Typically, each pair of adjacent stages is divided by a gate—a step in which management first translates the previous stage's product (compiled results) into an estimation of the future project's prospects, and then makes a "Go/no-go" decision on whether to proceed with later stages. A "No-go" decision typically ends the project. After a "Go" decision, the content of a previous stage's product shapes the next project stage (along with other contextual factors such as budget and the prevailing standards of practice). For example, a spacecraft design that selects nuclear power will require a different development team than one that relies on conventional fuel.

Gates enable managers to understand and control large and novel projects' complexities by decomposing, standardizing, and distributing the decisions to specialized organizations and processes. For example, decision makers often base gate decisions on standardized criteria, such as the degree of confidence in the project's ability to meet a company-wide target for return on investment.

Limits in Addressing the Research Questions

The stage-gate project form manages the complexity of gate decisions by condensing information into standardized deliverables. Organization theory terms the loss of information that occurs when subordinates report only their condensed findings to supervisors as "uncertainty absorption" [March and Simon 1958, Simon 1977]. Uncertainty absorption helps managers to make decisions by controlling complexity, but also hides important and sometimes politically-charged information. When information about the engineering behavior is ignored, decisions about whether to proceed with the next stage can become more tractable (Management science uses analogous "pinch points" to make complex mathematical algorithms tractable).

In particular, standardizing deliverables and decision-making criteria before gates can occlude evidence relevant to the likelihood of flaws in the product that engineering activities produce. §3.1.1 explains several pieces of information, such as developer backlog and skill, that are relevant to risk (and considered by the Defect Model) but typically excluded gate decision making, Well-intentioned managers or accident investigators looking at operations failures have difficulty detecting the root causes of many engineering failures, because there are few records of the conditions that created those flaws.

The thesis Defect Model, which is introduced in Chapter 3, provides a formal model that decision makers can use during gate decision analysis to interpret the volume of ignored developer requests to determine the amount of elevated risk that will affect operations.

2.1.2 The Critical Path Method (CPM)

The Critical Path Method (CPM) defines task durations and their precedence relationships, and then calculates a project schedule that managers can track their progress against [developed by DuPont and Remington Rand; see Walker and Sawyer 1959]. CPM is perhaps the most common way that managers track stage-gate projects, and it underlies the most popular project tracking systems including Microsoft Project and Primavera. Project Evaluation and Review Technique (PERT) extends the method by explicitly incorporating uncertainty in task durations.

Limits in Addressing the Research Questions

> *Although schedule deadlines are an important management tool, those deadlines must be regularly evaluated to ensure that any additional risk incurred to meet the schedule is recognized, understood, and acceptable. - NASA 2003*

Both CPM and PERT rely on managers to assess the nature and extent of interactions between interdependent tasks and teams. For instance, neither CPM nor PERT have a specific model of how coordination requirements limit the effectiveness of schedule compression through fast-tracking. Rules of thumb commonly used in conjunction with CPM can cause dramatically mistaken assessments [Brooks 1995], whereas this thesis aims to assist project shaping by leveraging extant theory on emergent project behavior. Moreover, CPM's simple and clear evaluation of schedule tempts managers to neglect other aspects of productivity, such as quality and risk.

2.1.3 Multi-Attribute Collective Decision Assistance for Design Integration (MACDADI)

Multi-Attribute Collective Decision Assistance for Design Integration (MACDADI) is a research program that addresses conceptual design in the Architecture, Engineering, and Construction industry. The research identifies practical barriers to the adoption of

theoretically applicable management science methods, and documents the delivery of tolerated methods that enhance consensus building [For an academic conference paper see Haymaker and Chachere 2006; for a description from practice see EHDD 2006; the research is not yet published in a peer-reviewed journal].

Multi-attribute decision analysis (§2.2.5 below) [Keeney and Raiffa 1976] provides a theoretically defensible gold standard for multi-attribute decision making, however industry-specific procedures in place today can make practical compromises to that decision-making process attractive. For example, project success typically relies on the sense of meaningful participation by many diverse stakeholders (such as hundreds of community members or potential building occupants). Stakeholders and modelers lack the time to conduct full-blown utility assessments for all those the decision-maker wishes to benefit. MACDADI addresses this challenge by formulating a class of simple utility functions (linear, to date), parameterized for individual respondents using a survey. Figure 2.1 (on page 29) presents one survey used to ascertain the relative strengths of stakeholders' preferences, which are essential to design decision making. MACDADI then combines the survey responses into a total valuation function based on measures of the stakeholders' significance to the decision maker. MACDADI is a decision support method, not a decision making method; MACDADI's use process emphasizes discussion of the formalized data and intermediate figures, rather than the final valuation associated with each alternative, because the method includes significant approximations. Figure 2.2 (on page 30), a chart with aggregated results of the stakeholder survey, provides a map of agreement and discord among the stakeholder groups. In the figure, high values indicate greater importance of a goal compared to lower values. For example, Energy Use was most important overall (total score 33), but on average Graduate School of Business students viewed that measure as one-fourth as important as Stanford University community members (individual score 8 versus 2).

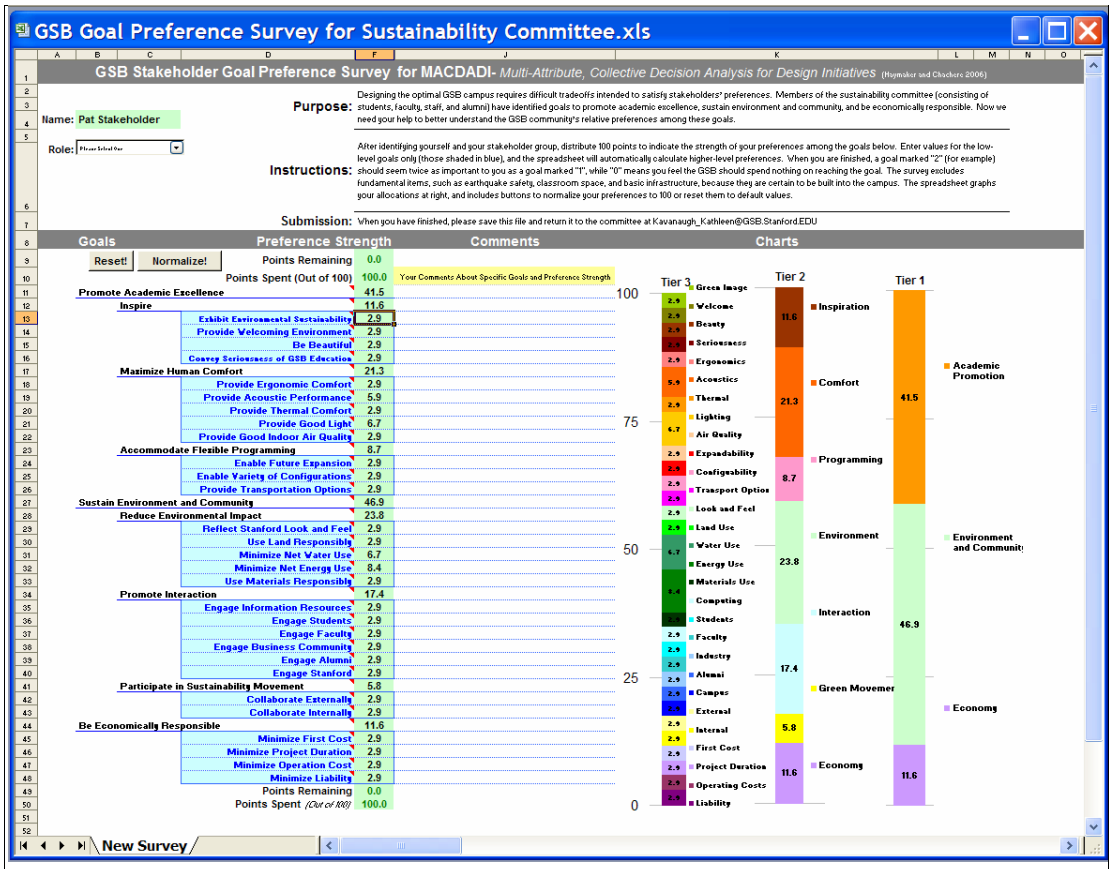


Figure 2.1 Sample MACDADI Survey of Stakeholder Preferences

Courtesy of the Stanford Graduate School of Business Environmental Sustainability Task Force Final Report [GSB 2006]

Figure 2.2 Sample MACDADI Chart Comparing Stakeholder Preferences
Courtesy of the Stanford Graduate School of Business Environmental Sustainability Task Force Final Report [GSB 2006]

Rather than "warranting" that analyses are theoretically optimal, MACDADI claims only to improve decision making processes currently used in conceptual design for the Architecture, Engineering and Construction industry [aiding "decision hygiene", see Howard 1992]. The method involves hazards (such as gaming [Howard 1991]), but may suffice when they can be countered (using, for example, game theory's Revelation Principle [Gibbons 1992]), or when the results serve only to inform an individual decision maker of the impacts on stakeholders' benefits. In addition to notions from multi-attribute decision analysis, MACDADI currently combines notions from Quality Function Deployment (a common systems engineering method, see Hauser and Clausing 1988) and integer programming together with industry-appropriate formulation, use processes, and visualization tools.

The first MACDADI applications, to Stanford University Campus construction projects in the 20 to 200 million-dollar range, began with observations of the difficulties that design teams experienced when communicating their goals, preferences, options, and analyses. In following the MACDADI process guide: the project team collected, synthesized, and hierarchically organized their goals; stakeholders established their relative preferences with respect to these goals (Figure 2.1); the design team formally rated the design options with respect to the goals; and the project team then visualized and assessed the goals (Figure 2.2, on page 30), options, preferences, and analyses to assist in a transparent and formal decision making process.

Limits in Addressing the Research Questions

Existing MACDADI applications have modeled product performance and utility functions as linear, and have excluded uncertainty, thereby limiting the range of specific project decisions for which it is justified. Moreover, MACDADI relies on human experts to assess the performance of developed products; the method has no

model of operating context or development. In contrast, this thesis assesses the limits of how projects sometimes may fail to achieve the intended results.

§5.3 explains how the thesis field study used data collected initially for MACDADI to help assess the behavior of a complex product during operations.

2.2 Decision and Risk Analysis

Decision and Risk Analysis can improve decision making and help reduce the chance of failures, but provide no specific method of assessing the behaviors and consequences of knowledge work.

2.2.1 Failure Mode Effects Analysis (FMEA)

In cases where time is short and data are complex, very rapid analyses that use qualitative or ordinal data can help decision-makers prioritize risks of greatly differing significance. For example, Failure Mode Effects Analysis, or FMEA, helps many developers to chart and communicate the rough probability and consequences of various risk sources [USDOD 1980]. Project shaping then focuses on risks with both high probability and severe consequences. NASA was instrumental in developing FMEA (for the Apollo program) and uses the method currently (e.g., for accelerated conceptual design sessions [Meshkat and Oberto 2004]).

Limits in Addressing the Research Questions

Although methods like FMEA can aid intuition and communication, the thesis goal is to enable decision makers to identify and quantify these risks to support difficult project-specific decisions. These decisions must forecast complex interactions among project elements that FMEA can record, but cannot assess.

2.2.2 Bayesian Probabilities and Influence Diagrams

Influence diagrams progressively refine and decompose complex behavior into clearly distinguished uncertainties that expert testimony, existing statistics, or formal models can characterize [Howard and Matheson 1983, Howard 1988]. This illustration of nodes (representing uncertainties) and arcs (representing conditional probabilistic relevance) enhances intuition regarding the domain of interest at the same time as it encodes mathematically formal statements. Constructing and reasoning about influence diagrams is powerful and subtle, partly due to the method's flexibility. Typically, influence diagrams having many nodes are solved computationally, such as using spreadsheets [Howard 1989.1].

2.2.3 Probabilistic Risk Analysis (PRA)

The Probabilistic Risk Analysis (PRA) method [Barlow and Lambert 1975, USNRC 1980] is commonly used to evaluate systems (such as nuclear power plants [Cornell and Newmark 1978, Garrick and Christie 2002] and the practice of anesthesiology [Paté-Cornell et al 1996]) that contain many interdependent components whose aggregate reliability is difficult to assess. NASA, for example, has used PRA to quantitatively estimate the failure probabilities of complex engineered systems [Paté-Cornell and Fischbeck 1993.1, 1993.2], and is continuing to apply the technology on the International Space Station. PRA methods estimate the probability of failure for a complex product by first conceptually decomposing it into functional blocks, then assessing component- and subsystem- failure rates in the context of uncertain external events, and finally aggregating to a total failure probability. To determine the reliability of basic elements, PRA relies upon methods of field inquiry that include expert assessment, statistics, and mathematical models [Paté-Cornell 2004].

Consider Figure 2.3 Functional Block Diagrams and Corresponding Fault Trees (on page 34). Some complex systems are *single string*, meaning they require all their subsystems to function, while others are *redundant*, meaning they can withstand some

component failures without suffering a total system failure. When a system includes a single-string component with a high failure probability, risk analysts often recommend redesigning it to incorporate redundancy, while recognizing that functions' probabilistic independence determines that strategy's effectiveness. These tools for analyzing reliability form a fundamental element of the PRA-VDT framework used in this thesis.

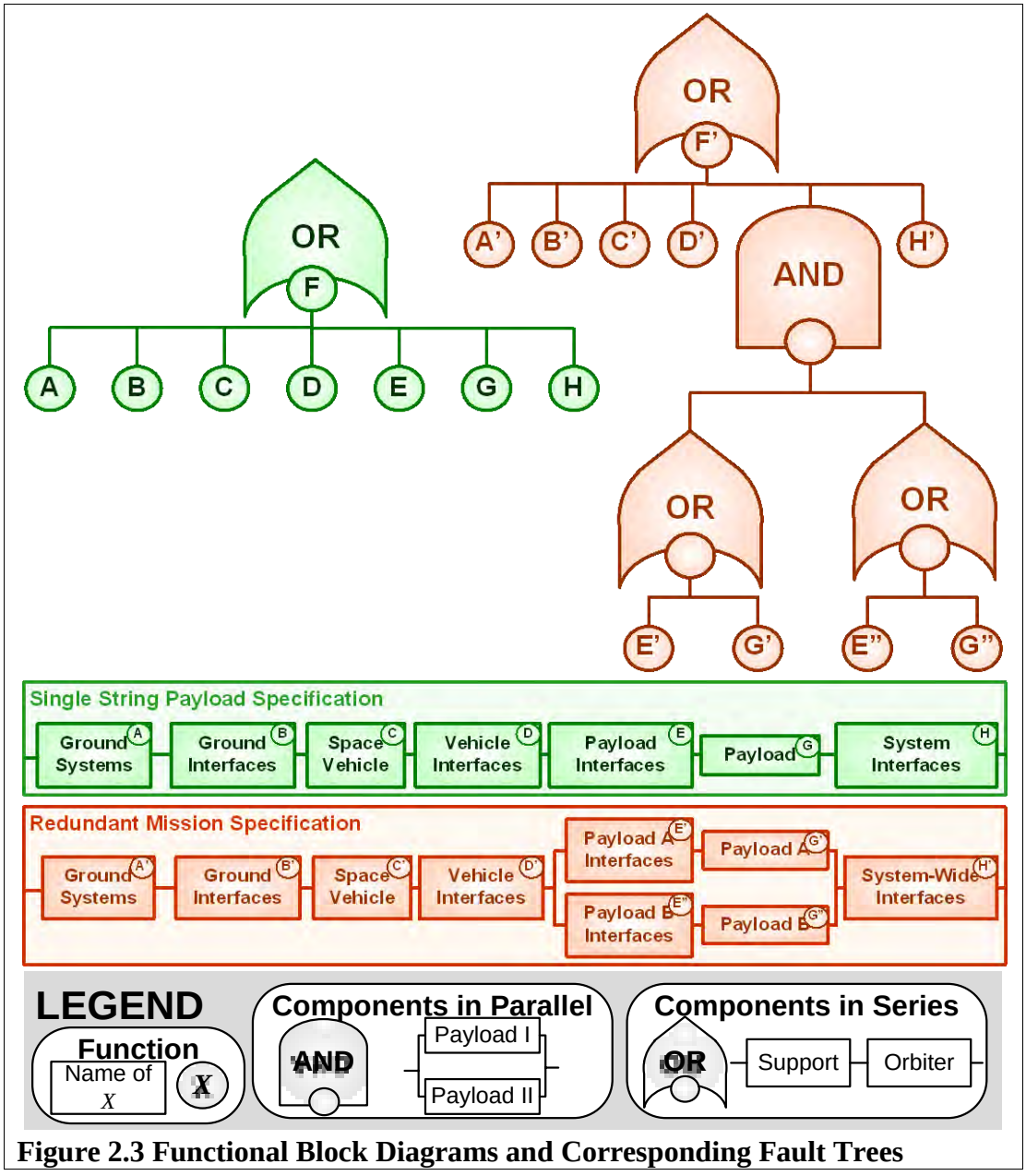


Figure 2.3 Functional Block Diagrams and Corresponding Fault Trees

Justification

Justifying PRA is difficult because the method reasons about events and attributes that are uncertain. From a theoretical perspective, the "prior" probabilities and updating procedures that PRA develops in conjunction with domain experts are often the most hotly debated. The most troublesome subjects for PRA validation are those domains that have no existing statistical data or theoretical foundation, and that are difficult for domain experts to assess. These subjects are, however, the most difficult for other methods as well. In particular, many large projects, such as in aerospace or construction, are exceedingly complex, largely unprecedented, and face a legacy of mixed results with risk management [NASA 2003].

PRA has achieved a level of popular support through strikingly accurate assessments (such as the Columbia space shuttle's thermal protection system failure [Paté-Cornell and Fischbeck 1993.1, 1993.2]) in spite of those barriers to adoption. In some industries, PRA has a strong tradition of application and routinely operates in a prospective mode.

Limits in Addressing the Research Questions

NASA has devoted tremendous resources to risk management and accident investigations, and it has consistently traced downstream errors in operations to upstream development and design precursors [Bergner 2005]. In many projects the upstream design organization and process is a common failure source (external event) that influences all engineered elements; therefore, that failure source is of the utmost importance in estimating failure risk. Because the PRA method does not provide a specific model of development organizations or processes, the method requires significant extensions to estimate the influence these factors will have on operations risks [Davoudian et al 1994.1, Ghosh and Apostolakis 2005]. Furthermore, regardless of their physical relationships, the manifestations of engineering errors during operations are probabilistically dependent on one another because the upstream processes interact in complex ways (Pooled or stronger interdependence, see

Thompson 1967). Fault trees cannot describe the effects of factors upon which all other variables are conditioned [Paté-Cornell 1984.1], including the excessive hastening of upstream development. PRA's accuracy in addressing the posed problem therefore heavily depends upon the existence of reliable information (typically either comparable data or experts' assessment) regarding human and organizational risks.

§3.3 explains how the thesis-enabled framework links PRA with Decision Analysis, the Virtual Design Team, and a new model of engineering defects.

2.2.4 The System-Actions-Management Framework (SAM)

The System-Action-Management method describes how management policy decisions influence human actions that, in turn, influence engineered systems' performance. This important principle is a foundation of the thesis method. The fraction of major system failures that can be traced to human and organizational shortcomings is estimated to range from fifty to ninety percent [Paté-Cornell 1990, Murphy and Paté-Cornell 1996]. The System-Actions-Management framework (SAM) [Murphy and Paté-Cornell 1996] uses PRA to address human and organizational behavior specifically. The method extends analysis first from engineered systems to the actions that affect it, then to the management decisions that influence those actions. The original SAM formulation provides several examples of theory-based models of action, including rational, boundedly rational, rule-based, and execution (under limited effectiveness).

Limits in Addressing the Research Questions

SAM provides a way to formalize a model of organizational behavior and psychology and to integrate it with models of physical processes, but provides no specific theory of knowledge work, such as design. As part of the PRA toolkit, general-purpose tools including Bayesian statistics and decision analysis empower, but do not guide

modelers who must evaluate the myriad social dynamics that affect knowledge work's efficacy. The method also provides little specific guidance for synthesizing social science theory and informant testimony.

2.2.5 Work Process Analysis Model (WPAM)

WPAM [Davoudian et al 1994.1, 1994.2] is a formal risk assessment method designed to incorporate organizational considerations into nuclear plant safety assessment. Taking PRA as a fundamental point of departure (see §2.2.3 on page 33), the method adjusts failure probability formulae to account for probabilistic dependencies between component failures resulting from shared organizational and procedural elements. WPAM first (and rightly) points out that assessments of this type require formal, detailed knowledge of the organization (in WPAM, termed "Working Unit"), process ("Work Process"), defect type ("Candidate Parameter Group"), and operating product ("System/Component Identification"). WPAM next (and rightly) identifies the importance of assessing the strength of interactions between these factors. For those interactions that seem most important (assessed in terms of potential to impact failure probability associated with minimal cut sets), WPAM calculates an adjusted failure probability that accounts for those interactions. As an illustration, the literature on WPAM identifies a diverse range of important factors (e.g., organizational centralization) and their interactions for a specific application to maintenance processes for a nuclear power plant.

Limits in Addressing the Research Questions

Regarding the research question, WPAM lacks a formal model of decision making. Extending WPAM to support decisions (such as using DA), however, would be straightforward and similar to the work presented in this thesis.

WPAM also lacks a theory-based model of organizational behavior, relying directly on human expert judgment to assess complex, subtle, and interacting organizational

and procedural effects on risk. A nonlinear systems view illuminates the complex interacting phenomena targeted in Chapter 1, and this thesis uses VDT to capture that view.

Consider, for example, centralization – the tendency to make decisions at senior (versus junior) levels of the organizational hierarchy. A wealth of theory describes circumstances where centralization can aid, harm, or otherwise influence organizational performance [March 1994]. VDT captures the view that centralization's long-term effect upon organizational behavior depends upon its nonlinear interactions within subtle "system dynamics" influenced by many theory-based factors (such as organizational culture, structure, workload, and uncertain project events like unattended meetings) [Levitt et a 1999].

WPAM is able to model an expert's testimony that two component failures are positively dependent if they are both linked to organizational "Centralization." Unlike VDT (and by extension, the method in this thesis), however, WPAM does not make predictions about the interactions between relevant factors like centralization. Of clear relevance to the study of risk would be the interaction between centralization and an organization's tendency to make risky decisions at higher versus lower levels of the organization.

2.2.6 Decision Analysis (DA)

Decision Analysis (DA) is a method of structured conversation, leading to clarity of action [Ramsey 1931, von Neumann and Morgenstern 1944, Savage 1954, Fishburn 1964, Matheson and Howard 1968]. DA has been applied to systems engineering for decades [Matheson and Howard 1973] via mathematical modeling, and more recently those models have been extended using computers in intelligent decision systems [Holtzman 1985, Howard 1988]. Specifically, the method identifies: a set of *alternatives* among which a decision maker must choose, *information* about how those alternatives would lead to a distribution of possible outcomes, and a set of *preferences*

that indicate how much satisfaction the decision maker would derive from each of the possible outcomes [Matheson and Howard 1968]. The decision analysis process includes drawing distinctions that are important to the specific decision, and quantifying uncertainties about them through field inquiry (such as expert testimony, statistics, and formal models). Following from a decision-maker's agreement with a few simple axioms, decision analysis applies the *theory of rational choice* by identifying the alternative maximizing expected utility as the best choice. Important extensions of DA address decisions having fundamentally different attributes [Keeney and Raiffa 1976], public sector and multiple constituencies [Paté-Cornell 1983, Triubs 1973], and attributing a value to human safety risks [Paté-Cornell 1984.2, Graham and Vaupel 1981, Howard 1983, 1989.2].

Justification

Although theorists and practitioners have challenged DA over its decades of use, meeting those challenges often requires only reinforcing the need for skilful application [Howard 1991, 1992]. As a structured conversation, the use of DA places substantial responsibility for appropriate use in the analyst's hands [Howard 1983]. DA recommends decisions by calculating expected utility based on decision makers' stated preferences and adherence to five simple axioms. Many informed decision makers accept Decision Analysis' simple, clear mathematical foundation easily.

Research by Tversky and Kahneman [1974, as well as Kanheman and Tversky 2000] has found that collecting data from informants without introducing bias requires care, and that decision makers' actual choices deviate from the recommendations of decision analysis in systematic and important ways [Spetzler and von Holstein 1972]. This may be because the best method of making decisions is determined in part by the amount of time available for decision-making [Simon 1977]. Regardless, the very deviation of decision analysis recommendations from observed behaviors underscores the method's importance [Howard 1991].

Limits in Addressing the Research Questions

DA relies upon PRA to provide specific models of operations risk wherever they are needed. Like PRA, DA does not specifically leverage the organizational theory that bears on knowledge work, and relies instead upon field inquiry to model development behavior.

§3.4 explains how the thesis-enabled PRA-VDT framework links DA with Probabilistic Risk Analysis, the Virtual Design Team, and a new model of engineering defects.

2.3 Organizational Theory and Social Psychology

Organization theory and social psychology offer many insights of value to project managers, but lack the precision to support some difficult real-world project shaping decisions.

2.3.1 Safety Culture

> *The Committee found that NASA's drive to achieve a launch schedule of 24 flights per year created pressure throughout the agency that directly contributed to unsafe launch operations. The Committee believes that the pressure to push for an unrealistic number of flights continues to exists in some sectors of NASA and jeopardizes the promotion of a "safety first" attitude throughout the Shuttle Program… -NASA 2003*

Social science and engineering researchers have developed rich theories of collaborative activities and their relationships to risk (for descriptions of over a hundred human and organizational risk factors, and related literature reviews, see Ciavarelli [2003] or Cooke, Gorman and Pedersen [2002]). The need for organizations to consistently act appropriately in light of the urgency of safety issues has been particularly embraced by the nuclear [IAEA coined "safety culture" in 1986, see IAEA 1991], aerospace [Leveson et al 2003], and medical industries [Singer et al

2003; Gaba et al 2003, 2007]. Important theories relating to safety culture include *conformity*, which decreases the likelihood that individuals will contradict peers' public statements, regardless of their error's obviousness [Festinger 1954]. *Compliance* research shows that most ordinary people will commit atrocities at the request of authority [Milgram 1963]. *Groupthink* reduces the likelihood of thorough, critical evaluation of alternatives in a group setting [Janis 1982.2]. Finally, the *risky shift* phenomenon leads groups to select choices that are more risky than those that participants would individually choose [Bem et al 1965].

Limits in Addressing the Research Questions

Because these theories lack a quantitative definition, it is difficult to evaluate rigorously their range of valid application, their potential interactions, and their relative importance when in conflict with other theories. Therefore, they often can inform, but cannot resolve, difficult real-world project shaping decisions.

2.3.2 Normal Accident Theory (NAT)

Normal Accident Theory (NAT) observes that catastrophic failures often result when highly improbable circumstances magnify the effects of seemingly innocuous initiating events [Perrow 1984, 1994, 1999, 2000, Sagan 1993]. In spite of their seeming improbability, according to NAT these accidents are likely to occur frequently in *complex systems*. Complex systems are those having both interactive complexity, meaning numerous potentially unidentified failure modes, and tight coupling, meaning the potential for one subsystem to rapidly and significantly impact the behaviors of other subsystems. According to NAT, any formal risk analysis of complex technologies (such as nuclear power plants and nuclear weapons) is liable to be incomplete, so that in extreme cases, simpler technologies should be substituted [Perrow 1984].

Limits in Addressing the Research Questions

Project shaping decisions should act to diminish risks due to interactive complexity and tight coupling, however NAT provides insufficient guidance to determine the limits of these measures. The method also provides no specific model of nominal (catastrophe-free) performance, and provides insufficient precision to support non-trivial project shaping decisions.

2.3.3 High Reliability Organizations (HRO)

The theory of High Reliability Organizations (HRO) regards attributes shared by teams that experience fewer accidents than NAT assesses [Roberts 1989 and 1990, Roberts et. al 1994, Reason 1997]. Observations of an Air Traffic Control Center, a nuclear power plant, and aircraft carriers inspired the hypothesis that failures will occur less often where there is more attention to process auditing, reward systems, comparative quality, perception of risk, and command and control [Weick 1987, 1993, Weick and Sutcliffe 1999]. The theory explores command and control attributes in particular detail, including power distance, organizational redundancy, and formalization.

Limits in Addressing the Research Questions

The literature discusses relationships between HRO and NAT [Sagan 1993 and 2003, Marais 2004, Cooke et al 2002], but because the theories are qualitative, they lack the necessary precision to make many complex quantitative tradeoffs that shape specific projects. Moreover, HRO has no specific models of physical dynamics or of non-risk metrics essential to project shaping, such as cost, schedule, and quality.

2.3.4 Information Processing View

Galbraith's ***information processing theory*** (1973) assumes that the functions of routing and processing information dominate organizational behavior. Shortcomings

in information flow or knowledge in an organization produce ***exceptions***— events that require information to be referred to management for decision making. Exceptions occur during work with a frequency based on task complexity, as well as on the adequacy of the assigned agent's experience and skills. ***Exception handling*** is the manner in which organizations respond by routing information or queries to complementary resources such as management or technical experts. ***Hidden work*** [Kunz and Levitt 2002] is the coordination and exception handling efforts that can represent a substantial fraction of the total labor and schedule pressures in complex and interdependent projects. One reason project managers underestimate the emergent workloads of subordinates whose work is highly interdependent is that hidden work is hard to assess and not explicit in traditional planning theories and schedule tracking systems.

When a supervisor oversees many agents, all of which are performing tasks in parallel, the exception handling workload sometimes becomes unmanageable. The resulting backlog can cause a failure to respond to coordination attempts, creating a ripple effect of problems extending through all of the interdependent activities. Overloaded workers who fail to respond to communications also compound the information supply problem and compromise others' performance. Projects that involve complex and interdependent tasks impose additional direct and communication requirements and tend to create more unhandled exceptions.

A second factor that critically affects the model behavior is the amount of time between a request for action or information and this ***response latency*** metric is both a cause and consequence of diverse and critically important success factors [Chachere et al 2004.1, Chachere et al 2004.2, and Chachere et al 2004.3]. When projects fall far behind schedule due to managerial or technical bottlenecks, latency reaches a point at which rework decisions no longer occur in a timely fashion. Under these circumstances, rework requirements are often ignored, often leading to a rapid degradation of process quality [Jin and Levitt 1996].

Under adverse conditions, project performance can falter and can degrade rapidly in a manner that is analogous to the emergence of turbulence in fluid flows [Fayol 1949/1916]. When projects fall behind, well-founded decision making and corner cutting alike frequently push risks from the program into the product.

§2.4.3 explains how the Virtual Design Team quantitatively models engineering projects by operationalizing the information processing view. Appendix A provides more detail on VDT's exception handling model, which is a cornerstone of the thesis model of engineering defects.

Limits in Addressing the Research Questions

Assessing the degree of breakdown in knowledge work's quality under specific project circumstances is a challenging research question that the observed problems need addressed. The information processing view, however, provides no explicit model of operations-stage physical processes and how failings in development work can affect them. The theory also provides no specific definition of project outcomes as better or worse for a given decision maker.

2.4 Computational Organizational Modeling

Computational organizational models can help project managers apply organizational theories to difficult real-world decisions, but the models do not explicitly consider physical processes or product risks.

> *Flawed practices embedded in NASA's organizational system continued for 20 years and made substantial contributions to both accidents ... For all its cutting-edge technologies, "diving-catch" rescues, and imaginative plans for the technology and the future of space exploration, NASA has shown very little understanding of the inner workings of its own organization… -NASA 2003*

Computational organizational modeling quantitatively operationalizes established organizational theories, some of which are relevant to the study of risk. Its practical

appeal is that virtual testing of project plans and interventions can produce valuable insights before project resources are committed.

Whereas most project planners forecast the behavior of engineering efforts based largely on personal experience, those who employ computational organizational models focus on accurately assessing engineering project parameters (such as tasks' complexities and workers' skills). Based on this information, VDT simulates the engineering work and coordination that established organizational theories assess would emerge in practice.

By grounding a computational model explicitly in a theoretical framework, researchers can explore complex ramifications of a theory (or set of theories) that extend qualitatively beyond the reach of human intuition. Although some of the earliest and most influential work in organizational science was developed in concert with formal models [Cyert et al 1959, Cohen et al 1972, Lave and March 1975], the method has never become a standard in the discipline. In recent years, however, the computational modeling of organizations has enjoyed a popular resurgence among researchers seeking to better understand new and established theories [March 2001 and Burton 2001].

2.4.1 The Organizational Consultant (OrgCon)

The Organizational Consultant (OrgCon) is a rule-based expert system that Richard Burton and Børge Obel developed and documented in *Strategic Organizational Diagnosis and Design* [2003]. When provided with values for a set of contingency variables such as structure and environment, this system assesses an organization's potential weaknesses in terms of mismatches between its strategy, structure, climate, management style, and other factors. OrgCon's recommendations are firmly rooted in a range of established organization contingency theories.

Limits in Addressing the Research Questions

OrgCon typically assesses the long-term behaviors of whole companies (or subsidiaries), rather than short-term project teams that often assemble for aerospace or construction projects. The system does weigh the strength of confidence between potentially conflicting recommendations from theory. The system's recommendations, however, lack the degree of precision that is necessary to support specific project shaping decisions.

2.4.2 Interaction Value Analysis (IVA)

Interaction Value Analysis (IVA) uses mathematical queuing theory and a game theory analysis to assess long-term organizational efficiency [Nasrallah et al 2003, Nasrallah 2006]. IVA assesses that organizations satisfying certain criteria (described as high diversity, low interdependence, low differentiation, low urgency, or low load) are likely to gradually develop perfectly efficient operations (in which Pareto optimality equals global optimality). In contrast, projects that do meet at least one of the criteria require an externally-imposed communication structure to prevent substantial inefficiencies in resource allocation.

Limits in Addressing the Research Questions

Like OrgCon, IVA focuses on the long-term performance of organizations, rather than the behavior of project teams assembled for a limited duration. IVA also provides no explicit model of the work product, or of the operations context it can affect.

2.4.3 The Virtual Design Team (VDT)

The Virtual Design Team (VDT) is a discrete-event simulation model based on several of the most established theories of organizations (notably Galbraith 1977 and Thompson 1967). VDT uses the information processing view (§2.3.4) to assess how specific configurations of organizational hierarchy, process network, and culture

interact during a project to determine participants' backlog, coordination effectiveness, schedule risk, labor costs, and other emergent objectives [Kunz et al 1998; Jin et al 1995, Levitt et al 1999]. Since its inception [Cohen 1992], researchers have developed enhancements to apply the tool to address multicultural teams [Horii 2005], fast-tracking [Salazar-Kish 2001], learning [Oralkan 1996], goal congruency [Thomsen 1998] and trust in distributed teams [Zolin 2002]. For a review of VDT's theoretical basis see Christiansen 1994, and for a technical explanation of its internal mechanics see also [Jin and Levitt 1996].

VDT assesses the impacts of many development precursors of risk, such as missed communications, unattended meetings, and the handling of exceptions in technical work. The VDT model has shown some remarkable successes in assessing engineering phenomena in real projects that lead to operations failures [For an aerospace example see Levitt et al 1999]. For this reason, many organizations, including NASA, have used VDT to quantitatively assess the behaviors of complex development projects with significant downstream (operations stage) risks.

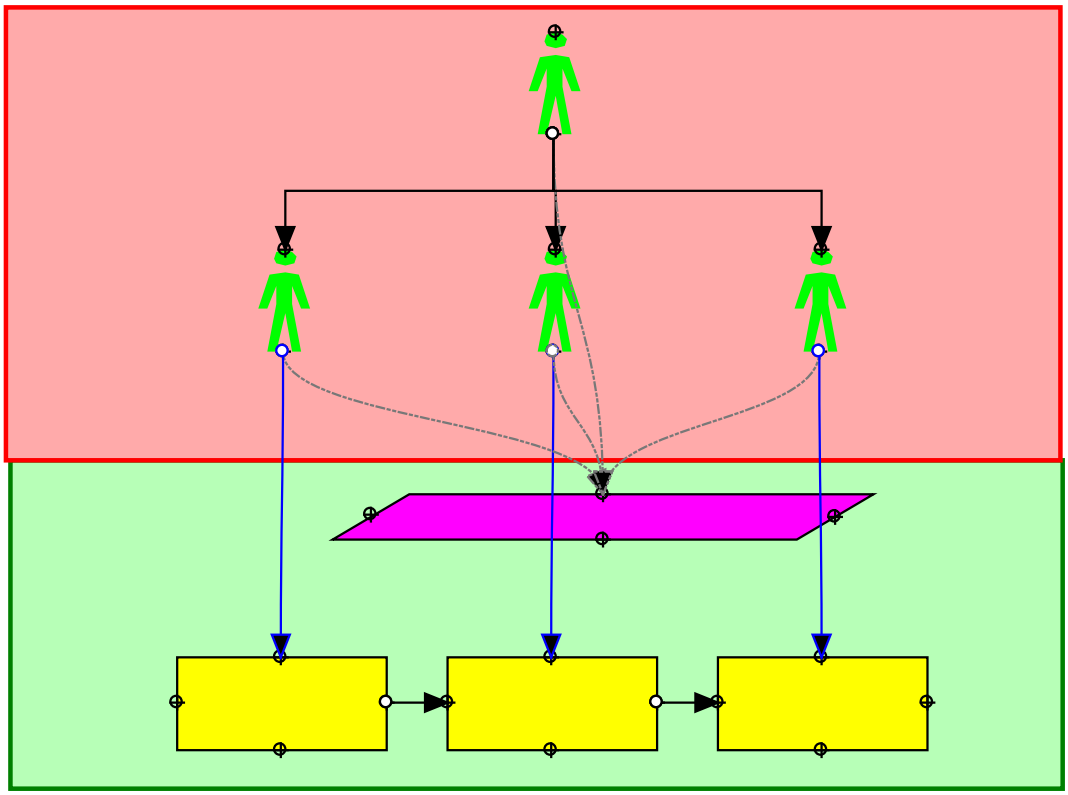


Figure 2.4 Screen Image of Virtual Design Team (VDT) Software for a Simple Example Satellite Project Model

Consider Figure 2.4 Screen Image of Virtual Design Team (VDT) Software for a Simple Example Satellite Project (above). VDT models design project participants with individual characteristics (green human shapes), organizational exception handling structure (black links), scheduled meetings (purple parallelogram), tasks with individual characteristics (yellow boxes) and precedence (horizontal, black arrows), information exchange requirements (not shown), and rework dependencies (not shown). VDT compares the process's information processing load with the organization's information processing capacity. It produces detailed estimates of a project's emergent cost, schedule, quality, and other measures.

VDT analysis models the agents of organization hierarchy and the tasks within a precedence network. The method represents organization centralization, formalization, and matrix strength (project versus functional orientation); agents' varying size and levels of skill and experience; and process tasks with varying levels

of procedural uncertainty, complexity, and required skills. The VDT model also defines relationships among these basic elements, including authority hierarchies that interrelate agents; primary and secondary task assignments indicating which agents address which tasks; and links interrelating tasks that have precedence, rework and information exchange dependencies.

VDT outputs a range of performance assessments, including emergent direct and hidden work volumes, a project schedule, and coordination rates. VDT estimates overall design quality using coordination time such as information exchange and meetings, and decision waiting time. Users can view these metrics at an aggregate project level, and as detailed assessments about individual agent or task results.

Appendix A (on page 192) provides more detail on VDT's exception handling model, which is a cornerstone of the thesis model of engineering defects.

Justification

VDT is based on organization theories that are well documented and academically mainstream. However, VDT's detailed representation and complex reasoning present novice users with a significant barrier. Nearly two decades of published documentation of VDT modeling efforts include prospective and retrospective field studies in addition to thought experiments and theory-building. As yet, there is no documented, rigorous scientific evaluation of VDT's assessment accuracy.

Currently, VDT serves academic research, engineering management education, and in the field (typically under professional consultants' supervision). VDT's practical achievements include the quantitative assessment of schedule delay and the identification of critical organizational faults in a Lockheed satellite's cabling system developer [Levitt et al 1999].

Limits in Addressing the Research Questions

As with any model, VDT's assessment power results from a set of assumptions. As its name suggests, the Virtual Design Team focuses on routine knowledge work, and it models the "physics" of rational organizational behavior, but not the "chemistry" of, for example, ambiguity [March and Olsen 1985] or deeply creative design. In addition, VDT regards input quantities (such as agent skill) as known, even though the informants providing data are often uncertain about the details of team or task composition (Chachere 2004.1 proposes a solution to this problem). VDT does not simulate agent decisions to strategically change the organization or process during the project, even though this behavior is common in large real-world projects. VDT leaves it to users to compare the merits of assessed project outcomes and determine which among several similar outcomes is most preferred.

VDT lacks an explicit product model, which can limit consistency in setting assumptions and obscure the relative importance of interventions that individual projects merit. Translating VDT results into recommendations is difficult because even though VDT assesses schedule improvements, for example, the product often is affected in ways that VDT does not estimate explicitly. VDT reports some of its most important measures of outcome simply as "risks" that not explicitly related to downstream products or their operating environments, and therefore VDT cannot assess the impacts these behaviors will have on the probability of different failure modes in operations. For example, it is clearly more important to address wiring risk to an aircraft's navigation system than to its in-flight entertainment, but VDT leaves it to users to account for the relative importance of process risks in the corresponding tasks.

Even though these shortcomings prevent VDT from fully addressing the research questions, the thesis explains in Chapter 3, and demonstrates in Chapter 4 and Chapter 5, that the method serves a critical function within a framework (termed PRA-VDT) that does address the research questions.

2.5 Project Optimization Models

Two existing models optimize projects in a manner similar to this thesis, but one omits engineering defects (a source of dependency between development and operations), and the other omits development organizations' capacities (that approach their limits in many novel projects).

Planners today find little research providing precise analysis of common but difficult practical decisions involving specific project risks, costs, schedules, and other objectives. Exceptions include quantitative programmatic and risk models developed by engineers (most notably Paté-Cornell 1990, Paté-Cornell and Fischbeck 1993.1 and 1993.2, Murphy and Paté-Cornell 1996, Paté-Cornell et al 1996, and Dillon and Paté-Cornell 2001).

This section describes two existing project planning methods that consider both project failure risks and program risks (such as schedule overruns). Although the models contain many elements of the four fields addressed previously in this chapter, the two existing project optimization models do not fully address the observed problems that Chapter 1 introduced.

2.5.1 Advanced Programmatic Risk Analysis and Management (APRAM)

APRAM [Dillon and Paté-Cornell 2001, Dillon et al 2003] addresses both program and project risks by defining a budget that can be divided between expenditures on the configuration and reinforcement of the engineered system (that mitigate project failure risks) and budgetary reserves available to counter uncertain development events (that mitigate program failure risks).

APRAM identifies the ideal, continuous-valued amount to reinforce components in each alterative configuration of physical components (using the KKT conditions to solve a nonlinear optimization). APRAM then uses decision trees to model the

uncertainties and decisions that contribute to program risks. APRAM balances a model of uncertain development costs against up-front expenditures on reinforcement of the operational system.

Limits in Addressing the Research Questions

APRAM shares with PRA and DA the reliance on field inquiry to model development behavior, and includes no specific organization-theory based model. In addition, APRAM gains much of its power from linking development and operations only through the budget constraint and stage-wide operations and management failure probabilities. With regard to the research question, for example, APRAM does not explicitly model probabilistic dependencies between development outcomes that can influence the expected prevalence of engineering defects and the risks of failures in individual components during operations.

2.5.2 Exception-Detection Model

Carlo Pugnetti's 1997 doctoral dissertation approaches the problem of simultaneously estimating program and project failure risks using PRA and VDT. The Exception-Detection model uses a discrete time queuing model of development processes to estimate the probabilities of failure in operations functions. Like this thesis, Pugnetti's Exception-Detection model allows undetected exceptions to contribute to error probability in a linear or nonlinear fashion, and bases decisions on expected utility. Pugnetti's thesis uses VDT as well, in a separate analysis step, to verify the organizational viability of optimized process and product configurations.

Pugnetti recognizes unidentified engineering errors as important contributors to operations risks. In the most sophisticated of the models his thesis outlines, Pugnetti [1997] p.71 uses "undetected exceptions" to measure the risk associated with a simulated engineering project. Pugnetti's "undetected exception" measure resembles the "degree of verification" concept that this thesis models as relevant to the

emergence of engineering defects (see §3.1.3). This thesis, however, observes the social science tradition of using "exception" to describe a procedural event [Galbraith 1973] (§3.1.2), and adopts the term *defect* to reference an engineering flaw (§3.2.1). Regardless of nomenclature, Pugnetti's thesis and this one both interpret these events as impacting individual component failure probabilities in downstream operations.

When conditions are at their worst, Pugnetti's models assess that engineers can create more exceptions than they fix, and that the project will go on forever. By providing model convergence criteria that define sufficient (though not necessary) conditions under which this occurs, the Pugnetti thesis sheds light on the important "turbulence" problem §2.3.4 described.

Limits in Addressing the Research Questions

The Exception-Detection Model effectively addresses very similar problems to those motivating this thesis (see Chapter 1). This thesis therefore regards the Exception-Detection model as a fundamental point of departure (even though Pugnetti has yet to publish the work in a peer-reviewed journal). The remainder of this subsection explains specific aspects of the targeted problems that require expansion beyond the Exception-Detection Model's feature set.

The principal advance of PRA-VDT over the Exception-Detection model is in depth of integration with the VDT model of development behavior. Pugnetti himself clearly lays out the potential benefits of deeper integration, which the PRA-VDT method achieves:

> *While the results obtained using [the Exception-Detection Model and] VDT are a good indicator of the influence of organizational structure on the design process, it is also true that more integration between [the Exception-Detection Model] and VDT would result in a better understanding of the overall interaction between organizational structure and design schedule. The current solution of the combined models is, in general, suboptimal. The uncapacitated models optimize the design schedule, and VDT investigates a number of organizational structures to determine the one best suited to accomplish the chosen design schedule. The two decisions happen sequentially, and the current setup does not allow any interaction between the structure and the schedule. Unless manually coordinated, this system also does not offer the investigation of the interaction between different near-optimal schedules and the organizational alternatives. Even if an automatic way to create the combination existed, it would not be possible to compare the results, as VDT does not consider the presence and propagation of undetected exceptions, the key phenomenon in the model driving the tradeoff between design duration and failure probability.*
>
> *–Pugnetti 1997 pp. 108-109*

The Exception-Detection Model focuses on the effects of procedural concurrency, so it uses an organization model that is "Uncapacitated," meaning that the process tasks are assumed to be executed by agents with uniform competence. In contrast, Chapter 1 provided evidence that in the applications this thesis addresses, development performance depends critically upon features of organizational capacity, including organizational culture, coordination overhead, emergent work backlog, and the matches between agent skill and task complexity. Pugnetti underscores this fact in Future Research- Integration with VDT Model (p.142), by indicating the need to "Allow for parallel search through organizational structures and design schedules."

The Exception-Detection model does not distinguish the occurrence and severity of defects from the occurrence of exceptions during development; The model assumes that agents fix all the problems they are simulated to observe, and that reworked elements do not contribute to failure probability. These assumptions do not hold in the applications this thesis addresses, according to field evidence (notably at NASA [2003], see §1.2 on page 7), according to theory (notably bounded rationality [March

and Simon 1958], see §2.3 on page 40), and according to other models of the domain (notably SAM [Murphy et al 1996], see §2.2.4 on page 36).

The Exception-Detection model estimates a marginal probability of failure for each component, and then calculates a total failure probability using PRA. By contrast, the problems this thesis addresses involve events occurring in development, as well as in operations, that are probabilistically dependent. This thesis therefore requires, and uses, full joint probability distributions on the development outcomes. For further discussion of probabilistic dependencies in the PRA-VDT method, see Appendix B (on page 196).

Chapter 3
Model Definition

The Defect Model provides a quantitative interface linking engineering defects' explicit causes (ignored development-stage exceptions) to corresponding consequences (increased component failure probabilities). The PRA-VDT Defect Model links a VDT development-stage organization and task model with an operations-stage PRA product failure model. The PRA-VDT framework provides a formal, theory-founded method to systematically assess a product-organization-process design trade space.

Successfully developing and deploying a new product requires the careful consideration of potential cost, duration, and failure risk. Particularly for products that are novel and complex (such as space missions, unique civil facilities, and medical devices), the prospect of technical failures rooted in engineering defects is important to many operational decisions.

It is difficult to identify, assess, and mitigate risks from accident sequences involving engineering defects because they span projects' development and operations stages. Analyzing the dependencies that connect operations failures to their causes in development involves diverse conceptual challenges, including:

Distances in space (often teams are at different locations),

Distances in organization (often different companies manage development and operations),

Distances in control (often different companies or divisions have loosely coupled management, command, and control systems),

Passages of time (often failure occurs years after a design error is committed), and

Conflicts of incentives (often developers are rewarded for schedule and cost, while operations are rewarded for technical success)

For example, the profitability of a traditional apartment building relies on complex interactions between diverse factors such as architectural and engineering quality, the rental market, level of luxury in design, and seismic conditions.

Currently, major project management decisions (such as how much to overlap development tasks) rely on professional judgment without formal models' aid. Theory-founded models can, however, help managers understand those novel projects that lack clear precedents, reusable statistics, and reliable expert judgment. Chapter 2, Existing Practice and Theory (which starts on page 21), provides many examples of these models, and explains the extent of each one's power, generality, and limits. Models of development organizations and of the produced system's reliability during operations assess different measures of interest to managers and rely upon different assumptions. Their focus amplifies the methods' assessment powers, but limits their value in cases where development and operations strongly interact. For example, modeling an apartment building's cash flow can help managers budget maintenance expenses. Typical cash flow models provide no information about the risks of management deferring the work, however, even though decision makers should consider the risks.

This thesis relates most directly to three existing theoretical frameworks, which are fundamental points of departure. Development managers can use the Virtual Design Team (VDT) to model their organizations and processes, but they also need to estimate the influence that development process quality will have on operations performance. Operations managers can use Probabilistic Risk Analysis (PRA) to model their product functions and operating contexts, but they also need to estimate the severity, location, and influence of engineering defects. Project managers can use

Decision Analysis (DA) to suggest the best project alternative, but they also need a joint probability distribution of development and operations impacts to identify synergistic risks and opportunities that simplistic analyses based on marginal distributions would miss. To illustrate these interactions, a VDT model of apartment building developers might assess that overlapping design tasks reduces schedule and cost. Traditional methods cannot assess the likelihood that overlapping might lead to quality degradation and an operational problem (such as wiring defects that jeopardize building occupants' safety).

This chapter presents an integrated method, the PRA-VDT framework (PRA-VDT), providing intuition, formal definition, and illustration. The PRA-VDT framework quantitatively estimates the effects of product, organization, process, and context choices on development and operations events, such as component and total systems failures that affect a decision maker. Compared with standalone models of development or operations, the integrated PRA-VDT Framework offers project planners an integrated method to assess the individual and joint impacts of change in a product-organization-process trade space. These features are particularly important for projects where component redundancy, human error, management backlog, safety culture, and other considerations interact.

The thesis describes a quantitative model (the Defect Model) that links the causes and consequences of various engineering defects in given project contexts (This thesis does not address non-engineering defects, such as manufacturing defects or installation defects). The PRA-VDT framework's new Defect Model uses expert opinions to structure the VDT-assessed volume of ignored work in subtask exceptions and assess the numbers, locations, and severities of engineering defects, and to assess the resulting loss of feature capacities (such as success probabilities) in operations. The thesis situates that interface in the context of the PRA-VDT Framework, a set of complementary models that includes VDT, the Defect Model, PRA, and DA.

Organization of Sections and Framework Overview

This chapter formally defines the Defect Model and the PRA-VDT framework, whereas Chapter 2 introduced the standalone models that serve as fundamental points of departure, Chapters 4 and 5 illustrate the method using illustrative studies of a satellite and a dormitory, and Chapter 6 draws conclusions and presents ideas for extending the thesis contribution.

This chapter shows how using the Defect Model within a framework (termed PRA-VDT in the thesis) can identify which of several given project alternatives offers the best distribution of possible outcomes for a given decision maker. PRA-VDT links PRA, and VDT models using the Defect Model, which formally defines a ***product*** as a system that development creates, and that influence operations behavior in ways that can affect a decision maker. Figure 3.1 outlines how PRA-VDT uses the Defect Model to support project shaping decisions (for example, between two alternative organizational designs).

Each of this chapter's four sections addresses one PRA-VDT Model (see Figure 3.1):

§3.1 Development Model uses VDT (introduced in §2.4.3 on page 46) to estimate the distribution of development outcomes: the work's duration, cost, and degree of verification. The method forecasts the emergent behaviors of a development organization taking on specific tasks and coordination needs. In schematic figures, Development Model elements are green.

§3.2 Defect Model uses the Defect Model (introduced in this section) to estimate the distributions of different engineering defect types by identifying the conformance probability corresponding to development's degree of verification, and to assess how those defects affect different subsystems during operations. This thesis does not address non-engineering defects, such as manufacturing defects or installation defects. In schematic figures, Defect Model elements are blue.

§3.3 Operations Model uses PRA (introduced in §2.2.3 on page 33) to estimate the distribution of operations effects (time to failure) by evaluating joint distributions of engineering defect-influenced operations capacities. In schematic figures, Operations Model elements are red.

§3.4 Decision Model uses DA (introduced in §2.2.5 on page 37) to identify the project (development and operations) alternative with best expected results, based on the assessed distributions of operations failures. In schematic figures, Decision Model elements are purple.

The representation's principal entities and relationships appear in Figure 3.1 Decision Diagram Overview of the PRA-VDT Framework (on page 61). Project Managers aim to maximize project expected utility (purple) by choosing the best possible combination of development, product, and operations alternatives (green, blue, and red respectively). Their decision is sometimes difficult because the choices impact diverse project behaviors that interact in complex ways. The PRA-VDT Framework can help project managers by synthesizing the assessments of four models that each focus on one aspect of project behavior. The integrated model includes a VDT simulation of product development (e.g., orbiter design activities), the thesis definition of engineering defects (e.g., software bugs in the orbiter thruster controls), a PRA of impacted operations (e.g., thruster misfires), and benefits to the decision maker (e.g., expected degree of success in operations).

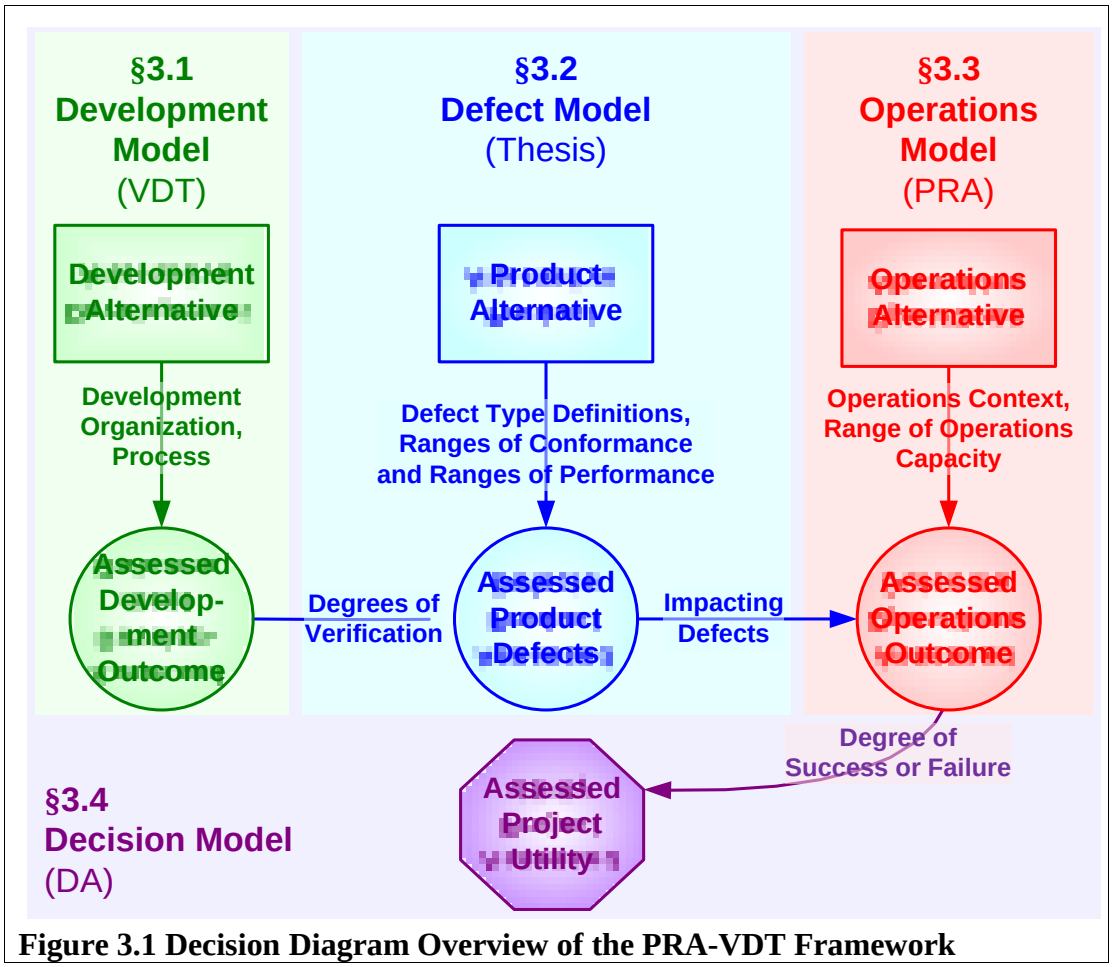


Figure 3.1 Decision Diagram Overview of the PRA-VDT Framework

The method's reasoning steps appear in Figure 3.3 PRA-VDT Framework Analysis Method (on page 64). For all of the alternative project configurations that a decision maker can choose from (such as one involving parallel and one involving serial development tasks), the PRA-VDT Framework requires four steps: analysis of development organization and process using VDT, assessment of resulting defects by the Defect Model, assessment of resultant operations outcomes using PRA, and determination and comparison of those outcomes' desirability using DA.

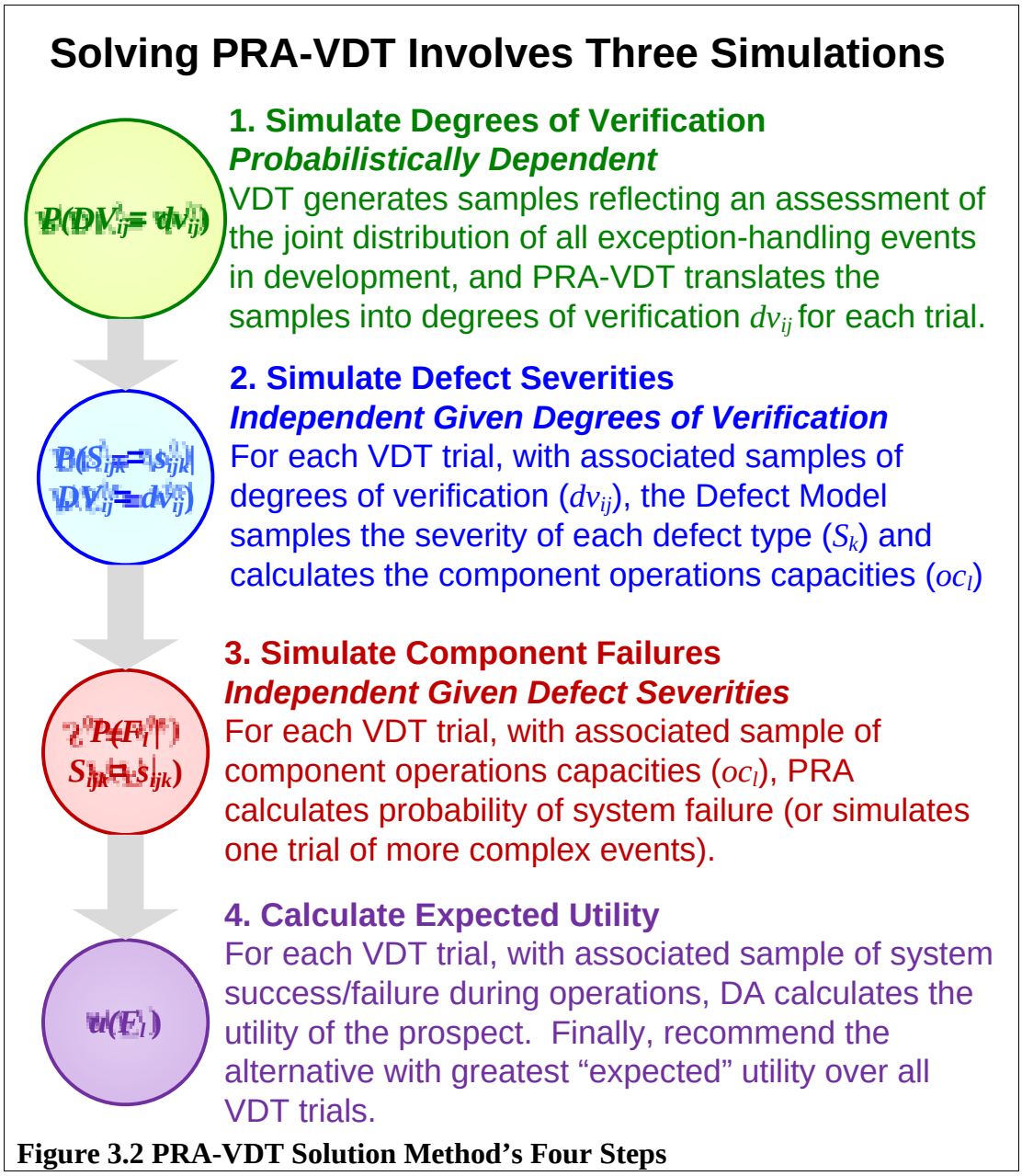


Figure 3.2 PRA-VDT Solution Method's Four Steps

Figure 3.2 (above) introduces the four steps of the method that this thesis uses to solve PRA-VDT models. The applications in this thesis use a general-purpose method of estimating each project alternative's generated defects and expected utility by simulation (using a computer program to sample values that reflect the distribution of assessed project outcomes), even though broader use of closed form analysis may be possible for some applications. In the first step (green, top), VDT generates a set of samples from a model of the joint distribution of all exception-handling events in

development. In the second step (blue, middle), for each trial the Defect Model generates one sample of the severity of each defect type and calculates the component operations capacities. In the third step (red, middle), for each trial the Operations Model samples the distributions of component failures that result from corresponding defect severities. In the fourth and final step (purple, bottom), the Decision Model uses DA to combine all the simulation trials' results and identify as "best" the alternative that maximizes expected benefits. The solution method transfers to later stages the probabilistic dependencies between events occurring at each stage, which is essential to the estimation of complex projects' behaviors. A detailed discussion of this method of solving by simulation and the implications for probabilistic dependencies appears in Appendix B (on page 196).

Figure 3.4 (on page 65) shows the "index structure" that the Defect Model uses to formalize and interface knowledge about relationships between a project's tasks, defects, and product components, enabling more broad and precise interpretations of VDT output. Research has hypothesized [Sosa et al 2004] that a perfect mapping ("isomorphism") between the development "work breakdown structure" and the operations "product breakdown structure" is ideal, but is not always in place because products often change more rapidly than organizations can [Sosa et al 2004]. Efforts to prove that an isomorphism is ideal have had limited success. The Defect Model provides enough flexibility to represent any mapping between development tasks and defect types (using *conformance probabilities*, see page 80), and between those defect types and attributes of product components, interfaces, or even the complete system (using *defect influences*, see page 95). Projects that have such an isomorphism should model defects by relating functional ("internal," task-based) exceptions directly to the behavior of components. These implementations can also relate project ("external," rework link-based) exceptions directly to the behavior of interfaces between interacting components. Figure 4.4 (on page 115) shows the mapping between tasks and product components for the illustrative satellite example.

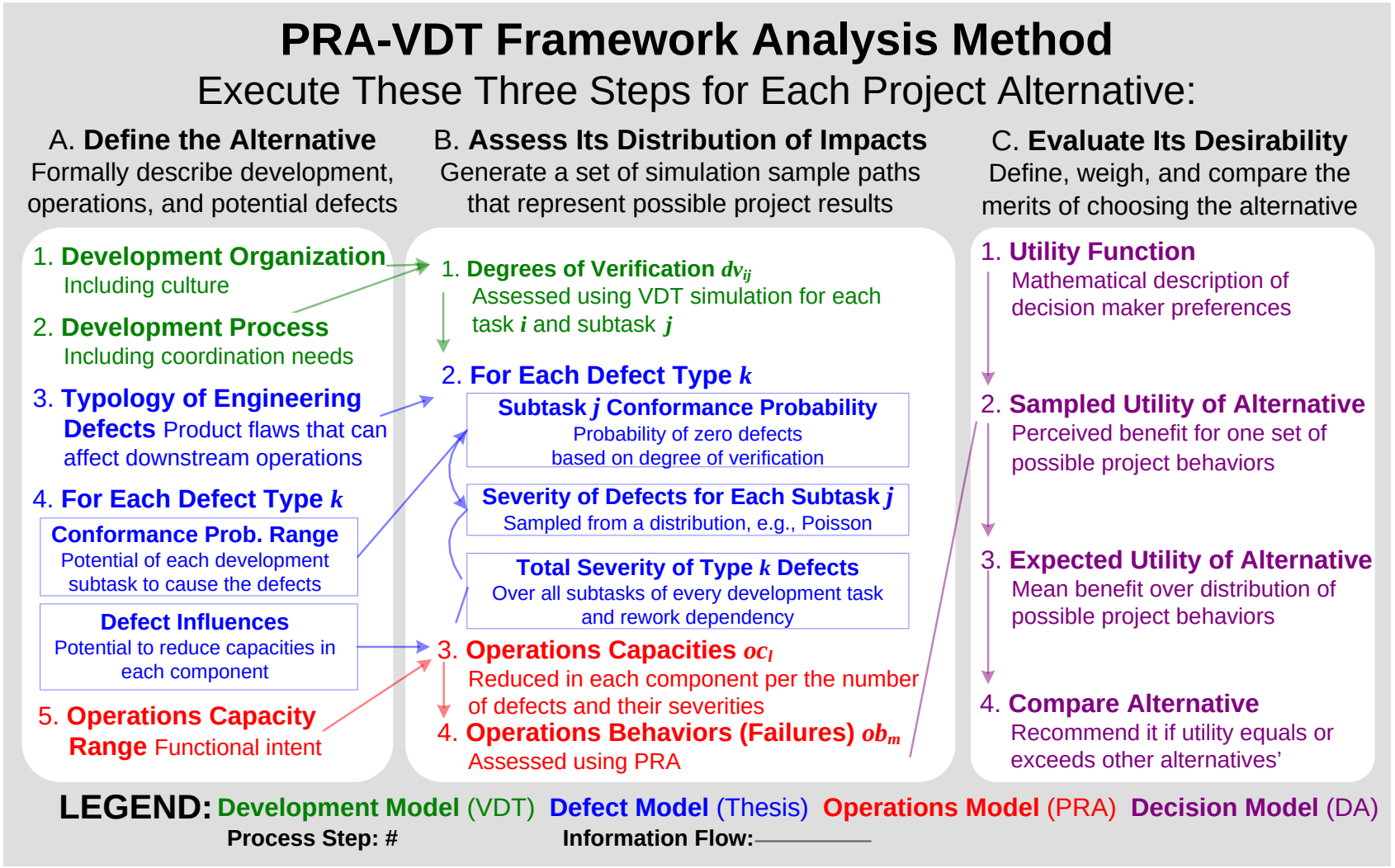


Figure 3.3 PRA-VDT Framework Analysis Method

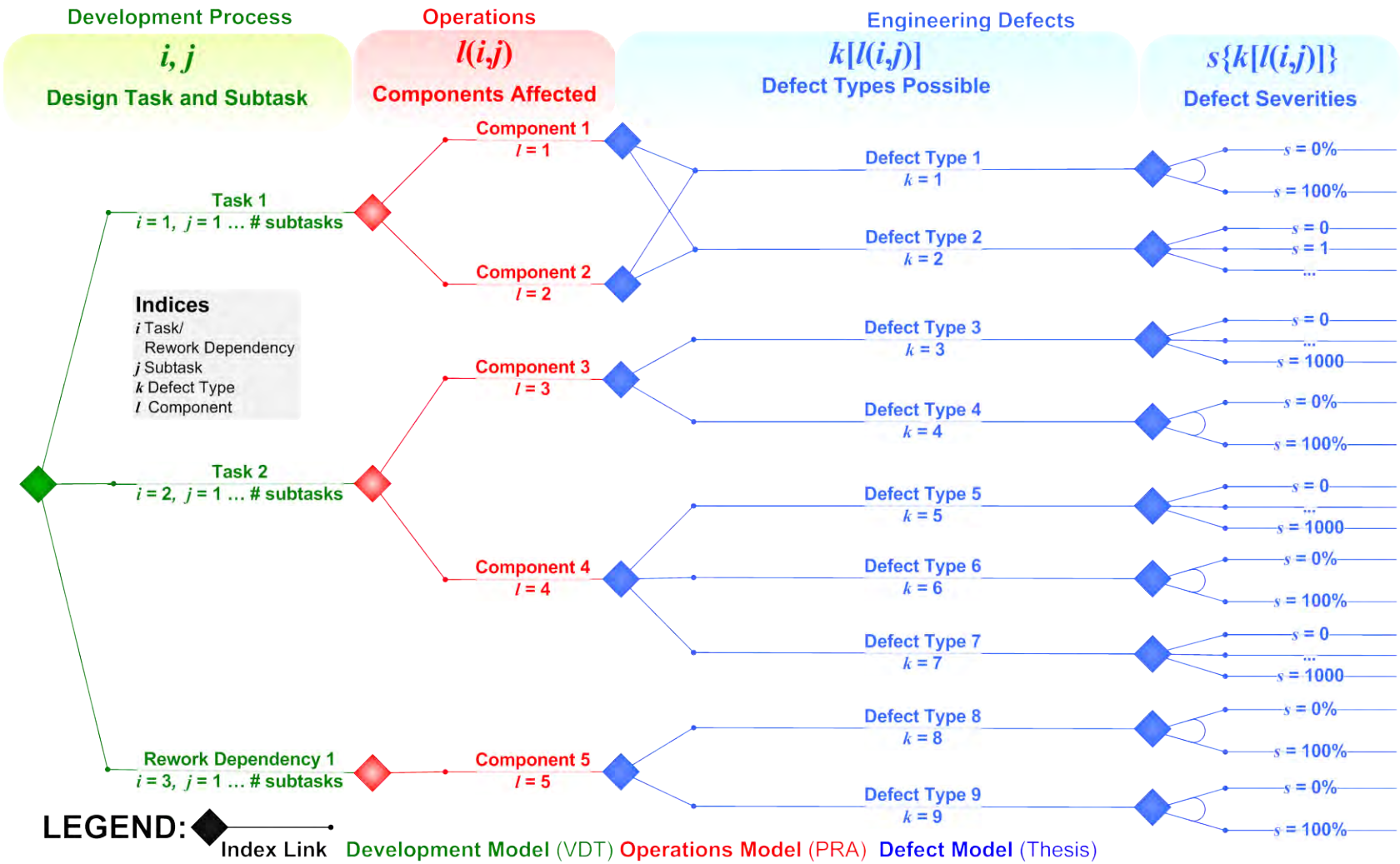


Figure 3.4 Indexing Structure Relating Elements of the PRA-VDT Framework Analysis Method

3.1 Development Model

This section explains how the Development Model uses the Virtual Design Team (VDT) to assess the distribution of work verifications and exceptions that will occur during a project's development stage.

The Development Model uses the Virtual Design Team (VDT) model to estimate the (probability) distribution of a set of important events that can occur during the given development project. VDT takes input from structured interviews about the development organization, process, and project culture. As output, VDT produces a joint distribution over a range of activities of different types (including direct work, rework, waiting, and communications) that occur in continuous time.

VDT assesses development duration and labor cost, which frequently interest a decision maker directly, however the thesis addresses only the impact of development on the risk of defects (see §3.4). The thesis focuses therefore on each work item's ***degree of verification*** – the fraction of associated work that the developer successfully verified as meeting the specification. The degree of verification is relevant to the distribution engineering defects (see §3.2) that can influence operations failure risk (see §3.3).

VDT is an object-oriented discrete-event simulation that models a wide range of distinctive real-world development factors, such as developer skill, coordination requirements, and management bottlenecks. VDT uses a Generalized Semi-Markov Process to represent a stochastic process that formally operationalizes organizational theories, but that has no practical closed form representation. The result is a set of sample paths from that stochastic process. Currently, the need for manual data input and for post-simulation results analysis limits the number of alternatives that is practical to evaluate using VDT.

Table 3.1 Development Model Variables

Term	Name	Data	Source	Description
i	***Task or Dependency***	Index	Modeler	The index identifies VDT "tasks", which are homogeneous blocks of development work. In complex projects, the index also identifies any VDT "rework dependencies", which indicate critical interdependence between blocks of work (tasks). Tasks consist of many small subtasks.
j	***Subtask***	Index	Modeler	The index identifies VDT "subtasks", which are small portions of a task's work that are typically performed in one simulated day
wv_{ij}	***Direct Work Volume***	Work Volume	VDT Output	Total amount of work scheduled for task or rework dependency *i* subtask *j* (A one-day subtasks equals 8 hours' direct work)
ev_{ij}	***Exception Volume***	Work Volume	VDT Output	Amount of work that generated exceptions for task or rework dependency *i* subtask *j* (An exception for that subtask equals 8 hours' exception volume)
rv_{ij}	***Rework Volume***	Work Volume	VDT Output	Total amount of rework executed for task *i* subtask *j* (if the exception caused an ignore decision the rework volume is zero)
dv_{ij}	***Degree of Verification***	Fraction R.V.	Development Model Output	**Fraction of direct work** in task or rework dependency *i*, subtask *j* that was verified (did not raise exceptions, or resulted in verified rework).
dv'_{ij}	***Rework Item Degree of Verification***	Fraction R.V.	Development Model Output	**Fraction of rework** generated for task or rework dependency *i*, subtask *j* that was verified (did not raise exceptions, or resulted in verified rework).

3.1.1 Development Alternatives

The Development Model uses the VDT model of the information processing theory of organizational behavior (see §2.3.4 Information Processing View on page 42). The Model first ascertains, through a series of structured interviews, the essential features of product ***development organization*** – a hierarchy of information processing agents that are available to create a product. Next, the Model assesses similarly the project's ***development process*** – a network of interrelated information processing tasks that incite agents to create product elements.

Through a series of structured interviews, the development manager provides a consistent set of management alternatives that VDT can assess, each including an organizational hierarchy, task network, and operating culture. For example, one

alternative might include a flat hierarchy, parallel tasks, and centralized culture, while another alternative includes a deep hierarchy, sequential tasks, and decentralization.

Development Organization

As with VDT, the Development Model views development teams (or other groups of knowledge workers) as a hierarchy of organizational agents that collectively possess information processing capacity (the potential to conduct knowledge work).

Table 3.2 describes the organization and culture data that development project managers provide to the VDT model (see Jin and Levitt 1996 for more detail).

Table 3.2 Types of Organization and Culture Data Input to VDT

Organization	
Agents	Set of teams involved in the development project
FTEs	Number of full time workers each agent represents
Skill*	Amount of technical ability each agent applies. *With an assigned task's requirement complexity, this influences task duration and internal exception rate.*
Experience*	Amount of similar projects each agent has worked. *With an assigned task's solution complexity, this influences task duration and external exception rate.*
Role	Decision making authority (Subteam, Subteam Leader, or Project Manager). *Role determines which types of decisions an agent will make and how it will make them.*
Salary	Wage per completed hour of simulated work.
Supervisor	Which agent (if any) the team reports to for exception handling
Project Culture	
Functional Error Rate	Nominal probability of a functional (internal to the failing task) exception after each subtask completion (initial probability assuming that all task and agent attributes are medium)
Project Error Rate	Nominal probability of a project (external to the failing task) exception after each subtask completion (initial probability assuming that all task and agent attributes are medium)
Communication Rate	Nominal probability of communication after each subtask completion (initial probability assuming that all task and agent attributes are medium)

Table 3.2 Types of Organization and Culture Data Input to VDT	
Centralization*	Determines the distribution of decision-making authority between supervisors (High) and work teams (Low) in the organizational hierarchy. *Determines which roles are more often responsible for deciding whether rework is necessary.*
Formalization*	Determines the volume of information communications that agents use.
Matrix Strength*	Determines agents' preference for meetings vs. informal communications.
*	These input values are approximated as High, Medium or Low; Real-valued calibration constants operationalize these qualitative levels in the simulation.

Development Process

The VDT model describes the requirements of development as a network of procedural tasks that require information processing by assigned actors.

Table 3.3 describes the process data that development project managers provide to the VDT model (see Jin and Levitt 1996 for more detail).

Table 3.3 Types of Process Data Input to VDT	
Process	
Tasks	Blocks of work that the agents perform for this project
Assigned Agent	Identifies the team that will perform the task. *Assigning a capable agent encourages project success.*
Work Volume	Amount of work the task comprises. *With the assigned agent FTEs, this determines task direct work duration.*
Requirement Complexity*	Difficulty of finishing the work while satisfying the task's requirements. *Along with assigned agent skill, this influences task duration and internal exception rate.*
Solution Complexity*	Difficulty of finishing the work while allowing other tasks to satisfy their requirements. *Along with assigned agent experience, this influences task duration and external exception rate.*
Uncertainty*	Volume of information that must be collected from related tasks. *With higher uncertainty, more information communications occur.*
Predecessor Links	Which other tasks (if any) must be completed before a given task can begin

Table 3.3 Types of Process Data Input to VDT	
Communication Links	Which other tasks (if any) must share information with a given task
Rework Dependencies	Which other tasks (if any) may require rework because of work on a given task.
Meetings	Short, repeated coordination items to which multiple agents are invited.
Invitees	Set of agents who are invited to a given meeting
Schedule	Frequency, timing, and duration of a given meeting
*	These input values are approximated as High, Medium or Low; Real-valued calibration constants operationalize these qualitative levels in the simulation.

3.1.2 Development Assessments

This section explains the distribution of impacts that VDT assesses by simulating a development organization and process. The results of interest to a decision maker typically include the development project duration, cost, and exception handling behavior. However, this thesis only considers the latter measure as influencing engineering defect risks. §6.3.5 (Modeling Broader Utility Functions, on page 185), and §6.3.7 (Assessing the Accuracy of Cost, Quality, and Schedule Estimates, on page 187) explain two extensions to PRA-VDT for considering cost and duration as well as defect risk.

The PRA-VDT Framework uses index *i* to identify the development stage's tasks and rework dependencies (for illustrative examples, see the corresponding sections in Chapter 4 and Chapter 5). The framework uses index *j* to distinguish individual subtasks, which are the small elements (typically representing a day's work) that each task comprises. In addition to the direct work of process tasks and subtasks, indices *i* and *j* distinguish the exceptions corresponding to work: functional exceptions (corresponding to tasks) and project exceptions (corresponding to rework dependencies). For more about exception handling and the distinction between tasks and rework dependencies see Appendix A (on page 192).

3.1.3 Development Impacts

The probabilistic dependencies between design cost and duration suggest that a thorough analysis requires assessing the full *joint* distributions of outcomes, rather than reliance on marginal distributions (a common practice for VDT analyses). Appendix B on page 196 details both how and why this thesis assesses the full joint distribution using simulation.

Exceptions and Rework

VDT assesses the distributions of phenomena that, according to the organizational and social psychological literature, produce risky decisions. More specifically, Chapter 1 explained that post-accident investigations often find that development displayed signs warning of danger, such as the escalation of engineering concerns that management subsequently ignored. Because the engineering contributors to operations failure frequently correspond to work that should have been redone, the Development Model analysis focuses on exceptions and rework.

Exceptions and Rework comprise the left half of Figure 3.5 Generic Event Tree Relating VDT-Assessed Exception Handling to Degree of Verification, Conformance Probability, and the Distribution of Engineering Defects (on page 73). (The following section of this thesis explains the right half of the figure, which discusses defect creation) After finishing each work item, a VDT-simulated agent raises an exception with a probability determined by factors linked by theory to the creation of errors.

If an exception occurs, a VDT agent chooses whether to conduct rework. VDT agents use a simple behavioral (stochastic, not decision-analytic) model of the actor's choice, which trades off extra duration and cost versus defect risks. An agent's choice to rework or to quick-fix creates a work item that might also create exceptions. The right half of Figure 3.5 illustrates the relationship of exception handling to the Defect Model (§3.2, starting on page 77), which assesses the distributions of defect severities

resulting from each exception handling result, and combines those distributions to assess the total severities of each defect type.

VDT has no explicit product model; it neither explicitly represents the content of design decisions, nor directly forecasts the chance of creating defects. The thesis therefore provides an explicit Defect Model that infers the risk of engineering defects from VDT simulation exception handing assessments and from the way, as part of this thesis, modelers organize the VDT results. Specifically, the Development Model (VDT) estimates the amount of ignored rework demand during development, and subsequent models infer from that measure a corresponding risk of underperformance by developed products. §6.3.7 (on page 187) describes Defect Model extensions for analyzing other VDT-assessed risk indicators, such as the volume and distribution of backlogged work (which tends to elevate stress and increase the risk of errors).

The next section of this thesis explain defect creation, which is illustrated in the right half of Figure 3.5 Generic Event Tree Relating VDT-Assessed Exception Handling to Degree of Verification, Conformance Probability, and the Distribution of Engineering Defects (on page 73). The subsequent section explains how those defects' severities combine to determine total defect severities, as illustrated in Figure 3.6 (Generic Thesis-Assessed Distributions of Defect Severities Caused by Development Process Quality, on page 74).

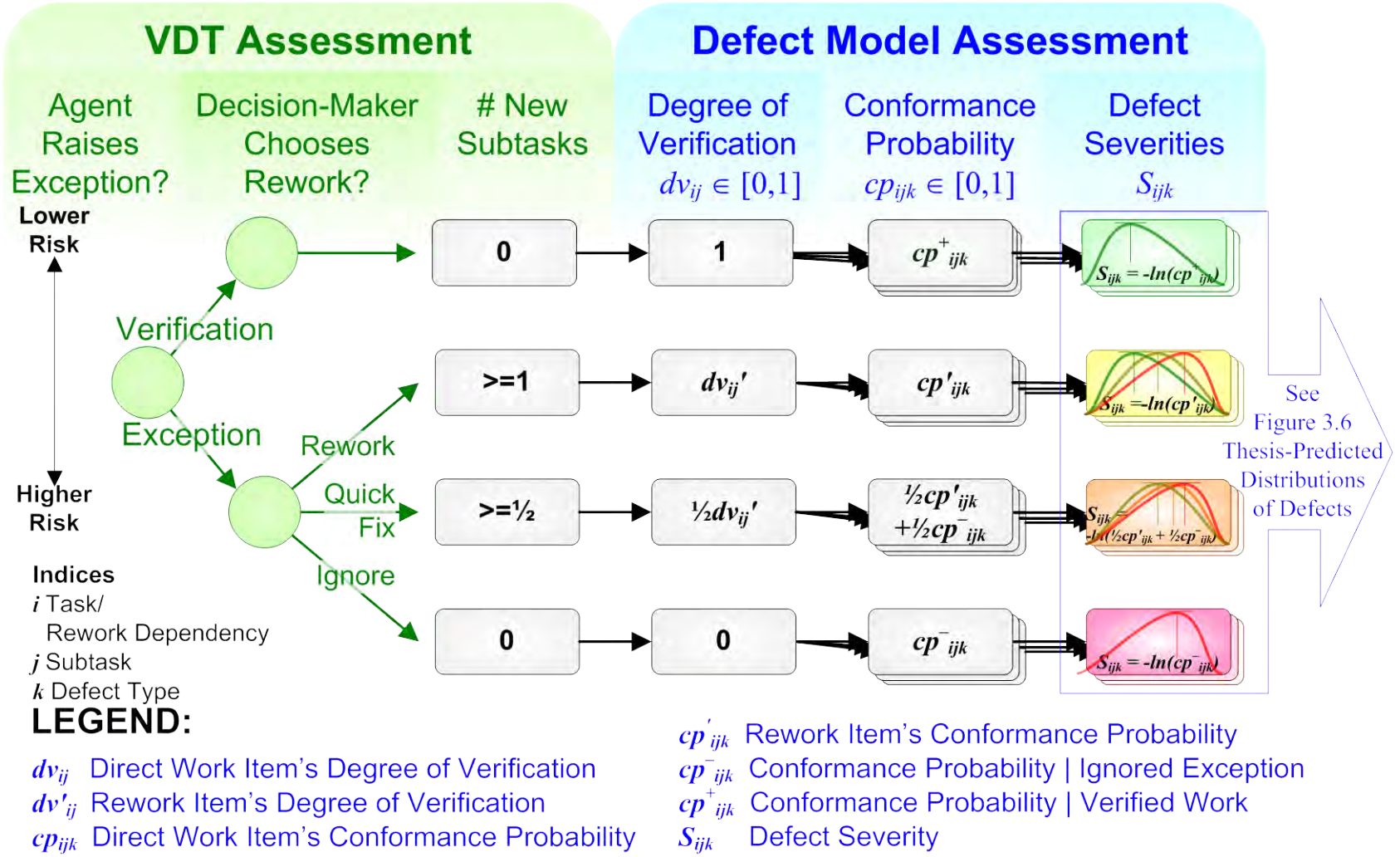


Figure 3.5 Generic Event Tree Relating VDT-Assessed Exception Handling to Degree of Verification, Conformance Probability, and the Distribution of Engineering Defects

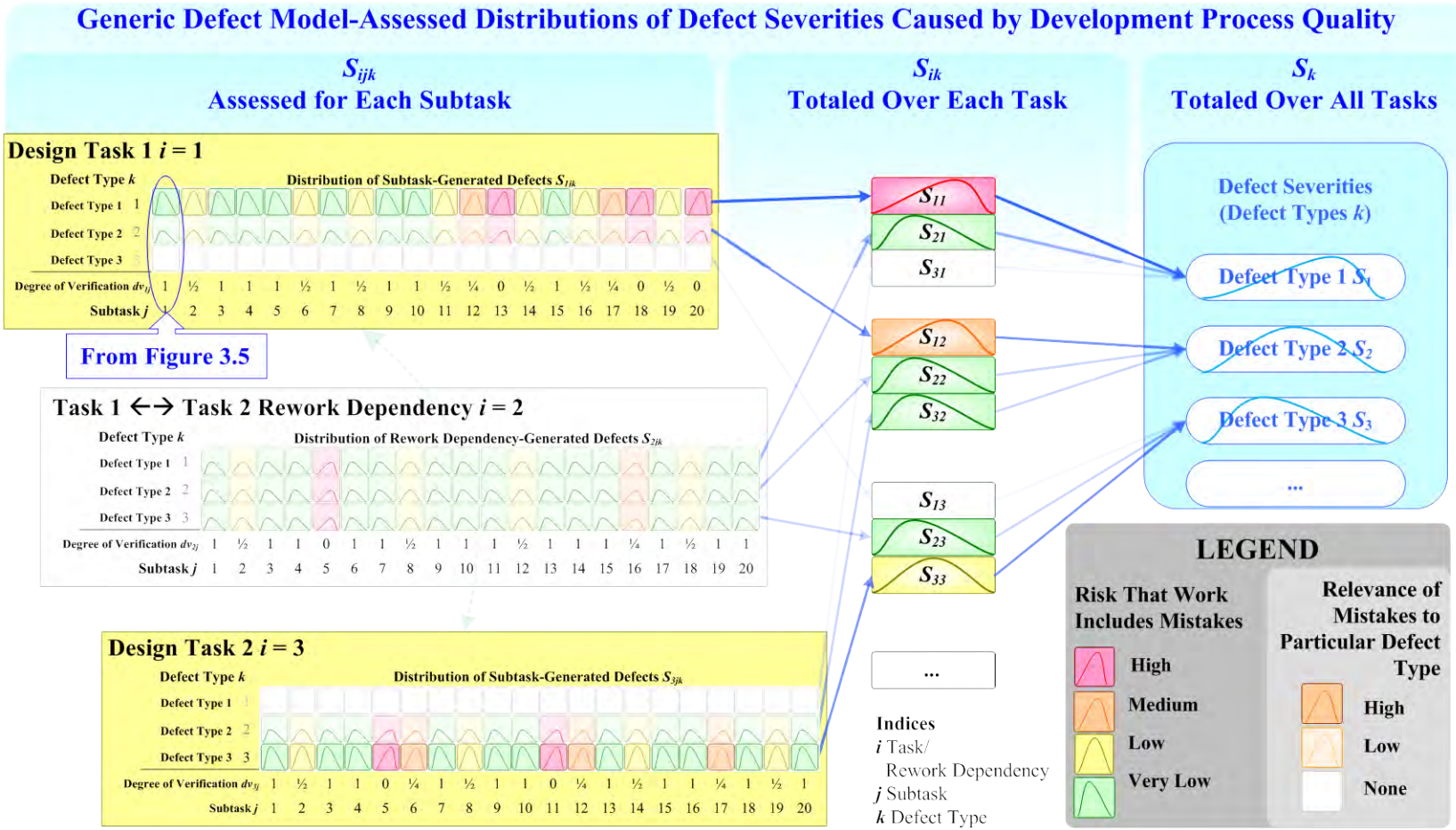


Figure 3.6 Generic Thesis-Assessed Distributions of Defect Severities Caused by Development Process Quality

VDT provides the following quantities for each subtask *j* of task or rework dependency *i*:

- wv_{ij} is the **input total volume (amount) of direct work scheduled** for subtask *j* of task or rework dependency *i* (A one-day subtask typically equals 8 hours' direct work),
- ev_{ij} is the **output total volume (amount) of work that generated exceptions** for subtask *j* of task or rework dependency *i* (An exception for that subtask equals 8 hours' exception volume), and
- rv_{ij} is the **output total volume (amount) of rework executed** for subtask *j* of task or rework dependency *i* (if the exception caused an ignore decision, the rework volume is zero)

Degree of Verification and Rework Deficit

The thesis derives two new measures of process quality from existing VDT output measures.

The Defect Model assumes that the likelihood of building defects into the product increases as exceptions occur that are not followed up by rework. The Defect Model introduces the ***degree of verification*** measure – the fraction of development work volume that was eventually verified– to assess the distribution of engineering defects. As a complementary measure, to aid intuition only, the thesis defines ***rework deficit*** as the fraction of work volume that caused ignored exceptions. The Development Model interprets VDT output for each development simulation by assessing the degree of verification for each subtask of the development tasks and rework dependencies. In the Defect Mode, the lower the degree of verification (the higher the rework deficit), the greater the expected severity of defects.

The thesis model uses VDT exception handling output to calculate the following quantities for each subtask *j* of task or rework dependency *i*:

- ***Degree of verification*** ≡ DV_{ij} Random variable (with realization dv_{ij}) representing the fraction of work, for subtask *j* of task or rework dependency *i*, ***that eventually resulted in verified work***
- ***Rework deficit*** ≡ 1 – dv_{ij} Fraction of work, for subtask *j* of task or rework dependency *i*, ***that raised exceptions without corresponding rework***

The thesis uses simulation to derive a discrete approximation to the theoretically continuous probability distributions of DV_{ij} (for reasons detailed in Appendix B, on page 196). The model bases that distribution (for each subtask *j* of task or rework dependency *i*) on representative samples of DV_{ij}. The following equation shows how the model derives those samples (denoted dv_{ij}) from the three pieces of VDT data defined in the previous section – the input volume of direct work (wv_{ij}), the output volume of corresponding exceptions simulated (ev_{ij}), and the output volume of rework competed because of those exceptions (rv_{ij}):

$$dv_{ij} = \frac{wv_{ij} - \left(ev_{ij} - rv_{ij}\right)}{wv_{ij}} \qquad \textbf{Eq. 3.1}$$

Over many simulation trials, these samples come to approximate the continuous distribution. For more about this method and its rationale, see Appendix B (on page 196).

Figure 3.5 (on page 73) presents the relationship between one subtask's exception handling results and its degree of verification using an event tree. For example, verified work items have degrees of verification 1, and work items ending in ignored exceptions have degree of verification 0. If subtask *j* of task or rework dependency *i* raises an exception and is reworked, then the rework item is quick-fixed (half-ignored, half-reworked), and then the (half-size) quick-fix item is verified, the degree of verification is $dv_{ij} = 1 \times dv_{ij}' = 1 \times \left(\frac{1}{2} \times dv_{ij}''\right) = 1 \times \left(\frac{1}{2} \times 1\right) = \frac{1}{2}$ (where dv_{ij}' denotes the degree of verification for a subtask created to represent rework for subtask *j* of task or

rework dependency *i*). Just half the work volume in the example resulted in verification.

Probabilistic Dependencies between Degrees of Verification

This chapter discusses probabilistic dependencies between variables because risks often depend upon multiple simultaneous events. Total system failures, for example, often occur due to multiple subsystem failures that have a common cause in development conditions (and the resulting, external events). Probabilistic dependencies first manifest within the Development Model results, and the PRA-VDT model preserves these dependencies through later stage analyses.

VDT simulates emergent events (such as the accumulation of agent backlog) that can affect *different work items within the same task* persistently over time. The effects of these uncertain factors manifest as probabilistic dependencies between the degrees of verification of subtasks *within each task*. The random variables DV_{ij} and DV_{ij+1}, for example, typically are probabilistically dependent.

VDT also simulates interactions between *work items on different tasks* during development (such as corresponding communications completion rates). The effects of these uncertain factors manifests as probabilistic dependencies between the degrees of verification of subtasks *in different tasks*. The random variables DV_{ij} and DV_{i+1j}, for example, typically are probabilistically dependent.

3.2 Defect Model

This section explains how the thesis assesses the distribution of engineering defects based on the simulated degrees of verification for development-stage knowledge work.

The development of every product component and interface begins with a specification of its intended behavior during operations. VDT simulates a range of

important development behaviors but includes no explicit means to estimate either the degree of compliance with this specification or the resulting influences on operations. The counterpart model in operations, PRA, can forecast a range of important operations behaviors, but has no specific means of assessing the likelihood of problems rooted in development. The Defect Model links VDT and PRA through a formal representation of ***engineering defects*** – shortcomings in development work that can influence behavior during operations.

This section explains how the Defect Model estimates the distribution of engineering defects based on development impacts (§3.1.3 on page 71). Figure 4.4 (on page 115) illustrates how first VDT estimates the degree of verification for development tasks and rework dependencies (green), then the Defect Model assesses the distribution of engineering defects (blue), and finally the Operations Model estimates the resulting operations performance (red). The Defect Model adds up the risk of defects that individual work items place on the product to determine a probability distribution on total engineering defects. The Defect Model provides the Operations Model's operations risk model with an assessed distribution of product shortcomings based on the Development Model's development impacts model.

Table 3.4 Defect Model Variables

Term	Name	Data	Source	Description
k	***Defect Type***	Index	Modeler	Identifies a category of defects that have the potential to influence operations behavior and (therefore) decision maker utility. k is used to subscript variables, introduced in this section and the next, that differ according to defect type.
cp^{-}_{ijk}	***Minimum Conformance Probability***	$\Re \in [0,1]$	Development Manager	Probability that development subtask j of task or rework dependency i causes zero defects of type k, **if the VDT-simulated work was totally unverified** (all work resulted in exceptions that were not reworked).
cp^{+}_{ijk}	***Maximum Conformance Probability***	$\Re \in [0,1]$	Development Manager	Probability that development subtask j of task or rework dependency i causes zero defects of type k, **if the VDT-simulated work was fully verified** (all exceptions were fully reworked).

Table 3.4 Defect Model Variables

Term	Name	Data	Source	Description
cp_{ijk}	***Estimated Conformance Probability***	$\Re \in [0,1]$	Thesis Product Model Output	Probability that development subtask *j* of task or rework dependency *i* causes zero defects, **given the VDT-simulated degree of verification**.
$E()$	***Expectation***	*Expectation*	Modeler Discretion	Average value of a distribution, weighted by probability
S_{ijk}	***Severity of Subtask Defects***	$\aleph \in [0,\infty)$	Thesis Product Model Output	Severity of type *k* defects that development subtask *j* of task or rework dependency *i* causes
S_{ik}	***Severity of Task Defects***	$\aleph \in [0,\infty)$	Thesis Product Model Output	Total severity of type *k* defects that development task or rework dependency *i* causes
S_k	***Severity of Defects***	$\aleph \in [0,\infty)$	Thesis Product Model Output	Total severity of type *k* defects that development causes

3.2.1 Defect Type Definitions

This section explains how the Defect Model identifies and describes the elements of developed products at significant risk of engineering defects. The Defect Model defines ***engineering defects*** in the developed product as violations of the ***development specification*** (declared goals) that have the potential to influence operations, and in turn, the decision maker's utility.

The Defect Model partitions the set of possible defects into categories called ***defect types***. Each defect type corresponds to a distinct portion of the development specification and has distinct causes and potential consequences. For each PRA-VDT simulation, the Defect Model estimates how many defects of each type result from the development work items.

The set of defect types should cover all the foreseeable defects exactly once and should use higher levels of abstraction, rather than excluding defect types, when modeling resources require compromises in analytic detail. One aid to determining the extent of coverage for a given set of defect types is to organize it into a hierarchical typology. Figure 3.7 (on page 80) provides example typology of potential defects created in the detailed design of a physical component.

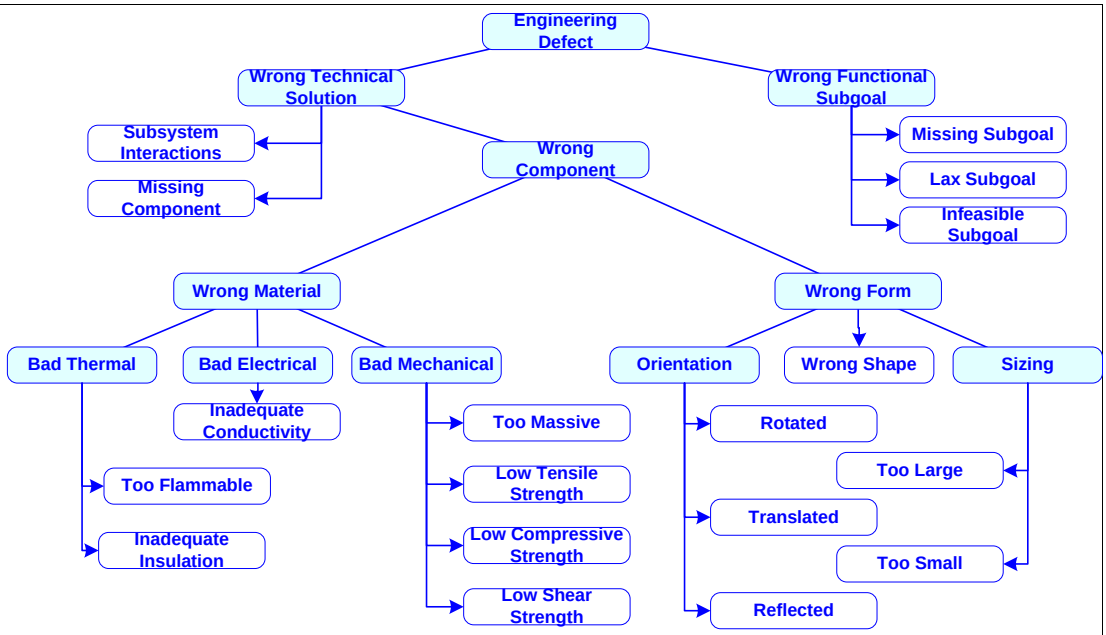


Figure 3.7 Generic Example of Engineering Defect Types for a 'Detailed Design' Stage

The Defect Model uses the index *k* to identify a defect type (category of defects). For example, in the satellite illustration, defects of type 1 (with index k =1) represent deviations from the Orbiter's specification. The severity of those defects (assessed by the Defect Model, as defined later in this section), and the influence these defects can have upon different components' failure probabilities (provided as input, as defined in the following section), are all subscripted by *k* (=1, for example, in the "Orbiter" case).

3.2.2 Conformance Probabilities

Each subtask of every task and rework link has the potential to create certain defects but does not have the potential to create others. The Defect Model models this distinction using a ***conformance probability*** – a chance of perfect compliance with a portion of the specification – corresponding to every possible pairing of subtask and defect type. This measure represents the probability the given work generates no defect of the given type, and it may equal one (certainty) if the pairing is between work and product items that are not directly related. Appendix B, on page 196, explains how the Defect Model preserves complex, indirect VDT- and PRA- assessed relationships among development and operations elements. This section explains how

the Defect Model estimates the conformance probabilities of each subtask of every development task or rework dependency. As input, this step uses the Development Model-assessed degree of verification and field inquiry-derived estimates of a product alternative's defect-generation potential.

Even the most expert real-world developers sometimes fail to satisfy their target specifications. Appendix A (on page 192) details how VDT agents correspondingly will attempt to verify work once it is complete and will sometimes rework items that are suspected of containing errors. In the thesis interpretation, a development agent raises exceptions if (and only if) it assumes that its work will lead to a defective product. This process is imperfect; "Verified" work sometimes creates defects, and unverified work sometimes does not.

The Defect Model **assumes** that only the final (possibly reworked) versions of product elements directly affect the probability that they are defective; In this model, when a decision maker orders rework for a subtask, the potential for defects in the rework subtask replaces the potential for defects in the original subtask that created the exception. Each subtask's final degree of verification (see §3.1.3) thus distinguishes the conformance probability levels for a given project simulation run. The event tree in Figure 3.5 (on page 73) illustrates the different implications of reworking an exception (which can sacrifice schedule and cost) and ignoring it (which can sacrifice quality or reliability).

The Defect Model defines conformance probability cp_{ijk} to be the modeled probability that work on subtask j of task or rework dependency i created no defect of type k in the product (regardless of whether the initial work satisfied the corresponding portion of the specification, or whether any initial defects were later replaced by defect-free rework).

The model estimates cp_{ijk} within a field-derived range of conformance probabilities. cp^{+}_{ijk} is the best-case reliability: the probability of being defect-free, given the

simulated work item was verified (degree of verification $dv_{ij} = 1$). Similarly, cp^{-}_{ijk} is the worst-case reliability: the probability of being defect-free, given the simulated work item raised an ignored exception ($dv_{ij} = 0$).

The estimated conformance probability cp_{ijk} is a convex combination (weighted average) of the best- and worst-case limits, weighted by the VDT-simulated work's degree of verification:

$$cp_{ijk} = cp^{-}_{ijk} + dv_{ij} \times \left(cp^{+}_{ijk} - cp^{-}_{ijk} \right) \qquad \textbf{Eq. 3.2}$$

The event tree in Figure 3.5 shows how the Defect Model derives cp_{ijk} from VDT assessments of verification level dv_{ij} and verification accuracy limits cp^{-}_{ijk} and cp^{+}_{ijk}.

Each work item directly addresses only a small portion of the total project specification. For example, a building's structural engineering work has no *direct* potential to create plumbing defects. In this case, work on task or dependency *i* subtask *j* never contributes to type *k* defects, so $cp^{+}_{ijk} = cp_{ijk} = cp^{-}_{ijk} = 100\%$.

3.2.3 Distribution of Defect Severities

This section explains how the Defect Model assesses each defect type's distribution of total severity (s_k) using the previously calculated conformance probabilities (cp_{ijk}). Each defect type's severity identifies the extent of a product's deviation from a corresponding specification. For example, severities would measure the number of undersized bolts for a "loose attachments" defect type, the size of a crack, or the number of bugs in a software program.

For each defect type *k*, the Defect Model first estimates a probability distribution of the severity of defects that correspond to each subtask. The Defect Model then combines these distributions to estimate the distribution by task or rework dependency and (finally) for the whole project. The Defect Model derives these estimates from

conformance probability (previous section) and the Operations Model uses them to forecast operations capacity (§3.3).

Formulating the Distribution of Defect Severity

The severity of a defect type can reflect many possible conditions, as assessed while developing the typology of defects (see §3.2.1 on page 79). For instance: the size of a crack is a positive-valued real number, the number of software bugs is natural number, and the number of loose attachments is an integer between zero and the total number of attachments in the design. Similarly, probabilistic dependencies between defects (as Appendix B, on page 196, details) also arise in accordance with the type of work and verification processes.

The range of possible distributions of defect severities is large and their use is subtle, therefore effective PRA-VDT modeling requires proficiency with probabilistic analysis (as in Law and Kelton 2000). Determining the shape of mathematical distribution that is best for modeling a defect type's severity requires knowledge and assumptions about different defects, and about how tasks generate them.

This subsection demonstrates the thesis method's generality and provides some guidance to modelers by introducing several distributions to model alternative conditions. The subsequent portion of this chapter demonstrates the method of analysis fully and formally on a subset of the introduced models. The sections below explain how in this subset, "memorylessness" is a good approximation, and design work generates a Poisson distribution of defects.

Boolean Defect Severity Levels (Defective or Conforming)

In some applications, a single specific defect is important enough to merit analysis that is separate from all others. With only one possible defect of a particular type, the probability of that defect resulting from a task is one minus the probability of zero

successes after a series of Bernoulli trials. The probability of success at each trial represents the prospect of creating the defect at that subtask, which is one minus the subtask's conformance probability cp_{ijk}.

Discrete, Finite Defect Severity Levels

In some applications, there is a large but finite number of possible defect severities. For example, the defect type "loose attachments" may have a severity value defined as the number of poorly-fitting bolts out of twelve total bolts. Binomial distributions can represent the total severity of a given type in the cases where there are a fixed number of possible defects of that type. In this model, the number of trials equals the maximum severity value, and the probability of success equals the product of conformance probabilities cp_{ijk} for all subtasks. As the number of possible defects increases, even though it remains finite, the Poisson model becomes an increasingly close **approximation** – especially when the *foreseeable* severity of defects is much smaller than the theoretical maximum (e.g., when cp_{ijk} is low).

Continuous-Valued Defect Severity Levels

An exponential distribution can elegantly model the real-valued severity of defects contributed by one subtask. Because the exponential (as well as gamma, the sum of exponentials) is continuous, modelers must define corresponding conformance probabilities cp_{ijk} as a probability of having severity below a certain threshold (rather than having severity zero). Using this model, the total severity of defects equals a sum of gamma distributions that are each parameterized by the number of subtasks resulting in a given degree of verification and by the conformance probability corresponding to that degree of verification.

Because normal distributions have positive density over the whole real axis, using one to describe defect severities would require either truncation or a definition of negative severity (the thesis addresses the latter prospect only as an extension, in §6.3.1,

Further PRA-VDT Justification, on page 181). The normal distribution is the limiting case for a Poisson distribution, however, as the mean goes to infinity. Therefore, to achieve bell-curved forms on the positive line, modelers can define a Poisson severity model (detailed below) with high mean severity and (commensurately) low defect influences.

Severity of Defects Corresponding to Subtasks

This section explains how the Defect Model uses the previous section's estimated conformance probability (probability of zero defects, cp_{ijk}) to identify a probability distribution of the number of each defect type that each subtask creates.

The Defect Model represents the assessed number of type *k* defects created by a subtask *j* of development task or dependency *i* using the random variable S_{ijk}. By definition, the S_{ijk} probability mass at zero equals the conformance probability cp_{ijk}:

$$cp_{ijk} = p(S_{ijk} = 0) \qquad \textbf{Eq. 3.3}$$

This thesis provides detailed examples of modeling defect creation *within each subtask* using a Poisson Process. The Poisson model best suits those cases where defects, of all types, are incremental (as in "The subtask created an error at this moment"), rather than absolute (as in "The subtask never completed the circuit"). In this model, probability of *each subtask* creating defects is the same after having just created an error as after having just created conforming work.

Figure 3.6 (on page 74) illustrates the calculation of Eq. 3.14. Development tasks and dependencies (large blocks at left) are composed of many subtasks (vertical slices). The VDT-simulated subtask degree of verification influences the thesis-assessed distribution of defects (indicated in Figure 3.6, on page 74, by green for few expected defects, red for many). Summing these distributions for subtasks within each task or dependency, and among tasks (according to their potential to create specific defects, indicated in the figure by lightness) determines the total distributions for each

engineering defect type (at right), and therefore operations capacity (see §3.3.1) and project utility (§3.4). For example, this emulation of the satellite project has low VDT-assessed degree of verification for payload design, resulting in more assessed payload defects (§3.3).

Probabilistic Dependencies among Subtask-Generated Defect Distributions (on page 90) explains the various dependencies between defects created by subtasks.

The Poisson model **assumes** that creating a defect while working on a subtask does not affect the probability of creating another defect later in that subtask. This condition constitutes ***memorylessness***, or conditional independence between the occurrences of defects. The only type of continuous distribution for durations between defect generation points that satisfies this condition is the exponential, and the only type of distribution for the total severity (number) of defects that a subtask creates is the Poisson [Law and Kelton 2000]. The thesis also **assumes** that the probability of creating a defect while working at the beginning of a subtask is the same as the probability while working at the end, although that condition is not required for memorylessness.

Models of the target industries typically support the memoryless property when there are a large number of potential defects, high cp_{ijk}, and low exception probabilities.

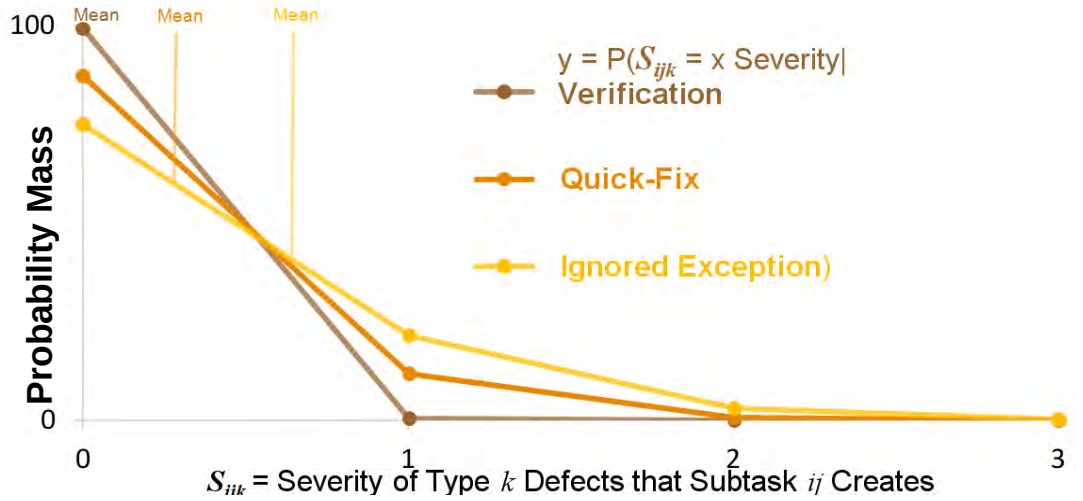


Figure 3.8 Defect Model-Assessed Distributions of Defects resulting from Hypothetical Subtasks with Varying Degrees of Verification

The probability of sampling y from a poisson distribution having mean x [Law and Kelton 2000] is:

$$p(Poisson(x) = y) = \frac{e^{-x} \times x^{y}}{y!} \quad \textbf{Eq. 3.4}$$

Substituting the definition of conformance provided in Eq. 3.3 into the above equation provides an equation for the probability of conformance:

$$cp_{ijk} = \frac{e^{-E(S_{ijk})} \times E(S_{ijk})^{0}}{0!} \quad \textbf{Eq. 3.5}$$

Solving the above equation produces a formula for $E(S_{ijk})$, the expected severity of type k defects created by a subtask j of development task or dependency i:

$$cp_{ijk} = e^{-E(S_{ijk})} \quad \textbf{Eq. 3.6}$$

$$\ln(cp_{ijk}) = \ln(e^{-E(S_{ijk})}) \quad \textbf{Eq. 3.7}$$

$$-\ln(cp_{ijk}) = E(S_{ijk}) \quad \textbf{Eq. 3.8}$$

Reasserting that the expectation in Eq. 3.8 is for a Poisson distribution compactly states the assessed marginal distribution of S_{ijk}, the severity of type k defects created by a subtask j of development task or dependency i:

$$S_{ijk} \sim Poisson\left\lfloor -\ln(cp_{ijk}) \right\rfloor \qquad \textbf{Eq. 3.9}$$

Substituting the definition of conformance probability (cp_{ijk}) provided in Eq. 3.2 on page 82 provides the following model of the Poisson distribution of defect severity:

$$S_{ijk} \sim Poisson\left\{ -\ln\left[cp^{-}_{ijk} + dv_{ij} \times \left(cp^{+}_{ijk} - cp^{-}_{ijk} \right) \right] \right\} \qquad \textbf{Eq. 3.10}$$

The above equation shows that when memorylessness holds for a given defect type, *the conformance probability uniquely determines the probability of each severity level*.

Figure 3.5 (on page 73) reviews how the PRA-VDT Framework links assessed development processes (VDT output) to engineering defects, and Eq. 3.10 defines the model formally. The Defect Model views the VDT assessments for subtask j of task i using degree of verification dv_{ij} and rework item verification level dv_{ij}' (see §3.1.3). In turn, this indexes conformance probability cp_{ijk} (§3.2.2), which is defined as the probability of creating zero type k defects (§3.2.3). The Poisson distribution of engineering defects S_{ijk} is minimized when the agent's work is verified (either initially or as rework), and it is maximized when an exception occurs that is not followed by corresponding rework. Figure 4.7 (on page 124) shows how the Defect Model combines all of the S_{ijk} values to determine the total severity of type k defects, S_k.

Knowing the distribution of defects assessed to result from specific subtasks aids model calibration and forensic analysis, but higher levels of abstraction better serve intuition for most prospective analyses. This section derives distributions of the total severity of defects that results from each work task or rework dependency, and the next section aggregates to the total number that result overall from development.

Total Severity of Defects of Each Type

This section next explains how the Poisson Defect Model derives a probability distribution of the total severity of each type of defect that a complete task or dependency creates. The Poisson Defect Model **assumes** engineering defect creation is memoryless (based on conditions provided at this section's outset), therefore in the model all subtasks affect the number of engineering defects independently (given Development Model results).

The Defect Model-estimated distribution of type k defects that task or rework dependency i creates, denoted S_{ik}, is the sum of corresponding defect severities S_{ijk} over all subtasks j:

$$S_{ik} = \sum_{j \in \{task \quad or \quad dependency \quad i \quad subtasks\}} S_{ijk} \qquad \textbf{Eq. 3.11}$$

Knowing the distribution of defect severities assessed to result from specific tasks or dependencies aids intuition as well as model calibration and forensic analysis. However, in the Operations Model (§3.3) it is the total (project-wide) severity of each defect type that influences operations performance. This section explains how the Defect Model derives a probability distribution of the total severity of each defect type that a development project creates. The Poisson Defect Model-estimated distribution of defect severity S_k for each type k equals the sum of corresponding defect severities S_{ik} over all tasks and dependencies i:

$$S_k = \sum_i S_{ik} \qquad \textbf{Eq. 3.12}$$

Substituting into Equation Eq. 3.12 the formula for S_{ik} from Eq. 3.11 provides the severity of type k defects in terms of subtask-generated defect severities:

$$S_k = \sum_i \sum_j S_{ijk} \qquad \textbf{Eq. 3.13}$$

Substituting into Eq. 3.13 the formula for S_{ijk} (from Eq. 3.9) states the distribution of type *k* defect severities in terms of subtask conformance probabilities cp_{ijk}:

$$S_k \sim \sum_i \sum_j Poisson\left[-\ln(cp_{ijk})\right] \qquad \textbf{Eq. 3.14}$$

Figure 4.7 Thesis-Assessed Distributions of Defects Caused by Development Process Quality (on page 124) illustrates the calculation of Eq. 3.14. In the figure, development tasks and dependencies (large blocks at left) consist of many subtasks (vertical slices). VDT assesses for each subtask a degree of verification that influences the thesis-assessed distribution of defect severities (indicated in the figure by color: green for few expected defects, red for many). Summing these distributions for subtasks within each task or dependency, and among tasks (according to their potential to create specific defects, indicated in the figure by lightness) determines the total distributions for each engineering defect type's severity (at right), and therefore operations capacity (see §3.3) and project utility (§3.4). For example, this model of the satellite project has low VDT-assessed degree of verification for payload design, resulting in more severe payload defects and higher component failure risk (§3.3).

Probabilistic Dependencies among Subtask-Generated Defect Distributions

The conformance probabilities for different tasks and subtasks, cp_{ijk}, are probabilistically dependent because they are directly related (by Eq. 3.2 on page 82) to the probabilistically dependent degrees of verification assessed by VDT. The marginal distributions of defect severities generated by different tasks and subtasks, S_{ijk}, are also probabilistically dependent because they are directly related (by Eq. 3.9 on page 88) to the conformance probabilities.

The Poisson analysis in this thesis **assumes** that the two VDT-modeled dependencies described in the previous paragraph are the only ones linking work items. More precisely, this model **assumes** the existence of one engineering defect is not relevant

to the existence of any other potential defects *given the VDT output*. Equivalently, this model **assumes** that, for a set of VDT-assessed development impacts corresponding to a single simulation trial, defects are ***memoryless*** in that they result from work items according to conditionally independent distributions.

Simplified Calculation of Poisson-Distributed Defects

Two simplifications can accelerate the assessment of total Poisson defect severity in Eq. 3.14 (on page 90) for analyses that need not distinguish which subtask created each defect. Accelerating the calculations is important because it enables larger numbers of simulation trials and, therefore, greater precision.

First, observe that the Defect Model-assessed total severity of defect of each task or dependency is Poisson distributed because the sum of Poisson distributions is also Poisson distributed. This observation permits the following description of total type *k* defect severity (S_k) that requires just one sample from a Poisson distribution (whereas Eq. 3.14 requires one sample per combination of task and subtask):

$$S_k \sim Poisson\left[\sum_i \sum_j -\ln\left(cp_{ijk}\right)\right] \qquad \textbf{Eq. 3.15}$$

Next, note that the sum of the logarithms (having the same base) of several values equals the logarithm of the product of those values. The following alternative to Eq. 3.15 requires just one calculation of logarithm per defect type, rather than one per combination of task, subtask, and defect type:

$$S_k \sim Poisson\left[-\ln \prod_i \prod_j cp_{ijk}\right] \qquad \textbf{Eq. 3.16}$$

Figure 3.9 charts the distributions of defect severities for a hypothetical project's tasks as the fraction of subtasks verified decreases from 95% (dark brown) to 85% (medium

brown) to 75% (light brown). The Defect Model sums the distributions of defect severities generated by tasks and dependencies to determine the total distributions of severity (for each defect type) generated by the project.

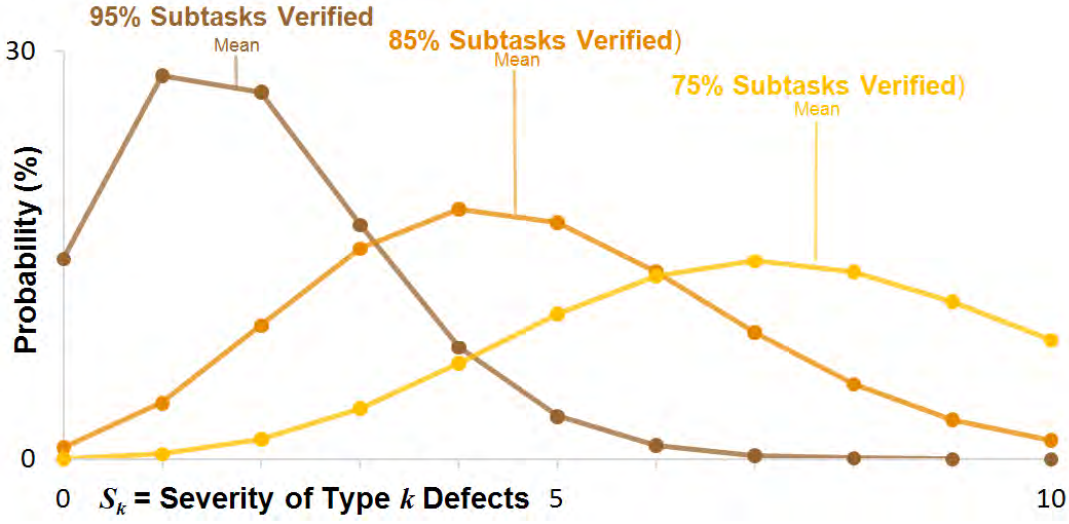


Figure 3.9 Defect Model-Assessed Poisson Distributions of Defects resulting from Hypothetical Tasks with Varying Degrees of Verification

Probabilistic Dependencies between Engineering Defect Severities

The PRA-VDT framework samples the S_k probability distributions in order to preserve the *joint distribution's* probabilistic dependencies resulting from the emergent degrees of development verification (simulated by VDT). It illustrates the model structure nevertheless to state the *marginal distributions* of engineering defects by substituting Eq. 3.2, the formula for conformance probability, into Eq. 3.16, the formula for total defect severities:

$$S_k \sim Poisson\left\{-\ln \prod_i \prod_j \left[cp_{ijk} = cp^-_{ijk} + dv_{ij} \times \left(cp^+_{ijk} - cp^-_{ijk}\right)\right]\right\} \qquad \textbf{Eq. 3.17}$$

Thus, the thesis models the distributions of defect severities using VDT-assessed degrees of verification for tasks and dependencies (dv_{ij}) and using conformance probability limits (cp_{ijk}^- and cp_{ijk}^+). Because this equation for each defect type k includes the degrees of verifications for each i, the model conserves the VDT-assessed

dependency between different tasks' performance. The equation also maintains the dependencies for each subtask *j*, thereby conserving the VDT-assessed *dependency over time of performance within individual tasks* in a simulation trial.

3.3 Operations Model

The thesis assesses the distribution of operations-stage, physical and functional capacities, and the resulting operations outcomes, based on the Defect Model-assessed number of engineering defects.

The PRA-VDT Framework's Operations Model uses the Defect Model-assessed severities of engineering defects to estimate the product's reliability. §2.2.3 (on page 33) introduced the model's principal method, Probabilistic Risk Analysis (PRA).

This section formally defines the Operations Model and illustrates it using the satellite example. The first goal of this section is to impart an intuition for how the PRA-VDT Framework uses PRA and other math models to estimate the distribution of component failures in a project's operations phase. The second goal is to introduce the Operations Model's toolkit using mathematical precision, without limiting the toolkit's flexibility. The last goal is to illustrate the method's power using the satellite example.

Table 3.5 Operations Model Variables

Term	Name	Units	Source	Description
l	***Operations Function Capacity Index***	Index	Modeler	Identifies a portion of the ability to perform or to resist failure during operations
oc_l^+	***Best Operations Capacity***	Real Calibration Constant	Operations Manager	Capacity resulting from a product with no impacting defects
oc_l^-	***Worst Operations Capacity***	Real Calibration Constant	Operations Manager	Capacity resulting from a product with the maximized number of impacting defects
OC_l	***Operating Capacity***	Stochastic Process	Operations Model Output	Ability of function l to perform or resist failure during operations
oc_l	***Operating Capacity***	Sample Path of OC_l	Operations Model Output	Ability of function l to perform or resist failure during operations (realization)
di_{kl}	***Defect Influence***	Real	Operations	Marginal effect that each unit of type k

Table 3.5 Operations Model Variables

Term	Name	Units	Source	Description
		Calibration Constant	Manager	engineering defect severity has on operations capacity l
n	***Operations Behavior Index***	Index	Modeler	Identifies a subset of the events that occur as operations capacity and load interact
F_{system}	***System Failure***	Event	Modeler	Failure of a system of components
F_n $F_{1,2}$	***Component Event***	Event	Modeler	Failure of a component
OB_n	***Operations Behavior***	Real	Operations Manager	Subset of events that occur as operations capacity and load interact (fixed in time)
ob_n	***Operations Behavior***	Realization of OB_n	Operations Model Output	Subset of events that occur as capacity and load interact (realization, fixed in time)
E	***External Event***	Event	Modeler	External event that affects multiple components' failure probabilities

3.3.1 Operations Alternatives

The Operations Model describes the distributions of component- and total failures that will manifest using operations capacity, a measure of potential behavior within a context of uncontrolled forces in operations.

Operations Capacity

Defining Capacity

Operations capacity identifies the distribution of responses that developed products will exhibit when operated under the range of foreseen circumstances (operating contexts). Operating capacity characterizes a product's contingent response to its operating context. In the simplest models, operations capacity simply represents a component's failure probability, although in extensions (see §6.3 on page 181) it can represent peak performance (productivity) as well as reliability (resistance to failure).

Engineering defects (see §3.2 starting on page 77) influence impacts of interest to the decision maker by reducing capacity over the period of operations. The severities of

defects relevant to a particular operating capacity influence where, between best and worst case limits, it is likely to be.

This thesis does not comprehensively address the full range of possible relationships between defects and capacities. Instead, like Pugnetti 1997, this thesis presents its method using a single intuitively clear and plausible model in which defects have independent, multiplicative (geometric) effects on capacity.

Operations Capacity Range

For each capacity attribute indexed in *l*, the Operations Model requires field inquiries (such as expert interviews or evaluation of statistics) to ascertain oc_l^+, the failure probability that results in the best case, when there are no relevant engineering defects. Similarly, the model requires a capacity level oc_l^- that manifests in the worst case, when the (foreseeable) relevant defects are at their highest severities.

For each VDT simulation trial, the Defect Model estimates an operating capacity oc_l that is a weighted average of the limits, oc_l^- and oc_l^+. The weight is a function of the severities of defects (s_k) and the influences they have over the operations capacity (di_{kl}, see below).

Defect Influences

Defect influences di_{kl} define the incremental effects that each unit of severity in type *k* defects will have on a capacity indexed in *l*. A value of di_{kl} = 100% indicates defects of type k have no effect on capacity *l*, whereas a value of 50% indicates that the presence of any type k defects cuts capacity *l* in half, and a value of 0% reduces capacity *l* to its minimum value oc_l^-.

The PRA-VDT modeler must derive influence values (in conjunction with capacity limits) from field inquiry, using methods such as expert interviews, evaluation of statistics, and theory (e.g., engineering) modeling.

Calculating Operations Capacities

Eq. 3.18 (below) calculates operating capacity in a manner that mathematically parallels the calculation for conformance probability (see Eq. 3.2 on page 82).

$$oc_l = oc_l^- + \left(oc_l^+ - oc_l^-\right) \times \prod_k \left(di_{kl}\right)^{s_k} \qquad \textbf{Eq. 3.18}$$

Figure 3.10 Illustrative Operations Capacity oc_l as a Function of Assessed Total Defect Severity s_k for Hypothetical Data (on page 97) shows that the conditional independence approximation in this model implies that operations capacities fall geometrically (within their limits) as defect severity increases.

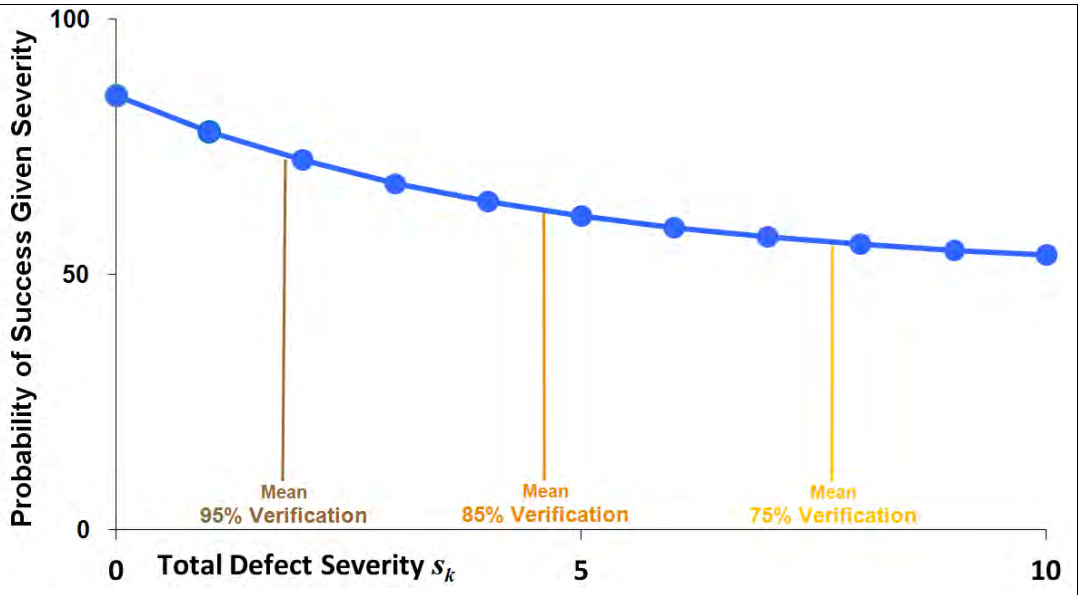


Figure 3.10 Illustrative Operations Capacity oc_l as a Function of Assessed Total Defect Severity s_k for Hypothetical Data

Probabilistic Dependencies between Operations Capacities

The PRA-VDT framework simulates the OC_l probability distributions in order to preserve the joint distribution's probabilistic dependencies resulting from the emergent degree of development verification (simulated by VDT) and the occurrence of defects (determined by the Defect Model).

It illustrates the model structure nevertheless to state the marginal distributions of operations capacities by substituting Eq. 3.16 into Eq. 3.18:

$$OC_l \sim oc_l^{-} + \left(oc_l^{+} - oc_l^{-}\right) \times \prod_k di_{kl}^{\,Poisson\left\{-\ln \prod_i \prod_j \left[cp^{-}_{ijk} + dv_{ij} \times \left(cp^{+}_{ijk} - cp^{-}_{ijk}\right)\right]\right\}} \qquad \textbf{Eq. 3.19}$$

The above equation shows that the Defect Model assesses the capacity as a function of capacity limits (oc_l^{-} and oc_l^{+}), VDT-assessed degrees of task and dependency verification (dv_{ij}), conformance probability limits (cp_{ijk}^{-} and cp_{ijk}^{+}), and defect influences (di_{kl}).

3.3.2 Operations Behaviors

The Operations Model estimates the distribution of operations behaviors – events that will result from using the developed product within the given context. The model can incorporate a broad range of techniques to match defects' project-specific dynamics and important events. This section explains (and illustrates) how the Operations Model compares the operations capacity and load measures, and forecasts operations behavior, using PRA models of subsystems and total system lifetime.

The Operations Model describes operations behaviors, indexed by n, as random variables OB_n with realizations ob_n.

Probabilistic Dependencies between Operations Behaviors

The PRA-VDT framework simulates these equations in order to preserve the full *joint distributions* without removing probabilistic dependencies. Specifically, each trial of Eq. 3.16 (on page 91) uses a single sample from the Poisson distribution to assess the severity (number) of defects S_k of each type k, so the calculation preserves dependencies between operations that can fail due to root causes in the same defects.

The mathematical formulae for *marginal distributions* nevertheless reveal how the method preserves those dependencies. Substituting the marginal distribution of engineering defects S_k from Eq. 3.17 provides the following formula, which expresses the probability of failure in terms of degrees of verification dv_{ij} assessed by VDT in the Development Model:

$$p(OB_n = Failure) = p\left[\prod_k di_{kn}^{\,Poisson\left\{-\ln \prod_i \prod_j \left[cp^-_{ijk} + dv_{ij} \times \left(cp^+_{ijk} - cp^-_{ijk}\right)\right]\right\}} < \frac{\left(ol_n - oc^-_n\right)}{\left(oc^+_n - oc^-_n\right)}\right] \qquad \textbf{Eq. 3.20}$$

Eq. 3.20 explains how the PRA-VDT framework assesses the probability of failure is a function of VDT-assessed degrees of task and dependency verification (dv_{ij}), conformance probability limits (cp_{ijk}^- and cp_{ijk}^+), defect severities (ds_{kn}), operations

capacity limits (oc_n^- and oc_n^+), and operations load (ob_n). Because (for each defect type *k*) Eq. 3.20 includes the degrees of verifications for each *i*, the model conserves the VDT-assessed dependency between different tasks' performance. Because simulating the equation also maintains the dependencies for each subtask *j*, the model conserves the VDT-assessed dependency over time of performance for individual tasks in a simulation run.

PRA Using Functional Block Diagrams

PRA calculates the probability of complex system failures F_{system} as a function of failures F_n in components indexed by *n*. The following equation states this purpose formally:

$$p(F_{system}) = PRA[p(F_1), p(F_2),...] \quad \textbf{Eq. 3.21}$$

This section assumes that component failure probabilities are independent, and the following section on external events removes that assumption.

PRA's functional block diagrams are a compact means of defining ***minimal cut sets*** – combinations of ***component failures*** that are necessary and sufficient to cause ***system failure*** in the integrated whole. The following calculation of total system failure probability accounts for the possibility of double- or triple-counting simultaneously-occurring failure modes by removing the appropriate number of duplicates (" - *doubles* + *triples…*"). When failures are rare, modelers often omit those terms (this is the "rare event approximation").

$$p(F_{system}) = \sum p(FailureModes) - doubles + triples - ... \quad \textbf{Eq. 3.22}$$

Figure 3.11, Basic Functional Block Diagrams (on page 100), provides three basic configurations of functional block diagrams. PRA analyses combine blocks

representing components in series and in parallel to describe highly complex systems intuitively and formally.

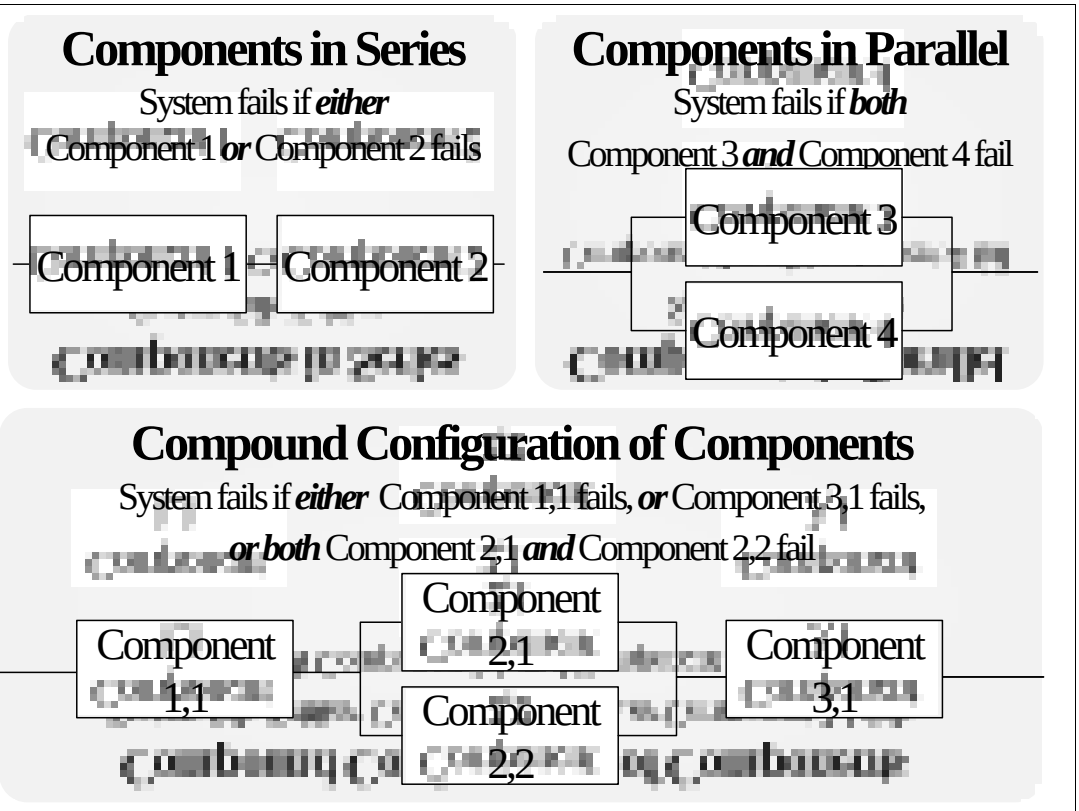


Figure 3.11 Basic Functional Block Diagrams

For ***components in series*** (systems requiring all components to operate), the probability of failure equals the sum of independent failure probabilities over all subsystems, adjusted to prevent double counting in cases of simultaneous failures (or triple counting…). The following formula defines total system failure as a function of probabilistically independent subsystem failures:

$$SerialCase: p(F_{system}) = \sum_{components\ n} p(F_n) - doubles + triples - ... \qquad \textbf{Eq. 3.23}$$

For example, Figure 3.11's upper left diagram describes the failure probability formula:

$$p(F_{system}) = p(F_1) + p(F_2) - p(F_1, F_2) \qquad \textbf{Eq. 3.24}$$

For ***components in parallel*** (systems requiring at least one component to operate), the probability of failure equals the product of all component failure probabilities. The following formula defines total system failure as a function of probabilistically independent subsystem failures:

$$ParallelCase: p(F_{system}) = \prod_{components\ n} p(F_n) \quad \textbf{Eq. 3.25}$$

For example, Figure 3.11's upper right diagram describes the failure probability formula:

$$p(F_{system}) = [p(F_3) \times p(F_4)] \quad \textbf{Eq. 3.26}$$

Functional block diagrams easily describe compound systems containing both serial and parallel subsystems. For example, the lower diagram in Figure 3.11 describes a slightly more complex system with failure probability:

$$p(F_{system}) = p(F_{1,1}) + [p(F_{2,1}) \times p(F_{2,2})] + p(F_{3,1}) - doubles + triples \quad \textbf{Eq. 3.27}$$

The next chapter defines the prospect of satellite failure using a six- to nine-block diagram, and the subsequent chapter defines various degrees of failure in a dormitory's electrical system using a diagram of approximately fifty blocks.

Modeling Time to Failure

In models of operations over time, *ob* identifies the time of total system failure. Representing subsystem failures as independent stochastic processes OB_n, indexed by component *n*, and with realizations ob_n, the serial and parallel cases are:

$$SerialCase: ob = \min_n (ob_n) \quad \textbf{Eq. 3.28}$$

$$ParallelCase: ob = \max_n (ob_n) \quad \textbf{Eq. 3.29}$$

The time to failure for more complex systems is easily calculated by composing the above two functions, using the same method as the previous section explained.

Modeling External Events such as Engineering Defect Severities

One of the most common methods of risk reduction, component redundancy, relies upon independence between component failures. However, external events can reduce the effectiveness of that strategy by causing components to disproportionately fail at the same time.

Functional block diagrams do not explicitly represent external events' effects on subsystem failures. From the total probability formula [Law and Kelton 2000], the calculation of total system failure probability in the presence of external event *E* is:

$$p(F_{system}) = p(E) \times PRA\left[p(F_1 \mid E), p(F_2 \mid E), ...\right] + p(\neg E) \times PRA\left[p(F_1 \mid \neg E), p(F_2 \mid \neg E), ...\right] \quad \textbf{Eq. 3.30}$$

PRA-VDT views engineering defect severities as external events that impact the failure probabilities for multiple components. The probabilistic dependencies these external events carry over from events in development significantly impact the total failure probabilities of complex systems.

For example, in the case where there is only one Poisson-distributed defect type, the total failure probability is:

$$p(F_{system}) = \sum_{x=0}^{\infty} p(S_1 = x) \times PRA\left[p(F_1 \mid S_1 = x), p(F_2 \mid S_1 = x), ...\right] \quad \textbf{Eq. 3.31}$$

PRA-VDT similarly views engineering team performance as another source of probabilistic dependency, operating through the degree of verification measure.

3.3.3 Operations Impacts

The Operations Model estimates the distribution of ***operations impacts***: those uncertain (but clearly distinguished) results of operations that directly interest the decision maker. Examples of operations impacts include launch failure, or failure at a particular time during operations. In extensions to the model (see §6.3.5, on page 187) these may include profits for a business venture, scientific discoveries for a space mission, and lives lost for an emergency management system. The Operations Model assessment of operations impacts rests on traditional probabilistic analysis of the assessed operations behaviors' relationship to the decision-maker utility. The Decision Model (see §3.4), rather than the Operations Model, originates those quantities that are personal to a decision maker, such as risk attitude, discount rates, or the relative preferences among possible impacts.

3.4 Decision Model

This section explains how the Decision Model first models the set of available alternatives, then uses the Development, Defect, and Operations Models to provide information about the implications of each alternative, then recommends the best choice given a decision maker's preferences.

The purpose of PRA-VDT is to help decision-makers choose among alternative plans for projects that include both development and operations phases. In particular, PRA-VDT focuses on decisions affecting the organization, process, product, and operating context. For example, PRA-VDT can assess which of two alternative engineering organizations (one that is centralized and hierarchical versus another that is decentralized and flat) would minimize a particular project's risk of defect-influenced failures during operations. In general, decisions with narrow effects may warrant using simple methods (as discussed in Chapter 2, Existing Practice and Theory, on page 21), and decisions with broad effects may warrant using extensions to PRA-VDT (such as those discussed in §6.3, on page 181).

The method integrates analyses from several specialized models to support a Decision Analysis (DA). The Decision Model first models the set of available alternatives, then uses the VDT, Thesis, and PRA methods to provide information about the implications of each alternative, then finally recommends the best choice for a given decision maker's preferences.

The first decision element is a selected set of alternative plans for the development and operations efforts. Some of these alternatives involve development only, such as the number of engineers to assign to each development task or the amount of schedule overlap. Other sets of alternatives directly influence operations only, such as the location of a physical plant. Finally, some sets of alternatives may directly affect both development and operations, such as the inclusion of redundancy in physical systems (which tends to increase development complexity while reducing the risk of technical failure).

The Decision Model uses the VDT, PRA, and Defect Models to determine how each choice of alternative project influences the distribution of important behaviors in all phases, including product performance, technical failures, development project length, and development project cost.

The Decision Model then determines total project benefit by interpreting assessed impacts in the context of decision-maker preferences, taking into account the value of safety, the discount rate of money, and attitude toward risk. The Rational Decision Making model recommends choosing the alternative that leads to greatest expected benefits.

VDT provides samples of a distribution that has unknown form, rather than a mathematically defined joint probability distribution, and the integrated framework may call for mathematics of arbitrary complexity. The thesis therefore solves the integrated system by simulating one sample path through the Defect, Operations, and Decision Models for each VDT output value generated in the Development Model.

Table 3.6 Decision Model Variables

Term	Name	Data	Source	Description
A	***Project alternatives***	***Set***	***Decision maker***	Set of all consistent choices *a* for three project elements: development, product, and operations
a	***Project alternative***	***Element***	***Decision maker***	Consistent choice for three project elements: development, product, and operations
DA	***Development alternatives***	***Set***	***Development manager***	Set of possible choices for the development phase organization, process, and culture
a_d	***Development alternative***	***Element***	***Development manager***	Individual choice for the development phase organization, process, and culture
PA	***Product alternatives***	***Set***	***Development manager***	Set of possible choices for the deployment of developed products to operations, including their potential defects
a_p	***Product alternative***	***Element***	***Development manager***	Individual choice for the deployment of developed products to operations, including their potential defects
OA	***Operations alternatives***	***Set***	***Operations manager***	Set of possible choices for the use of developed products (within a context) to create value, including failure modes and redundancies
a_o	***Operations alternative***	***Element***	***Operations manager***	Individual choice for the use of developed products (within a context) to create value, including failure modes and redundancies
$B(a)$	***Project Behavior***	***Set of Events***	***Decision maker***	Set of development, product, and operations events.
$DV(a_d)$	***Degrees of Verification***	***Random Variable***	***Development Model***	Distribution of degrees of verification that occur during development guided by plan a_d Fraction of each task and subtask that was verified (created no exception). Elements are dv_{ij} (see §3.1.2)
dv	***Degrees of Verification***	***Realization of DV***	***Development Model***	Events of interest that actually do occur during development guided by plan a_d
$S(dv, a_p)$	***Defect Severities***	***Random Variable***	***Defect Model***	Defines the distribution of engineering defect severities in a product based on alternative a_p and developed with degrees of verification *dv*
s	***Product Description***	***Realization of S***	***Defect Model***	Defines the actual resulting engineering defect severities of a product
$O(s, a_o)$	***Operations***	***Random***	***PRA Operations***	Defines the distribution of degree of

Table 3.6 Decision Model Variables

Term	Name	Data	Source	Description
	Behaviors	***Variable***	***Model***	failure as a product operates, based on alternative a_o , and based on defect severities *s*
o	***Operations Behavior***	***Realization of O***	***Operations Model***	Actual degree of failure when a product operates
$f(X=x)$	***Probability Density***	***Function***	***Modeler Discretion***	Measure of the likelihood a continuous random variable *X* takes value (infinitesimally close to) *x*
$p(X=x)$	***Probability Mass***	***Function***	***Modeler Discretion***	Measure of the likelihood a discrete random variable *X* takes value *x*
ρ	***Discount Rate***	***Fraction***	***Decision maker***	Amount by which dollars in the future should be discounted to support a given decision-maker
$u(o)$	***Utility***	***Utility***	***Decision maker***	Measures the desirability of operations outcome *o* to a decision maker.
$E()$	***Expectation***	***Expectation***	***Modeler Discretion***	Average value of a distribution, weighted by probability
t	***Project Sim Trials***	***Natural Number***	***Modeler Discretion***	Number of trials used to estimate the full distribution of project behaviors
r	***Project Random Seed***	***Integer***	***Modeler Discretion***	Number provided to simulate project and generate a sample behavior
dv_{ar}	***Sampled Development Behavior***	***Realization of D***	***Development Model Output***	Events of interest that occur during development guided by project plan *a*, sampled using random seed *s*
s_{ar}	***Sampled Product Description***	***Realization of P***	***Thesis Product Model Output***	Attributes of interest that hold for a product guided by project plan *a*, sampled using random seed *s*
o_{ar}	***Sampled Operations Behavior***	***Realization of O***	***Operations Model Output***	Events of interest that occur during operations guided by project plan *a*, sampled using random seed *s*

3.4.1 Project Alternatives

Each project alternative identifies a consistent set of choices for the project's product, organization, and process that affect the distribution of possible results.

In Eq. 3.32 the Decision Model defines the ***project alternatives*** *A* as the set of all consistent choices *a* for three project elements:

$$A = \{a\} = \{(a_d \in DA, a_p \in PA, a_p \in OA) \mid a_d, a_p, \textit{and } a_o \textit{ are mutually consistent}\} \subseteq DA \times PA \times OA \quad \textbf{Eq. 3.32}$$

where

development alternatives $DA = \{a_d\}$ defines the organization, process, and culture of product creation,

product alternatives $PA = \{a_p\}$ defines the deployment of developed products to operations, including the potential defects, and

operations alternatives $OA = \{a_o\}$ defines the possible results from using developed products to create value.

3.4.2 Project Performance

The PRA-VDT Framework focuses on project behaviors that are ***material*** – that the decision maker either directly (like the cost of engineering labor) or indirectly (like the creation of defects that can influence downstream operations costs). The PRA-VDT Framework partitions these behaviors into development, product, and operations behaviors, according to the earliest possible time of observation.

For each available alternative, the Decision Model estimates the distribution of these types of resulting behavior:

Development behavior defines the events of interest during product design and creation as the degrees of verification for completed work $DV(a_d)$.

Product description defines the developed product's salient features as severities of engineering defects represented using $S(dv, a_p)$.

Operations behavior defines the total value created or destroyed by operating the developed product. It is a random variable $O(s, a_o)$ with distribution assessed by the

Operations Model based on engineering defects *s* and the chosen operations alternative a_o. Note that, given engineering defect severity *s*, the degree of development process validation *dv* is not relevant to operations behavior.

§3.1.2, §3.2.2, and §3.3.2 detail these definitions.

Project behavior *B(a)* maps an alternative to a ternary vector that indicates the Development, Defect, and Operations Models' assessments:

$$B(a) = B\left[\left(a_d, a_p, a_o\right)\right] = \left\lfloor DV\left(a_d\right), \quad S\left\lfloor DV\left(a_d\right), a_p\right\rfloor, \quad O\left\{S\left\lfloor DV\left(a_d\right), a_p\right\rfloor, a_o\right\}\right\rceil \qquad \textbf{Eq. 3.33}$$

For random variable *X* and realization *x*, the thesis defines the probability density function as $f(X=x)$ for continuous *X* and the probability mass as $p(X=x)$ for discrete *X*. The thesis defines development and operations behaviors as continuous, and product behaviors as discrete. The PRA-VDT Framework **assumes** each phase's alternatives are not directly relevant to the distribution of behaviors from other phases, so that:

$$f\left\{B\left[\left(a_d, a_p, a_o\right)\right] = \left(D, P, O\right)\right\} = f\left\{DV\left(a_d\right) = dv\right\} \times p\left[S\left(dv, a_p\right) = s\right] \times f\left[O\left(s, a_o\right) = o\right] \qquad \textbf{Eq. 3.34}$$

Maximizing Expected Utility

The PRA-VDT Framework supports an individual's decisions by assessing how each prospect's impacts differentially influence a personal measure of desirability called ***utility***. In general, Decision Analysis (DA, introduced in §2.2.5 on page 37) derives utility functions by discerning diverse attributes of interest using structured conversations with a decision maker [Matheson and Howard 1968].

This dissertation only considers the operations-stage degree of success or failure in the utility function. The Decision Model defines a ***utility function*** $u(o)$ that determines the relative desirability of a project outcome using only the operations success, failure, or partial failure outcome *o*.

When choosing between plans assessed to cause different distributions of possible impacts, ***rational decision makers*** (see §2.2.5 on page 37) will adopt the plan that maximizes expected utility. The normative ideal of rational choice is a foundation of classical economics and of Decision Analysis, and therefore of the PRA-VDT Decision Model.

PRA-VDT models the influence of putting alternatives into action using the Development, Product, and Operations Models. The Decision Model recommends using the alternative plan $a = (a_d, a_p, a_o)$ that maximizes:

$$E\{u[B(a)]\} = \int_{dv} \sum_{pd} \int_{o} f\{B[(a_d, a_p, a_o)] = (dv, s, o)\} \times u[(o)]\, ddv\, do \qquad \textbf{Eq. 3.35}$$

$$E\{u[B(a)]\} = \int_{dv} \sum_{pd} \int_{o} f[D(a_d) = dv] \times p[P(dv, a_p) = s] \times f[O(s, a_o) = o] \times u[(o)]\, ddv\, do \qquad \textbf{Eq. 3.36}$$

Each previous section of this chapter developed one of the terms in Eq. 3.36. §3.1 explained how the Development Model estimates the first term, the distribution of development impacts given the organizational and procedural plan for development. §3.2 explained how the Defect Model estimates the essential product parameters using the product plan and impacts from development. §3.3 explained how the Operations Model estimates the distribution of impacts from operations by analyzing the impacts from development, the essential product features, and the operational plan.

Chapter 4
Satellite Illustration

This chapter illustrates the PRA-VDT method on a hypothetical satellite project that has alternatives with different degrees of complexity and component redundancy.

This chapter exercises all of the PRA-VDT framework's major features by applying the method to the design and operations stages of a hypothetical satellite design project. The illustrative satellite model's data and structure are not derived from any particular real-world project, but are instead devised merely to clarify the structure of the model.

The hypothetical, illustrative satellite case's decision maker wishes to build and deploy a communications satellite, and must choose between three alternative project plans. The three hypothetical project alternatives are alike in design of the development organization, but they differ in development process task complexities and their operations- stage fault trees. The Low Redundancy product alternative is to build a satellite that is typical of those already fielded, that requires little technical innovation, and that offers moderate operational reliability. The Medium Redundancy alternative is to field a novel design with doubly redundant payload subsystems. The High Redundancy alternative is to field a cutting-edge design with quadruply redundant payload subsystems. The more redundant alternatives present greater engineering challenges, but also offer greater resilience to payload component failures

because the satellite requires only one fully-functioning payload component to operate. Figure 4.1 (below) formally defines the differences between the three alternatives by stating the projects' development task complexities and operations fault trees.

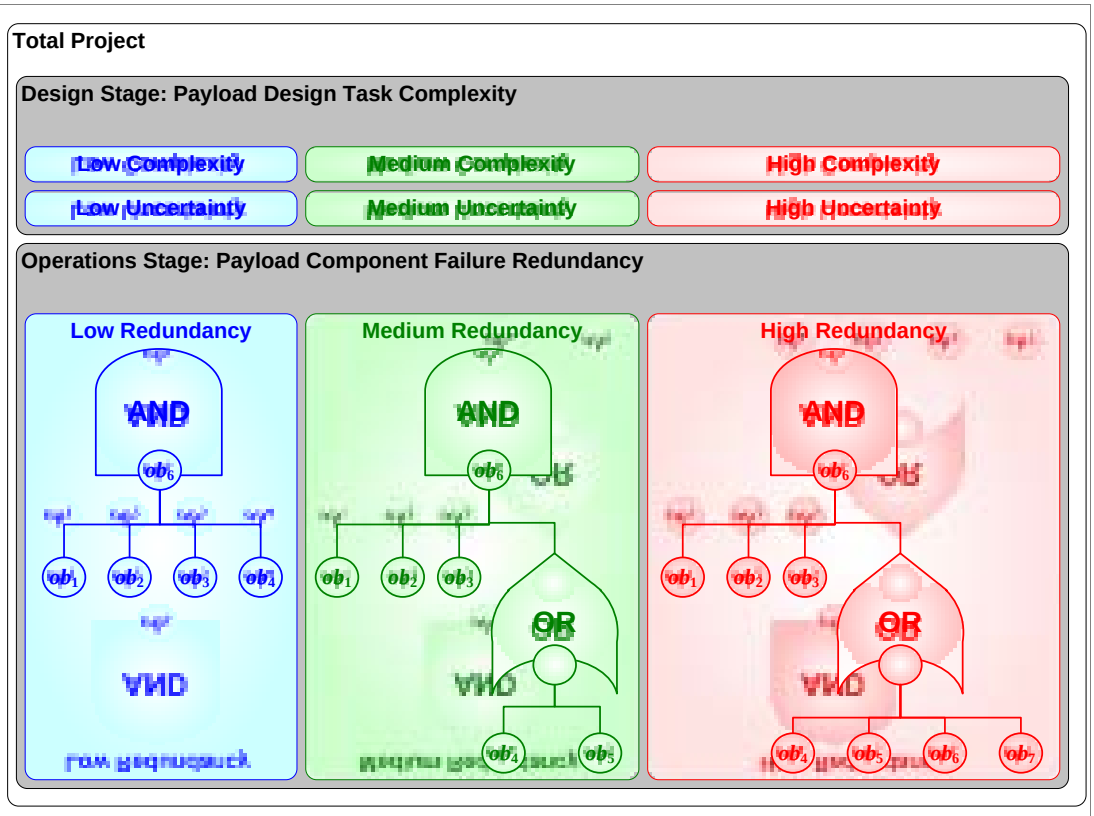


Figure 4.1 Illustrative Satellite Project's Three Cases have Different Design-Stage Complexities and Uncertainties, and have Different Operations-Stage Failure Redundancies

Figure 4.2 Index Structure of the PRA-VDT Model for the Illustrative Satellite Project (on page 113) shows how development tasks relate to product components, and how product components relate to possible defects of different severities. Figure 4.3 Event Tree for the Illustrative Satellite Project's PRA-VDT Model (on page 114) provides the first step towards formal failure probability analysis by identifying and structuring the key uncertainties (and probabilistic dependencies) of project performance. A principal function of this chapter is to explain how these relationships can be defined and operationalized formally to support project optimization.

Figure 4.4 Detailed Schematic of the PRA-VDT Model for the Illustrative Satellite Project (on page 115) maps the entities and relationships for the Medium Redundancy satellite example used throughout this chapter. The illustration's simple Development Model (green) compares the capabilities of three organizational agents against the corresponding requirements placed by two dependent tasks, and assesses the distributions of work verification. The Defect Model (blue) derives the numbers of each defect type based on development results. The Operations Model (red) uses the severity of defects to estimate distributions of component failures, and resulting total satellite lifetime. The time to failure informs the Decision Model, which identifies the best of three alternatives: one of a singly-, doubly- (shown), or quadruply- redundant payload subsystems that have increasing design complexity but greater ability to withstand payload component failures.

Failure dependencies are important and complex phenomena that emerge from the integrated model's structure. In most practical cases, models that fail to capture probabilistic dependencies between failures underestimate the probabilities of failure [Davoudian et al 1994.1, 1994.2]. Wherever the detailed schematic (in Figure 4.4) includes a path from one defect type to multiple components (as is the case with Payload components), the Defect Model-assessed severity of defects includes a probabilistic dependency of component failures due to a shared design. Similarly, wherever the diagram includes a path from one task to multiple components, the VDT-simulated quality of the development process (though the Defect Model) includes a probabilistic dependency of component failures due to a shared design team. For example, defects in the Orbiter and Support components are dependent; they are likely to occur together because they result from work on the same development task. This chapter includes several discussions of the different types and sources of dependencies, and of how PRA-VDT mathematically expresses these relationships. Appendix B (on page 196) provides a detailed discussion of how PRA-VDT handles probabilistic dependencies using simulation.

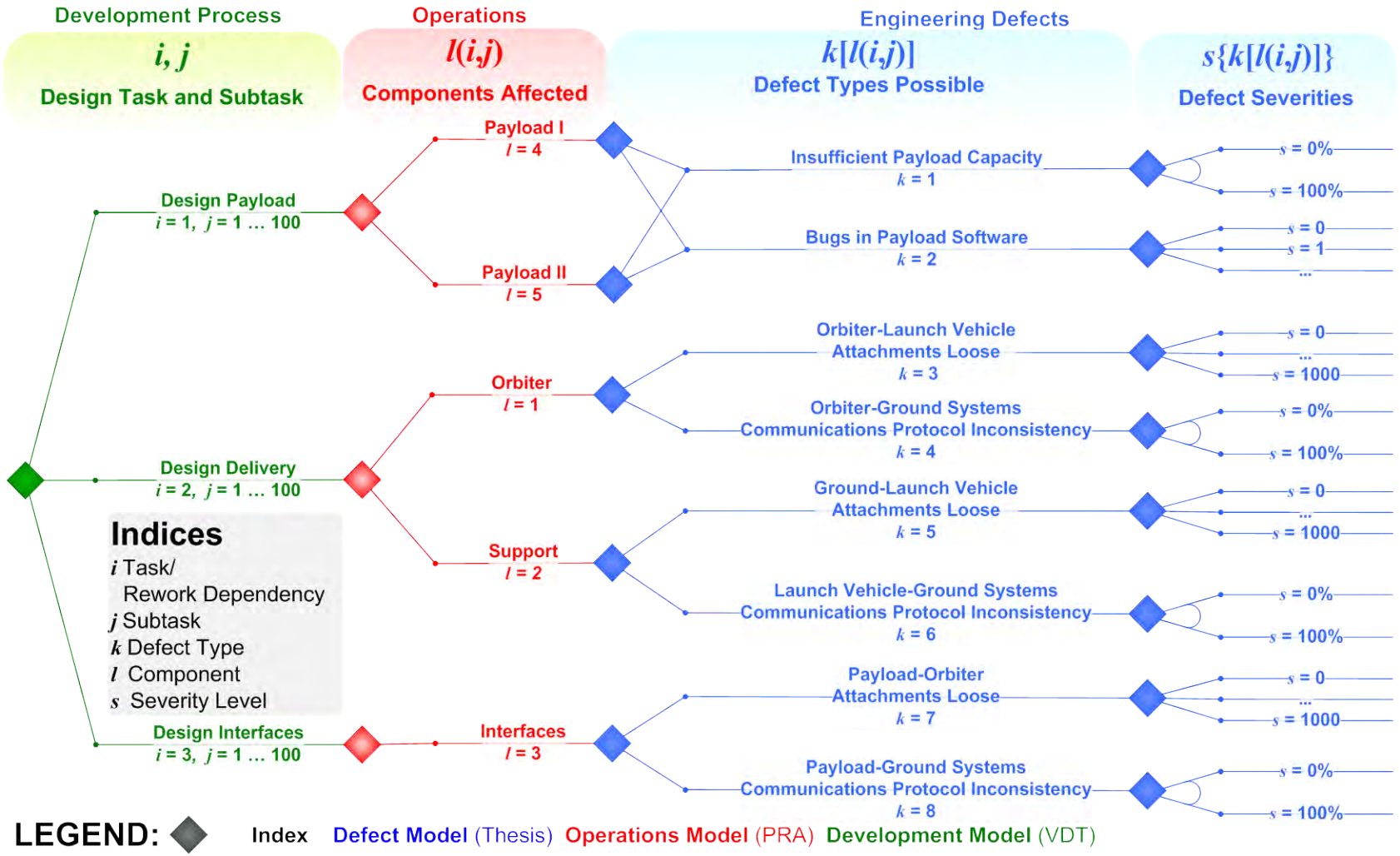


Figure 4.2 Index Structure of the PRA-VDT Model for the Illustrative Satellite Project

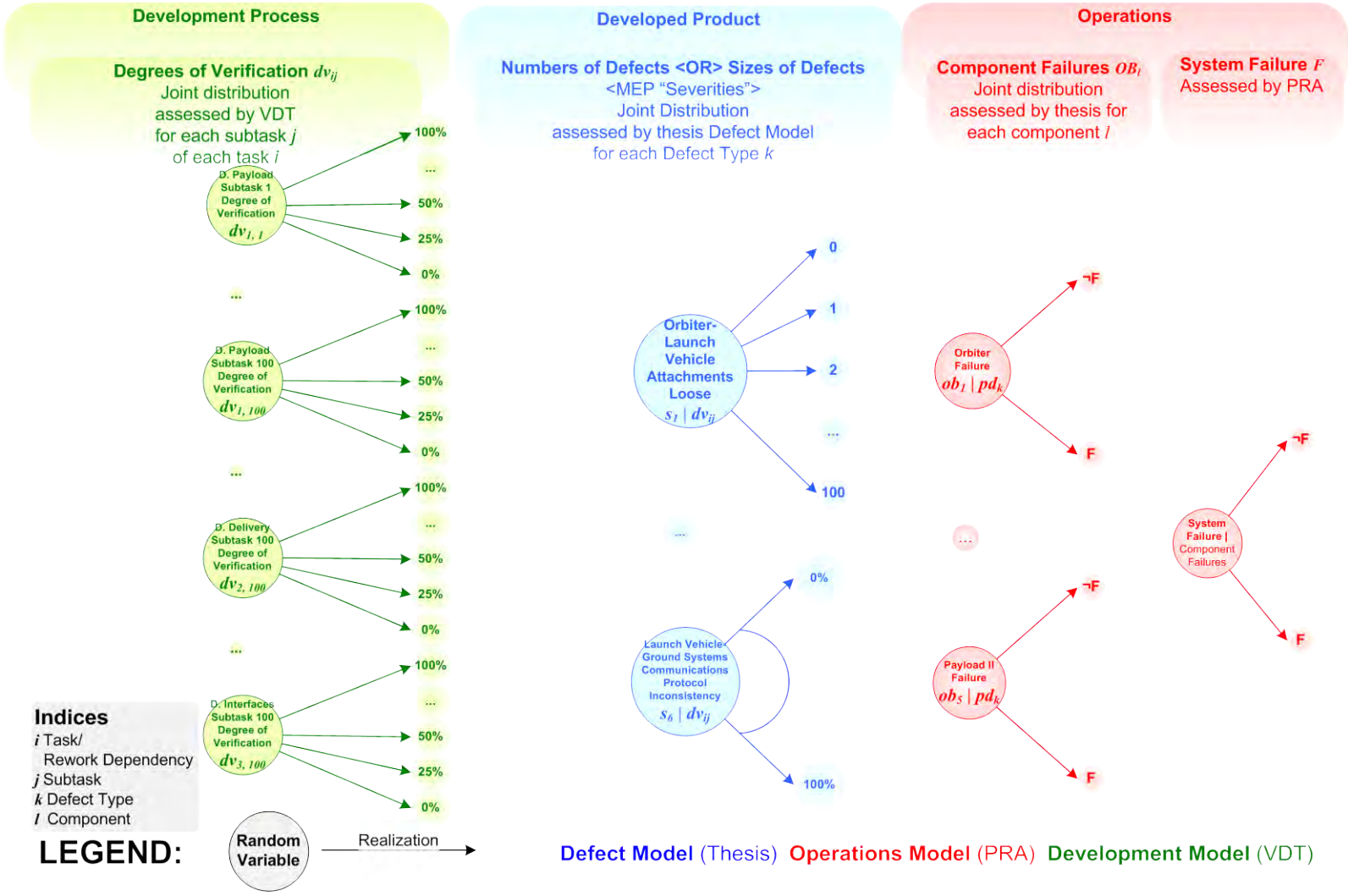


Figure 4.3 Event Tree for the Illustrative Satellite Project's PRA-VDT Model

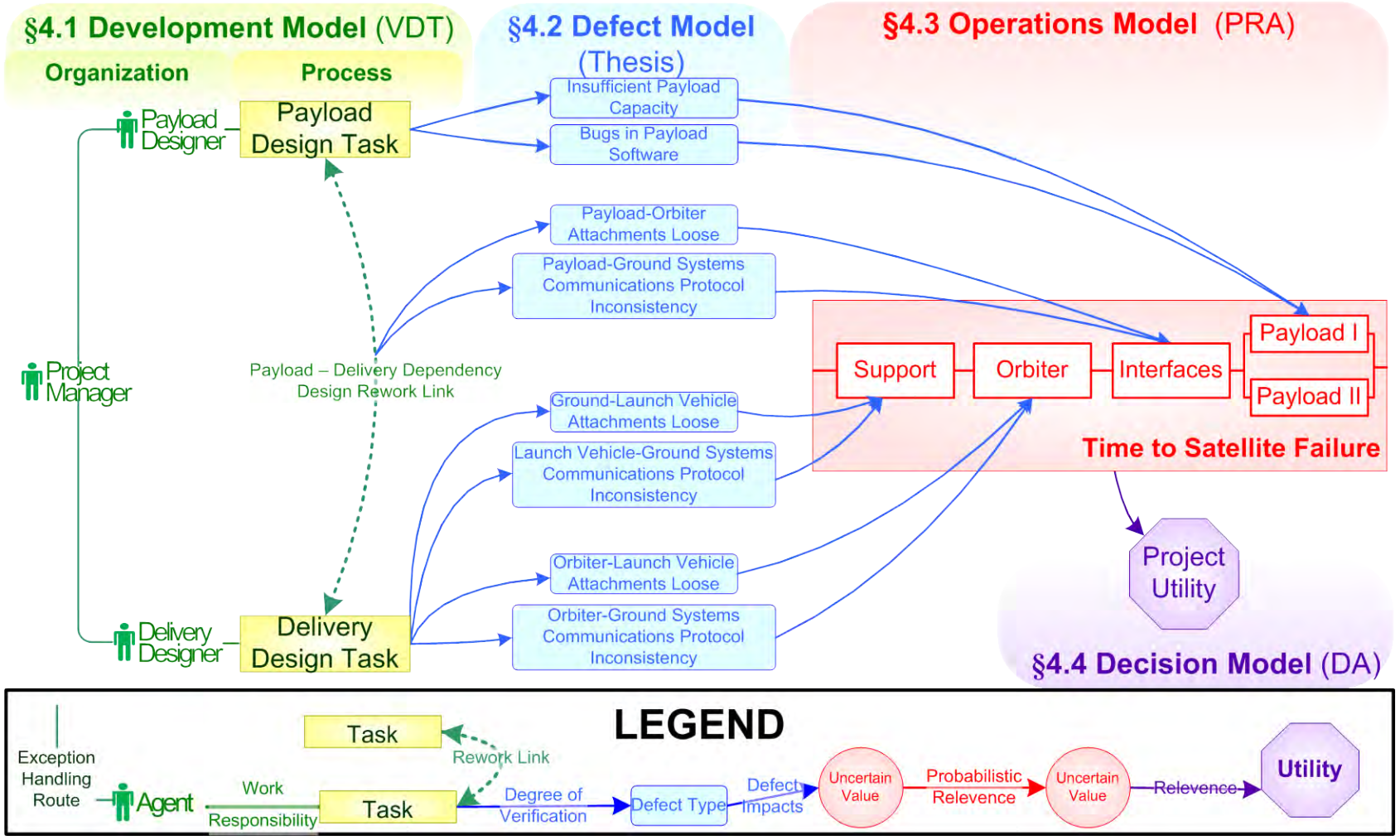


Figure 4.4 Detailed Schematic of the PRA-VDT Model for the Illustrative Satellite Project (Medium Redundancy alternative)

4.1 Development Model

This section introduces a satellite development project in which a project manager oversees execution of two interdependent engineering tasks: Payload Design and Delivery Design. Delivery Design includes design of the launch vehicle, orbiter, and support (ground control and launch) systems, whereas Payload Design includes design of the revenue-generating "business end" of the satellite. The thesis defines a new measure, degree of verification (dv_{ij}), to characterize the process quality that VDT simulates for each subtask (indexed *j*) that comprises one of the tasks or rework dependencies (indexed *i*) in the VDT development process model.

4.1.1 Development Alternatives

Input Development Organization

Consider the left region of Figure 4.4 (on page 115). The satellite design stage's organization model (far left) includes delivery and payload design agents, supervised by a project manager agent. The designers dedicate their full-time effort to corresponding tasks (mid left), and a payload design can cause rework in the delivery design, and vice versa (green arrow). Both tasks in the Low Redundancy alternative have low complexity and uncertainty values, while the Medium and High Redundancy alternative's tasks have medium and high values respectively. Assessments about the manner of work conduct (exception handling in particular) inform assessments of the distribution of engineering defects (see §3.2.1).

Table 4.1 and Table 4.2 describe the development organization and culture. The three alternatives (High-, Medium-, and Low-Redundancy) have identical organizations and cultures (although, as the next section explains, the three alternatives have different task properties).

Table 4.1 Illustrative Development Organization Input to VDT						
Agent Name	**Supervisor**	**Skill**	**Experience**	**Role**	**FTEs**	**Salary**
Payload Designer	Project Manager	Payload Design, Medium	Medium	Subteam	1	300
Delivery Designer	Project Manager	Delivery Design, Medium	Medium	Subteam	1	300
Project Manager				Project Manager	1	300

Table 4.2 Illustrative Development Project Culture Input to VDT						
Project Error Rate	Functional Error Rate	Communication Rate	Noise Rate	Centralization	Formalization	Matrix Strength
0.1	0.1	0.1	0.1	Medium	Medium	Medium

Input Development Process

Figure 3.5 (on page 73) illustrates and Table 4.3 describes the satellite project's development tasks and dependency. The High Redundancy alternative's Design Payload task has High uncertainty, requirement complexity, and solution complexity, whereas the Medium Redundancy alternative's has Medium values and the Low Redundancy alternative's has Low values. In all other respects, the three alternatives are identical. The satellite illustration includes no scheduled meetings (the Green Dorm illustration, in the next chapter, includes two meetings).

Table 4.3 Illustrative Development Load: Tasks				
i	**Task Name**	**Subtasks |{*j*}|**	**Primary Development Responsibility**	**Uncertainty, Requirement- and Solution-Complexities (Alternative)**
1	Payload Design	100	Payload Designer	**Low (**Low Redundancy**)** **Medium (**Medium Redundancy**)** **High (**High Redundancy**)**
2	Delivery Design	100	Delivery Designer	**Low (**Low Redundancy**)** **Medium (**Medium Redundancy**)** **High (**High Redundancy**)**
3	Payload → Delivery Dependency	100		**Low (**Low Redundancy**)** **Medium (**Medium Redundancy**)** **High (**High Redundancy**)**

Output Degrees of Verification

Figure 4.5 (below) charts the Cumulative Probability Distribution Function of all three tasks' degrees of verification, averaged over all their subtasks, based on output from 1000 VDT trials of the Development Model (calculated using Eq. 3.1, on page 76). In the figure, color distinguishes the alternatives: Low Redundancy project in blue, Medium Redundancy in green, and High Redundancy in red. Shape indicates task, with circles indicating the Payload task results, squares indicating Delivery results, and diamonds indicating results for the rework dependency between tasks.

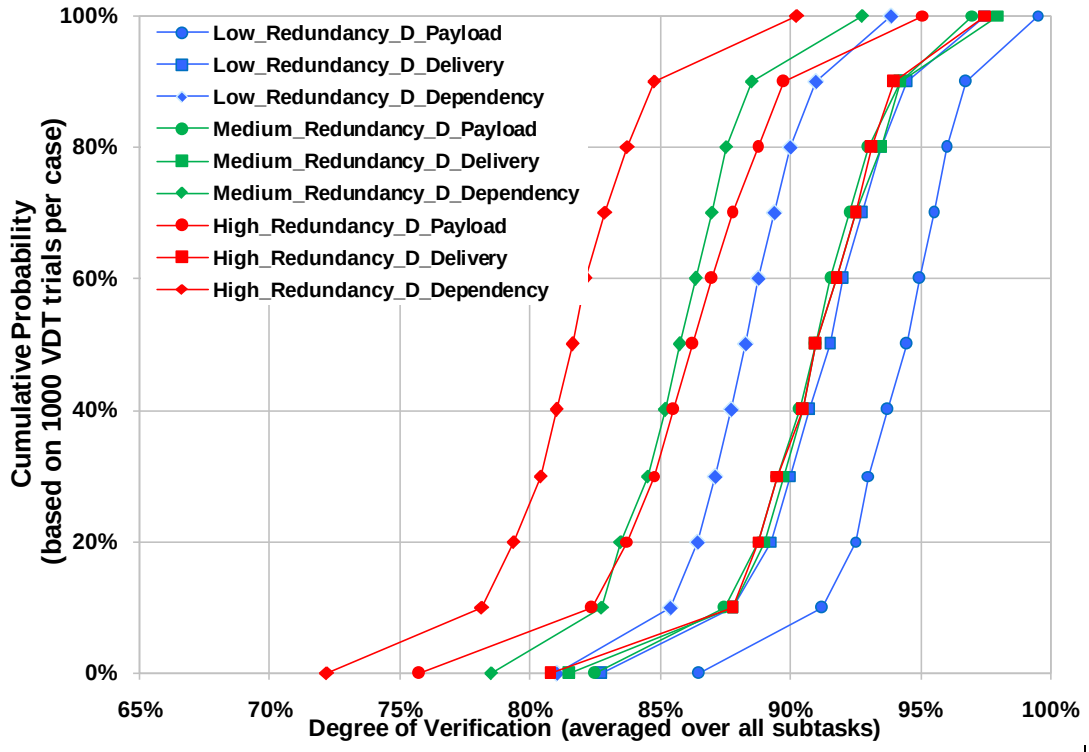


Figure 4.5 VDT Assessment of Satellite Design Work's Degrees of Verification dv_{ij}

(Cumulative Distribution Function based on percentiles of 1000 VDT trials)

In each of the three alternatives, the degree of verification for the tasks (D_Payload, with index i = 1, illustrated using circles, and D_Delivery, with index i = 2, illustrated using squares) is higher than that for the rework dependency (D_Dependency, with index i = 3, illustrated using diamonds). In the Low Redundancy alternative

(illustrated in blue), where the payload design has low complexity, the Payload has highest degree of verification. In contrast, for the Medium Redundancy alternative (green) the Payload and Delivery tasks have equal degrees of verification, and for the High Redundancy alternative (red) the Payload task has highest degree of verification. In this simple example, the higher complexity tasks have lower degrees of verification, but in projects that are more complex VDT may assess more nuanced results.

The remainder of this PRA-VDT analysis approximates the full *joint* distributions of the degrees of verification by assessing each sample path individually, rather than fitting *marginal* distributions to the sample results (which Figure 4.5 portrays).

4.2 Defect Model

4.2.1 Defect Type Definitions

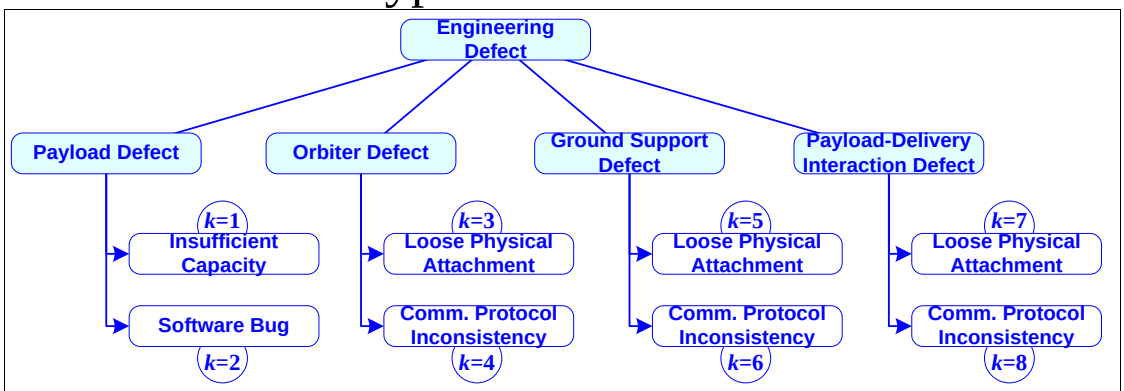


Figure 4.6 Illustrative Typology of 'Pre-Phase-A'-Stage Satellite Defects

Figure 4.6 (above) presents the defects analyzed for the illustrative satellite project. The satellite's payload specification requires a communications system that supports a certain data throughput, so a defect is failing to provide adequate throughput, e.g. from software bugs that cause data transmission loss during operations. As another example, the impact of having loose physical attachments poses a risk of vehicle structural failure during launch, and an inconsistent communication protocol could cause failure during operations.

Table 4.4 (below) lists the types of defects that the satellite illustration assumes and then models and indicates their possible causes in development and effects in operations.

Table 4.4 Input Data for the Satellite Illustration's Defect Model

Engineering Defect Type *k*	**Development Element that Causes Defect, *i***	**$[cp^{+}_{ijk}, cp^{-}_{ijk}]$ = [p(No Defects | Verified), p(Zero Defects | Ignored]**	**di_{kl} = Marginal influence of defect *k* severity on operations component *l***	**$[oc_l^{-}, oc_l^{+}]$ = Best- and worst-case component *l* failure probabilities**
Insufficient Payload Capacity (*k* = 1)	Payload Design Task (*i* = 1)	$\forall j$ $[cp^{+}_{1j1}, cp^{-}_{1j1}]$ = [90%, 50%]	Payload I-IV (*l* = 4-7) $di_{14} = di_{15} = di_{16} = di_{17}$ = 90%	$[oc_4^{-}, oc_4^{+}]$ = $[oc_5^{-}, oc_5^{+}]$ = $[oc_6^{-}, oc_6^{+}]$ = $[oc_7^{-}, oc_7^{+}]$ = [50%, 85%]
Bug in Payload Software (*k* = 2)	Payload Design Task (*i* = 1)	$\forall j$ $[cp^{+}_{1j2}, cp^{-}_{1j2}]$ = [90%, 50%]	Payload I-IV (*l* = 4-7) $di_{24} = di_{25} = di_{26} = di_{27}$ = 90%	$[oc_4^{-}, oc_4^{+}]$ = $[oc_5^{-}, oc_5^{+}]$ = $[oc_6^{-}, oc_6^{+}]$ = $[oc_7^{-}, oc_7^{+}]$ = [50%, 85%]
Orbiter-Launch Vehicle Attachments Loose (*k* =3)	Delivery Design Task (*i* = 2)	$\forall j$ $[cp^{+}_{2j3}, cp^{-}_{2j3}]$ = [90%, 50%]	Orbiter (*l* = 1) di_{31} = 90%	$[oc_1^{-}, oc_1^{+}]$ = [85%, 99%]
Orbiter-Ground Systems Communications Protocol Inconsistency (*k* = 4)	Delivery Design Task (*i* = 2)	$\forall j$ $[cp^{+}_{2j4}, cp^{-}_{2j4}]$ = [90%, 50%]	Orbiter (*l* = 1) di_{41} = 90%	$[oc_1^{-}, oc_1^{+}]$ = [85%, 99%]
Ground-Launch Vehicle Attachments Loose (*k* = 5)	Delivery Design Task (*i* = 2)	$\forall j$ $[cp^{+}_{2j5}, cp^{-}_{2j5}]$ = [90%, 50%]	Support (*l* = 2) di_{52} = 90%	$[oc_2^{-}, oc_2^{+}]$ = [85%, 99%]
Launch Vehicle-Ground Systems Communications Protocol Inconsistency (*k* = 6)	Delivery Design Task (*i* = 2)	$\forall j$ $[cp^{+}_{2j6}, cp^{-}_{2j6}]$ = [90%, 50%]	Support (*l* = 2) di_{62} = 90%	$[oc_2^{-}, oc_2^{+}]$ = [85%, 99%]
Payload-Orbiter Attachments Loose (*k* = 7)	Payload-Delivery Rework Dependency (*i* = 3)	$\forall j$ $[cp^{+}_{3j7}, cp^{-}_{3j7}]$ = [90%, 50%]	Interfaces (*l* = 3) di_{73} = 90%	$[oc_3^{-}, oc_3^{+}]$ = [75%, 95%]
Payload-Ground	Payload-Delivery	$\forall j$	Interfaces	$[oc_3^{-}, oc_3^{+}]$ =

Table 4.4 Input Data for the Satellite Illustration's Defect Model				
Engineering Defect Type *k*	**Development Element that Causes Defect, *i***	**$[cp^{+}_{ijk}, cp^{-}_{ijk}]$ = [p(No Defects | Verified), p(Zero Defects | Ignored]**	**di_{kl} = Marginal influence of defect *k* severity on operations component *l***	**$[oc_l^{-}, oc_l^{+}]$ = Best- and worst-case component *l* failure probabilities**
Systems Communications Protocol Inconsistency (*k* = 8)	Rework Dependency (*i* = 3)	$[cp^{+}_{3j8}, cp^{-}_{3j8}]$ = [90%, 50%]	(*l* = 3) di_{83} = 90%	[75%, 95%]

4.2.2 Conformance Probabilities

Table 4.4 Input Data for the Satellite Illustration's Defect Model (on page 120) shows the conformance probability limits (cp^{-}_{ijk} and cp^{+}_{ijk}) for the satellite project. In the satellite illustration, the conformance probability limits do not vary by subtask *j*. The High Redundancy, Medium Redundancy, and Low Redundancy projects have the same ranges of conformance probabilities, but the emergent conformance probabilities cp_{ijk} within those ranges vary. Specifically, because the more redundant alternatives are more complex to develop, their emergent conformance probabilities cp_{ijk} are lower, on average, than those of the less redundant cases.

For example, suppose that a simulation trial probabilistically assumes that the work item corresponding to *i* = 1 (the Design Payload task) and subtask *j* = 8 (out of 100 subtasks) has a "verified" result (no exception). By definition, the corresponding degree of verification is dv_{18} = 100%. The "verified" result means that the corresponding conformance probability dv_{18} is optimal (= 1.0). From Table 4.4 Input Data for the Satellite Illustration's Defect Model (on page 120), $cp^{-}_{181} = cp^{-}_{182}$ = 50% and $cp^{+}_{181} = cp^{+}_{182}$ = 90%, which means the Design Payload task affects the conformance probability limits for *k* =1 and *k* = 2. Therefore (by Eq. 3.2) $cp_{181} = cp_{182}$ = 90%, which means that there is a ten percent chance that Design Payload's 8th work item increased the severity of type 1 defects (and similarly for type 2). Regarding other defect types, note that $cp^{-}_{183} = cp^{-}_{184} = cp^{-}_{185} = cp^{-}_{186} = cp^{-}_{187} = cp^{-}_{188} = cp^{+}_{183} = cp^{+}_{184} = cp^{+}_{185} = cp^{+}_{186} = cp^{+}_{187} = cp^{+}_{188}$ = 100%. Therefore, regardless of degree of

verification (dv_{18}), $cp_{183} = cp_{184} = cp_{185} = cp_{186} = cp_{187} = cp_{188} = 100\%$, which means that Design Payload's 8th work item never contributes to defectiveness of other types.

If, by contrast, the simulation probabilistically assumes that the subtask has an ignored exception, then $dv_{188} = 0.0\%$, and therefore $cp_{181} = cp_{182} = 50\%$, which means that there is an even chance that the defectiveness increased. If, instead, the trial shows an exception, followed by "quick-fix" on just half the subtask, the model assigns $dv_{18} = 50\%$ and $cp_{181} = cp_{182} = 70\%$, indicating a thirty percent chance of increasing type 1 (and type 2) defects.

4.2.3 Distribution of Defect Severities

Figure 3.8 (on page 87) shows how the satellite project's conformance probability figures modulate the distribution of resulting engineering defects. Based on Eq. 3.5 (on page 87) the illustrative project the greatest *expected* severity of defects that any subtask could create is $E(S_{ijk}) = -\ln(50\%) = 0.7$. The theoretically worst-case process quality, where all subtasks resulted in ignored exceptions, would result in the task creating an expected 70 defects of that type (Both in practice and in the VDT simulation, however, a project would never finish with such abominable performance).

Total Severity of Defects of Each Type

Figure 4.7 (on page 124) shows how the Defect Model sums the distributions of defect severities contributed by subtasks to determine the defect severity contributed by whole tasks and dependencies.

Consider Figure 4.8 Defect Model's Assessments of Defect Severities Sk for Three Satellite Cases (on page 125). For each VDT simulation trial, the Defect Model estimates the severity of defects that development creates using a Poisson distribution. The High Redundancy case trials (red) are most likely to create a greater severity of defects than the Medium Redundancy or Low Redundancy trials (green and blue,

respectively) because of the greater complexity of design tasks. The Operations Model will assess whether the greater severity of defects offsets the benefits of redundancy.

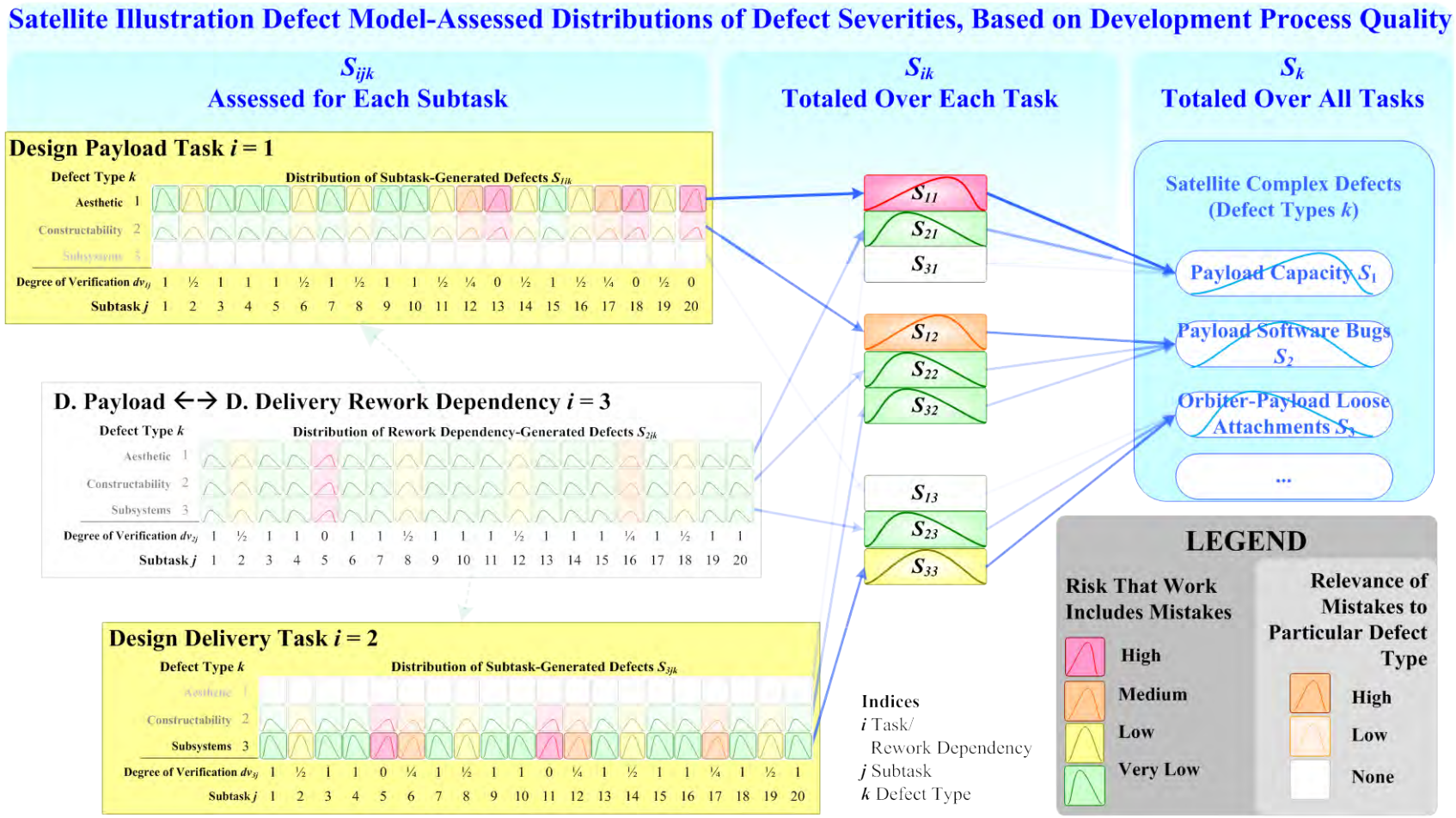


Figure 4.7 Thesis-Assessed Distributions of Defects Caused by Development Process Quality

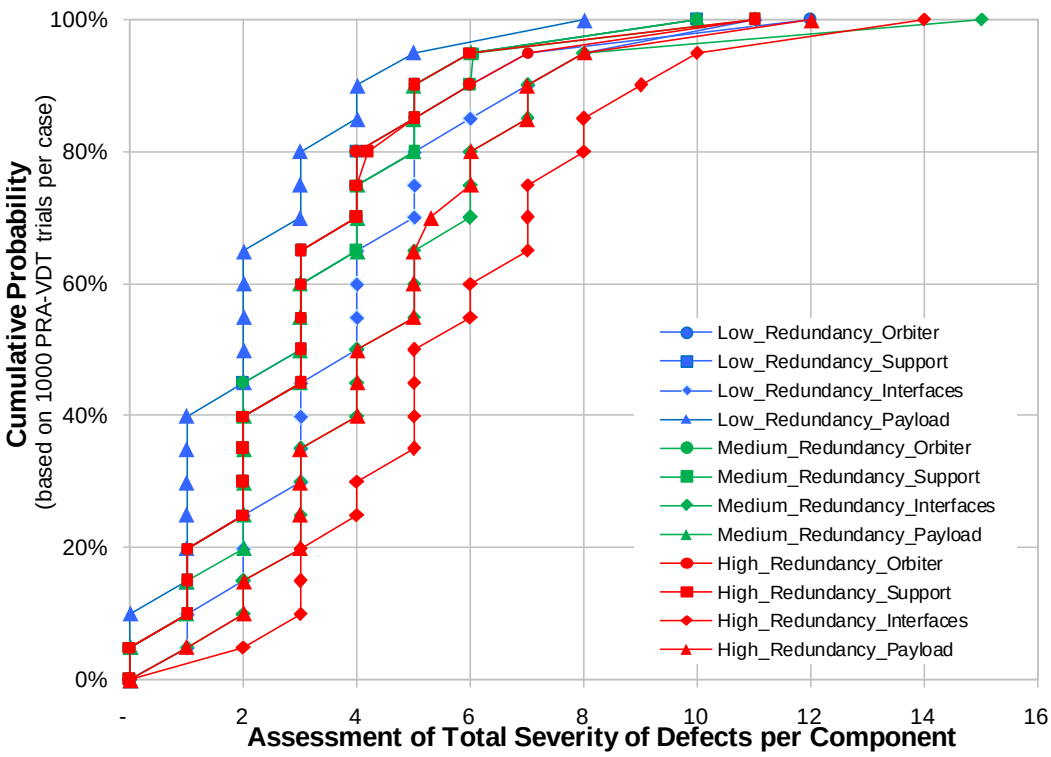


Figure 4.8 Defect Model's Assessments of Defect Severities S_k for Three Satellite Cases Severities Subsystems in the Satellite Project.
(cumulative probabilities based on 1000 PRA-VDT simulation trials)

4.3 Operations Model

4.3.1 Operations Alternatives

Table 4.4 Input Data for the Satellite Illustration's Defect Model (on page 120) provides the values of oc_l^+ and oc_l^- for the satellite illustration. The illustration shows the breadth of data that the model can link to engineering defects. The Orbiter, Support, Interfaces, and Payload capacities represent failure probability for those individual subsystems in the satellite project.

Table 4.4 also provides the values of di_{kl} from the satellite example. In the satellite illustration, each engineering defect type affects exactly one operating capacity,

however PRA-VDT's indexing structure mathematically supports models where this isn't the case.

4.3.2 Operations Behavior

Satellite Component Capacities

The illustrative example presents hypothetical data from field inquiry in a practical application. The hypothetical data include, for each satellite subsystem indexed in l, defect influence values di_{kl} as well as operations capacity limits oc_l^- and oc_l^+ (in Table 4.4 Input Data for the Satellite Illustration's Defect Model on page 120). Substituting those data into Eq. 3.18 (on page 96) yields the operations capacities (oc_l) as a function of the number of (type k) engineering defects (s_k).

Substituting the defect influences di_{kl} in Table 4.4 Input Data for the Satellite Illustration's Defect Model (on page 120), and the sampled severity of product defects s_k (the percentiles of which appear in Figure 4.8 on page 125) provides operations capacities for each subsystem. Figure 4.9 (on page 127) graphs the percentiles for satellite components' operations capacities, and uses color to distinguish the three alternatives: Low (blue), Medium (green) and High (red) Redundancy. The graph makes clear that the alternatives involving higher design complexities (greater redundancy) create lower operations capacities in corresponding components. The payload component's operating capacity falls as product redundancy increases because of increases in the corresponding (Design Payload) task's degree of verification. Redundancy has the same effect on the Interfaces component, though it is less pronounced. The Design Delivery task changes little as the product redundancy increases, so the operations capacities of Support components vary little between cases (in accordance with indirect effects, such as manager backlog).

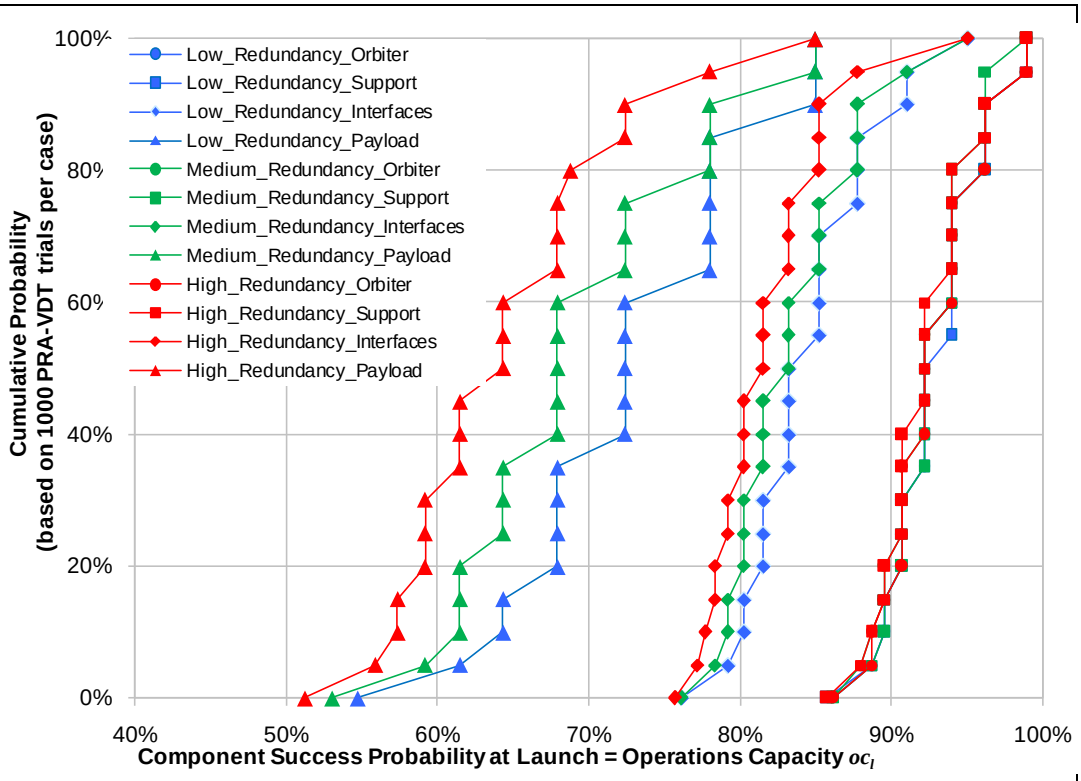


Figure 4.9 Defect Model's Assessments of Component Operations Capacities (Component Success Probabilities at Launch)
(cumulative probabilities based on 1000 PRA-VDT simulation trials)

For example, in the satellite example, di_{22} = 90% is the amount that each unit of type 2 defect severity (support) reduces the capacity of function 2 (support subsystems' failure probability). Therefore, if the severity of type 2 defects is 3, then the support subsystems' failure probability will fall to the point $90\%^3$ = 73% of the way between best and worst case limits. Table 4.4 Input Data for the Satellite Illustration's Defect Model (on page 120) indicates best case operations capacity is 99% and worst case is 75%. The support subsystems' failure probability in this example therefore is 75% + (99% – 75%) * 73% = 92.5%.

4.3.3 Operations Performance

The satellite example models operations context implicitly by defining behaviors as random variables, where the randomness about the operations environment. For these cases, capacities named for components (such as Payload and Orbiter) indicate the probability of successful launch. In the model, OB_n represents the time from launch (defined as time = zero) to failure for each of the subsystems (with subsystems indexed using n). The expected value of operations capacity (for components with the ability to fail) thus provides the marginal probability of component success at launch. Adapting Eq. 3.19 provides the formula:

$$\begin{aligned} & p\left(OB_n > 0\right) \\ & \quad = E\left(OC_l\right) \\ & \quad = oc_l^{\,-} + \left(oc_l^{\,+} - oc_l^{\,-}\right) \times E\left(\prod_k di_{kl}^{\;Poisson\left\{-\ln \prod_i \prod_j \left[cp^-_{ijk} + dv_{ij} \times \left(cp^+_{ijk} - cp^-_{ijk}\right)\right]\right\}} \right) \end{aligned} \qquad \textbf{Eq. 4.1}$$

Assessed Distribution of Time to Failure for Satellite Components

In the satellite example, each of the Low Redundancy case's subsystems (Orbiter, Support, Interfaces, and Payload I) has a chance of immediate failure at launch. Assuming the satellite successfully launches, the model assumes that each subsystem has a constant hazard rate (chance of failure per moment in time) during operations. Engineering defects within the subsystems decrease capacity, thereby increasing the corresponding probabilities of failure at launch and the hazard rates during operations.

The illustrative model calculates the distribution of time to failure OB_n for the first four components (indexed using n) using the corresponding realization of operations capacity oc_n:

$$\forall n \in [1,4] \quad OB_n \sim \begin{cases} 0 & with\ probability\ \left(1 - oc_n\right) \\ Exp\left[50 \times \left(1 - oc_n\right)\right] & with\ probability\ oc_n \end{cases} \qquad \textbf{Eq. 4.2}$$

Each of the subsystems fails during launch with probability equal to one minus capacity. During operations, each subsystem can operate over an exponentially distributed lifetime with a mean of 50 years times one minus capacity.

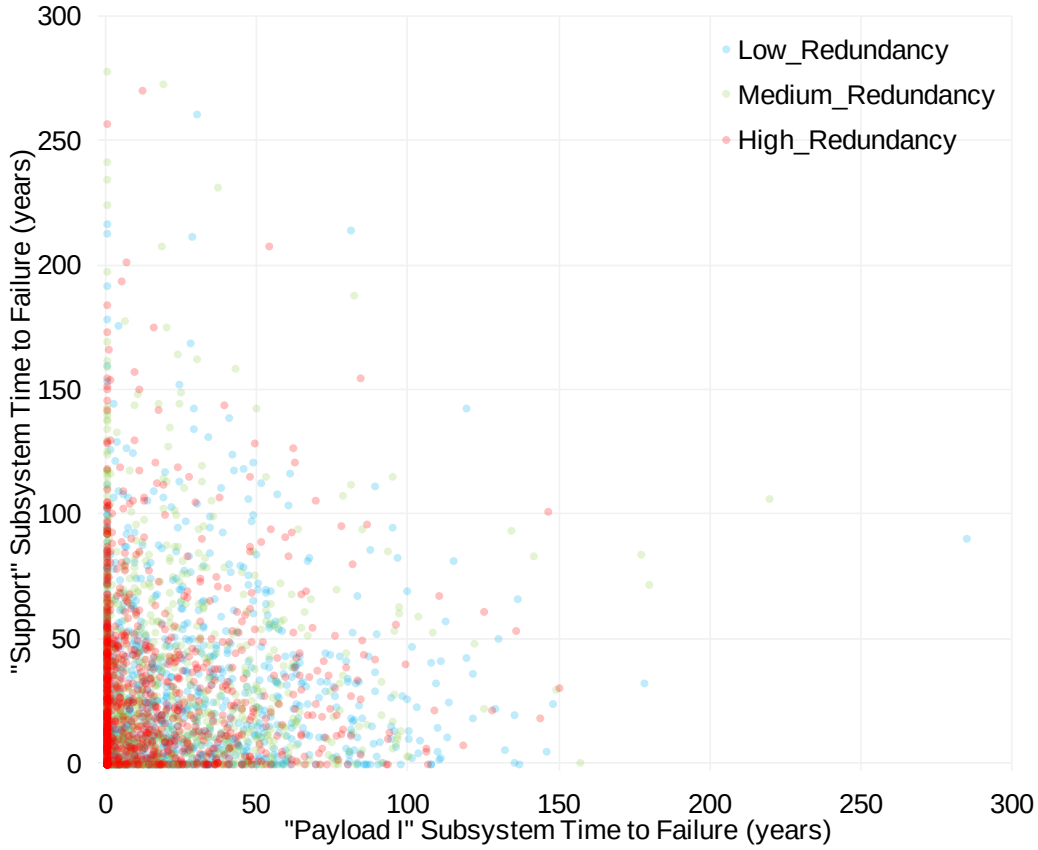


Figure 4.10 Samples of PRA-VDT-assessed Payload I and Support Subsystem Lifetimes

Figure 4.10 (above) plots pairs of Support and Payload I subsystem lifetimes from 1000 PRA-VDT simulation trials of each case. The low redundancy alternatives, having less design complexity and (therefore) fewer engineering defects, create subsystems that typically last longer. The Payload subsystem is more likely to fail during launch than the Support subsystem, and generally fails earlier. This difference is most evident for the alternatives with higher redundancy because their greater Payload engineering complexity leads to more defects. PRA Using Functional Block Diagrams (on page 99) shows that the higher redundancy alternatives survive longer, however, because there are multiple payloads that fail with conditional independence.

Derivation for the Medium Redundancy Alternative

The Medium Redundancy project builds a satellite like the Low Redundancy project, with the addition of Payload II (component n = 5), a redundant subsystem with the same distribution of time to failure as Payload I (subsystem n = 4). Payload II does not share the same realization as Payload I, so it might fail at a different time. The following equation states this relationship formally (note that "~" means "Is distributed as"):

$$OB_5 \sim OB_4 \qquad \textbf{Eq. 4.3}$$

The High Redundancy project builds a satellite like the Low Redundancy project, with the addition of Payload II, Payload III, and Payload IV (components 5, 6, and 7), redundant subsystems with the same distribution of time to failure as Payload I (subsystem 4). The following equation states this relationship formally:

$$OB_7 \sim OB_6 \sim OB_5 \sim OB_4 \qquad \textbf{Eq. 4.4}$$

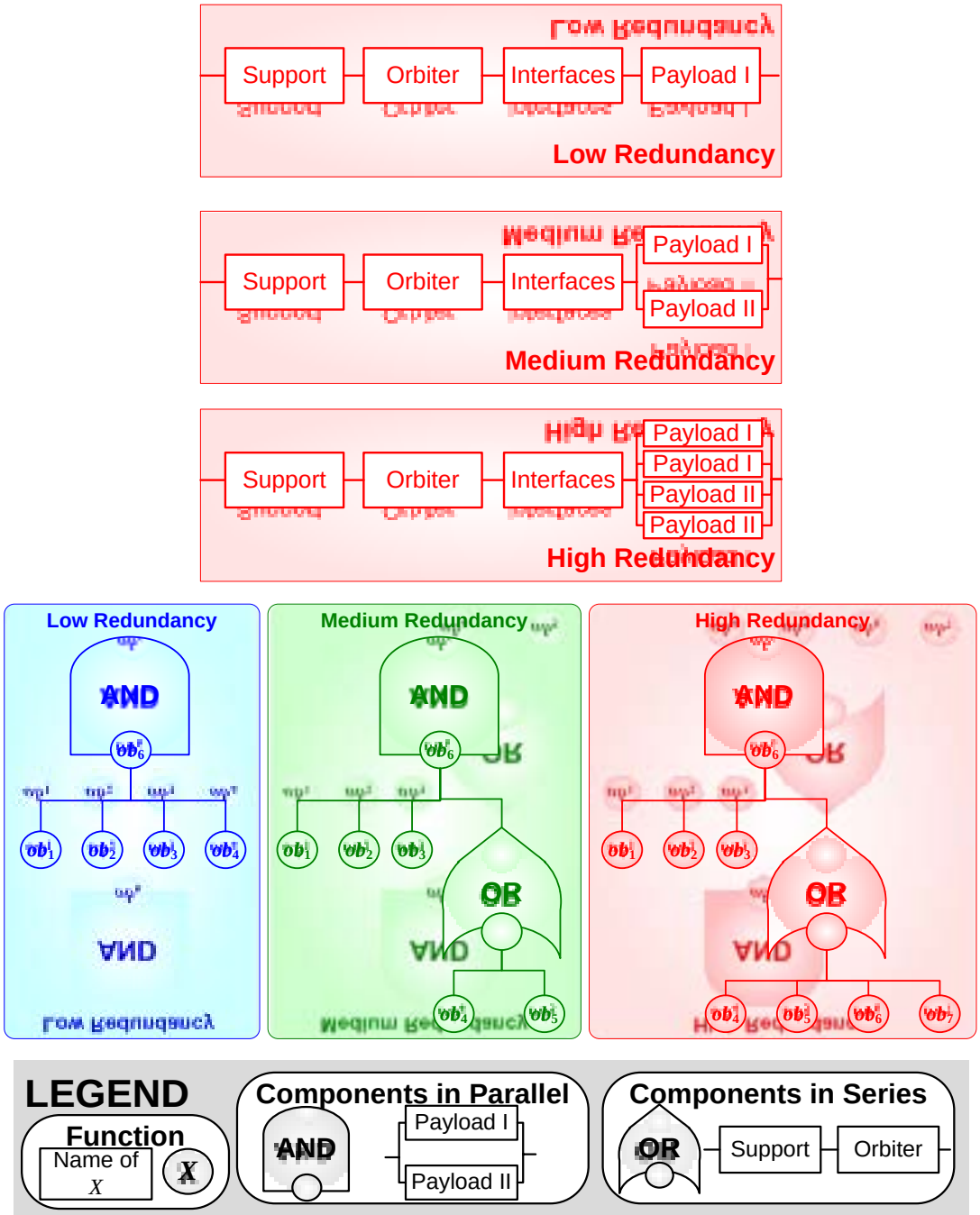


Figure 4.11 Satellite Functional Block Diagrams (top) and Fault Trees (bottom) for PRA Calculation of Satellite Lifetime in terms of Component Lifetimes

The functional block diagrams and fault trees in Figure 4.11 (above) formalize the following intuition based on PRA methods (see §3.3, on page 93). The satellite will

generate revenue continuously from the date of launch to the date of total system failure. The decision of which satellite alternative to build therefore hinges upon the total assessed system lifetime expectation. Total system failure occurs immediately upon failure either in the Orbiter itself, in the Support subsystem (which includes the launch vehicle and ground control), in the Interfaces between subsystems, or in *all* Payload subsystems. The Low Redundancy alternative has one payload subsystem, the Medium Redundancy alternative has two, and the High Redundancy alternative has four.

The distribution of total assessed system lifetimes therefore depends upon every subsystem's operations capacity (therefore on the defect types, and therefore on the engineering project's degrees of verification). As with the individual subsystems, with the total system there is a moderate probability of failure immediately upon launch of the mission, followed by a constant hazard rate.

Derivation for the Medium Redundancy Alternative

The following terse equation formally states the Medium Redundancy case's time to total system failure:

$$ob_8 = \min\left[ob_1, ob_2, ob_3, \max\left(ob_4, ob_5\right)\right] \qquad \textbf{Eq. 4.5}$$

In this illustration, the realization of defects is the only type of external event causing subsystem failures to be probabilistically dependent on one another (for a given sample of VDT-assessed development outcomes). Stated more formally, the failures are conditionally independent of one another *given* the number of engineering defects; the values ob_n are probabilistically independent of one another *given* the operations capacities oc_l. Probabilistic independence between subsystem failures implies that substituting the formulae for ob_n from Eq. 4.2 provides an appropriate formula for total system failure.

In a total success, no subsystems fail. Using PRA, this means the probability of a total launch success is the product of the probabilities of success for all subsystems:

$$p\left(ob_n > 0 \ \forall \ n \in [1,5]\right) = oc_1 \times oc_2 \times oc_3 \times oc_4^2 \qquad \textbf{Eq. 4.6}$$

A partial failure occurs when one payload subsystem fails, but all other subsystems (including the alternate payload) succeed. In this case, the satellite continues operating but suffers an increased hazard rate (chance of failure per unit time). The formula below states the probability of a partial launch failure with Payload II failing:

$$\begin{aligned} & p\left(ob_n > 0 \ \forall \ m \in [1,4] \ ; \quad ob_5 = 0\right) \\ & \quad = p\left(ob_n > 0 \ \forall \ n \in \{1,2,3,5\} \ ; \quad ob_4 = 0\right) \\ & \quad = oc_1 \times oc_2 \times oc_3 \times oc_4 \times \left(1 - oc_4\right) \end{aligned} \qquad \textbf{Eq. 4.7}$$

The probability of partial launch failure with Payload I failing is the same as that with Payload II failing, and the two partial failure prospects are mutually exclusive. Therefore the total probability of a partial launch failure is $2 \times oc_1 \times oc_2 \times oc_3 \times oc_4 \times \left(1 - oc_4\right)$.

The probability of failure at launch is one minus the probability of total success (from Eq. 4.6) minus the probability of partial success ($2 \times oc_1 \times oc_2 \times oc_3 \times oc_4 \times \left(1 - oc_4\right)$):

$$p\left(ob_8 = 0\right) = 1 - oc_1 \times oc_2 \times oc_3 \times oc_4^2 - 2 \times \left[oc_1 \times oc_2 \times oc_3 \times oc_4 \times \left(1 - oc_4\right)\right] \qquad \textbf{Eq. 4.8}$$

Simplifying the above equation clarifies the intuition that the combined system fails at launch unless all of the first three subsystems, and one or more payload subsystems, succeed at launch:

$$p\left(ob_8 = 0\right) = 1 - oc_1 \times oc_2 \times oc_3 \times oc_4 \times \left(2 - oc_4\right) \qquad \textbf{Eq. 4.9}$$

Combining the scenario probabilities in Eq. 4.6, Eq. 4.7, and Eq. 4.9 provides the distribution of satellite lifetime OB_8:

$$OB_8 \sim \begin{cases} 0 & \text{with probability } 1-oc_1\times oc_2\times oc_3\times oc_4\times\left(2-oc_4\right) \\ Exp\left(1-oc_1\times oc_2\times oc_3\times oc_4\right) & \text{with probability } 2\times oc_1\times oc_2\times oc_3\times oc_4\times\left(1-oc_4\right) \\ Exp\left[1-oc_1\times oc_2\times oc_3\times oc_4\times\left(2-oc_4\right)\right] & \text{with probability } oc_1\times oc_2\times oc_3\times oc_4{}^2 \end{cases}$$ **Eq. 4.10**

4.4 Decision Model

This section illustrates the Decision Model by assessing which of the satellite project alternatives (Low Redundancy, Medium Redundancy, or High Redundancy) has lowest expected failure probability (§6.3.5, on page 185, introduces decision-making based on broader utility functions). As VDT and its underlying theory highlights, simpler engineering tasks (such as those in the Low Redundancy project) tend to result in fewer mistakes and (by extension using the Defect Model) in fewer component failures. As PRA and its underlying theory highlights, higher levels of component redundancy (such as those in the High Redundancy project) increases the number of component failures required to cause total system failure. In the satellite illustration (as in many practical applications), designing component redundancy into the product increases engineering complexity, therefore it could either reduce or increase total system failure risk. The satellite analysis therefore illustrates how PRA-VDT assesses two conflicting tendencies' relative significance for a given project.

4.4.1 Project Alternatives

The illustration formally defines the satellite project's alternatives as follows:

Development Alternative:

DA = {Low Redundancy, Medium Redundancy, High Redundancy}

Product alternative:

PA = {Baseline}

Operations Alternative:

OA = {Low Redundancy, Medium Redundancy, High Redundancy}

Project Alternative:

A = { (Low Redundancy, Baseline, Low Redundancy),

(Medium Redundancy, Baseline, Medium Redundancy),

(High Redundancy, Baseline, High Redundancy)}.

Note that (Low Redundancy, Baseline, High Redundancy) is not in *A* because it is not consistent; a Low Redundancy design project does not produce a High Redundancy satellite.

4.4.2 Project Performance

Figure 4.12 Distribution of Satellite Lifetime (on page 136) shows the percentiles for total satellite lifetime that PRA-VDT assessed using the Low Redundancy (blue), Medium Redundancy (green) and High Redundancy (red) alternatives. The data come from analyses of 1000 simulation trials, as the illustrative portions throughout this chapter describe.

Table 4.5 Satellite Analysis Summary Statistics: Expected Values

(from 1000 PRA-VDT simulation trials)	Redundancy		
	Low	Medium	High
Expected Time to Failure	4.8	6.0	**6.9**

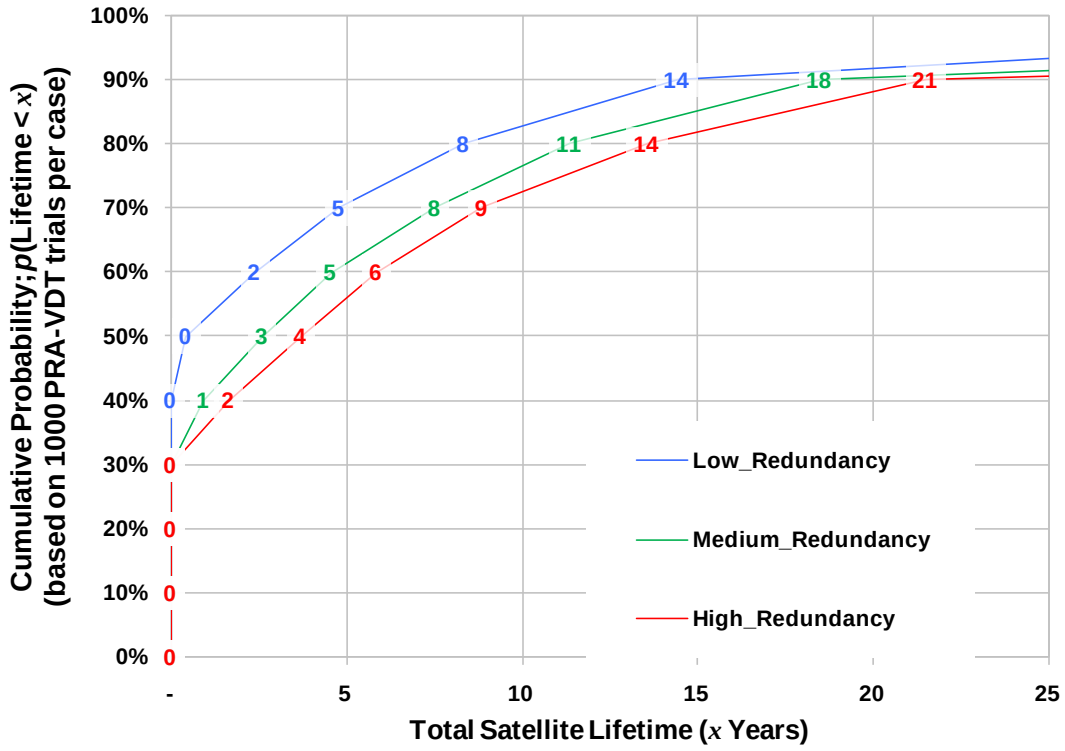


Figure 4.12 Distribution of Satellite Lifetime
(cumulative probabilities based on 1000 PRA-VDT simulation trials)

In the graph, observe that the blue line is always at least as high as the green line, which is always at least as high as the red line. Interpreting the graph in the context of project inputs, the Low Redundancy case (blue) was least likely to achieve each satellite lifetime target because the lack of redundancy in the payload led to frequent failures. The Medium Redundancy case gains enough of the benefits of redundancy to offset an increased engineering cost, duration, and severity of defects. The High Redundancy case does also gain enough benefit from the further increase in redundancy to offset its much greater severity of defects.

In conclusion, I interpret these simulation results and conclude that the High Redundancy case (red) achieved each price target in the greatest percentage of trials.

This result strongly suggests (but does not prove) that the High Redundancy alternative is stochastically dominant, and is therefore best for rational decision makers regardless of risk attitude.

As with other management science methods, responsibly solving practical problems using PRA-VDT requires modeling judgments that are specific to project conditions. For example, the results in this chapter do not consider factors other than satellite lifetime, such as the engineering cost and duration (which VDT assesses). Providing different project input can also dramatically affect the results. For example, increasing the range of operations capacities could make the satellite lifetime depend more on defects, and make the simpler (Low Redundancy) case seem like the best choice.

Chapter 5
Green Dorm Field Study

This chapter uses PRA-VDT to assess risks due to potential engineering defects in the electrical system of a proposed, sustainability and research-oriented dorm at Stanford University.

This chapter documents a field study that applies the PRA-VDT framework to a real design project at Stanford University. The field study builds the model justification by instantiating the thesis model (provided in the previous chapter) at the schematic design level of detail. The illustrative case suggests that that a knowledgeable user can create meaningful recommendations based on the descriptions and assessments of the thesis Defect Model. This initial demonstration of the power of the thesis Defect Model to generate plausible descriptions of real engineering situations in turn provides evidence that the underlying theory and methods of thesis Defect Model represents a contribution to knowledge in the field of project modeling and optimization.

Figure 5.1 Stanford Green Dorm's Sustainable Strategies (on page 139) illustrates the Green Dormitory, a 47-room student housing proposal developed for Stanford University. The schematic diagram, from the Green Dorm Feasibility Study Final Report [EHDD 2006], shows how the many novel technologies in an advanced Living Laboratory dorm relate to one another, and to occupants, during operations. According to the 2005-2006 School of Engineering Annual Report, the Green Dorm will "embody three important goals: to demonstrate how buildings can be designed to be positive contributors to the environment; to provide a living laboratory that allows

faculty and students to explore innovations in green technology; and to be a great place for students to live, study, and socialize." Compared with a traditional dorm, the building includes incremental changes, like high-efficiency water fixtures, and completely new features, like a building systems laboratory and information center.

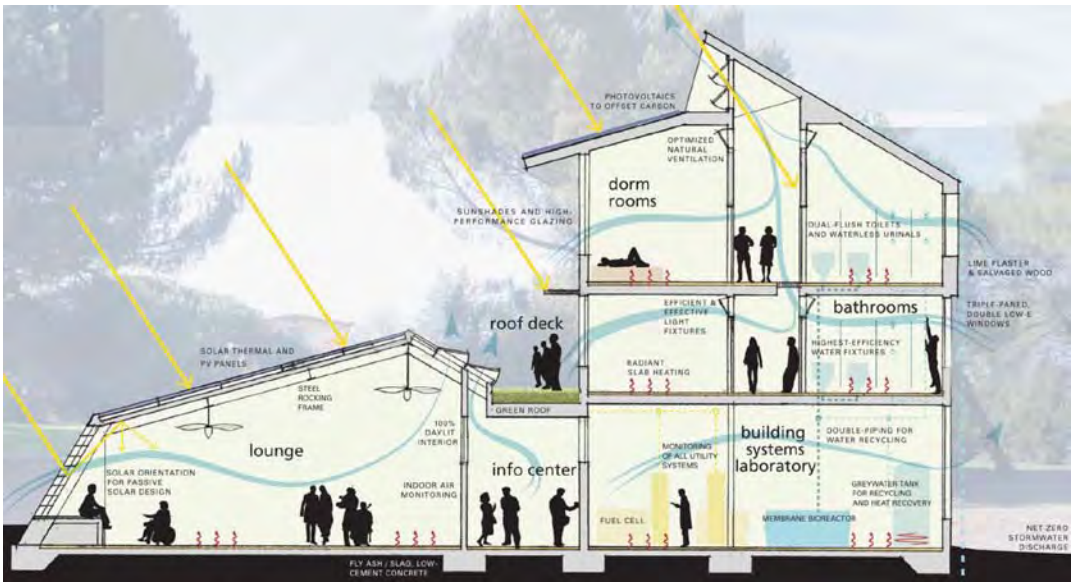


Figure 5.1 Stanford Green Dorm's Sustainable Strategies
Courtesy of the Green Dormitory Feasibility Study Final Report [EHDD 2006]

In this chapter, the Model-based PRA-VDT analyses demonstrate the thesis methods by comparing the risks and rewards of including advanced sustainability and research components in the proposed dorm. During the Feasibility Study, the University faced a decision of whether to build simpler or more feature-rich alternatives, e.g., power from the grid versus self-generated photovoltaic power, which may be more environmentally benign but more expensive. Better understanding the relationships between development and operations risks can help owners choose the best mix of product components, and can help project managers focus resources on efficiently mitigating the building performance risks.

The field study focuses on the risks that electrical system components fail during occupancy. Alone, VDT can assess the effectiveness with which designers engineer the complex, novel dorm. Alone, PRA can assess the rate of failures in electrical subsystems, based on the rates of component failures. The integrated PRA-VDT is

required, however, to assess how shortcomings in the design team interact with vulnerabilities s in the electrical system design. The illustrative PRA-VDT analysis shows how defects in the dorm could increase the failure rates of electrical subsystems, and uses DA to identify the best building alternative.

Figure 5.2 Macro-Level Influence Diagram of the Green Dorm Project Model (below) provides an influence diagram containing the major model components and the influences between them. The PRA-VDT framework uses the existing VDT method to assess the distribution of development-stage outcomes based on the organization and process of design. The Defect Model assesses the distributions of various kinds of defects in the building features that would result from those design stage behaviors. The Defect Model then uses those defect distributions to inform a PRA analysis of events occurring during operations. Finally, the PRA analysis provides figures that enable a decision maker to compare the project outcome, using DA, against model-assessed outcomes from analyzing alternative projects.

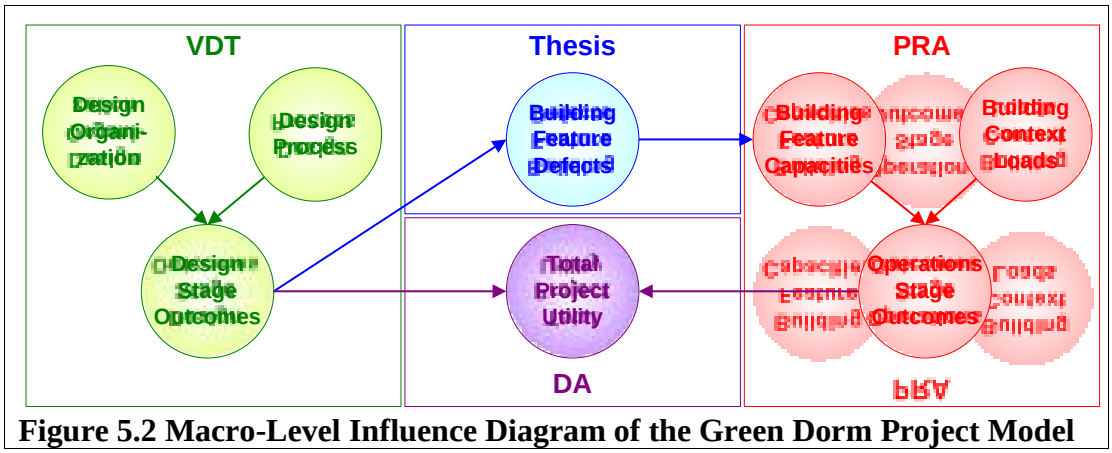


Figure 5.2 Macro-Level Influence Diagram of the Green Dorm Project Model

This chapter formally presents a model and analysis of the Green Dorm Project using the PRA-VDT Framework (PRA-VDT). The chapter's organization matches that of Chapter 3, Model Definition (on page 56), and of Chapter 4, Satellite Illustration (on page 110). For example, §5.1.1 describes a model of the Green Dorm Development organization that uses the formal approach that §3.1.1a introduced. This chapter has the following outline:

§5.1 The Green Dorm Development Model estimates the distribution of development impacts (degrees of verification) by comparing development capacities (organization and culture) and corresponding load (tasks and dependencies).

§5.2 The Green Dorm Defect Model assesses the distribution of electrical system-related engineering defect severities by identifying the conformance probabilities corresponding to development's degrees of verification.

§5.3 The Green Dorm Operations Model estimates the distribution of electrical component and subsystem failure rates based on distributions of engineering defect-influenced operations capacities.

§5.4 The Green Dorm Decision Model identifies the project alternative with greatest expected utility using the PRA models' assessed distributions of subsystem failures.

§5.5 Photovoltaic Team Decision explains the full arc of PRA-VDT analysis to assess whether hiring an expert photovoltaic team is worth the price premium.

The field study addresses the Green Dorm Project because it fits within the targeted project definition of Chapter 1; the project is complex, involves a significant risk of components underperforming during operations due to shortcomings in design quality, involves difficult project shaping decisions preceding design, and follows a stage-gate structure. The following quotations from interviews with the design team testify to the challenges of this novel project:

> *[The Baseline Green and Living Lab] are particularly difficult because we have an absolute goal. Usually you can compare the design with that of other buildings. However, we need to know the absolute performance, and there is less data on that. We have to calibrate our estimate of design performance to actuals, not relative performance measures.*
>
> *-Green Dorm Engineer*
>
> *In our industry, designers have immediate feedback and incentives to ensure we deliver plans both on time and within budget. Because these buildings aren't constructed and occupied until much later, however, the link between quality and rewards is broken.*
>
> *-Green Dorm Engineer*

With this distinctive combination of features, the Green Dorm Project highlights the thesis method's ability to assess the near limitlessness of potential failures in complex projects. The PRA-VDT model capturing these sequences includes four models with consistently manageable amounts of detail, but integrates them to capture myriad causal relationships. There are hundreds of potential interventions to the circumstances of Green Dorm development, such as increasing an agent's skill or reducing the complexity of a design task. Each of these interventions can have ripple effects that impact the (VDT) Development Model -assessed degrees of verification for every one of the design tasks. In turn, each of these degrees of verification helps determine the severity of defects, and resulting failure rates, that the thesis assesses for the model's product components. The electrical components' failure rates inform the (PRA) Operations Model -assessed failure rate for each electrical subsystem, including circuits, panels, and the full electrical system, during operations. The total electrical subsystem's probability of total or partial failure informs the decision maker as to which alternative building configuration has greatest expected utility. In total, the probabilistically dependent model variables represent millions of causal chains connecting development decisions to decision-maker utility.

Methods

The model's structure and input data are "based on a true story," meaning that they are grounded in field observations and project documentation (as cited throughout the chapter). However, the model excludes some project-specific details that would needlessly limit the intuitiveness of analysis results. For example, even if the selected structural designers were known to be more skilful than the electrical designers were, this chapter's model would illustrate them as having the same skill. This compromise helps comparing model results against intuition, the purpose of this chapter, but hurts comparing model assessments against real-world project results, which would be the purpose of a follow-on study.

The Green Dorm Project Model synthesizes data from an existing formal model as well as existing documentation, observation, interviews, and surveys. The inspiration to include a "Design Verification" task, but no specific model data, comes from a VDT model that Caroline Clevenger, Anna Fontana, and Vincent Rabaron created as a class project based on interviews with Green Dorm Project Architect Brad Jacobson, Professor Raymond Levitt, Professor John Haymaker, and John Chachere [Clevenger et al 2006].

The initial vision for a Green Dorm model came from observing students and faculty funded by the EPA to learn by participating in the development effort. Early documents outlining the project's shape included the winning EHDD proposal, Stanford guidelines for project management, global standards for environmentally sustainable building, and web-based wiki correspondence among project participants. Regular all-hands meetings between the designers and stakeholder representatives conveyed the team's principal uncertainties and decisions, and these data motivated the overall model structure and design of experiments. In addition to reinforcing these messages, the feasibility study report and its appendices [EHDD 2006] provided data that calibrated the initial model. I conducted over a dozen follow-up interviews focused on PRA-VDT model building, calibration and analysis, that included the project manager, principal power systems engineer, and several students, architects and professional engineers familiar with (but not employed by) the project.

The analysis in this chapter includes the full design process and organization planned for the Green Dorm's schematic design stage. This chapter's analysis assesses engineering defect risks only in the electrical subsystem, and includes a detailed look at the photovoltaic system. Excluded building subsystems that are amenable to PRA-VDT analysis include plumbing, structures, and HVAC (heating, ventilation, and air conditioning).

Electrical Subsystems Analysis

Figure 5.3 (on page 146) shows how the PRA-VDT model links design tasks, electrical components, and defects. The electrical system has defect-influenced risks following initiating events in which individual components (such as outlets, lights, or photovoltaic arrays) fail, and engineering defects elevate those risks. Providing inadequate built-in lighting, for example, leads residents to use more risky light sources, such as hot halogen lamps. Providing insufficient electrical outlets leads to hazardous extension cord use and plug overloading. The PRA model captures these complex failures using fault trees.

A PRA model describes the escalation of failures in electrical components that coincide with failures in corresponding protective devices (such as circuit breakers). The higher levels of failure, at the main panel or subpanel, have the greatest potential for catastrophic damage, such as fires and electrocution, because they allow greater amounts of electricity to flow in an uncontrolled fashion. The higher levels' failsafe devices also shut down larger portions of the electrical service. The analysis in this chapter assesses the incidences of component and subsystem failure rates, but leaves for external models the assessment of the specific amounts of harm, such as injury or property damage, that these failures could cause.

The field study's Development Model uses VDT to compare the information processing capacity of the design team against the corresponding load placed by their interdependent tasks, and assesses the distributions of degrees of task verification. The Defect Model, a new model of the thesis, interprets the degrees of verification to estimate the severities of defects in each of the building's electrical components. The Operations Model uses PRA to estimate distributions of achieved electrical component failure rates. The three Models' results inform the Decision Model, which uses Decision Analysis to identify the better of three alternatives: the traditional Row House, the Baseline Green model of sustainability, or the innovative Living Lab. In the analysis, the advanced alternatives' greater design complexities lead to more

defects, and therefore to greater risk of electrical failures. The analysis also includes two idealized cases, which contrasts model results based on the full range of PRA-VDT features against those assuming perfect design processes or a defect-free design.

The electrical system analysis serves merely to illustrate the PRA-VDT method, and this thesis makes no claim that the analytic results are plausible in industry. In contrast, a field expert did review the photovoltaic system at the end of this chapter, which explains the assessment of whether hiring a particularly expert photovoltaic team is worthwhile. In an interview, Stanford Civil and Environmental Engineering Professor Haymaker (an architect familiar with the Green Dorm project) stated his view that the principal data (input, output, and inferences) are plausible for the project.

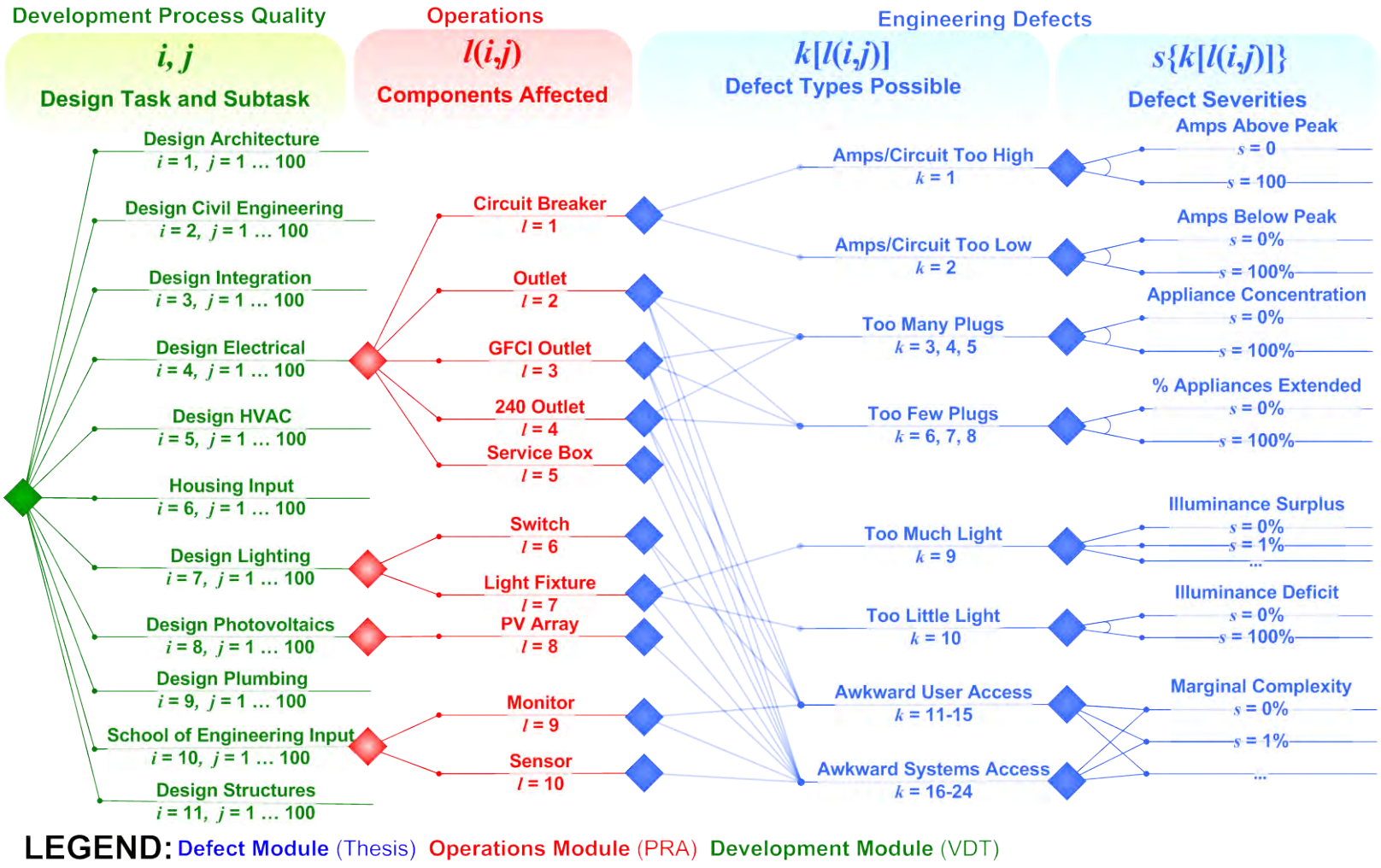


Figure 5.3 Indexing Structure for the PRA-VDT Model of the Green Dorm Electrical Subsystem

Green Dorm Model Cases

The analysis in this chapter compares the failure probabilities for electrical systems using five different project models. The models have essentially the same organizational design, but process complexity and product configuration depends upon the University's choice of project. Starting with the feature-rich "Living Laboratory" (the design Stanford initially proposed), the analysis progresses to two simpler but viable alternatives, and finally to two idealized cases that highlight the roles of design process quality and of engineering defects in PRA-VDT. The chapter's model analyses assess that the less complex alternatives would generate less severe design defects and, therefore, lower failure probabilities for the electrical subsystems.

Baseline Alternative: the Full-Featured "Living Laboratory"

The "Living Laboratory" Stanford proposes is the base case for this chapter's illustration.

Living Laboratory The baseline alternative would exemplify sustainability and serve as a testbed for new building technologies. This alternative would be an exemplar of sustainable building, and would include a broad range of "cutting edge" features to demonstrate new technologies and support ongoing research by the Stanford School of Engineering. For example, the building would have a green roof and would recover heat from shower and laundry wastewater. In the electrical system, the Living Lab would include photovoltaic arrays as well as monitors and sensors that report power use at all levels of the building (in real-time). Intuitively, the complex Living Lab alternative would have high task complexities, resulting in the highest failure probabilities among all the modeled cases.

Intervention Alternatives: the Simpler "Baseline Green" and "Row House"

Early in the feasibility study stage, the design team considered three primary alternatives, each of which represented a configuration of building technologies [EHDD 2006]. The principal differences between these primary alternatives are that the less sophisticated ones have fewer potential benefits, but are also simpler to design.

Baseline Green This alternative would show that an exemplar of sustainable building could make economic sense. A simplified version of the Living Laboratory design, the design retains sustainability technologies that are individually common and well understood, but are rarely combined in such large numbers. For example, the building would collect and reuse rainwater, and natural ventilation would provide summertime cooling. In the electrical system, the Baseline Green Case includes the Living Lab's monitors and sensors, but does not include photovoltaic arrays. Intuitively, the Baseline Green alternative would have medium task complexities, resulting in lower failure probabilities than the Living Lab alternative.

Row House This alternative would house students in the traditional manner of existing buildings in the neighborhood. The semi-autonomous student dormitory would house the undergraduate students reliably and economically. The organization, process, and product technologies of a Row House project are all "tried and true." In the electrical system, the Row House Case includes neither monitors and sensors nor photovoltaic arrays. Intuitively, the Row House alternative would have low task complexities, resulting in lower failure probabilities than the Living Lab or Baseline Green alternatives.

Calibration Cases: the Idealized "Perfect Process" and "No Engineering Defects"

To serve model calibration, the analysis in this chapter also considers two idealized cases. Neither case represents a real alternative that the University can choose, but each case provides a point of reference for project analysts as well as theorists.

Perfect Process This case uses the Living Laboratory model but assumes that all design tasks result in perfect (100%) degrees of verification. With this "perfect" design process quality, in the model engineering defects would still occur (albeit rarely) because model agents conducting the verification process have a "false negative" rate in checking for defects. Intuitively, the Perfect Process case would have lower failure rates than even the Row House case.

No Engineering Defects This case uses the Living Laboratory model but assumes that all tasks produce zero defects. With this "perfect" design, component failures still occur (albeit rarely) due to alternative causes such as installation or maintenance errors. Although the PRA model does not distinguish between causes other than engineering defects, the method does estimate how the resulting individual component failures could combine to produce broader subsystems failures. Intuitively, the No Defects case would have the lowest failure rates out of all the modeled cases.

5.1 Development Model

In the Development Model input, adding building features increases design task complexity and uncertainty. VDT's assessment is that degrees of verification suffer moderately under those conditions.

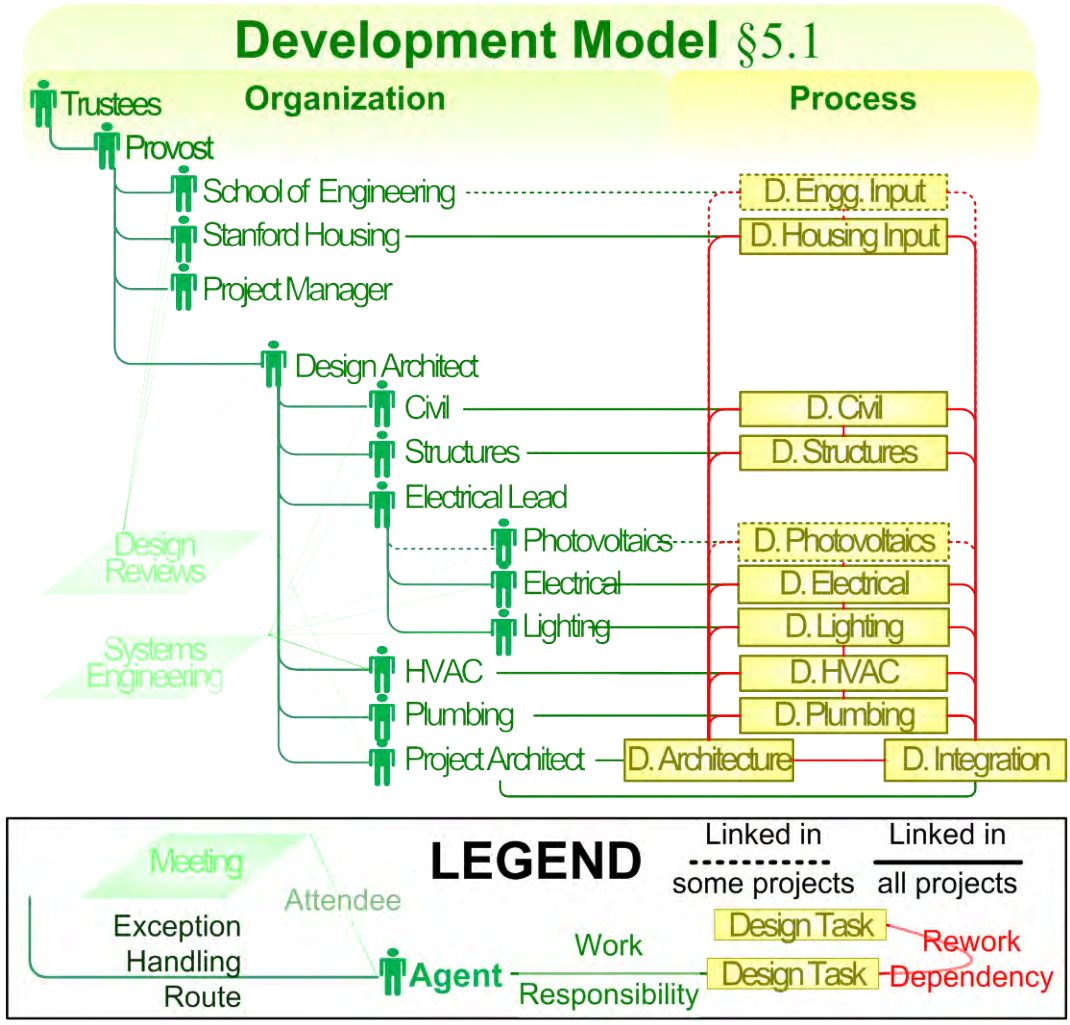


Figure 5.4 Green Dorm Development Organization and Process

5.1.1 Development Alternatives

Consider Figure 5.4 Green Dorm Development Organization and Process (above). Structurally, this VDT Model of a Green Dorm development stage is typical for small projects. The organizational hierarchy (left; details in Tables 4.2 and 4.3) merges

owner and designer teams, and the design tasks (right; details in Tables 4.4 and 4.5) revolve around architecture and integration.

Development Organization

Table 5.1 and Table 5.2 describe input to the VDT model of Green Dorm project development organization and culture (see Table 3.2 on page 68 for attribute definitions). The Green Dorm Project's development stage involves university management, department representatives, and a range of subsystem designers. The Living Lab, Baseline Green, and Row House alternatives are identical, except that the School of Engineering would not participate in a Row House project, and the Photovoltaic design team would participate only in a Living Lab project. The Perfect Process and No Engineering Defects cases do not include Development Models.

Table 5.1 Green Dorm Organization Data Input to VDT

Agent Name	Supervisor	Skill	Experience	Role	Full-Time Equivalents
Trustees		***	***	Project Manager	0.01
Provost	Trustees	***	***	Project Manager	0.05
***School of Engineering**	Provost	Medium	Medium	Subteam	1.0
Stanford Housing	Provost	Medium	Medium	Subteam	1.0
Project Manager	Provost	***	***	Project Manager	0.05
Design Architect	Project Manager	***	***	Subteam Lead	0.1
Civil	Design Architect	Medium	Medium	Subteam	1.0
Structures	Design Architect	Medium	Medium	Subteam	1.0
Electrical Lead	Design Architect	***	***	Subteam Lead	0.1
****Photovoltaics**	Electrical Lead	Medium	Medium	Subteam	1.0
Electrical	Electrical Lead	Medium	Medium	Subteam	1.0
Lighting	Electrical Lead	Medium	Medium	Subteam	1.0

Table 5.1 Green Dorm Organization Data Input to VDT					
Agent Name	**Supervisor**	**Skill**	**Experience**	**Role**	**Full-Time Equivalents**
HVAC	Design Architect	Medium	Medium	Subteam	1.0
Plumbing	Design Architect	Medium	Medium	Subteam	1.0
Project Architect	Design Architect	Medium	Medium	Subteam	1.0
* The Row House does not include a SU Engineering team.					
** Only the Living Lab alternative includes a Photovoltaic team.					
*** Managers without direct work tasks need no skill or experience values.					

Table 5.2 Green Dorm Culture Data Input to VDT								
Project Exception Prob.	Functional Exception Prob.	Commun-ication Prob.	Institutional Exception Prob.	Noise Prob.	Team Experience	Centrali-zation	Formali-zation	Matrix Strength
0.2	0.2	0.2	0.0	0.0	High	Low	Medium	Medium

Development Process

Table 5.3 and Table 5.4 present input to VDT for the Green Dorm project development tasks and project (see Table 3.3 on page 69, and Jin and Levitt 1996, for attribute definitions). Structurally, each design team has a dedicated task, and those developing interdependent systems share rework and communication links (at right in Figure 5.4 Green Dorm Development Organization and Process, on page 150). The project architect's "Architecture" and "Design Integration" tasks share rework and communication links with all other design tasks.

The principal difference between the three primary alternatives is that the more advanced alternatives have greater requirement and solution complexities. In addition, Photovoltaics and School of Engineering efforts are not required for the Row House alternative, and the Photovoltaics are not required for the Baseline Green alternative.

Table 5.3 Green Dorm Task Data Input to VDT					
i	**Task**	**Rework and Communication Dependencies**	**Subtasks |{*j*}|**	**Primary Development Responsibility**	**Uncertainty, Requirement Complexity, and Solution Complexity**
1	Design Architecture	All	100	Project Architect	Low (Row House) Medium (Baseline Green) High (Living Lab)
2	Design Civil	Structures, Site Management	100	Civil	"
3	Design Integration	All	5	Project Architect	"
4	Design Electrical	Photovoltaics, Lighting	100	Electrical	"
5	Design HVAC	Lighting, Plumbing	100	HVAC	"
6	Housing Input	Engineering Input, Site Management, Architecture, Design Integration	100	Stanford Housing	"
7	Design Lighting	Electrical, HVAC	100	Lighting	"
8	* Design Photovoltaics	Electrical	100	Photovoltaics	No Task (Row House) No Task (Baseline Green) High (Living Lab)
9	Design Plumbing	HVAC	100	Plumbing	Low (Row House) Medium (Baseline Green) High (Living Lab)
10	** School of Engineering Input	Housing Input, Architecture, Design Integration	100	School of Engineering	No Task (Row House) Medium (Baseline Green) High (Living Lab)
11	Design Structures	Civil	100	Structures	Low (Row House) Medium (Baseline Green) High (Living Lab)
** Only the Living Lab design includes a Photovoltaic task.					
* The Row House design does not include an Engineering Input task.					

Table 5.4 Green Dorm Meeting Data Input to VDT			
Meeting Name	**Description**	**Period**	**Invitees**
Design Reviews	High-level discussion of dorm requirements	2hrs./ 2 wks.	Stanford Housing, School of Engineering, Project Manager, Design Architect, Project Architect
Systems Engineering Meeting	Low-level discussion of dorm solutions	1hr./wk.	Civil, Structures, Electrical Lead, Photovoltaics, Electrical, Lighting, HVAC, Plumbing, Project Architect

Development Model of Intervention Alternatives

Figure 5.5 Increased Complexity of Living Lab Water Systems (on page 155) illustrates the differences between simpler and more advanced building alternatives using the Baseline Green and Living Laboratory water systems. In a traditional dormitory configuration (top), the Green Dorm would include five water systems: potable water, blackwater, stormwater, lake water and rainwater. The more feature-rich Living Laboratory configuration (bottom) adds treated greywater and treated stormwater lines, as well as new flows to a green roof, treatment facilities, and a rainwater cistern. Design tasks for all of the building's subsystems experience similar process effects as plumbing (notably electrical, which includes photovoltaics and monitors only for some alternatives).

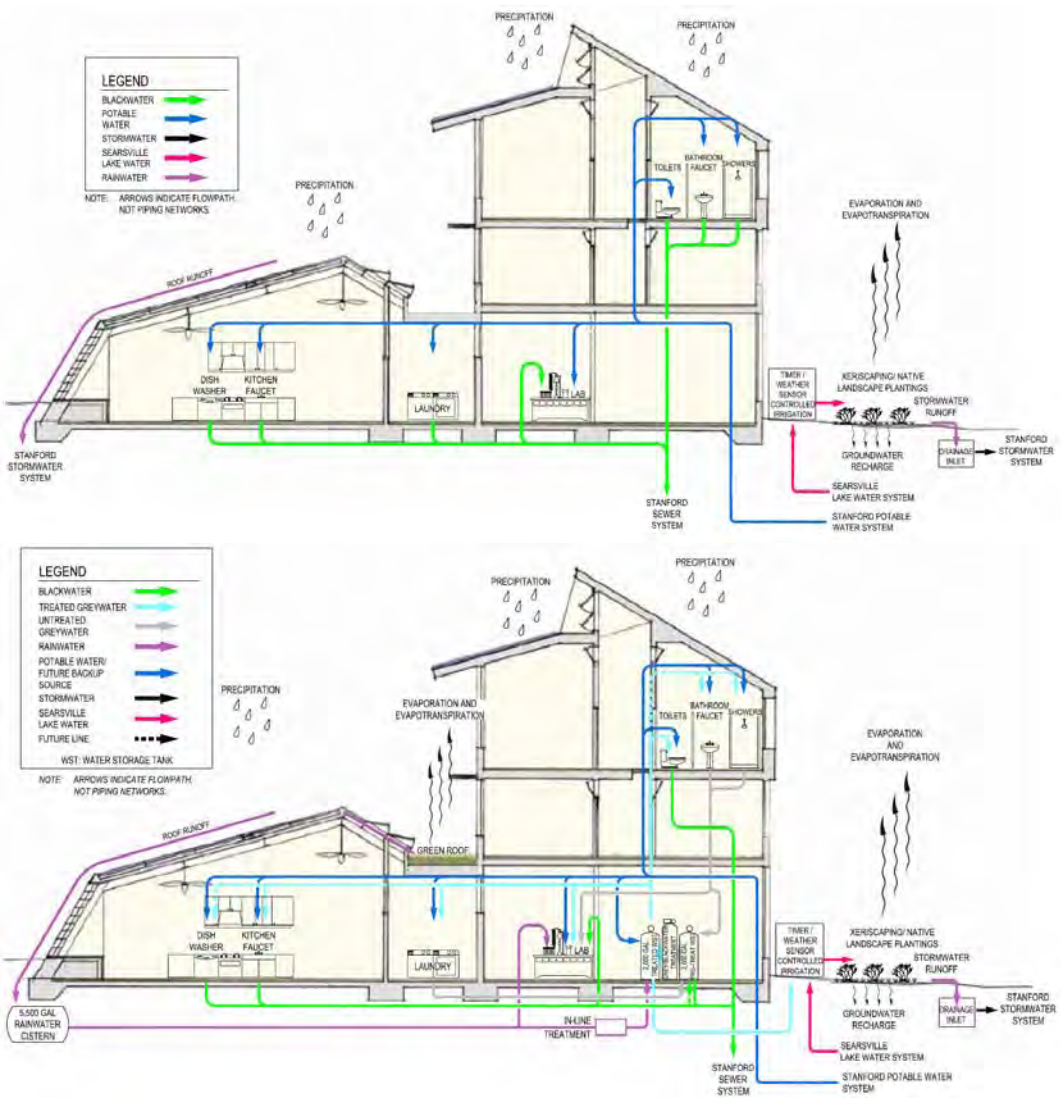


Figure 5.5 Increased Complexity of Living Lab Water Systems
Courtesy of the Stanford Graduate School of Business Environmental Sustainability Task Force Final Report [GSB 2006]

The more advanced alternatives include tasks with greater *requirement complexities* because they include *more interacting subsystems*. For example, the Baseline Green alternative includes blackwater (sewer), potable water, stormwater, lake water, and rainwater subsystems, and therefore has medium requirement complexity, whereas the Living Lab plumbing task's addition of treated and untreated greywater brings requirement complexity to high.

The more advanced alternatives also include tasks with greater *solution complexities* because they serve a *greater number of purposes* such as research and education. For example, the Row House water systems should support housing desirability with clean drinking water and hot showers, whereas the Baseline Green systems should additionally include green features that conserve water and heat, and the Living Lab systems should additionally serve research by monitoring use and accommodating adaptation to novel technologies over time.

These adjustments to the model of design complexity enable the model to assess the differences in potential for defects in the engineered systems and in interdependent tasks like Design Architecture.

5.1.2 Development Performance

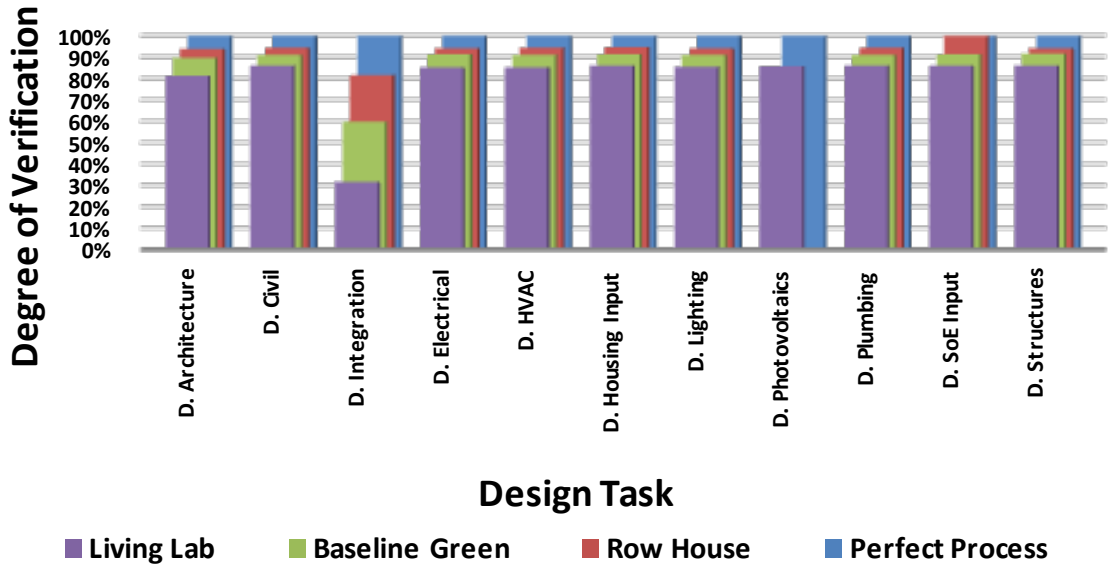


Figure 5.6 Development Tasks' VDT-Assessed Degrees of Verification (expectations based on 100 simulation trials per case)

Figure 5.6 Development Tasks' VDT-Assessed Degrees of Verification (above) presents the results of 100 VDT simulation trials. The thesis interprets a VDT model's Green Dorm exception handling behavior to estimate the degree of verification for each development task's work. The Row House alternative (red), which is simplest to

develop, results in the highest degree of verification, followed by the Baseline Green (green) and Living Lab (purple) alternatives. The Perfect Process case (dark blue), by definition, assumes 100% verification, and degree of verification is not relevant for the No Engineering Defects case (not shown). The Defect Model, detailed in this chapter's next section, quantitatively estimates the distribution of defect severities resulting from those degrees of verification.

5.2 Defect Model

The Defect Model assesses defects to be most severe in cases where design complexity is greatest.

5.2.1 Electrical Defect Types

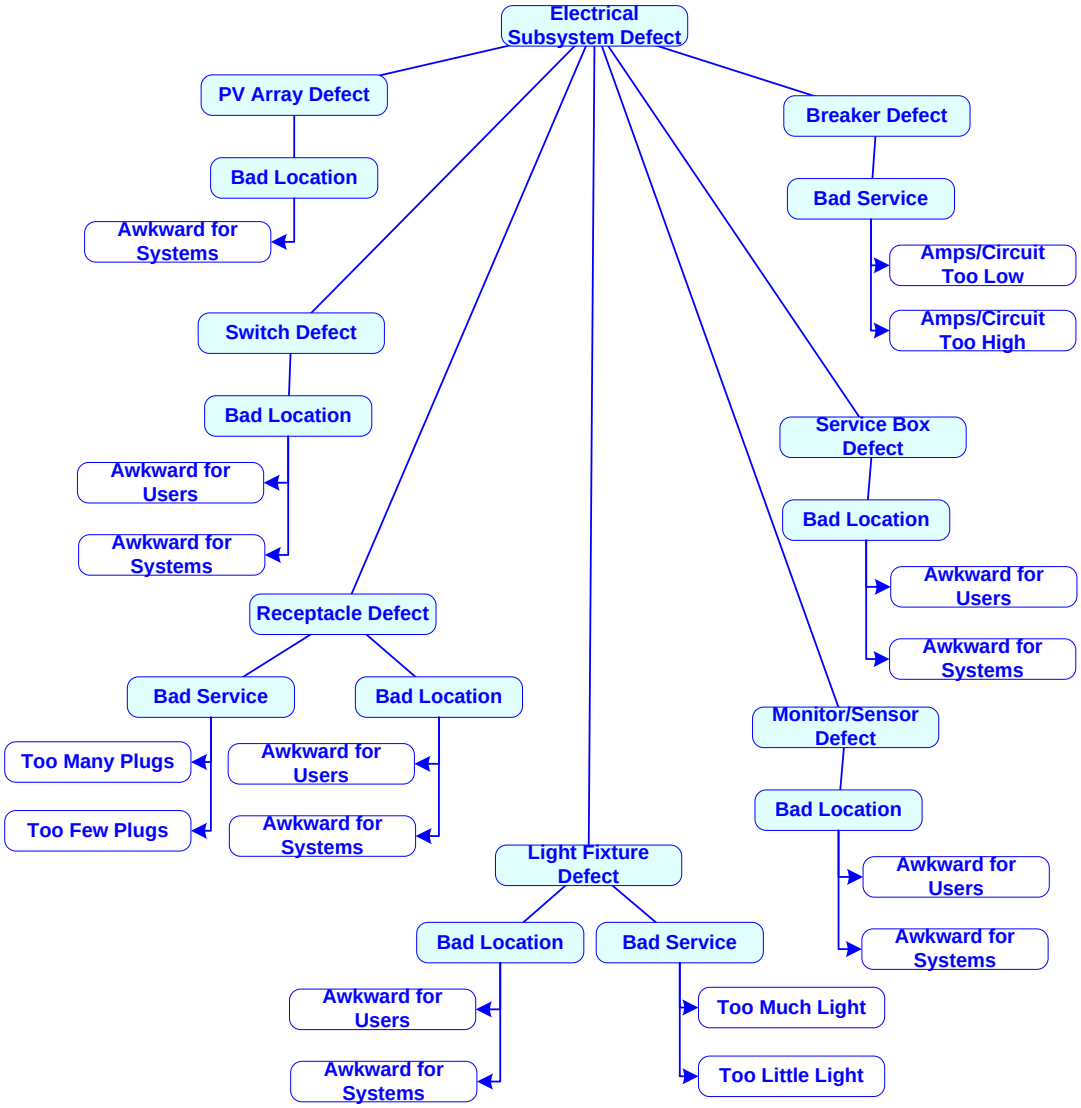


Figure 5.7 Typology of Engineering Defects for the Green Dorm Electrical Subsystem

Figure 5.7 Typology of Engineering Defects (above) provides the hierarchy of potential engineering defects in the Green Dorm electrical subsystem. The hierarchy

provides an organization around specific defects that an HVAC engineer and an architect familiar with the project defined for the thesis in interviews. Although the hierarchy illustrates the Defect Model method, it is not a formally validated typology. Figure 5.3 (on page 146), shows how PRA-VDT relates each defect type to the responsible design task and affected electrical component. The Defect Model of engineering defects connects VDT assessments about development performance to PRA and DA models of the dormitory product and operations in order to determine the most reliable among several alternative building designs.

5.2.2 Conformance Probabilities

The Defect Model quantitatively assesses the distribution of severity for each of these types that will result from VDT-assessed degrees of verification for development tasks and rework dependencies (see previous section). The analyses used one range of conformance probabilities, $[cp^{+}_{ijk}, cp^{-}_{ijk}]$ = [75%, 99.5%] for all task-defect pairs liked using the indexing structure in Figure 5.3 (on page 146). Task-defect pairs with no such link used the conformance probability range [100%, 100%]. Intuitively, each day's unverified work has a 25% chance of increasing the severity of related defect types, whereas verified work has only a 0.5% chance of generating defects. The conformance probability limits do not vary by subtask j, meaning work throughout each development task is equally capable of generating defects.

Figure 5.8, PRA-VDT-Assessed Conformance Probabilities for Each Electrical Component Type (on page 160), charts the Defect Model's assessments of realized conformance probabilities (cp_{ijk}) for each component. Those figures represent the assessed probability of each component containing no defects, and result from 100 VDT simulation trial-assessed degrees of verification (see Figure 5.6 on page 156). The results match the intuition that those electrical components in the less complex cases, and especially those in the Perfect Process case, have the highest probabilities of conforming to the design specification.

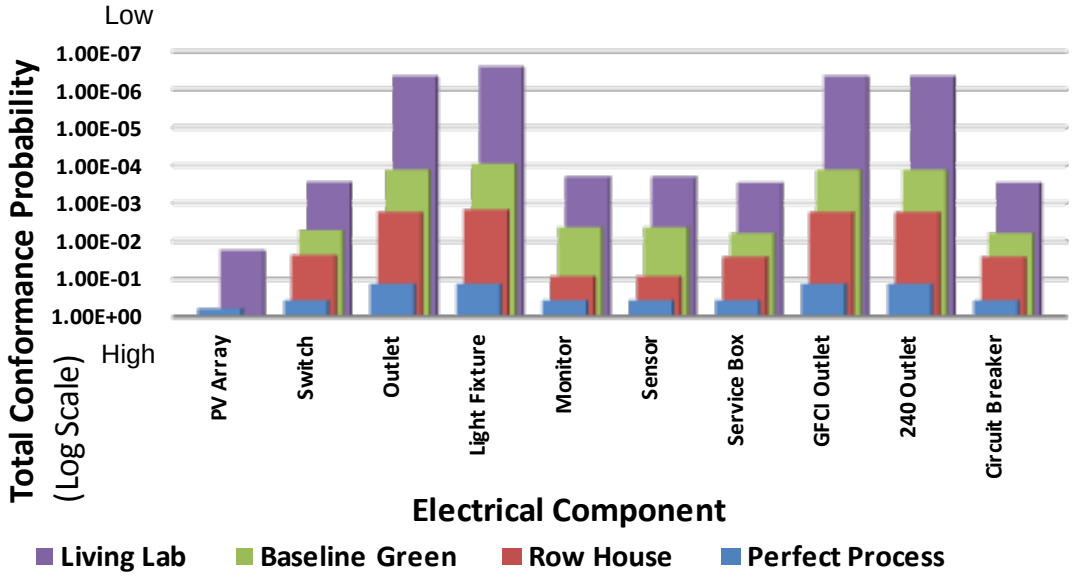


Figure 5.8 PRA-VDT-Assessed Conformance Probabilities for Each Electrical Component Type (expectations based on 100 simulation trials per case)

5.2.3 Distributions of Defect Severities

Figure 5.9 Defect Model-Assessed Total Severities of Defects affecting Green Dorm (on page 161). The Defect Model assesses different numbers of expected defects (y-axis) occurring for each of the product components (x-axis) based on the alternative chosen (represented by bar colors). The Defect Model assesses that the more complex alternatives generate more severe defects. For example, all of the alternatives (other than the No Defects case) could create defects affecting the Switch components (such as by specifying awkward placement), but are likely to create more severe defects affecting the Light Fixture components (such as by specifying illumination levels that are insufficient).

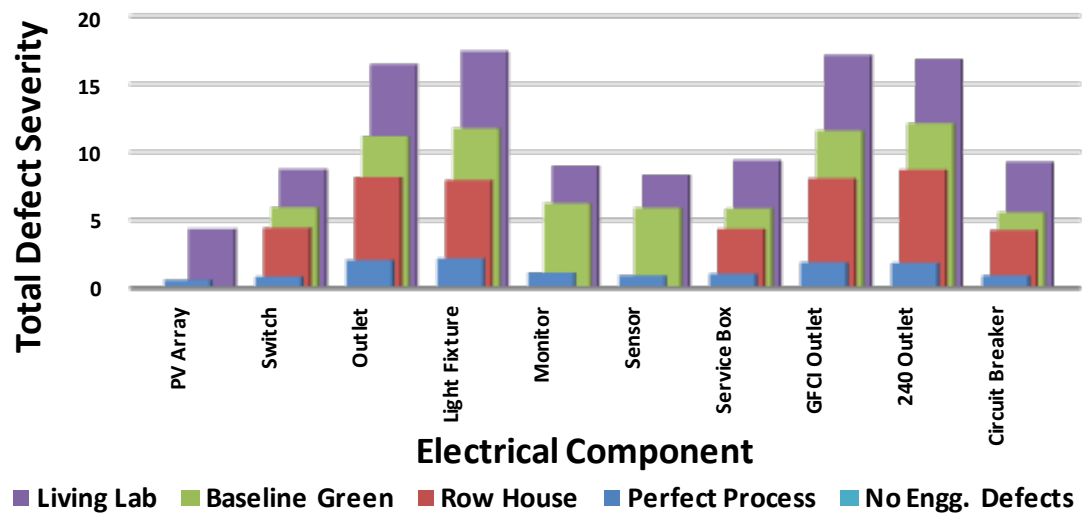


Figure 5.9 Defect Model-Assessed Total Severities of Defects affecting Green Dorm Electrical Components (expectations based on 100 simulation trials per case)

All PRA-VDT analyses use full joint distributions of engineering defects, rather the mean values illustrated in the figures. Because these assessments are quantitative, PRA-VDT can use them to assess the distribution of downstream outcomes such as loss of power in circuits and larger subsystems (see below).

5.3 Operations Model

The Operations Model assesses that the electrical system would have the least reliable components in cases where the design complexity is greatest. Higher-level subsystems that contain less reliable components would have the highest annual failure probability.

5.3.1 Component Failure Rates

Each Green Dorm Model capacity value provides the annual failure probability for an element of the electrical subsystem. For example, a 96% Photovoltaic Array capacity would indicate that each solar array is expected to fail with a 4% probability per year. The purpose of this chapter's PRA-VDT analysis is to determine the expected failure rates of components and the subsystems they compose.

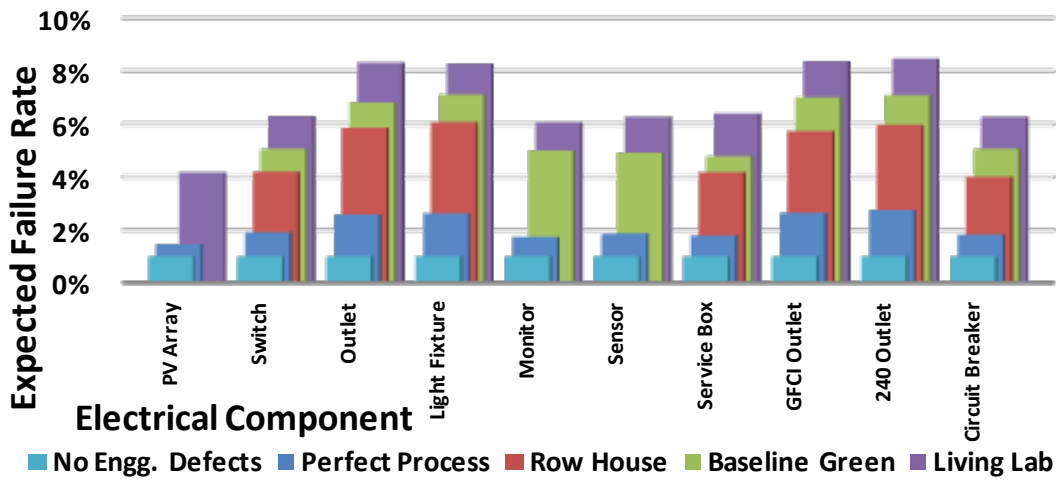


Figure 5.10 Defect Model-Assessed Failure Rates of Product Components (expectations based on 100 simulation trials per case)

Figure 5.10 Defect Model-Assessed Failure Rates of Product Components (above) presents the PRA-VDT assessed expected component failure rates from each modeled case. All of the fundamental electrical components, such as switches and outlets, use

an operating capacity (failure rate) range of [90%, 99%], which means that all components fail with a frequency between once per decade and once per century. All links between defects and corresponding capacities are calibrated using defect impacts of 90%, which means that each unit of defect severity, regardless of defect type, reduces the failure rate per year by ten percent within that [90%, 99%] range.

For each alternative project that the decision-maker chooses (represented by color), the Defect Model estimates the fraction of potential benefits (y-axis) that each product component (x-axis) can provide. These limits on capacity result from engineering defects assessed to occur when development unfolds according to VDT assessments. Although Figure 5.10 aids intuition, all PRA-VDT analyses used the simulated joint distributions of all failures, rather than the illustrated marginal probabilities.

The next section describes PRA use to assess the phenomena of fundamental component failures combining to produce broader electrical subsystem failures.

5.3.2 Subsystems' Failure Rates

Functional Block Diagram

This section uses PRA to calculate each alternative's failure rate of electrical subsystems based on the joint probability distributions of component failure rates.

Figure 5.11 (Functional Block Diagram Describing a PRA Model of the Living Lab Electrical Subsystem, on page 164) defines the PRA model linking component failures to failures in progressively larger (compound) subsystems. The diagram formally represents knowledge of residential electrical system design and specific failures that engineers described in interviews for the thesis. The block diagram merely illustrates PRA; it was not formally validated. Each of the next sections explains a level of potential failure and presents the PRA-VDT assessments of expected failure rates for the Living Lab, Baseline Green Row House, Perfect Process, and No Defects cases.

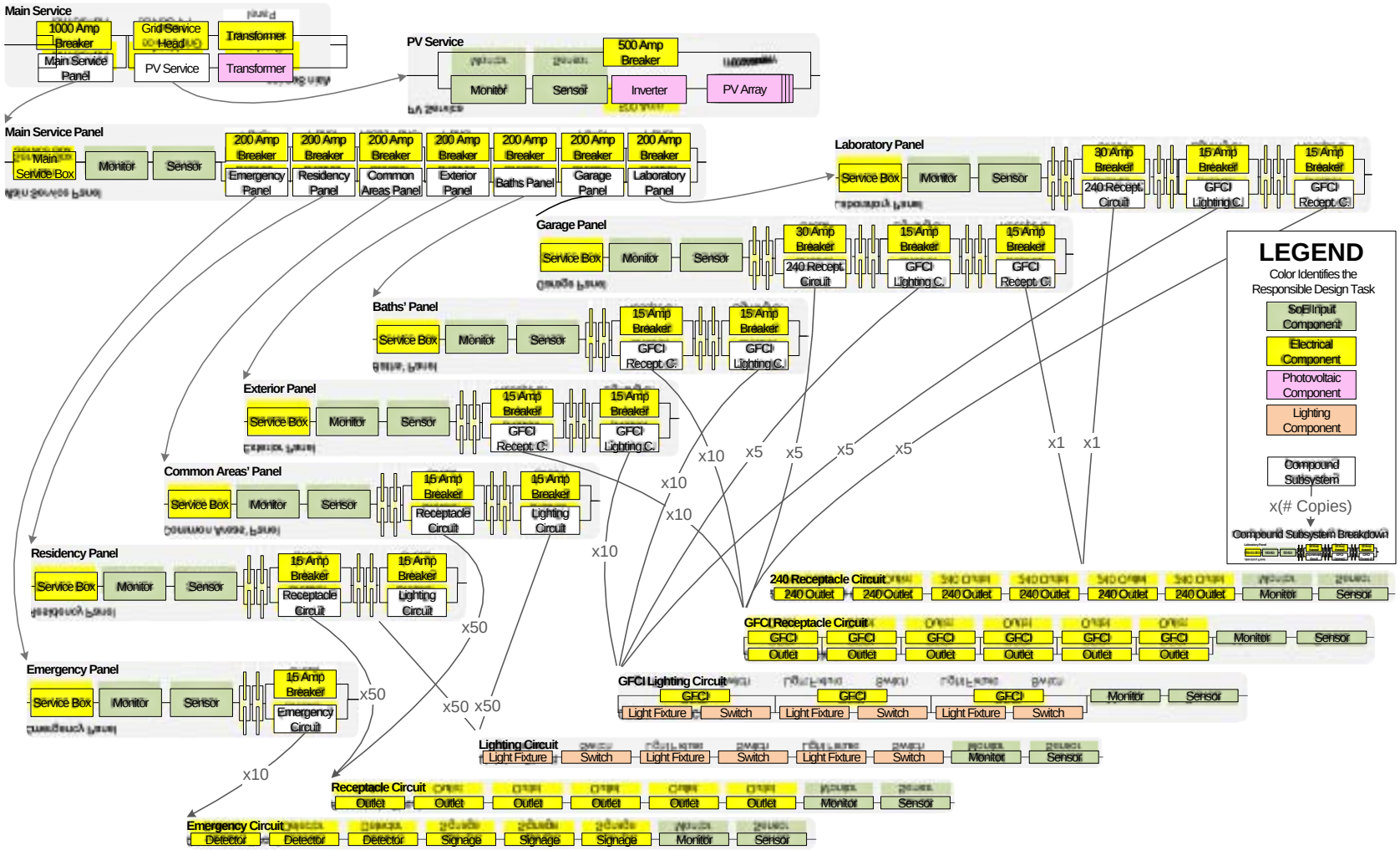


Figure 5.11 Functional Block Diagram Describing a PRA Model of the Living Lab Electrical Subsystem

Circuit Failure Rates

At the lowest level, failures in components such as switches and outlets typically cause failure of the small group of components in a circuit. When the circuit breaker trips, all the components in the circuit cease functioning. Figure 5.12 (below) charts the expected failure rates of each type of electrical circuit, for each of the five cases.

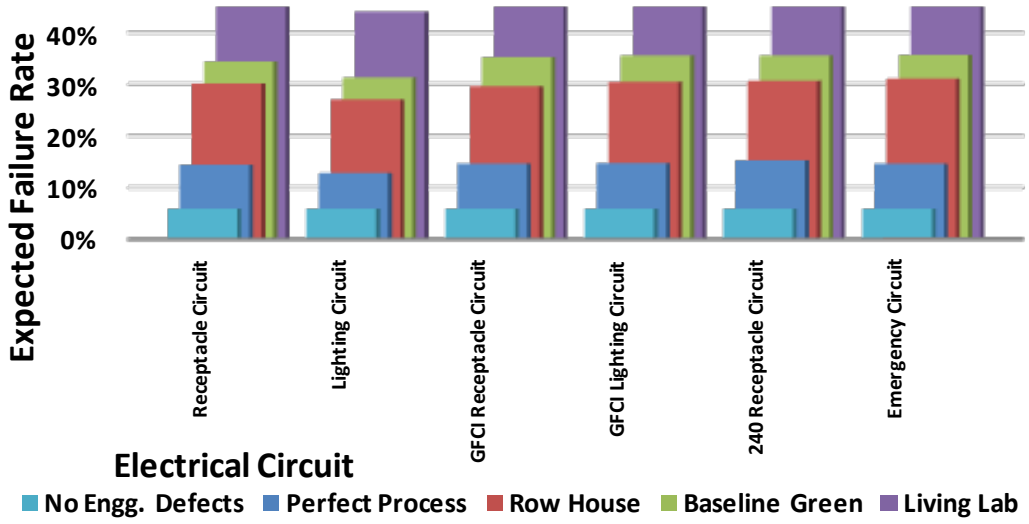


Figure 5.12 PRA-VDT-Assessed Failure Rates of Electrical Circuits (expectations based on 100 simulation trials per case)

Subpanel Failure Rates

In rare cases, circuit breakers may fail to shut down the circuit even though a failure has occurred. In this case there is a failure in the electrical subpanel that serves a large segment of the dormitory. When the subpanel's main breaker trips, the dormitory segment will lose power. However, on rare occasions the subpanel's breaker could also fail. In that case, a breaker in the main dormitory panel will typically shut down power to the full dorm. Figure 5.13 (on page 166) charts the expected failure rates of each electrical subpanel, for each of the five cases.

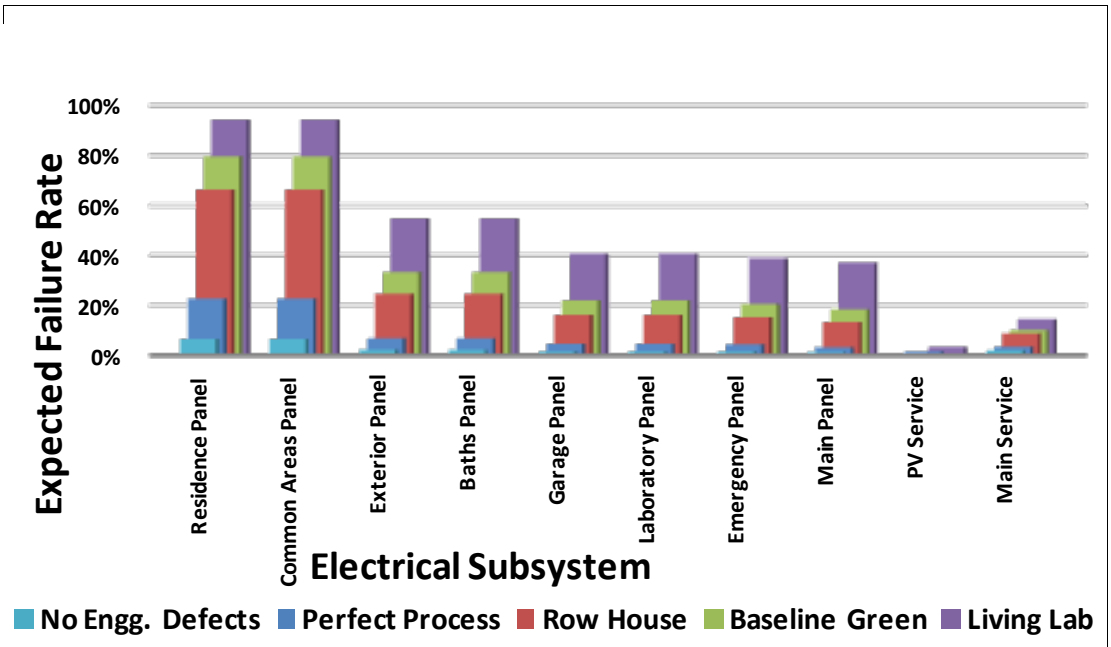


Figure 5.13 PRA-VDT-Assessed Failure Rates of Electrical Subsystems: Panels and Overall Service (expectations based on 100 simulation trials per case)

Service Failure Rates

The highest level failures defined for the electrical subsystem model occur when there is a failure in the main building electrical service, when a subpanel fails and the main breaker also fails. In addition to the electrical services typically included in a dormitory, the Living Lab will have photovoltaic (solar) cell arrays on its roof. The photovoltaic service can fail if the inverter or arrays fail catastrophically and if the built-in circuit breaker fails. Figure 5.14 (above) charts the expected failure rates of the photovoltaic and main services for each of the five cases.

5.4 Decision Model

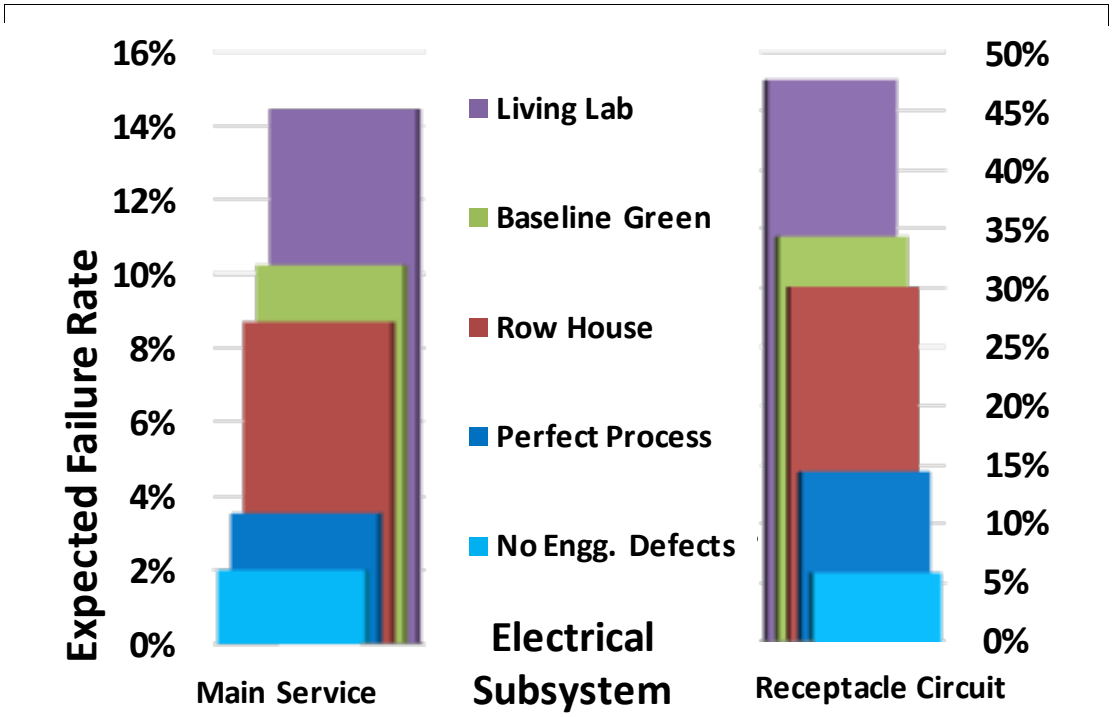


Figure 5.14 PRA-VDT-Assessed Failure Rates used to Represent Utility (expectations based on 100 simulation trials per case)

Figure 5.14 (above) presents the electrical subsystem's failure rates at the main service and residence panel levels. The analysis in this section uses these measures to indicate the future frequencies of total failures and of partial failures. This thesis finally analyzes the tradeoff between hiring a high-, medium- or low-skill photovoltaic design team. The thesis does not explicitly analyze the diverse differences in benefits that alternative projects – Living Lab, Baseline Green, and Row House- might provide.

5.4.1 Analysis of Total Failure Risks

By assessing performance level quantitatively, the thesis enables PRA to estimate the outcomes from operating the building, and enables DA to recommend which of the alternatives maximizes decision maker utility.

The analysis of main service failure rate (graphed at left in Figure 5.14 on page 167) assesses the future frequency of catastrophic failures in which an electrical failure occurs and none of the building's failsafe devices acts to limit the ensuing damage.

This thesis does not assess these failures' specific consequences. However, it would be straightforward to extend the analysis using PRA and DA to consider: specific risks of injury, property damage, loss of use, and liability.

Regardless of the magnitude of potential consequences, the PRA-VDT analysis indicates that the Living Lab's complex design will result in roughly fifty percent more catastrophic failures compared with traditional Row House dormitories. Notably, even the Row House has a far greater rate of catastrophic failures than one assessed based on the idealized Perfect Process, which in turn is greater than the case in which No Engineering Defects occur.

5.4.2 Analysis of Partial Failure Risks

The analysis of residence service failure rate (graphed at right in Figure 5.14 on page 167) accounts for the future frequency of power outages that are localized properly by circuit breakers.

This thesis does not assess these failures' specific consequences. However, it would be straightforward to extend the analysis using PRA and DA to consider the costs of temporary, localized loss of service, of routine service calls (to repair the offending outlet and reset the circuit breaker, for example), and the risk of broader harm due to low voltage arcing.

Regardless of the magnitude of potential consequences, the PRA-VDT analysis indicates that the Living Lab's complex design will result in a third more partial failures compared with traditional Row House dormitories. Notably, even the Row House has a far greater rate of localized failures than one assessed based on the

idealized Perfect Process, which in turn is greater than the case in which No Engineering Defects occur.

5.5 Photovoltaic Team Decision

The preceding analysis illustrated the PRA-VDT method's ability to assess the rates of failure for electrical subsystems in alternative projects. The analysis in this section drills down on the Living Lab alternative's photovoltaic system and meets three additional goals. First, the photovoltaic system analysis takes more full advantage of VDT's unique capabilities to look at a choice between alternative design teams that have different levels of skill and experience. Second, the photovoltaic system analysis takes more full advantage of DA's ability to make complex tradeoffs by analyzing project utility as a function of the team's labor cost in addition to failure risk. Third, the photovoltaic system analysis achieves a basic degree of formal validation. Specifically, in an interview a field expert described this section's input, analysis results, and decision-making interpretation as plausible for this project.

5.5.1 Input

Based on the assumption that the University will construct a Living Lab, the thesis goes further to assess the decision of whether to hire a Photovoltaic Design team that has low, medium, or high skill and experience. In the VDT model, the more able agents complete their work more rapidly than the less able ones. Further, the agents representing teams with greater ability tend to generate fewer exceptions. In PRA-VDT, this reduction in exceptions translates into less severe defects and fewer failures. The problem this analyses addresses is how to balance the reduced risk from having a more capable team against the increased cost.

Photovoltaic systems frequently manifest two distinctive defect types. The first, *loose connections*, occurs because the photovoltaic arrays' exposure to wind, precipitation,

and extreme temperature swings (exacerbated by the arrays' own heat generation) causes individual panels to occasionally cease delivering power to the building. Although individual connection failures would be fixed during routine maintenance, they would also tend to recur in a faulty design.

The second distinctive defect type, *wrong orientation*, occurs when the panels collect less than the full available amount of power. This can occur because of gross errors, such as miscalculation of the appropriate orientation of panels, or because the local microclimate (which includes shading by nearby construction, tree growth, and the flow of fog) differs from engineers' best assessments.

Table 5.5 Living Lab Photovoltaic Design Team Decision Data Input to VDT

Case	Supervisor	Skill	Experience	Role	Full-Time Equivalents
High PV Skill	Electrical Lead	High	High	Subteam	1.0
Medium PV Skill	Electrical Lead	Medium	Medium	Subteam	1.0
Low PV Skill	Electrical Lead	Low	Low	Subteam	1.0

Table 5.5 (above) provides the new data input for the three photovoltaic team alternatives. All other model data, notably including the PRA model, Design Photovoltaics task, and Electrical Lead agent, are the same as for the basic Living Lab.

5.5.2 Analysis

The PRA-VDT Green Dorm photovoltaic analysis used the same method as the electrical system analysis did. The first step was to simulate three VDT cases based on the Living Lab, with the only differences being between the photovoltaic teams' experience and skill levels (see Table 5.5, above). After the Development model simulated those cases to generate one hundred trials' data, the Defect Model assessed corresponding distributions of defect severities and sampled those distributions. The resulting one hundred trials' data output from the Defect Model provided input for a

PRA analysis of the probability of success for the photovoltaic array. Finally, the simulation trials' PRA-assessed success probability was interpreted as an assessment of the potential photovoltaic benefits, and the development labor (output from VDT) was interpreted as an assessment of total costs.

5.5.3 Output

Table 5.6 Photovoltaic Design Team Decision Data Output from PRA-VDT

Case	Expected PV Downtime	Expected Development Labor
High PV Skill	6.5%	1,710
Medium PV Skill	7.5%	1,719
Low PV Skill	8.9%	1,738

Table 5.6 (above) provides raw data output from the three alternatives. To analyze the decision of which team to choose, these data were combined with two additional assumptions. The first assumption is that a nominal photovoltaic design team costs $1000 per day. The second assumption is that even though the $525,000 photovoltaic array [EHDD 2006] is expected to lose money, its nonfinancial environmental benefits are important enough that the subsystem is viewed as a break-even investment when fully functional. Based on this, the DA portion of PRA-VDT analysis considers one percent of downtime equal in value to one percent of $525,000, or $5,250, of net present cost.

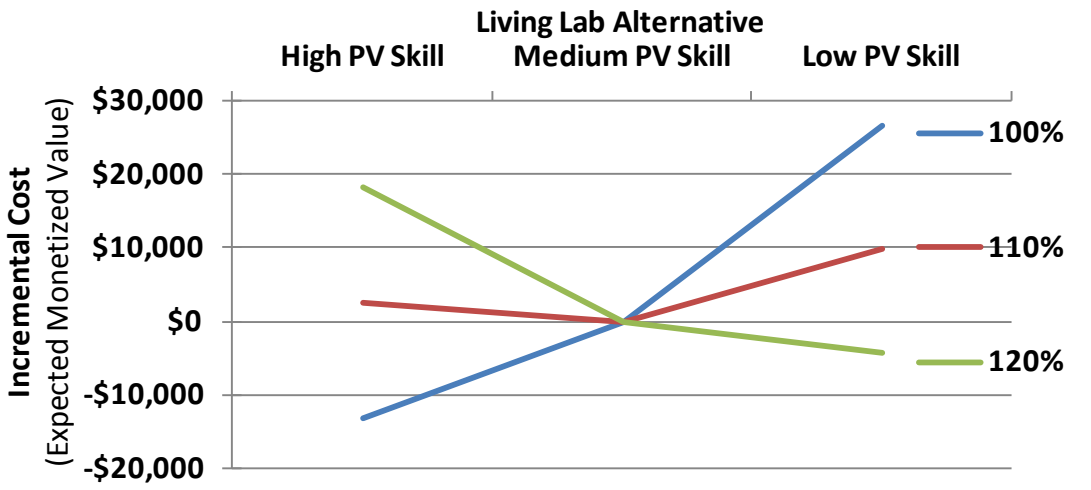


Figure 5.15 Living Lab Incremental Benefits from Alternative Photovoltaic Teams based on Three Levels of Wage Rate Premiums (based on 100 PRA-VDT simulation trials per case)

Figure 5.15 (above) charts the expected, incremental monetized cost of the photovoltaic agent decision. In the figure, the blue line indicates that the high skill agent is preferred when labor rates are the same ($1000/day) across all choices. This holds because the high skill (and experience) agent finishes the work more quickly, and produces a photovoltaic array with lower failure rates. The red line indicates that the sane trend hold, but just barely, when wages are 10% higher for a high skill agent and 10% lower for a low skill agent. When wages are 20% higher for a high skill agent and 20% lower for a low skill agent, the trend clearly reverses so that a low skill agent is preferred.

The result of this analysis is to recommend the decision maker choose an especially capable Photovoltaic Design Team if (and only if) its price premium is less than ten percent. Similarly, a less-capable Photovoltaic Design Team (having ability akin to that of the modeled agent) is the best choice if (and only if) it is more than ten percent cheaper than a basic team (or twenty percent cheaper than a highly capable team).

Conclusions

Practically, the analysis in this chapter demonstrates the PRA-VDT method on the Green Dorm electrical subsystem. The analysis highlights the Living Lab's great complexity, and shows how it could elevate the risk of both catastrophic and localized electrical failures. The analysis also contrasts the shortcomings of real projects against idealized views, either in which engineering defects never occur or in which design processes adhere to perfectionist standards. Within the photovoltaic system, the analysis in this chapter demonstrates PRA-VDT's ability to help with a localized decision that trades off the price premium of hiring expert designers against the benefits of their designing a more reliable product.

Within the electrical and photovoltaic systems, further calibrating defect and failure rate parameters, such as the conformance and capacity ranges, are appropriate for refining this estimate before attributing actionable confidence to the results. Further, analysts should assess additional subsystems, such as plumbing, structures, and heating, to develop an overall profile of risk for the building. Finally, analysts supporting the Living Lab versus Baseline Green or Row House decision should weight the assessed risks against the benefits provided by each alternative dormitory design (such as the power generated by Living Lab's photovoltaic array).

Theoretically, the electrical system analysis illustrates the PRA-VDT method. This thesis makes no claim that the resulting analytic results are plausible in industry. In contrast, a field expert did review the photovoltaic system analysis at the end of this chapter, which explains the assessment of whether hiring a particularly expert photovoltaic team is worthwhile. In an interview, Stanford Civil and Environmental Engineering Professor Haymaker (an architect familiar with the Green Dorm) stated his view that the principal data (input, output, and interpretation) are plausible for the project.

Chapter 6
Conclusion

The preceding analyses provide intuitive evidence that the Defect Model of engineering defects contributes to theories of both engineered systems and their development processes in ways that can benefit practice.

This chapter revisits the thesis analyses and claims thesis contributions and foundations, then presents avenues for further study. The thesis provides new intuition and quantitative analytic techniques that are valuable to practitioners and (as this chapter shows) to theorists.

The thesis uses a project analysis framework, PRA-VDT, which integrates the model of engineering defects with the existing Virtual Design Team (VDT) simulation of development organizations and processes, the Probabilistic Risk Analysis (PRA) model of product functions and operating contexts, and the Decision Analysis (DA) method of rational decision support. In PRA-VDT, the thesis provides a Defect Model that translates VDT output (defects' causes) into PRA input (defects' consequences), thus enabling the framework to formally explain relationships between diverse project features (such as component redundancy, engineering defects, and developer backlog) typically addressed by separate theories.

The thesis's first contribution to knowledge is a method to quantitatively relate the causes and consequences of various engineering defects in given project contexts. The

Defect Model has a method to interpret a model-based simulation to translate the volume of ignored rework in engineering subtask exceptions (derived from a VDT model of the development process and organization) into an estimate of engineering defect severities and their consequences (in a PRA model) and to assess the resulting loss of features' capacities in operations. The thesis's second contribution is a method to assess engineering defect risks in novel projects by defining the PRA-VDT Framework of interfaces between VDT, the Defect Model, PRA, and DA.

This chapter explains the Defect Model's justification and how that justification relates to the justifications of PRA, VDT, and DA. This chapter also explains how the research also sets up several exciting research possibilities, such as the analysis of interactions among existing qualitative risk theories, the assessment of additional risk factors in a project context, the expansion of automated project optimization methods, and the advancement of justification for the proposed method.

6.1 Thesis Contributions

This thesis provides a quantitative model to assess engineering defects which links causes in development-stage knowledge work and consequences in operations-stage physical processes. The examples in this thesis provides evidence for the claim that the theoretical PRA-VDT modeling method is a contribution to the knowledge of the nascent, holistic field of project optimization.

6.1.1 Contribution to Theory

The thesis defines and operationalizes relationships between development-stage knowledge work and operations-stage physical processes. It provides an integrated model that supports the assessment and selection of alternative project plans.

Chapter 1 highlighted that in spite of prior efforts, engineering-stage errors routinely result in the operations-stage failures of novel products such as space missions and oil platforms. That discussion motivated two research questions:

1. *What is a theoretically founded method that can quantitatively assess the impacts of dependencies that engineering defects create from a project's development-stage knowledge work to its operations-stage physical outcomes?*

2. *What is a theoretically founded method that can leverage the engineering defect model to compare and choose between alternative projects that have both a development stage (consisting of knowledge work with the capacity to produce defects) and an operations stage (including physical processes)?*

Chapter 2 described how existing methods can help practitioners and researchers analyze important dynamics related to these questions. However, no single existing method can answer the research questions well enough to allow decision makers to quantitatively assess impacts of difficult project-specific choices.

Chapter 3 observed that an unmet need for rework during a development stage tends to create engineering defects that increase the probability of subsequent operational failures. Based on that observation, the chapter defined a new quantitative model that relates VDT assessed engineering stage events (degree of verification), to dependent PRA estimates of engineering product capacities (for components, interfaces, and interdependent multi-systems).

Chapter 3 also situated the Defect Model within a context of existing tools. Table 1.1 (a, b, and c) presented the intuition that risk mitigation should involve preventing the overlap of weaknesses among product, organization, process, and context factors. The PRA-VDT framework introduced by the thesis sharpens this intuition (in particular that of Table 1.1c) to serve decision makers with quantitative decisions regarding real-world projects. The framework first determines the adequacy of an organization to its assigned process, and then determines the adequacy of the product to endure its operating environment. When a required process will be difficult for a given organization to complete, for example, the method will tend to recommend a product that is resilient when compared to its environment (for example by including

component redundancy or by adopting a slower but more robust operations plan). Similarly, if the product will confront overwhelming hazards, the method will tend to recommend an organization that is able to execute the engineering to a very high standard (for example by hiring the best available team, or adding extra test cycles).

Chapter 4 demonstrated the Defect Model and PRA-VDT framework on a hypothetical communications satellite project. The illustrative analysis evaluated the tradeoffs between two risk reduction methods: design simplification and operations redundancy. For a hypothetical data set, the assessment found component redundancy was marginally beneficial in spite of creating additional design complexity and engineering defects. *The chapter intuitively demonstrates that PRA-VDT can provide new insights by comparing the weights of counterbalancing imperatives.*

Chapter 5 demonstrated the Defect Model and PRA-VDT framework on the Stanford Green Dormitory Project. The illustrative analysis, based on field informants' data, indicated that the highly sophisticated building options would suffer most from electrical system defects due to the complexity of design. However, the least sophisticated dorm options are still likely to provide far more reliable and safe electrical systems than projects modeled under assumptions of ideal engineering quality. *The chapter provides intuitive evidence that PRA-VDT can provide new practical insights by comparing the weights of counterbalancing practical imperatives.*

6.1.2 Theoretical Implications

This research extends project planning and risk management research traditions. Its contribution to engineering risk analysis is that it formally defines theory-founded methods to relate risks in the operational phase product with design phase choices of organization and process design and the design phase environment. The thesis uses VDT information about the match between organization and process to calculate a new *degree of verification* measure, and it assesses the ways in which a low degree of

verification can lead to defects and reduced functional capacities in operations. The thesis also provides a PRA-VDT framework that enables each of several models to work together and apply unique strengths, while compensating for the others' shortcomings. The assessed result includes an estimate of operations failure probability that accounts for flaws introduced during the early design and development stages. The PRA-VDT framework therefore offers a structured method for the formal evaluation of the risk effects of many controllable design and management choices. The thesis provides several illustrative examples and discusses methods for expanding on the thesis.

Because the thesis preserves the core, theory-founded PRA and VDT models, it provides a formal definition of the ways in which many theoretical factors—such as component redundancy, human error, constructive oversight, and information processing—interact to determine and decrease technical failure risk. This contribution can lend precision to the communications among traditionally engineering and social science disciplines and this precision can improve the rates of constructive consolidation and agreement in the field.

Using the Defect Model, PRA-VDT can coordinate the product, organization and processes of the operations phase through consideration of risk elements in early design and development. Planning decisions that appear to fit easily into the integrated model include product component and subsystem redundancy and configuration; organizational participants' skills and structure; processes of design and development; and engineering collaboration mechanisms and authority distribution (e.g., centralization). The framework's broad view will provide a more realistic assessment of operational failure risks than models that are limited to consider operations or engineering alone, and the method will consequently make a broad range of mitigation strategies analytically tractable. With a united model of the engineered system, engineers will be better equipped to make decisions consistently and in alignment with management objectives.

6.2 Validation and Justification

The illustrative studies indicate that applying the Defect Model in the context of PRA-VDT can provide insight, and might therefore be justified for large, novel projects having significant engineering defect risks.

6.2.1 Purposes and Processes of Justification

A central matter in the evaluation of management science methods is their justification- their ability to provide benefits that outweigh the costs of modeling effort. The illustrations in this thesis indicate that modeling in PRA-VDT adds a small amount of effort to that required to use VDT, PRA, and DA.

The section considers the degree to which the Defect Model can address the observed problem by helping human experts to assess project outcomes and to identify the best of several alternative plans. The model's assumptions limit its generality, and the model's approximations limit its power, so that the goal of modeling is to make the best of statistician George Box's observationthat "All models are wrong, but some are useful" [Box and Draper 1987]

Regardless of the Defect Model's assessment accuracy, it can serve as a creativity tool that suggests possible project dynamics that would not otherwise be considered, and that can be independently verified. The Defect Model has become increasingly useful, however, and can continue to become increasingly useful, through stages of progressive refinement [Burton and Obel 1995, Carley 1996, and Thomsen et al 1999]. Models, like theories and human experts, are most influential after they earn stakeholders' confidence [Feigenbaum 1988]. The goal of formal justification therefore is to provide broadly acceptable evidence that the model serves its purpose: that under specific circumstances (generality), the modeling investment provides returns (power) exceeding those of known alternatives (the points of departure). The thesis method's many explicit assumptions limit the method's generality to those cases

where the assumptions hold. Similarly, the thesis method's many explicit approximations limit the method's power to assess uncertainties only with a corresponding limit to resolution. Varying or eliminating these assumptions and approximations provides many avenues for extending the value of this thesis, and §6.3 (starting on page 181) details some of these avenues.

6.2.2 Relationships to Justification of Foundations

The Defect Model's justification relates to the justifications of PRA, VDT, and DA in several ways. The Defect Model's justification rests on PRA's validity because PRA translates the Defect Model's assessments of component failure rate into risk estimates that can support project shaping decisions. §2.2.3 (on page 35) explains PRA's current academic validation and practical justification. The Defect Model's assessments of development behavior impacts can help justify PRA for a given project by improving the accuracy of failure probability estimates.

The Defect Model's justification rests on VDT's validity as well, because VDT provides development behavior assessments that the thesis uses to assess defect quantities. §2.4.3 (on page 49) explains VDT's current academic and practical validation. The Defect Model's assessments of development behavior's impacts can help calibrate VDT by improving the precision and verifiability of development process risk reporting.

Finally, the Defect Model's justification rests on DA's validity, because DA translates the Defect Model's insights into recommendations for action. §2.2.5 (on page 39) explains DA's current academic and practical validation. The Defect Model's assessments of dependencies between development and operations outcomes can help justify DA for a given project by informing decisions with more accurate joint outcome distributions.

6.2.3 Defect Model and PRA-VDT Justification

The thesis has taken the first essential steps toward justifying the Defect Model. Chapter 1 observed a problem in the field that requires better assessment, and explicitly delimited the Defect Model's purpose. Chapter 2 delimited the Defect Model's claim of generality by locating the research questions within a gap among currently available methods. Chapter 3 delimited the Defect Model's claim of power to enable PRA, VDT, and DA to address the observed problem more fully than existing methods. Chapter 4 and Chapter 5 also conveyed the algorithm's intuition and established face validity by presenting simple "Intellective" "Toy Problems" [Burton and Obel 1995, Carley 1996]. Chapter 5 further demonstrated the method's power by providing plausible answers to difficult questions regarding a real world project, using a set of transparent and plausible assumptions and reasoning steps. The field study also demonstrated the method's intuitive validity by evaluating two idealized cases.

6.3 Extensions

Substantial enhancements could refine the method's risk assessments, or could expand the method to answer difficult theoretical and practical questions.

6.3.1 Further PRA-VDT Justification

The next steps in Defect Model validation would include a formal statistical validation that the PRA-VDT model improves project planning decision-making. Such a study would compare assessments with those of recognized human experts to determine whether the model could serve in the role of human expert. In cases where human expertise is less well established, the validation could compare model assessments, human expert assessments, and actual outcomes (such as bug reports) in real projects. Because PRA-VDT includes PRA, VDT, and DA, such a formal validation would rely upon the prior advancement of similar validation for those existing methods.

6.3.2 Engineering Enhancements: an Inverse Model of Engineering Defects

Just as *low-functioning* development organizations often fail to meet the product's target performance specification, *high-functioning* organizations often *exceed* the target specification. Although exceeding performance goals sometimes happens by accident, enhancements result more often when particularly well-qualified engineers identify and implement beneficial changes based on the discovery of unexpected opportunities [Thomsen 1998].

A minor change to the Defect Model enables PRA-VDT to assess the emergent deviation of product from specification for beneficial engineering enhancements (in addition to detrimental engineering defects). In this model, enhancements are more likely for verified work than for unverified work, which is the inverse relationship from defects. One simple method assesses the distribution of enhancements that increase capacities, using a parallel method as for defects that reduce capacities.

The enhancement model requires just three changes to the base model. First, the modeler defines an "enhancement type," indexed in k, along with the existing "defect types." For example, $k = 8$ could index engineering enhancements to the satellite payload that can increase expected longevity beyond the specification's target.

Second, for the new enhancement types, conformance represents the failure to enhance, rather than the successful avoidance of defects. The enhancement model, therefore, inverts the conformance probability limits (cp^{-}_{ijk} and cp^{+}_{ijk}, used in Eq. 3.2 on page 82); a high degree of verification (dv_{ij}) implies a low conformance probability (cp_{ijk}):

$$cp_{ijk} = cp^{+}_{ijk} + dv_{ij} \times \left(cp^{-}_{ijk} - cp^{+}_{ijk} \right) \qquad \textbf{Eq. 6.1}$$

Third, whereas defects reduce capacities, enhancements increase capacities. This model, therefore, inverts the operations capacity limits (oc^{-}_{l} and oc^{+}_{l} used in Eq. 3.18

on page 96); development projects creating many enhancements (high values of s_k), or enhancements with significant influences (low values of di_{kl}), produce high operations capacities (oc_l):

$$oc_l = oc_l^+ - \left(oc_l^+ - oc_l^-\right) \times \prod_k di_{kl}^{s_k} \qquad \textbf{Eq. 6.2}$$

As with defect types, enhancement types may represent novel behavior requiring the formulation of significant model extensions. Preliminary analysis indicates, however, that the existing thesis model and PRA-VDT framework readily accommodate those extensions.

6.3.3 Post-Accident Investigation

The dissertation focuses on a contribution to project modeling that can help design projects in advance, however the thesis and PRA-VDT can also help analyze projects that have already gone awry. This "forensic" mode of analysis could particularly benefit investigations that have limited information with which to reconstruct conditions existing before a catastrophe, and could help guide the attention of teams having limited resources with which to investigate large design teams.

Consider, for example, this claim from the Columbia Accident Investigation Board:

> *The four flights scheduled in the five months from October 2003, to February 2004, would have required a processing effort comparable to the effort immediately before the Challenger accident.*
>
> *-NASA 2003*

Comparing models that represent typical shuttle mission preparations to models representing the preparations preceding the final Columbia and Challenger launches could help direct attention to the distinctive aspects of complex launch failures.

6.3.4 Projects Having Multiple Stages

In practice, many projects involve myriad stages, such as specification, design, development, testing, operations, and decommissioning (after operations). Figure 6.1 Influence Diagram Illustrating Multiple-stage and Multiple-Error-Source (on page 185) shows how using a multi-stage VDT model could support analysis of dependencies between all of these stages [Chachere 2005]. These extensions begin by using VDT to model a multiple-stage project. VDT analyzes each engineering stage's organization and process and assesses three types of risk related data. Each of these measures of engineering conformance relates to a distinct risk to the engineered product using three corresponding data integration points. PRA then calculates the significance of these risks within the broader operating context.

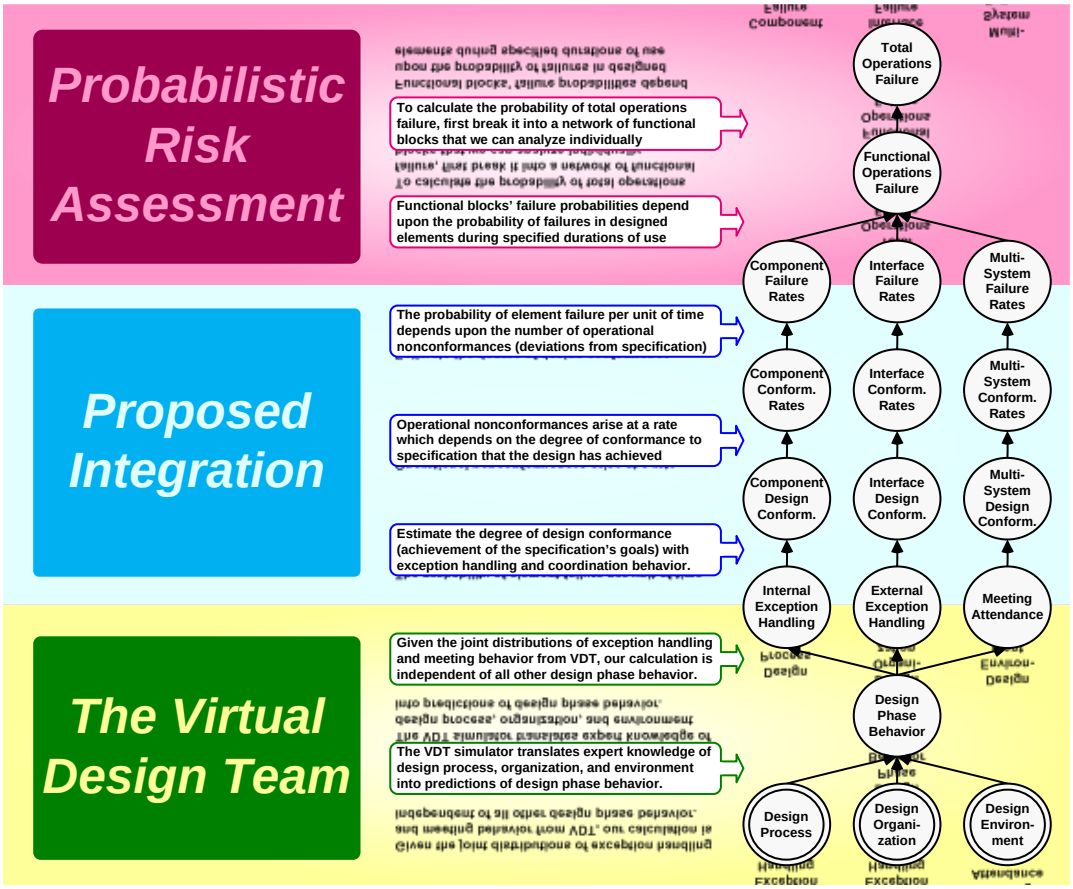


Figure 6.1 Influence Diagram Illustrating Multiple-stage and Multiple-Error-Source Extensions to the Thesis

6.3.5 Modeling Broader Utility Functions

Project modelers may wish to broaden the utility function in the PRA-VDT Decision Model to consider additional measures of interest, such as development cost and operations revenue. Mathematically, this extension only requires adjusting Eq. 3.36 (on page 109), as shown below:

$$E\{u[B(a)]\} = \int_d \sum_p \int_o f[D(a_d) = dv] \times p[S(dv, a_p) = s] \times f[O(d, s, a_o) = o] \times u[(d, s, o)] dd\ d$$

Schematically, this extension substitutes Figure 6.2 Decision Diagram Overview of the PRA-VDT Framework (on page 186) for Figure 3.1 (on page 61).

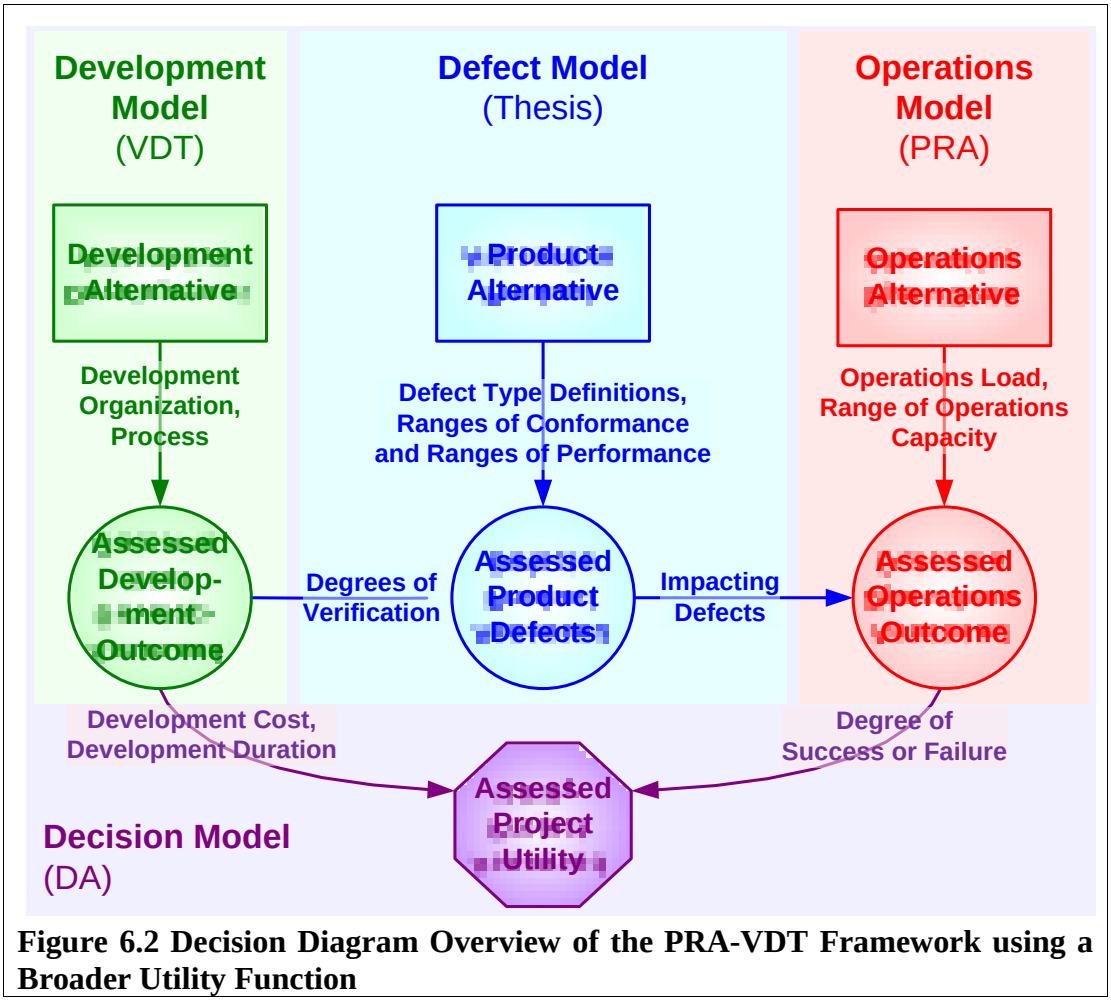


Figure 6.2 Decision Diagram Overview of the PRA-VDT Framework using a Broader Utility Function

6.3.6 Automating the Search for Optimal Plans

The thesis shows how planners can test plans for organization, process, product, and context, but not how to generate those alternatives. The thesis features thus complement existing research that automatically generates project alternatives, but that uses primitive methods for evaluation. Pugnetti [1997, p. 143] noted that genetic algorithms [Koza 1992, Holland 1975] can effectively generate the large set of highly interdependent organization and process variables. KHosraviani, Levitt, and Koza [2004] and KHosraviani [2005] applied genetic algorithms to project design using VDT and were able to outperform human experts. However, previous work has lacked sophisticated fitness functions that the thesis and PRA-VDT offers. Genetic

algorithms could also generate compatible operations stage plans, creating a more complete multi-stage project optimization engine.

6.3.7 Assessing the Accuracy of Cost, Quality, and Schedule Estimates

The Defect Model-assessed defect severity distributions could help estimate the chance of cost, schedule, and other overruns resulting from flawed (mistaken) designer estimates of downstream behavior. As with traditional engineering tasks, project planners require appropriate experts as well as an effective collaborative process to correctly estimate a project's programmatic risk, costs, and schedule. The Defect Model therefore applies to assessing the likely severity of defects in ***estimates*** of the cost, quality, or schedule of downstream project stages, even when the actual cost and other measures are not explicit within the model.

6.3.8 Modeling Additional Risk Sources

One avenue to enhance the Defect Model is to assess additional risk precursors (other than ignored exceptions) and to determine their implications using new integration points between VDT, the Defect Model, PRA, and DA.

Models of Backlog and Corner-Cutting

The backlog and exception-handling behavior that VDT assesses are important measures of safety culture according to Ciavarelli 2003, and Cooke et al 2003. Workers under stress are particularly liable to make mistakes [Cooke et al 2003], and so it is reasonable to expect a relationship between the number of gross errors that enter operations and the backlog of engineers. Altering VDT to increase precision in this area could view work that falls behind as creating a conflict of interest between an agent's need to meet schedule and a project's need for design robustness, because this moral hazard tends to produce risky "corner-cutting" behavior. VDT models of

coordination by backlogged team members models corner cutting by "oversight", and it reports the amount of backlogged work for every agent over the course of the simulation. A post-simulation analysis (parallel to that provided in this thesis) could preserve the VDT behavior and justification by leaving the VDT reasoning intact and merely extending the interpretation of backlog output.

Refining the Views of Defect Causes and Consequences

Applications requiring greater detail in pinpointing the source of defects could expand upon the Defect Model's claim that the probability of defects is a function of exception handling. In particular, VDT models exception generation and handling as a function of several different underlying organizational causes (as outlined in §3.1.2) that each might have distinctive operations consequences. The information dependency relationships among engineering tasks are often (but not always) isomorphic to the operational behavior dependencies among engineered product components [Sosa et al 2004].

Figure 6.1 illustrates this proposal in a multiple-stage project. When VDT agents fail to attend a meeting, this increases the probability of exceptions, and therefore of defects, in all of the tasks whose assigned agents are invited to the meeting. In the existing Defect Model, low meeting attendance therefore tends to increase the severity of defects in all of those tasks. However, a further refinement of the model could create an explicit mapping between meeting attendance and defects. These defects could be particularly likely to create "system failures," that Normal Accident Theory (§2.3.2) defines as prevalent in highly interdependent systems. Informal communications completion rates could also help determine the expected number of "interface failures" corresponding to the operations of two related subsystems. These model assessments could lend precision to discussions of why complex engineered systems have not, in fact, failed as often as was once assessed [Sagan 1993, Sagan 2004].

6.3.9 Quantifying and Comparing Theories of Human and Organizational Risk

Although many of the dynamics this model illustrates have precedents in the social science literature, it is unprecedented for these sometimes-conflicting theories to be operationalized quantitatively. An important contribution would be to determine the degree of agreement among the PRA-VDT model's components, social science theories, and the empirical evidence that has supported those theories.

Deploying the Defect Model in a PRA-VDT context provides an opportunity to quantify, test, and compare seminal theories of organizational risk that are often qualitative, controversial, and conflicting. Specifically, the PRA-VDT framework's precision might enable the comparison of competing theories of human and organizational risk management, and the eventual determination of how their dynamics interact under specific project circumstances.

For example, the proposed method can cross validate qualitative risk analysis theories such as Normal Accident Theory NAT (§2.3.2) and the theory of High Reliability Organizations (HRO) (§2.3.3). The satellite example in Chapter 3 provided one example of many simple intellective experiments that could use idealized or representative PRA-VDT models to illuminate the ways in which NAT's assessed risk sources, complexity and interdependence, balance against HROT's remedies, effective communications and redundancy.

PRA-VDT can also evaluate changes in safety culture by adjusting the VDT-simulated agent decision-making behavior (by changing the fraction of decisions to rework versus quick-fix or ignore exceptions). In the field, both strategic and tactical mission designers' concern over safety balances against the importance of meeting schedule and cost budgets. Although choosing to rework a component in a simple response to strong safety climate can improve the reliability of a finished product, the increased

schedule pressure can lead unexpectedly to stress-induced errors instead [Cooke et al 2003].

6.4 Summary

This thesis identifies engineering defects as a frequent source of failure in complex engineered systems, such as spacecraft and novel civil facilities. Currently, theoretical and practical methods to assessments of design team performance and of designed system reliability handle defect-induced failures imprecisely or ignore them altogether.

This thesis both provides a Defect Model for quantitatively assessing engineering defect risks, and defines its interfaces with the VDT model of engineering projects, the PRA model of engineered system risks, and the DA method of decision support. The Defect Model elicits quantitative judgments from project experts regarding different engineering defects' causes in knowledge-based development and those defects' consequences in physical operations. With those data, the thesis models development-stage shortcomings as a function of failures to complete necessary rework, interprets those shortcomings to assess distributions of engineering defects, and estimates those defects' potential to reduce the developed product's capacities during operations.

The thesis method provides particular *theoretical* value by providing a quantitative theory linking phenomena that occur across time and across disciplinary boundaries. The thesis demonstrates that fact when illustrating the method on a hypothetical satellite project, by showing how the method resolves a conflict between two theory-based risk management heuristics (increasing redundancy and reducing complexity).

The integrated method provides particular *practical* value by quantitatively assessing risks in projects involving complex engineering efforts that culminate in operations at significant risk of failure. The thesis demonstrates that fact when illustrating the method on a real, highly innovative dormitory project, by showing how the method assesses the risks incurred by increasing complexity beyond existing precedents.

Appendix A
VDT Exception Handling

Exceptions

VDT divides each simulated development task into smaller portions of activity, termed subtasks. Figure A.1 (on page 193) illustrates how, after completing each subtask, a VDT agent evaluates the work's internal consistency (conformance to specification), and sometimes raises a functional exception. The agent also evaluates the work's external consistency (conformance to constraints placed by other tasks) and sometimes raises project exceptions for one or more rework-linked tasks.

The exception handling process can lead to four events:

Functional Exception Event that the agent assigned to task *i* believes its completed work on subtask *j* will lead to functional defects if used in the product, and generates an exception that may cause rework in the task

Functional Verification Event that the agent assigned to task *i* believes its completed work on subtask *j* will not lead to functional defects, and generates no exception

Project Exception Event that the agent assigned to rework dependency *i* believes its completed work on subtask *j* will lead to project defects if used in the

product, and generates an exception that may cause rework in another, dependency-linked task

Project Verification Event that the agent assigned to rework dependency *i* believes its completed work on subtask *j* will not lead to project defects, and generates no exception

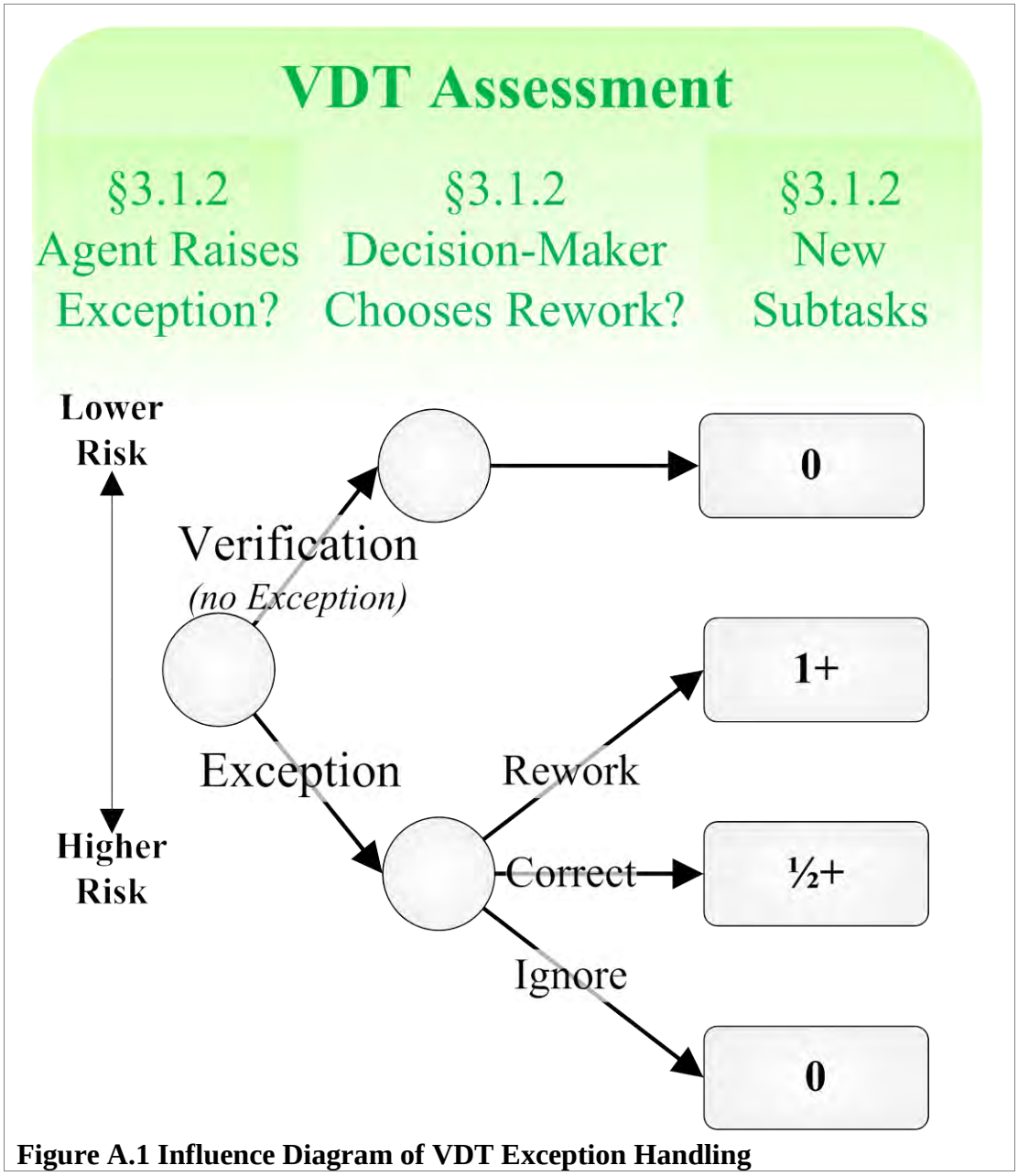


Figure A.1 Influence Diagram of VDT Exception Handling

Exception Probabilities

The first clear indicator of product performance VDT mechanics provide is the ***Verification Failure Probability***, or simply VFP. VFP is the probability that an agent will perceive a just-completed work item as not conforming to specifications, and will raise an exception. An exception is a warning sign, suggesting that the work may not meet all the appropriate requirements – the organization can respond to an exception in various ways.

These VDT assessments support a risk model by deriving exception-handling behavior from a wide range of factors that are linked to downstream operations failures. The factors include:

Conditions that **directly** increase (decrease) a task's *functional VFP* are:

- Assigned agent has low (high) *skill*
- Task has high (low) *requirements complexity*
- Prior *functional exceptions* are (not) ignored

Conditions that **directly** increase (decrease) a task's *project VFP* are:

- Assigned agent has low (high) *experience*
- Task has high (low) *solution complexity*
- Prior *project exceptions* were (not) ignored
- Prior *meetings* had many (few) absentees
- Prior *communications* were (not) dropped

Conditions that often **indirectly** increase a task's VFP are:

- Backlog and overloading of agents with related work (per coordination and meetings)
- Backlog and overloading of decision-making supervisors
- Delegation of decisions to roles with weak safety culture (tendency to ignore exceptions)

In the Defect Model, higher VFP typically increases the number of engineering defects and therefore significantly reduces performance during the operations phase.

Exception Handling

After simulating an exception, VDT creates a virtual *decision*—a work item that is routed to an agent at or above the working agent's level in the hierarchy (probabilistically, depending on the project's centralization). The decision-making agent chooses among three virtual alternatives diagramed in Figure A.1 (on page 193): money- and time-consuming ***rework***, cheap and fast – but high-risk – ***ignore***, and middle-ground ***quick-fix***. Choosing to rework or quick-fix creates a work item (of full or half work volume, respectively) that can also create exceptions. VDT samples the simulated decision as a random variable distributed according to safety culture, agent skill, and decision-maker role.

The thesis identifies these VDT assessments of development process behavior as essential to determining the risk of engineering defect inclusion, and captures the VDT assessments using a new metric, the degree of verification. Figure 3.5 (on page 73) extends Figure 6.1 by linking exception handling explicitly to distributions of defects.

Appendix B
Discussions of Simulation and Probabilistic Dependencies

This section first explains why the thesis uses simulation to solve the mathematical equations relating development, product, and operations measures. Following that rationale, the section reviews and expands upon previous discussions of how and why the method preserves many (but not all) of the possible probabilistic dependencies between development, product, and operations factors.

Simulation Method

Closed form descriptions of project dynamics are ideal, because they provide exact precision and symbolically identify the sources of contributing factors. The availability of closed form solutions is certain, however, only for analyses that have specific formulations. The Development Model, in particular, is based on the existing VDT simulation, rather than on closed form mathematics.

Simulation in the Development Model

Analytically solving Eq. 3.36 is difficult because VDT uses Monte Carlo simulation to describe the complex, stochastic development process; the Development Model

assessments are not closed form equations, but instead are samples (representing possible development behavior) from a distribution with complex and unknown form.

In general, compared with simulating consistently throughout PRA-VDT, substituting a continuous probability distribution that reasonably approximates VDT assessments into the continuous models of defects, operations, and decisions carries an elevated risk of developing a formulation having no known closed form solution. Prior VDT research has characterized output using just the mean and variance of each outcome measure, but this lacks the precision required for analysis of risks due to unlikely events. It is possible to approximate the VDT output by statistically fitting a continuous joint probability distribution having more degrees of freedom. One can approximate the sampled data, for example by calibrating Gaussian distributions (with full covariance matrices) using a maximum likelihood estimator. However, using a continuous approximation can occlude features that are important to risk analysts dealing with rare events. For example, whereas Gaussian distributions have nonzero densities everywhere, the ratio of design cost to duration *never* exceeds the wage rate. Even accepting the approximation, estimating a continuous distribution's implications for the rest of the project (by expanding Eq. 3.36), however, may complicate the framework's analytic complexity to an arbitrary degree.

The Development Model uses VDT to generate samples of *dv* for Eq. 3.36. Where *t* is the number of simulation trials, and d_{ar} is the development behavior (realization of *D*) that VDT simulates for plan *a* (using arbitrary random seeds indexed by *r*):

$$\max_{a \in A} E\{u[B(a)]\} =$$

$$\lim_{t \to \infty} \sum_{r=1}^{t} \frac{1}{t} \sum \int p\left[P\left(dv_{ar}, a_p\right) = s\right] \times f\left[O\left(s, a_o\right) = o\right] \times u(o) do \qquad \textbf{Eq. B.1}$$

Simulating the Defect and Operations Models

Analysts can maximize flexibility when formulating the PRA-VDT model by planning to simulate *all* of the terms in Eq. 3.36: by sampling the distributions of *D* (development), *P* (defects), and *O* (operations). This assumes sufficient computing

power (which Chapter 5, on page 138, speaks to). The mean of these samples forms an unbiased estimator of the expected value of the full distribution [Law and Kelton 2000]. That value, the average simulated outcome, approximates (to any desired precision) the expected utility that rational decision makers wish to maximize.

Simulating all four PRA-VDT models is sufficient because it allows analysts to model using a nearly complete range of mathematical tools and to obtain results that have any required level of precision [Law and Kelton 2000]. The applications in this thesis solve the integrated system by simulating one sample path through the Defect, Operations, and Decision Models for each VDT output value generated in the Development Model.

Simulating all four PRA-VDT models is sometimes necessary because of this method's mathematical complexity. Extending the VDT tradition of simulation through the Defect and Operations Models controls the PRA-VDT application's analytic complexity.

The full-simulation method alters Eq. B.1 by simulating one trial in the Defect and Operations Models for each of the *t* development trials. Defining p_{ar} as the product sampled for alternative *a*, and defining o_{ar} as the operations behavior sampled for alternative *a* using seed *r*, produces:

$$\max_{a \in A} \left\{ E\{u[B(a)]\} = \lim_{t \to \infty} \sum_{r=1}^{t} \frac{1}{t} u(o_{ar}) \right\} \qquad \textbf{Eq. B.2}$$

Table B.1 (on page 199) illustrates the process of sequentially simulating VDT and the Defect Model. The table includes ten columns, each of which provides data for a single simulation trial with both VDT (upper portion of the table) and Defect Model calculations (lower portion). Models typically require more trials to establish the full distribution (for example, the next chapters' illustrations use 100 and 1000 trials respectively).

Table B.1 Illustrative Data Generated for Ten Simulation Trials of the Green Dorm Project PRA-VDT Model

	Simuation Trial # 1	2	3	4	5	6	7	8	9	10
$E_j(dv_{ij})$ **Task Mean Degree of Verification**										
D. Architecture	76.4%	79.3%	80.9%	78.2%	79.4%	80.2%	85.6%	84.9%	81.0%	72.1%
D. Civil	84.4%	85.4%	88.0%	86.3%	87.8%	84.8%	83.4%	90.4%	83.3%	87.4%
D. Integration	24.4%	30.2%	37.5%	25.6%	25.0%	24.0%	37.5%	62.5%	27.0%	32.5%
D. Electrical	81.1%	89.7%	88.8%	89.0%	84.0%	87.5%	84.6%	92.4%	86.0%	87.3%
D. HVAC	83.1%	82.0%	91.8%	85.0%	83.4%	83.8%	91.1%	84.1%	83.0%	88.3%
D. Housing Input	84.5%	85.0%	86.2%	86.7%	83.3%	85.3%	87.5%	89.8%	85.5%	82.9%
D. Lighting	80.9%	85.0%	83.0%	89.3%	85.4%	84.6%	82.4%	80.8%	83.3%	83.3%
D. Photovoltaics	90.8%	82.0%	82.5%	90.0%	85.9%	80.0%	87.3%	89.3%	84.9%	86.6%
D. Plumbing	82.5%	87.6%	87.8%	80.5%	83.5%	89.9%	88.8%	88.6%	87.0%	85.5%
D. SoE Input	87.5%	86.5%	86.8%	83.4%	84.6%	88.5%	81.2%	88.0%	85.5%	83.8%
D. Structures	86.0%	86.8%	89.5%	87.3%	82.1%	83.8%	84.8%	85.3%	83.6%	86.5%
cp_k **Conformance Probability**										
PV Array	4.51%	0.37%	0.43%	3.59%	1.14%	0.21%	1.65%	2.90%	0.84%	1.38%
Switch	0.00%	0.01%	0.00%	0.08%	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
Outlet	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Light Fixture	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Monitor	0.01%	0.04%	0.04%	0.01%	0.01%	0.04%	0.00%	0.14%	0.01%	0.01%
Sensor	0.01%	0.04%	0.04%	0.01%	0.01%	0.04%	0.00%	0.14%	0.01%	0.01%
Service Box	0.00%	0.11%	0.06%	0.07%	0.00%	0.03%	0.01%	0.49%	0.01%	0.03%
GFCI Outlet	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
240 Outlet	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Circuit Breaker	0.00%	0.11%	0.06%	0.07%	0.00%	0.03%	0.01%	0.49%	0.01%	0.03%
$E(S_k)$ **Defect Severity Expectation**										
PV Array	3.1	5.4	5.3	3.3	4.4	6.0	4.0	3.5	4.7	4.2
Switch	11.1	9.0	10.1	6.8	8.9	9.2	10.4	11.2	9.9	9.9
Outlet	20.9	12.8	13.7	13.4	18.2	14.9	17.6	10.1	16.3	15.1
Light Fixture	21.0	17.3	19.1	13.2	16.9	17.6	19.7	21.2	18.9	18.8
Monitor	9.4	7.4	7.6	8.4	9.4	7.5	10.1	6.3	8.7	8.8
Sensor	9.4	7.4	7.6	8.4	9.4	7.5	10.1	6.3	8.7	8.8
Service Box	11.0	6.6	7.1	7.0	9.6	7.8	9.2	5.2	8.5	7.9
GFCI Outlet	20.9	12.8	13.7	13.4	18.2	14.9	17.6	10.1	16.3	15.1
240 Outlet	20.9	12.8	13.7	13.4	18.2	14.9	17.6	10.1	16.3	15.1
Circuit Breaker	11.0	6.6	7.1	7.0	9.6	7.8	9.2	5.2	8.5	7.9
s_k **Defect Severity Realization**										
PV Array	4	5	4	2	7	5	7	3	4	8
Switch	10	12	9	3	4	9	17	16	10	8
Outlet	13	19	15	19	19	20	20	12	18	12
Light Fixture	22	19	17	11	13	14	18	22	14	15
Monitor	8	9	4	8	12	8	9	5	11	12
Sensor	14	6	8	4	13	7	11	5	10	12
Service Box	10	10	13	9	13	8	7	5	4	5
GFCI Outlet	24	12	12	6	21	8	16	12	17	15
240 Outlet	21	14	15	12	12	13	20	5	22	23
Circuit Breaker	11	4	9	8	8	10	10	9	7	6
oc_l **Operations Capacity Realization (Component Failure Probability)**										
PV Array	95.9%	95.3%	95.9%	97.3%	94.3%	95.3%	94.3%	96.6%	95.9%	93.9%
Switch	93.1%	92.5%	93.5%	96.6%	95.9%	93.5%	91.5%	91.7%	93.1%	93.9%
Outlet	92.3%	91.2%	91.9%	91.2%	91.2%	91.1%	91.1%	92.5%	91.4%	92.5%
Light Fixture	90.9%	91.2%	91.5%	92.8%	92.3%	92.1%	91.4%	90.9%	92.1%	91.9%
Monitor	93.9%	93.5%	95.9%	93.9%	92.5%	93.9%	93.5%	95.3%	92.8%	92.5%
Sensor	92.1%	94.8%	93.9%	95.9%	92.3%	94.3%	92.8%	95.3%	93.1%	92.5%
Service Box	93.1%	93.1%	92.3%	93.5%	92.3%	93.9%	94.3%	95.3%	95.9%	95.3%
GFCI Outlet	90.7%	92.5%	92.5%	94.8%	91.0%	93.9%	91.7%	92.5%	91.5%	91.9%
240 Outlet	91.0%	92.1%	91.9%	92.5%	92.5%	92.3%	91.1%	95.3%	90.9%	90.8%
Circuit Breaker	92.8%	95.9%	93.5%	93.9%	93.9%	93.1%	93.1%	93.5%	94.3%	94.8%
oc_l **Operations Capacity Realization (Circuit Failure Probability)**										
Receptacle Circuit	61.8%	57.6%	60.1%	57.6%	57.6%	57.1%	57.1%	62.8%	58.1%	62.8%
Lighting Circuit	60.7%	60.1%	62.6%	72.0%	69.3%	63.7%	58.4%	57.8%	63.0%	64.1%
GFCI Receptacle Circuit	55.7%	62.8%	62.8%	72.5%	56.7%	68.4%	59.3%	62.8%	58.7%	60.1%
GFCI Lighting Circuit	56.0%	60.1%	60.7%	68.1%	59.2%	64.5%	58.7%	59.5%	59.8%	60.1%
240 Receptacle Circuit	56.7%	60.9%	60.1%	62.8%	62.8%	61.8%	57.1%	75.0%	56.4%	56.0%
Emergency Circuit	56.4%	57.6%	58.7%	64.0%	61.8%	60.9%	58.1%	56.4%	60.9%	60.1%
oc_l **Operations Capacity Realization (Panel Failure Probability)**										
Residence Panel	5.5%	17.0%	7.2%	10.6%	9.6%	6.0%	5.0%	6.9%	9.9%	13.9%
Common Areas Panel	5.5%	17.0%	7.2%	10.6%	9.6%	6.0%	5.0%	6.9%	9.9%	13.9%
Exterior Panel	49.0%	67.8%	55.7%	64.8%	54.8%	58.9%	53.3%	57.1%	59.9%	62.6%
Baths Panel	49.0%	67.8%	55.7%	64.8%	54.8%	58.9%	53.3%	57.1%	59.9%	62.6%
Garage Panel	65.4%	78.2%	69.8%	76.0%	69.5%	72.4%	68.8%	72.6%	73.9%	75.4%
Laboratory Panel	65.4%	78.2%	69.8%	76.0%	69.5%	72.4%	68.8%	72.6%	73.9%	75.4%
Emergency Panel	67.8%	78.2%	70.3%	74.8%	72.8%	71.5%	70.4%	71.4%	76.6%	77.2%
Main Panel	69.8%	82.4%	72.4%	76.4%	73.7%	73.2%	72.4%	75.2%	78.9%	80.5%
PV Service	96.1%	97.9%	96.3%	95.9%	97.3%	96.4%	96.9%	96.0%	96.9%	97.8%
Main Service NoPV	**97.8%**	**99.3%**	**98.2%**	**98.6%**	**98.4%**	**98.2%**	**98.1%**	**98.4%**	**98.8%**	**99.0%**
Main Service PV	**96.8%**	**98.4%**	**97.1%**	**97.7%**	**97.2%**	**97.2%**	**97.2%**	**97.7%**	**98.2%**	**98.2%**

Probabilistic Dependencies

This section reviews and expands upon previous discussions of probabilistic dependencies between variables. The subject is important for evaluating the benefits of PRA-VDT because risks often depend upon multiple simultaneous events. Total system failures, for example, often result only from multiple component failures that have a common cause in development conditions (and the resulting, external events). Failing to recognize the common causes of these component failures would lead to systematic miscalculation (typically, underestimation) of total system failure risk [Davoudian et al 1994.1]. The PRA-VDT method's management of probabilistic dependencies is essential to the quality of the thesis contribution.

Error! Reference source not found. (**Error! Bookmark not defined.**) provides a Decision Diagram that formally defines which probabilistic dependencies between model variables PRA-VDT captures. Probabilistic dependencies first manifest within the Development Model results, and the PRA-VDT model preserves these dependencies through later stage analyses.

Table B.1 (on page 199) provides a set of sample data for the Green Dorm project, illustrating the specific method (sampling the full joint distribution) by which PRA-VDT preserves the identified probabilistic dependencies. The figure and table correspond: In the table, the VDT sampled data (green) corresponds to the leftmost portion of **Error! Reference source not found.**; The Defect Model-sampled data corresponding to each VDT sample (blue) corresponds to the middle portion of **Error! Reference source not found.**; and the Operations Model-derived data (red; not sampled, but solved in closed form for the Green Dorm) corresponds to the rightmost portion of **Error! Reference source not found.**.

The remainder of this section describes the implications of modeled dependencies and independence assumptions that the diagram (reading left to right) and table (reading

top to bottom) illustrate. This section also illustrates how modelers can easily extend PRA-VDT to include dependencies that the basic method currently excludes.

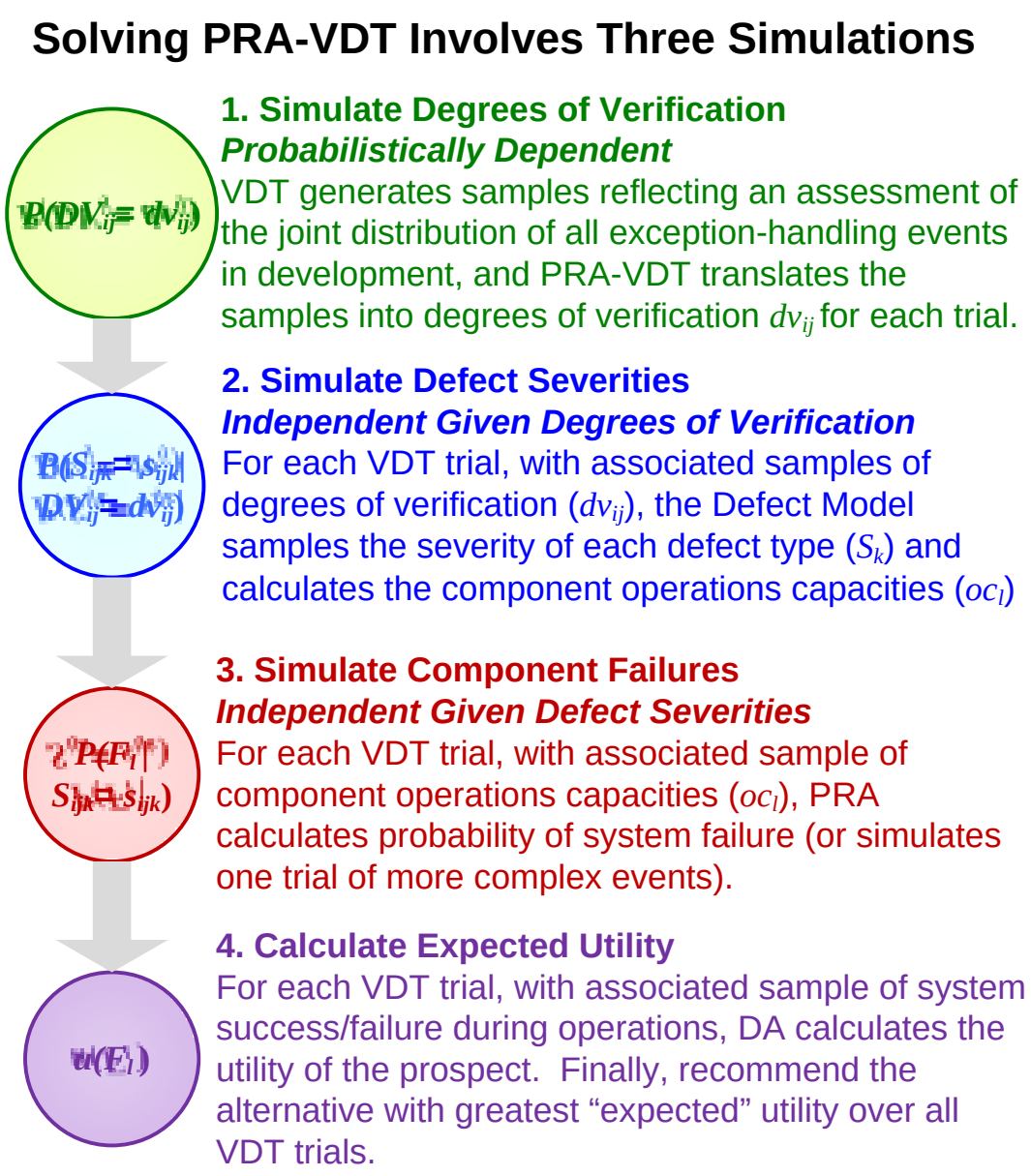


Figure B.1 PRA-VDT Solution Method's Four Steps

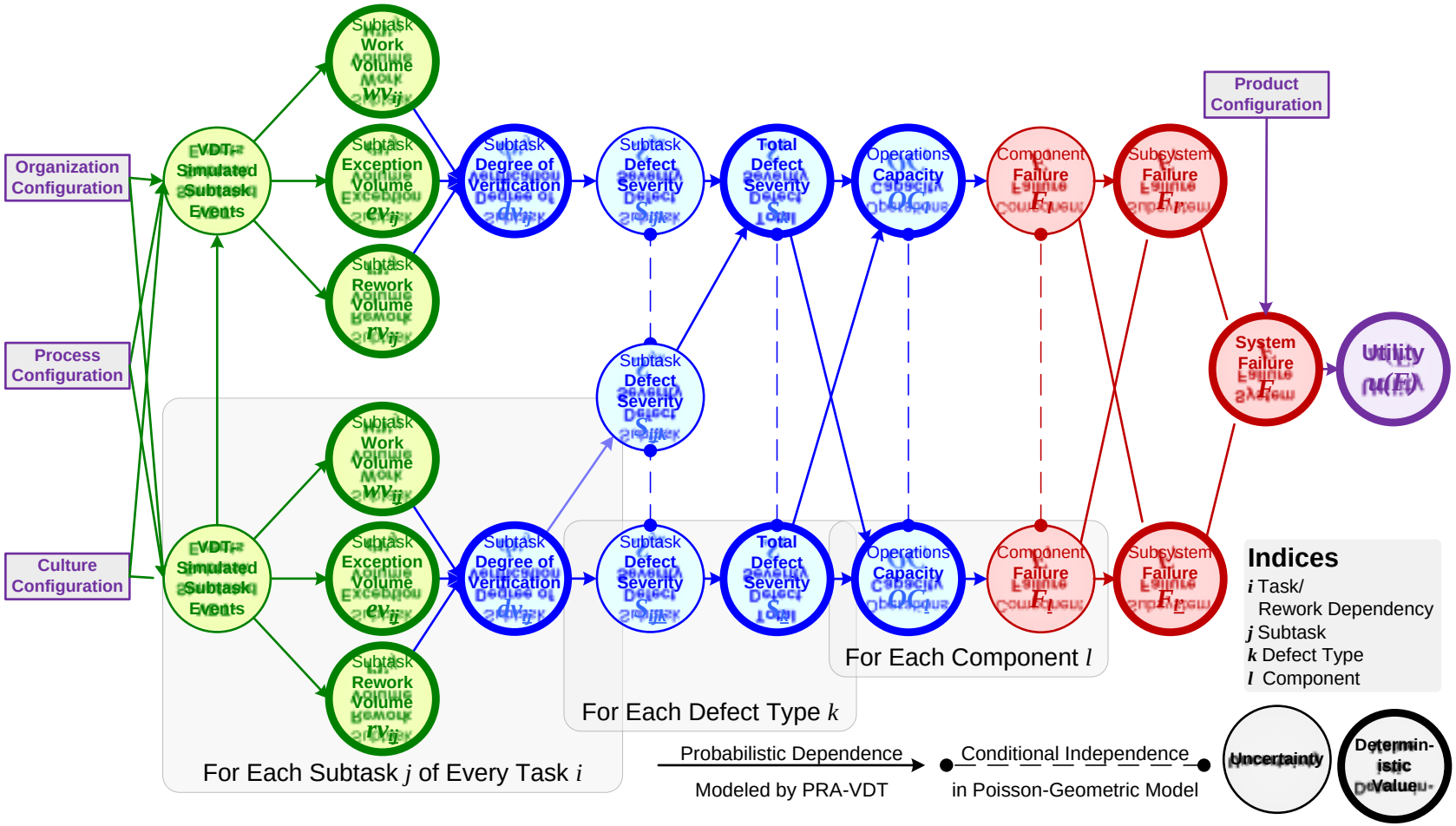


Figure B.2 Generic PRA-VDT Decision Diagram Highlighting the Modeling of Probabilistic Dependencies.

Overview of Simulations in Series

PRA-VDT has three simulations in series: VDT, the Defect Model, and PRA. A principal challenge of integrating these simulations is to account for all the dependencies among each model's many interrelated variables. PRA-VDT analysis consists of simulating the development, defect, and operations models in series until the assessed distribution of outcomes of interest (such as system failure) reaches the desired degree of precision. In this section, the index *r* identifies one trial of the VDT simulation as well as one trial each of the subsequent Defect Model and PRA simulations based on that VDT trial's output. This section introduces the method by which PRA-VDT analysis generates a set of *t* simulation trials (indexed with *r*) for the Development Model, then how a Defect Model trial for each *r* is generated, then how an Operations Model trial for each *r* is generated. Once all models are populated with data for each PRA-VDT simulation *r*, the model interprets those data as a discrete, joint probability distribution assessment of total project behavior.

Development Model Simulation

The first PRA-VDT step is the Development Model simulation, in which VDT generates *t* samples of degrees of verification DV_{ij}. These samples are termed dv_{ijr}, indexed by simulation trial *r* for each subtask *j* of every task *i*. VDT calculates these degrees of verification using a discrete event simulation with no known closed-form representation [Jin and Levitt 1996].

PRA-VDT does not require mathematically stating the joint distribution of degrees of verification across different work items because (as shown below) it suffices to continue simulating in the Defect Model, and the global model carries all scenario results forward. However, to gain insight into the intermediate results (e.g., at the end of VDT), analysts can use the following formula for the joint distribution of degrees of verification:

$$p\left(DV_{11}=dv_{11},DV_{12}=dv_{12},...\right)=\lim_{t\to\infty}\frac{1}{t}\times\left|\left\{\, r\in \mathrm{N}\mid 0<r<t,\quad dv_{11}=dv_{11r},dv_{12}=dv_{12r},...\right\}\right| \qquad \textbf{Eq. B.3}$$

Intuitively, the above formula states that the probability of a given set of degrees of verification equals the long run fraction of simulation trials that generates the given set of results.

Defect Model Simulation

The second PRA-VDT step is the Defect Model simulation, which calculates defect severities s_{kr} for each of the t simulation runs. This step individually carries forward from the Development Model Simulation every scenario indexed in r. PRA-VDT indexes these samples of S_{kr} by simulation trial r for each defect type k. Each Defect Model simulation trial uses only the degrees of verification (sampled for each subtask of every task or dependency i in the Development Model Simulation) that have the same index r:

$$\begin{aligned} &p\left(S_{1r}=s_{1r},S_{2r}=s_{2r},...\mid DV_{11r}=dv_{11r},DV_{12r}=dv_{12r},...\right)\\ &=\prod_k p\left(Poisson\left\{-\ln\prod_i\prod_j\left[cp^-{}_{ijk}+dv_{ijr}\times\left(cp^+{}_{ijk}-cp^-{}_{ijk}\right)\right]\right\}=s_{kr}\right) \end{aligned} \qquad \textbf{Eq. B.4}$$

The above formula makes clear that in this model, defect severities are independent *given the degrees of verification output from the Development Model*. PRA-VDT thus independently samples the distribution of defect severities to generate each s_{kr}:

$$\begin{aligned} &p\left(S_{kr}=s_{kr}\mid S_{1r}=s_{1r},S_{2r}=s_{2r},...;DV_{11r}=dv_{11r},DV_{12r}=dv_{12r},...\right)\\ &=p\left(S_{kr}=s_{kr}\mid DV_{11r}=dv_{11r},DV_{12r}=dv_{12r},...\right)\\ &=p\left(Poisson\left\{-\ln\prod_i\prod_j\left[cp^-{}_{ijk}+dv_{ijr}\times\left(cp^+{}_{ijk}-cp^-{}_{ijk}\right)\right]\right\}=s_{kr}\right) \end{aligned} \qquad \textbf{Eq. B.5}$$

PRA-VDT does not require stating the joint distribution of defect severities because (as shown below) it suffices to continue simulating in the Operations Model. However, to gain insight into the intermediate results, analysts can use the following formula to approximate the joint distribution of defect severities:

$$p(S_1 = s_1, S_2 = s_2, ...) = \lim_{t\to\infty} \frac{1}{t} \times \left| \{ r \in \mathrm{N} \mid 0 < r < t, \quad s_1 = s_{1r}, s_2 = s_{2r}, ... \} \right| \qquad \textbf{Eq. B.6}$$

Operations Model Simulation

The third PRA-VDT step is the Operations Model simulation, which samples component and subsystem outcomes f_{lr} for each of the t simulation runs. This step individually carries forward from the Defect Model Simulation every scenario indexed in r. PRA-VDT indexes these values of F_{lr} by simulation trial r for each component l. Each Operations Model calculation (adapted below from Eq. 3.18) uses only the defect severities (sampled in the Defect Model simulation) that have the same index r:

$$p(F_{lr} = success) = oc_{lr} = oc_l^- + (oc_l^+ - oc_l^-) \times \prod_k (di_{kl})^{s_{kr}} \qquad \textbf{Eq. B.7}$$

In calculating the failure probabilities of complex subsystems having many components with dependent failure rates, PRA-VDT uses the following formula to approximate the joint distribution of component failures:

$$p(F_1 = f_1, F_2 = f_2, ...) = \lim_{t\to\infty} \frac{1}{t} \times \left| \{ r \in \mathrm{N} \mid 0 < r < t, \quad f_1 = f_{1r}, f_2 = f_{2r}, ... \} \right| \qquad \textbf{Eq. B.8}$$

In summary, PRA-VDT analysis consists of simulating the development, defect, and operations models in series until the assessed distribution of outcomes of interest (such as system failure) reaches any desired degree of precision.

Discussion of Dependencies

Distributions of Degrees of Verification

Dependence Modeled in PRA-VDT

VDT simulates emergent events (such as the accumulation of agent backlog) that affect *different work items within the same task* persistently over time. The affects of these uncertain factors manifests as probabilistic dependencies between the degrees of

verification of subtasks *within each task*. The random variables DV_{ij} and DV_{ij+1}, for example, typically are probabilistically dependent.

VDT also simulates interactions between *work items on different tasks* during development (such as corresponding communications completion rates). The affects of these uncertain factors manifests as probabilistic dependencies between the degrees of verification of subtasks *in different tasks*. The random variables DV_{ij} and DV_{i+1j}, for example, typically are probabilistically dependent.

By simulating the implications of every VDT simulation trial separately, PRA-VDT preserves dependencies between all Degrees of Verification based on VDT assessments of development organization behavior. This feature ensures that PRA-VDT captures the subtle interactions between organizational behaviors, as well as the behaviors that manifest infrequently (within the range assessed by VDT).

Mathematically, PRA-VDT assesses all subsequent variables (defect severities, etc) separately for every VDT trial and set of corresponding degree of verification realizations dv_{ij}, as shown in the below form of Eq. 3.1:

$$dv_{ijr} = \frac{wv_{ijr} - \left(ev_{ijr} - rv_{ijr}\right)}{wv_{ijr}} \qquad \textbf{Eq. B.9}$$

Distributions of Defect Severity

The thesis models the distributions of defect severities using VDT-assessed degrees of verification for tasks and dependencies (dv_{ij}) and using conformance probability limits (cp_{ijk}^{-} and cp_{ijk}^{+}). Because this equation for each defect type *k* includes the degrees of verifications for each *i*, the model conserves the VDT-assessed *dependency between different tasks' performance*. The equation also maintains the dependencies for each subtask *j*, thereby conserving the VDT-assessed *dependency over time of performance within individual tasks* in a simulation trial.

Dependence Modeled in PRA-VDT

In assessing defect severity distributions, the implication of PRA-VDT conditioning all subsequent calculations on degrees of verification is that the complex interactions between events that VDT simulates in development shape a corresponding joint distribution of defect severities.

At the level of subtasks, conformance probabilities cp_{ijk} are probabilistically dependent because they are calculated deterministically (by Eq. 3.2 on page 82) using probabilistically dependent VDT-assessed degrees of verification. The marginal distributions of defect severities generated by different tasks and subtasks, S_{ijk}, are also probabilistically dependent because they are directly related (in the following substitute for Eq. 3.9 on page 88) to the conformance probabilities:

$$cp_{ijkr} = cp^{-}{}_{ijk} + dv_{ijr} \times \left(cp^{+}{}_{ijk} - cp^{-}{}_{ijk} \right) \qquad \textbf{Eq. B.10}$$

At the level of tasks, the PRA-VDT framework samples the S_k probability distributions in order to preserve the *joint distribution's* probabilistic dependencies resulting from the emergent degrees of development verification (simulated by VDT).

$$S_{kr} \sim Poisson\left\{ -\ln \prod_i \prod_j \left[cp_{ijkr} = cp^{-}{}_{ijk} + dv_{ijr} \times \left(cp^{+}{}_{ijk} - cp^{-}{}_{ijk} \right) \right] \right\} \qquad \textbf{Eq. B.11}$$

Independence Modeled in PRA-VDT

This thesis describes several distributions that modelers can use under different circumstances, and drills down on the Poisson distribution because it is highly flexible and frequently appropriate. The Poisson analysis, however, assumes that *given the VDT output*, the existence of one engineering defect is not relevant to the existence of any other potential defects. In terms of the modeled development processes, this model assumes that, for a set of VDT-assessed development impacts corresponding to a single simulation trial, defects are *memoryless* in that they result from work items according to conditionally independent distributions. Computationally, PRA-VDT

samples defect severities for each subtask independently based on degrees of verification generated for a single VDT trial.

In assessing defect severity distributions, one implication of the Poisson case in PRA-VDT not conditioning on any additional data is that physically complementary or contradictory defect types are not modeled. These cases warrant extending the method according to the needs of a specific project, or using distributions other than the Poisson. For example, PRA-VDT has no explicit mechanism for modeling a single object with mutually exclusive defects "Too heavy" and "Too light.". If the affects those defects could have on operations are the same, modelers may be able to condense them into a single, more likely defect "Wrong weight."

Distributions of Operations Capacity

Dependence Modeled in PRA-VDT

The PRA-VDT framework calculates values for oc_l deterministically for each set of defect severities (which corresponds to one VDT trial of development outcomes). This method preserves the operations capacities (OC_l) joint distribution's probabilistic dependencies resulting from the emergent degrees of development verification (sampled by VDT) and defect severities s_k (sampled by the Defect Model).

$$oc_{lr} = oc_l^- + \left(oc_l^+ - oc_l^-\right) \times \prod_k \left(di_{kl}\right)^{s_{kr}} \qquad \textbf{Eq. B.12}$$

Although the simulation method is based on joint, rather than marginal distributions, it illustrates the model structure nevertheless to state the marginal distributions of operations capacities by substituting the above substitute for Eq. 3.16 into the below s Eq. 3.18:

$$OC_{lr} \sim oc_l{}^- + \left(oc_l{}^+ - oc_l{}^-\right) \times \prod_k di_{kl}{}^{Poisson\left\{-\ln \prod_i \prod_j \left[cp^-{}_{ijk} + dv_{ijr} \times \left(cp^+{}_{ijk} - cp^-{}_{ijk}\right)\right]\right\}} \qquad \textbf{Eq. B.13}$$

The above equation shows that the Defect Model assesses the capacity as a function of input capacity limits (oc_l^- and oc_l^+), VDT-assessed degrees of task and dependency verification (dv_{ij}), input conformance probability limits (cp_{ijk}^- and cp_{ijk}^+), and input defect influences (di_{kl}).

Independence Modeled in PRA-VDT

This thesis drills down on the case of Geometric degradation of operations capacity as a function of defect severity because it is highly flexible and frequently appropriate. For example: setting $di_{kl} = 0$ provides a step function, which minimizes capacity in response to the smallest defect; setting $di_{kl} = 0.5$ provides a pronounced curve, which cuts marginal capacity in half for each level of defectiveness; and setting $di_{kl} = 0.99$ provides a nearly linear function that degrades significantly only under severe defects.

The Geometric analysis, however, assumes that *given the VDT output*, and *within the range of defect dependency*, the marginal effect of each level of defect severity is not relevant to the marginal effect of other defect severities. This assumption is analogous to arranging in series a set of components with conditionally independent failure probabilities, where the number of components is a (linear) function of the defect severity.

In assessing defect severity distributions, because the Poisson case in PRA-VDT does not condition on any additional data, physically complementary or contradictory levels of defect severity are not modeled. These cases warrant extending the method according to the needs of a specific project, or using distributions other than the Geometric. For example, a distribution other than the Geometric is required to model a single defect type that affects capacity only at odd severity levels (such as a defect "reflected" that is insignificant if it occurs twice). Modelers could overcome that challenge by calculating a new "effective severity" $s_{k'} = \lfloor s_k / 2 \rfloor$ and setting $di_{k'l} = 0$.

Distributions of Operations Behavior

Dependence Modeled in PRA-VDT

The PRA-VDT framework simulates these equations in order to preserve the full *joint distributions* without removing probabilistic dependencies. Specifically, each trial of Eq. 3.16 (on page 91) uses a single sample from the Poisson distribution to assess the severity (number) of defects S_k of each type k, and the probability of failure during operations, so the calculation preserves dependencies between operations that can fail due to root causes in the same organizational behavior or in the same defects.

The mathematical formulae for *marginal distributions* nevertheless reveal how the method preserves those dependencies. Substituting the marginal distribution of engineering defects S_k from Eq. 3.17 provides the following formula, which expresses the probability of failure in terms of degrees of verification dv_{ij} assessed by VDT in the below substitute for the Development Model's Eq. 3.20:

$$OC_{lr} \sim oc_l^{-} + \left(oc_l^{+} - oc_l^{-}\right) \times \prod_k di_{kl}^{\,Poisson\left\{-\ln \prod_i \prod_j \left[cp^{-}{}_{ijk} + dv_{ijr} \times \left(cp^{+}{}_{ijk} - cp^{-}{}_{ijk}\right)\right]\right\}} \qquad \textbf{Eq. B.14}$$

Eq. 3.20 explains how the PRA-VDT framework assesses the probability of failure is a function of VDT-assessed degrees of task and dependency verification (dv_{ij}), conformance probability limits (cp_{ijk}^{-} and cp_{ijk}^{+}), defect severities (ds_{kn}), operations capacity limits (oc_n^{-} and oc_n^{+}), and operations load (ob_n). Because (for each defect type k) Eq. 3.20 includes the degrees of verifications for each i, the model conserves the VDT-assessed dependency between different tasks' performance. Because simulating the equation also maintains the dependencies for each subtask j, the model conserves the VDT-assessed dependency over time of performance for individual tasks in a simulation run.

Independence Modeled in PRA-VDT

The implication of PRA-VDT not *directly* conditioning the distributions of operations failures among components is that PRA-VDT provides no knowledge of structure

between operating context factors. For example, PRA-VDT would require enhancement to model two components that never operate together, or always operate together, due to operations processes rather than physical or organizational constraints. PRA-VDT would also require enhancement to model a common load placed on multiple components. Modelers can easily overcome this omission by explicitly defining the operations context probabilistically (using an influence diagram, for example), sampling one trial of operations context for each PRA-VDT simulation (which corresponds to one VDT trial), and finally calculating the failure probabilities for each component given both the operations capacities and the operations context loads.

List of References

Asch, S. E. (1987, original work published 1952). *Social Psychology*. New York: Oxford University Press.

Barlow, r., and H. Lambert. (1975) "Introduction to Faiult Tree Analysis", *Reliability and Fault Tree Analysis* (SIAM) United States Nuclear Regulatory Commission, pp.7-35.

Bea, R. (2003). "Quality, Reliability and Human Factors in Deepwater Drilling & Production" Conference Keynote Paper, 21st International Conference on Offshore Mechanics & Arctic Engineering, American Society of Mechanical Engineers, Cancun, Mexico.

Bem, D., M. Wallach, and N. Kogan (1965). "Group Decision Under Risk of Aversive Consequences" *Journal of Personality and Social Psychology*, 1(5), 453-460.

Benjamin, J. and M. E. Paté-Cornell (2004). "Risk Chair for Concurrent Design Engineering: Satellite Swarm Illustration" *Journal of Spacecraft and* Rockets Vol. 41 No. 1 January-February 2004.

Bergner, D. (2005). Personal communication by an employee of the NASA Ames Research Center's Human and Organizational Risk Management research program.

Box, G., and Draper, N. (1987) *Empirical Model0Building and Response Surfaces*, p. 424, Wiley.

Brooks, F. (1995). *The Mythical Man-Month* (20th Anniversary Edition), Addison-Wesley.

Burton, R., and Obel, B. (1995) "The Validity of Simulation Models in Organizational Science: From Model Realism to Purpose" *Computational & Mathematical Organization Theory*, Vol. 1, No. 1, Springer Netherlands, , available as Center for Integrated Facility Engineering Working Paper WP030, Stanford University, Palo Alto, CA.

Burton, R. and Obel, B. (2003) *Strategic Organizational Diagnosis and Design: The Dynamics of Fit* Third Edition. Kluwer Academic Publishers.

Burton, R. (2001) "Afterword" in A. Lomi and E. Larsen (eds.) *Dynamics of Organizations: Computational Modeling and Organization Theories*, the MIT Press, Cambridge, MA.

Carley, K. (1996). "Validating Computational Models" Working paper prepared at Carnegie Mellon University for the Office of Naval Research, available as http://www.econ.iastate.edu/tesfatsi/EmpValid.Carley.pdf.

Chachere, J. (2004.1) "Methods and Benefits of Integrating The Virtual Design Team with Probabilistic Risk Analysis for Design Project and Program Planning" Working Paper developed during a Management Science and Engineering Ph.D. Tutorial for Elisabeth Paté-Cornell, available as Center for Integrated Facility Engineering Working Paper WP094, Stanford University, Palo Alto, CA.

Chachere, J. (2004.2) "Design Project Optimization" Working Paper developed during a Management Science and Engineering Ph.D. Tutorial for Elisabeth Paté-Cornell, available as Center for Integrated Facility Engineering Working Paper WP095, Stanford University, Palo Alto, CA.

Chachere, J. (2005) "Improving Project Performance by Predicting Engineering Impacts on Operations Failures using a Quantitative Model of Product, Organization, Process, and Environment" Center for Integrated Facility Engineering Working Paper WP096, Stanford University, Palo Alto, CA.

Chachere, J., J. Kunz, and R. Levitt (2004.1). "Can You Accelerate Your Project Using Extreme Collaboration? A Model Based Analysis" 2004 International Symposium on Collaborative Technologies and Systems; Also available as Center for Integrated Facility Engineering Technical Report T152, Stanford University, Palo Alto, CA.

Chachere, J., J. Kunz, and R. Levitt (2004.2). "Observation, Theory, and Simulation of Integrated Concurrent Engineering: 1. Grounded Theoretical Factors that Enable Radical Project Acceleration" available as Center for Integrated Facility Engineering Working Paper WP087, Stanford University, Palo Alto, CA.

Chachere, J., J. Kunz, and R. Levitt (2004.3). "Observation, Theory, and Simulation of Integrated Concurrent Engineering: 2. Risk Analysis Using Formal Models of Radical Project Acceleration" available as Center for Integrated Facility Engineering Working Paper WP088, Stanford University, Palo Alto, CA.

Chachere, J., Kunz, J., and Levitt, R. (2004.4). "Observation, Theory, and Simulation of Integrated Concurrent Engineering: Grounded Theoretical Factors and Risk Analysis Using Formal Models" forthcoming in Project Risk Management Principles and Practices, Institute of Chartered Financial Analysts of India, Banjara Hills, India.

Christiansen, T.R. (1993) *Modeling Efficiency and Effectiveness of Coordination in Engineering Design Teams* Ph.D. Dissertation, Department of Civil and Environmental Engineering, Stanford University.

Ciavarelli, A. (2003). "Organizational Risk Assessment: The Role of Safety Culture" Unpublished manuscript prepared at the Naval Postgraduate School for NASA-Ames Research Center.

Clevenger, C., Fontana, A., and Rabaron, B. (2006). "Stanford Green Dorm Design Project Organizational Risk Assessment" Unpublished manuscript prepared as a final term project for Stanford University course CEE 242, Organizational Design for Projects and Companies.

Cohen, G. P. (1992), *The Virtual Design Team: An Object-Oriented Model of Information Sharing in Project Teams.* Ph.D. Dissertation, Department of Civil Engineering, Stanford University.

Cohen, M., J. March, and J. Olsen (1972) "A Garbage Can Model of Organizational Choice" *Administrative Science Quarterly*, Vol. 17, No. 1, pp. 1-25.

Cooke, N., J. Gorman, and H. Pedersen (2002). "Toward a Model of Organizational Risk: Critical Factors at the Team Level" Unpublished manuscript prepared at New Mexico State University and Arizona State University.

Cornell, A., and N. Newmark (1978) "On the Seismic Reliability of Nuclear Power Plants" *Invited Paper, ANS Topical Meeting on Probabilistic Reactor Safety*, Newport Beach, CA.

Cyert, R., E. Feigenbaum, and J. March (1959) "Models in a Behavioral Theory of the Firm" *Behavioral Science*, Vol. 4, No. 2.

Daly, A. (2006). Personal communication intended as expert testimony as Principal in a building systems design firm.

Davoudian, K., J. Wu, and G. Apostolakis (1994.1) "Incorporating Organizational Factors into Risk Assessment Through the Analysis of Work Processes" *Reliability Engineering and System Safety*, Vol. 45, pp. 85-105.

Davoudian, K., J. Wu, and G. Apostolakis (1994.2) "The Work Process Analysis Model (WPAM)" *Reliability Engineering and System Safety*, Vol. 45, pp. 107-125.

Dillon, R. L., and M. E. Paté-Cornell (2001). "APRAM: an advanced programmatic risk analysis method," *International Journal of Technology, Policy, and Management*, Vol. 1, No. 1, pp.47-65.

Dillon, R. L., M. E. Paté-Cornell, and S. D. Guikema (2003) "Programmatic Risk Analysis for Critical Engineering Systems Under Tight Resource Constraints," *Operations Research*, May/June.

Dyer, J.S. (1990). "Remarks on the Analytic Hierarchy Process" *Management Science* Vol. 36, Issue 3.

EHDD (2006) *Stanford Green Dorm Feasibility Report: a Living Laboratory Concept Proposal by the Civil and Environmental Engineering Department with EHDD Architecture*, available as http://www.stanford.edu/group/greendorm/greendorm/feasibility_study.html.

Fayol, H. (1949/1916) *General and Industrial Management* Pitman (French original published in 1916).

Feigenbaum, E. (1988). *The Rise of the Expert Company*, Times Books, New York and Macmillan, London.

Festinger, L. (1954) "A Theory of Social Comparison Processes," *Human Relations* Vol. 7, pp. 117-140.

Fishburn, P. (1964). *Decision and Value Theory*, Wiley, New York.

Gaba, D., S. Singer, A. Sinaiko, J. Bowen, A. Ciavarelli (2003) "Differences in Safety Climate between Hospital Personnel and Naval Aviators" *Human Factors*, Vol. 45.

Gaba, D., S. Singer, and A. Rosen "Safety Culture: Is the 'Unit' the Right 'Unit of Analysis'?" *Critical Care Medicine*, Vol. 35 No. 1.

Garrick, B., and Christie, R. (2002) "Probabilistic Risk Assessment Practices in the USA for Nuclear Power Plants," *Safety Science*, Vol. 40, No. 1.

Galbraith, J. (1977). *Organization Design*. Reading, MA: Addison-Wesley.

Ghosh, S.T., and G. Apostolakis (2005) "Organizational Contributions to Nuclear Power Plant Safety" *Nuclear Engineering and Technology*, Vol. 32, No. 3.

Gibbons, R. (1992). *Game Theory for Applied Economists*. Princeton, NJ: Princeton University Press.

Graham, J., and Vaupel, J. (1981). "Value of a Life: What Difference Does It Make?" *Risk Analysis*, Vol. 1, No. 1.

GSB (2006) *Stanford University Graduate School of Business Environmental Sustainability Task Force Final Report*, December 1, 2006.

Hauser, J., and D. Clausing (1988) “The House of Quality” *Harvard Business Review*, May-June.

Haymaker, J., and J. Chachere. (2006). “Coordinating Goals, Preferences, Options, and Analyses for the Stanford Living Laboratory Feasibility Study” *Lecture Notes in Computer Science*, No. 4200, pp. 320-327, Springer-Verlag, Berlin, Germany.

Holland, J. (1975). *Evolution in Natural and Artificial Systems* University of Michigan Press, Ann Arbor, Michigan.

Horii, T. (2005) *Impact of Multiple Normative Systems on the Organizational Performance of International Joint Ventures*, Ph.D. Dissertation, Department of Civil and Environmental Engineering, Stanford University.

Howard, R. (1988). "From Influence to Relevance to Knowledge," *Proceedings of the Conference on Influence Diagrams for Decision Analysis, Inference, and Prediction*, Berkeley, CA.

Howard, R. (1989.1). "Knowledge Maps," *Management Science*, Vol. 35 No. 8, pp. 903-922.

Howard, R. (1989.2). "Microrisks for Medical Decision Analysis," *International Journal of Technology Assessment in Health Care*, Vol. 5 No. 3, pp. 357-370.

Howard, R. (1991). "In Praise of the Old-Time Religion", in *Utility: Theories, Measurement, and Application*, Kluwer publishers.

Howard, R. (1992). "Heathens, Heretics and Cults: The Religious Spectrum of Decision Aiding", *Interfaces*, 22, pp. 15-27.

Howard, R. and J. Matheson (eds.) (1983). *Readings on the Principles and Applications of Decision Analysis*, Decision Analysis, Strategic Decisions Group, Menlo Park, CA.

IAEA (1991) *Safety Culture: A Report by the International Nuclear Safety Advisory Group*, International Atomic Energy Agency, Vienna, Austria.

Janis, I. (1982.1). *Stress Attitudes, and Decisions: Selected Papers* New York, Praeger Publishers.

Janis, I. (1982.2). *Groupthink: Psychological Studies of Policy Decisions and Fiascoes*, Houghton Mifflin Company, 01 June.

JPL (2004). Personal communication by an anonymous employee of the Jet Propulsion Laboratory’s Advance Studies (mission design) program.

JPL Special Review Board (2000) *Report on the Loss of the Mars Polar Lander and Deep Space 2 Missions*, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA.

Jin, Y., R. Levitt, T. Christiansen, and J. Kunz. (1995). "The Virtual Design Team: Modeling Organizational Behavior of Concurrent Design Teams," *International Journal of Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol.9, No.2, (April) 145-158.

Jin, Y. and R. Levitt (1996). "The Virtual Design Team: A Computational Model of Project Organizations" *Computational and Mathematical Organization Theory*, 2(3): 171- 196.

Kahneman, D., and Tversky, A. (2000) *Choices, Values, and Frames*, Cambridge University Press, Cambridge, Mass.

Keeney R., and H. Raiffa. (1976). *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, New York: John Wiley and Sons, Inc.

KHosraviani, B. (2005) "An Evolutionary Approach for Project Organization Design: Producing Human-Competitive Results using Genetic Programming," Doctoral Thesis, Department of Civil and Environmental Engineering, Stanford University, Palo Alto, CA.

KHosraviani, B., R. Levitt, and J. Koza (2004) "Organization Design Optimization Using Genetic Programming" in Maarten Keijzer (ed.) *Late Breaking Papers at the 2004 Genetic and Evolutionary Computation Conference*, Seattle, Washington.

Koza, J., (1992). On the Programming of Computers by the Means of Natural Selection MIT Press, Boston, MA.

Kranz, G. (2000). Failure is not an Option: Mission Control from Mercury to Apollo 13 and Beyond New York: Simon and Schuster.

Kunz, J., and R. Levitt (2002) "Design Your Project Organization as Engineers Design Bridges" available as Center for Integrated Facility Engineering Working Paper WP073, Stanford University, Palo Alto, CA.

Kunz, J., T. Christiansen, G. Cohen, Y. Jin, and R. Levitt (1998). "The Virtual Design Team: A Computational Simulation Model of Project Organizations," *Communications of the Association for Computing Machinery*, (November) pp.84-92.

Law and Kelton (2000). *Simulation Modeling and Analysis* 3rd Edition, New York: McGraw-Hill.

Lave, C. and J. March (1975). *An Introduction to Models in the Social Sciences*. New York: Harpers and Row.

Leveson, N. (2004) "A New Accident Model for Engineering Safer Systems," *Safety Science*, Vol. 42, No. 4 pp. 237-270.

Levitt, R., J. Thomsen, T. Christiansen, J. Kunz, Y. Jin, and C. Nass (1999). "Simulating Project Work Processes and Organizations: Toward a Micro-Contingency Theory of Organizational Design," *Management Science 45 (11),* November, pp. 1479-1495.

Luce, R. D. and H. Raiffa (1957). *Games and Decisions: Introduction and Critical Survey*. New York: Wiley. (Reprinted New York: Dover, 1989).

Luce, R. and H. Raiffa (1990). "Utility Theory" In Moser, P.K. (Ed.) *Rationality in Action: Contemporary Approaches* (pp. 19-40) Cambridge University Press: New York, NY.

Marais, K., N. Dulac, and N. Leveson (2004). "Beyond Normal Accidents and High Reliability Organizations: The Need for an Alternative Approach to Safety in Complex Systems" Working paper prepared at the Massachusetts Institute of Technology.

March, J. (1994). *A Primer on Decision Making: How Decisions Happen*. New York: Free Press.

March, J. (2001) "Foreword" in A. Lomi and E. Larsen (eds.) *Dynamics of Organizations: Computational Modeling and Organization Theories*, the MIT Press, Cambridge, MA.

March, J. and Olsen, J. eds. (1985) *Ambiguity and Choice in Organizations* Scandinavian University Press, Oslo, Norway.

March, J. and Simon, H. (1958). *Organizations*. New York, John Wiley & Sons, Inc.

Matheson, J. and Howard, R. (1968). "An Introduction to Decision Analysis" in Howard, R. and J. Matheson (eds.) (1983). *Readings on the Principles and Applications of Decision Analysis*, Decision Analysis, Strategic Decisions Group, Menlo Park, CA.

Maule, A. and A. Edland (1997). "The Effects of Time Pressure on Human Judgment and Decision Making" in R. Ranyard, W. Crozier, and I. Svenson (Eds.) *Decision Making: Cognitive Models and Explanations* (pp. 189-204) New York: Routledge.

Meshkat, L., and R. Oberto (2004). "Towards a Systems Approach to Risk Considerations for Concurrent Design" Working paper prepared at the Jet Propulsion Laboratory, California Institute of Technology.

Milgrom, P, and J. Roberts (1992). *Economics, Organization & Management* Englewood Cliffs, N.J.: Prentice Hall.

Moder, J. and C. Phillips (1983). Project Management with CPM, PERT and Precedence Programming 2nd Edition.

Murphy, D. and M. E. Paté-Cornell (1996). "The SAM Framework: A Systems Analysis Approach to Modeling the Effects of Management on Human Behavior in Risk Analysis", *Risk Analysis*, Vol. 16, No. 4, pp.501-515.

NASA (1995). *NASA Systems Engineering Handbook*, National Aeronautics and Space Administration, Washington, D.C.

NASA (2003). *Columbia Accident Investigation Board Report Volume 1* Government Printing Office, Washington, D.C.

Nasrallah, W. (2006) "When Does Management Matter in a a Dog-Eat-Dog World: an 'Interaction Value Analysis' Model of Organizational Climate" *Computational & Mathematical Organization Theory* Col. 12, No. 4 pp. 339-359.

Nasrallah, W., Levitt, R., and Glynn, P. (2003) "Interaction Value Analysis: When Structured Communication Benefits Organizations" *Organization Science* Vol. 14, No. 5, pp. 541-557.

Oralkan, G. (1996) *Explorer: A Computational Model of Organizational Learning in Response to Changes in Envieonrent & Technology*. Ph.D. Dissertation, Department of Civil Engineering, Stanford Unviersity.

Paté-Cornell, M.E. (1983). "Acceptable Decision Processes and Acceptable Risks in Public Sector Regulations," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 13, No. 3.

Paté-Cornell, M.E. (1984.1). "Fault Trees vs. Event Trees in Reliability Analysis," *Risk Analysis*, Vol. 4, No. 3 pp. 177-186.

Paté-Cornell, M.E. (1984.2). "Discounting in Risk Analysis: Capital versus Human Safety," *Proceedings of the Symposium on Structural Technology and Risk*, Griogoriu, M. (ed.), July 17-20, University of Waterloo Press, Waterloo, ON.

Paté-Cornell, M.E. (1990). "Organizational Aspects of Engineering System Safety: The Case of Offshore Platforms," *Science*, Vol. 250, November 1990, pp. 1210-1217.

Paté-Cornell, M.E. (1993). "Learning from the Piper Alpha Accident: A Postmortem Analysis of Technical and Organizational Factors" *Risk Analysis*, Vol. 13, No. 2, pp. 215-234-1217.

Paté-Cornell, M.E. (2004). "Engineering Risk Analysis: Volume 1" MS&E 250A Course Reader, Stanford University, Stanford, CA.

Paté-Cornell, M.E., and P. Fischbeck (1993.1). "Probabilistic risk analysis and risk-based priority scale for the tiles of the space shuttle" *Reliability Engineering and System Safety*, Vol. 40, pp. 221-238.

Paté-Cornell, M.E., and P. Fischbeck (1993.2). "PRA as a management tool: organizational factors and risk-based priorities for the maintenance of the tiles of the space shuttle orbiter" *Reliability Engineering and System Safety*, Vol. 40, pp. 239-257.

Paté-Cornell, M.E., D. Murphy, L. Lakats, and D. Gaba (1996). "Patient risk in anaesthesia: Probabilistic risk analysis and management improvements" *Annals of Operations Research*, Vol. 67, pp. 211-233.

Perrow, C. (1984, 1999). *Normal Accidents: Living with High-Risk Technologies* New York: Basic Books.

Perrow, C. (1994). "The Limits of Safety: The Enhancement of a Theory of Accidents" *Journal of Contingencies and Crisis Management*, 2, (4), 212-220.

Perrow, C. (2000). "An Organizational Analysis of Organizational Theory," *Contemporary Society*, Vol. 29.

Powell, W. and P. DiMaggio (Eds.) (1991). *The New Institutionalism in Organizational Analysis* Chicago: University of Chicago Press.

Pugnetti, C. (1997) *Scheduling Design Processes with Interdependent Tasks: a Systems Analysis Approach* Ph.D. Dissertation, Department of Industrial Engineering and Engineering Management, Stanford University.

Ramsey, F. (1931). "Truth and Probability" in *Foundations of Mathematics and Other Logical Essays*, Routledge and Keegan, Paul, London.

Reason, J. (1997) *Managing the Risks of Organizational Accidents* Ashgate, Aldershot, England.

Roberts, K. (1989) "New Challengers in Organizational Research: High Reliability Organizations" *Organization & Environment*, Vol. 3, No. 2, pp. 111-125.

Roberts, K. (1990). "Managing High-Reliability Organizations" *California Management Review*, 32, (4), 101-113.

Roberts, K., S. Stout, and J. Halpern (1994) "Decision Dynamics in Two High reliability Military Organizations" *Management Science* Col. 40 No. 5 pp.614-624.

Rogers Commission (1986). *Report of the Presidential Commission on the Space Shuttle* Challenger *Accident*, Washington, D.C.

Saaty, T. L. (1990), "How to make a decision: The Analytic Hierarchy Process", *European Journal of Operational Research*, Vol. 48, No 1, pp. 9-26.

Sagan, S. (2004) "Learning from Normal Accidents" *Organization & Environment*, Vol. 17, No. 1, 15-19.

Salazar-Kish, J.M. (2001) *Modeling Concurrency Tradeoffs and their Effects on Project Duration and Rework* Ph.D. Dissertation, Department of Civil and Environmental Engineering, Stanford University.

Savage, L. (1954). *The Foundations of Statistics*, Wiley, New York.

Scott, W. R. (1998). *Organizations: Rational, Natural, and Open Systems* 4th Edition New Jersey: Prentice-Hall.

Selby, R.G., F. Vecchio, and M. Collins (1997). "The Failure of an Offshore Platform" *Concrete International*, August, Vol. 19, No. 8.

Sherif, M., O. J. Harvey, B. J. White, W. Hood, and C. Sherif (1961). *Intergroup conflict and cooperation: the robbers cave experiment* Norman, OK University Book Exchange.

Simon, H. (1977). *The New Science of Management Decision* 3rd revised edition (1st edition 1960) Prentice-Hall, Englewood Cliffs, NJ.

Singer, S., D. Gaba, J. Geppert, A. Sinaiko, S. Howard, and K. Park (2003) "The Culture of Safety: Results of an Organization-Wide Survey in 15 California Hospitals," *Quality and Safety in Health Care* Col 12 pp. 112-118.

Sosa, M., S. Eppinger, and C. Rowles (2004) "The Misalignment of Product Architecture and Organizational Structure in Complex Product Development," *Management Science* Vol. 50, No. 12 1674-1689.

Spetzler, C., and C. von Holstein (1972) "Probability Encoding in Decision Analysis," presented at the ORSA-TIMS-AIEE 1972 Joint National Meeting, Atlantic City, New Jersey.

Thompson, J. (1967). *Organizations in Action: Social Science Bases in Administrative Theory*, McGraw-Hill, New York.

Thomsen, J. (1998) *Virtual Team Alliance (VDA): Modeling the Effects of Goal Incongruency in Semi-Routine, Fast-paced Project Organizations*, Ph.D. Dissertation, Department of Civil and Environmental Engineering, Stanford University.

Thomsen, J., R. Levitt, J. Kunz, C. Nass, and D. Fridsma (1999) "A Trajectory for Validating Computational Emulation Models of Organizations" *Journal of Computational & Mathematical Organization Theory* Vol. 5, No. 4, pp. 385-401.

Triubs, M. (1973). "Decision Analysis Approach to Satisfying the Requirements of the Flammable Fabrics Act" *Standardization News*, Vol. 1, No. 2.

Tversky, A. and D. Kahneman (1974). "Judgment Under Uncertainty: Heuristics & Biases" in D. Kahneman, P. Slovic, & A. Tversky (Eds.) *Judgment Under Uncertainty: Heuristics and Biases*. Cambridge: Cambridge University Press.

USCG (1982). *Marine Casualty Report Mobile Offshore Drilling Unit (MODU) Ocean Ranger*, Report No. USCG 16732/0001 HQS 82, United States Coast Guard Marine Board of Investigation, Washington, D.C.

USDOD (1980). *Military Standard Procedures for Performing a Failure Mode, Effects and Criticality Analysis*, MIL-STD-1629A, United States Department of Defense, Washington, D.C.

USNRC (1980) *The Fault Tree Handbook*, NUREG/0492, United States Nuclear Regulatory Commission Washington D.C.

Vaughan, D. (1996). *The Challenger Launch Decision* The University of Chicago Press, Chicago, IL.

von Neumann, J., and O. Morgenstern (1944). *Theory of Games and Economic Behavior*, Princeton University Press, Princeton, NJ.

Walker, M.R., and Sawyer, J. (1959), "Project Planning and Scheduling," Report 6959, E.I. DuPont de Nemours and Co., Wilmington, Delaware.

Weick, K. (1987) "Organizational Culture as a source of High Reliability," *California Management Review*, Vo. 29, pp. 112-127.

Weick, K. (1993) "Collective Mind in Organizations: Heedful Interrelating on Flight Decks." *Administrative Science Quarterly* Vol. 38 pp. 357-381.

Weick, K., and K. Sutcliffe (1999) "Organizing for High Reliability: Processes of Collective Mindfulness" *Research in Organizational Behavior* B. Staw and L. L. Cummings. Greenwich CT. JAI press Vol. 21 pp. 81-123.

Zolin, R. (2002) *Trust in Cross-Functional, Global Teams: Developing and Validating a Model of Inter-Personal Trust in Cross-Functional, Global Teams* Ph.D. Dissertation, Department of Civil and Environmental Engineering, Stanford University.