TOWSON UNIVERSITY
OFFICE OF GRADUATE STUDIES

A FUZZY COMPUTATIONAL INTELLIGENCE APPROACH TO MONITORING INFORMATION TECHNOLOGY PROJECT SCOPE

by

Joseph M. McQuighan

A Dissertation

Presented to the faculty of

Towson University

in partial fulfillment

of the requirements for the degree

Doctor of Science

Department of Computer and Information Sciences

Towson University
Towson, Maryland 21252

May 2013

TOWSON UNIVERSITY
OFFICE OF GRADUATE STUDIES

DISSERTATION APPROVAL FORM

This is to certify that the dissertation prepared by **Joseph M. McQuighan**

entitled **A Fuzzy Computational Intelligence Approach to Monitoring Information Technology Project Scope**

has been approved by the thesis committee as satisfactorily completing the dissertation requirements for the degree **Doctor of Science**

2/25/2013
Chair, Dissertation Committee: Robert J. Hammell, II Ph.D. Date

Suranjan Chakraborty 02.26.2013
Committee Member: Suranjan Chakraborty, Ph.D. Date

J Scott Hilberg 2/25/2013
Committee Member: J. Scott Hilberg, Ed.D. Date

2/25/2013
Committee Member: Michael McGuire, Ph.D. Date

Janet V. DeLany 4/22/13
Dean of Graduate Studies Date

Acknowledgements

I would like to acknowledge the guidance and support from my dissertation committee: Dr. Chakraborty, Dr. Hilberg and Dr. McGuire. Their advice and counsel was essential to the successful completion of this dissertation.

My most sincere appreciation goes to Dr. Hammell, my dissertation committee chair, who introduced me to fuzzy sets and fuzzy logic. His mentorship was absolutely invaluable.

A special thanks goes to my wife, Carolyn, who sustained me while I wrote and researched papers many late nights and weekends. I am truly grateful for her love, patience, and encouragement. This dissertation is dedicated to her, the recipient of the 2011 Heart Award at Sibley Memorial Hospital, Johns Hopkins Medicine. Her high standards and devotion to quality have been an inspiration.

Abstract

A Fuzzy Computational Intelligence Approach

To Monitoring Information Technology Project Scope

Joseph M. McQuighan

Monitoring and understanding scope status is essential to the management of information technology (IT) projects. Many IT project failures have been due to mismanagement of scope, a subjective constraint. The result has been a reliance on monitoring systems that measured quantitative data to indirectly infer the status. This study designed methods to address monitoring IT scope when it is intangible.

The design science research methodology was implemented using processes identified by Peffers, Tuunanen, Rothenberger, and Chatterjee. The methodology involved six activities in the building and evaluating of artifacts to demonstrate computational intelligence methods.

The first activity in the methodology established IT scope as a source of costly project failures. The second step defined the objectives for a solution to the problem. The third activity in the research methodology was to build a design. The output of the design process was a set of methods utilizing computational intelligence for both collection and processing of subjective opinions.

The fourth and fifth activities of the design science approach were to demonstrate and then to evaluate the effectiveness of the methods compared to conventional status reports. The evaluation by experts confirmed that the methods can monitor subjective

status. After establishing the utility of the methods, their evaluation considered two extensions of the design: computational linguistic hedges and leading indicators for scope status. The sixth activity in the methodology was to communicate findings to both IT researchers and project management professionals. That was accomplished by presenting the design artifacts in peer reviewed conference proceedings.

In summary, the design science research methodology framed the research questions, presented a design of the methods, the demonstration and evaluation of design artifacts by subject matter experts, and the communications of the findings. The primary contribution of this research is the employment of computational intelligence methods to the subjective, qualitative scope constraint in the monitoring and execution phases of an IT project. The other contributions are utilizing computational linguistic hedges to modify the scope status, and the implementation of CI methods for leading indicators. It was clear that the methods provided information to project managers to initiate corrective actions.

Table of Contents

List of Tables ix
List of Figures x
Chapter 1 Introduction 1
1.1 Outline of the Dissertation 4
Chapter 2 Background 8
2.1 Scope Issues in IT Projects 9
2.2 Monitoring Scope 12
Chapter 3 Literature Review 13
3.1 Project Management 13
3.2 Computational Intelligence 30
3.3 Monitoring IT Projects 52
3.4 Risk Triggers for Monitoring Projects 67
3.5 Design Science Research Methodology 84
3.6 Summary of the Literature Review 97
Chapter 4 CI Applied to IT Project Scope 99
4.1 Implementing the Design Science Methodology 99
4.2 Problem Identification 104
4.3 Research Objectives 105
4.4 Artifact Creation: Building a Design of CI Methods (RQ-2) 110
4.5 Exercising the Artifacts by Demonstration 120
4.6 The Artifact Evaluation Process 133
4.7 The Communications Process 139
Chapter 5 Evaluation and Discussion 143
5.1 Demonstration Scenarios Used in the Evaluation 143
5.2 The Evaluation by Subject Matter Experts 148
5.3 Discussion 165
Chapter 6 Conclusion 184
6.1 Contributions 184
6.2 Limitations of Current Research 187

6.3 Recommendations for Future Work 189
6.4 Summary 196
Appendices 198
Appendix A - Published Papers and Conference Proceedings 199
Appendix B - Evaluation Package 212
Appendix C - Towson University IRB Approval 236
Bibliography 237
Curriculum Vita 251

List of Tables

Table 1 The March and Smith Research Framework 90
Table 2 The Design Science Research Guidelines (Hevner, et al., 2004, p.83) 91
Table 3 The Activities in the DSRM process (Peffers, et al., 2008)................................. 92
Table 4 Computational Intelligence for IT Scope (CI-IT) in the Research Framework. 100
Table 5 Addressing the Research Questions.. 109
Table 6 Research Questions RQ-3, RQ-4, and RQ-5 Mapped to Scenarios................... 150
Table 7 Evaluation Questions Related to Gantt and Earned Value Reports................... 152

List of Figures

Figure 1 Example of Critical Path 22
Figure 2 Operations on Fuzzy Sets 35
Figure 3 Example of a Fuzzy Set 38
Figure 4 Computational Hedges 51
Figure 5 Mathematical Definition of Sigmoid curves 52
Figure 6 Conventional Processes to Monitor Projects with Scope Added 111
Figure 7 Web Gadget for Collecting Scope Status 112
Figure 8 Scope status captured using the Web gadget 113
Figure 9 Steps in Fuzzy Control (Wi, et al., 2012, p. 50) 115
Figure 10 Trapezoid and Center of Sums 117
Figure 11 Example of Fuzzy Sets Membership 118
Figure 12 Status Green 122
Figure 13 Yellow-Green Figure 14 Status Yellow Figure 15 Yellow-Red 122
Figure 16 Status Red 122
Figure 17 Sample Critical Path 126
Figure 18 Current Time on Project for Activity B 126
Figure 19 Sample Subjective Opinion for Activity B 127
Figure 20 Applying Linguistic Hedges to the Critical Path 127
Figure 21 Path with Activity C not started 130
Figure 22 Sample Subjective Opinion for future Activity C 130
Figure 23 Fuzzy Status of a Risk Trigger 194

Chapter 1 Introduction

Monitoring and understanding scope status is fundamental to the management of information technology (IT) projects. The motivation driving this study is the high failure rates and cost overruns of IT projects, many times due to mismanagement of scope constraints (Kerzner, 2011; Schwalbe, 2010b; Gido & Clements, 2009). Research has confirmed status reporting and analysis as a major problem in IT projects (Snow & Keil, 2001). The main focus of this work was to design new processes or methods to assist in monitoring and reporting IT scope. This was accomplished by applying fuzzy computational intelligence (CI) concepts through the implementation of the design science research methodology.

Dow and Taylor (2008) described the scope management process as beginning with the identification of all work items to complete the project. The term "scope" will be taken to mean the project scope from the perspective of the project manager, which is the entire body of work that must be accomplished to deliver the final product. Project managers lack methods or processes capable of handling qualitative data as might be found when reporting scope status. Researchers have shown that because software is intangible, estimates of the work completed are often inaccurate (Snow & Keil, 2002). Brooks (1995) described software as tractable and invisible, with flaws not obvious until late in the project during testing phases or final implementation. The very nature of IT projects makes the management of scope difficult.

The subjective criteria that are factors in project failures have traditionally been ignored in deference to more measureable entities such as currency or time. Existing techniques such as earned value management (EVM) can fail to completely capture scope status on IT projects (Fleming & Koppelman, 2010). In a study of complex projects it was found that soft entities such as scope are difficult to measure, but can provide early warnings of problems (Klakegg, Williams, Walker, Andersen, & Magnussen, 2010). Those same researchers stated that at the point in time when cost and schedule issues are recognized is often too late in a project lifecycle for corrective action. Other measures are needed to provide leading indicators of project success.

Thus a solution that addresses the fuzzy, soft nature of scope and looks to address the need for leading indicators should prove valuable to project managers. Using those criteria as design objectives leads naturally to the use of the computational intelligence techniques of fuzzy logic. Zadeh (1965) created the concepts and fundamental principles of fuzzy logic, one of the areas of computational intelligence. Works by others (Zimmermann, 1996; Klir, St.Clair, & Yuan, 1997; Cox, 1999) offered practical guidance for implementations of theories first espoused by Zadeh.

In his first paper on fuzzy sets Zadeh described the benefit of fuzzy graded membership as providing "a natural way of dealing with problems in which the source of imprecision is the absence of sharply defined criteria of class membership" (Zadeh, 1965, p. 339). Managers of IT projects are often presented with issues that have that type of imprecise boundaries, in which the scope is ill defined, imprecise, and vague. Fuzzy methods share those same properties and therefore are well suited as a potential solution.

The design science research methodology (DSRM) was utilized in order that a design of a solution could be produced and evaluated by project managers. This study conformed to Peffers, Tuunanen, Rothenberger, and Chatterjee's (2008) definition of DSRM through the implementation of their six research steps. Those six activities fit into the "build and evaluate" framework for a design science of March and Smith (2003). The six activities are: (1) identify an important problem, (2) define objectives for a solution, (3) build artifacts, (4) demonstrate the artifacts, (5) evaluate the artifacts, and (6) communicate the importance and novelty of the artifacts (Peffers, et al., 2008).

By following the DSRM, this study produced a design of CI methods for monitoring IT scope. The primary research question was: if IT project scope is monitored for subjective, qualitative criteria using fuzzy computational intelligence methods, could this additional information have a positive impact? The primary research question was then decomposed into five more specific areas of inquiry. Those can be summarized as: a) aspects of monitoring a project that might benefit from computational intelligence methodologies, b) the design of CI methods to capture project scope status, c) determining if a meaningful scope status can be captured, d) ascertaining whether computational hedges are useful in clarifying project scope status, and e) investigating scope data as a leading indicator of project status.

The primary contribution of this research was the creation of design products that utilized computational intelligence for IT project management. The significance of the study was in the application of CI methods to the monitoring of scope on IT projects when the scope is subjective. Traditionally there has been a reliance on monitoring systems that measured quantitative data to indirectly infer the scope status. This study designed methods to address monitoring IT scope when it is qualitative, vague, or even misunderstood. As a result this study moved beyond conventional management techniques of quantitative time and money to managing qualitative scope data. This allowed for the capturing what Klakegg, et al., (2010) called gut feeling and intuition, in a format that can be mathematically resolved into a meaningful project status using computational intelligence techniques.

A second contribution of this research was derived from the primary. It was shown that computational linguistic hedges can generate a more precise expression of the status of the scope constraint. Utilizing CI hedges as qualifiers on scope for critical path tasks more accurately reflects the status. The third original contribution of this research was that computational intelligence methods when applied to project scope are capable of gathering data that can be used as leading indicators.

These contributions demonstrated that monitoring the status of the scope constraint can be accomplished with computational intelligence methods. The CI methods were evaluated by subject matter experts who validated their effectiveness in a business environment. These CI methods resulted in an increased understanding of the status of scope on IT projects. Since errors in managing scope can lead to cost overruns and schedule delays, an enhanced awareness of scope status can become a catalyst to corrective actions by project managers.

1.1 Outline of the Dissertation

This dissertation is divided into six chapters. The structure and organization is explained in the following paragraphs.

This chapter, chapter 1, introduced the research problem. It provided an introduction to the research topic, and included a brief discussion on the computational intelligence methods that were selected; and how computational intelligence is applied in this research.

Chapter 2 is the background. It provides information essential to understanding project management and the phenomenon that leads to misunderstandings of the status of the scope constraint. The background section outlines management issues in software and IT projects. It looks at the multiple constraints of an IT project and why the scope constraint is difficult to measure. Scope was chosen as the subject because of the lack of existing measuring techniques and its role as a major contributing factor to many project failures.

Chapter 3 is a literature review which covers five topics: project management, computational intelligence, project status on IT projects, risk triggers, and the design science research methodology. The review of project management describes how errors in scope lead to cost overruns and schedule delays. The early awareness of scope issues should contribute to rapid corrective actions, thus increasing project success. Computational intelligence and the selection of fuzzy logic is the second topic covered in the literature review. In as much as scope has properties of imprecision and vagueness, fuzzy logic would be an appropriate branch of computational intelligence. The third topic in the literature review looks at how managers gather data for the scope constraint and how that information is aggregated on dashboards to influence decisions on IT projects. The fourth topic is the monitoring of a project using risk triggers.

The last topic in the literature review is the design science research methodology that was used throughout this work. The literature review found that research papers written for IT historically used natural science or behavioral science methods. In an analysis of research methods for IT, March and Smith expanded the possibilities by adding design science as an alternative research methodology (March & Smith, 1995).

This chapter describes design science and enumerates activities for conducting design science research in information technology.

Chapter 4 is the implementation of the design science research methodology for IT project scope. This chapter identifies the choices made where there were alternatives in the design science research methodology. It specifies how the activities that make up design science research were achieved. These activities are aligned with the objective of determining the applicability of computational intelligence to monitor the status of information technology project management scope. The artifacts produced redefine project management by moving beyond traditional metrics to the capturing of subjective inputs for scope during the execution phase.

Chapter 5 presents the results and discussion. The evaluation section describes how experts in the field of project management found the CI methods beneficial. With CI logic and processes to capture subjective data the monitoring of IT projects is transformed from analysis of deterministic and lagging data, to insights from monitoring subjective leading indicators. Techniques to measure scope are shown to improve the understanding of the overall project status.

Chapter 5 includes the analysis and critique of a variety of alternative scenarios that compared the new CI methods for subjective scope to conventional qualitative budget and schedule reports. The expert evaluations of the demonstration scenarios for using CI methods as a leading indicator or early warning of scope issues were very favorable, stating that the new information from the CI methods becomes a catalyst to initiate corrective actions. The experts pointed out that the benefit of the CI methods was not in the capacity to declare a particular status, but the capability to alert management to potential problems. Getting status information for scope meant getting additional information on current activities and early warnings of future problems that could mitigate scope problems, a major issue in IT project management.

Chapter 6 concludes with the contributions of this research. The limitations are covered in this chapter, as are recommendations for future research. The contribution of this research is the innovative application of CI methods to the subjective, qualitative constraint of scope in the monitoring and execution phases of an IT project. The demonstration scenarios clearly showed that the use of CI techniques to monitor scope would assist managers when making a decision to initiate corrective actions. Previous research had applied CI to quantitative constraints in the planning and estimating phases, whereas this study created designs for the monitoring and execution phases in IT projects.

These findings were communicated in four peer reviewed conference proceedings. The common theme was the investigation of methods for monitoring projects that could benefit from computational intelligence solutions. The CI methods and designs were presented at conferences representing two different communities of research. The first community was comprised of researchers in CI and the second was composed of project management professionals and researchers. Appendix A provides full details on the conferences and the interest areas of the attendees.

The appendices contain the evaluation instrument and the communications of the design methods. The series of scenarios demonstrating the methods defined for capturing subjective inputs are documented in the appendices. The scenarios were used to respond to the research questions through an evaluation of the demonstrated concepts by subject matter experts.

Chapter 2 Background

Information technology (IT) projects go through well defined phases starting with the initiating and planning phases and ultimately moving into an execution phase where the bulk of the implementation work is accomplished. One key to success in managing IT projects is the monitoring and controlling process during the performing and executing phase. Projects are typically constrained by factors such as time, cost, and scope (Gido & Clements, 2009). If the project is drifting from those objectives, then it is important for management to recognize the variances and to make adjustments. Since IT projects have high failure rates, this research study looked for causes and potential solutions to resolve the issues that hinder success.

Organizations such as the Project Management Institute (PMI) and the International Project Management Association (IPMA) have published many documents containing recommendations, tools, and techniques to help practitioners manage projects. One example is the PMI's <u>A Guide to the Project Management Body of Knowledge, Fourth Edition</u>, abbreviated as the "PMBOK." The PMBOK identifies processes and methods that can be used to manage cost, time, and scope on projects (PMI, 2008). For example the PMBOK offers the critical path method as a recommended technique that can assist in time management. For cost management the PMBOK suggests using the earned value method to calculate cost variances.

Scope is the set of work deliverables for a project (Dow & Taylor, 2008). For the scope constraint some authors have proposed techniques such as function points (Richardson & Butler, 2006). Function points can be used in IT projects in the early phases to break a project into manageable components. The function points allow estimates as to the level of effort on a project, thus quantifying an otherwise immeasurable entity. However, function points depend on sufficiently detailed requirement definitions, which might not always be available. Kerzner (2011) believes that some metrics by necessity must be qualitative citing goodwill, reputation, or customer satisfaction as examples, emphasizing that difficult to measure metrics can be critical to a project.

Furthermore Fleming and Koppelman (2010), proponents of the earned value model, demonstrated that the accurate measurement of project performance depends upon a complete scope definition. They conceded that software projects are notorious for poor requirements or undefined scope (Fleming & Koppelman, 2010). This suggests that implementing earned value on an IT project might not be sufficient for successful project completion. In fact Kerzner (2011) stated that complex projects by definition have uncertain requirements and uncertain scope, especially when responding to changing business conditions.

2.1 Scope Issues in IT Projects

When 400 organizations were surveyed on the status of their IT projects by the Standish Group, which collects data on real-life software development projects (Standish Group, 2011), their CHAOS Report found that close to a quarter of the reported projects were failures (Levinson, 2009) with only 29% reported as successful. This leaves a significant percentage of IT projects in a less than successful status. CIO Magazine has reported that only about one-third of all projects are successful, with only three percent of projects a with labor cost of over $10 million successful (Cook, 2007). Perhaps more noteworthy is that in spite of advances in software engineering and project management the cost of failures is extremely high. If IT failures occur between five to fifteen percent of the time, Business Insider extrapolates that to an estimated cost of $50 to $150 billion dollars wasted per year in the United States (Hardy-Vallee, 2012).

With such high and costly rates of failure, a search for the causes has lead to the conclusion that vague requirements and poor scope management are significant contributing factors (Levinson, 2008a). Other researchers and authors support this finding, pointing to cases such as the bankruptcy of FoxMeyer Drug in 1996 due to an IT project that had scope problems (Bulkeley, 1996) or the $170 million project failure by McDonalds Restaurants in 2001 due to scope problems (Youngkuk, 2008). IT failures even get reported in 10-Q filings with the Securities and Exchange Commission. In 2008 Levi Strauss & Co. reported a drop in profits due to a faulty ERP implementation (Perry, 2008). The actual charge against earnings was $192 million for an IT project that was estimated originally to cost less than $5 million (Flyvbjerg & Budzier, 2011).

There are a number of reasons why scope is problematic. According to Weill and Broadbent (1998) projects are sometimes late due to new business needs that occur during the project. These changing needs impact cost and schedule (PMI, 2008). Gido and Clements (2009) list a variety of causes for scope problems: the project team might change the design, verbal agreements contribute to misunderstandings of scope, or even a response to the occurrence of events that were identified as risks. Schwalbe (2010b) pointed out that sometimes from the beginning of a project it is well known that the scope is unclear. It has been suggested that on large complex projects humans are incapable of completely describing the scope (Kerzner, 2011). With so many different sources of scope problems, it is the responsibility of the project manager to quickly identify scope issues, so that corrective actions can be taken.

Scope is also much more difficult to monitor because of the lack of an objective unit of measurement on IT projects. Schwalbe (2010b) stated that managing scope is especially difficult on IT projects, because verification is a word based activity. Fleming and Koppelman (2010) in the fourth edition of their text on earned value management (EVM) stated that scope management is the greatest challenge of any project manager. They go on to state that scope is crucial to earned value management, yet software projects fail to implement earned value because of the difficulty of defining the full scope of an IT project.

The United States government has mandated EVM on large projects which has been implemented with varying success. Kwak and Anbari reported in the 2012 issue of the *Project Management Journal* on findings at the National Aeronautics and Space Administration (NASA) on ways to improve NASA's implementation of EVM. They pointed out that NASA only uses EVM for cost and time schedule management, which is the norm for most organizations. However they also stated that NASA needs a scope management indicator to capture what they called "stability" of the scope constraint on NASA projects (Kwak & Anbari, 2012). They did not describe what form this scope indicator would take, or how it might work.

European researchers concluded that on complex projects "soft issues" were hard to measure, but those same soft issues were clearly capable of acting as early warning signs on problem projects. The early warning signs were often in the form of "gut feelings" (Klakegg, Williams, Walker, Andersen, & Magnussen, 2010), which they insisted must be recognized. They added that additional measures beyond cost, and time must be found to reveal those early warnings of project issues.

2.2 Monitoring Scope

If scope is defined in words, then it stands to reason that the monitoring of scope should be comprised of a system that can capture the meaning of words. In the 1990's this was not a realistic possibility, given the limitations of available and accessible computing power. Advances in computing power combined with newly established mathematics and computer science algorithms enable such a solution to be feasible.

This concept of computational linguistic variables was first defined by Zadeh in 1973 in the context of decision processes (Zadeh, 1973), but implementation has awaited sufficient computing power. Later Zadeh (1999) proposed that an implementation of fuzzy set theory could be applied to handle words and propositions. His computational theory of perceptions offered solutions to problems that required making decisions with imprecise information. This dissertation extends Zadeh's paradigm to the problem of monitoring scope on IT projects by the measurement and reporting of subjective scope status using a web based gadget to handle linguistic meanings.

When monitoring the scope for activities on a project's critical path, linguistic hedges can be implemented using computational intelligence. The purpose would be to further refine the interpretation of the scope on the critical path. CI methods would capture abstract, nonobjective scope status that takes into account the relative importance of the critical path tasks. The monitoring and reporting processes would then aggregate the data into an overall project scope status that more accurately reflects the underlying nature of the project in terms of the set of activities on the critical path.

Chapter 3 Literature Review

This literature review encompasses five areas of research. The first four areas are related to the research questions and the fifth area delves into the research methodology. The first area is project management and how scope is one potential source of major problems on IT projects. The literature review confirmed that scope is a subjective criteria that is difficult to measure. The second area that is reviewed is computational intelligence which can provide methods or techniques that can be applied to the identified problem area. The third is the monitoring and reporting processes on IT projects. Research was undertaken to determine if others have combined scope and computational intelligence to offer improvements to the field of project management. This was not found to be the case. The fourth area covered in this literature review is risk triggers as it relates to the monitoring of a project. The last area covered in the literature review is the design science research methodology.

3.1 Project Management

Project managers collect data on the performance of their projects in order to be able to report the status and to forecast future performance. The Standish Group, a respected information technology (IT) research organization, has surveyed IT executives for many years. They recently reported that less than a third of information technology projects were successful, with close to a quarter of the projects reported were failures (Levinson, 2009). Data gathered from IT executives by the Standish Group for 2010 were slightly higher, but this is dismal considering the vast sums of money spent on IT projects.

The magnitude and number of project failures is alarming, highlighted by InformationWeek featuring an IT blunders hall of shame. Notable failures reported by InformationWeek include the McDonald's restaurants multi-million dollar ERP fiasco blamed on scope mismanagement (McDougall, 2006) and the bankruptcy of FoxMeyer Drug in 1996 due to an IT project that had scope problems (Scott, 1996). Another example of an enormous IT project failure is the federal upgrade of the FBI Virtual Case File system that contained over 730,000 lines of computer code. The headlines in the Washington Post stated that the FBI spent $170 million on an unusable system (Eggen & Witte, 2006). Strategic PPM reported that the FBI project took four years, but the requirements were never met (Moore, 2010). The failure of the FBI to deliver a working case file management system was blamed for the most part on scope mismanagement.

Levinson (2008b) stated that problems with requirements and ineffective scope management contributes to failures on IT projects. Much has been written about how to manage scope, starting with improvements to business cases by establishing clear objectives, to ensuring requirements specify an acceptance criteria, and to change management processes. The measuring of scope status has largely been ignored because of the difficulty of measuring requirements.

3.1.1 Phases of Project Management

For the life cycle of management on information technology projects there have been a variety of abstractions and models. One model comes from the Project Management Institute (PMI) which has defined five major groups in their PMBOK (PMI, 2008). Klakegg, et al., (2010) have defined a higher level approach which they labeled a Coarse Project Reference Model. Their first phase is the project definition and design where scope and content define the financing and timelines for a given project with scope definition establishing the work to be accomplished in the project. This corresponds to the PMI initiation and planning process groups.

Their second phase is the project execution, which includes the groups that the PMI labels the executing process group and the monitoring & controlling process group. The final process group for the PMI would be the closing of a project, at which point the project is essentially completed, which Klakegg's team would put into their second phase. In either model, the bulk of the labor and expenditure of funds is done in the execution phase (PMI, 2008), while the majority of the project management tasks are in the planning phases.

There has already been work done on the applicability of fuzzy measurements to the project definition & design phase, which are described in more detail later in this dissertation. This research assumed that the early phases of project estimation have been completed and limited this dissertation to the project execution phase. The assumptions were that the correct project has been selected and the charter is fully supported by all stakeholders. Testing the validity of these assumptions might provide an opportunity for future research regarding the application of CI methods and processes to the pre-implementation phase.

The PMI describes the execution phase of a project as being overseen by the monitoring and controlling processes (PMI, 2008). Lewis (2001) stated that monitoring progress should answer three questions about the execution. The first is to understand the actual status. Second, if there is a deviation from the plan, what is the cause. The third is to understand what corrective action should be taken to fix deviations from plans. Lewis stated that in responding to the third question an alternative might be to cancel the project. Many times this becomes a political decision with ramifications outside the project managers control.

For projects of any complexity beyond the trivial, a stepwise refinement of the requirements is the norm (PMI, 2008). This means that the full, complete, and exact specifications are to be determined at some future date, which can be a contributing cause of scope problems. Not only is scope incompletely defined to start, but it also has the potential to spin out of control as the project progresses, a phenomenon known as scope creep. Meredith and Mantel (2009) described scope creep on software projects as "legendary". Problems with scope creep are pointed to as the number one reason projects fail in a poll of project managers as reported in May 2012 (PMI, 2012).

The goal of the computational intelligence methods would be to identify quickly the project activities where the scope has failed to be understood by project participants and therefore has to be refined, repaired, or replaced. Scope changes most often require additional budget and schedule changes which should be approved through a change management process. In the project execution phase there is both the monitoring of scope status, and the feeding of that status to the risk management process as a risk trigger in the execution of a project. This is one of the boundaries for this research study: the monitoring, measurement, and analysis of scope status during the project execution.

3.1.2 Project Constraints

Many authors and experts on project management (Meredith & Mantel, 2009; Gido & Clements, 2009; Schwalbe, 2010b) use the PMI's model architecture which breaks down into knowledge areas. For controlling a project, the crucial knowledge areas are scope, time, and cost constraints (PMI, 2008). Each of these PMI knowledge areas have a set of processes with prescribed inputs, outputs, and a list of recognized tools and techniques. The general flow according to the PMI is that a project charter should be written, followed by a precise requirements document. A work breakdown structure (WBS) is then built which defines the boundaries of the project. This has been described as the outer limits which clarifies the scope of the project (Fleming & Koppelman, 2010). Dow and Taylor (2008) state that the WBS is used to manage scope, which itemizes the work items to necessary complete the project.

There is a variance of opinions as to the meaning of the term "scope" in the context of IT projects, depending on the perspective of the author. Sometimes the phrase "scope of work" is used to mean the boundaries of the business area, with "scope of the product" referring to the functionality (Robertson & Robertson, 2006). Software engineers are more precise by separating "requirements" from the outputs of the system and detailed design phases (Sommerville, 2011). For the purposes of this dissertation the term "scope" will be taken to mean the project scope from the perspective of the project manager, which is the entire body of work that must be accomplished to deliver the final product. This is the PMI definition (PMI, 2008), and one that is used by a number of authors of project management texts (Dow & Taylor, 2008; Gido & Clements, 2009; Schwalbe, 2010a).

One way to illustrate the difference between scope and requirements is to look at an example in IT projects. The requirement might be to create a graphical user interface (GUI) that is web enabled. What is not stated in the business requirements, but is implicit, are the tasks that become the entire body of work that needs to be done, that is to say, the total scope for the project. In the GUI example the scope might include the need to draft a detailed design, to create test scripts, and to build test data that might never be seen by customers or end users of the GUI. These implicit tasks are expected to be performed by a professional software engineer, but are not documented as part of identifying the business needs (Sommerville, 2011). Requirements might be products, results or components while scope includes both all the work necessary to create the products and the processes used to create the products (Schwalbe, 2010a).

After a work breakdown structure is derived from the scope definition, the budget and schedule are then constructed (PMI, 2008). At this point project managers would control the project by monitoring the expenditures compared to the budget, and tracking the schedule based on time (Gido & Clements, 2009). This is based on the fact that these two constraints have objective measures that can be directly monitored. However, Fleming and Koppelman (2010) have pointed out that scope management is the greatest challenge, and with an incomplete or ill-defined scope projects are at risk. Kerzner (2011) agreed that time and cost are the metrics used on most projects, but also stated that these two together might fail to indicate future problem areas.

For the purposes of this research, it is important to note that scope is a predecessor to the establishment of both budgets and schedules. Scope feeds the construction of the WBS, and time and cost are charged to the WBS activities. A search of the existing literature finds there are no methods that can effectively monitor scope when the scope is deficient, incomplete, or obscure. The current state of the art assumes that project scope must be well understood and well defined, and relies on words to communicate scope status. For example, function point analysis is a structured methodology, which works well when the screen formats, report layouts, interfaces, and other components of an IT project can be clearly defined (Richardson & Butler, 2006). If these conditions do not exist, then function points should not be used, or more work must be done to define the scope. This would delay the start of a project and is counter to the concept of stepwise refinement that is in common use on IT projects.

The PMI recognizes that there are cases where deliverables or subprojects are not clarified, and instead of decomposing the WBS in total and complete detail, rolling wave planning is utilized (PMI, 2008). With new technologies and new methodologies constantly being introduced in the IT world, and with scope as a predecessor to time and cost, there is a need for systems and processes that can monitor the undefined scope or tentative requirements. Drucker supports this in his works on management by stating that controls should be "appropriate to the character and nature of the phenomena measured" (1993, p. 500). The measurements should match the structure of events to become information upon which action can be taken. Techniques in common use today do not match the subjective properties of poorly defined scope or ambiguous requirements.

3.1.3 Objective Measurements

The cost and schedule constraints of project management have numerical quantities that can be measured. The numbers have an element of objectivity which can be used in forecasts. Econometric methods such as regression analysis and autoregressive moving averages, or time series methods such as linear prediction, trend estimation, and moving averages have been used by practitioners of project management (PMI, 2008). Currencies are tracked and reported using time series methods such as earned value. Calendar dates and/or labor hours can be tracked for the time constraint.

Depending on the project, quality might also be measurable and reportable. The PMI states that the purpose of the metrics is to predict future project performance, which can be done with a number of different tools and techniques (PMI, 2008). However, all of the methods assume the collection and reporting of hard data using concrete and objective metrics.

The traditional process in managing software projects has been to estimate and quantify the objective measures of time and cost and then to track progress against those estimates. The Project Management Institute lists a variety of categories for project estimating such as expert judgment or historical relationships, all with the goal of producing a baseline (PMI, 2008). For the cost constraint the PMI baseline leads to a time-phased budget that is monitored as the project progresses.

To come up with estimates for constraints there are a wide range of methods from COCOMO II (Boehm, 2010), Function Point analysis (Heller, 2002), COSMIC Functional Size Measurement Method (COSMIC Method Publications, 2011), and commercial products such as PRICE Systems (DeMarco A. , 2011) which integrates source lines of code (SLOC) with USE Cases and function points. An estimate to determine the number of tests that should be built in a structured testing environment can be done using cyclomatic complexity. Cyclomatic complexity is a metric for software that counts the number of independent paths in a computer program. Fuzzy logic has been applied to give better estimates when using cyclomatic measurement tools (Martin, Leboeuf, Yanez, & Gutierrez, 2005).

Since its publication in 1981 the Constructive Cost Model (COCOMO) has helped in estimating the size of a software project by running scope or requirements inputs through models to produce the estimates. Functional sizes for comparative purposes have been documented in an industry supported historical database (ISBSG, The International Software Benchmarking Standards Group, 2011) and are used to help in project planning and estimation. The ISO/IEC standard 14143 on information technology software measurement documents an established process for using functional size on projects (ISO/IEC, 2007). The general principle of many of these techniques listed above is the documentation of functional requirements to the business. Together with historical experiences as inputs, a realistic project estimate can be output, when the requirements are well established.

Meredith and Mantel (2009) list a variety of time management techniques including Program Evaluation and Review Technique (PERT), the Critical Path Method (CPM), Gantt charts, and Goldratt's critical chain. The PMBOK states that the critical path method calculates the theoretical early start and finish dates by performing a forward and backward pass analysis of the network schedule (PMI, 2008). This generates a critical path, which is the set of activities that determines the duration of the project. A critical path by definition is the longest path through the project. Managers focus their attention on the critical path, which is usually represented as a network diagram (Gido & Clements, 2009), because deliverables on the critical path that occur on time lead to projects that complete on time.

An extremely simple example of a critical path is illustrated in Figure 1. For the purposes of this example, the critical path shall be defined as A-B-C-E. The status of the time constraint would normally be seen as the status of the critical path. Using the Critical Path Method, activity D might have no impact on the project schedule if it runs late (Gido & Clements, 2009). This study assumes that with the availability of well established tools and documented historical experiences, a reasonable project schedule can be constructed. With the current state of high failure rates on information technology projects, this study addresses the monitoring of the abstract and intangible variables that have been identified as a major contributor to project failures. What has been lacking are the techniques to capture the state of the intangible constraints during the project execution phases. The upfront estimates on a project might be good, but tracking subjective status throughout the implementation phase can be dubitable.

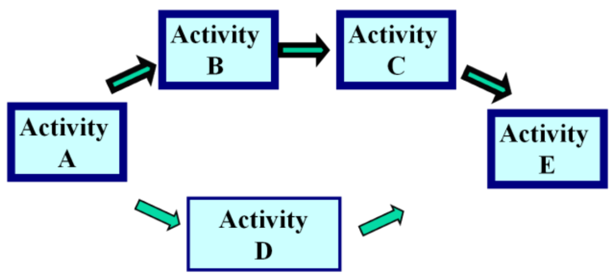


Figure 1 Example of Critical Path

3.1.4 Information Technology Project Scope

Monitoring subjective scope is difficult due to the perceived impossibility of quantifying subjective criteria. As already described above, there are numerous techniques for estimating scope, but once the project is underway, problems on projects are not recognized in a timely fashion. However, Deming (1994), one of the most respected writers on the subject of quality control, emphasized the importance of measurements to managers. Deming's success in Japan with statistical methods is widely recognized, and one can infer that the lack of measurements for monitoring scope when scope is subjective could be one of the problems with the management of software projects. Kerzner (2011) justifies the use of metrics because of their ability for early identification of issues, which leads to informed decision making.

With computational intelligence methods there is the potential to record project scope status so that it can be stored and evaluated by leading edge CI techniques. The first benefit of capturing subjective scope status is that what had previously been tacit and unexpressed is brought into management attention. A second benefit is that scope can become a leading indicator of project status, in addition to the conventional measures of time and cost which are essentially lagging indicators. The concept of leading indicators is explained later, but the point is that effective decisions regarding a project must be made in a timely manner based on current, accurate information.

Gido and Clements (2009) emphasized that scope creep, which is unapproved changes to a project, is a frequent cause of budget overruns and late deliveries. They stated that causes range from customers verbally asking for more, to changes added by the project team with the intention of improving the end product. Articles in CIO magazine blamed poor requirements and scope management as major contributing factors to IT project failures (Levinson, 2008a). Kerzner (2011) would add that insufficient time may not be given to scope definition during the initiation of a project, leaving project managers with a less than perfect set of requirements. Meredith and Mantel (2009) opined that because software is an intellectual exercise without a physical element, developers are highly optimistic, leading to estimating errors. Adding to that mistake they point out that there is an assumption that workers are interchangeable, with the result that organizations ignore the importance of communications and experience.

Regardless of the source of scope creep, the impact is a change in the work which almost always increases costs or causes time delays. Dow and Taylor (2008) have stated that project scope control is a critical task. The conventional solution is to implement change control, but this is based on the assumption that all scope changes can be recognized and managed. Given the high rate of project failures, this assumption has been proven false. An element of this assumption is that scope problems are communicated quickly. With limited time and resources to manage a project, it is likely that scope issues get buried under the weight of other issues. Given that time and cost have visible measures of money and calendar days, software project managers rely on earned value or the critical path method, means to measure and monitor these visible quantities, ignoring the challenge of measuring subjective criteria.

3.1.5 Subjective Nature of Scope

Scope is much more difficult to measure, and at the same time is the critical element from which the WBS time and cost are derived. It has been observed that "the concept of project scope is a foundation idea. It establishes the base for much of the subsequent management activities" (Richardson & Butler, 2006, p. 115). At a high level overview of the project management processes defined by the PMI, scope is derived from the project charter and requirements with the scope baseline then feeding into the Work Breakdown Structure (WBS) (Dow & Taylor, 2008). The WBS is the input to time management, which was an output of the scope definition being decomposed into activities. "Activities provide a basis for estimating, scheduling, executing, and monitoring and controlling the project work" (PMI, 2008, p. 133). However, it has been pointed out that in many cases expectations are not well known, and therefore during the execution of a project deviations are not recognized, directly impacting project success (Meredith & Mantel, 2009).

In a similar manner, scope and the WBS feed into cost estimates and cost management. This means that if the scope is wrong, the time and cost estimates will be wrong, or if the scope changes then time and cost can be severely impacted. Time and cost estimates are calculated by activity, but the list of activities comes from the scope definitions and WBS that were completed early in the life cycle of a project (Gido & Clements, 2009). This implied that scope and requirements errors early in a project can impact and carry over into other constraints that are perceived to be more objective, such as cost. Schwalbe (2010a) stated that managing scope is especially difficult on IT projects. Scope can be relatively undefined at the beginning, can grow out of control due to creep, and later be difficult to verify (Schwalbe, 2010b). Kerzner (2011) agreed that IT development projects are infamous for scope creep due to both business and technical reasons.

The list of IT disasters is extensive enough that there is even a website that documents them year-by-year (Calleam, 2011). One example of an IT disaster is Waste Management Inc. losing $100M in a system implementation because of underestimating the complexity of the project (Calleam, 2011). Other examples of IT disasters include Qantas Airlines abandoning a $40M parts management system (Woodhead, 2007) and the NHS National Programme for IT which was described by the UK government as a costly failure (Maude, 2011). Estimated at over $10 billion over budget, the modernization of the NHS IT systems for healthcare in the UK was years behind schedule (McDougall, 2006) and failed to deliver expected benefits. One more example of the magnitude of project failures was cited in the *The San Francisco Chronicle,* when they reported that the state of California has wasted over $1 billion on failed computer automation projects (Lucas, 1999).

In September 2011, the Harvard Business Review reported that although the average IT project cost overrun is 27%, one in six projects was an unbelievably enormous failure (Flyvbjerg & Budzier, 2011). They stated for a sixth of IT projects studied there were cost overruns of 200% and schedule overruns of 70%. Their conclusion was that it was not that the average of the project overruns was shocking, but the fact that there were such a large number of outlier projects of extraordinary overruns.

Weill and Broadbent (1998) have stated that projects are late sometimes due to specification changes, or new business needs that occur during the project. Kerzner (2011) stated that these types of changes might produce favorable results or competitive advantages. This impacts the budget and schedule which management might be tracking for status reporting. The criteria that are more readily measured by objective criteria (time, cost, and resources) are directly impacted by scope (PMI, 2008). It is the uncontrolled changes of scope creep that adds costs, delays schedules, and reroutes critical resources.

The IT industry is full of examples of scope creep. A Google search of the term "project scope creep" produced over a half million hits in November 2011. A quick review of just a fraction of those web sites demonstrated a common assumption: that project team members know and understand the full project scope. Fleming and Koppelman (2010) have stated that deterministic models such as earned value depend upon a complete scope definition, and that scope creep kills the ability to measure project performance. Their solution was to rigidly control the project baseline through a change control process. They admitted that oftentimes software developers do not attempt to implement earned value because of the difficulty of managing scope. Other references offer the same advice: manage scope through a change control process (Dow & Taylor, 2008).

Change control is a respected technique, but it is based on the assumption that the scope can be well defined, and that the changes will be recognized in a timely fashion. Some project management experts such as Kerzner (2011) do not necessarily support the position that scope control is a guaranteed method to stop scope creep. In his definition of a complex project, inadequate or abstruse scope is the norm which implies the scope is out of control from the beginning. In reporting project status the ascertaining and reporting of scope status is critical, and yet lacks a clear and measureable standard for software projects. Stakeholders and executives have difficulty making decisions based on vague, subjective, and imprecise inputs.

An issue in managing scope on software projects is that people attempt to precisely measure that which is not truly measureable. Hulett (2009) stated that managers know that duration values stated as exact are actually only estimates based on assumptions that could be wrong. In many cases a software developer is only putting forth an opinion that they are 90% complete. Experienced managers recognize a software status of exactly 90% as highly unlikely, and being stuck at the 90% threshold is a common indicator of potential problems (Gerosa, 2003). It has been explained that "it is a common practice to ignore imprecision, treating what is imprecise as if it were precise" (Zadeh, 2009, p. 65). Zadeh suggested that by using fuzzy set theory, a fuzzy data collector could be created to capture subjective inputs, a concept that will be explored in more detail later in this dissertation.

Therefore, concerning IT project management, scope can be inherently fuzzy. At the same time conventional earned value systems are highly dependent upon knowing at all time the percentage of physical work that has been accomplished (Fleming & Koppelman, 2010) This discontinuity or impedance mismatch raises questions as to whether solely measuring quantifiable costs and schedules is sufficient on IT projects. Scope and the corresponding set of requirements are a collection of words describing an end product, and whether or not the deliverable meets the requirements can be open to interpretation.

3.1.6 Warning Signs

Earned value and other techniques are recognized as excellent ways to understand what has already transpired with the intention of predicting future performance of a project. Since earned value measures what has already happened, by definition it is a lagging indicator. When earned value is used to forecast future performance, it is based on outdated data measuring events that have transpired. One insight from the literature review for the research of this dissertation was that project managers need a way to capture "gut feelings" as a means to provide predictive capabilities.

Klakegg, Williams, Walker, Andersen, and Magnussen (2010) performed a comprehensive analysis of complex projects which was published by the Project Management Institute. Their study acknowledged the increased use of sophisticated techniques to monitor and control projects, including audits, reviews, assessments, and benchmarking. As part of their detailed analysis of complex projects in different settings, Klakegg and his team summarized the criteria that they found could be used as measurements that would ensure project success. At the same time they conceded that warning signs of problems are often unclear and imprecise (Klakegg, et al., 2010). They acknowledged that conventional measurements were not a complete solution, and that squishy, soft human intuition sometimes provided better insight to the true status of a project.

Klakegg, et al., concluded that many project managers and sponsors did not recognize problems in a timely fashion, contributing to the high costs of project failures. They attributed this to both the complexity of the projects, and to what they described as tacit knowledge that is undetected. One of their major conclusions was the validity of emotions in assessing project status. Kerzner (2011) would add to their findings that failure to recognize the early warning signs has significant cost implications later in a project.

In summary, scope is difficult to measure and therefore has traditionally been ignored in favor of quantitative criteria such as cost and time. Yet the high failure rate of IT projects has been blamed on scope management problems by numerous studies and publications. Kerzner (2011) went on to point out that time and cost might fail to indicate future problems, instead only pointing out trends based on historical data. Scope on IT projects has fuzzy properties which makes fuzzy computational intelligence look like an appropriate solution. There are a wide variety of technologies under the umbrella of computational intelligence. They are defined in the next section, and the selection of fuzzy logic from amongst those is explained.

3.2 Computational Intelligence

3.2.1 CI Defined

The IEEE Computational Intelligence Society website defines CI as "the design, application, and development of biologically and linguistically motivated computational paradigms" (IEEE, 2011). CI systems are a computing paradigm which McNeill and Freiberger (1993) described as tolerant to imprecision and uncertainty, with the ability to handle continuums of vagueness and degrees of truth. As such, computational intelligence should be able to provide methods and techniques that can handle capturing scope status on software development projects which has those same properties of imprecision and vagueness.

McNeill and Thro (1994) stated that computational intelligence offers a revolutionary set of tools capable of responding to fuzzy, inaccurate inputs. Further supporting this is Zimmermann's description of computational intelligence based systems as capable of mimicking human activities, especially the processing of symbols and meaning, not just numbers (Zimmermann, 1996). The ability to cope with vague data and/or vague rules makes CI methods effective in interfacing to human data input, and as such, can be the new, additional measure that project managers need.

The members of the IEEE Computational Intelligence Society look to implement CI in a variety of core technologies, among them fuzzy systems, neural networks, evolutionary programming, and genetic algorithms (IEEE, 2011). They have determined that these technologies build intelligent systems to help with complex problems in which the information and data are vague, approximate, and uncertain. One aspect of computational intelligence systems is that they are linguistically motivated, which relates to the subject of this study. This research envisions that these tools, techniques, and methods can be effectively applied to project status assessment.

3.2.2 CI Technologies: Fuzzy, Neural, Evolutionary, Genetic, Swarm

The field of computational intelligence encompasses a diverse collection of approaches in computer science, ranging from genetic algorithms, neural networks, swarm, evolutionary programming, to fuzzy systems (IEEE, 2011). The members of the Computational Intelligence Society build intelligent systems from this wide assortment of theoretical possibilities. Chen, Wu, and Cudre-Maurox (2012) looked at the current state of the art of computational intelligence and classified the different methods into three general categories: fuzzy logic systems, evolutionary computation, and artificial neural networks.

In their classification CI methods, fuzzy logic systems included fuzzy logic and fuzzy ontology. Fuzzy systems first gained traction as solutions to control engineering problems, but as computing power increased, new applications of fuzzy logic have emerged. Fuzzy logic works well for modeling complex, nonlinear systems, and those situations where there is vagueness and uncertainty that normally would be resolved using human reasoning (Chen, Wu, & Cudre-Maurox, 2012). The ability to handle imprecise data or approximate reasoning is the domain of fuzzy systems.

When Chen, Wu, and Cudre-Maurox described the evolutionary computation class they included genetic algorithms and swarm intelligence. Evolutionary systems go through an iterative process that mimics natural evolution. Genetic algorithms solve problems through software recombination and mutation. Starting with a potential set of answers the solution evolves through random changes (mutations) to the original set and then the testing of the new set to determine if a better solution has been generated. A variant of evolutionary is swarm intelligence which has a collection of simple agents that interact without a central controller (Chen, Wu, & Cudre-Maurox, 2012). The metaphor is from the natural world: the ant colony or the bee hive, where individual units contribute to solutions through optimization processes at a local level. Evolutionary systems can create optimized and specialized solutions by the generating of alternatives in a software process.

The third class described by Chen, Wu, and Cudre-Maurox was artificial neural networks. Zurada (1992) defined the difference between conventional computing and artificial neural networks as conventional computing must be programmed, whereas artificial neural network systems must be taught or trained. Zurada went on to explain that artificial neural networks change their processing structure in response to learned inputs. Therefore, artificial neural networks need to be taught by either supervised training or unsupervised learning. Thus, these systems depend upon the selection of a representative training sample. Whereas the computations in conventional systems are sequential and precise, requiring the coding of algorithms, artificial neural networks are nonlinear, and the solution is described as collective processing. The benefit of an ANN is that it can handle high data volumes, and high dimensions, that are normally beyond normal human capacity (Zurada, 1992). Chen, Wu, and Cudre-Maurox (2012) stated that neural networks can learn complex relationships in a manner similar to the human brain.

Computational intelligence is a field that has grown rapidly in the past 20 years. New concepts, methods, and paradigms might emerge that alter Chen, Wu, and Cudre-Maurox's classification system. This would be understandable given the newness of the field of CI. As it exists in its present form, their classifications are sufficient for this literature review. Their taxonomy gives structure to the search for one of the CI methods to provide for a solution to project management problems.

From the three general categories in the Chen, Wu, and Cudre-Maurox classification system the benefits of each class was evaluated against the defined problems in project management. The issue with scope revolves around the project manager's ability to recognize the status of the scope constraint. The problem is magnified when the scope is not a clean, crisp metric but is immeasurable, as can be the case in IT. The number of dimensions was small, so artificial neural networks would not be applicable. The ability to handle imprecise data or approximate reasoning is the domain of fuzzy systems, and therefore was the best match for this research.

3.2.3 Fuzzy Sets

This research used computational intelligence (CI) methods, in particular fuzzy sets, to understand the status of a project's scope. This use of CI is in contrast to classical Boolean logic, which Klir, St. Clair, and Yuan (1997) described as working well with exact numbers, intervals, and probabilities. Rather than the hard, crisp nature of bivalent logic, fuzzy logic can work with numbers that are less than exact. This section of the literature review looked at the history of fuzzy sets and the types of fuzzy systems developed from those theories.

Zadeh proposed the concept of fuzzy variables that are linguistic in his seminal paper on decision processes in 1973. Zadeh put forth the concept that human reasoning is based on approximate reasoning that collects, extracts, and summarizes information and that type of reasoning can be mathematically modeled (Zadeh, 1973). He pointed out that in humanistic systems precision, digital computing, and formal mathematics do not work and must be replaced by systems that can handle imprecision and partial truths. Three years later it was proposed that fuzzy sets can substitute for units of measure when units do not exist (Zadeh, 1976). This was a radical departure from engineering mechanics and standard mathematics that had depended upon units such as length, width, area, or height. He went on to state that fuzzy algorithmic systems can be utilized for solving complex linguistic problems when the objects do not have crisp boundaries.

McNeill and Thro (1994) described fuzzy set theory as improvements on Lukasiewicz set theory. These improvements originated with the creation of a theory of graded memberships by Zadeh (1965). Graded memberships model the grey, vague areas in the real world that humans easily comprehend, but that traditional crisp logic and mathematics have difficulty handling (McNeill & Freiberger, 1993). Fuzzy sets are often contrasted to conventional binary (crisp) sets. The difference is that with fuzzy sets the boundaries are not absolute, they are continuous with membership not limited to the binary 0 or 1. Examples are words that span a range of possibilities such as late or early, or even colors such as green to represent a good status.

McNeill and Thro (1994) believed that one of Zadeh's main contributions was the recognition that many times words can better describe reality, and that fuzzy logic can better capture and manipulate words and concepts. McNeill and Freiberger (1993) pointed out that adjectives and adverbs moderate or qualify words by acting as hedges. This allows for degrees of membership that helps people reconcile the vagueness in many situations through the use of hedges.

Researchers in fuzzy systems all agree on Zadeh's basic operations as the theoretical foundation for handling fuzzy sets (Zimmermann, 1996; Klir, St.Clair, & Yuan, 1997; Cox, 1999). Membership in a fuzzy set is described as being the degree to which an individual is compatible with the concept represented by the fuzzy set (Klir, St.Clair, & Yuan, 1997). A well accepted definition has been: a fuzzy set *A* has a domain D_A of real numbers with a *membership function* $\mu_A : D_A \dashrightarrow [0,1]$ (Mendel & Wu, 2010).

Intersection	$A \cap B$	=	$\min(\mu_A[x], \mu_B[x])$
Union	$A \cup B$	=	$\max(\mu_A[x], \mu_B[x])$
Complement	$\sim A$	=	$1 - \mu_A[x]$

Figure 2 Operations on Fuzzy Sets

Three of the operations that can be done on fuzzy sets are listed in Figure 2, with the intersection being of particular importance later in this research. These operations are similar to classical set theory (Klir, St.Clair, & Yuan, 1997), but with A and B defined as fuzzy sets, not crisp values. Klir, St.Clair, and Yuan give examples of membership functions that are triangular, trapezoid, and Gaussian curves, allowing that there is an infinite variety of possibilities, with triangular and trapezoid being common.

The difference between fuzzy logic and probability has been explained by offering that with fuzzy logic “you have all the information you need. The situation itself makes either Yes or No inappropriate. … Fuzzy answers…handle the actual ambiguity in descriptions or presentations of reality" (McNeill & Thro, 1994, p. 6). To this three characteristics of fuzziness were added: 1) word based, not number based, for example: "hot", not 85 degrees; 2) nonlinear and changeable; and 3) analog or ambiguous, not digital (yes/no) (McNeill & Thro, 1994).

Expanding upon Zadeh's concepts of fuzziness, Zimmermann described one way to perceive fuzziness is as that of possibility, with the idea of a possibility distribution. Zimmermann’s example is that a fuzzy set can be described mathematically as:

$$F\sim \; = \; \{ \; (1,1.0), \; (2,1.0), \; (3,0.8), \; ... \; \}$$

This means the set has a possibility distribution such that 0.8 is the possibility that X is 3 (Zimmermann, 1996). The possibility distribution thus allows for something to be both “true” and “fairly false” at the same time. This is an outcome of the classical Law of Contradiction ($A \cap \sim A = \emptyset$) not applying to fuzzy sets.

Zadeh has stated that a valid application of fuzzy logic is in the handling of poor quality data. Zadeh has gone on to say that "imperfect information is defined as information which in one or more respects is imprecise, uncertain, vague, incomplete, unreliable, partially true or partially possible" (Zadeh, 2009, p. 8). This leads to the core concept that membership in a fuzzy set is a matter of degree. For project managers, when looking at the status of a given line item's scope, that status is often a matter of degree, and not a precise value.

Rather than the hard, crisp nature of binary logic which has been prevalent in the computing industry since the 1960’s, fuzzy logic when implemented in soft computing techniques recognizes graded or grayscale memberships (Klir, St.Clair, & Yuan, 1997). These new programming techniques handle the fuzzy complement shown in Figure 2 which is subtle, because the set is fuzzy, not crisp, and the classical rules of logic such as the Law of Contradiction does not apply. The intersection of a fuzzy set with its complement might result in a subset, not the null set, which is an important feature of fuzzy systems (Cox, 1999). This is very different from classical logic where complementary sets never overlap (Klir, St.Clair, & Yuan, 1997).

This feature of fuzziness allowed for Zimmermann's argument (1996) that fuzzy possibilities could be both “true” and “false” at the same time. This might seem to be an extreme case, but with the measurement of project status being inherently fuzzy, it makes sense to select a branch of mathematics that can cope with contradictory input, such as might be found with scope status. In practical terms this means that the scope of a project can be of status *mostly well understood*, and at the same time that exact same scope item can be of status *a little confusing*.

This was one of the issues that drove this research: is it possible that the measurement of scope is inherently fuzzy, and therefore does it make sense to use methods and techniques that can capture the fuzziness associated with scope status. The computing power available in the 21st century allows for the implementation of the fuzzy set mathematics on common desktop computers. Given the imprecise nature of project scope due to the linguistic nature of requirements, it makes more sense to use fuzzy intervals and fuzzy sets to capture the essence of a fuzzy criteria.

3.2.4 Fuzzy Logic Principles

Klr, St. Clair, and Yuan (1997) stated that fuzzy logic is an application of fuzzy set theory for approximate reasoning. The ability to model the human capacity to reason with vague and incomplete data is one of the strengths of fuzzy systems. Cox (1999) stated that some fuzzy sets could model subjective domains that have no units or metrics. He called these psychometric scales which can be used in either the consequent or antecedent side of the rule process. The models for these scales have membership functions that could be triangular, trapezoidal, sigmoid, or even Gaussian curves, but for computing with words Barranquero and Guadarrama (2010) stated that four point trapezoids (t_1, t_2, t_3, and t_4) are the most common. They pointed out that trapezoids include triangular cases when $t_2 = t_3$.

The design of the functions is the result of knowledge acquisition by a knowledge engineer. The membership degree is usually plotted on the y-axis, the vertical axis of a two dimensional graph, while the x-axis is the opinion or perception that is input by the system user. An example of such a scale is illustrated in Figure 3, which could represent a good or satisfactory status on an activity on a project. In this example, the knowledge engineer has decided that values from 0 to 20 represent a total and complete green status, and above 50 represents a total lack of the green status, with a partial membership when x is between 20 and 50. In this example, a user might decide that a given activity is a good status or "green" which the user communicates by a value of 10 on the x-axis.

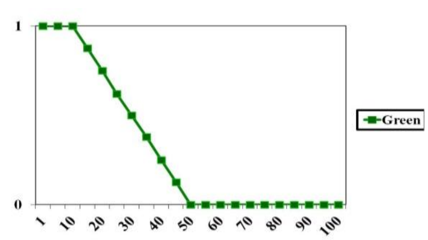


Figure 3 Example of a Fuzzy Set

As was discovered in the literature review when researching project status, end users are reluctant to give a precise number such as "10" to represent a status. For this reason, a fuzzy mouse input mechanism was proposed, the data from which would generate a trapezoid with an associated center of mass. That center of mass value is then applied to the fuzzy set. Barranquero and Guadarrama (2010) investigated computing algorithms that would generate a best fit trapezoid overlay onto fuzzy end user inputs. They created and tested JavaScript prototype gadgets that collected user opinions with an emphasis on ease of use, intuition, and requiring reasonable computations. For usability they wanted to ensure that end users did not need any knowledge of fuzzy logic. They tried three different algorithms in an attempt to discover an effective method to generate a trapezoid to overlay the user's input. The result was that their Iterative approach seemed to best capture the intent of a fuzzy user input and the underlying fuzzy sets.

With the fuzzy data as input, the next steps are to be able to store and process that data into meaningful information. Zimmermann (1996) described seven decision parameters that he found in Mamdani type controllers where linguistic variables are inputs to rule processing. For Mamdani controllers the variables are measured or derived from a signal. The design artifact of this research would use a Barranquero and Guadarrama style input gadget for linguistic variables. Zimmermann went on to describe how the inputs fire the rules to generate control actions, those actions being crisp. The analogous step in this research is that the rules will generate a crisp decision of the status of a task on a project. Zimmermann's seven decision parameters (1996, p. 208) are:

1. INPUT: the number of signals and the scaling
2. FUZZIFICATION: membership functions
3. RULES: number of antecedents, rule base, consequences
4. RULE EVALUATION: aggregation operator in the antecedent
5. AGGREGATION: combining the results of individual rules
6. DEFUZZIFICATION: the procedure
7. OUTPUT: number of signals and scaling

These decision parameters become the components of a fuzzy inference system. One published example was Wi, Oh, Mun, and Jung's model for a fuzzy system. Their model was broken down into five components of crisp inputs, fuzzification, a fuzzy inference engine, defuzzification, and the output (Wi, Oh, Mun, & Jung, 2012). To map their model against Zimmermann's parameters, Wi, Oh, Mun, and Jung's fuzzy inference engine contain the elements of Zimmermann's third, fourth, and fifth steps: the rules, the rule evaluation, and the aggregation while the rest mapped directly to Zimmermann. The design artifact of this research follows principles established by Zimmermann.

3.2.5 Selection of Fuzzy Sets

The characteristics that McNeill and Thro (1994) defined for fuzziness: word based, nonlinear, changeable, and analog are also the characteristics of the scope of an IT project. Fuzzy sets are a good match for monitoring scope status on IT projects because fuzzy systems are capable of replicating human decision making. This includes handling vague data, which can include perceptions. Zadeh (1999) has made the comparison that measurements are crisp and perceptions are fuzzy, an observation that can be applied to the perceptions of project managers as to the state of the scope constraint.

Fuzzy set theory is different than probability theory in that fuzzy sets handle uncertainties in natural languages (Klir, St.Clair, & Yuan, 1997). The vagueness inherent in human communications is a limitation that has traditionally been seen as an unsolvable obstacle for computing systems. Conventional crisp logic dictated that a series of precise and fixed mathematical principles be followed. One such Boolean rule is the Law of Contradictions, which basically states that something is either true or false, but cannot be both at the same time. This is sometimes known as the Law of the Excluded Middle used in mathematical tautologies (Klir, St.Clair, & Yuan, 1997). Fuzzy logic removes this restriction, and recognizes that in many cases when words are used to describe a situation the words might be overlapping or even conflicting in meaning.

Conventional (crisp) set theory is more restrictive than fuzzy sets in that most crisp sets define clean, clear boundaries. An item is either in a set or outside of a set. With fuzzy sets, items are seen as potentially belonging to a grey area that is shaded and nuanced. For project managers, this is exactly the type of data that needs to be collected: the fuzzy intuitions and hunches of the team members as to the real status of the project.

Fuzzy sets were selected for this study for a number of reasons. One is that fuzzy sets can handle imprecise or approximate data (Cox, 1999). Cox has designed and implemented fuzzy systems which he states can reflect the real world for complex, non-linear problems. Fuzzy sets handle gradual boundaries gracefully instead of forcing artificial lines, in fact depending upon overlapping fuzzy regions (Cox, 1999). This allows for ranges that can transition, and not require an exact point. Zadeh has likened fuzzy systems to a broad stroke of a paint brush, instead of the precise point of a sharp pencil. In his acceptance speech when receiving the Ben Franklin award at Villanova University in 2009, Zadeh provided an analogy for fuzzy logic: "in bivalent logic, the writing/drawing instrument is a ballpoint pen. In fuzzy logic, the writing/drawing instrument is a spray pen—a miniature spray can — with an adjustable, precisely specified spray pattern" (Zadeh, 2009, p. 5).

With gut feelings and tacit knowledge as the source of information on IT projects for scope status, fuzzy logic methods and fuzzy set theory are a valid choice. The measurement of scope is inherently fuzzy, and could range over antipodal possibilities, therefore it makes sense to use methods and techniques that can cope with contradictory input on scope status. Since modern computers can implement the linguistic concepts that Zadeh called computing with words, it is reasonable to expect fuzzy systems to provide a solution. From a wide palette of alternatives in the field of computational intelligence, this study selected fuzzy logic as the best match to capturing the status of a project's scope.

Mendel and Wu (2010) have expanded upon Zadeh's computing with words paradigm and applied it to the process of making subjective judgments. They found that fuzzy sets themselves could be fuzzy, an anomaly that they rectified by using Type-2 fuzzy sets. This meant that the underlying fuzzy set also had grey edges, but this allowed them to build systems that could make linguistic decisions that closely resembled the human decision-making process. Whereas a type-1 fuzzy set has a domain and a membership function with grades of membership that are crisp, type-2 fuzzy sets have grades of membership that are fuzzy (Mendel & Wu, 2010). Mendel and Wu describe this as still capable of set-theoretic operations such as Zadeh's extension principle or interval arithmetic. These operations were explained earlier in this literature review by reference to works by Klir, St. Clair, and Yuan (1997).

Mendel and Wu coded the type-2 fuzzy sets in systems which they labeled "Per-C" for perceptual computing. Ultimately the Per-C system required a substantial set of data defined in advance of using the Per-C because the system requires a pre-built vocabulary. The vocabulary would arise from a panel of experts who would define ranges for a series of English words such as: *tiny, smidgen, little, small, modest, large, huge*, ... and so forth. With English words there would be overlap between terms, but generally Mendel and Wu found consensus as to the relative positioning of the meanings of the terms. This property could then be used in computing systems. The reasons why type-2 fuzzy sets were not investigated further in this research is described below after the discussion on Likert systems.

3.2.6 Computational Intelligence for Project Constraints

Computational intelligence has already been applied to project constraints by researchers. Some researchers have applied Zadeh's ideas to project costs where the linguistic variables might be defined using fuzzy terminology such as: (costs = {over, on cost, under}) (Li, Moselhi, & Alkas, 2006). Li, et al., proposed a forecasting method for budget and schedule constraints using Fuzzy Logic to compensate for the variability found on construction projects. They looked at four different, generalized methods to forecast project status. The first were stochastic methods that assumed each unit of work has a mean and standard deviation, but according to those researchers those methods were weakened by variability in costs per reporting period. The second methods were deterministic, such as earned value. The third method that they looked at was social judgment theory based, using human judgment in lieu of mathematical methods. The last method was their suggested use of fuzzy logic for project forecasting and status. Even though cost and time constraints have objective criteria that are measureable quantities, there can be fuzziness in the interpretation of those numbers. Their methodology used fuzzy logic for project forecasting and status for the otherwise concrete constraints.

Other researchers have applied computational intelligence tools to project management for schedule control. As recently as 2007 Wang and Hao proposed fuzzy linguistic extensions to the Program Evaluation and Review Technique (PERT) to replace stochastic methods (Wang & Hao, 2007). Their emphasis was on the imprecision in time estimates, which they resolved by expanding PERT and Critical Path Methods so that the estimates were stored for each activity duration as a fuzzy set. They asserted that too much data may be needed to obtain a random distribution, so fuzzy methods were more applicable (Wang & Hao, 2007). In that their techniques were heavily involved in the estimation phases of a project, this dissertation differs from their approach in two regards. First it moves to the monitoring phases of a project. Second, by going beyond the more easily quantifiable time and cost constraints this dissertation addressed the subjective scope issues so prevalent on projects.

Computation intelligence has been applied to functional sizing and software project estimation by other researchers who devoted their efforts to fuzzy analysis when assessing the size and extent of a software project (Azzeh, Neagu, & Cowling, 2010). Their research on fuzzy grey relational analysis (FGRA) concentrated on the planning processes which are the early phases of a project, not the later phases during the execution and monitoring. Their improvements extended the estimation by analogy technique offered by other researchers (Keung & Kitchenham, 2008).

These data intensive techniques are seen as providing better solutions to estimation problems when the IT or software product is not well defined. Azzeh, Neagu, and Cowling implemented grey system theory to handle the uncertainty inherent in predictions on software projects that have hazy requirements. The cost was that FGRA required a historical database of projects that are similar to the work effort that is being appraised. Their conclusions were that when validating the grey fuzzy estimations with live software projects the FGRA tended to be more accurate than case based reasoning

(CBR) and artificial neural network (ANN) models in accuracy, and superior to COCOMO predictions (Azzeh, Neagu, & Cowling, 2010).

Fuzzy systems have been proven to replicate human decision making by handling vague data, to the point of coping with noisy and/or missing data (Yen & Langari, 1999). Researchers have built a fuzzy inference system for human resource management. Wi, Oh, Mun, and Jung (2012) devised fuzzy rules and a fuzzy inference engine to help in the selection of a project manager. The attributes that can be input into their system include talent, problem solving, supervision, and other predictors of success as a manager. These attributes are rather fuzzy in nature, not crisp and precise. In addition to the fuzzy evaluation system, they also used genetic algorithms to process data in order to generate an evaluation of knowledge competencies of candidates for project manager.

3.2.7 Fuzzy compared to Scaling

In contrast to the relatively new application of fuzzy systems, psychometric scales have historically been used by social scientists in survey research. In Likert scales the survey participants are asked to select one number from a range of numbers. The scales tend to be short and fixed. An example would be to select a unique number on a scale that ranges from 1 to 7. Likert scales require that only one choice be made on the ordered scale, forcing the user to select one and only one value (Trochim, 2011). Likert and similar scales are normally used in social surveys of hundreds to thousands of people. The result is that the evaluator of a Likert survey can take advantage of the mathematical properties of large quantities of data and apply conventional statistical tests, such as variance from a mean.

One drawback from using Likert scales is that it cannot handle those cases when people perceive a given choice as falling into two categories simultaneously. Psychometric models view this human tendency as a paradox, or a violation of the rules. In contrast a fuzzy system allows that project managers might perceive the status in a less than crisp manner. If a project manager uses stoplight reports as described later in this dissertation, the status might be reported as marginal, even though there might be some elements of doubt that the status really was in crisis. Having both statuses at the same time for one activity is an acceptable possibility in fuzzy systems. Likert and other social survey systems require a crisp delineation.

Another drawback to scaling systems is that a mathematically valid number of participants are required in order to perform the statistical analysis. However, the point of gathering scope data is that is a subjective opinion of potentially only one participant might be the only information available. The data collection is not an exercise in group consensus because only one opinion counts: that of the person inputting the data. In IT projects it is often the case that there are only two or three participants working on a WBS activity, a sample size not amenable to conventional probability. For example, by asking the three, and only three, participants working on an activity to input their opinions, the entire population (three elements) has been sampled, and therefore the statistical validity is perfect. This is a false conclusion, because the real issue is not that the entire population has been selected for sampling, but whether or not the data that has been collected can be used in a meaningful manner.

Another benefit to using a fuzzy logic based system instead of a Likert scale is that software developers, and people in general, seem to have problems coming up with an exact number. When given a scale of 1 to 10 and asked to specify a precise number that represents their understanding or beliefs, many people find that objectionable. Software developers at a computer science conference (MAICS 2011) were questioned about this. The informal response confirmed that people really did not like giving a fixed number such as "7" to represent their status. They found that too restrictive and limiting, or thought that the precision was unwarranted. The informal feedback at the conference was that a variable range was highly preferred over an exact number.

There were a variety of reasons given for this observation, but the main reason seemed to be that the fixed number failed to capture the nuances and complexities of a given situation. Selecting one number from a scale from 1 to 10 implied precision for a fuzzy opinion that seemed awkward or even silly. This finding was informal, but Mendel and Wu (2010) found a similar result when implementing computing with words in decision sciences. They stated that individuals are uncomfortable with giving a precise, exact number, but are much more content when selecting a word such as *strong*, *fair,* or *poor* to describe a situation. Their computational intelligence tools implemented Zadeh's computing with words concepts using type-2 fuzzy sets. The type-2 fuzzy sets represented words with overlapping ranges of meaning. Mendel and Wu found that the parties providing precise numbers for opinions detested it, and that fuzzy methods were much more acceptable for capturing fuzzy inputs.

In conclusion, there is extensive research on computational intelligence. From the variety of CI methods, the properties of fuzzy sets for handling vague data are potential solutions for project management constraint issues in the planning and estimating phases.

3.2.8 Application of CI to Project Status

Klakegg, et al, analyzed what should be measured on projects to identify warning signs. They found that signs of approaching problems are "often unclear and imprecise" (Klakegg, Williams, Walker, Andersen, & Magnussen, 2010). In their research they identified the importance of subjective data and they described the vagueness of warning signs, but not *how* to measure them. While other researchers have proposed how fuzzy set theory can be integrated into project management for time and schedule constraints, this dissertation focused on the scope constraint. Artifacts in the form of CI methods were designed to capture scope status directly from experts in a more realistic, human friendly form. Once the subjective data has been captured, it can then be aggregated into an overall scope status for an activity and ultimately the project using CI methods. Without an objective criterion such as currency spent or elapsed time, scope is difficult to measure. Fuzzy systems allow for the capture of this subjective data, and then the manipulation, analysis, and reporting of that data using fuzzy set mathematics.

This dissertation presents a fresh approach to monitoring scope by using CI methods for assessment of the gut feeling on the scope constraint. The scope constraint of IT projects is selected because of the inherent lack of an objective mechanism to measure scope. The subjective nature of scope implies that it is abstract which is in contrast to objective quantities that are physical and tangible. For the reason that the inputs to status reports for the scope constraint are fuzzy leads to one of the research questions of this dissertation: can fuzzy systems offer a method that can capture the status of the scope of an individual activity in an IT project? As will be shown in the evaluation phase of this design science research, subject matter experts found that CI methods can report project status when the inputs are vague or imprecise.

3.2.9 Linguistic Hedges

In the English language when describing the status of scope, verbal hedges are quite common. Adverbs and adjectives are used to modify or clarify the base meaning of a term. If a project manager asks if a software developer has completed a task, a common reply might be "mostly done." The project manager wants a clean, crisp decision and instead gets a status that has been qualified by the hedge of "mostly." Other examples of hedges include "somewhat", "rather", "nearly", or "almost." The English language contains numerous hedges, and it represents the human ability to distinguish, separate, and attempt to communicate shades of meaning and nuances.

Similar to English hedges, computational intelligence hedges modify fuzzy sets. McNeill and Freiberger (1993) explained that hedges operate on fuzzy sets by creating subsets. For example, the word "very" concentrates or shifts a fuzzy set, making a smaller set or subset. A cost constraint that is very over budget has moved from the set of "over budget" into the subset of "very over." They categorized hedges into three groups. *Contrast intensification* hedges separate (very, extremely), *quantifiers* help define (most, several, or few), and *truth values* nudge in a direction (quite true, mostly false.) Cox (1999) supported McNeill and Freiberger’s categorization of the hedge *very* as an intensifier and stated that it is the most commonly used hedge.

Another interpretation of hedges has been offered by Kosko (1993) who likened hedges to weights that are some degree from 0% to 100% that strengthen or weaken a proposition. Kosko used fuzzy cognitive mapping to predict when there is a causal link with the hedges, providing an increase or decrease in the connection. Verbal hedges "a little", "somewhat", "more", or "less" are vague, but computational intelligence methods have defined rules that are capable of handling the fuzziness. Cox (1999) described this as intensifying or diluting a fuzzy region and provides sample C++ code to implement these hedges.

To relate this to project management, an activity might have a set of requirements that are "not totally understood." The hedge *not totally* takes the state called *understand requirements* and modifies that scope status by a little in the direction of confusion. What was a crisp set of understood requirements is now a set that is fuzzy. This research looked at the issue of verbal hedges and how they might contribute to scope management on projects. On an IT project the quantity hedge *not totally* is imprecise, but is realistic and meaningful. It plays a role as a qualifier for the state "understand requirements."

Inasmuch as computational intelligence methods can mimic the English language, this capability can be exploited in applied computer applications. For the purposes of this dissertation's design demonstration, only two hedges were used for simplicity. Those hedges are defined in the next section. A complete production grade CI system would likely span a more complex range of possibilities. Applying computational intelligence to scope on a project in a production environment might entail a project manager creating a more complete set of hedges, and then use computational intelligence techniques to adjust the reported status for an activity. The fuzzy status gathered with the Web gadget would be tempered by the appropriate hedge.

3.2.10 Computational Intelligence Hedges

For the purposes of this example, the scope constraint might have hedges that weight the relative importance of the scope for a given an activity as one of three criteria: very important, neutral, or less important. Figure 4 illustrates the fuzzy qualifier "very" as a curve. Cox (1999) calls these Sigmoid or Logistic representations, or the easier to remember name: S-curves. The role of the S-curve for the hedge "very" is to increase the importance of a subjective evaluation. Also in Figure 4 is illustrated a curve for the hedge "less" which diminishes the importance of the subjective evaluation.

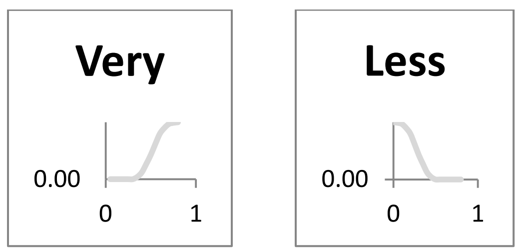


Figure 4 Computational Hedges

The formula to derive the Sigmoid curves is taken from Cox, who defined the inflection point **β**, as the point where the domain value is 50% true. The **β** is selected to put the curves close to the corresponding ends. The **α** and **γ** represented the extremes, with **α** as the zero membership value, and **γ** the complete or 100% membership value. The selection of **α**, **β**, and **γ** would be determined by an appropriate authority, such as the project manager or a project management office. The formulas for creating the curves illustrated in Figure 4 is shown in Figure 5. These computational qualifiers could then be used in mathematical operations to modify scope, a process which will be detailed later.

$$S(x: \alpha, \beta, \gamma) = \begin{cases} 0 & \text{-->}\ x \le \alpha \\ 2*((x-\alpha)/(\gamma-\alpha)) & \text{-->}\ \alpha \le x \le \beta \\ 1 - 2*((x-\gamma)/(\gamma-\alpha))^2 & \text{-->}\ \beta \le x \le \gamma \\ 1 & \text{-->}\ x \ge \gamma \end{cases}$$

Figure 5 Mathematical Definition of Sigmoid curves

In conclusion, this section on computational intelligence started off with an overview of the field and then focused on fuzzy systems. Zadeh's concepts of linguistic variables and how fuzzy systems can capture and manipulate concepts was explored. The explanation of fuzzy logic principles included Zimmermann's seven decision parameters which will be used later in chapter 4. Examples of project management researchers applying CI to objective constraints of cost and time in the estimating phases of project management were found in the literature search. The works of multiple CI researchers (McNeill & Thro, 1994; Klir, St.Clair, & Yuan, 1997; Cox, 1999; Mendel & Wu, 2010) are used as the basis of the designs provided later in chapter 4.

3.3 Monitoring IT Projects

This research began with an assumption that projects played a critical role in information technology. Weill and Broadbent supported this assumption in their statements that information technology (IT) is "very strongly project based" (Weill & Broadbent, 1998). Understanding the status of a project is important to stakeholders, which includes project managers, upper management, and the executive sponsors of a project (PMI, 2008). There are also other parties interested in the status of a project. Many times stakeholders are outside of a project's organizational structure (PMI, 2008). When reporting an overall project status many times it is done as an aggregation of the status of the traditional project constraints: cost, schedule, and scope. Depending on the project, the fourth constraint of quality could also be important (PMI, 2008). Kerzner suggested that there might be many competing constraints, including safety or aesthetic value, in addition to the primary factors already listed (Kerzner, 2011).

These constraints are applied to the WBS activities with the status of each specific constraint representing an aggregation or rolling up of the status of each *activity* status for that constraint (Dow & Taylor, 2008). For example, each activity is examined to judge how it is meeting the cost constraint; the *project* status related to cost is an amalgamation of all the individual activity cost-related statuses. The project status for schedule is similarly determined by first considering each how each activity is performing with respect to that constraint. Finally, the overall project status can be established by combining the individual constraint statuses (Dow & Taylor, 2008). Thus, for the entire project, the constraints are dimensions that are evaluated independently for activities and then later aggregated into a project status.

It would seem that the overall project status should be determined by mechanically and objectively aggregating the individual activity statuses. Instead, it is often reported as the opinion of the project manager, which is subjective and prone to error (Snow & Keil, 2002). Recent postings on a LinkedIn internet site for professional project managers requesting help on project status met with a wide variety of rapid responses indicating the high interest level of practicing project management professionals (Dhanji, 2009). Further research by Snow and Keil (2002) found variances between the true status of a software project from the reported status, and found that accuracy was a major problem. "The intangible nature of software makes it difficult to obtain accurate estimates of the proportion of work completed, which may promote misperceptions regarding project status" (Snow & Keil, 2002, p. 20). Brooks (1995) made a related observation about the risk of misunderstanding as to when the project work is complete. He blamed this on the fact that because software is easily altered, it exposes developers to constant user requests for changes. This risk of an increase in scope directly impacts schedule and budget plans.

Snow and Keil went on to state that executives tend to focus on problem areas and projects in trouble, so in an attempt to avoid negative attention there is a tendency to under report project status. They suggested that this tendency is compounded due to the nature of software which is not easily measureable as compared to construction projects that are visible and physical (Snow & Keil, 2002). Estimating the percentage of work completed for a software development project is an error prone exercise, which Snow attributed to two types of error. One was errors in determining the status, and the second was bias errors on the part of the project managers.

Snow and Keil stated that determination errors were often due to misperceptions or misunderstandings about the status of a software project, but that bias errors were caused by project managers deliberately upgrading or censoring the status reports of poorly performing projects. They cited an example of a project that lost $125 million over 3 years, yet senior management did not have any insights into the problems. "The combined effects of project manager misperceptions (errors) and bias in reporting leads to what we call 'distortion' in the project status information received by senior executives" (Snow & Keil, 2002, p. 20). Snow identified the need for better tools for understanding project status, and the necessity to automate the reporting of status to avoid project manager bias and reporting errors. With other research focused on schedule and budget constraints and a collection of tools already available for time and cost management, this study investigates scope, even though it is the least easily measured constraint.

Errors in judgment and bias in reporting are recognized issues with any monitoring system. Both of these can impact the reporting of status on a project, but these issues are still based on an assumption of integrity. The deliberate misreporting of status due to political or personal integrity issues is a different issue. This dissertation assumed that project managers and team members will honestly record what they know or believe to be true when using the CI methods. In defense of that position, Meredith and Mantel (2009) expressed the opinion that project managers can "tolerate almost any kind of behavior except dishonesty" (p. 444). The CI methods when they were demonstrated to subject matter experts assumed that the data input is a best effort and honest reflection of the state of a given task.

An alternative perspective was provided by Drucker (1993) who explained that problems in corporations can be social events that are not normally distributed, but hyperbolic in nature. By this he meant that some problems arise in clusters and are not evenly and randomly spread throughout a Gaussian curve. He supported this assertion by studies of failed enterprises. In one example, Drucker described how one small team of employees that had difficulties caused the bankruptcy of a corporation. Drucker found that traditional management techniques downplay a singular problem as a small anomaly of a larger statistical measurement. Drucker stated that some problems occur together, and can be significant beyond what a purely statistical analysis would allow. He believed that retaining the relevant, but social, markers of distress is important.

3.3.1 Project Status Reporting

The PMI stated that projects by definition are unique and involve uncertainty which must be managed. "The project team must be able to assess the situation and balance the demands in order to deliver a successful project" (PMI, 2008, p. 7). Part of the process of managing is project communications using status reports to keep upper management or customers informed (Dow & Taylor, 2008). The data provided is used in assessments becoming the feedback mechanism during the execution of a project so that the project can be guided to a successful completion. It should be added that correct metrics must be selected in order to determine project status and enable effective management decisions (Kerzner, 2011).

As projects move forward, project managers are constantly gathering data on the status, converting that data into useful information. Actions are then taken to correct the course of the project. That data can be vague, require more detail, or need interpretation. An example of vague data is that it is difficult to determine to what extent the scope is being met when writing software. To label the scope on a software project as 67.35% complete is recognized as impractical precision. The imprecision in the data was a subject of this literature review. No traditional methods were found that can effectively quantify scope when it is vague. This opened the door for computational intelligence methods and techniques for capturing the vagueness inherent in scope. Computational intelligence methods have the ability to take semantic project management concepts which are ambiguous or obscure and map them to fuzzy sets (Cox, 1999). Cox described fuzzy systems which can adjust to concepts such as which projects are expensive, late, or have increasing costs. The fuzzy data can be resolved to solutions that can be analyzed by project managers.

With over 380,000 members worldwide the Project Management Institute (PMI) is recognized as an authority on project processes, including over 472,000 people who hold the PMI Project Management Professional (PMP) credential (PMI, 2012). The PMI Project Management Book of Knowledge (PMBOK) does not spell out the format of status reports, nor does it tell project managers specifically how to write a status report. This task was left to project management authors (Dow & Taylor, 2008; Gido & Clements, 2009) to provide examples of such reports. Preformatted reports might be included in project management software packages, such as Microsoft Project's visual reports (Schwalbe, 2010a). In the case of the federal government, agencies such as DoD, DoE, or NASA, would create their own reporting standards (Kerzner, 2011).

Instead the PMBOK identified processes, defines inputs, tools and techniques, and the data flows that tie the processes together. The PMBOK should be seen as an assembly of recognized good practices that has the consensus and general agreement of project management professionals. The PMBOK stated that it "is a guide rather than a methodology. One can use different methodologies and tools to implement the framework" (PMI, 2008, p. 4). This allowed practitioners the flexibility to choose techniques that work for their given situation.

The Project Management Institute's PMBOK identified the performance reporting process as part of their Monitoring and Controlling process group. The PMBOK lists three outputs from the performance reporting process: 1) Performance reports, 2)

Organizational process assets updates, and 3) change requests (PMI, 2008). The reports should point out areas where changes need to be made and then generate the change requests as course corrections (PMI, 2008). To this extent, data is converted into actionable information guiding the project to completion. This process of analyzing and reporting performance is crucial to initiating corrective actions and preventive actions, and becomes part of the organization's lessons learned historical database.

Reporting systems that aggregate the constraints into an overall project status has been studied by a number of authors. They found there were complexities that make the automatic summarization difficult. For example, a project might be ahead of schedule, but might also be significantly over cost at a given point in time. Determining the true status of that project is not simple. Just looking at the raw data might yield a green status on schedule, but a "red" status for cost. However, the costs might reflect that fact that the project is ahead of schedule, so it might be the case that the project were finish ahead of schedule, and ultimately within cost constraints. This meant that making status a simple mechanical output of numeric inputs might produce status errors. Management intervention has traditionally been required to interpret the data into meaningful information, but as Snow and Keil (2002) pointed out the interpretation itself can be distorted by human bias and other errors.

In view of the fact that executives and sponsors tend to focus on problem areas, reporting schedule delays would be a prime example of a constraint that attracted attention. The critical path method is commonly used to track schedule progress for the reason that when critical path tasks are behind schedule that is an indication that a project is in trouble (Gido & Clements, 2009). Considering that project managers tend to underreport project status, Snow and Keil (2001) identified the need for better tools for understanding project status, and the necessity to automate the reporting of status to avoid project manager bias and reporting errors.

3.3.2 Stoplight Reports & Dashboards

When reporting the status of projects, Dow and Taylor (2008) found that project dashboards are often used by senior managers. Dashboards are a graphical summary of the status of a project, but some dashboards have a drill down capability to the project details. The purpose is to give a quick, high level overview of a project. Customers and upper management whose role is to prioritize and review projects, make funding decisions based on that information (Benson, Bugnitz, & Walton, 2004). One well publicized dashboard is the Federal government's Open Government Initiative dashboard which uses the stoplight color coding scheme of red, yellow, and green to indicate status (Whitehouse.gov, 2010).

Dow and Taylor stated that the common technique for cost and schedule, is that the constraints are evaluated independently and then summarized in the reports. The PMI mentioned the possibility that a status dashboard might be part of the performance reporting process for scope and quality, but the PMI only provided time and cost examples in their PMBOK (PMI, 2008). Kerzner (2011) however believed that communications via dashboards is an art, not a science, with a design goal of providing critical information concisely and clearly. Some researchers have automated the aggregation of status for each constraint can then computed a cumulative status for the project using computational intelligence methods (Barnes & Hammell, 2008). In that research the assumption was made that scope status was provided, but the means for collecting the scope data was not spelled out.

Since dashboards were designed for quick, high level management review, Dow also found that to assist with quick problem identification stoplight reports are also used. They are produced so that each area of concern is assigned a color to represent its status. Typically the stoplight colors of red, yellow, and green are used to represent the status of a constraint (Dow & Taylor, 2008). Kerzner (2011) agreed that the three traffic light colors are the most common, but added examples of organizations using purple for scope changes. Dow and Taylor specifically noted that in common usage stoplight reports are driven by schedules and costs, with no mention of scope. According to them the green-yellow-red traffic light status reporting is widely used because of its simplicity, and the speed with which people can identify if there is a problem. This traffic light technique is especially popular in status reports to stakeholders who might have little time or inclination to understand the project details (Dow & Taylor, 2008).

Performance reports are essential components to monitoring and controlling projects (PMI, 2008), but the dashboards are designed to visually attract attention. Stewart (2001) has extended the application of stoplight reporting concepts to project oriented balanced scorecards. Kerzner explained the distinction between a dashboard and a scorecard. In his terminology a project dashboard monitored core processes with data that is updated frequently. A scorecard was strategic with a target audience of executives interested in only summary data with scheduled long term updates (Kerzner, 2011).

It is believed that in many organizations a project management office (PMO) would establish the criteria for each color (Dow & Taylor, 2008). Dow and Taylor went on to explain that the next stage in the reporting process would be that the individual activities are then aggregated into an overall project status. The reality is that there are many factors that influence the decision to label a project status with a particular status value for a singular activity, and that the aggregation of those statuses for multiple activities of the critical constraints is open to interpretation as well. Looking at the case of rating just one of the activities in a project, it is simple for status green. Almost all managers would look at a truly on target activity and agree that the status is okay, or green.

If the status moves away from green, it becomes problematic as Snow and Keil, and later Barnes and Hammell, discovered. The latter have shown that yellow status has potential to be misinterpreted and then communicated as green. When the project status is not a clear green or definite red, Barnes and Hammell (2008) found that experts had difficulty deciding that the status of a project is yellow. The problem is much worse when the project is in serious trouble. Snow and Keil (2002) found that the status of high risk IT projects is frequently misreported. For those high risk IT projects in trouble the status is frequently reported in a more positive light than the reality, specifically an IT project status of red is frequently misreported as yellow. Projects that are failing need the most attention from the executive management team, but without the knowledge that the status is red, the proper level of actions are not taken to bring a troubled project into compliance.

There is a wide variety of opinions as to what meaning to assign to a given color when using the stoplight reporting system. The dividing line between the different states is open to debate. Schwalbe (2010a) gives thumbnail definitions for the color indicators as green = on target, yellow=fair, and red=poor, a decidedly vague and fuzzy set of definitions. Others have ascribed variations in the assigning and interpretation of the colors to the risk tolerance levels of project managers (Dow & Taylor, 2008).

Meredith and Mantel (2009) offered that monitoring systems that report deviations from expectations should be brought to attention of the executive management team, so they can apply the proper responses. Alternatively, management might select the option of cancelling a failing project, which would be an appropriate, although unpopular, decision. To remove the human bias and reporting errors CI researchers have proposed a system to collect the data and automate the reporting of status using corporate standards (Barnes & Hammell, 2008). Snow and Keil (2002) only offered that executives should question status reports by looking for bias and reporting errors. Barnes assumed that budget, schedule, and scope statuses can be gathered, and proceeded to the issue of aggregation. However, there is a need for the ability to collect and report scope status when that status is vague and integrating into stoplight reports or red-yellow-green dashboard systems.

3.3.3 Leading Indicators

With the high cost of failures, early detection of problems would allow for more opportunity to make corrections. It has been pointed out that managing by exception has the flaw of raising the exception after it has happened, and thus being too late to correct. Project managers should use monitoring systems to predict potential problems, not to react after problems arise (Meredith & Mantel, 2009). They went on to point out that data that has been easy to collect has been traditionally monitored. Since that data which is objective is easy to measure, it therefore has been what was measured. However, the data which is truly important might be subjective and soft, and difficult to measure. The typical reporting systems in their view are "after-the-fact" in that they measure variances after they occur. Meredith and Mantel stated that project managers should be more interested in preventing future problems than fixing reported variances.

To solve this same prediction problem in a different context, corporate strategic planners have successfully implemented concepts that supported early course corrections by measuring the correct indicators. In the 1990's Kaplan and Norton in The Balanced Scorecard prescribed a methodology for converting business strategy into a practical measurement system. One of their innovations was finding the correct performance drivers which they labeled "leading indicators." They contrasted this to lagging indicators amongst which were the conventional measures of profitability or market share (Kaplan & Norton, 1996).

They did not imply that lagging indicators be ignored, but suggested that the lagging indicators were often too late to implement positive corrective actions. In their view lagging indicators were measures of the success of the changes instituted, and still were necessary. Their innovative ideas were that an organization needed to identify a set of leading indicators that would provide faster feedback and could contribute to a better implementation of the corporate strategy. This literature review took the concept of leading and lagging indicators and concentrated research on leading indicators on IT development projects. Scope was found to be a candidate for a leading indicators on software development projects, but first a look at the Kaplan and Norton definitions and processes is in order.

Kaplan and Norton were concerned with strategic planning of major corporations and how their balanced scorecard could improve financial success. Their processes looked at both long-wave innovation cycles and short term operations of corporations, merging them with the business unit strategy. "The Balanced Scorecard (BSC) translates mission and strategy into objective and measures, organized into four different perspectives... to channel the energies, the abilities, and the specific knowledge of people throughout the organization toward achieving the long-term goals" (Kaplan & Norton, 1996, p. 25). In their BSC process the high level objectives moved through an innovation cycle that produced new products. Products are monitored and controlled by traditional financial measures for costs and budgets (Kaplan & Norton, 1996). One of Kaplan and Norton's conclusions was that these financial measurements have lead to dysfunctional actions, which they described as sub-optimization driven by incorrect measurements.

Instead Kaplan and Norton proposed, and later saw implemented in a number of major corporations, their BSC approach which supported corporate strategic plans. The second of these principles was that a corporation needed to manage both conventional outcomes (profit, market share, customer retention) and also manage performance drivers which are leading indicators. The leading indicators resolved the sub-optimization that was seen when using solely financial measures (Kaplan & Norton, 1996). One of the questions raised by this research was whether or not this concept of indicators can be transferred to IT project management, and if so, what those predictors or leading indicators of project success might be.

According to Kaplan and Norton (1996) the corporate BSC required enhancing information technology systems. Many times these enhancements would be implemented as IT projects, designed to support the delivery of new solutions, products, or services. Projects resemble organizations in that they should have measureable goals and outcomes, but projects have a much more limited set of boundaries than the corporate wide strategic plans (Stewart, 2001). Projects start with a project charter which develop into a set of requirements (PMI, 2008). In the best of circumstances a project has a clear set of requirements, but researchers have found that stakeholders often vary in their level of confidence in requirements (Boness, Finkelstein, & Harrison, 2011).

When itemizing the requirements managers in software development use risk analysis to identify potential causes of project failure. As a result researchers discovered that identification of risks can force a total redesign and even a rewrite of the initial requirements (Asnar, Giorgini, & Mylopoulos, 2011). Other researchers have also found that requirements that produce a clear project scope are crucial to success (Ferrari & Madhavji, 2008) but often organizations fall short of the ideal requirements definition (Saiedian & Dale, 2000).

Leading indicators were those measurements that should increase strategic success by offering an early opportunity for course corrections. To apply the Kaplan and Norton concept of leading indicators to IT software project management, and in recognition of Klakegg, et al.'s, (2010) work on the importance of gut feel and the tacit knowledge of the project team, scope would be an ideal leading indicator. The traditional measures used to manage projects of cost and time would be by definition lagging indicators, important to be sure, but measuring the past, not the possibilities. Keyes (2011) touched upon this by adding analytics to project management when aligning projects with corporate BSC. However, Keyes use of change management statistics is an example of lagging indicators on projects.

Monitoring scope during the execution phase of a project has been a challenge to measure. With computational intelligence (CI) methods, scope status would be captured early in a project life cycle. The ascertaining and reporting of scope status is critical, and stakeholders, project managers, and executives would have a new technique to help make difficult decisions. Scope status inputs still retain the properties of being vague, subjective, and imprecise, but as Klakegg pointed out it is exactly those fuzzy gut feelings that contained potential insights into project problems. This is supported by other researchers who stated that most organizations are faced with complex to chaotic problems that require sensing skills in managers. They believed that the domain of complex problems requires an experimental mode of management to cope with the "unknown unknowns" (Snowden & Boone, Nov 2007). Kerzner (2011) would add that the business world is full of nontraditional projects with changing technologies, ill defined statements of work, and assumptions that must be altered during long duration projects.

Scope status as a leading indicator would be about predicting future outcomes from the subjective opinions of the experts working on an activity. This might vary from a software developer questioning the clarity of requirements, to testers offering an opinion that delivered functionality will not meet customer expectations. If the status of the scope constraint is collected before the expenditure of other fixed resources there is the potential for scope to be a predictor of project status. The addition of a leading indicator could contribute to faster recognition and response to problems on software development projects, thus increasing success.

This section of chapter 3 can be summarized as the literature search identifying current methods for monitoring IT project status. The unique aspects of software as an intangible entity cause errors in assessing project status, especially for the scope constraint. The use of stoplight reports to report status is widespread, but tends to emphasize quantitative cost and time reporting, criteria which are lagging indicators. Kaplan and Norton pointed out that sub-optimization by selecting the wrong financial measurements has often lead to poor decisions at a corporate level. This principle can be applied to the management of projects.

3.4 Risk Triggers for Monitoring Projects

InformationWeek's 2012 project management survey of 508 technology professionals found there were high levels of dissatisfaction with projects (Feldman, 2012). The numbers for business users neither satisfied nor extremely satisfied hovers at around 50% for the time and cost to deliver IT solutions. Feldman pointed out that in a multi-national analysis of 258 projects the Oxford Review of Economic Policy reported that the majority of projects analyzed had cost overruns. These high levels of failure have continued a trend that has been recorded for years by the Standish Group, a respected information technology (IT) research organization, and was described earlier in this dissertation's literature review. They have reported numbers as low as 28% for IT project success (Standish Group, 2011) with data gathered from IT executives by the Standish Group for 2010 only slightly higher. This is dismal considering the vast sums of money spent on IT projects.

There are financial motivations to detect and prevent project failures, from the federal government to the private sector. One prevention method is risk management, but the majority of risk management tools measure probabilities of events that have not transpired. One discovery in this literature review was that project managers need better insights during the execution of a project. Risk monitoring during the performance of project activities would involve determining the extent of an actualization of a risk trigger. An appropriate method for risk events that are fuzzy and obtuse in nature is a subject of this dissertation.

Risk by itself is not necessarily bad, and in fact, DeMarco and Lister argued that risk on projects is desirable. Their position was that companies must take risks to be competitive, and big risks could generate big benefits (DeMarco & Lister, 2003). They also identified another benefit: that success by an organization in taking high risks can overwhelm the ability of competitors to respond. However, risks must be managed and the monitoring process depends upon an awareness of transition indicators. DeMarco and Lister stated that triggers represented that transition from a risk event to an actual problem. When risk triggers are found, contingency plans would be implemented.

Projects of medium to large complexity generally have a life cycle where the largest expenditure of the staffing effort occurs in the performance of the execution phase. This principle of project life cycles is a widely accepted axiom of project management. Meredith and Mantel described this as a curve that starts slow and rises to a peak where there is a strong correlation between effort and progress on a project (Meredith & Mantel, 2009). This effort is the work of a project, and the role of the project manager is to monitor, control, plan, and schedule that work. In looking at projects Schwalbe believed that 70 to 80 percent of the time of a project manager is spent monitoring activities during the executing phase (Schwalbe, 2010a). The importance of monitoring the execution phase was found to be statistically of the highest importance by Hobbs in his study of project governance (Hobbs, 2007).

The monitoring and controlling processes contain a collection of processes that looks at the performance of the other processes on the project. If the monitoring phase identifies the need for changes to the plan, then those changes are incorporated into a revised baseline (PMI, 2008). The PMBOK lists the processes in the Monitoring and Controlling group as scope, procurement, communications, time, cost, quality, and risk. It is the last topic, risk, that is related to scope in the way that scope creep is a well know risk on IT projects.

The monitoring and controlling of risk, according to the PMI, takes as inputs: 1) risk registers, 2) project management plans, 3) work performance information, and 4) performance reports (PMI, 2008). The purposes of monitoring risk are to act as feedback into the processes that regulate the progress of the project, to make changes to the plans, and then to implement those changes. To this extent, data is converted into actionable information guiding the project to completion. This monitoring process of reporting and assessing performance is crucial to initiating corrective actions and preventive actions.

Risk management is one of nine knowledge areas that the PMI considers fundamental to project management. From their perspective, risks are uncertain events that might occur, and could have either a positive or negative effect on a project (PMI, 2008). For the purposes of this research, only negative risks were considered. The PMI techniques proceed as if all risks can be identified and recorded. Although the PMI acknowledged that some risk events cannot be foreseen, their solution is to respond in crisis mode with workarounds. The PMI would manage identified risks by creating a risk register with pre-planned responses, assuming that all risks can be identified in advance.

3.4.1 Sources of Risk

Robertson and Robertson (2006) have stated unequivocally that all projects involve risk. Although their focus was on gathering requirements and specifications, to them part of the requirements process was the identification of risks. Risks are then associated with requirements that have cost or time impacts so that they can be managed. But they point out that projects must go beyond risk analysis in the requirements process to managing risk as an ongoing process project wide.

Hancock expressed reservations about current practices in risk management. It was his opinion that "behavioral and societal aspects of risk are underrepresented in the project risk management processes" (2010, p. 3). As references for conventional risk control based on probability theory, Hancock reviewed the PMI, the British Office for Government Commerce PRINCE2, and the Association for Project Management (APM) standards. Hancock agreed that there are cases where deterministic and statistical models are appropriate, but believed that complex projects are systems where the interrelationships of activities can cause cascading failures (Hancock, 2010). His argument was that system complexities and interactions far outweigh uncertainty measured by quantitative methods. A project with high complexity and a tight coupling of activities with no slack or buffer does not allow time to stop disasters from escalating in Hancock's view. He therefore recommended that high complexity projects add back the buffers and redundancies that were removed by optimizing critical paths.

In discussing risk analysis Robertson and Robertson (2006) listed project drivers as potential sources of risk. Risk items were generally based on the purpose of the project. Risks derived from the project goals might include determining if the goals are reasonable, the potential of cooperation (or lack of) from customers, and user involvement. They found that "project leaders often cite the quality of user contributions to requirements as their most serious and frequently encountered risk" (2006, p. 341).

Project constraints and assumptions can also be sources of risk in their view. They also pointed out that project assumptions might be false and therefore are really risks that have been accepted. Unrealistic schedules, budgets, and scope can also be sources of risk according to them. In particular scope can be a common risk because of the phenomenon of scope creep. As a solution they offer function points as a technique to get a better handle on different sources of risks. Function points were described earlier in this literature review as working well for well defined projects, but not in rapidly changing or ill defined environments.

Based on his practical experiences on very large IT projects, Brooks contributed a number of ideas on risk. One idea that he presented was the need for research and exploration of totally new techniques to increase success on projects (Brooks, 1995). He allowed that research and development were both risky activities, and that the risk needed monitoring. Another idea was that the complexities of computer systems are a source of risk issues. Large complex projects collapse when team members sub-optimize their pieces, at the cost of system integrity. One example is that on large projects he found that the complexity of computer systems introduced new risks when fixing defects, with each fix many times generating new problems (Brooks, 1995). Lastly, Brooks was convinced that users do not know what they want, and generally were incapable of specifying system requirements in enough detail to answer all the questions of design engineers. The issues raised by Brooks were the same issues raised by other researchers already identified in this literature review, confirming earlier findings.

Gido and Clements (2009) suggested that risks can be put into categories, each of which can be further broken down. They offered the following examples of categories: technical, schedule, cost, human resources, external factors like weather or legal, and sponsor/customer. Amongst the technical risks were the inherent risks of any advanced technology, the risks of not meeting performance requirements, the first time use of rare and complex equipment, and the possibility that the originally selected technology became obsolete during the course of the project. Brooks (1995) amplified this concept by going beyond just the technology becoming obsolete. He stated that products can also become obsolete before the projects to produce the products are finished. The remaining categories listed by Gido and Clements can be similarly decomposed. Gido and Clements stated that the impacts of the identified risks could be schedule delays, additional expenditures, customer refusal to accept the product, or sponsors terminating projects. They noted that listing all risks is impossible, and that progressive elaboration of risks as the project proceeds has been an accepted technique. What they made clear was that there are risks that cannot be anticipated.

Hancock (2010) predicted that the trend will continue to be for projects to become more complex, and with that complexity comes less visible sources of risk. In support of this position Yourdon (1997) stated that politics will add unseen external risks to projects. Typical project managers focus on internal risks because these are under their control.

The external risks of organizational, business, or market competition issues are beyond the control of the project team. Yourdon advised that risk management must include a global perspective to cope with both internal and external risks. In summary, there are an enormous number of sources of risk on projects.

3.4.2 Conventional Risk Control

When writing about project risk management many authors refer back to the PMI. Schwalbe (2010a) directly quoted the PMI , Gido and Clements (2009) enhanced their 5th edition by supporting the PMI approach, and Meredith and Mantel (2009) pointed out the historical benefits of the PMI and their methods. The PMI has written a standard for risk management, and outlined the risk management knowledge area in their PMBOK Guide (PMI, 2008). Their basic point concerning risk was that risks must be managed otherwise realized risks could lead to project failure according to the PMI.

In the PMBOK, they described the vast majority of risk management tasks for the project manager as part of the planning process. These range from creating an overall management plan for risk, identifying the list of risks and prioritizing them, numerically analyzing the risks, and planning responses to those risks. Once the plans are in place, the project moves into the execution phase where the risks are monitored and response plans might be implemented. The PMI makes clear that risk refers to future, potential, not actual events (PMI, 2008) and that all projects contain uncertainty. The PMI methods for risks depend upon whether response plans can be developed, or require contingency plans. The PMI added that how an organization responds depends on their risk tolerance.

One of the outputs of risk planning is a listing of potential risks events in a risk register. Dow and Taylor (2008) defined a risk register as all the identified risk events, the potential responses, and the risk owners. Their definition of a risk register matches that of the PMI which uses the register as an input to three other planning processes: qualitative analysis, quantitative analysis, and risk response planning (PMI, 2008). The ultimate result is an updated risk register to be used to monitor and control risks in the executing phase of a project.

Microsoft Corporation considered risk management to be a crucial component of successful IT projects and software development. They have formalized and published their processes in the Microsoft Solutions Framework (MSF) (Microsoft, 2005). There are three disciplines in the MSF: Risk Management, Readiness Management, and Project Management. In Risk Management risks are tracked and reported based on risk levels, which is independent of task completion tracking under operational project management. A key concept of MSF is assessing risk by defining and monitoring risk triggers (Microsoft, 2003). The risk control step kicks in when the risk status changes, thus generating project change control requests. The assumption was that there is recognition that the status has changed, which implies a crisp, clean jump from one state to another.

3.4.3 Risk Responses

The potential responses to risk for the purposes of this research are passive acceptance, active acceptance, and mitigation. Other potential responses to risks are avoidance and transference. Schwalbe described the difference as risk avoidance would eliminate a threat, mainly by adding work to the project, while risk transference would move the responsibility to a new owner (Schwalbe, 2010a). Using those responses would mean those risks no longer threaten a project. Acceptance and mitigation both still leave risks on the project, which is a subject of interest later in this literature review. Those responses to risks are listed in the register and must be monitored and controlled during the execution of the project.

Dow & Taylor (2008) stated that risk acceptance is the norm for low probability risks. For those risks the PMI differentiated between passive and active acceptance. Passive acceptance meant that no action should be taken for that risk. If the event were to occur then an unbudgeted corrective action, called a workaround, must be taken. An active acceptance strategy involves the creation of a contingency reserve (PMI, 2008). The expected monetary value (EMV) approach to creating a contingency reserve allocates funds to cover known risks (Dow & Taylor, 2008). It involved taking the probability of an event and multiplying it times the cost if the event were to occur. Then the fractional costs are summed for all the risk events into a contingency fund.

DeMarco and Lister described EMV as containing the entire set of risks by setting aside "enough resources, on average, to offset the risks that are likely to materialize" (2003, p. 63). The EMV technique assumed that some, but not all, of the events might happen, and that the total funds reserved should cover those few events that do occur. Contingency reserve funds would have an associated action plan that kicks in if the risk event happens. Dow & Taylor (2008) positioned EMV as a method that is dependent upon guessing the probabilities, therefore there is a risk of estimation errors. In addition to estimation errors Zadeh pointed out that the crisp values used in probability calculations are the center of a fuzzy perception, and not a precise value (Zadeh, 2010). Extending Zadeh's concepts to EMV, a thirty percent chance of a risk event is perhaps the center of a range from 15 to 45%, and not an exact number.

An alternative strategy to acceptance is mitigation. The purpose of mitigation is to reduce the probability of a risk item to a lower level. Additional activities must be put into the plans, along with corresponding cost and schedule changes. Mitigation still leaves some risk, in that it is a strategy to reduce the probability of occurrence, but still there is a chance of an event occurring. DeMarco and Lister (2003) added that since mitigation expends funds before a risk materializes, those costs cannot be recovered. They pointed out that mitigation requires the costs of a definite action that must be included in plans, while the contingencies of an acceptance strategy are put into a work breakdown structure only as possibilities. After deciding to mitigate, the remaining risks must be addressed by being accepted (passive or active), transferred, or avoided, just like all other risk events.

3.4.4 Risk Triggers

As was stated earlier, the bulk of project costs and labor hours are expended during the execution phase. The execution of the project work is monitored and controlled by any or all of six techniques: 1) variance analysis, 2) performance measurement, 3) reserve analysis, 4) risk reassessment, 5) risk audits and 6) status meetings (PMI, 2008). The first three techniques use quantifiable measures to identify and manage potential problems. The others might have both a quantifiable component and a qualitative subjective component, which can be vague or imprecise.

The key for all of these techniques when monitoring the execution of a project is to discover which potential problems have changed state from a probable future event to an actual problem that must be resolved (Gido & Clements, 2009). Sometimes this determination is easy, as when physical items are either delivered on time or not. The term triggers is used to describe indicators that are symptoms of actual events (Schwalbe, 2010a). The triggers are warning signs of an occurrence, many times not precise, and not discrete events.

Symptoms and not direct measurements are the only means for qualitative risks that are imprecise or fuzzy in nature. Gido and Clements (2009) listed examples including lack of cooperation of end users, the extent of design flaws, or some integration problems. There is a fuzziness in the terms "lack", "extent", or "some" that make qualitative judgments difficult. Other vague symptoms might be sponsor dissatisfaction that forces scope changes, or a lack of skills to complete the tasks. Precision in measuring dissatisfaction has been problematic. Gido and Clements suggested using regular reviews of the triggers for each risk, looking for risk events that have materialized. If an evaluation of the trigger determined that a risk event has happened, it is no longer a risk. It is an actual event that puts a contingency or response plan into play. Risks are about potential and future events, and are in contrast to an executed trigger which has become a problem to be managed.

3.4.5 Vague Risk Triggers

There are a wide variety of situations in which risk triggers are not precise. Recognizing that a risk event has occurred is simple for physical deliveries. Computers arrive in boxes on the designated date, offices get painted, or bankers write checks to continue funding a project. Information technology projects and software development projects are different. Brooks (1995) pointed out in The Mythical Man-Month that software programming is both invisible and tractable. Given the hidden nature of software, Brooks described software as starting as an idea, and remaining conceptual until final delivery. Designs as concepts might be faulty, but that would not be obvious until late in the project during testing phases or final implementation.

In a review of sources of risk on IT projects, the imprecision of the items becomes apparent. Schwalbe (2010a) authored a table of negative risk conditions that essentially covered the PMI knowledge areas (PMI, 2008). Her list included that scope definition might be *poor,* productivity might be *inadequate*, leadership might be *lacking*, and the workers might have a *bad attitude* towards quality. None of those qualifiers in italics is capable of precise measurement. Instead all are fuzzy and span a range of imprecise possibilities. When monitoring the execution of a project each of those risks would have an owner whose responsibility it is to determine if the risk has been triggered. The basic technique is to treat the trigger in a yes/no or binary fashion. However, when the measurements are imprecise, a technique for collecting that data should reflect the fundamentally fuzzy nature of the trigger.

The norm for events that are possible, but not probable has been to passively accept the risk (Schwalbe, 2010a). An example of a risk that is passively accepted is the possibility of earthquakes in Washington, D.C. Projects might be impacted by a total shutdown and evacuation of facilities in Washington DC due to earthquakes. Planners in the nations' capitol passively accept the risk, unlike California where earthquakes must be taken into account. If that highly unlikely event happened, then workarounds would be created on the fly to cope, as was the case in Washington, D.C. in August 2011 (Stone, 2011). For those possible but not probable risks it has been recommended they be put in a watch list that must be monitored (Schwalbe, 2010a).

Sommerville (2011) emphasized that risks should be monitored aggressively at all stages of a project. He advised that serious risks should go on a short list that is evaluated at every management review. The review would determine if the risk event has occurred. Risks would be assessed as binary events with a statistical probability of occurring. For some risks, this is the case: the event has either happened or not, an example being that hardware defects are repaired on schedule, or not. Yet, in a chart of risk types by Sommerville (2011) some of the risks he identified had both a probability of occurring and fuzzy properties. Two examples were "*poor* staff morale" and "*reluctance* to use tools". "Poor" is a linguistic term that is better reported and monitored by fuzzy methods. It is likely that poor morale has already happened, and the important issue is to what extent. Sommerville did not offer a means for measuring risks when they are grayscale, fuzzy, or vague.

3.4.6 Unforeseeable Risks

The PMI methods for controlling risks was a formal process for managing conventional projects. Loch, DeMeyer, and Pich (2006) identified two other types of projects, both containing an extraordinary number of risks. The first of these high risk project types has residual risks, which most authors and the PMI agree are the risks that remain after handling the managed risks. These residual risks are identified and placed on a sentinel list. Conventional risk management is then done in parallel with the monitoring and management of the residual risks.

Loch, DeMeyer, and Pich (2006) also defined a second type of high risk project, with unforeseeable and unpredictable risks which they termed unknown/unknowns. They labeled these projects with excessive risk as novel projects. They argued that when a project is of very high complexity the interactions of components cannot be anticipated, resulting in the ablating of planning efforts. One response might be to cancel the novelty projects as too risky. However, they pointed out that "industry dynamics and sophistication force project managers to push the envelope in seeking new markets and new technologies" (Loch, De Meyer, & Pich, 2006, p. 3). High risk can lead to high payoffs, at a cost of high uncertainty on projects. This leads to project failures using the conventional tools of management, simply because those tools do not apply. Loch's, et al., criticism of the time critical path method was that it implied deterministic time durations, which they stated was a fiction (Loch, De Meyer, & Pich, 2006). They asserted that complex projects are highly nonlinear systems, where a change in one area can have a ripple effect of disastrous consequences.

Although they were focused on novel projects, there is a wide spectrum of project types into which IT projects fall. Given the high failure rate of IT projects, many projects would benefit from considering that there is always unforeseeable uncertainty. To Loch, et al., unknown/unknowns are fundamental to high technology projects, and recognizing that the unpredictable has occurred is a key to success. When a project only has residual risks it can be managed if the goals are well established. They add that extending the techniques of conventional risk management to include fast response and quick reaction can control deviations from project goals (Loch, De Meyer, & Pich, 2006). The category of novelty projects is radically different and Loch took the position that new methods such as selectionism and learning should be used.

Klakegg, Williams, Walker, Andersen, and Magnussen (2010) also stated that projects can fail due to risks beyond the known risks. They performed a comprehensive study of complex projects looking at why existing methods failed to recognize early warning signs of project failure. Their findings were summarized into three areas: risks, complexity, and inter-personal effects. For managing the known risks they stated that there already exists a large body of knowledge on that topic. They pointed to the publications of the Association for Project Management (APM), a professional organization that has 18,000 members mainly in Europe, as one such source. The APM publishes a Body of Knowledge that contains guidelines to managing projects, including risk management. Klakegg, et al, (2010) opined that documents from the APM provide adequate techniques for managing epistemic or known risks.

However, it was Klakegg's team's second and third categories of project failure, complexity and interpersonal effects, that they stated contained early warning signs of problems. They believed that the complexity of solving difficult problems generates aleatoric risks, which are chaotic and random. Added to that the subjective uncertainty of human interactions and conventional approaches are overwhelmed. They were convinced that early warning signs are often unclear, imprecise, and tacit, and at the same time must be identified and addressed quickly for projects to succeed. They offered some ideas on how to capture feelings and opinions that they admitted were futuristic and not realistic with existing technologies. This dissertation questions that assumption because computational intelligence can bring solutions to vague and imprecise risk problems, an option they had not considered.

McManus (2004) emphasized the significance of monitoring risk because of changes that happen over time. In addition to monitoring the known risks on a risk register, like other researchers he warned that unknown risks can arise during the course of software development. The normal process of monitoring in his view included adjusting probabilities, refining mitigation plans, and finding new risks not identified in the planning process. McManus stated that new risks can derail a project. In Appendix B of his book on software risk management, he listed 46 different computerized tools for risk management, and categorized them as either generic, decision support, analysis, simulation, or risk status monitors (McManus, 2004). Of the 46 tools in the list only one was categorized as a risk status monitor. There is a need to create CI methods to support risk status monitoring, considering the importance of risk monitoring and the paucity of tools in that category.

3.4.7 Fuzzy Properties of Project Risk

In discussing the types of systems where fuzzy logic can be best applied McNeil and Thro (1994) listed numerous criteria. First on their list was complex systems that are difficult to model. Second and third were systems controlled by human experts, and systems with complex and continuous inputs and outputs. Other criteria were systems that use human observation as inputs, and lastly systems that are naturally vague. IT projects and risk events and risk triggers matches these criteria.

When comparing the properties of project risk to that list for fuzzy logic, it matched on numerous points. Projects are systems that are controlled by human experts, the project managers. Secondly, there is a dependence upon human observation that a risk event has transitioned to a problem. There are cases where the risk events are naturally vague. A risk event is about a probability in the future that might not happen, and therefore conventional crisp logic has been applied. However, risk triggers are symptoms that an event has occurred to some degree. A fuzzy logic methodology would be able to measure to what extent a fuzzy risk event has been triggered. That the monitoring of risk can be inherently fuzzy, it would be appropriate to use tools and techniques that can capture the fuzziness associated with risk.

As a result of their findings that risky projects suffer from misperceptions and bias in reporting, Snow & Keil identified a need for better controls for understanding project status and ongoing risks (2001). They found that when a high risk project was in severe trouble, it was correctly perceived to be in that state only about 50% of the time. The remainder of the time the project status was misinterpreted as only having modest or no problems. As projects get riskier, the chances of being correctly reported using current methods goes down rapidly. They suggested that automated tools should be used to correct for the misinterpretations, but did not identify any specific tools or techniques that could accomplish that task.

3.4.8 Fuzzy Risk Methods

A literature search on the combination of the two terms "fuzzy" and "risk" found there have been studies on that subject. Those works were associated with the planning stages of projects and not the monitoring stage, or were related to non-IT types of projects. As examples, Xu, Khoshgoftaar, and Allen (2002) looked at fuzzy expert systems for operational, not project, risk assessment. Wat and Ngai (2001) studied risk analysis in electronic commerce using fuzzy sets. Li and Hu (2006) applied multilevel fuzzy tools to investment risk**,** and Kangari and Riggs (1989) investigated construction risk assessment through fuzzy linguistics. None of the works found were associated with the monitoring stages of projects, nor were they related to information technology.

It was noteworthy that no works on the combination of fuzzy risk and project scope were found in the literature search, which corresponds with a lack of fuzzy methods for monitoring scope. However, as will be explained in the discussion on feedback from the CISSE conference, recorded in the appendix under the subject of communications to practitioners, there was less interest in applying CI methods to risk. Therefore, the design artifacts, demonstration, and evaluation by subject matter experts did not include risk triggers. It is believed that this is an area for great potential, and will be suggested in the conclusion section of this dissertation.

Section 3.4 can be summarized as a literature review on the topic of risk and risk triggers for monitoring IT projects. Risk management involves both estimating the risks using probabilistic techniques, and the monitoring of triggers for risk events. Hancock pointed out that fuzzy risks with social aspects need to be included to increase the effectiveness of risk management, especially as projects become more complex. New techniques need to be found that go beyond expected monetary value (quantitative) methods for risks that are qualitative and subjective.

3.5 Design Science Research Methodology

March and Smith proposed a framework for design science as an information technology research methodology in 1995. Their invited paper in *Decision Support Systems* analyzed and compared natural science and design science, recognizing that both were important to progress in IT research (March & Smith, 1995). Natural science for IT traditionally took the form of descriptive research to create theories with support for them by gathering evidence and testing. They characterized natural science methods as using formal mathematical proofs, statistics, or well established behavioral science techniques to explain why or how IT works. They stated that the IT research at the time of the publication of their framework followed natural science and behavioral science methods.

Later works published in *IEEE Computer* reinforced March & Smith's ideas of a behavioral science component to IT by pointing out that behavioral science evolved as part of natural science, but with an emphasis on humans and organizations (Hevner & March, 2003). Hevner and March stated that natural science-based IT theories explained or predicted human behavior and interactions in organizations when computer technologies were involved. However, their paper in 2003 pointed out that a limitation of the behavioral science approach was that by demanding theories, it had the potential to focus on explaining existing outmoded technologies, rather than looking for imaginative IT solutions.

3.5.1 Design Science Defined

In establishing a contrast to natural science methods March and Smith (1995) described design science (DS) for IT as prescriptive research with the aim of applying knowledge. They pointed out that architecture, engineering, operations research, and management science were also prescriptive and problem solving. Information technology, like the other prescriptive fields, builds objects that add value or improve performance. The key characteristics are that a design science creates artifacts and that those artifacts have utility and are innovative (March & Smith, 1995).

After the introduction of design science for information technology by March and Smith, it quickly got subsumed by a larger audience in the computing field. It is now recognized as a generalized methodology for information technology, information systems, software engineering, and computer science. One example of the broad reach of design science in computing was demonstrated by a study that conducted a review of the state of the art in design research for the computing domain. The study looked at proceedings and publications that spanned from conferences on human interface design, to software engineering research conferences, to information systems researchers (Haynes, Carroll, Kannampallil, Xiao, & Bach, 2009). They found that design science has made an impact on academic research in the computer field.

Computing as a design science was originally proposed by Herbert A. Simon, who received the Association for Computing Machinery (ACM) Turing Award for contributions to artificial intelligence in 1975 (ACM, 2012), and who later won the Nobel Prize in Economics in 1978. Simon put forth the idea that there needed to be a science for the invention and design of objects and interfaces (Simon, 1996). Simon adopted the term "artifacts" to describe products of a design science. He stated that an advantage of creating artifacts or interfaces was the ability "to characterize the main properties of the system and its behavior without elaborating the detail of either the outer or inner environments" (Simon, 1996, p. 9). Wieringa (2010) clarified what might be meant by an artifact for software engineering by enumerating algorithms, techniques, methods, tools, notations, or conceptual frameworks.

March and Smith (1995) stated clearly that IT research must include both natural behavioral science and design science. Clarifying this idea they identified three areas where the two sciences interrelate. The first is that design science builds products that can be researched by natural science research. The second is that natural science creates knowledge to be utilized by the design sciences. That is to say that the "how" and "why" of natural/behavioral science feeds the creation of design artifacts. The third is that design science can provide a proof or test of the theories of natural research. In that they were defining a new set of methods for IT research, they stressed that it was not a replacement for the natural science methods, but rather a complement or alternative to it. Wieringa added to that by stating that after software engineering artifacts have been built through design science methods they could independently be evaluated or validated through natural science methods of empirical research (Wieringa, 2010).

Hevner and March (2003) described the difference from natural science for IT research as "design science addresses research through the building and evaluation of artifacts" (p. 112), while the behavioral and natural sciences developed theories and then justified them through formal proofs. They stated that a framework that included both approaches gave IT research synergies to move the IT field forward rapidly.

Building on Hevner and March's framework Peffers, Tuunanen, Rothenberger, and Chatterjee (2008) proposed a set of principles and procedures for design science research methodology that they believed integrated the best practices in common use. They identified six activities in a DS process that they found was shared by most researchers when they reviewed case studies in the computing field. The six steps will be expanded upon later in this dissertation. They argued that their analysis revealed the best practices for information systems DSRM. First of all they believed that their research represented the current state of published literature on design science. Next, their analysis itemized a formal process for research that would use design science. Lastly they provided a model for the evaluation of DS research in information technologies. One of their points was that until recently research in information technology and information systems was taken from descriptive social and natural sciences or interpretive behavioral science. They emphasized that design science solves problems by applying the theories, not by building and testing hypotheses.

Kuechler and Vaishnavi's (2012) knowledge representation framework for design science in information systems outlined three levels of relationships between IT theory and IT artifacts. In connecting the theory domain to the design domain, their first level was that theories directly connect to the artifacts. They stated that this relationship is most applicable for groundbreaking innovations where the artifact is a technical solution. Their second and third levels were what they termed mid-range theories, which contained an intermediary step when moving from theory to artifact. The second level would codify the design effort in an intermediate step as information system design theories (ISDT). Their third level would capture knowledge as design relevant explanatory predictive theories (DREPT) before progressing to an artifact. Both the second and third levels result in design artifacts after the interim step.

This dissertation connects the theory domain to the design domain directly, and as such is equivalent to Kuechler and Vaishnavi's first level. They stated the first level was applicable for cases where the artifact was an innovative solution implemented through technology. The artifact of this dissertation does not generate Kuechler and Vaishnavi's second and third levels. The output of this research corresponds to their concept of a theory domain directly connected to the design domain.

The design science research methodology has been endorsed by business administration researchers. The specialty field of accounting information systems, a field closely related to IT, has accepted the design science methodology as found by Geerts (2011). He reviewed recent accounting information systems research that implemented the design science research methodology. In many cases the papers reviewed did not cover all the DSRM activities set out by Peffers, et al., which Geerts considered an acceptable condition. He ascribed this to both the significant time it takes to perform each activity and to the need for very different skill sets to accomplish each activity (Geerts, 2011).

3.5.2 The Design Science Research Methodology Framework and Guidelines

March and Smith's (1995) framework for a design science research methodology (DSRM) had two dimensions. The first dimension was those research activities associated with design science and natural science. The activities for design were to *build* and *evaluate,* while the activities for natural science were to *theorize* and *justify*. These activities were described as being complementary, and not necessarily mutually exclusive. The *build* activity created artifacts simply to demonstrate what can be done. The *evaluation* activity took the artifacts that have been built and determined if they perform. These are both activities of design, while the natural science activities have been to create theories to explain, and then to prove the theories with evidence.

The second dimension of their framework was the four outputs of IT research: *constructs*, *models*, *methods*, and *instantiations* (March & Smith, 1995). March & Smith defined *constructs* as the concepts or vocabulary of IT, giving relational calculus as an example for database design. *Models* attempted to represent the reality of concepts, with mathematical models for proving the efficiency of data block accesses as the example for database technologies. The third research output was algorithms or *methods* to perform tasks. They stated that methods “transform user needs into system requirements” (March & Smith, 1995, p. 257). The last potential IT research output could be *instantiations* which demonstrated the feasibility of a model or method. The two dimensions of March and Smith's framework (1995, p. 255) for IT research are reproduced in Table 1.

	Design Science Build	**Design Science Evaluate**	**Natural Science Theorize**	**Natural Science Justify**
Constructs				
Models				
Methods				
Instantiation				

Table 1 The March and Smith Research Framework

The distinguishing characteristics of design science has been that the objects created should be both innovative and solve important problems. Selecting *constructs*, *models*, or *methods* would be sufficient for building a design in a design science-based research methodology. In speaking of design science was stated that "the research contribution lies in the novelty of the artifact and in the persuasiveness of the claims that it is effective. Actual performance evaluation is not required at this stage" (March & Smith, 1995, p. 260). Peffers, Tuunanen, Rothenberger, and Chatterjee (2008) later added that "outcomes from DS are clearly expected to differ from those of theory testing or interpretative research" (p. 52).

Other researchers published works that looked at the type of problems in information systems best suited for design science solutions (Hevner, March, Park, & Ram, 2004). They summarized their findings as the types of problems that have unstable requirements, problems that are not clearly defined, and with complex interactions of subcomponents. They believed that creativity would be required to produce solutions for the types of problems for which design science can be effective. They emphasized that designs must be driven by a business need, and the solutions have practical utility. In contrast to behavioral science based on empirical analysis, design science would build artifacts to meet those business needs.

Hevner, March, Park, and Ram put forth seven guidelines for using design science as a research methodology in information systems. Their guidelines were published in *MIS Quarterly* (2004, p. 83) and are summarized below in Table 2. They added that design science artifacts are not full-grown information systems, but the artifacts are innovative ideas or potential technologies.

Guideline	**Description**
1. Design as Artifact	Must produce a viable artifact in the form of a construct, a model, a method, or an instantiation
2. Problem Relevance	Develop technology based solutions to important and relevant business problems
3. Design Evaluation	The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods
4. Research Contributions	Must provide clear and verifiable contributions in the areas of the design artifact, design foundation, and/or design methodologies
5. Research Rigor	Rigorous methods should be applied in both the construction and evaluation of the design artifact
6. Design as Search process	Solution process is iterative and heuristic: search for a good design close to optimal. Artifacts that work well for a class of problems can "satisfice".
7. Communication of Research	Must be presented to both technology-oriented and to management-oriented audiences

Table 2 The Design Science Research Guidelines (Hevner, et al., 2004, p.83)

Peffers, Tuunanen, Rothenberger, and Chatterjee (2008) analyzed seven works on design science for IT, including March and Smith's framework. They integrated that framework with writings on IT design science by other researchers resulting in a list of six common activities of design research for information systems. Their goal was to define a process model that was consistent with and integrated those papers, resulting in six activities they held in common for IT research.

Their integrated methodology started with the need to identify an important problem that would be valued by the information systems profession. Their second activity would define the objectives for a solution to the problem. This would be based on what is realistic, and could be either quantitative or "qualitative, such as a description of how a new artifact is expected to support solutions to problems not hitherto addressed" (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008, p. 55). The third activity in their methodology would be the building of artifacts which could take the form of one of March and Smith's research outputs: *constructs*, *models*, *methods*, or i*nstantiations*.

The fourth and fifth steps go beyond the definition of the artifacts. These steps demonstrate and evaluate the artifacts. The fourth step, demonstration, establishes the utility of the artifacts in a business environment. The fifth activity, evaluation, provided feedback as to the success of the fourth activity, demonstration. The demonstration might take the form of a simulation, a survey, a client review, case studies, or experiments that were built to support the artifact. Their sixth activity would be to communicate the importance and novelty of the artifact to both researchers and professionals in computing. Their six activities (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008, pp. 52-55) are summarized in Table 3.

Activity	**Description**
1.	Problem identification and motivation
2.	Define the objectives for a solution
3.	Design and develop the artifact
4.	Demonstrate use of the artifact
5.	Evaluation as to how well the artifact performs
6.	Communication of the problem and its importance, the artifact, and its effectiveness to researchers and practicing professionals

Table 3 The Activities in the DSRM process (Peffers, et al., 2008)

Peffers, Tuunanen, Rothenberger, and Chatterjee (2008) stated that these six activities of design science are the equivalent of the empirical research process that follows a sequence of literature review, creating a hypothesis, testing, analyzing data, and then presenting results. The guidelines from earlier design science researchers were incorporated into this list of activities of a design science research methodology.

3.5.3 The Evaluation of Design Science Artifacts

The design science equivalent to the validation step of natural science methods is the evaluation of design artifacts. When discussing the evaluation of design, it has been pointed out that computer science is heavily oriented towards the fourth research output, *instantiation*, which is the delivery of working products (March & Smith, 1995). In the case of computer science research the evaluation of a design science *instantiation* is straightforward: does it work in an efficient manner.

March and Smith (1995) believed that the IT field needed all four outputs of research to make significant progress as a science. They went on to define evaluation criteria for the other three research outputs. Evaluation for *constructs* should look for completeness or simplicity. Their criterion for evaluating *models* is that they should correctly reflect the real world. Lastly, the evaluation of *methods* should be that the methods are operationally capable of doing the design task. The evaluation of the artifacts might require "obtaining a subject group to do the exercising" (March & Smith, 1995, p. 262). The goal should be to evaluate how well design artifacts met the design objectives and not how or why the artifact functions.

Although Peffers, Tuunanen, Rothenberger, and Chatterjee had a more comprehensive and integrated list of design science activities, they did not go into detail on how to evaluate design artifacts. But the earlier work by Hevner, March, Park, and Ram (2004) enumerated five alternative techniques to consider using when evaluating information technology design science. Their list started with *observation,* which included case studies and field studies. The second evaluation technique was *analytics,* composed of optimization, performance, or structural analysis. Third on their list was *experiments and simulations*, with the fourth alternative for evaluations being *tests* both functional (black box) and structural (white box). The fifth and last was the *descriptive* evaluation technique which could use arguments to convince. They also stated that *descriptive* evaluation could use *scenarios* built around the artifact to prove its utility. The selection of an evaluation technique depended upon the design of the artifact and the goal of the research. From Hevner, March, Park, and Ram's list this dissertation used *descriptive scenarios* for the evaluation of the design.

Pries-Heje, Baskerville, and Venable (2008) presented a similar interpretation of evaluation strategies for design science research in information systems. They separated the evaluation of information systems research artifacts into either *ex ante* (evaluation of the design) or *ex post* (evaluating the product). The construction of a functioning IT product in their view was not necessary for validation of a design. They differentiated between the *ex ante* perspective which is the evaluation of the design, and the *ex post* evaluation which would entail the evaluation of an instantiation or construction of functioning products. Using their definitions, the design itself is the artifact of the research process, and *ex ante* evaluation is the scrutiny of the design.

Under their definition of the *ex ante* approach a functioning IT product was not a requirement, nor was it necessary for validation of a design. Pries-Heje, et al., would postpone until an *ex post* instantiation the construction of a real product, ready to be deployed and installed to customers. With the *ex ante* approach an evaluation performed by a group of subject matter experts is sufficient for evaluation. This separation of instantiation from design fundamentally agreed with Hevner, March, Park, and Ram.

3.5.4 DSRM Compared with Qualitative Methods

Alternatives that were considered for this research, and investigated as part of the literature review, were qualitative research methodologies. Because the subject of this research is the collection of qualitative data, it at first appeared that qualitative methods might be applicable. Patton (2001) listed sixteen theoretical varieties of qualitative research, amongst which was included ethnography, positivist, phenomenology, and feminist inquiry. The majority of those identified by Patton were more applicable to the social sciences of psychology, anthropology, applied sociology and philosophy than to information technology. Systems theory and grounded theory were included in his list, which are both potential methodologies for IT. Janesick, in Denzin & Lincoln's Handbook of Qualitative Research, provided a list similar to Patton's with eighteen qualitative research strategies, most of which were identical to or a modest a substitution for Patton's items, such as replacing grounded theory with case studies (Janesick, 2000).

Neuman (2011) in discussing foundations of qualitative research set out three purposes for conducting qualitative studies. The first was exploratory research which can be used when the topic is new, unknown, or relatively unstudied. The result is to gather enough information to build a better set of research questions so that a more formal study can be undertaken. The second purpose of research was to describe a phenomenon. This type of descriptive research could produce a detailed report, a specific sequence of steps, a classification of findings, or document a situation. The third purpose was explanatory research that looked to answer "why" questions, or to identify the cause or reason for an event. The purpose of this dissertation could fall into Neuman's category of "exploratory", but this category did not encompass the goal of designing a solution using technologies. As a result, Neuman's categorization was insufficient to establish a methodology for conducting the research of this dissertation.

Grounded theory as described by Corbin and Strauss (2008) was a dynamic approach that was more flexible than the quantitative methods that are structured and use statistical sampling. They explained it as oriented towards words and meanings. In this regard, grounded theory seemed to fit the needs of this dissertation. According to Corbin and Strauss good grounded theory requires creativity, imagination, a humanistic bent, and the ability to live with ambiguity.

However, the goal of grounded theory was to develop theory, while the purpose of this dissertation was to create new designs to solve problems. For this reason design science was a better match, and case studies or grounded theory were not selected. However, the output of this research might lead to tools and techniques that can be utilized by qualitative researchers. This point will be identified in the conclusion section as a suggestion for future work.

3.5.5 Summary of the Literature for the Design Science Research Methodology

In summary, IT researchers needed a design science research methodology to meet the unique nature of the computing fields. Comparing to other research domains such as engineering and architecture, Simon (1996) conceptualized design as a research methodology for computing. March and Smith (1995) created a framework for IT research that would implement the design science methods. The second dimension of their framework defined four outputs of IT research: *constructs, models, methods,* and *instantiations*. Hevner, March, Park, and Ram (2004) added to the framework by detailing seven guidelines for using design science as a research methodology in information systems. Peffers, Tuunanen, Rothenberger, and Chatterjee (2008) analyzed writings on IT design science by other authors resulting in the identification of six activities for IT design science research.

Design science is a valid, accepted IT research methodology. It is used in this research to build and evaluate design artifacts. Design science allows for artifacts that are methods, and for this research those artifacts were computational intelligence methods to monitor IT project scope. The *descriptive* evaluation of design science was implemented as demonstration scenarios built around the artifact to prove its utility to subject matter experts. The descriptive evaluation has also been known as *ex ante* evaluation of the design. This meant that the design itself was the artifact of the research process, and *ex ante* evaluation would be the scrutiny of the design. The specifics of how design science was utilized in this research will be presented in detail in the next chapter.

3.6 Summary of the Literature Review

The goal of this research study was to determine if IT projects can be improved by monitoring subjective constraints using computational intelligence methods. In support of that goal the literature review spanned five areas of study. The first was the confirmation that the management of the scope constraint continues to be a major source of problems on IT projects. The second area was an examination of computational intelligence (CI) and how CI might provide solutions to subjective, fuzzy constraints. The third area in the literature review was the monitoring process on IT projects, in particular the widespread use of dashboards and stoplight reports. The fourth area looked at risk management as one means for monitoring a project during the execution phase. The last area in the literature review was an examination of the design science research methodology and how it is an established and accepted methodology for information technology research.

The literature review provided the background and basis for utilizing computational intelligence to design a solution for project scope issues. The artifacts that were created as an output of following a design science research methodology are described in the next chapter. The literature review identified six DSRM activities that were followed in this study. The fourth and fifth activities listed in the DSRM and implemented in this research were demonstration and evaluation. The evaluation by subject matter experts in chapter 5 will establish that additional data from CI methods can lead to improvements in IT project management.

Chapter 4 CI Applied to IT Project Scope

This chapter specifies how the activities that make up design science research were achieved through the creation of artifacts in the application of computational intelligence for IT project scope. When implementing design science research there are alternatives in the specifics for some of the steps of the methodology. The introduction to this chapter describes those alternatives and the selection process.

The detailed description of the implementation of the methodology is given in sections 4.2 through 4.7 in the execution of the six activities of DSRM that were identified in chapter 3. In the process of going through the DSRM activities the dissertation research questions are defined, a design of CI methods is presented, the demonstration of the artifacts is explained, and how the evaluation by subject matter experts was conducted is detailed. The results of that evaluation and the discussion is in chapter 6.

The output of the DSRM activities accomplished the fulfillment of the objective of determining the applicability of computational intelligence to monitor the status of information technology project management scope. The result is design artifacts as the product of the design science research methodology that utilize computational intelligence methods.

4.1 Implementing the Design Science Methodology

The literature review in the previous chapter went through the history of design science starting with Simon's proposals through to the enumeration of six activities of a DSRM approach by Peffers. et al. This section presents an outline of how those activities were accomplished in this study. Given a number of different alternatives from which to choose for each activity, this section describes the alternatives and justifies which of the alternatives were selected for implementation.

4.1.1 The Processes Chosen for Building of Artifacts

The March and Smith framework for IT research was used to define the high level research process for this dissertation. The two dimensions of March & Smith's framework for IT research (1995, p. 255) were reproduced earlier in Table 1. The building and evaluating of computational intelligence methods when they were applied to project management scope were the selections made for this dissertation. This research was an innovative application of CI to a significant issue in project management. The research product chosen to be built as artifacts was the design of methods using leading edge computational intelligence concepts. When mapping the substance of those objectives to the March and Smith framework, design science was determined to be the appropriate approach for a research methodology, and is juxtaposed to theorizing in natural science in Table 4.

	Design Science Build	Design Science Evaluate	Natural Science Theorize	Natural Science Justify
Constructs				
Models				
Methods	**CI-IT Scope**	**CI-IT Scope**		
Instantiation				

Table 4 Computational Intelligence for IT Scope (CI-IT) in the Research Framework

Peffers, et al., (2008) performed an analysis of works by multiple, diverse researchers on design science methodology which resulted in the integration of their findings into a comprehensive summarization of six common activities of IT design research. Their definition of the basic elements of an IT design science methodology established the principles of how to write hypotheses and the evidence required to support the conclusions. This dissertation implemented Peffers, Tuunanen, Rothenberger, and Chatterjee's methodology as six research activities. The six activities fit into the *build* and *evaluate* elements of the framework for design science defined by March and Smith, which was illustrated in Table 4.

The first activity defined by Peffers, et al., was to identify an important problem that would be valued by information systems professionals. This was done by a literature review of project management practices, which clearly established IT scope as source of costly project failures. Their second activity called for defining the objectives for a solution to the problem. This activity was accomplished by the description of new artifacts composed of CI methods to monitor IT scope. The measurement of scope was a problem that had not been addressed in the past because it was assumed that such a solution was impossible. CI methods and advances in computing systems have made this assumption dated and false.

The third activity in this dissertation's implementation of the design science methodology was the building of artifacts. Artifacts were selected from March and Smith's research outputs: *constructs*, *models*, *methods*, or *instantiations*. This dissertation chose to focus exclusively on creating computational intelligence *methods*. The CI methods and artifact creation are explained in detail in section 4.4.

The fourth and fifth activities of DSRM were to demonstrate and then to evaluate the artifacts. During the fourth activity, demonstrating the artifact, it was necessary to rebuild and modify those artifacts based on feedback. Peffers', et al, process model for DSRM assumed that artifacts could be improved, and therefore included a loop that allowed for iteration of earlier activities (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008). The final design of subjective data collection supported by computational intelligence methods represented the output of multiple iterations of design. The iterative process has been endorsed by IT design science researchers (Hevner, March, Park, & Ram, 2004 ; Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008), and was an integral part of this study.

The sixth step was to communicate the importance and novelty of the artifact to both IT researchers and professionals. This research has accomplished that by presenting the design artifact to both of those differing communities. Papers were presented at conferences representing the two different audiences: researchers in computational intelligence and project management professionals. Appendix A describes the conferences and the interest areas of the attendees. It has already been pointed out that Peffers', et al., process model for DSRM included multiple return loops that allowed for the iteration of earlier activities. Under their model the communications activity might be repeated multiple times, and after each repetition the process could loop back to any of the other activities. This was the case when feedback from the conferences was then incorporated in the building of improved an improved design. The evaluation of the artifacts by subject matter experts occurred after the communications activities took place.

4.1.2 The Processes Chosen for Evaluation of Artifacts

Hevner, March, Park, and Ram (2004) identified five evaluation methods that might be used for information technology design science: *observation, analytics, simulations, tests,* and *description*. These were identified and detailed in chapter 3 section 5. This dissertation chose the *descriptive* evaluation alternative because of the need to create a design for technical solutions. The *descriptive* evaluation was performed by a series of project management scenarios built around the artifacts to demonstrate utility and effectiveness. These demonstration scenarios were then evaluated by subject matter experts to validate the relevance of the artifacts.

In the March and Smith framework the evaluation ensures that the methods are operationally capable of doing the design task. Other criteria when evaluating a design method would include that the methods are efficient and easy to use (March & Smith, 1995). March and Smith stated that artifacts should be subject to a group evaluation that exercises the artifacts to determine if those criteria have been met. The use of experienced, certified project managers to evaluate the artifacts generated by this study met that criteria.

Hevner, March, Park, and Ram's guidelines for using design science as a research methodology in information systems research identified the third guideline as evaluation. They stated that a “design artifact is complete and effective when it satisfies the requirements and constraints of the problem it was meant to solve” (2004, p. 85). For the purposes of this dissertation this meant that the evaluation process asked the subject matter experts to only determine if the solution would accomplish the intended purpose, not that it was a perfect solution.

Denning (1997) stated that new ideas and innovations should be evaluated according to their originality and novelty. Since it is a subjective opinion as to what is original and novel, the judgments in this regard came from feedback the conferences at which the dissertation concepts were presented, and from the appraisals by the subject matter experts in the evaluation activity.

4.1.3 Summary of the Approach

The approach taken by this research follows precedent by implementing the six activities defined by Peffers, Tuunanen, Rothenberger, and Chatterjee for a design science research methodology. By implementing the six activities specified by Peffers, et al., this dissertation will have incorporated the *ex ante* approach described in the literature review where Pries-Heje, Baskerville, and Venable identified design as the artifact of the DSRM research process. Design science first proposed by Simon, formalized by Peffers, Tuunanen, Rothenberger, and Chatterjee, and endorsed by other researchers formed the methodology used in this dissertation.

The remainder of chapter 4 goes through an application of the six activities to the problem of monitoring scope on IT projects. The result is a design artifact output of the design science research methodology that utilizes computational intelligence methods.

4.2 Problem Identification

The first step in the design science research methodology (DSRM) was to identify a problem of importance to information technology professionals. The literature review found that managing scope on projects has historically been a significant problem in the IT field. This has not changed over time as shown recently by a quote in the Project Management Institute's *PM Network* July 2012 magazine. When asked "what is the biggest error project managers make?" Shailendra Kadre responded "improper scope management" (PMI, 2012, p. 24). As established in the literature review the need for scope management is a relevant and important issue to practitioners.

4.3 Research Objectives

The second step in the design science research methodology (DSRM) was to define the objectives that an artifact would need to meet. The detailed list is in section 4.3.1 in the form of the research questions that are the basis of this dissertation. This research was initiated with an objective of designing new artifacts to address problems in monitoring IT scope. In searching for a solution, one of the goals was to find a way that emerging technologies or technical approaches could be applied to address problems that were previously considered unsolvable by technology. The output of the DSRM process was artifacts that were innovative, applied new information technologies, and addressed an important and relevant problem.

Previous research on applying computational intelligence (CI) to project management primarily addressed two constraints of a typical project: the budget (Li, Moselhi, & Alkas, 2006) and the schedule (Wang & Hao, 2007). Those studies focused on the planning processes which are the early phases of a project where estimating skills could be improved by CI tools. For the reason that cost and schedule constraints have objective dimensions that can be quantified, traditional operations research methods have measured those dimensions in currency or elapsed time using proven techniques such as Earned Value or the Critical Path Method (Meredith & Mantel, 2009; Gido & Clements, 2009; Schwalbe, 2010b). Those studies that applied fuzzy logic or fuzzy linear programming did so as enhancements to the traditional techniques normally used in the planning and estimation phases.

This research considered a third constraint of projects, scope, in a manner that differed from previous research efforts in major regards. The first was that this dissertation shifted from applying computational intelligence in the early planning or estimating stages and moved to monitoring progress during the execution phase of an IT project. Secondly, this dissertation investigated how a subjective, intangible constraint such as scope can be monitored vis-a-vis the tangible constraints of schedule and budget. This was a notable shift from managing quantitative time and money to managing the qualitative aspects of scope. This research addressed the scope constraint which had not been investigated in previous research because of its subjective nature, and investigated that phase of project management where there has not been a measurement method for subjective data. In two more significant regards, with the establishment of a repository of subjective scope data, this dissertation explored how CI algorithms for verbal hedges can raise attention to problems on an activity, and how that repository of fuzzy data can act as a leading indicator of project success.

This research built design artifacts to establish the feasibility and applicability of computational intelligence to monitoring the status of information technology project management scope during the execution phase of a project. In that project managers, sponsors, and the responsible executives must make decisions regarding resource allocations during the course of a project, this research looked at whether computational intelligence methods can both assist in the assessment of a subjective constraint during the execution of a project, and improve the decisions made as a result of the scope measurements. The objectives of this research were:

- To design a process based on computational intelligence that would allow the recording of the subjective opinions of the status of the scope constraint for an activity on an information technology project.

- To investigate how data from computational intelligence methods can enhance IT project performance to determine:
    - Whether scope status can be captured for a task in progress by using the computational intelligence methods when that status is vague, imprecise, or unclear
    - Whether linguistic hedges provide an ability to elucidate the interpretation of the scope status for tasks on a project critical path
    - To explore how computational intelligence methods can be used to capture potential leading indicator data for the scope constraint on a project
- To gather Subject Matter Expert (SME) feedback to determine if the additional information provided by monitoring scope could have a positive impact on decisions when measured with CI methods

4.3.1 Research Questions

The research questions were formulated to define the objectives that the design artifact would need to meet. The evaluation of the artifacts as part of the design science research methodology will ensure that these questions were addressed. The literature review revealed that there were a number of facets to monitoring IT projects. This raised a series of research questions concerning IT projects which can be summarized in the overall question**:**

- If IT project scope is monitored for subjective, qualitative criteria using fuzzy computational intelligence methods, could this additional information have a positive impact?

This primary question is decomposed into the following research questions (RQ) to be answered by this dissertation:

RQ-1. What aspects of monitoring a project might benefit from computational intelligence methodologies? This is broken down into the following sub questions:

a. Is there a qualitative element to scope problems on IT projects that can be measured with CI methods?
b. What tools, techniques, or methods are in current use for subjective, non-quantitative constraints during the monitoring phase of a project?
c. Can CI methods produce leading indicators that contribute to project success?
d. Can linguistic hedges enhance the interpretation of data gathered using CI processes?

RQ-2. What might CI processes look like in order to capture subjective inputs? This entails the design of CI methods for and the establishment of CI operations on scope status.

RQ-3. Can CI methodologies capture a meaningful scope status for those cases when the scope is subjective? Can this be validated for a variety of alternative scenarios?

RQ-4 Do computational linguistic hedges enhance the understanding of the scope constraint for tasks on a project critical path?

RQ-5. If computational intelligence methods are applied to a project's scope, can the data gathered be used as a leading indicator?

The first two sub questions of research question one were addressed in the literature review. The third and fourth sub questions of RQ-1 are addressed both in the design and in the evaluation sections later in this dissertation. The second research question, RQ-2, is resolved in the design of the CI methods constructed in section 4.4. This encompasses both a design for data collection and a design of the backend processing by CI methods.

The remaining research questions will be addressed in the demonstration of the artifacts and the corresponding evaluation of those artifacts by subject matter experts. In general this chapter covers the demonstration of the artifacts, and the evaluation is addressed in the next chapter. Where the questions are answered is listed in Table 5, and further itemized in Table 6 in chapter 5 for RQ-3, RQ-4, and RQ-5.

Research Question	**Where the Research Question is Addressed**
RQ-1a RQ-1b	Addressed in Chapter 3 Literature Review
RQ-2	Addressed in section 4.4 Building Design Artifacts
RQ-3	Design in section 4.5.1; Evaluation in section 5.2.1 and in section 5.3 Discussion
RQ-4 RQ-1d	Design in section 4.5.2; Evaluation in section 5.2.2
RQ-5 RQ-1c	Design in section 4.5.3; Evaluation in section 5.2.3

Table 5 Addressing the Research Questions

The research questions that are the basis of this dissertation fulfilled the requirements of the second step in the design science research methodology. The remainder of this chapter covers the implementation of the methodology This research was initiated with an objective of designing new artifacts to address problems in monitoring IT scope. This was a transition from previous works that applied computational intelligence in the planning stages to monitoring progress during the execution phase of an IT project. Secondly, the shift from managing quantitative time and money to managing the qualitative aspects of scope was a significant difference from prior works. Third, with a database of subjective scope data this dissertation explored how CI algorithms for verbal hedges can raise attention to problems on an activity. Lastly, how that repository of fuzzy data can act as leading indicators of project success was investigated.

Research questions RQ-2 through RQ-5 were addressed by the design of artifacts to establish the feasibility and applicability of computational intelligence to monitoring the status of information technology project management scope during the execution phase of a project. The evaluation of the artifacts as part of the design science research methodology will ensure that these questions were answered.

4.4 Artifact Creation: Building a Design of CI Methods (RQ-2)

The third activity in the design science research methodology (DSRM) was to build artifacts that would meet the research objectives. March and Smith (1995) enumerated four potential artifacts as research outputs: *constructs, models*, *methods*, or *instantiations*. This research chose to create *methods*, specifically the design of computational intelligence methods to capture subjective inputs monitoring the status of scope on an IT project. The development of an instantiated tool was not a goal of this research. Instead the focus was on the design of methods for capturing the qualitative data, the composition of background processes that would enhance monitoring capabilities, and the validation of the methods by professional projects managers. Later steps in the DSRM process to demonstrate and evaluate the design artifact required the writing of a series of scenarios that exercised the CI methods.

An overview of where the new methods fit into existing project management processes is illustrated in Figure 6. Historically, project managers have monitored the schedule and budget constraints during the execution phase of a project. Objective data is input into the reporting system, after the activities have transpired. Both of those constraints have mathematically based methods, unlike scope which often is word based. The proposed new methods would add the ability to monitor scope by a project manager through the application of fuzzy computational intelligence methods. Three methods are defined: one to gather the fuzzy data when input, the second to process and store the data, and a third method to present a status for a project activity based on the data.

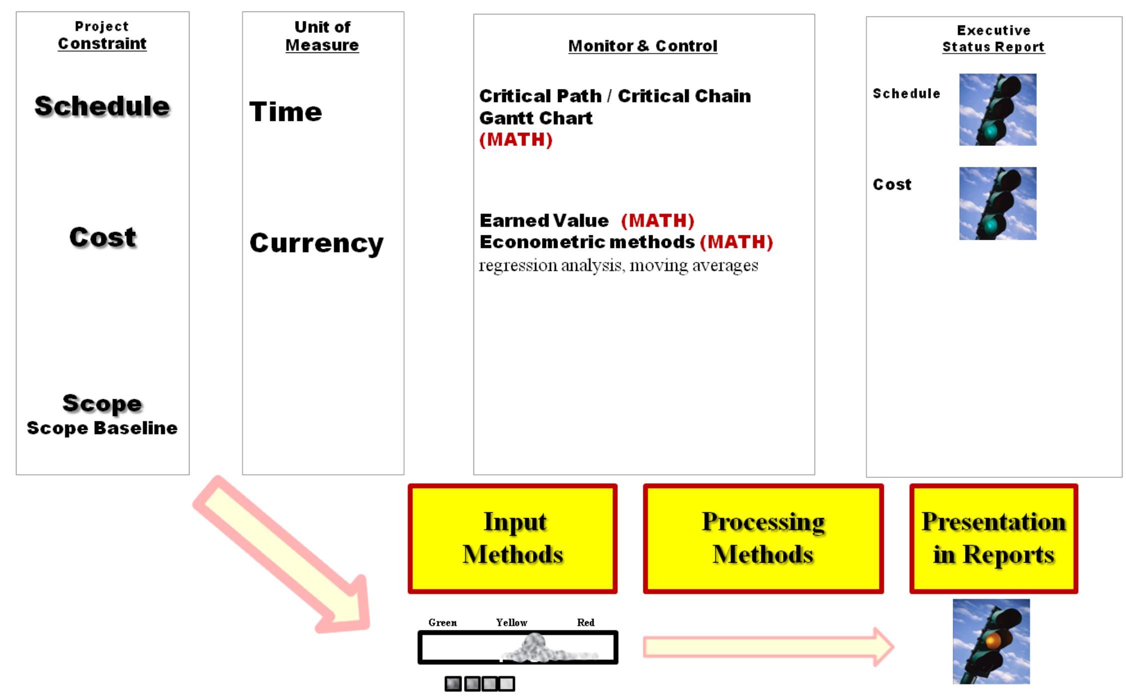


Figure 6 Conventional Processes to Monitor Projects with Scope Added

4.4.1 Design of CI Methods to Capture Subjective Inputs for Scope Status (RQ-2)

Kerzner (2011) emphasized the importance of metrics, but allowed that metrics might extend over ranges. The design artifact of this research expanded upon that conceptualization of metrics in combination with the recent publications and example instantiations by Barranquero and Guadarrama (2010) who prototyped Zadeh's fuzzy decision support concepts. They implemented a computerized web gadget which they call the Z-mouse, named in honor of Zadeh, to gather fuzzy opinions, or perceptions, from users. The design artifacts of this dissertation built upon their implementation by evaluating the fitness of the Z-mouse web gadget concepts when applied to project management. The scenarios developed in this stage of the research were based on the processes discovered in the literature review. Later in this dissertation, for the demonstration step of the DSRM, the data presented to project managers was illustrated as the rated scope for a WBS activity on a scale that was word based, not numeric.

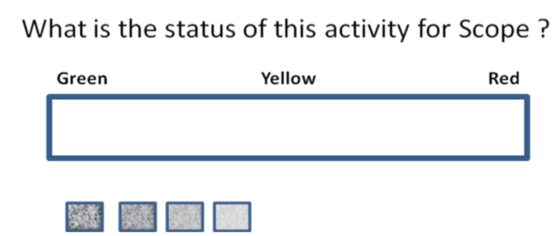


Figure 7 Web Gadget for Collecting Scope Status

The design of the non-numeric scale in the demonstration of the artifacts was such that it was easy to understand and use by experienced project managers. The design assumed that project managers should not have to have knowledge of how fuzzy logic works. For the purposes of this dissertation, the input gadget is illustrated in Figure 7 as a non-numeric, linguistic scale. One of the criteria for evaluating DSRM solutions was that any solution should be complete, and embody simplicity, elegance, and understandability (March & Smith, 1995). The simple stoplight method of reporting status was quickly recognized by the individuals who evaluated the artifacts of this dissertation.

Figure 8 illustrates the design of the web gadget artifact after being used to input data. The person responsible for reporting the scope status for an activity would use a computerized "spray paint" mechanism to input their impressions or judgments using the web gadget. These spray paint data points would be converted to numeric values and evaluated using mathematical rules of fuzzy sets. These numbers would then be associated with and stored in a repository or data base as part of an activity’s basic information. The benefit would be that the reported status, albeit subjective, is now a permanent part of a monitoring system. Using computational intelligence algorithms the status can be tracked, evaluated, and summed with the status of other activities.

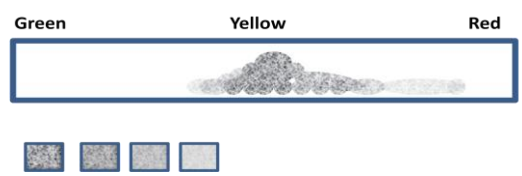


Figure 8 Scope status captured using the Web gadget

This numeric data collected by the web gadget will represent the status using words (green-yellow-red). Kerzner (2011) stated that "red" is the color used most often to indicate action is required or that something deserves attention. For this dissertation the word "red" was used in that context, and the color red was represented by a text label, and not the actual color. The use of text labels and grey scales was justified by Kerzner for two reasons. The first was that printing might later be done on black and white media, so the output product and photocopies will appear in grey scale not color. His point was that starting in black and white was the best way to ensure that any reproductions properly represent the intention of the original inputs.

Kerzner's second reason for using labels and not colors was that about 10% of the male population is color blind to some degree. Two types of colorblindness that are relevant to this dissertation are deuternopia where yellows and greens look alike, and protanopia where reds and yellows can be blurred (Kerzner, 2011). Since green, yellow, and red will be used in this study, the color challenge issue needed to be considered. For these reasons, the stoplight colors to represent status were presented as words, not actual colors. Tritanopia, another type of colorblindness where the greens are confused with blues, was not an issue in this research, but could be an issue if an instantiation used those colors.

4.4.2 Design of CI Methods for Backend Processing of Scope Status (RQ-2)

This section establishes the process flow to implement the CI methods in order to process the subjective data. The inspiration for the methods that are used in this design originated with Zimmermann's description of physical Mamdani fuzzy controllers (Zimmermann, 1996). Zimmermann's seven parameters of Mamdani systems were described in section 3.2 in the literature review. Instead of controlling physical devices like Mamdani systems, the artifacts of this research assist project managers in the monitoring and controlling of IT projects. For Mamdani controllers the input was electrical signals, whereas for monitoring project scope the input would be the subjective opinion as to the status of a task. A customized spray paint web gadget as described in the previous section replaces the electric signals, and becomes the input device to collect the subjective impressions or judgments.

Klir, St. Clair, and Yuan (1997) itemized a four step process for generalized fuzzy controllers. Wi, Oh, Mun, and Jung modified those computational processes for use in human resource oriented systems. This research is based on applying those methods to project scope management using a design similar to Wi, et al.'s. The steps are illustrated in Figure 9 in an adaptation of that process flow (Wi, Oh, Mun, & Jung, 2012, p. 50).

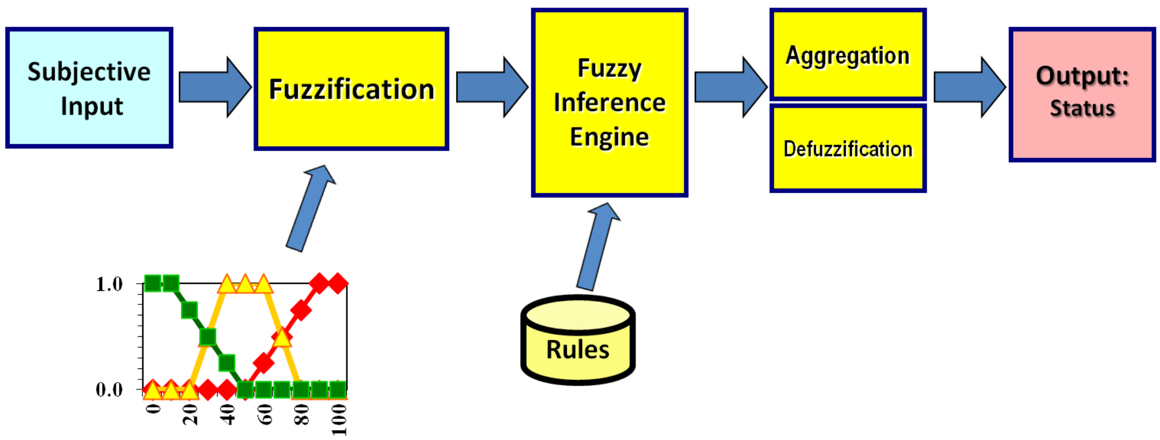


Figure 9 Steps in Fuzzy Control (Wi, et al., 2012, p. 50)

Under this process model the first step would be the fuzzification of inputs. This step takes the antecedent inputs and determines the degree of membership in the fuzzy sets for each. The fuzzy sets are illustrated in Figure 9 as inputs. The second step is the computation of consequents by an inference engine that utilizes predefined rules stored in a rule base. Klir, St. Clair, and Yuan (1997) described these rules as conditional propositions implemented as *fuzzy if-then* rules. They stated that only rules with positive measured values should fire. Zimmermann (1996) explained that fuzzy controllers contain rules that connect inputs to control outputs using linguistic terms. In this design the linguistic terms are fuzzy sets represented as triangles or trapezoids, shapes that Zimmermann recommended for fuzzy sets because they are computationally efficient.

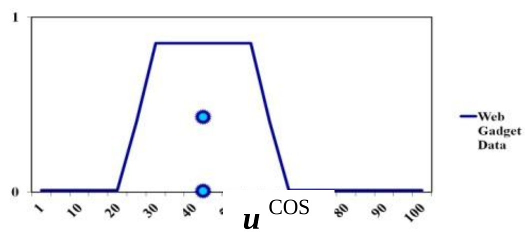
The third step illustrated in the process flow in Figure 9 is the aggregation of the individual consequents in a composition process resulting in an overall conclusion represented as one fuzzy set. For project scope monitoring that output fuzzy set is associated with one activity, which is one line item on a Gantt chart. The fourth and last step would be the defuzzification of the overall conclusion to an output crisp value, one value for each activity in a project. In the case of project status, the aggregation and defuzzification results in the reporting of a status for a given project activity to be used by a project manager to aid in understanding if actions are necessary to control a project.

In chapter 3 hedges were defined as capable of modifying fuzzy set conclusions. The use of hedges was not part of Zimmermann's original description of Mamdani controllers, nor were hedges part of the steps itemized by Klir, St. Clair, and Yuan. Hedges will be described later as to how and under what circumstances they will be used to modify the conclusions before the fourth step.

4.4.2.1 Fuzzy Processing CI Methods

Decomposing the fuzzy control process illustrated in Figure 9 into a deeper layer of detail, the subjective inputs in a project status monitoring system need to be reconciled to numerical values before they can be mapped to the fuzzy sets in the first step, fuzzification. This subjective data creates a new, alternative input mechanism from the web gadget to the fuzzification step. This is a predecessor step that would be done utilizing computing algorithms such as those investigated by Barranquero and Guadarrama. Their technique was to generate a best fit trapezoid overlay onto the end user inputs that were entered through the web gadget (Barranquero & Guadarrama, 2010).

From that trapezoidal overlay of the inputs a center of the trapezoid $\boldsymbol{u}^{\mathrm{COS}}$ is calculated to come up with a numeric value to be used as input to the fuzzification step. The center of the overlay can be calculated using the center of sums (COS) method provided by Zimmermann (1996, p. 213) as shown in Figure 10, where ***r*** is the rule that is being evaluated. This same method to determine the center of an area will be used later in the aggregation step. The trapezoid's center, $\boldsymbol{u}^{\mathrm{COS}}$, becomes the subjective input value to the first step in Figure 10, the fuzzification.



$$\boldsymbol{u}^{\cos} = \frac{\int_U u \;.\; \sum_r \mu_r^{conseq}\;(u)\,du}{\int_U \;\sum_r \mu_r^{conseq}\;(u)\,du}$$

Figure 10 Trapezoid and Center of Sums

For the first step, fuzzification, the input value from the web gadget would be mapped to the three fuzzy sets of project scope: green, yellow, and red. The fuzzy sets are shown as three trapezoids in Figure 11 only for illustration. In an implementation at a business entity the shapes of the fuzzy sets would be determined by the organization. In the example in Figure 11, what is illustrated is that for the subjective input value of 0.35 from the web gadget the membership in the fuzzy set green is 0.3, while the membership in yellow is 0.7. In this example, red has no membership, or a value of 0.0.

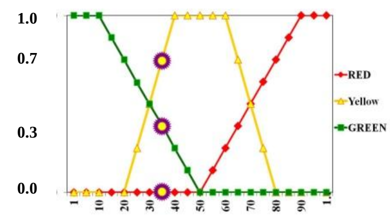


Figure 11 Example of Fuzzy Sets Membership

In the second step of the method the fuzzy inference engine would take the rules and apply them to the antecedents to determine the shape and size of the consequence fuzzy set. With the identification of the linguistic variables' degree of membership as input, the rules evaluation would compute the consequents. Theoretically, every rule in the rule base is fired each time for each input to create the overall output. The extent of the impact of a given rule is proportional to the membership value of an input to the defined fuzzy sets. However, most rules do not contribute at any given time, thus making the implementation more efficient. This was described by Zimmermann (1996) as a set of rules which connect the subjective input variables to the desired outputs. Li, Moselhi, and Alkass (2006) offered a simple example of a rule:

if *X is A* **then** *Z is C*

where *X* is the rule antecedent and *Z* is the consequent of a rule.

An example for project status would be the linguistic term "closeness" where if the subjective input is closer to a status of "red" then the reported status should be "red". For only one input, this simple case is obvious and the calculations are easy. When there are multiple people inputting a variety of subjective opinions, the rule processing becomes significant. For a given rule ***r*** the equation for determining antecedents would use the minimum operation:

$$\mu_r^{conseq}(u) = \min\{\alpha_r, \mu^j(u)\}$$

The third step in physical Mamdani fuzzy systems would be to take the consequents and aggregate into one combined fuzzy set, a principle followed by this study's design. The inputs to this step are the individual fuzzy sets from the inference engine, one fuzzy set consequent generated for each rule that fired. The aggregation of all consequents uses the maximum operation:

$$\mu^{conseq}(u) = \max_r \{ \mu_r^{conseq}(u) \}$$

The aggregated fuzzy set output would then be resolved into a crisp value in the fourth step, the defuzzification. In a system designed to monitor project scope the mathematical calculations would duplicate human decision making by aggregating the scope status from individual inputs into an overall scope status. The possibility of extending these methods into project risk management is described in the recommendations for future work.

In summary this research implemented the third activity in the design science research methodology (DSRM) in building artifacts that would meet the research objectives. *Methods* were chosen from the four alternatives for research outputs identified by March and Smith. These methods were detailed in this section in the design of computational intelligence methods to capture subjective inputs. This section also covered the design of methods to process that data using a fuzzy inference engine. The outputs from the methods will be used for monitoring the status of scope on an IT project. This design of methods for capturing the qualitative data, and the corresponding composition of background processes can be used to great benefit in project management decisions.

4.5 Exercising the Artifacts by Demonstration

The demonstration of design artifacts was the fourth activity identified in the DSRM (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008). Geerts (2011) expanded upon Peffers', et al., methodology definition by adding details on how the activities in the DSRM might be executed. For the fourth step of the DSRM Geerts stated that proving the DSRM artifact solves problems is a matter of demonstrating how to use the artifact. The demonstration was accomplished by the writing of a series of scenarios that exercised the CI methods. The evaluation of the design artifact was the next step after the demonstration, and is described in chapter 5. That evaluation of the scenarios was done by experienced project managers.

The scenarios were structured to determine if the artifacts answered the research questions concerning capturing a meaningful status for the scope constraint (RQ-3). This implementation of the fourth step in a design science research methodology, a demonstration, also investigated CI based linguistic hedges (RQ-4). This research presented the application of hedges to determine if there was value in modifying the perceived status in order to elevate problems to management attention under appropriate conditions. Included in the demonstration were scenarios to determine the benefit of using the data stored in the repository as leading indicators of project success or failure, thus addressing research question five (RQ-5). The scenarios are included in Appendix B. The first step in creating the scenarios was the creation of a design for the input graphical user interface. This design is described in the next sections, and was incorporated into the demonstration scenarios.

4.5.1 Interface Design to Demonstrate CI Methods for Status Collection (RQ-3)

When status reports are word based, Snow and Kiel's (2001) finding was that project managers tended to report an optimistic status and downplay problems thus losing critical information. To capture the subtleties and vagueness inherent in projects, this study used CI methods implemented as design artifacts of graphical user interfaces to capture scope status in a form that can be analyzed with computational intelligence.

To demonstrate that CI processes can capture a meaningful status for a task in progress for subjective scope a series of scenarios was developed. These demonstration scenarios for project tasks were evaluated by expert project managers as to the success of the CI in capturing the desired information. This research study encompassed scenarios that equated to a variety of stoplight style dashboard reports.

One scenario illustrated a status where all activities were meeting their objectives. For that scenario a written description was created to ensure that an experienced project manager would interpret the status as "green". An example of the graphical design artifact is shown in Figure 12 for a status of "green".

Scenarios were written to test a marginal or "yellow" status that can have three varying degrees. A status of yellow leaning toward green is shown in Figure 13, a status of pure yellow is shown in Figure 14, and yellow leaning toward red status is illustrated in Figure 15. Another scenario was designed such that the input should be construed as status "red" and is illustrated in Figure 16. A written description of the current status to be evaluated accompanied each of the screen captures of the data input using the CI web gadget.

Figure 12 Status Green

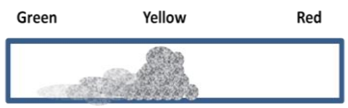


Figure 13 Yellow-Green

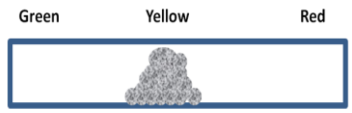


Figure 14 Status Yellow

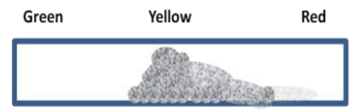


Figure 15 Yellow-Red

Figure 16 Status Red

Although the real possibilities are infinitely variable, these scenarios illustrated a small but representative subset of the universe of potential inputs. The scenario evaluation was designed to take an experienced project manager four hours to complete, therefore these scenarios were selected to test a reasonable, but comprehensive subset without exhausting the energies of the volunteer evaluators.

The evaluation package was constructed as follows. First the SME were asked to review conventional Gantt and earned value data, and comment upon the conventional stoplight reports associated with that data. Next they were invited to critique the design of the computational intelligence methods and comment on the value of the additional data provided by the design artifacts. These two sets were repeated for each of the six different scenarios. Each scenario addressed some aspect of the dissertation research questions. Last the SME were asked about issues related to the design science research methodology itself.

The scenarios in the demonstration were designed to address research questions RQ-3, RQ-4, and RQ-5. A detailed breakdown of the relationship between scenarios and research questions is illustrated in Table 6. The first section of each scenario was designed to resolve the research question of whether CI methods can capture a meaningful scope status for a task in progress when the scope is subjective (RQ-3). The evaluation package given to the subject matter experts was designed to have them respond to this research question.

Since errors in scope leads to errors in cost and schedule, the awareness of scope problems was expected to contribute to early corrective actions, increasing project success. When the CI methods collect and report the data it can be used to call attention to problems with the scope on an individual activity. This would give the project manager the opportunity to apply corrective action to that element of the activity that is the source of the problem. The determination as to the success of the design artifacts at meeting this goal was left to the subject matter experts and is detailed in chapter 5.

A decision might be made by a program management office to have more than one individual input data for a given activity. In that event there are well established CI methods to handle the possibility of multiple inputs. For example, a geometric centroid of the aggregated first moments could be calculated following established CI computing algorithms (Klir, St.Clair, & Yuan, 1997). For the purposes of this research it was assumed that only one person would input the opinions or subjective analysis of the activity status.

4.5.2 Interface Design Demonstrating Linguistic Hedges on a Critical Path (RQ-4)

The literature search on project management uncovered that activities on the critical path have more importance or urgency. By definition the critical path contains those activities that must be done to complete the project in the shortest possible time

(Schwalbe, 2010a). The faster that managers recognize issues with tasks on the critical path, the higher the probability that a satisfactory resolution can be found. If an activity on the critical path is delayed, then the project schedule is impacted. Many times this translates to increased costs in addition to schedule delays.

The design and evaluation of the artifacts from this research investigated whether escalation in the reporting of critical path activities with scope issues is useful to project managers. A "yellow" scope status might reasonably be adjudged as status "red" indicating the higher relative impact to the project if that activity is on the critical path. Since one of the contributing factors to project delays is scope issues, early recognition and response to scope problems on the critical path should increase project success.

Assuming that for activities on the time critical path the importance of scope issues for those activities is higher than for other, non-critical path activities, to adequately account for this linguistic qualifiers would be applied to modify the human determined status that was input using the fuzzy mouse. Klir, et al., (1997) described linguistic qualifiers as hedges that modify fuzzy values. In practice tasks on the critical path would be modified by an appropriate hedge such as "very" or "extremely", as determined by the project manager, thus elevating the status to management attention. For this research a single hedge of "very" was chosen as the linguistic qualifier.

Since the critical path can change during execution of a project, and the scope status might also change over time, the hedges would be reapplied on a regular basis onto critical path activities. The adjusted status for the critical path activities would be aggregated into an overall scope status for the project that reflects the relative importance of the critical path. The insights and feelings gathered for each activity using the CI methods would be modified by the reality of the critical path, leading to a meaningful status using computational intelligence hedges.

There were three components that were put in place for the scenarios to demonstrate linguistic hedges applied to a critical path. First was the algorithmic definitions of the computational hedges. Secondly, a project plan for each scenario with sample critical paths was created. The third was the establishment of a mechanism for gathering the subjective opinions on the status of the scope for the activities on the project. The normal use for a critical path is to manage the time constraint, and activities that exceed their allocated duration cause project problems. The purpose of the verbal hedge was to highlight those time issues caused by the scope constraint.

Since scope is the source of many project problems, adding information about scope to the critical path activities was expected to provide valuable insights. Gathering fuzzy status with CI methods should offer new opportunities for the project manager. The success of the CI methods in accomplishing this was a determination made by the subject matter experts in their evaluation of the artifacts.

Collecting the status by CI methods allows the project manager to adjust the interpretation of the scope status for an activity using CI hedges. A project manager could use the hedge "very important" to intensify the reported status for activities on the critical path, and to diminish the importance of reported status for activities not on the critical path with the "less" hedge. The ability to intensify or diminish by using a linguistic hedge is an established capability of fuzzy sets (Cox, 1999).

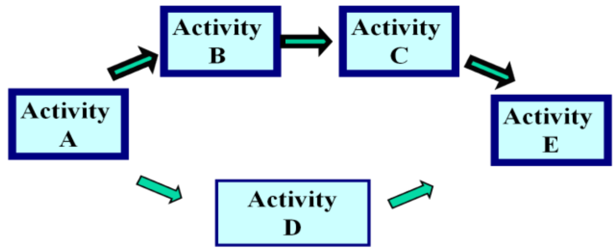


Figure 17 Sample Critical Path

Figure 17 contains an example of a project with only five activities shown to illustrate critical path concepts for this section of the dissertation. For the purposes of this example, the original time estimates would create a critical path that is defined as through activities A-B-C-E. As the project progresses through time, some activities complete such as Activity A, and other activities have not started, such as activities C and E. In this example the current time is in the middle of activity B and into the early beginnings of activity D as shown in Figure 18.

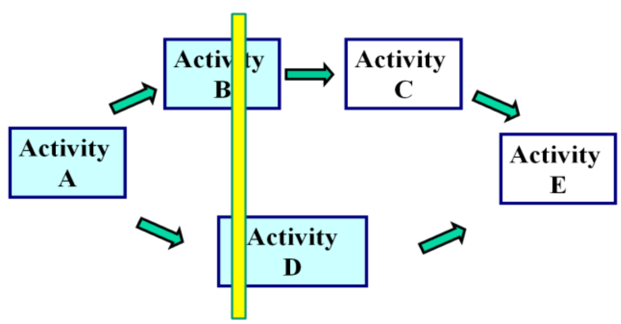


Figure 18 Current Time on Project for Activity B

For this example the assumptions are that an appropriate authority has defined and authorized the CI linguistic hedges, and that a scope status has been input with the fuzzy data collector. For activity B which is a partially completed critical path activity, a subjective opinion might report a scope status of "yellow" with a leaning toward a scope status of "red", as shown in Figure 19. Continuing the example, data from the critical path in Figure 18, and the subjective status for activity B as illustrated in Figure 19 would be the data input into the computational intelligence inference engine.

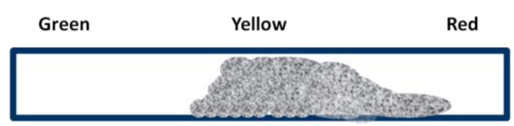


Figure 19 Sample Subjective Opinion for Activity B

With the hedge "very important" intensifying the reported status for activities on the critical path, the CI engine in the background might result in a status of "red" for an activity on the critical path which was originally status "yellow", as shown in Figure 20. The intention is to use the CI linguistic hedges to alter the meaning and thus raise attention to problems with the scope on a task.

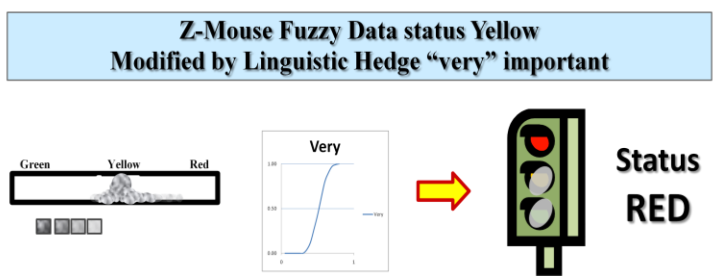


Figure 20 Applying Linguistic Hedges to the Critical Path

This would give the project manager the opportunity to apply corrective action to that element of the project task that is the source of the problem. Using conventional techniques, such as earned value, the problems with the scope would show up <u>after</u> time and money had been expended. With the CI methods the scope status can be captured as a subjective opinion much sooner, allowing for a quicker response by a project manager.

This research involved a much more complicated project plan, which is included in Appendix B as part of the evaluation scenarios used by the subject matter experts. The project plans shown above in Figure 17 and Figure 18 are for illustrative purposes only. At this stage of the design science approach the demonstration was created to resolve the fourth research question that asked if linguistic hedges enhanced understanding of the scope constraint for tasks on a project critical path.

4.5.3 Interface Design Demonstrating CI Methods for Leading Indicators (RQ-5)

For the time constraint using conventional tools an analysis of the critical path information available for a task is based on predecessor activities. A determination of the status of the critical path is derived from the time constraint, which for an task is "on schedule", "early", or "late". If a given activity has not started, there is no information other than that of the impact of previous activities. Budget management technique such as earned value only give insights to the financial status of an activity, after the funds have been spent. However, for large complex projects it is common to use rolling wave decomposition (PMI, 2008). This means that there are a set of tasks and activities that have not started, for which there are no requirements, and for which scope might not be defined. This process of progressively elaborating the scope as the project progresses is common in IT (Gido & Clements, 2009). The status of a future activity is unknown when using rolling wave decomposition.

Humans have the ability to speculate about the future, and for the scope constraint this means that people are able to report in advance that they have concerns about the scope of a future activity. This point was confirmed at the CONISAR 2011 conference when Dr. Schwalbe, a noted project management author, stated in an open forum discussion about bad projects that there are times when people knew that a project was going to be a failure from day one. This gut feeling or comfort level is the nebulous entity that was found to be needed to be captured for an early warning (Klakegg, Williams, Walker, Andersen, & Magnussen, 2010).

The mathematical principles for handling fuzzy data has been an area of research by computer scientists since the 1970's and should not be of concern for the end users of a project management reporting system. In talking about management systems, it has been stated that "the tool user, provided the tool is made well, need not, and indeed should not, know anything about the tool" (Drucker, 1993, p. 513). The concern of project managers and executives should be the identification of project activities with problems, not how a particular tool functions.

Since subjective opinions of individuals, and vague or contradictory inputs are precisely what fuzzy logic handles well, fuzzy systems can be applied to capture opinions as to the status of an activity not yet started. The scope status would be treated as a leading indicator that is analogous to cost and schedule measurements. This scope status would be resolved to useful information using fuzzy logic mathematics that have already been researched and refined.

The CI methods can be applied to capture exactly those opinions. For the purposes of illustrating an example scenario, a path would be defined as through activities A-B-C-E. In this scenario the current time would be in the middle of activity B and at the early beginning of activity D as shown in Figure 21, with activity C not started.

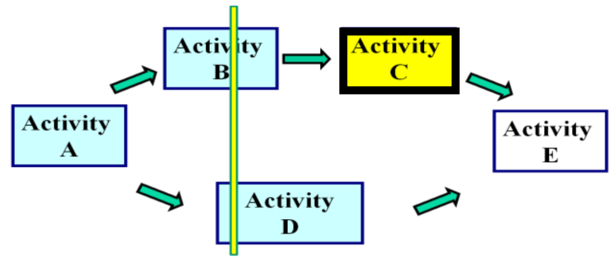


Figure 21 Path with Activity C not started

For this example assumptions have been made that a scope status has been input with the fuzzy data collector. For activity C which is waiting to start, a subjective opinion might report scope status of "yellow" with a leaning toward a scope status of "red", as shown in Figure 22. This is purely a subjective opinion, input into the web gadget, but based on a gut feeling as to the scope for this future activity.

Figure 22 Sample Subjective Opinion for future Activity C

For activity C the subjective data that is captured is recorded by the computational intelligence process. As already described earlier, conventional techniques expend resources then monitor, so that awareness of problems with scope would appear after time and money had been wasted. When the scope status can be captured as a subjective opinion before the activity has started, this raises an alert to problems with the scope constraint much sooner in the project life cycle. This gives the project manager the opportunity to apply corrective actions, and thus acts a leading indicator. Scenarios to exercise leading indicators were given to the subject matter experts for evaluation, and is included in Appendix B as part of the evaluation package.

With conventional methods, scope status was word based and recorded in reports that were subject to the interpretation of project managers. As already pointed out (Snow & Keil, 2002) project managers tend to be optimistic, and thus downplay the real status. By monitoring scope with computational methods, a new predictive capability can be added to the project assessment, rather than the retrospective monitoring that has been in common use. In the example above, the prediction that activity C has poorly defined scope, or has scope issues gives a project a predictive status that can be added to the conventional quantitative status reports.

An appropriate critical path was created in a demonstration scenario that was evaluated by the SME that collected data for activities that have not been started. The data was designed so that when computed by the CI engine the result for an activity's scope would yield a status that is less desirable than that collected. This determination of the CI methods was evaluated by project management experts. This research involved a more complicated project plan than the ones illustrated above. The project plan shown in Figure 21 with only five activities was solely intended as an example to explain the design concepts. For the evaluation of the CI methods the demonstration artifacts were expanded and enhanced to forty-six activities with overlapping durations. The purpose at this stage in the design science approach was to resolve the research question that sought to determine the feasibility of using CI methods to gather data that could be used as leading indicators when applied to project scope.

4.5.4 Summary of the Demonstration to Exercise the Artifacts

The fourth activity identified in the DSRM was to demonstrate the design artifacts (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008). Geerts (2011) stated that to prove the DSRM artifact solves problems demonstrate how to use the artifact. This study's demonstration was accomplished by the exercised CI methods in section 4.5 which described how the artifacts would function when utilized by project managers and the underlying CI processes.

The full demonstration is contained in Appendix B which involved more complex project plans than described in section 4.5. Appendix B contains the evaluation scenarios used by the subject matter experts, which were the complete and full demonstration package. The project plans shown above in Figure 17 and Figure 18 were for illustrative demonstration purposes with only five activities in a project. The demonstration package in Appendix B contained forty-six activities modified over six scenarios. Those activities were based on the artifacts in this section, and were critiqued by experts in project management.

The artifacts exercised in this section partially answered the research questions concerning capturing a meaningful status for the scope constraint (RQ-3), CI based linguistic hedges (RQ-4), and the leading indicators of project success (RQ-5). At this stage of the design science approach the demonstration was created to resolve part of the questions. A more complete breakdown as to which research questions are answered is identified in Table 6 Research Questions RQ-3, RQ-4, and RQ-5 Mapped to Scenarios found in section 5.2. The evaluation of the design artifacts in chapter 5 completes answering the research questions.

4.6 The Artifact Evaluation Process

This section discusses the fifth activity of the DSRM, which is to evaluate the design that was produced in the fourth activity. During the fourth activity, demonstrating the artifact, the design examples were illustrative, and not exhaustively detailed. The evaluation package that was used in the evaluation process contained a comprehensive set of six project scenarios, completely listed in Appendix B. This section describes the evaluation process, the selection process for the subject matter experts, and the iterative process that produced the final design. The evaluation activity itself is documented in this dissertation in chapter 5 along with the results and the discussion.

When discussing IT research Hevner and March (2003) distinguished design science from natural science by stating that "design science addresses research through the building and evaluation of artifacts". Peffers, Tuunanen, Rothenberger, and Chatterjee (2008) in their integrated best practices in design research listed the fifth activity in the design science research methodology (DSRM) as the evaluation of the artifact. The evaluation is closely related to the fourth activity of demonstrating the artifact. Whereas the demonstration in this dissertation is a design, the evaluation scrutinizes the design artifact. In keeping with March and Smith's concept that a subject group review the artifacts, this research recruited volunteer subject matter experts to critique the artifacts.

This dissertation chose the *descriptive* evaluation method from Hevner, et al.'s, (2004) five evaluation methods that can be used for information technology design science research. The *descriptive* evaluation was performed by a series of project management scenarios built around the design artifacts to illustrate the functionality and effectiveness. These scenarios were then critiqued by subject matter experts to confirm the relevance of the artifacts. In following the March and Smith framework (1995) the evaluation ensured that the CI methods were capable of meeting the design requirements. Other criteria in the evaluation of the design included that the methods were efficient and easy to use (March & Smith, 1995). The subject matter experts were queried as to their opinion of the human factors aspect of the CI methods in their evaluation of the design artifacts.

4.6.1 Iterations of the Design

The design science process is inherently iterative, with improved artifacts replacing older. This was a point that Hevner, March, Park, and Ram (2004) made in their sixth guideline to DSRM. This research went through two iterations before the final design. The first iteration in the design of the artifacts was during the literature review of the probability capture method by Zier and Inoue (2011). New understandings based on their findings resulted in adjustments to the design, which are explained in more detail later in this section. The second iteration of the design of the artifacts occurred as a result of the communicating the design at research conferences. Those conferences are documented in Appendix A. The remainder of this section explains the changes made to the artifacts as a result of the first iteration of the design.

The design of the web gadget for collecting scope status settled on a black and white fuzzy cloud for input. This modification came after reviewing Zier and Inoue's probability capture method. They investigated frontend visualization and devised two methods that used computational intelligence to capture probabilities across a two color scale. Their first method was a single point and their second method was an interval probability, both on a scale from red to green. Although it was not an issue in their research, it was observed that when moving across a red-to-green scale the middle of the scale was brown in color. This was a problem for the purposes of this research in project management where the middle of a red-green stoplight system is expected to be yellow, not brown. For this research the underlying design of the graphical interface was modified to go from green to yellow, and then from yellow to red. This essentially meant two scales, not one, but a design change that was imperceptible to end users.

A second insight from the literature review of the Zier and Inoue paper was that an interval method assumed that the physical center was the average of the endpoints. This worked for their capture of probabilities, but it was decided that for this dissertation the capturing of subjective opinions needed to allow for skewed, non-normal curves with fuzzy endpoints.

Finally, the online version of Zier and Inoue's paper was in color, but the printed copy was black and white (BW). Their pictures of color saturation and color brightness lost impact when presented in print in black-white scales. Again, not a problem for their purposes, but this raised the issue of a BW system losing the intended meaning of the data. The lessons learned were to use text labels and assume that color would not be available. This position was confirmed by Kerzner (2011). Kerzner also pointed out the problems with colorblind readers, and this was taken into consideration in the demonstration of the methods.

4.6.2 Evaluation of Artifacts by Subject Matter Experts

Subject matter experts (SME) were used to evaluate the artifacts, which fulfilled March and Smith's criteria that a subject group should exercise the artifacts (March &

Smith, 1995). The subject matter experts were given a set of project scenarios and then asked to evaluate the relevance of the artifacts in light of the research questions. The multiple project scenarios were related to each other as time snapshots of a project in progress. In the first section of each scenario the SME were asked to answer questions regarding the quality of information provided in conventional project management reports. This included an earned value report, a critical path Gantt chart, and a brief written description in a format that might be given to an executive.

After responding to that first set of questions the SME were asked to critique additional information in the form of the CI method inputs and outputs. They were shown the results of a CI method having processed the fuzzy inputs to determine the scope status for a given activity. The research gathered the SME opinions and observations of the impact of the additional information provided by monitoring scope. There were both structured questions and open ended questions to collect the SME feedback. The SME evaluations ensured that the CI methods were operationally capable of monitoring scope on information technology projects. The SME were also asked other questions relevant to evaluation of a design method to address the DSRM issue of whether the artifact methods were efficient and easy to use (March & Smith, 1995). The results of their evaluation of the artifacts is discussed in chapter 5.

4.6.3 Selection of Subject Matter Experts

The subject matter experts who participated in the evaluation were project managers in the information technology field. They were selected for this study based on their experience and certifications. The criteria for inclusion was that each of the selected experts: 1) had practical field experience managing IT projects, and 2) possessed the

Project Management Professional (PMP) certification. As part of the evaluation of the design artifacts, the SME were asked questions to gather background information on the their professional career and confirm their PMP status. These general survey questions can be found in Appendix B in the first section of the evaluation questions. The questions ranged from specific questions on their years of experience in project management to an open ended question on their views on scope creep on projects. The selected experts had experience in either commercial or government projects.

The PMP certification was a reasonable means to ensure basic knowledge on project management concepts, and ensured the ability to use the traditional tools and techniques considered best practices for managing projects. To obtain the best possible feedback on the research's design artifacts, the experts would need to understand concepts related to project management, and be able to use that knowledge to reach a decision on the status of an information technology project. The PMP criteria meant that the selected SMEs would have a common vocabulary for describing the state of a project, combined with an ability to read and interpret standard management tools such as Gantt charts and earned value reports. Twenty five percent of the items on the PMP certification exam covered the topics of monitoring and controlling a project. Therefore, the SME would understand what it meant to say that a project was in the monitoring phase, or the SME could correctly interpret an earned value report.

The general requirements for PMP certification are a four-year degree (bachelor’s) and at least three years of project management experience, with 4,500 hours leading and directing projects, 35 hours of project management education, and passing the PMP certification exam. A complete description of the requirements of PMP certification can be found at the PMI website: *www.pmi.org/en/Certification.aspx*. The PMP certification is accredited to the international standard ISO 17024. The subject matter experts who were invited to participate in the evaluation of this research came from the Baltimore and Washington area chapters of the Project Management Institute.

The participants were invited to join the evaluation on a strictly voluntary basis. Copies of the evaluation package, invitations to participate, advertisements, and emails were submitted to and approved by the Towson University Institutional Review Board (IRB) for the Protection of Human Participants. The principal investigator had no supervisory, nor any other type of control over the participants. Their participation was confidential in that the study was designed so that the participants were identifiable to the investigator, but their identity will not be revealed to anyone else. Their participation was confidential, but not anonymous, and with full disclosure to the participants. Each participant's evaluation was assigned a number that referenced that evaluation in all of the published documents. The principle investigator kept a log that associated the assigned number to the participant's identity. The log was not and will not be published. A copy of the log will be stored by Dr. Hammell at Towson University. A copy of the approval letter from the IRB for the use of human subjects is in Appendix C.

4.6.4 Summary of the Evaluation of Artifacts

The fifth activity in the design science research methodology (DSRM) was the evaluation of the artifacts. The combination of the demonstration of the design and the evaluation of the design artifacts completes answering the research questions. This research used the *descriptive* evaluation method of IT design science research (Hevner, March, Park, & Ram, 2004). The *descriptive* evaluation was performed by subject matter experts who critiqued a series of project management scenarios that demonstrated the design artifacts. Included in section 4.6 was a description of the background and experience of the subject matter experts who evaluated the artifacts.

4.7 The Communications Process

The sixth activity in the design science research methodology is communications. The common elements of design science processes published in the *Journal of Management Information Systems* introduced considerations for the communications activity in IT design science research (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008). It stated that the communications should disseminate "the problem and its importance, the artifact, its utility and novelty, the rigor of its design, and its effectiveness" (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008, p. 56).   The DSRM communications activity of this research was completed over two years as series of papers published in peer reviewed conference proceedings. Those papers investigated aspects of project monitoring that could benefit from computational intelligence. Highlights of those publications is summarized in this section and explored in more detail in Appendix A.

The first research question asked "what aspects of monitoring a project might benefit from CI methodologies." This was originally explored by investigating if CI methods would be valued for security risk management at CISSE 2010: the 14th Colloquium for Information Systems Security Education. The conclusion from that presentation at the short topic sessions and the corresponding feedback from the participants was that there was only minimal interest in applying CI methods to project risk management. It was decided that the emphasis of this dissertation needed to shift from risk to the issue of scope during the execution phase, where there was compelling literature documenting the problems and a lack of CI based solutions.

The submission to the 22nd Midwest Artificial Intelligence and Cognitive Science Conference (MAICS 2011) looked at research sub question 1.a "is there a qualitative element to scope problems on IT projects?" In contrast to CISSE 2010, the participants at MAICS 2011 responded enthusiastically about their personal experiences with scope problems and scope creep on IT projects. Discoveries made as part of MAICS 2011 lead to the inclusion in future papers of critical paths for scope in the context of project management scheduling. MAICS 2011 was heavily oriented towards computer science researchers involved in computational intelligence.

Work done as part of the poster presented at and the student paper submitted to the International Research Network on Organising by Project (IRNOP 2011) clarified the relationship of scope to other constraints. The highly subjective nature of scope was compared to objective budgets and schedules. (Note: the word "organizing" was spelled in the British manner "organising" that was in common use at that Canadian conference.) The publication at IRNOP addressed question 1.b "what tools, techniques, or methods are in current use for subjective, non-quantitative constraints during the monitoring phase of a project?" The conclusion was that the monitoring phase of a project was a promising area of research given the dearth of measurement techniques for handling non-quantitative data. IRNOP 2011 was composed of project management researchers and practitioners.

Question 1.c "can CI methods produce leading indicators that contribute to project success?" was explored in the paper submitted to SERA 2011, the 9th ACIS Conference on Software Engineering Research, Management, and Applications, with the conclusion that scope could act as a leading indicator of project success. A finding at SERA was that function points as an estimating technique could be considered as a leading indicator, but was found to only be applicable to traditional waterfall model project management. Kerzner explained this the best by stating that many complex projects have vague requirements where the rigid, highly defined waterfall approach would not work (Kerzner, 2011). Therefore, function points would not be usable. SERA 2011 was a conference of software engineers, some with an interest in project management research.

Lastly, sub question 1.d "can linguistic hedges enhance the interpretation of data gathered using CI processes?" was considered in a paper accepted for publication in the proceedings of the Conference on Information Systems Applied Research (CONISAR 2011). The discussions at CONISAR on linguistic hedges resulted in an outpouring of suggestions for this research, such as the merits of reporting scope trends over time. If scope data is time stamped and stored in a repository, then it was theorized that this could be reported in a time series to allow for trend analysis. This idea has merit, but has been postponed as a topic for future research and not a topic that was within the goals of this research. The presentation at CONISAR 2011 was in the project management track, which was composed of researchers and practitioners of project management.

The DSRM communications activity of this research was accomplished in the papers published in peer reviewed conference proceedings as described above. This met the requirement that the communications should disseminate the design artifacts, detailing its utility, novelty, and its effectiveness. Those papers investigated aspects of project monitoring that could benefit from computational intelligence. The communications and publications are accounted for in detail in Appendix A.

4.8 Summary of the Implementation

This chapter implemented the activities of the design science research methodology to create artifacts that applied computational intelligence to IT project scope. The detailed description of the DSRM steps was given in sections 4.2 through 4.7 in the execution of the six activities. The DSRM activities included the defining of the research questions, presenting a design of CI methods, and the demonstration and evaluation of design artifacts by subject matter experts. The implementation through the six activities created design artifacts that utilize computational intelligence methods applied to an important problem in project management.

The scenarios in the demonstration addressed research questions RQ-3, RQ-4, and RQ-5, which were then analyzed and critiqued by project management experts. The results of that evaluation and the discussion is in the next chapter. The communications of the effectiveness, utility, and rigor of the design was accomplished in peer reviewed conference proceedings described in section 4.7 and documented in Appendix A.

Chapter 5 Evaluation and Discussion

Peffers, et al., (2008) organized design science research into six activities, the fifth of which was the evaluation of artifacts. The *descriptive* evaluation method was the approach selected in this dissertation. The *descriptive* evaluation method was described as an alternative that is especially applicable to innovative artifacts (Hevner, March, Park, & Ram, 2004). The *descriptive* evaluation of this research was implemented as the critique of scenarios built to demonstrate the design. The evaluation process and a discussion of the results of the evaluation of this dissertation's artifacts are presented in this chapter.

5.1 Demonstration Scenarios Used in the Evaluation

The evaluation consisted of presenting demonstration scenarios to subject matter experts, asking them to determine the applicability of the CI methods to project management. The volunteer subject matter experts (SME) all possessed a certification from the Project Management Institute as Project Management Professionals (PMP). How the SMEs were screened and selected was explained in section 4.6.3.

The scenarios were derived from a time series set of snapshots of complex projects. The list of activities for the projects that these scenarios were based upon has been cut down significantly to focus on the issues of concern in this research study. Those snapshots have been modified to reflect a variety of project states that might occur ranging from being within acceptable limits (successful) to being in severe trouble. The original projects were much bigger in scale and number of people. These scenarios have been scaled down so that an analysis by the subject matter experts can be done in a reasonable amount of time.

5.1.1 Process Followed in Presenting the Project Management Scenarios

The scenarios for the evaluations presented a variety of cases that reflected the pertinent project status indicators that might be found when using the stoplight grading system (red, yellow, green). Also the design of the scenarios took into consideration that they needed to be compact enough that an experienced project manager could review them in a reasonable amount of time, considering that the SME were volunteers. A related, but conflicting goal to the goal of compactness was that the scenarios could not be overly simplistic. The scenarios needed to be complex enough to illustrate multiple potential critical paths and a variety of earned value metrics.

The time series snapshots used in the scenarios were for an IT project that included networking, hardware installations, and software development components. The scenarios were adapted and modified from original, real world projects. The demonstration scenarios covered a wider variety of alternative scenarios than is likely to be found in an actual project. To explain further, on an actual project the project manager might make corrections or adjustments that would change the course of the project to prevent later problems. Also, there is the possibility that an actual project might get cancelled if the project was in severe trouble. The purpose of these scenarios was to determine if the computational intelligence methods are useful across a spectrum of project states.

Rather than present the SME with seven different unrelated projects, it was decided that the learning curve would be shorter if the SME only had to view one continuous project composed of scenarios based on increasing time for that same project. This would mean that once the objectives of the project and the meaning of the activities were learned, they did not have to learn a new set of criteria for a new project. Other differences from actual projects were that the start date was 4 June 2012, and the scenarios started with the execution phase on 31 July 2012. The project charter and the planning and estimating phases were presumed to be completed.

The evaluation process that was followed was that the SME were given the scenarios in the time order sequence with just the budget and schedule data. The budget data was provided in the form of an earned value report. The schedule data was illustrated in a Gantt chart. Both sets of data were generated from the same Microsoft Project 2007 project plan for the scenario project for a given point in time. A summary of the status of selected tasks was provided in table format. Questions were asked about the status of those selected activities to focus attention on items of interest to this research. For the Gantt and earned value reports, the evaluation questioned the SME about the correctness of the stoplight reports when using conventional reporting methods.

Next the SME were shown the corresponding additional data from the CI methods. The next batch of questions had the SME provide evaluative feedback on the success or failure of the CI methods for that scenario. For instance, the SME looked at conventional budget and schedule reports for the scenario that indicated everything is going well. This would have been displayed as status "green" for those two criteria. The SME then received additional information from the CI method that indicated there was a problem with scope on that activity. This new information would be listed as representing the results of the computational intelligence engine evaluating the fuzzy web gadget inputs. The objective was to understand the SME’s decision-making process based on that additional data, and to gain insights into how the CI method might be used by a project manager. Overall, the intention of the project management scenarios was to provide an assortment of possible states that would allow for a complete evaluation of the web gadget system architecture and the CI methods by the subject matter experts.

5.1.2 Project Status Metrics

Each project management scenario gave the SME a written overview of the status of the project at a point in time, and metrics related to the constraints of schedule, cost, and scope. The schedule data was generated by MS Project using that product's Gantt chart view. For the schedule constraint the SME would have access to the activity start dates, the expected completion dates (or the actual completion dates), and the current date. For the cost constraint the SME would have basic earned value metrics for each of the activities provided in a report. The cost reports were generated from the same project in a standard MS Project report. Scope was the subject of this dissertation, so an extremely brief description of the scope status was provided as part of the written overview.

One of the common techniques for discovering scope problems has been to use earned value metrics. Earned value metrics point to symptoms of potential problems after the funds have been spent. Normally earned value is used in combination with verbal or written reports produced by the project team. For the evaluation of the artifacts the SME were given the earned value metrics and the Gantt chart. As a reminder to the SME, the meaning of the Cost Performance Index (CPI) metrics was included below the earned value tables. Using those sources of information the SME was asked to confirm the determination as to the status of those constraints using the stoplight grading system (red, yellow, green) for the selected activities.

Additional data for the scope constraint for those same activities would then be provided by the CI methods. The SME were asked two types of additional questions about scope in the evaluation of a given scenario. The first type of evaluation questions were designed to understand the SME's decision-making process based on the additional data provided by the new methods. The second type of evaluation questions were open ended questions to gain insight into how the method could be used, and especially looking for improvements to the methods.

Each scenario had one status color indicator assigned for each constraint for each activity in a project. The literature review pointed out that this seemed like creating artificial divisions (red vs. yellow) thus losing the subtle nuances of a complex problem, but the literature review revealed that an enormous number of projects use the stoplight grading system. As a reminder, the SME were given guideline definitions of performance for the color indicators which are: green = on target, yellow=fair, and red=poor (Schwalbe, 2010b). Also, the PMI definitions for the earned value metrics were placed at the bottom of the earned value reports as reminders for the SME.

Some of the activities in the scenarios were designed with missing information or data that was vague, in order to better understand the analysis process of the experts. The assumption was that project managers would recognize information is vague and/or missing for the scope constraint. The CI methods provided some measure of additional relevant data that would allow for a more accurate and meaningful project status. The complete scenarios used by the subject matter experts can be found in Appendix B. The evaluation consisted of presenting demonstration scenarios to the subject matter experts, and asking them to determine the applicability of the CI methods to project management.

5.1.3 Summary of Project Status Metrics

Project management scenarios were created that gave the SME a written overview of the status of the project at a point in time, and metrics related to the constraints of schedule, cost, and scope. The data was stored in MS Project and the reports were generated using that product's standard methods. The scenarios were derived from a time series set of snapshots of a complex project. The activities used in these scenarios were focused on the issues of concern in this research study. Those snapshots have been modified to reflect a variety of project states that might occur ranging from being within acceptable limits (successful) to being in severe trouble.

The series of snapshots used in the scenarios were for an IT project that included networking, hardware installations, and software development components taken from a real project. The focus of the SME evaluations were about scope in a given scenario. The first type of evaluation questions were designed to understand the SME's decision-making process based on the additional data provided by the new CI methods. The second type of evaluation questions were open ended questions to gain insight into how the method could be used, and especially looking for improvements to the methods.

5.2 The Evaluation by Subject Matter Experts

This section explains the evaluation package and the analysis and critique of that package by the experts. The discussion and conclusions drawn from their responses are supplied in section 5.3. The evaluation package given to the subject matter experts contained the demonstration scenarios of the design artifacts, questions to structure their responses as to the utility of the design, and questions to gather background information about the subject matter experts. The demonstration scenarios were devised to address research questions three, four, five, and partially address sub-questions of research question one. A series of questions were included in the evaluation package to confirm that the evaluation met DSRM criteria and guidelines. Five simple questions were included in the package to confirm the credentials of the subject matter experts.

It should be noted that the questions were intended to give structure to the responses by the subject matter experts, and that not all experts answered all the questions. In some cases some SME only gave one word answers such as "no" or did not respond at all. When the response of an SME was not clear, a follow-up by telephone or by email took place to clarify the intent of their response. The original responses and the follow-up conversations were recorded as typewritten documents in keeping with the need to provide "evidence of memos" (Corbin & Strauss, 2008). Grammatical errors and spelling mistakes were not corrected when directly quoting the SME. A copy of the responses will be stored by Dr. Hammell at Towson University.

The scenarios were constructed to establish that the design captured a scope status in a manner that would be useful to project managers. The questions associated with the six demonstration scenarios were built to give structure to the feedback from the subject matter experts to determine the success of the design in meeting the objectives. Each scenario had questions referencing line items in a project. Responses from the SME would contribute to answering some component of the research questions. Also the evaluation package addressed the issue of the adequacy of the design in terms in meeting DSRM guidelines.

How each of the scenarios are mapped to the evaluation of three of the dissertation research questions is shown in Table 6. In the previous chapter Table 5 in section 4.3 identified that research question RQ-1 was primarily addressed in the literature review. The design of the CI methods covered in section 4.4 addressed the issues raised by research question RQ-2. Research questions RQ-3, RQ-4, and RQ-5 are addressed in the scenarios and the evaluations in the next sections.

	Summary or Design meets DSRM guidelines and activities	Research Question Three	Research Question Three	Research Question Three	Research Questions Four and 1d	Research Questions Four and 1d	Research Questions Five and 1c
		capture a meaningful scope status	*capture a meaningful scope status*	*capture a meaningful scope status*	*linguistic hedges for Critical Path*	*linguistic hedges for Critical Path*	*leading indicators*
Evaluation Scenario-Question		confirm EV and Gantt	Scope same status as EV and Gantt	Scope different status from EV and Gantt	No impact to critical path	task on Critical Path	
Sc1-Q.1		x					
Sc1-Q.2			x				
Sc1-Q.3			x				
Sc1-Q.4	Summary						
Sc1-Q.5				x			
Sc2-Q.1		x					
Sc2-Q.2			x				
Sc2-Q.3				x			
Sc2-Q.4				x			
Sc3-Q.1		x					
Sc3-Q.2							x
Sc3-Q.3	Summary						
Sc3-Q.4							x
Sc4-Q.1		x					
Sc4-Q.2					x		
Sc4-Q.3						x	
Sc4-Q.4						x	
Sc5-Q.1		x					
Sc5-Q.2			x				
Sc5-Q.3				x			
Sc6-Q.1		x					
Sc6-Q.2					x		
Sc6-Q.3						x	
Q.7-1	x						
Q.7-2	x						
Q.7-3	x						
Q.7-4	x						
Q.7-5	x						
Q.7-6	x						
Q.7-7	x						
Q.7-8	x						
Q.7-9	x						

Table 6 Research Questions RQ-3, RQ-4, and RQ-5 Mapped to Scenarios

The full complete evaluation package was given to three PMP certified subject matter experts. Their appraisal of the design as presented in the demonstration scenarios is described in the next sections. One common comment from those evaluators was the length of time that it took to review and complete a thorough evaluation of the design artifacts that were demonstrated in the six scenarios. In order to gather more data about the design artifacts for the specific questions on leading indicators, an additional subject matter expert was recruited to evaluate just the leading indicator part of the design. Due to time constraints, SME-02 volunteered to participate in the research in a limited capacity, and therefore was only asked to review the scenarios that contained content on leading indicators. Their appraisal of the design in their shorter version of the evaluation package is included in the next sections along with the full evaluations.

The order of the topics presented to the evaluators was deliberately scattered throughout the evaluation package as shown in Table 6. The scenarios were evaluated by subject matter experts only to determine the adequacy of the design, in keeping with stated guidelines to implementing DSRM. The "principle aim is to determine how well an artifact works, not to theorize about or prove anything about why the artifact works" (Hevner, March, Park, & Ram, 2004, p. 88).

5.2.1 The SME Evaluation of Design Artifacts for Scope (RQ-3)

A portion of the demonstration scenarios were devised to address research question three which looked to establish if the methods could capture a meaningful scope status. The first question in each of the scenarios asked the subject matter experts about the status of the project at a point in time using conventional schedule and budget reports. These questions asked the SME to confirm the status as reported using the stoplight style dashboard interpretation of the data on traditional Gantt charts and the earned value reports. A quick summary of those questions is listed in Table 7.

It was expected that the SME would be in close agreement with the stoplight reporting based on those standard reports. These questions established the status that a project manager would normally receive using conventional methods. This became the baseline for later comparison to the new CI methods for the scope constraint. The correctness of these control reports was reviewed and verified by an individual who teaches project management at the graduate school level at a university.

The evaluation of the data by the subject matter experts in response to the first question in each scenario was their opinion of the meaning of the information in the Gantt and earned value reports compared to the stoplight report. For three questions, Sc1-Q.1, Sc3-Q.1, and Sc4-Q.1 the subject matter experts were in total agreement and confirmed that the status reported in the conventional reports was what they would report based on the information from the Gantt and earned value reports.

Scenario Questions	Cost and Schedule status
Sc1-Q.1	Start of project execution phase. Reported status for costs and schedules were green.
Sc2-Q.1	First week after the start of execution. Reported status for costs and schedules were green.
Sc3-Q.1	Confirmed that scope indicator agrees with EV and Gantt and therefore adds no value in these cases.
Sc4-Q.1	Reported status for costs and schedules were green.
Sc5-Q.1	Reported status for costs and schedules were green, except for activity 29 which was reported in Schedule as "yellow"
Sc6-Q.1	Activity 33 cost = yellow, schedule = yellow

Table 7 Evaluation Questions Related to Gantt and Earned Value Reports

For three questions some of the SME were moderately more aggressive in declaring the status of an activity. One subject matter expert, SME-01, believed that the cost constraint for Sc2-Q.1 should be reported as the next worst status. In responding to Sc2-Q.1 SME-04 also stated that they would report the cost status as the next worst case for activity 14, and for activity 28 would report the next worst case because that activity was on the critical path. SME-04 was also moderately more aggressive when looking at Sc5-Q.1, and Sc6-Q.1, suggesting that those should be reported as one grade worse.

SME-03 originally answered Sc1-Q.1, Sc3-Q.1, Sc4-Q.1, Sc5-Q.1, and Sc6-Q.1 as "yes", but without providing any details or reasoning. When asked about this in a follow-up conversation, SME-03 clarified that this represented responding after reading all six of the complete and entire scenarios including the scope data, rather than replying to just the conventional questions, and then continuing on to the scope monitoring questions. In other words, SME-03 had read all the scenarios and went back to answer the questions armed with new knowledge. In the follow-up conversation SME-03 agreed that those questions for just Gantt and earned value represented no change in reported status, and agreed that the response should have been a "no" answer to those questions.

Evaluation scenarios Sc1-Q.1, Sc2-Q.1, Sc3-Q.1, Sc4-Q.1, Sc5-Q.1, and Sc6-Q.1 establish the reported status from conventional methods in common use to monitor time and cost. The SME responses will be used for later comparison to the CI methods for scope. After reviewing the conventional data for a given scenario, the subject matter experts were given additional data on scope and asked about the information value of that new data to a project manager. This additional data was created to address research questions RQ-3, RQ-4, and RQ-5. For this part of the evaluation of the design artifacts the additional data was distributed between two types of scope information. The first type added scope data that contained no new information, and the second type added scope data that provided new information.

The first type of evaluation scenario designed to address research question three, as referenced in Sc1-Q.2, Sc1-Q.3, Sc2-Q.2 and Sc5-Q.2, was created *with little to no new information* concerning the status of an activity. The meaning of the new scope data provided to the SME did not vary significantly from the status established by using the conventional measurements of time and cost in the Gantt and earned value reports. This part of the evaluation was included to ensure that the artifact did not overshoot the intent of the data, meaning that it would report properly when there was no change in status. Sc6-Q.2 is similar in intent in that it also contains no new information concerning the status, but is related to the topic of critical paths and is included in the findings on research question four.

The second type of evaluation scenario designed to address research question three, referenced in Sc1-Q.5, Sc2-Q.3, Sc2-Q.4, and Sc5-Q.3, were created *to add new data* concerning the scope status. The data that was presented was a demonstration of data generated using CI methods, and contained additional information not found in the budget and schedule reports. The subject matter experts were asked if this additional scope data might alter their perception of the status of an activity.

The evaluation by the subject matter experts of the demonstration scenarios related to research question three is enumerated in the bullets below.

- SME all agreed that for Sc1-Q.2 the reported status was correct. For Sc1-Q.2 SME-01 replied that they would keep the same status. This was to be expected because the web gadgets showed status green in all cases, which matched the Gantt and EV reports.
- SME-01 stated that the information provided as part of Sc1-Q.3 would cause them to "evaluate why some team members feel that the scope is slipping" but that the status of "green" was correct. This was to be expected because the data from the CI method was only barely moving in the direction of yellow, but was predominately green status.

For Sc1-Q.3 SME-03 offered that based on this scope data a project manager should "monitor closely put contingencies in place" and felt the status should be changed to "yellow" because of the movement in that direction.

For Sc1-Q.3 SME-04 would also take actions, but not change the reported status. Those actions would include making entries into a risk register and initiating hiring the correct skills. SME-04 went on to say "I wouldn't change the status to Yellow, but the information is valuable to take early action to prevent it becoming yellow."

- For Sc1-Q.5 the opinion of SME-01 was that the CI methods should have reported a status of "yellow", which was not done.

  SME-03 used the written description of project problems, and would have changed the status for activity 28.

  For Sc1-Q.5 SME-04 stated that the written description in a status report would have necessitated an entry into a risk register, but did not need the CI methods.

- SME all agreed that for Sc2-Q.2 the reported status was okay. Their comments were that data provided in Sc2-Q.2 did not change the status of scope for scenario two. This was to be expected because the web gadget showed status green for activity 14, which matched the Gantt and EV reports.

- For question Sc2-Q.3 SME-01 stated that the information provided in the CI methods indicated concerns and potential problems and that "a PM could see that some team members are slightly uncomfortable with what may get done in that timeframe and take action to get everyone comfortable again." When asked if activity 21 should have been escalated to "yellow", the answer was "no". This was in agreement with one alternative response to the intended purpose of the question.

SME-03 in responding to scope data asked about in Sc2-Q.3 would change the status to "yellow" which was a reasonable alternative response from a manager with 15 years of experience who might be more aggressive.

For Sc2-Q.3 SME-04 stated that the status appeared to be "getting close to Yellow, but it's not there yet."

- SME all agreed that for Sc2-Q.4 the new data from the scope monitor changed their perception of the status. SME-01 stated for question Sc2-Q.4 the CI method provided data that would cause the status of the entire project to change to a status worse than first indicated. SME-01 added that this was new information that could not be found on the Gantt and EV reports.

  For Sc2-Q.4 SME-04 stated that this "definitely needs to be logged and tracked as a project risk."
- SME all agreed that for Sc5-Q.2 the reported status was okay. SME-01 stated that data provided in Sc5-Q.2 did not change the status of scope for scenario five. SME-03and SME-04 stated that no changes should be made. This was to be expected because the web gadget showed status green for activities 22, 29, and 31 which matched the Gantt and EV reports.
- SME all agreed that for Sc5-Q.3 the scope status pointed out that scope creep was occurring, a situation which was not evident in the Gantt and EV reports. SME-01 confirmed the result of the CI method, and would have the status for activity 15 be "yellow", and at the same time raise concerns about the status of the overall project. SME-03 stated that as a result of the new information action should be taken to "negotiate to get function into next release".

For Sc5-Q.3 SME-04 would take action to defer the additional scope, and thus delete the creep from the project, or alter the time constraint by schedule relief.

The analysis and discussion of these responses is presented in the discussion and conclusions for RQ-3 in section 5.3.

5.2.2 The SME Evaluation of Design Artifacts for Linguistic Hedges (RQ-4)

Research question four asked if computational linguistic hedges could increase the understanding of the scope constraint for tasks when those tasks were on the project critical path. This was related to the literature review discoveries for question 1.d which also concerned the project critical path. Question 1.d was specifically covered in presentations at, and feedback from, the conference proceedings listed in Appendix A. The evaluation of the demonstration scenarios related to research question four by the subject matter experts was:

- SME all agreed that for Sc4-Q.2 there was no new information. The data provided in Sc4-Q.2 did not change the status of scenario four for these critical path activities, so the SME expected that the CI methods should show status green for activities 15, 29, and 31, to match the Gantt and EV reports.
- SME all agreed that for Sc4-Q.3 there was significant, new information provided by the CI methods. In general the expert evaluators concurred that the information provided by the CI methods asked about in Sc4-Q.3 would cause a PM to take action. SME-01 would use this information to "determine if he needs to add another resource to the project." SME-01 would not change the overall project status because it was only 2 days into a 20 day long activity. SME-01 found this was an example of getting

extra information not found in the other reports which caused extra vigilance, but did not change the overall status.

The action that SME-03 stated a project manager should take would be to "put contingencies in place for code build not working".

For Sc4-Q.3 SME-04 stated that in addition to upgrading the status to "yellow", the project manager should review the additional functionality and assess the impact.

- SME all agreed that Sc4-Q.4 provided new information not received in the Gantt and earned value reports.

  SME-01 and SME-04 both felt that when looking at Sc4-Q.4 even though activity 22 was on the Time Critical Path, that the status should not change to "red". SME-01 believed that activity 22 should be left at "yellow" because there was still 18 days left on the activity in progress. Likewise SME-04 would not change the stoplight report, but give the team members one more day, and then review the status again.

  SME-03 in responding to Sc4-Q.4 would change the status to "red" because activity 22 was on the Time Critical Path.

- SME all agreed that for Sc6-Q.2 the reported status was correct for activities 15, 22, 29, and 31. SME-01 stated, with SME-03 and SME-04 concurring, that Sc6-Q.2 gave no new information for scenario six for those activities. This was to be expected because the web gadget showed status green for activities 15, 22, 29, and 31, which matched the Gantt and EV reports. Related to Sc6-Q.2 is activity 33 which was evaluated under Sc6-Q.3.

- SME all agreed that for Sc6-Q.3 the CI methods were correct in escalating the reported status. Sc6-Q.3 was created in order to test the CI methods for activities on the critical path for the case where an activity was "yellow leaning to red" status. SME-01 affirmed that the scope data from the CI methods was new, additional, and important information. SME-01 stated that for activity 33 this new information should cause the status of that activity to change to "Red as a whole".

  SME-03 would not change the cost, but would escalate the scope status to "red" and use the information to "escalate issue to delay other project start or get more resources to remain on schedule".

  For Sc6-Q.3 SME-04 stated that a project manager should use the new information to initiate action "to manage the impacts and transition."

The analysis and discussion of these responses is presented in the discussion and conclusions for RQ-4 in section 5.3.

5.2.3 The SME Evaluation of Design Artifacts for Leading Indicators (RQ-5)

Research question five looked at whether computational methods could gather data that would provide leading indicators for the scope constraint. This was related to the literature review investigation of research question 1.c which also concerned leading indicators. Question 1.c was addressed in presentations at, and feedback from, the conference proceedings, which are detailed in Appendix A. The subject matter experts provided the following in their evaluation of the demonstration scenarios:

- The SME stated that Sc3-Q.2 gave new information for activity 31, with the remaining activities staying unchanged. The CI methods reported status "green" for activities 15, 22, 28, and 29, which matched the Gantt and EV reports. The CI methods reported a status of "red" for activity 31, but SME-01 felt that because there was 2 weeks before the start of that activity there was sufficient time to for a PM to make adjustments to the project.

SME-02 combined answers to Sc3-Q.1 and Sc3-Q.2 and replied that "the Gantt and earned value reports alone don't give the complete picture. Those reports alone don't give visibility to the problem on the horizon in task 31." SME-02 went on to say that the "new fuzzy scope status graphic, along with the (written) explanation for the scope status, provides insights that might otherwise go unreported."

SME-03 would leave the status at "red" per the scope monitoring CI methods.

For Sc3-Q.2 SME-04 stated that "clearly there is a major issue" and this task should be red.

- The SME all agreed that Sc3-Q.4 provided new information about an activity that was two weeks away from starting. The consensus was that is gave the project manager the opportunity to take corrective action for activity 31. Question Sc3-Q.4 was a follow on question to question Sc3-Q.2. It was intended to confirm any responses to and allow for the clarification of comments on Sc3-Q.2 on the potential for leading indicators.

  SME-01 stated that a project manager might use the new information to change activity 31 status to "yellow". SME-01 felt that the CI methods would aid in managing the project by providing new information, but that a project manager might override the stoplight interpretation from an activity if the project manager has time to implement changes to mitigate the problems.

  SME-02 would initiate corrective action and "hire/outsource a translator".

SME-03 would use the information to initiate corrective action, "investigate and learn about the language issue and add the tasks to the plan".

For Sc3-Q.4 SME-04 pointed out that the project scope had to be adjusted, which is a corrective action.

The analysis and discussion of these responses is presented in the discussion and conclusions for RQ-5 in section 5.3.

5.2.4 The SME Evaluation of Design Artifacts for Monitoring Subjective Scope

The five research questions were summarized in an overall question on the impact additional data collected by fuzzy computational intelligence methods had when used to monitor subjective IT project scope. This overall question was directly addressed by the subject matter experts in responding to Sc1-Q.4 and Sc3-Q.3 in their evaluation. Their responses in their evaluations of the other scenarios contributed to the resolution of the primary research question and will be covered in the discussion section.

- SME-01 in responding to Sc1-Q.4 stated that "seeing the range of the data" helped in making a decision. For Sc1-Q.4 SME-04 stated that the CI methods gave "more nuanced information," which they would use to take action to prevent a task from getting off track.
- For Sc3-Q.3 the experts doing the evaluation agreed that the computational methods for monitoring scope status can aid in managing projects. In responding to scenario three Sc3-Q.3 the utility of the CI methods was confirmed with SME-01, SME-02, SME-03, and SME-04 responding positively that the CI methods could help a project manager. SME-04 added that "more information and sooner is always valuable" because it gives the project manager more time to make adjustments.

5.2.5 The SME Evaluation of Design Artifacts for Meeting DSRM Criteria

At the end of the evaluation package, Q.7-1 through Q.7-9, were included to determine if the artifacts and the demonstration scenarios met published DSRM guidelines (Hevner, March, Park, & Ram, 2004), and if this dissertation fulfilled the methodological processes for information technology DSRM (Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008) (March & Smith, 1995). The statements by the subject matter experts concerning their evaluation of the artifacts for meeting DSRM guidelines two, three, and six were:

- Question Sc1-Q.4 queried the SME if the visual inputs from the CI methods would aid in managing a project. This was an opportunity for the expert evaluators to explain if a project manager could use the CI methods in unintended ways, or if there were shortcomings in the design. The results of the responses to this step in the evaluation of the artifacts helped to decide that the research did not need to reiterate the design step in the DSRM framework.

  In response to question Sc1-Q.4, SME-01 stated that a project manager would want to see the raw data from the web gadget that was input to the CI method.

  For Sc1-Q.4 SME-04 stated that "by giving more nuanced information, you can take action to prevent a task from getting off track."

- Q.7-1 asked about the importance of monitoring of scope. The SME responses showed that they all agreed that scope is important. SME-04 stated that monitoring scope was "critical. It is nearly impossible to succeed and to successfully manage cost and schedule if you are trying to hit a moving target."

- Q.7-2 was an open ended follow-up question to Q.7-1 asking about the importance of monitoring scope. SME-01 indicated that customer satisfaction depended upon scope, which confirmed that the design artifacts were treating issues of significance to the recipients of information technologies.
- Q.7-3 simply asked the SME about their experiences with scope creep**.** All the SME indicated that they have had issues with scope creep. SME-04 went on to say that they had observed "scope creep that ultimately made the project technically infeasible."
- Q.7-4 asked about other options for monitoring scope status. SME-01 indicated that a WBS could be used to monitor scope, which is the conventional approach.

  For Q.7-4 SME-04 added that "sufficient detail in WBS to limit each line item, and thus scope changes require new lines in the WBS & schedule." SME-04 went on to emphasize change management processes that are clear to the project team.
- Q.7-5 asked if improvements could be made to the design of the CI methods by asking for ideas on changes to the scope monitor. This question and questions Q.7-6 and Q.7-8 were included in order to acquire new ideas for potential revisions of the design.

  SME-01's response asked if it was possible to see "clear identification" of the "opinions that comprise the sample." This response was confusing, so follow-up conversations clarified that what SME-01 wanted was to be able to look at the visual data that was input, especially if more than one person was using the web gadget for a given activity. This also helped clarify SME-01's response to Q.7-6 because SME-01 wanted to know who exactly had expressed an opinion.

For Q.7-5 SME-04 requested a "real time warning when a shift towards yellow or red is beginning, rather than simply point in time when a report is run."

- Q.7-6 was an open ended follow-up to Q.7-5.

  SME-01's response asking for the "weight" was confusing as to what was meant by "weight". A conversation with SME-01 helped clarify the response to Q.7-6. SME-01 wanted to know who exactly had expressed an opinion, because in SME-01's perspective some individuals had opinions that were more valid or relevant, and therefore held "more weight."

  SME-04 reiterated a desire to get "even earlier notification of potential problems."

- In responding to Q.7-7 all the SME agreed that using the CI method for scope would cost only a modest investment in time to gather the data. The only potential issue was SME-04 was not sure about the willingness of team members to take the extra time to "to do one more administrative thing."

- Q.7-8 was another open ended question, similar to Q.7-5 and Q.7-6, in that it asked what improvements could be made.

  SME-01 wanted to know the number of people inputting data. Follow-up conversations clarified that SME-01 wanted to be able to look at all the data that was input, so if multiple people were inputting data using the CI methods for a given activity, SME-01 wanted to know who the people were, by extension how many people input data. This response by SME-01 was related to their ideas in Q.7-6.

  For Q.7-8 SME-04 added there were different means to gather the CI method data giving examples "that explanations were delivered by the tool, or as a result of investigation by the project manager."

- Q.7-9 ensured that monitoring scope was an important concern to project managers. In response to Q.7-9 SME-01 wanted all the team members to feel confident that the scope objectives could be met, that the team members should feel like the "project was progressing as planned and meeting the scope was achievable."

  SME-04 hoped that they would learn about scope "as soon as possible and in time to intervene."

The analysis of statements by the subject matter experts concerning their evaluation of the artifacts for meeting DSRM guidelines two, three, and six is presented in section 5.3.

5.2.6 Summary of the SME Evaluation of Design Artifacts

Section 5.2 addressed the research questions through the SME evaluation of the design artifacts. The demonstration scenarios were devised to address research questions three, four, five, and sub-questions of research question one. A series of questions were included in the evaluation package to confirm that the evaluation met DSRM criteria and guidelines. Questions were included in the evaluation package to establish the credentials of the subject matter experts. Section 5.2 contained the evaluations, critiques, and responses of the SME to the design artifacts. An analysis and discussion of their responses is in the next section.

5.3 Discussion

The evaluation of the demonstration scenarios by subject matter experts established that the CI methods can capture scope status in a manner that would be useful to project managers. How each of the evaluation scenarios were mapped to the dissertation research questions was shown in Table 6 in section 5.2. The evaluation of the demonstration scenarios for using CI methods to monitoring scope was positive. The evaluation of the demonstration scenarios for using CI linguistic hedges for tasks on the critical path was generally positive, but less definitive. The evaluation of the demonstration scenarios for using CI methods as a leading indicator of scope issues was commendatory.

The evaluation package also included additional questions related to the design science research methodology. The SME evaluation of the artifacts also established that this dissertation met DSRM guidelines. The design artifacts addressed the important and relevant problem of monitoring scope on IT projects. The demonstration of the design was found to be practical to implement. The search process was iterative and heuristic, producing a solution that satisfactorily resolves the research questions.

5.3.1 Discussion of Traditional Reports Used for Comparison (RQ-3)

The SME acting as evaluators of the artifacts confirmed that the computational intelligence methods demonstrated in the design would be a helpful tool to project managers. To establish a control against which to compare the design, the SME responses to the first part of each scenario was their opinion of the meaning of the information in the Gantt and earned value reports. The subject matter experts were in general agreement and confirmed that the status reported in the conventional reports was in close proximity to what they would have reported themselves. For Sc1-Q.1, Sc3-Q.1, and Sc4-Q.1 the SME were in total agreement with the information in the Gantt and earned value reports.

In some cases the reported status was one degree more optimistic than what they would personally report, based on the data they were presented in the Gantt and earned value reports. In those cases the demonstration stoplight reports were deliberately created so that the reported status was optimistic, in keeping with Snow and Keil's observations on IT project status reports (Snow & Keil, 2001).

The interpretations by the SME in those cases were reasonable because the data in the reports for Sc2-Q.1, Sc5-Q.1, and Sc6-Q.1 were on the edges of two possibilities, as might be found on a real world project. As was stated in section 5.2, the intent was that the responses would be within a range, not an exact match. One subject matter expert, SME-01, believed that the cost constraint for Sc2-Q.1 should be reported as the next worst status, therefore SME-01 was in close agreement with the norm. In responding to Sc2-Q.1 SME-04 would reported one grade worse, as well as being moderately more aggressive when looking at Sc5-Q.1, and Sc6-Q.1, suggesting that those should be reported as one worse, too. None of the SME found that the reported status were grossly out of line, so the conventional reports were taken to be acceptable for later comparison to the CI methods.

When looking at the responses to Gantt charts and earned value reports it is interesting to note that in spite of the hard numeric data, latitude exists in the interpretation of those numbers. For the purposes of this dissertation, the value of the Gantt charts and earned value reports was that they established a point of reference for later comparison to the scope monitoring methods. With a general agreement that the reported results were within an acceptable range was sufficient for later comparison to the CI methods.

5.3.2 Discussion on CI Method Data Matching Traditional Reports (RQ-3)

For the demonstration scenarios where there was no new information that might change the interpretation of the status, the expert evaluators found that the CI methods produced results consistent with the conventional reports. The meaning of the data provided to the SME did not vary from the status already reported in the Gantt and earned value reports. The determination of the experts was that the CI methods properly reported when there was no change in status. This determination was necessary to ensure that the CI methods did not overcompensate or overshoot the meaning of the status.

All SME agreed that for Sc1-Q.2, Sc2-Q.2, and Sc5-Q.2 the status should remain the same. Most of the SME had a similar reaction to Sc1-Q.3, but one SME wanted to escalate the status based on the little information in the CI methods. This was a more aggressive response than the other SME, and came from SME-03, a manager with the most years of experience. This variation in responses aligned with research findings that have shown that reviewing activities is very difficult for status yellow (Barnes & Hammell, 2008). A modest range of opinions could be expected.

Sc1-Q.5 was a special case designed to determine the impact of not using the CI method for additional information. The question to be answered was if there were other scope issues presented in the written description that might cause a deterioration of the scope constraint, was the non use of the CI method noticeable. Since the additional written information was not input using the CI method, Sc1-Q.5 was intended to determine if the lack use of the CI method was significant.

For Sc1-Q.5 the opinion of SME-01 was that the CI method should have reported a status of "yellow", which meant that the CI method could have or should have been used, but was not, and therefore an opportunity was lost to capture scope status. SME-03 expressed a similar idea and would have used the written description of project problems to initiate a changed status for activity 28. Only SME-04 would not change the status, but it should be noted that SME-04 would have used the information to generate a risk register entry. The observation to be made was that evaluators felt that the lack of use of the CI methods meant missing a chance to take corrective action.

5.3.3 Discussion on CI Methods Providing Data Not in Traditional Reports (RQ-3)

For the demonstration scenarios which added new data for the scope that might change the activity's status, the expert evaluators found that the CI methods produced good results. The new data was a subjective opinion of the status, collected by using CI methods, and containing additional information not found in the budget and schedule reports. The evaluators found that getting this additional, albeit subjective, information gave them new insights into the condition and progress of a project.

The scope information provided in Sc2-Q.3 indicated concerns and potential problems, but it was open to interpretation whether or not the status for activity 21 should have been escalated to "yellow". The visual data showed a mostly "green" moving in the direction of "yellow", which is a marginal, minimal change. Since this activity was not on the critical path, the interpretation of the change was built to be vague in the demonstration. This was in alignment with the findings of other project management researchers who discovered that a status of yellow is frequently misunderstood (Snow & Keil, 2002). In a real world instantiation of the CI methods, a verbal hedge that reflects the management philosophy on what to do for "green leaning yellow" would adjust the status accordingly.

The significant finding was that all SME agreed that when the new data from the scope monitor was clearly a status "yellow" when all other reporting methods were indicating an okay "green" status, this new scope information changed their perception of the activity. SME-01 stated for question Sc2-Q.4 the CI method provided data that would cause the status of the entire project to change to a status worse than first indicated. SME-01 stated that this was new information that could not be found on the Gantt and EV reports. SME-04 would take the action of creating a new risk register entry.

Similarly Sc5-Q.3 was an evaluation scenario with scope issues on activity 15 that needed to be addressed by a project manager, and that the traditional reports did not provide any insights that a scope problem was occurring. SME-04 provided a list of potential actions that a project manager might take after viewing the scope data.

The subject matter experts reported that the new data from the CI methods captured meaningful scope information not found in the traditional earned value and Gantt charts. Their evaluation of the demonstration scenarios was that in some cases the visualizations communicated that there were moderate, subtle concerns that were only evident from the CI methods. This was validated for a variety of alternative scenarios.

The most noticeable impact was that the scope data would cause the evaluators, if they were the project managers, to take action. Sometimes the action was to request more information, an action that otherwise would not have been taken. SME-01 stated that in one instance a CI method status of "green" was correct for scope, but the impressions of the intent of the information provided as part of Sc1-Q.3 would cause an investigation as to "why some team members feel that the scope is slipping." The conclusion was that the data from the CI method for that instance was predominately green status, but by moving in the direction of yellow it indicated potential issues. For that same scenario SME-03 was alerted by the direction of the movement and stated that based on the scope data a project manager should "monitor closely put contingencies in place", again an example of taking action because of the additional information from monitoring scope.

There were other examples of the noticeable impact of the scope data. For scenario four all three of the expert evaluators would use the information provided by the CI methods in Sc4-Q.3 to take action. The SME varied in what they would do: one would add contingency plans, another would add more resources. A third would use the scope data to work on "team dynamics and communication, and it requires intervention." The extra information on scope which was not found in the other reports altered their behavior, even though they might not change the reported status. This finding was reinforced by Sc4-Q.4 which was included to clarify the SME thoughts on the critical path issue. There were some variation of opinions by the SME as to how the CI methods should report the data provided by Sc4-Q.4, but all expert evaluators concurred that managers should distinguish between the reporting process of selecting a stoplight color to represent status and the initiating of corrective actions. The color itself was not as important as the insights that generated and energized processes to analyze and fix. The evaluations for scenario four are discussed in more detail in the section on critical paths and linguistic hedges.

5.3.4 Discussion on CI Methods for Linguistic Hedges on the Critical Path (RQ-4)

For the demonstration scenarios which presented the subject matter experts with CI methods using linguistic hedges, their evaluation of the design was generally positive. Some found that the CI methods produced good results, with one researcher ambivalent about the demonstration for one specific instance. These scenarios addressed research question four which sought to determine if computational linguistic hedges could enhance the understanding of the scope constraint for tasks on a project critical path. This was related to question 1.d which was investigated in the literature review and discussed in Appendix A. These scenarios presented subjective data that was adjusted by the linguistic hedge "very" to accent the relative importance of that activity on the critical path.

As a starting point for comparative purposes, the data provided in Sc4-Q.2 did not change the status of scenario four, but only matched the Gantt and EV reports for activities 15, 29, and 31. If linguistic hedges were used, the design of the CI methods was such that it should not impact those tasks, even though activities 15, 29, and 31 were on the critical path. This part of the demonstration scenarios ensured that the design artifacts should report the critical path status with no change, when there was in fact no change. The design was such that additional scope data would not cause unwarranted boosting or unnecessary amplifying of the status. All of the SME agreed that Sc4-Q.2 represented a correct status for activities 15, 29, and 31.

In Sc6-Q.3 activity 33 was on the Time Critical Path, therefore the CI methods would use a linguistic hedge to modify the status. In this case the data input using the web gadget was predominately "yellow" with a significant move in the direction towards "red". As a result, the CI method used the hedge "very", downgraded the status, and reported status "red" because activity 33 was on the critical path. The SME would leave this new status as is for Sc6-Q.3, and be prompted to engage or intervene. As a corrective action SME-03 would look to modify the time schedule or get more resources. SME-04 would investigate first and analyze then initiate corrective actions. Similarly, SME-01 would analyze schedule and resource constraints, but also believed that the cost data in the earned value report should cause the cost status for activity 33 to be "red" as well. To SME-01 the impact would be that both the cost and the newly reported scope status would force the entire project into an overall "red" status.

It was noted that SME-04 was less inclined to allow the CI methods to automatically escalate the status for scope, but was more aggressive in escalating status for critical path activities. This observation was confirmed with SME-04 in a follow-up conversation on 5 October 2012. When responding to Sc2-Q.1 SME-04 stated in that situation they would report the next worst status because the activity in Sc2-Q.1 was on the critical path. In the follow-up conversation this point was clarified: SME-04 felt a need to emphasize the critical path activities. SME-04 would use the scope information, but in select cases was not inclined to allow the automation via CI methods. SME-04 showed that tendency in SC1-Q.3 when they stated they would not change the status to one level worse, but found "the information is valuable to take early action to prevent it becoming yellow."

When comparing the scenarios to each other, one of the differences between scenario four and scenario six was that in scenario four the data input to the CI methods was a solid "yellow" status, whereas for scenario six the input was a "yellow" with significant leanings towards "red". One of the premises of the demonstration scenarios was that the linguistic hedges would be seen as aiding in the understanding of the relative importance of a vague status by escalating it. For the case where the status is moving towards a crisis, the expert evaluators agreed that the escalation was informative. SME-04 in commenting on the methods in a follow-up conversation "felt that it was great." By using CI methods, the CI method would alter the meaning of the data that was input, thus alerting the manager to potential problems.

However, for the yellow status in scenario four, the opinions of the expert evaluators were mixed. Being able to distinguish states for a status "yellow" has been proven to be an extremely difficult and significant issue as found by project management researchers (Barnes & Hammell, 2008). Under certain circumstances there was not a definitive consensus of opinion by these evaluators as to what exact status the CI methods should report when they appraised the CI linguistic hedges for the critical path. Given that all of the evaluators felt positively about the CI methods, using linguistic hedges to adjust the critical path status remains a possible, but not definitive solution to the issue of monitoring scope. Some of the evaluators found that getting information that highlighted the critical path gave them new insights into the condition and progress of a project. Others wanted more personal control over the declaration of the status, which was counter to Snow and Keil's reservations about doing so (Snow & Keil, 2001). The evaluators wanted precisely what Snow and Keil's research says they should not have: total control over reporting status.

5.3.5 Discussion on CI Methods for Leading Indicators (RQ-5)

For the demonstration scenarios which presented the subject matter experts with CI methods for leading indicators, their evaluation of this component of the design was resoundingly positive. All of the SME found that the CI methods produced valuable information for future activities or activities that had just started. The scenarios that addressed research questions RQ-5 and RQ-1.c sought to determine whether computational intelligence methods applied to a project's scope could produce leading indicators or early warnings of project problems.

The SME found that Sc3-Q.2 gave new, actionable information for activity 31. For Sc3-Q.2 the CI methods reported information for activity 31 that impacted the project status for scope. This question, in asking about a future activity, determined the applicability of the CI method as a leading indicator. The caveat was that CI methods reported a status of "red" for activity 31, but SME-01 felt that because there was 2 weeks before the start of that activity there was sufficient time to for a PM to make adjustments to the project. To SME-01 the "red" for activity 31 was okay, given there were legitimate problems for activity 31, but the overall status for the project should be "yellow" reflecting that the potential problems were in the future. So SME-01 would use the CI method information for that activity to initiate corrective adjustments, but modify the project wide interpretation of the status depending on the situation.

SME-02 stated that the Gantt and earned value reports alone did not give the complete picture. In order to get visibility to problems on the horizon SME-02 went on to say that the "new fuzzy scope status graphic, along with the (written) explanation for the scope status, provides insights that might otherwise go unreported." This was a testimonial to the benefit of the CI methods to collect leading indicator data. SME-04 "had no concern" with the task until the CI method pointed out a major issue "that had not been otherwise identified."

Question Sc3-Q.4 was a follow on question to question Sc3-Q.2, intended to allow for comments and discussion of responses to Sc3-Q.2 on the potential for leading indicators. Three of the expert evaluators, SME-02, SME-03, and SME-04 would initiate immediate actions based on the leading indicator information. One would look to contract the new scope, and the other two would added tasks to the project corresponding to the scope increase. The other expert evaluator, SME-01 assumed that with significant time left on the activity the issue could be handled, but that evaluator allowed that the CI methods provided new information that required attention.

The demonstration scenario was clear that there was scope creep in the form of additional work that had to be performed, so the evaluators found that project managers needed to obtain additional funding, schedule relief, and/or rebaseline the project. Although there was some disagreement as to the use of stoplight colors, the data gathered should cause a project manager to initiate actions. Therefore the CI methods were being used as leading indicators.

In support of the quest for leading indicators, SME-04 responded to Q.7-5 with a desire for a "real time warning when a shift towards yellow or red is beginning, rather than simply point in time when a report is run." That SME felt that would give project managers even earlier notification of potential problems. This would entail a modification of the design artifacts, that was outside of the goals of this dissertation. This request is both reasonable and feasible, but was put on the list of potential areas of future research.

5.3.6 Discussion on SME Feedback related to DSRM

As part of the evaluation package the SME were asked to comment on topics related to the design science research methodology. Hevner's, et al., seven guidelines for DSRM as listed in chapter 3 were employed as being relevant for this research. The subject matter experts were asked about DSRM through their observations and remarks on items two, three, and six in that enumerated list. From their feedback came evidence to support that this dissertation met those guidelines. The other guidelines, such as guideline seven on communicating research, were covered by other means described earlier in this dissertation.

5.3.6.1 Guideline Two: Problem Relevance

It was clear from the literature review that monitoring scope was an excellent candidate for the application of computational intelligence methodologies. This point was repeatedly found in the literature review in findings by multiple researchers (Gido & Clements, 2009; Kerzner, 2011; Kwak & Anbari, 2012). In responding to Q.7-1, Q.7-2, Q.7-3, and Q.7-9, all of the SME concurred with the literature review in that scope was a significant issue on IT projects. SME-01 stated that customer satisfaction depended upon scope, which reinforced the significance of scope to the recipients of information technologies.

A good DSRM process will create artifacts that are relevant, and Q.7-9 corroborated that the SME who evaluated this research's artifacts had expectations that were in agreement with findings in the literature review. Question Q.7-9 confirmed that the goals of project managers concerning scope were comparable to those findings upon which the dissertation research questions were based.

The SME responses in combination with the literature review findings fulfilled the criteria that artifacts solve important business problems (Hevner & March, 2003). The responses to Q.7-1, Q.7-2, Q.7-3, and Q.7-9 confirmed that this research met the DSRM guidelines that an IT artifact should tackle a relevant problem. Furthermore, all the SME indicated that they have had issues with scope creep. This confirmed that the experiences of the SME matched the findings in the literature review that creep is a common event on IT projects, and one that could kill projects.

5.3.6.2 Guideline Three: Practicality of Design

Hevner, et al., (2004) stated that business organizations exist in an economic environment, and that a design must fit within that constraint. Their third DSRM guideline stated that artifacts from a design should have utility, quality, and efficacy. They believed that the strategy should be to "produce feasible, good designs that can be implemented in the business environment" (Hevner, March, Park, & Ram, 2004, p. 88). Associated with this aspect in design science research is the costs of implementing a given design in a corporation.

Q.7-7 was apposite to the practicality issue by establishing that the financial impact and other costs of implementing the CI methods would be reasonable. The SME expressed the opinions that the cost was mostly the time to gather the data on a report cycle basis. Only SME-04 expressed any additional concerns about the costs by questioning the effort to get employees to cooperate. SME-04 while affirming the value of the data to the management team, stated that "assuming it’s the people doing the tasks who provide those inputs, it can be challenging to get them to do one more administrative thing." However, the costs did not outweigh the benefits.

5.3.6.3 Guideline Six: Design as Search Process

Hevner, March, Park, and Ram's sixth guideline for using design science as a research methodology was that the solutions process was iterative and heuristic. It has been stated that after an evaluation has been completed researchers can decide to improve the design or to communicate their findings and save improvements for future research

(Peffers, Tuunanen, Rothenberger, & Chatterjee, 2008). The demonstration scenario Q.7-4 was seeking to find if any of the SME had discovered tools or used techniques that were equal to or better than the proposed CI methods for scope. In all cases the SME depended upon conventional word based methods, which were shown to be lacking in the literature review. The methods in use or recommended by the SME were tight control of scope changes and managing the WBS to limit scope. Both of these were traditional word based approaches, and were only offered in the context of conventional methods to managing scope issues.

Q.7-8 and the related Q.7-5 and Q.7-6 were open ended queries that asked what improvements could be made to the CI methods. This was pertinent to two issues in design science research of 1) iterative improvements to a design, and 2) that the CI method only needs to "satisfice" and is neither a complete nor perfect solution. On the first point the process model for DSRM assumed that artifacts could be improved, and therefore included a loop that allowed for iteration of earlier activities. This has been described this as a generate and test cycle (Hevner, March, Park, & Ram, 2004). The artifacts from the CI methods could and should be seen as always having the capacity for modifications. Unless the suggested improvements had uncovered significant flaws in the design, the design was considered finished. Q.7-8 along with Q.7-5 and Q.7-6 gathered from the expert evaluators evidence that the artifacts met the requisite design objectives. In the absence of major design flaws, and in the light of positive findings, the design was completed.

On the second point, the CI method was shown to function in a way that resonated with evaluators, and that the artifacts presented new methods that would change professional practices. The demonstration of concepts in the evaluation package provided descriptive detail that was credible to the SME and identified innovative properties and new dimensions. This research stimulated discussion of potential enhancements as shown in Q.7-6 when SME-04 asked for "even earlier notification of potential problems." SME-04 went on to explain that they wanted a "real time warning" to signal to the project manager when a shift in status was in progress "rather than simply point in time when a report is run." This suggestion to speed up the feedback cycle could be done as an enhancement or extension to the current design, and did not require a redesign of the current artifacts. The scenarios of the CI methods demonstrated to subject matter experts the efficacy of the system, far exceeding the requirement to simply "satisfice".

The CI methods induced discussion of the idea that some individuals understood the project status better than others. SME-01 stated that some team members had perceptions of status that were more valid or relevant, and therefore held "more weight." The design was flexible enough that this could be accomplished with personalized linguistic hedges, one hedge per team member. This use of the CI methods was a new variation that was outside the range of this dissertation, but could be part of an instantiation of the design.

In summary the evaluation instrument used in this study included additional opportunities for the SME to comment on the design science research methodology. This research embraced Hevner's, et al., guidelines for DSRM in order to ensure methodological consistency. The coherence to DSRM guidelines two, three, and six was affirmed by the subject matter experts in their responses to the evaluation package. From their feedback came the evidence to support that this dissertation met those guidelines.

5.3.7 Summary of the Discussion

The evaluation of the demonstration scenarios by subject matter experts supported the premise that IT projects can be monitored for the qualitative criteria of scope. The evaluation of the scenarios established that fuzzy computational intelligence methods could capture scope status in a manner that would be useful to project managers. Their evaluation provided a critique, comments, and observations applicable to research questions three, four, and five.

For research question three the SME confirmed that the CI methods produced proper results when done as a complement to conventional reports. The SME also found that there could be cases where the lack of a scope monitor meant the loss of an early warning that could have lead to rapid corrective actions. The SME also observed that the CI methods gave extra information that was not reported in the traditional earned value and Gantt reports. This was validated for a variety of alternative scenarios that compared subjective scope to conventional budget and schedule reports.

The evaluation of the scenarios demonstrating CI linguistic hedges for tasks on the critical path was less definitive. These findings addressed research question four. There was agreement that when the status was yellow moving towards red, and the activity was on the critical path, the CI methods provided valuable insights and should escalate the reported status. For shades of yellow or green leaning yellow, how the CI methods should report the status was open to interpretation. The significant finding was a consensus that whether the reported status was green or yellow, the scope monitor provided information that should cause a project manager to monitor an activity more closely. To that extent, the scope monitor provided a valuable service that initiated actions, but might not change the reported status.

For research question five (RQ-5) the evaluation of the demonstration scenarios for using CI methods as a leading indicator or early warning of scope issues was very favorable. The project management experts found that the scope monitor gave new, actionable information for activities that had not started. The CI methods reported information that impacted the project status for scope, and more importantly should cause a project manager to initiate corrective actions. When the scope monitor provided data about a future activity, the conclusion was that the CI methods as a leading indicator and early warning provided great value. The SME went on to suggest that the early warning be provided in real time, instead of waiting for a reporting cycle.

The process that produced these findings conformed to March and Smith's criteria for the evaluation of design science artifacts, which stated that methods should be operationally capable of doing the design task (March & Smith, 1995). SME-01 said that the visualization of the scope status increased the information for a project manager, and SME-03 responded succinctly that the visuals help manage projects. SME-02 stated that "scope status provides insights that might otherwise go unreported."

The SME evaluation of the artifacts also established that this dissertation adhered to design science research methodology guidelines. The design artifacts adduced evidence in support of the CI methods solving the problem of monitoring scope on IT projects. The demonstration scenarios proved that the design was feasible and practical.

The DSRM search process was iterative and heuristic, a principle that produced a solution that satisfactorily resolved the research questions.

To summarize this discussion of the evaluations, the design artifacts were demonstrated through the creation of six project management scenarios. Those scenarios were evaluated by subject matter experts who found that the additional data collected by fuzzy computational intelligence methods has an impact on a project manager's perception of the status. When used to monitor subjective IT project scope those new understandings become a catalyst to initiate corrective actions. Meredith and Mantel (2009) stated that data supplied by monitoring provides information to make decisions to control projects towards the project objectives. The major finding of the evaluation of CI methods to monitor scope was not the capability to declare a particular status or stoplight color, but the capacity to alert management to potential problems. Getting status information for scope meant getting additional information on current activities and early warnings of future problems that could mitigate scope problems. These are capabilities of great import and consequence to project managers.

Chapter 6 Conclusion

In this chapter is the limitations of this research, recommendations for future work, and the contributions achieved in this research. The contributions are connected back to the research questions identified in chapter 4.

6.1 Contributions

The primary contribution of this research was the design of methods utilizing computational intelligence for IT project management. The significance of the study was in the application of CI methods as a solution to the important problem of monitoring scope on IT projects. It was shown that a better understanding of the status of scope on information technology projects can become a catalyst to management action. In conducting this research, the specific contributions were the design artifacts that were generated as computational intelligence methods.

The design artifacts ranged from the frontend data collection to the backend processing of the collected data. The frontend data collection was implemented using computational intelligence based on Zadeh's concepts of fuzzy sets in order to capture subjective inputs for scope status. A component of this was the set of artifacts that applied fuzzy set theory and logic to project management monitoring of the scope constraint in order to classify the status for stoplight reports. The design artifacts included backend processing using fuzzy logic to aggregate the status of multiple task. The unique features were that CI methods were demonstrated to capture a meaningful scope status for those cases when the scope is subjective.

Other contributions were derived from the scenarios developed for the evaluation by subject matter experts. It was shown that computational linguistic hedges can enhance the understanding of the scope constraint for tasks on a project critical path. Another original contribution of this research was the demonstration that computational intelligence methods when applied to a project's scope are capable of gathering data that can be used as leading indicators.

An ancillary contribution of this work is the implementation of the design science research methodology to project management. Heretofore, research in the field of project management had been devoted to the natural and behavioral science methodologies. With the introduction of design science to the project management field, the same benefits that were seen when DSRM was implemented in information systems and information technology research were realized. This research methodology opens the field of project management to the possibility of extending "the boundaries of human and organizational capabilities by creating new and innovative artifacts" (Hevner, March, Park, & Ram, 2004, p. 75).

6.1.1 The Research Questions

This study was decomposed into five research questions which were summarized in Table 5 in chapter 4. The first research question looked into what aspects of monitoring a project might benefit from computational intelligence methods. This question was addressed in the literature review which established that there are qualitative aspects to scope problems on IT projects that could benefit from the application of CI methods. It was also established that there are no CI techniques in current use for the type of subjective, qualitative constraints that are often found during the monitoring of a project.

The second research question asked what CI processes should look like in order to capture subjective inputs. The result was a design of CI methods for monitoring scope status. This design was presented in the third activity in the design science research methodology (DSRM) in the building of artifacts in section 4.4. This encompassed a design of CI methods for both the frontend data collection and the backend processing.

The third research question entailed the evaluation of the CI methodologies in a detailed demonstration of capturing a meaningful status for when the scope is subjective. The demonstration was critiqued by subject matter experts for alternative scenarios.

Research questions four and five looked at linguistic hedge and leading indicator extensions of the design artifacts from research question three. Question four investigated and found limited benefit to utilizing computational linguistic hedges for tasks on a project critical path in the opinion of the subject matter experts. The evaluation by the experts found that for question five the CI methods positively enhanced and increased status understanding by project managers when implemented as leading indicator.

The artifacts produced as outputs of this dissertation are an innovative and creative application of computational intelligence methods to a significant issue in IT project management. The contributions of this research can be summarized as:

- The unique application of computational intelligence to a significant problem: monitoring the status of the scope constraint in the execution phase of information technology projects
- The design of methods to measure and monitor qualitative indicators on projects

- The design of methods for leading indicators as an early warning for scope issues on information technology projects, an original and novel application of CI
- The design of methods utilizing linguistic hedges implemented in computational intelligence methods to make adjustments for activities on the critical path
- The fulfillment of a formalized design science methodology to the project management research domain

6.2 Limitations of Current Research

In creating a two dimensional framework for IT research, March and Smith (1995) identified limitations of the design science methodology. One is that the design artifacts can become quickly obsolete. The original needs that drove the design might fade in importance, or the technologies get replaced by newer. Another limitation is that design artifacts might move too quickly into practice without supporting theories. They pointed to the failure of relational databases to handle applications such as CAD as one example.

This research was limited to the design science approach. As described in the literature review, March and Smith (1995) recognized that both natural science and design science were important to progress in IT research. The design science approach was done in contrast to natural science for IT which would include descriptive research to create and justify theories. Furthermore, this research was limited to the building of *methods* under the design science paradigm. This limitation of only providing *methods* meant that an *instantiation* was not attempted.

This research limited the demonstration and evaluation of the methods to IT projects. The methods and findings described could be generalized to any type of projects, but this was not investigated. Furthermore, this study was limited to conventional IT projects and did not consider newer paradigms such as agile or scrum.

The fuzzy sets that were described for the stoplight colors were created solely as an example for the design artifact. This is a limitation of the proposed design. In a production environment subjective observations from proper authorities would be used. They would create maps that represent the variables red, yellow, and green. Zimmerman stated that the creation of fuzzy sets should be done “empirically rather than from theoretical considerations” (Zimmermann, 1996, p. 183). In the evaluation of the design SME-04 pointed out that "one person's red may be another's yellow" and offered that providing a standard explanation "into how the status judgment is derived" would be valuable.

This research was limited to the study of projects of modest and reasonable risk. Loch, DeMeyer, and Pich (2006) identified a type of high risk project, with unforeseeable and highly unpredictable risks which they called novel projects. The existence of this type of project was confirmed by Snowden and Boone who stated that most organizations are faced with complex to chaotic problems that require sensing skills in managers. They believed that the domain of complex problems requires an experimental mode of management to cope with the "unknown unknowns" (Snowden & Boone, Nov 2007). Kerzner (2011) also confirmed that IT faces high risk projects by contributing that the business world is full of nontraditional projects with changing technologies, ill defined statements of work, or assumptions that must be altered due to the long duration of a project. This dissertation deliberately excluded novel projects, projects of very high complexity, and other nontraditional projects.

Consequently, many of the limitations of this dissertation's research provide opportunities for future research. Some of the possibilities are identified in the next section as recommendations for future work.

6.3 Recommendations for Future Work

This research was limited to *methods* in the design science approach. This limitation meant that alternative research approaches such as *models* were not attempted. Open questions that could be further researched are the other elements in the framework that were established by March and Smith (1995). The most obvious next choice in their framework would be the *instantiation* of the design artifacts. This is a task for the computer science field involving the selection of computing languages, operating systems, selecting a graphical user interface framework such as AJAX or HTML5, choosing which database management system to use such as Oracle, DB2, or MySQL, and then coding with optimal efficiency.

This research was limited to the design science approach, but both natural science and design science can contribute to IT research (March & Smith, 1995). Other behavioral science research on the CI methods could provide theoretical diversity if implemented in quantitative and qualitative studies. If there was to be an *instantiation* of the designs, that would open the door to behavioral science research to explain why or how CI methods work in the IT environment. One idea would be to implement the design in multiple organizations and to perform case studies.

The graphical user interface input gadget that was illustrated in Figure 7 implemented a non-numeric, linguistic scale that was simple and understandable. Given that improvements in design is a fundamental principle of design research, other researchers might uncover better methods or more efficient algorithms for the frontend input mechanism.

The graphical user interface and the CI methods in this research were limited to a very specialized application: information technology project scope monitoring and management. The methods and findings could be generalized for any type of project.

Future research could challenge some of the assumptions of this dissertation. This research assumed that project managers and team members will honestly record what they know or believe to be true when using the CI methods. This is an assumption that could be tested in future research.

Another assumption that could be challenged was limiting the study to projects with reasonable and manageable risk. Loch, DeMeyer, and Pich, Snowden and Boone, and Kerzner all gave examples of projects that were well outside those risk boundaries. There certainly could be room to research the how the computational intelligence methods could aid in the management of novel projects, projects of very high complexity, and other nontraditional projects.

6.3.1 Recommendations for Scope

Two issues not addressed in this dissertation were related to extending the use of linguistic hedges. One possibility is that a project manager potentially could use linguistic hedges to "water down" the opinions of alarmist individuals. Another is that if the variance in data input is so great, perhaps the CI method should flag the differences as NULL or UNKNOWN, and force management attention to the wide variances in opinions. These topics are left for future research, and were outside the boundary of this research.

Loch, DeMeyer, and Pich (2006) determined that there was a type of extreme high risk project, with unforeseeable and highly unpredictable risks which they designated novel projects. They argued that when a project is of very high complexity the interactions of components cannot be anticipated, resulting in the ablating of planning efforts. There is room for research of the application of the CI methods to these novel project types.

One of the items considered in the design of the artifact of this dissertation was to have project plans with multiple paths. The multiple paths would allow the creation of scenarios that change the critical path as time elapses on the project. This would demonstrate the flexibility of the CI methods to recalculate scope status. There is no reason to expect that this cannot be done, and is left as a potential future work to be done as an instantiation.

An assumption made implicitly was that time and scope status move forward in a linear fashion. What was not considered was the possibility that after a WBS activity is totally completed that its scope status might be less than a perfect or "green" condition. What could be studied would be to conduct empirical research to determine if this could happen, and if the CI methods proposed would be applicable.

Some of the suggestions provided by the SME during the evaluation of the design artifacts and by the attendees at presentations at conferences were taken as recommendations for future work. These ideas were considered beyond the scope of this dissertation, but worthy of mention. The experts proposed the following enhancements:

- Real time alerts on changes to project status. This extends the concept of early warnings to instant feedback. This could be done in an instantiation of the CI methods.
- Charting the scope status over the time domain, in a manner similar to the charting of earned value. This would allow for trend analysis of the scope on a project.
- Weighting of the scope data based on the trust of the project manager in the opinion of a given individual. This scope adjustment could be accomplished in an instantiation by using linguistic hedges with the more trustworthy individuals being assigned a "very reliable" hedge.

6.3.2 Recommendations for Risk

Risk triggers inform project managers when there has been a transition from a potential event to an actual problem. They act as signals and warning signs of the realization of risks. Capturing the state transitions of the triggers with a computational method that engages and exploits the fuzzy nature of those warning signs has the potential to make the reporting system congruous with project reality. The early warning and correct interpretation of risk triggers would allow project managers to take faster corrective actions.

When this topic was presented to practitioners at the CISSE conference there was little interest in applying CI methods to risk. Therefore, the design artifacts and later evaluation by subject matter experts did not include risk triggers. It is hoped that the completion of this research encourages further work in the area of risk management using computational intelligence. A design for CI methods for risk triggers is in the next section.

One area of risk research could be in the field of linguistic hedges applied to project risk fuzzy sets. Hedges operate on fuzzy sets by creating subsets (McNeill & Freiberger, 1993) such as the verbal hedge "less" which decreases the size of a set. CI could allow a project manager to adjust the interpretation of the risk event status using the CI hedges. One example might be if a project manager uses the hedge "very important" to intensify the reported status for risk events that impact the critical path. CI would also allow for diminishing the importance of risk events associated with activities not on the critical path with the "less" hedge. The use of hedges for scope when a task is on the project critical path is proposed for future work. It is believed that methods for gathering fuzzy status for a risk event with computational intelligence tools has merit.

A second area of potential research is in Hancock's classification of projects that he labeled Type IV problems. Type IV was described as when aleatoric uncertainty leads to unanticipated results and unknown outcomes. These types of projects are "heavily dominated by opinions, beliefs, and ideology about the future, requiring the use of qualitative methods" (Hancock, 2010, p. 70). The concern of this research paper has been fuzzy triggers for more conventional projects, which excluded Type IV projects from consideration. However, it can be anticipated that the methods for monitoring imprecise risk events could lead to applying CI to gathering opinions on Type IV projects.

6.3.2.1 Design of CI Methods for Risk Triggers

As described earlier, risk items are assigned to an individual who has the responsibility for determining to what extent a risk item has changed state. The CI methods described in the design artifacts can be extended to ask the risk owner to rate risk triggers on a scale that is word based, not numeric. This would be relevant when those risk events are fuzzy, not crisp, a case which was pointed out in the literature review. The non-numeric scale that was created for scope status was designed to be easy to use by experienced project managers. It could also be implemented for risk triggers by using the same look and feel of the scope methods.

As an example, this is illustrated in a simulation of the input web gadget using a non-numeric, linguistic scale for capturing risk event status. The person tasked with reporting the current status for a risk event would spray paint their impressions or judgments using the web gadget. Figure 23 would represent the gut feeling of the risk event owner who believed or estimated that a non-quantitative trigger has already occurred to a modest degree. The data input to the web gadget is purely the subjective opinion of the individual responsible for reporting the status.

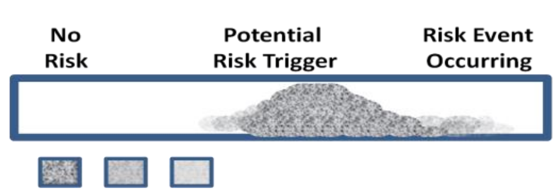


Figure 23 Fuzzy Status of a Risk Trigger

These spray paint data points are converted to numeric values in backend processes using the same principles of fuzzy sets as were defined for scope. These numbers are then associated with and stored in a central data base as part of the data for a risk event. The benefit is that the reported risk event is now a permanent part of a monitoring system that could be tracked over time. By using computational intelligence algorithms the risk event state can be tracked, evaluated, and integrated with other risk events.

Going back to Snow and Kiel's (2001) finding that project managers tend to report optimistic status and downplay the early warning signs, capturing the early warning signs from the risk event owner has the potential to make the reporting system congruous with reality. Just like the data in a centralized database for scope status leaves a persistent record in a format that can be analyzed and reconciled, the risk status would be a measurement that is analogous. Since errors in flagging risk triggers leads to errors in cost and schedule, the awareness of risk events would contribute to early corrective actions, increasing project success.

6.3.3 Recommendations for Behavioral Science Research

The design output of this research might lead to tools and techniques that can be utilized by qualitative researchers in behavioral sciences. Corbin and Strauss stated that grounded theory is a dynamic approach that is oriented towards words and meanings. Their views of grounded theory require the same attributes of creativity and imagination as design science. It also shares with computational intelligence methods the need to tolerate ambiguity. An instantiation of the design output of this research might lead to tools and techniques that can be utilized by qualitative researchers, especially for case studies where the collection and analysis of emotions is necessary. This might be implemented as a complement to interviews as an alternative mode of inquiry. When these methods become instantiated they become an alternative to gathering opinions with Likert scales.

6.3.4 Recommendations for Integrating Scope with Other Constraints

A final step might be the integration of the status of multiple, diverse constraints, an option that has already been demonstrated by Barnes and Hammell using recognition primed decision making to integrate budget, schedule, and scope constraints (Barnes & Hammell, 2008). An instantiation of the design artifact from this dissertation could provide a means for delivering a scope status into their intelligent decision system which used Recognition-primed Decision-making enabled Collaborative Agents for Simulating Teamwork (R-CAST). This integration with other constraints would complete the automation of status monitoring on IT projects.

6.4 Summary

In conclusion this dissertation implemented the design science research methodology in the creation of artifacts applying computational intelligence methods to a significant problem in IT project management. Previous research had applied CI to quantitative constraints in the planning and estimating phases. The unique contribution of this research is the innovative employment of CI methods to the subjective, qualitative constraint of scope in the monitoring and execution phases of an IT project. The demonstration scenarios clearly showed that the CI methods can aid project managers in determining the status of activities when the status was subjective. An evaluation of the artifacts by experts in the field of project management found that the use of CI techniques to monitor scope would assist managers when making a decision to initiate corrective actions.

Chapter 5 established the design of the methods which used fuzzy computational intelligence to acquire and process data on the status of project scope. The artifacts demonstrated that linguistic hedges when applied to activities on the critical path improved the understanding of the overall project status. It was clear that the additional, new information provided by the CI methods would prompt project managers to initiate actions even if there was no change in the reported status. A significant finding as confirmed by the subject matter experts was that the CI methods were capable of capturing data that were leading indicators to project success or failure. This research using the design science methodology confirmed that by capturing scope status with computational intelligence methods during the execution phase of information technology projects aids in the management of IT projects in the initiating of corrective actions.

Appendices

Appendix A - Published Papers and Conference Proceedings

Listed below are the peer reviewed papers that were presented at conferences as part of this research. The reviewers and participants provided comments and feedback for each of these papers which is described in this appendix.

- *MAICS 2011: The 22nd Midwest Artificial Intelligence and Cognitive Science Conference. April 2011, Cincinnati, OH. "Computational Intelligence for Project Scope"*
- *SERA 2011: 9th ACIS Conference on Software Engineering Research, Management, and Applications. August 2011, Baltimore, MD. "Scope as a Leading Indicator for Managing Software Development"*
- *CONISAR 2011: Conference on Information Systems Applied Research. November 2011, Wilmington, NC. "Computational Linguistic Hedges Applied to a Project Critical Path"*
- *IADIS 2013: The IADIS International Conference on Information Systems. March 2013, Lisbon, Portugal. "Applying the Design Science Research Methodology to IT Project Management"*

Listed below are the peer reviewed presentation and the peer reviewed poster that were presented at conferences as part of this research. The reviewers and participants provided comments and feedback for each of these which is described in this appendix.

- *CISSE: The 14th Colloquium for Information Systems Security Education. June 2010, Baltimore, MD. Presentation in the Short Topics track. "Computational Intelligence Applied to Information Security Risk Management"*
- *IRNOP 2011: International Research Network on Organising by Project. June 2011, Montreal, Canada. Poster and student paper. " Monitoring Project Scope Using Computational Intelligence"*

The primary discussions with participants at the 14th Colloquium for Information Systems Security Education (CISSE) were about how the algorithms would reconcile the inputs into a meaningful output. At the short time period allocated for discussion at CISSE it was difficult to go into detail with the participants, but this question was easily answered by referring the participants to the many different authors who have written on the topic of fuzzy math (Zimmermann, 1996; Klir, St.Clair, & Yuan, 1997). The submitted paper did not intend to explore new ground in mathematical algorithms, but instead exploited an established set of principles from other disciplines when the processing of fuzzy data was necessary for security risks. Considering that the focus of this conference was IT security, the insight gained from this conference was that there was modest interest in risk and security, but not sufficient interest to continue down this path. As a result the emphasis of this research shifted to IT project management where the level of interest was much higher and the existing literature on applying computational intelligence was very limited.

The reviewers from the 22nd Midwest Artificial Intelligence and Cognitive Science Conference (MAICS 2011) provided very little written feedback. They requested that the references to Levinson be updated and that the section numbers be removed. These simple cosmetic changes were made to the final submission to MAICS. In contrast to CISSE 2010, the MAICS conference participants made enthusiastic comments about their personal experiences on troubled software projects. A common theme was how they directly observed scope creep or scope problems on their IT projects. This was met with a general agreement by the majority of the participants, and seen as confirmation that there is a definite need for better scope control.

One of the participants at the MAICS conference commented about the applicability of Monte Carlo processes in the context of project management. That participant referred to research on project management which only applied Monte Carlo to the early phase of scheduling and planning. The paper presented at MAICS assumed that the planning had been done, risks had been assessed, and the project was underway. Therefore Monte Carlo scheduling would be a predecessor in the planning phase to this research which focuses on the execution phase of a project. In one example of Monte Carlo, Khedr (2006) investigated how deterministic scheduling tools and methods, such as CPM, address schedule risk compared to new probabilistic (stochastic) scheduling techniques. Khedr used Monte Carlo simulation to account for the risk inherent in projects when creating schedules.

The 9th ACIS Conference on Software Engineering Research, Management, and Applications (SERA 2011) provided comments from three reviewers. The first reviewer stated that they agreed that scope can be a leading indicator and requested a working definition of scope be included in the paper. As a result, more detail on scope was included in the final submission. That reviewer stated that they were expecting software projects, the specification of methodological steps, or the reporting of experiments. They were correct that at that point in time the submitted paper had not reached conclusions, and no experimental results were included in the SERA paper.

The second SERA reviewer stated that the paper presented a good motivation, a good problem, and a great number of related works. They requested a description of what the input of the method should be, who should be the individuals who put in the status, and perhaps a description of the procedure of the study. This section of the paper was not detailed enough for the reviewer, but to include a response to those questions would have been a repeat of the paper submitted to MAICS earlier in the year, and exceeded the mandatory SERA page count. It was believed that a reference to MAICS should have been sufficient. The last set of comments from the second reviewer was that the connections from the problem to the solution needed clarification. That reviewer observed that the paper addressed the problem of scope management but the solution measured the scope status. It is intended that this research were make this connection, and it is acknowledged that the paper submitted to SERA did not.

The third SERA reviewer stated that the paper was very well written, but suggested that the paper include function points as a means for measuring scope. The reviewer pointed out that the purpose of functional sizing is precisely aimed at measuring the scope of software development projects. This reviewer's comments were taken as a valid, but applicable to the types of projects that have clean, well defined scope, such as is found in the waterfall model. In recognition of the relevance of function points to the topic of scope, the final submission to the SERA conference included a discussion of function points, and function points are included in the literature search.

The final submission to SERA added clarification that function point analysis only works when the scope is fleshed out in enough detail to be able to add up the point values for each component. Fleming clearly demonstrated that the accurate measurement of project performance depends upon a complete scope definition (Fleming & Koppelman, 2010) but that software projects are notorious for poor requirements or undefined scope. Function points do work well in the waterfall model which is a sequential design process. The PMI also recognizes rolling wave as a legitimate technique in which the requirements become clarified as the project progresses. A complete and thorough function point analysis would halt the project at the design stages until the necessary details have been established. In support of this Kerzner has found that complex projects responding to business needs often do not have clear requirements (Kerzner, 2011). The Agile software development is described by the PMI as an application of rolling wave planning, in that there is an iterative and incremental delivery process (PMI, 2011). More recent models such as agile are the antithesis of rigid total-plan-before-any-work waterfall models. In that not all software projects are managed using the waterfall model, function points might not always be a solution.

An additional response to the third reviewer was that in some cases there is a lack of clarity about requirements during the execution phase. This was similar to a question raised by an attendee at the SERA conference. One of the purposes of this dissertation is to investigate using CI for those cases which are not a black or white decision concerning scope. Both the anonymous reviewer and the questioner at SERA assumed that all requirements and scope are clean, crisp, and well defined and therefore can be analyzed and number crunched using function points. Capturing the status to report problems with fuzzy scope happens during the execution phase at a point in time long past the creation of the function point estimates. A fuzzy scope collector can be a counterpart or complement to function points. In that the fuzzy data can be used during the execution phase for vague scope, while function points are used in the early estimation phase for crisp requirements, they measure different aspects of a project. Both attempt to achieve the same goal of project success.

Lastly the third anonymous SERA reviewer commented that fuzzy logic has been applied to functional sizing and software project estimation. The reviewer cited Azzeh, Neagu, and Cowling's work as an example. Upon further investigation it was found that their work was devoted to fuzzy analysis for software effort estimation, which occurs in an early phase of a project life cycle. Their paper is now referenced in this dissertation. Azzeh, et al, did not look into the monitoring and controlling aspects of a project, and therefore was seen as playing a role in predecessor activities which are not the same as this research. This dissertation concentrated on the execution and monitoring of a project which is the implementation phase, while estimating is a process accomplished in the planning stages.

The International Research Network on Organising by Project (IRNOP 2011) provided very little feedback on the content of the submitted paper. The paper that was submitted was a short paper published under the student category, along with a poster that was presented at the conference. Although there were a number of people who stopped to read the poster, there were very little comments of substance. That might have been due to language barriers, since many of the attendees did not appear to be fluent in English, in spite of English being the official conference language. The written reviewers did request modest changes to the poster such as increasing the font size. The submitted paper also clarified the relationship of scope to the other constraints. The conclusion was that using CI methods for the monitoring phase had not been investigated previously, and therefore there was little to no feedback from other researchers.

The presentation at the Conference on Information Systems Applied Research (CONISAR 2011) attracted a group of attendees interested in project management. One of the points made at the conference by Schwalbe, an author on project management, was that the critical path is the time schedule critical path, but the most important technical activities might not be on the *time* critical path. Her opinion was that those technical activities might have the most severe scope problems. From the perspective of the scope constraint, there might be non-time critical activities that are in serious trouble that the critical path method were not show. This implied that there is a need for a scope critical path. This research would support that conclusion in that instead of hedging only the time critical path activities the project manager could define an additional set of scope critical activities. Those activities would have a hedge applied like “very important” to them to create a scope critical path. This dissertation is only looking at adding the hedges to the time critical path, and leaves this thought as a potential topic for future research.

The anonymous written reviewers from CONISAR had comments on the originally submitted paper that lead to minor rewrites of sections with the intent to make it easier to comprehend. One comment asked for clarification of the point that scores collected by the CI method were have predictive capability. This indicated that the CONISAR paper was not clear on what was meant by predictive, and therefore that section was rewritten to be more precise. The rewrite stated that the Web gadget captures the predictions of the humans. The system does not, nor do the CI methods, predict. The methodology is dependent upon humans looking at future activities and offering their insights (predictions) as to the status of the scope constraint for an activity that has not started. That humans can look to the future and make predictions is irrefutable. Whether their predictions are valid was the focus of Klakegg's study. With Klakegg's finding that people instinctively knew there were unreported problems, and with so many projects in trouble regarding scope, a solution is needed. This methodology would capture and report those concerns about future events. Then the concerns should be addressed by the project management team.

One benefit to a system that uses CI methods is the bias errors by project managers would be removed if the scope status was tracked by computers. The CI process would capture scope status predictions and report that information, thus scope can be a predictor of success. The CI methods provide the means for collecting and preserving the predictions. Scope as a predictor has traditionally been ignored because it was fuzzy or subjective, and not measureable. This was supported by recent findings on complex projects (Klakegg, Williams, Walker, Andersen, & Magnussen, 2010). Kerzner (2011) adds to the support of this finding by his illustrations of nontraditional projects where the time duration extends over years and the assumptions about scope must adapt and change.

A second comment from the CONISAR reviewers was that much of the value of these fuzzy logic scores might lie in comparisons across multiple individuals who might be evaluating a given project element. Since the scores are not cardinal measurements, the reviewer stated it would be difficult to interpret a single fuzzy logic score. These comments about comparing multiple individuals lead to the section describing the fuzzy aggregation process being rewritten with more detail before final submission. The response to the reviewer's question was that the fuzzy logic scores are not compared, but are aggregated into an overall score when more than one person inputs data. This would be done using the mathematics of fuzzy logic including but not limited to conjunction, tautology, and implication (Klir, St.Clair, & Yuan, 1997). More sophisticated mathematical solutions were presented by Yager for the UNION and INTERSECTION operators, which ultimately converge on Zadeh's min/max as the class parameter that measures fuzziness as a model grows large, a property detailed in Cox (1999). For the purposes of this dissertation, Zadeh's min/max rules were determined to be sufficient.

One comment from a CONISAR participant was that one activity might only have one person inputting the status for that activity, and thus there was nothing to which to compare. In response when there is only one person working on an activity this one opinion is the only relevant data, and must be considered by itself. The CI system would keep that solitary, yet critical, opinion from getting lost. An excellent non-IT example of critical subjective data getting lost was the O-ring failure which caused the crash of the Space Shuttle Challenger (Rogers Commision, 1986). Before the launch it was the opinion of knowledgeable field engineers that the O-ring would not hold up in cold temperatures (Bergin, 2007). That critical information about a potential failure needed to be both recognized and addressed. The CONISAR participant was correct that CI methods do not manage cardinal values, instead fuzzy values. The single problem raised by a single person can have validity, which would be captured and utilized by the CI processes.

The need for a methodology to manipulate singular measurements is buttressed by Drucker (1993). Drucker originated the concept of distressed events being social phenomena that are hyperbolic. By hyperbolic social events he meant that problems involving social groups can arise in clusters and are not evenly spread throughout a Gaussian curve. A hyperbolic social event was certainly present in the Challenger disaster and can often be seen in software development projects. A small team on only one work breakdown structure element of a project might have an insight or understanding that should raise alarms to project managers.

Drucker stated that control systems should highlight, not conceal, the nature of the measurements. To apply this to IT projects, an example might be that one software developer might be tasked with validating security on a system that has thirty other software developers working for six months. If this security task is scheduled after all other developers have completed their code, the individual responsible for the task might reasonably be concerned about the scope of that task. This should be captured as a status issue that must be considered as relevant, to the point of being of potentially great consequence. This concept was added to the CONISAR submitted paper, and has been added to the literature review in this dissertation.

Another consideration in response to the CONISAR reviewers was that the difficulty in interpreting the fuzzy scores is left to the computational engine. To quote Cox "all the propositions are run in parallel to create an output space that contains information from all the propositions" (1999, p. 284) This means that each statement is evaluated as to its degree and then is added to the output solution. The example provided by Cox looked to make decisions about project risks by considering project duration, staffing, or complexity using an additive model. The fuzziness arose when duration is described as long, staffing is large, or project complexity is high, all of which are vague imprecise terms. The truer the statement, the higher is the degree of inference with each rule contributing to the solution (Cox, 1999). The final fuzzy set that is produced must be resolved to a scalar, a process called defuzzifying the data. Zimmerman (1996) covers both aggregation and defuzzification. Cox outlines centroid defuzzification, and also describes Kosko's Standard Additive Model (SAM) (Cox, 1999). The assumption for the CONISAR paper was that the fuzzy methods have been done before in other applications since the 1990's and should continue to work for project management applications.

The CONISAR reviewers commented that trends over time might be useful in a way that would highlight changes. The issue of tracking scope status changes over time is an excellent extension to this system, but one that was seen as a potential topic for future research. The reviewer raised an excellent point that if the scope status for an activity is slowly slipping from green to yellow to reddish-yellow over time, then this should raise management concerns. This suggests that there is a need for a scope trends chart, similar to earned value charts. At this time, the limit to this research is to just capture and report the scope, and to determine if this alone is of value to project managers. This point was added to the final submission to CONISAR.

The last question raised by the CONISAR reviewers was how can one assess whether differences are large enough to be meaningful. Without the ability to dialog about this question, it was difficult to determine what the reviewer was referring to as differences. If they meant small variances in opinions, then that is what the CI engine does: aggregate the various opinions as already explained above by Cox. This question by the reviewer does raise a couple of interesting possibilities if what they meant was wide variations in the subjective opinions when there are more than one person evaluating a task. For example: person A thinks the status is solid green, and person B thinks it is solid red. A solution that is not addressed in this dissertation is that a project manager potentially could use linguistic hedges to "water down" the opinions of alarmist individuals. Another is that if the variance in data input is so great, perhaps the CI method should flag the differences as NULL or UNKNOWN, and force management attention to the wide variances in opinions. These are topics that can be left for future research, but are outside the boundaries of this research.

The implementation of the design science research methodology (DSRM) as performed in this research was accepted for publication in the peer reviewed conference proceedings for the IADIS Information Systems Conference (IADIS 2013). The DSRM approach was presented at that conference in Lisbon, Portugal in March 2013 in the track for Design Research and the Sciences of the Artificial in Information Systems. At the conference two different participants asked about the applicability of the CI methods to agile software development. It was pointed out that agile was not considered in this research. This has been identified as a limitation of the current research, and a potential area of future research.

In summary, the acceptance of the peer reviewed papers and the experiences at the conferences with the written feedback from the reviewers served to confirm that scope status was an issue in project management that deserved attention. The vast majority of the topics that were suggested as related or relevant to this dissertation were usually found to be connected to the early estimating and planning phases of a project, and not to the monitoring phase which is the subject of this research. Those topics have now been included in the literature review, including function points, and Azzeh's fuzzy software assessments, but not integrated into this research study as they are applicable to the estimation phase of a project not the execution phase. Those topics confirmed that all known works on applying CI concepts to project management addressed the planning phases, not the monitoring phase of the execution of a project. The attendees at

CONISAR offered a number of suggestions for future expansions of this research such as scope trends over time, or multiple person inputs to one activity. These suggestions were not addressed in this research, but do demonstrate the potential for future research into CI methods applied to project management.

Appendix B - Evaluation Package

In this appendix on the next twenty-three pages is the evaluation package that contained the demonstration scenarios that were reviewed and critiqued by the subject matter experts. The package given to the subject matter experts contained six scenarios, each with 46 line items spread over 4 calendar months. It was reproduced on 11x17 paper using 12pt Times New Roman so that the Gantt charts would be readable. When that instrument was cut and pasted into this dissertation, the font was changed to 11pt Times New Roman so that it fit within the page layout requirements of the dissertation.

Evaluation of the Design
of new methods in
Project Management

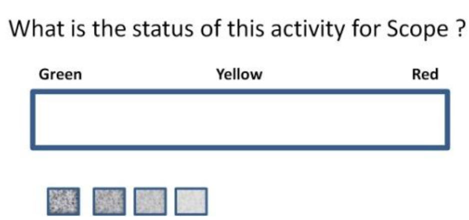
Thank you for your time and support of this research. You are being asked to review and comment upon the design of new methods to help manage projects.

These methods are still in the design stage. The purpose of the methods is to give Project Managers better insights into the status of constraints when monitoring the execution phase of a project. The PMI definitions for project terms will be used throughout this research.

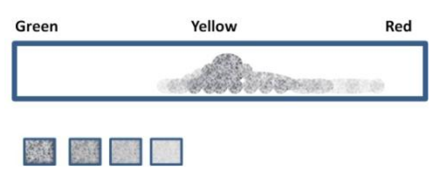
You will be given the management reports for one project. One set of reports are a snapshot along an 18 week time line, taken as shown in the table. Each snapshot of the project at a given point in time is labeled a "scenario".

The project starts in June and ends in October. The Earned Value reports and Gantt charts were generated using Microsoft project. An 11x17 copy of the Gantt chart will be provided. There are about 2 questions for each scenario about the status of the project.

After your review and analysis of the six scenarios, you will get an extra chart for each scenario (6 extra charts). You will be asked additional questions about these new charts and the new method, referred to as the fuzzy scope status. It is expected that the data will be collected using a web gadget that looks like the one on the left.



Web Gadget Before



Web Gadget After Collecting Data

The design on the right represents after the data has been collected for the scope status for an activity. The web gadget scale uses the stoplight colors for reporting status: Green, Yellow, Red.

Questions to Establish Background of Subject Matter Expert

Background Questions

Subject Matter Expert Identifier Number (*for confidentiality*): **SME-________**

1. Name: _______________*Withheld for Confidentiality*_____________________

2. Current Job Title: ____

3. How many years of experience do you have as a project manager? ___________

4. How long have you been PMP certified? __________

5. Have you managed commercial projects ? **Yes / No**

6. Have you managed government or non-profit projects ? **Yes / No**

Evaluation of the Design
of a new
Project Management Method

The Gantt charts were generated using Microsoft project.

The 11x17 copies of the Gantt charts are attached.

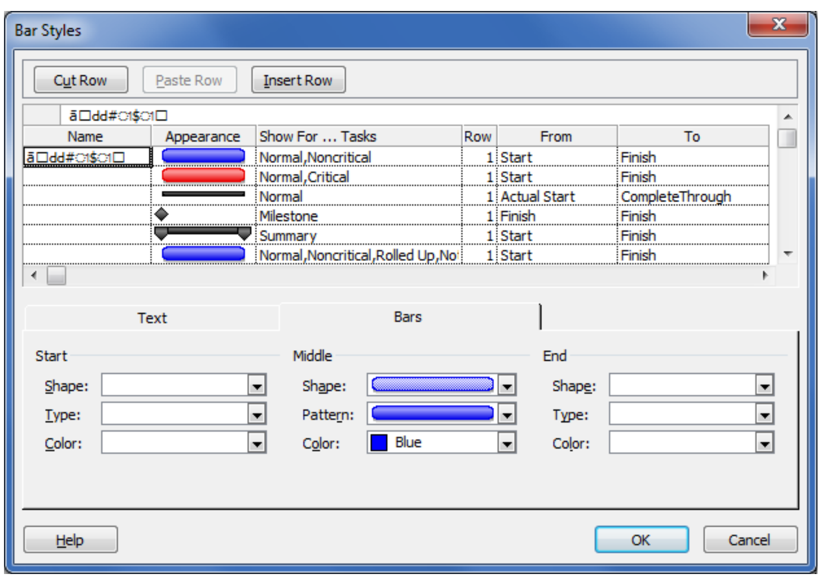
Scenario	Date the Reports were generated	Description
1	1 August	Start of the project's Execution phase. The Initiation and Planning phases are complete.
2	6 August	Monday after first week of project execution
3	13 August	Project execution. Some Analysis & Design tasks are done.
4	20 August	Project execution. All analysis and designs are complete.
5	3 September	All analysis and designs are complete. Software coding tasks are proceeding according to plan.
6	24 September	Almost finished with project execution. Systems testing starting.

The meaning of the bars in the Gantt charts are:

RED = Critical Path

BLUE = Completed Task

BLACK bar inside BLUE = Work Completed



Scenarios for the Evaluation of a New Management Tool

Background for all Scenarios:

ABC Inc. is a company that writes software to gather statistics from wide area network (WAN) equipment, such as Cisco routers. They sell this product to customers so the customer can manage their own WAN networks. As such, ABC Inc. must constantly keep up with the new equipment that their customers are using or intending to purchase from hardware vendors.

*Note: the ABC Inc. product is for the **WAN** market. It does not compete with network monitors, such as Microsoft's (netmon.exe) or WireShark, which are LAN based.*

The ABC Inc. product is called the *Network Monitor Tool*. The *Network Monitor Tool* gathers data by sending a request to network devices over a TCP/IP network using the Simple Network Management Protocol (SNMP). Data is transmitted using the industry standard Abstract Syntax Notation 1 (ASN.1). All makers of networking equipment follow these industry standards, however, they might make custom tweaks or changes that are unique to their equipment. These adaptations would be documented in their manuals and sample software. Sometimes their manuals are incomplete or wrong.

The data gathered by the ABC Inc. *Network Monitor Tool* is stored in a database on a server. This allows the users to query the database using web based GUI interfaces written by ABC Inc. to determine the state of their network. This database is a proprietary design of ABC Inc. The ABC Inc. software, database, and server hardware are sold as a turnkey product.

ABC Inc. must order and install equipment from network manufacturers into the ABC workplace so that ABC software developers can test their SNMP code. In some cases, this means getting "beta" hardware and firmware from the network manufacturers.

The Project:

ABC is writing software to add the capability to manage three new pieces of networking equipment. This is a major enhancement of the ABC base product to coincide with expected customer installations of these new network devices. The ABC software will support two new routers that are being rolled out 3rd Quarter (July-Aug-Sept) of this year by their respective companies (Cisco and Juniper), and one new switch manufactured by Nortel that will reach the customer market in the 4th Quarter. ABC Inc.'s relationship with these three manufacturers is excellent, and hardware delivery is normally on time.

ABC Inc. is aware that Nortel had filed for Chapter 11 bankruptcy protection, but is not concerned because the Nortel switching business unit had been sold to Ciena Corp. The management of ABC Inc. fully expects that Nortel's switching equipment will continue to be sold and supported under Ciena.

Project Start Date: 4 June 2012
Completion Date: 30 October 2012

Scenario 1

Current Date: 1 August 2012 (start of project execution)

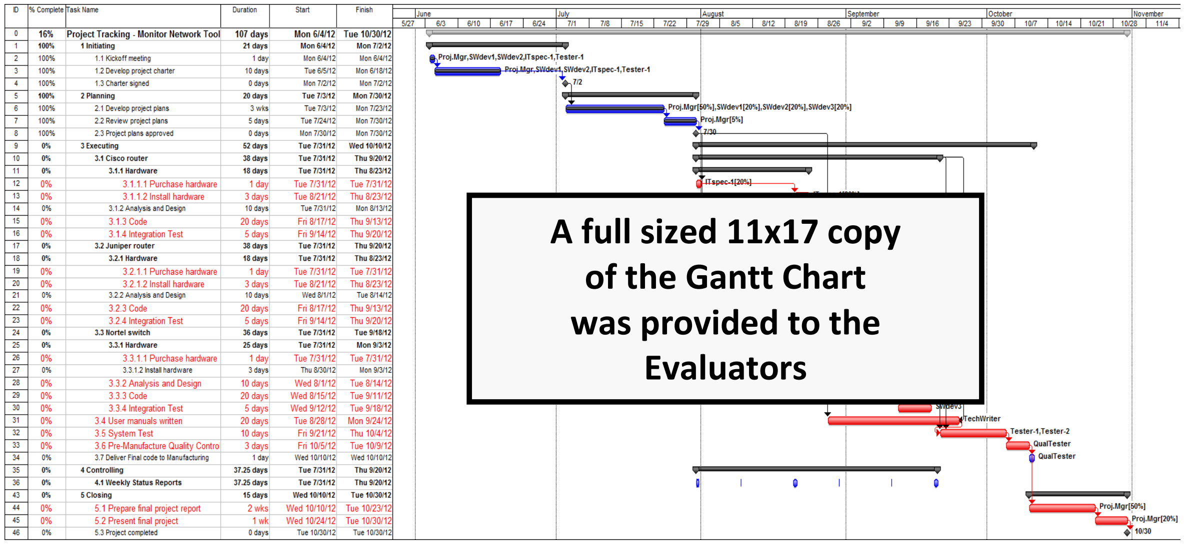
Activities completed at this point in time: The Project Charter has been signed, the project manager has completed the project plans and is in the process of reviewing them with the organization.

Current Activities: The Gantt chart shows that the hardware from the 3 manufacturers needs to be ordered. This was done, but the Gantt chart has not been updated with that information. The analysis and design activities for the software to support 3 pieces of networking equipment (Cisco, Juniper, and Nortel) has started and is in progress.

Status of Activities:

Activity ID	Task Name	Cost Status	Schedule Status	Comments
14	3.1.2 Analysis and Design Cisco	Green	Green	Just started
21	3.2.2 Analysis and Design Juniper	Green	Green	Just started
28	3.3.2 Analysis and Design Nortel	Green	Green	Starts today

The Gantt chart:



Sc1-Q.1. With the information provided below in the Gantt charts and in the Earned Value report on the next page, would you change the reported status for cost or schedule of activities 14, 21, or 28?

The Earned Value metrics:

	Earned Value as of 1 August								
ID	Task Name	PV Planned Value	EV Earned Value	CV	CV %	CPI	BAC	EAC	VAC
0	Project - Monitor Network Tool	$16,811.85	$14,886.00	$0.00	0%	1	$62,591.80	$62,591.80	$0.00
1	**Initiating**	$8,836.00	$8,836.00	$0.00	0%	1	$8,836.00	$8,836.00	$0.00
2	Kickoff meeting	$1,456.00	$1,456.00	$0.00	0%	1	$1,456.00	$1,456.00	$0.00
3	Develop project charter	$7,380.00	$7,380.00	$0.00	0%	1	$7,380.00	$7,380.00	$0.00
4	Charter signed	$0.00	$0.00	$0.00	0%	0	$0.00	$0.00	$0.00
5	**Planning**	$6,050.00	$6,050.00	$0.00	0%	1	$6,050.00	$6,050.00	$0.00
6	Develop project plans	$5,940.00	$5,940.00	$0.00	0%	1	$5,940.00	$5,940.00	$0.00
7	Review project plans	$110.00	$110.00	$0.00	0%	1	$110.00	$110.00	$0.00
8	Project plans approved	$0.00	$0.00	$0.00	0%	0	$0.00	$0.00	$0.00
9	**Executing**	$1,848.00	$0.00	$0.00	0%	0	$43,720.00	$43,720.00	$0.00
10	Cisco router	$904.00	$0.00	$0.00	0%	0	$12,800.00	$12,800.00	$0.00
11	Cisco Hardware	$120.00	$0.00	$0.00	0%	0	$880.00	$880.00	$0.00
12	Purchase hardware	$40.00	$0.00	$0.00	0%	0	$40.00	$40.00	$0.00
13	Install hardware	$0.00	$0.00	$0.00	0%	0	$120.00	$120.00	$0.00
14	Analysis and Design	$784.00	$0.00	$0.00	0%	0	$3,920.00	$3,920.00	$0.00
15	Code	$0.00	$0.00	$0.00	0%	0	$6,400.00	$6,400.00	$0.00
16	Integration Test	$0.00	$0.00	$0.00	0%	0	$1,600.00	$1,600.00	$0.00
17	Juniper router	$512.00	$0.00	$0.00	0%	0	$12,800.00	$12,800.00	$0.00
18	Hardware	$120.00	$0.00	$0.00	0%	0	$880.00	$880.00	$0.00
19	Purchase hardware	$40.00	$0.00	$0.00	0%	0	$40.00	$40.00	$0.00
20	Install hardware	$0.00	$0.00	$0.00	0%	0	$120.00	$120.00	$0.00
21	Analysis and Design	$392.00	$0.00	$0.00	0%	0	$3,920.00	$3,920.00	$0.00
22	Code	$0.00	$0.00	$0.00	0%	0	$6,400.00	$6,400.00	$0.00
23	Integration Test	$0.00	$0.00	$0.00	0%	0	$1,600.00	$1,600.00	$0.00
24	Nortel switch	$432.00	$0.00	$0.00	0%	0	$10,280.00	$10,280.00	$0.00
25	Hardware	$120.00	$0.00	$0.00	0%	0	$1,160.00	$1,160.00	$0.00
26	Purchase hardware	$40.00	$0.00	$0.00	0%	0	$40.00	$40.00	$0.00
27	Install hardware	$0.00	$0.00	$0.00	0%	0	$120.00	$120.00	$0.00
28	Analysis and Design	$312.00	$0.00	$0.00	0%	0	$3,120.00	$3,120.00	$0.00
29	Code	$0.00	$0.00	$0.00	0%	0	$4,800.00	$4,800.00	$0.00
30	Integration Test	$0.00	$0.00	$0.00	0%	0	$1,200.00	$1,200.00	$0.00
31	***User manuals written***	$0.00	$0.00	$0.00	0%	0	$3,520.00	$3,520.00	$0.00
32	***System Test***	$0.00	$0.00	$0.00	0%	0	$3,520.00	$3,520.00	$0.00
33	***Pre-Manufacture Quality Control***	$0.00	$0.00	$0.00	0%	0	$600.00	$600.00	$0.00
34	***Deliver to Manufacturing***	$0.00	$0.00	$0.00	0%	0	$200.00	$200.00	$0.00
35	**Controlling**	$77.85	$0.00	$0.00	0%	0	$1,345.80	$1,345.80	$0.00
36	Weekly Status Reports	$77.85	$0.00	$0.00	0%	0	$1,345.80	$1,345.80	$0.00
43	**Closing**	$0.00	$0.00	$0.00	0%	0	$2,640.00	$2,640.00	$0.00
44	Prepare final project report	$0.00	$0.00	$0.00	0%	0	$2,200.00	$2,200.00	$0.00
45	Present final project	$0.00	$0.00	$0.00	0%	0	$440.00	$440.00	$0.00
46	Project completed	$0.00	$0.00	$0.00	0%	0	$0.00	$0.00	$0.00

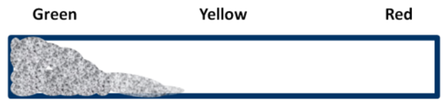
A reminder of the PMI definitions:

> ***CPI is the ratio of Earned Value to Actual cost.***
> ***CPI less than one is over budget.***
> ***CPI greater than one is under budget.***

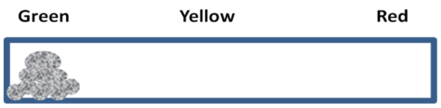
Scenario 1 - Additional Scope Information

The computational intelligence monitoring method gathered information from the software developers for activities 14, 21, and 28. This reflects their opinion as to the status of the scope constraint for those activities.

Scope Data for Activity ID=14



Scope Data for Activity ID=21



Scope Data for Activity ID=28

Activity ID	**Task Name**	**Fuzzy Scope Status**	**Comments**
14	3.1.2 Analysis and Design Cisco	Green	Just started
21	3.2.2 Analysis and Design Juniper	Green	Just started. Concerns about database.
28	3.3.2 Analysis and Design Nortel	Green	Starts today

Sc1-Q.2. With the new information provided, does this change your determination of the status of activities 14, 21, or 28 ?

Sc1-Q.3. What might a project manager do with information provided by the Fuzzy Scope Status data for activity 21 ? Should activity 21 be escalated to Yellow ?

Sc1-Q.4. Would the visual inputs aid in managing the project ?

Sc1-Q.5. Does the bankruptcy sale of Nortel's switching business to Ciena change your opinion of the status of the scope data for activity 28 ?

Explanation for Activity 21 Scope status
The software developer on the Juniper router tasks learned that the assigned DBA knows Microsoft SQL Server, but one of the business analysts told him that the likely customers for the ABC product will only accept Linux or Unix with Oracle as the database.
The organization has a Senior DBA trained on Oracle, but availability is unknown. The Senior DBA gets paid more per hour than the assigned DBA.

Scenario 2

Current Date: 6 August 2012 (Monday after first week of project execution)

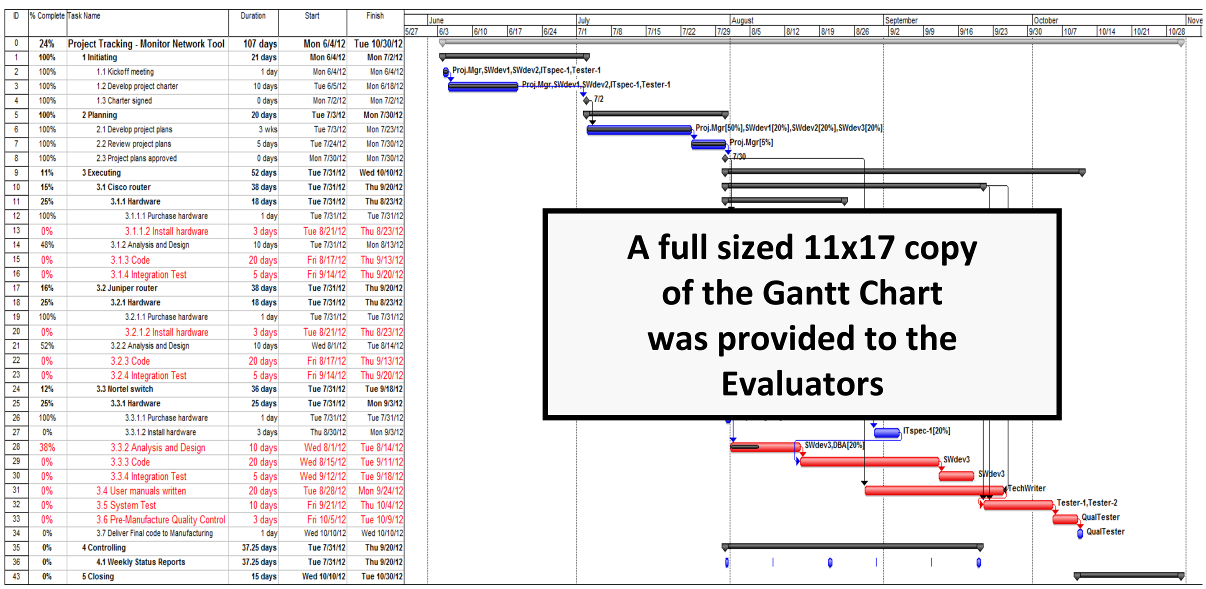
Additional Activities completed since last status report: The ordering of hardware from the 3 manufacturers has been done.

Current Activities: All three Analysis & Design tasks are about half completed.

Status of Activities: Cost & Schedule status for these activities was determined to be:

Activity ID	Task Name	Cost Status	Schedule Status	Comments
14	3.1.2 Analysis and Design Cisco	Green	Green	Half complete
21	3.2.2 Analysis and Design Juniper	Green	Green	Half complete
28	3.3.2 Analysis and Design Nortel	Green	Green	Little over one third complete

The Gantt chart:



Sc2-Q.1. Based on the information provided in the Gantt chart and Earned Value report on the next page, would you change the status of activities 14, 21, or 28 ?

The Earned Value metrics:

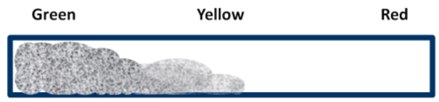
	Earned Value as of 6 August								
ID	Task Name	PV Planned Value	EV Earned Value	CV	CV %	CPI	BAC	EAC	VAC
0	Project Tracking - Monitor	$20,564.25	$20,208.33	$646.33	3%	1.03	$62,591.80	$60,589.89	$2,001.91
1	**Initiating**	$8,836.00	$8,836.00	$0.00	0%	1.00	$8,836.00	$8,836.00	$0.00
2	Kickoff meeting	$1,456.00	$1,456.00	$0.00	0%	1.00	$1,456.00	$1,456.00	$0.00
3	Develop project charter	$7,380.00	$7,380.00	$0.00	0%	1.00	$7,380.00	$7,380.00	$0.00
4	Charter signed	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00
5	**Planning**	$6,050.00	$6,050.00	$0.00	0%	1.00	$6,050.00	$6,050.00	$0.00
6	Develop project plans	$5,940.00	$5,940.00	$0.00	0%	1.00	$5,940.00	$5,940.00	$0.00
7	Review project plans	$110.00	$110.00	$0.00	0%	1.00	$110.00	$110.00	$0.00
8	Project plans approved	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00
9	**Executing**	$5,496.00	$5,322.33	$646.33	12%	1.14	$43,720.00	$38,410.73	$5,309.27
10	***Cisco router***	$2,200.00	$2,098.33	($81.67)	-4%	0.96	$12,800.00	$13,298.17	($498.17)
11	Hardware	$240.00	$220.00	$0.00	0%	1.00	$880.00	$880.00	$0.00
12	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
13	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
14	Analysis and Design	$1,960.00	$1,878.33	($81.67)	-4%	0.96	$3,920.00	$4,090.43	($170.43)
15	Code	$0.00	$0.00	$0.00	0%	0.00	$6,400.00	$6,400.00	$0.00
16	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,600.00	$1,600.00	$0.00
17	***Juniper router***	$1,808.00	$1,788.00	$0.00	0%	1.00	$12,800.00	$12,800.00	$0.00
18	Hardware	$240.00	$220.00	$0.00	0%	1.00	$880.00	$880.00	$0.00
19	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
20	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
21	Analysis and Design	$1,568.00	$1,568.00	$0.00	0%	1.00	$3,920.00	$3,920.00	$0.00
22	Code	$0.00	$0.00	$0.00	0%	0.00	$6,400.00	$6,400.00	$0.00
23	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,600.00	$1,600.00	$0.00
24	***Nortel switch***	$1,488.00	$1,436.00	$728.00	51%	2.03	$10,280.00	$5,068.41	$5,211.59
25	Hardware	$240.00	$240.00	$0.00	0%	1.00	$1,160.00	$1,160.00	$0.00
26	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
27	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
28	Analysis and Design	$1,248.00	$1,196.00	$728.00	61%	2.56	$3,120.00	$1,220.87	$1,899.13
29	Code	$0.00	$0.00	$0.00	0%	0.00	$4,800.00	$4,800.00	$0.00
30	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,200.00	$1,200.00	$0.00
31	***User manuals written***	$0.00	$0.00	$0.00	0%	0.00	$3,520.00	$3,520.00	$0.00
32	***System Test***	$0.00	$0.00	$0.00	0%	0.00	$3,520.00	$3,520.00	$0.00
33	***Pre-Manufac. Quality Control***	$0.00	$0.00	$0.00	0%	0.00	$600.00	$600.00	$0.00
34	***Deliver to Manufacturing***	$0.00	$0.00	$0.00	0%	0.00	$200.00	$200.00	$0.00
35	**Controlling**	$182.25	$0.00	$0.00	0%	0.00	$1,345.80	$1,345.80	$0.00
36	Weekly Status Reports	$182.25	$0.00	$0.00	0%	0.00	$1,345.80	$1,345.80	$0.00
43	**Closing**	$0.00	$0.00	$0.00	0%	0.00	$2,640.00	$2,640.00	$0.00

Note: ***CPI is the ratio of Earned Value to Actual cost.***
CPI less than one is over budget.
CPI greater than one is under budget.

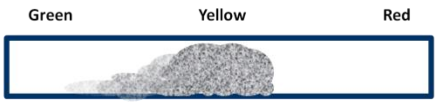
Scenario 2 - Additional Scope Information

The computational intelligence status monitoring method gathered information for the activities below. This is a subjective opinion as to the status of the scope constraint.

Scope Data for Activity ID=14



Scope Data for Activity ID=21



Scope Data for Activity ID=28

Activity ID	Task Name	Fuzzy Scope Status	Comments
14	3.1.2 Analysis and Design Cisco	Green	Half completed
21	3.2.2 Analysis and Design Juniper	Green	Half completed. Oracle DBA started work, needs to learn the application specifics
28	3.3.2 Analysis and Design Nortel	Yellow	Little over one third complete

Sc2-Q.2. With the new information provided, does this change your determination of the status of activity 14 ?

Sc2-Q.3. What might a project manager do with information provided by the scope data for activity 21 ?
Should activity 21 have been escalated to Yellow ?

Sc2-Q.4. What might a project manager do with information provided by the scope data for activity 28 ?

Explanation for Activity 28 Scope status
Task 28 The documentation from Nortel for their switch is labeled "BETA" and is dated over 90 days old. Experience with Nortel is that the final delivery will be different.

This has the potential for causing last minute code changes. History with Nortel switch suggests that the changes might vary from modest to significant.

Scenario 3

Current Date: 13 August 2012 (project execution)

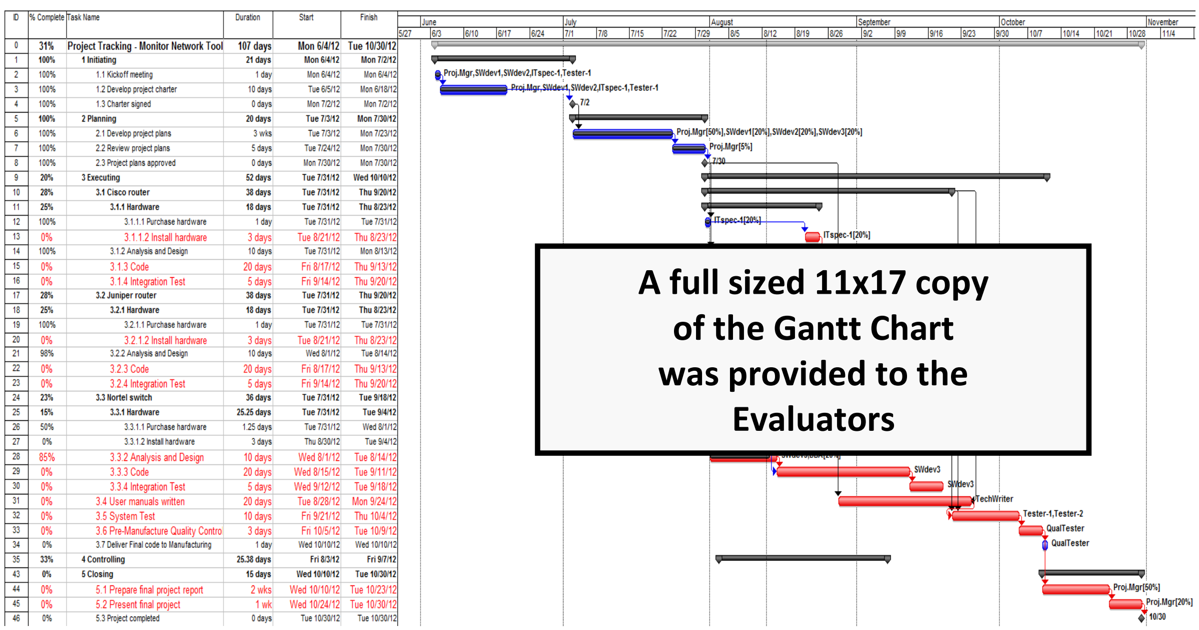
Additional Activities completed since last status report: Two of the three Analysis & Design tasks are completed.

Current Activities: The third Analysis & Design task is about completed.

Status of Activities: Cost & Schedule status for these activities was determined to be:

Activity ID	Task Name	Cost Status	Schedule Status	Comments
15	3.1.3 Code Cisco	Green	Green	
22	3.2.3 Code Juniper	Green	Green	
28	3.3.2 Analysis and Design Nortel	Green	Green	Mostly complete
29	3.3.2 Code Nortel	Green	Green	
31	3.4 Write User Manuals	Green	Green	Starts in two weeks

The Gantt chart:



Sc3-Q.1. With the information provided in the Gantt and Earned Value reports, would you change the status of activities 15, 22, 28, 29, or 31?

The Earned Value metrics:

	Earned Value as of 13 August								
ID	Task Name	PV Planned Value	EV Earned Value	CV	CV %	CPI	BAC	EAC	VAC
0	Project - Monitor Networ	$26,826.50	$25,967.16	$5,703.06	22%	1.28	$62,591.80	$48,845.02	$13,746.78
1	**Initiating**	$8,836.00	$8,836.00	$0.00	0%	1.00	$8,836.00	$8,836.00	$0.00
2	Kickoff meeting	$1,456.00	$1,456.00	$0.00	0%	1.00	$1,456.00	$1,456.00	$0.00
3	Develop project charter	$7,380.00	$7,380.00	$0.00	0%	1.00	$7,380.00	$7,380.00	$0.00
4	Charter signed	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00
5	**Planning**	$6,050.00	$6,050.00	$0.00	0%	1.00	$6,050.00	$6,050.00	$0.00
6	Develop project plans	$5,940.00	$5,940.00	$0.00	0%	1.00	$5,940.00	$5,940.00	$0.00
7	Review project plans	$110.00	$110.00	$0.00	0%	1.00	$110.00	$110.00	$0.00
8	Project plans approved	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00
9	**Executing**	$11,576.00	$10,707.06	$5,589.06	52%	2.09	$43,720.00	$20,898.26	$22,821.74
10	***Cisco router***	$4,360.00	$4,140.00	$1,960.00	47%	1.90	$12,800.00	$6,740.10	$6,059.90
11	Hardware	$440.00	$220.00	$0.00	0%	1.00	$880.00	$880.00	$0.00
12	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
13	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
14	Analysis and Design	$3,920.00	$3,920.00	$1,960.00	50%	2.00	$3,920.00	$1,960.00	$1,960.00
15	Code	$0.00	$0.00	$0.00	0%	0.00	$6,400.00	$6,400.00	$0.00
16	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,600.00	$1,600.00	$0.00
17	***Juniper router***	$3,968.00	$3,748.00	$1,568.00	42%	1.72	$12,800.00	$7,445.04	$5,354.96
18	Hardware	$440.00	$220.00	$0.00	0%	1.00	$880.00	$880.00	$0.00
19	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
20	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
21	Analysis and Design	$3,528.00	$3,528.00	$1,568.00	44%	1.80	$3,920.00	$2,177.78	$1,742.22
22	Code	$0.00	$0.00	$0.00	0%	0.00	$6,400.00	$6,400.00	$0.00
23	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,600.00	$1,600.00	$0.00
24	***Nortel switch***	$3,248.00	$2,819.06	$2,061.06	73%	3.72	$10,280.00	$2,764.12	$7,515.88
25	Hardware	$440.00	$167.06	($122.94)	-74%	0.58	$1,160.00	$2,013.62	($853.62)
26	Purchase hardware	$40.00	$20.00	($20.00)	-100%	0.50	$40.00	$80.00	($40.00)
27	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
28	Analysis and Design	$2,808.00	$2,652.00	$2,184.00	82%	5.67	$3,120.00	$550.59	$2,569.41
29	Code	$0.00	$0.00	$0.00	0%	0.00	$4,800.00	$4,800.00	$0.00
30	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,200.00	$1,200.00	$0.00
31	***User manuals written***	$0.00	$0.00	$0.00	0%	0.00	$3,520.00	$3,520.00	$0.00
32	***System Test***	$0.00	$0.00	$0.00	0%	0.00	$3,520.00	$3,520.00	$0.00
33	***Pre-Manufac. Quality Co***	$0.00	$0.00	$0.00	0%	0.00	$600.00	$600.00	$0.00
34	***Deliver to Manufacturing***	$0.00	$0.00	$0.00	0%	0.00	$200.00	$200.00	$0.00
35	**Controlling**	$364.50	$374.10	$114.00	30%	1.44	$1,345.80	$935.69	$410.11
43	**Closing**	$0.00	$0.00	$0.00	0%	0.00	$2,640.00	$2,640.00	$0.00

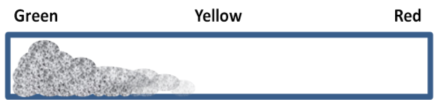
Note: ***CPI is the ratio of Earned Value to Actual cost.***
CPI less than one is over budget.
CPI greater than one is under budget.

Scenario 3 - Additional Scope Information

The computational intelligence status monitoring method gathered information for the activities below. This reflects the subjective opinion as to the status of the scope constraint.

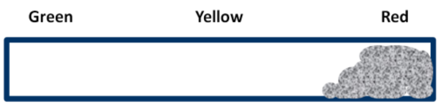
Scope Data for Activity ID=15

Scope Data for Activity ID=22



Scope Data for Activity ID=28

Scope Data for Activity ID=29



Scope Data for Activity ID=31

Activity ID	Task Name	Fuzzy Scope Status	Comments
15	3.1.3 Code Cisco	Green	
22	3.2.3 Code Juniper	Green	
28	3.3.2 Analysis and Design Nortel	Green	Mostly complete. BETA code issue resolved with final (production) product from Nortel
29	3.3.2 Code Nortel	Green	Might start one day late
31	3.4 Write User Manuals	Red	Starts in two weeks

Sc3-Q.2. With the new information provided, does this change your determination of the status of activities 15, 22, 28, 29, or 31 ?

Sc3-Q.3. Does the Fuzzy Scope Status aid in managing the project ?

Sc3-Q.4. What might a project manager do with information provided by the scope data for activity 31 ?

Explanation for Activity 31 Scope status
The Marketing Manager asked the Technical Writer how the translation of the manuals into French was going. The Marketing Manager emphasized that any products sold into Canada must be in both French & English. The Technical Writers only speak English.
The Profit margin for ABC Inc. is determined by the extra sales to Canadian companies.

Scenario 4

Current Date: 20 August 2012 (project execution)

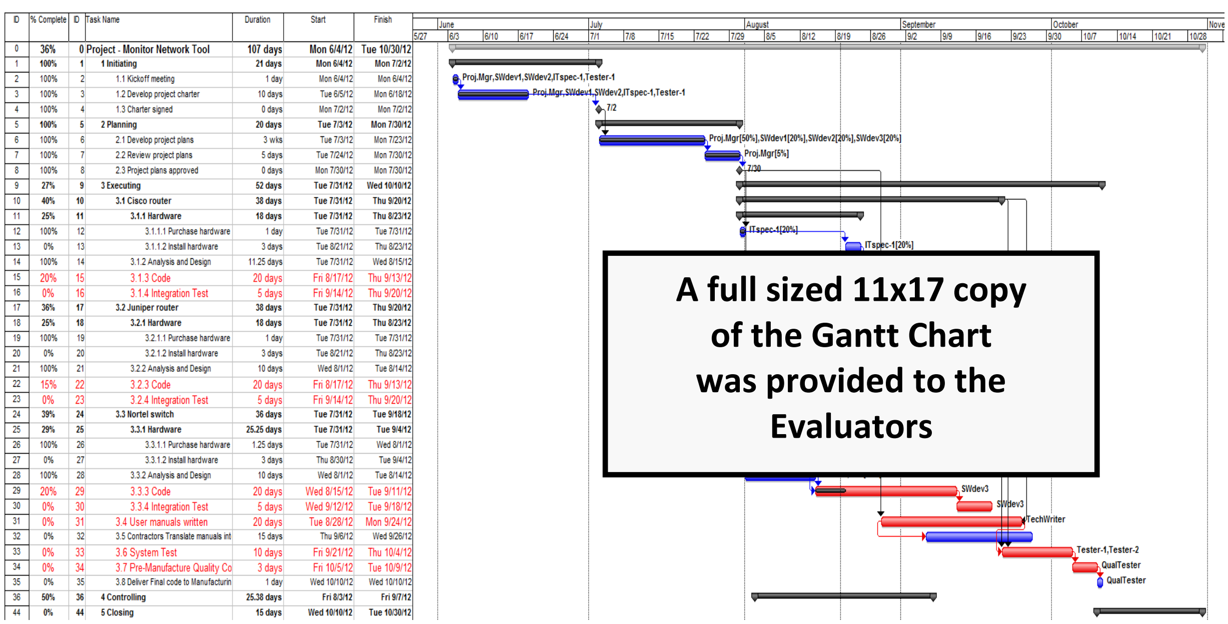
Activities completed since last status report: All analysis and designs are complete.

Current Activities: Coding tasks 15, 22, and 29 are proceeding according to plan. Additional funding was put into the budget for contractor to translate manuals into French. This will show up next week on the Earned Value reports.

Status of Activities: Cost & Schedule status for these activities was determined to be:

Activity ID	Task Name	Cost Status	Schedule Status	Comments
13	3.1.1.2 Hardware Install Cisco	Green	Green	Hardware was delivered. Task should start on schedule
20	3.2.1.2 Hardware Install Juniper	Green	Green	Hardware was delivered. Task should start on schedule
27	3.3.1.2 Hardware Install Nortel	Green	Green	Hardware should be on schedule. Task should start on schedule.
15	3.1.3 Code Cisco	Green	Green	
22	3.2.3 Code Juniper	Green	Green	
29	3.3.3 Code Nortel	Green	Green	
31	3.4 Write User Manuals	Green	Green	
32	3.5 Translate User Manuals to French	Green	Green	Additional funding for contractor

The Gantt chart:



Sc4-Q.1. With the information provided in the Gantt and Earned Value reports, would you change the status of activities 15, 22, 29, 31, or 32?

The Earned Value metrics:

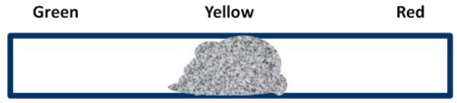
ID	Task Name	PV Planned Value	EV Earned Value	CV	CV %	CPI	BAC	EAC	VAC
	Earned Value as of 20 August								
0	Project - Monitor Networl	$30,550.00	$29,478.74	$6,805.44	23%	1.30	$62,591.80	$48,141.91	$14,449.89
1	**Initiating**	$8,836.00	$8,836.00	$0.00	0%	1.00	$8,836.00	$8,836.00	$0.00
2	Kickoff meeting	$1,456.00	$1,456.00	$0.00	0%	1.00	$1,456.00	$1,456.00	$0.00
3	Develop project charter	$7,380.00	$7,380.00	$0.00	0%	1.00	$7,380.00	$7,380.00	$0.00
4	Charter signed	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00
5	**Planning**	$6,050.00	$6,050.00	$0.00	0%	1.00	$6,050.00	$6,050.00	$0.00
6	Develop project plans	$5,940.00	$5,940.00	$0.00	0%	1.00	$5,940.00	$5,940.00	$0.00
7	Review project plans	$110.00	$110.00	$0.00	0%	1.00	$110.00	$110.00	$0.00
8	Project plans approved	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00
9	**Executing**	$15,120.00	$13,974.12	$6,616.12	47%	1.90	$43,720.00	$23,020.54	$20,699.46
10	***Cisco router***	$5,200.00	$4,780.00	$1,960.00	41%	1.70	$12,800.00	$7,551.46	$5,248.54
11	Hardware	$640.00	$220.00	$0.00	0%	1.00	$880.00	$880.00	$0.00
12	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
13	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
14	Analysis and Design	$3,920.00	$3,920.00	$1,960.00	50%	2.00	$3,920.00	$1,960.00	$1,960.00
15	Code	$640.00	$640.00	$0.00	0%	1.00	$6,400.00	$6,400.00	$0.00
16	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,600.00	$1,600.00	$0.00
17	***Juniper router***	$5,200.00	$4,780.00	$1,960.00	41%	1.70	$12,800.00	$7,551.46	$5,248.54
18	Hardware	$640.00	$220.00	$0.00	0%	1.00	$880.00	$880.00	$0.00
19	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
20	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
21	Analysis and Design	$3,920.00	$3,920.00	$1,960.00	50%	2.00	$3,920.00	$1,960.00	$1,960.00
22	Code	$640.00	$640.00	$0.00	0%	1.00	$6,400.00	$6,400.00	$0.00
23	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,600.00	$1,600.00	$0.00
24	***Nortel switch***	$4,720.00	$4,414.12	$2,696.12	61%	2.57	$10,280.00	$4,001.04	$6,278.96
25	Hardware	$640.00	$334.12	$44.12	13%	1.15	$1,160.00	$1,006.83	$153.17
26	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
27	Install hardware	$0.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
28	Analysis and Design	$3,120.00	$3,120.00	$2,652.00	85%	6.67	$3,120.00	$468.00	$2,652.00
29	Code	$960.00	$960.00	$0.00	0%	1.00	$4,800.00	$4,800.00	$0.00
30	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,200.00	$1,200.00	$0.00
31	***User manuals written***	$0.00	$0.00	$0.00	0%	0.00	$3,520.00	$3,520.00	$0.00
32	***Contractors Translate manuals into French***	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00
33	***System Test***	$0.00	$0.00	$0.00	0%	0.00	$3,520.00	$3,520.00	$0.00
34	***Pre-Manufac. Quality Cor***	$0.00	$0.00	$0.00	0%	0.00	$600.00	$600.00	$0.00
35	***Deliver to Manufacturing***	$0.00	$0.00	$0.00	0%	0.00	$200.00	$200.00	$0.00
36	**Controlling**	$544.00	$618.62	$189.32	31%	1.44	$1,345.80	$933.93	$411.87
44	**Closing**	$0.00	$0.00	$0.00	0%	0.00	$2,640.00	$2,640.00	$0.00

Note: ***CPI is the ratio of Earned Value to Actual cost.***
CPI less than one is over budget.
CPI greater than one is under budget.

Scenario 4 - Additional Scope Information

The computational intelligence status monitoring method gathered information for the activities below. This reflects the subjective opinion as to the status of the scope constraint.

Scope Data for Activity ID=15



Scope Data for Activity ID=22

Scope Data for Activity ID=29

Scope Data for Activity ID=31

Activity ID	Task Name	Fuzzy Scope Status	Comments
15	3.1.3 Code Cisco	Green	
22	3.2.3 Code Juniper	Yellow	Code builds failed over weekend
29	3.3.3 Code Nortel	Green	
31	3.4 Write User Manuals	Green	
32	3.5 Translate User Manuals to French	n/a	Contractor hired to translate

Sc4-Q.2. With the new information provided, does this change your determination of the status of activities 15, 29, or 31 ?

Sc4-Q.3. What might a project manager do with information provided by the scope data for activity 22 ?

Sc4-Q.4. If activity 22 is on the Time Critical Path, should the status change to RED ?

Explanation for Activity 22 Scope status
Task 22 The Friday night Code build failed. Programmers were called in on Saturday to fix code, but the rebuild on Saturday night also failed.

Team leader opinion is that additional code added to meet new requirements from one salesperson are the source of code build failures. Salesperson claims the additional work must be done for a major customer. Software developers are angry and not talking to the business analyst. The team leader is optimistic that the build will work on Monday night.

Scenario 5

Current Date: 3 September 2012 (project execution)

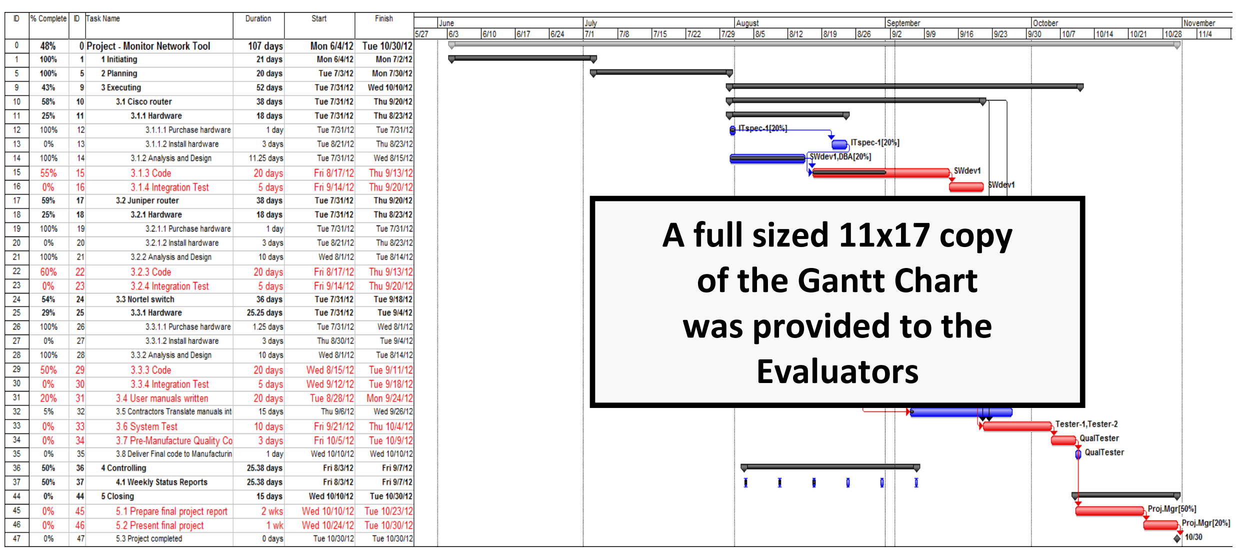
Activities completed since last status report: All analysis and designs are complete.

Current Activities: Coding tasks 15, 22, and 29 are proceeding according to plan. Additional funding was put into the budget for contractor to translate manuals into French. This will show up next week on the Earned Value reports.

Status of Activities: Cost & Schedule status for these activities was determined to be:

Activity ID	Task Name	Cost Status	Schedule Status	Comments
13	3.1.1.2 Hardware Install Cisco	Green	Green	Hardware was delivered. Task should start on schedule
20	3.2.1.2 Hardware Install Juniper	Green	Green	Hardware was delivered. Task should start on schedule
27	3.3.1.2 Hardware Install Nortel	Green	Green	Hardware should be on schedule. Task should start on schedule.
15	3.1.3 Code Cisco	Green	Green	
22	3.2.3 Code Juniper	Green	Green	
29	3.3.3 Code Nortel	Green	Yellow	
31 & 32	3.4 and 3.5 Write User Manuals	Green	Green	Additional funding for 3.5

The Gantt chart:



Sc5-Q.1. With the information provided in the Gantt and Earned Value reports, would you change the status of activities 15, 22, 29, or 31?

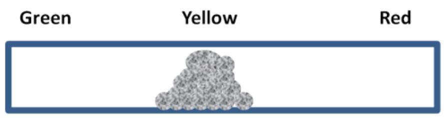
The Earned Value metrics:

ID	Task Name	PV Planned Value	EV Earned Value	CV	CV %	CPI	BAC	EAC	VAC
	Earned Value as of 3 September								
0	Project - Monitor Networl	$41,589.00	$38,022.74	$6,805.44	18%	1.22	$62,591.80	$51,388.91	$11,202.89
1	**Initiating**	$8,836.00	$8,836.00	$0.00	0%	1.00	$8,836.00	$8,836.00	$0.00
5	**Planning**	$6,050.00	$6,050.00	$0.00	0%	1.00	$6,050.00	$6,050.00	$0.00
9	**Executing**	$25,800.00	$22,518.12	$6,616.12	29%	1.42	$43,720.00	$30,874.49	$12,845.51
10	***Cisco router***	$8,640.00	$7,980.00	$1,960.00	25%	1.33	$12,800.00	$9,656.14	$3,143.86
11	Hardware	$880.00	$220.00	$0.00	0%	1.00	$880.00	$880.00	$0.00
12	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
13	Install hardware	$120.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
14	Analysis and Design	$3,920.00	$3,920.00	$1,960.00	50%	2.00	$3,920.00	$1,960.00	$1,960.00
15	Code	$3,840.00	$3,840.00	$0.00	0%	1.00	$6,400.00	$6,400.00	$0.00
16	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,600.00	$1,600.00	$0.00
17	***Juniper router***	$8,640.00	$7,980.00	$1,960.00	25%	1.33	$12,800.00	$9,656.14	$3,143.86
18	Hardware	$880.00	$220.00	$0.00	0%	1.00	$880.00	$880.00	$0.00
19	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
20	Install hardware	$120.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
21	Analysis and Design	$3,920.00	$3,920.00	$1,960.00	50%	2.00	$3,920.00	$1,960.00	$1,960.00
22	Code	$3,840.00	$3,840.00	$0.00	0%	1.00	$6,400.00	$6,400.00	$0.00
23	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,600.00	$1,600.00	$0.00
24	***Nortel switch***	$7,640.00	$5,854.12	$2,696.12	46%	1.85	$10,280.00	$5,545.54	$4,734.46
25	Hardware	$1,160.00	$334.12	$44.12	13%	1.15	$1,160.00	$1,006.83	$153.17
26	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
27	Install hardware	$120.00	$0.00	$0.00	0%	0.00	$120.00	$120.00	$0.00
28	Analysis and Design	$3,120.00	$3,120.00	$2,652.00	85%	6.67	$3,120.00	$468.00	$2,652.00
29	Code	$3,360.00	$2,400.00	$0.00	0%	1.00	$4,800.00	$4,800.00	$0.00
30	Integration Test	$0.00	$0.00	$0.00	0%	0.00	$1,200.00	$1,200.00	$0.00
31	***User manuals written***	$880.00	$704.00	$0.00	0%	1.00	$3,520.00	$3,520.00	$0.00
32	***Translate manuals to Fren***	$0.00	$0.00	$0.00	0%	0.00	$4,000.00	$0.00	$4,000.00
33	***System Test***	$0.00	$0.00	$0.00	0%	0.00	$3,520.00	$3,520.00	$0.00
34	***Pre-Manufac. Quality Cor***	$0.00	$0.00	$0.00	0%	0.00	$600.00	$600.00	$0.00
35	***Deliver to Manufacturing***	$0.00	$0.00	$0.00	0%	0.00	$200.00	$200.00	$0.00
36	**Controlling**	$903.00	$618.62	$189.32	31%	1.44	$1,345.80	$933.93	$411.87
37	Weekly Status Reports	$903.00	$618.62	$189.32	31%	1.44	$1,345.80	$933.93	$411.87
44	**Closing**	$0.00	$0.00	$0.00	0%	0.00	$2,640.00	$2,640.00	$0.00
45	Prepare final project repo	$0.00	$0.00	$0.00	0%	0.00	$2,200.00	$2,200.00	$0.00
46	Present final project	$0.00	$0.00	$0.00	0%	0.00	$440.00	$440.00	$0.00
47	Project completed	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00

Note: ***CPI is the ratio of Earned Value to Actual cost.***
CPI less than one is over budget.
CPI greater than one is under budget.

Scenario 5 - Additional Scope Information

The computational intelligence status monitoring method gathered information for the activities below. This reflects subjective opinions as to the status of the scope constraint.



Scope Data for Activity ID=15

Scope Data for Activity ID=22

Scope Data for Activity ID=29

Scope Data for Activity ID=31

Activity ID	**Task Name**	**Fuzzy Scope Status**	**Comments**
15	3.1.3 Code Cisco	Yellow	Need simulator
22	3.2.3 Code Juniper	Green	Code build problem fixed
29	3.3.3 Code Nortel	Green	
31	3.4 Write User Manuals	Green	

Sc5-Q.2. With the new information provided, does this change your determination of the status of activities 22, 29, or 31 ?

Sc5-Q.3. What might a project manager do with information provided by the scope data for activity 15 ?

Explanation for Activity 15 Scope status
Upper management has decided that it would be nice to have a simulator for customer training on the Cisco router. This would act like the Cisco network device on a customer network and could be expanded later to include other equipment (Juniper, Nortel, Alcatel, ...), The belief is that customers could become familiar with the Network Monitor method, even though the customer has not received their network equipment yet.

Software developer thought this would cause the team to work on the weekends to add this extra requirement in order to make the schedule, so they flagged it as status "yellow".

This is a new requirement, and too late in the cycle to add, so it was recommended that it be postponed for a future release.

Scenario 6

Current Date: 24 September 2012 (near finished with project execution)

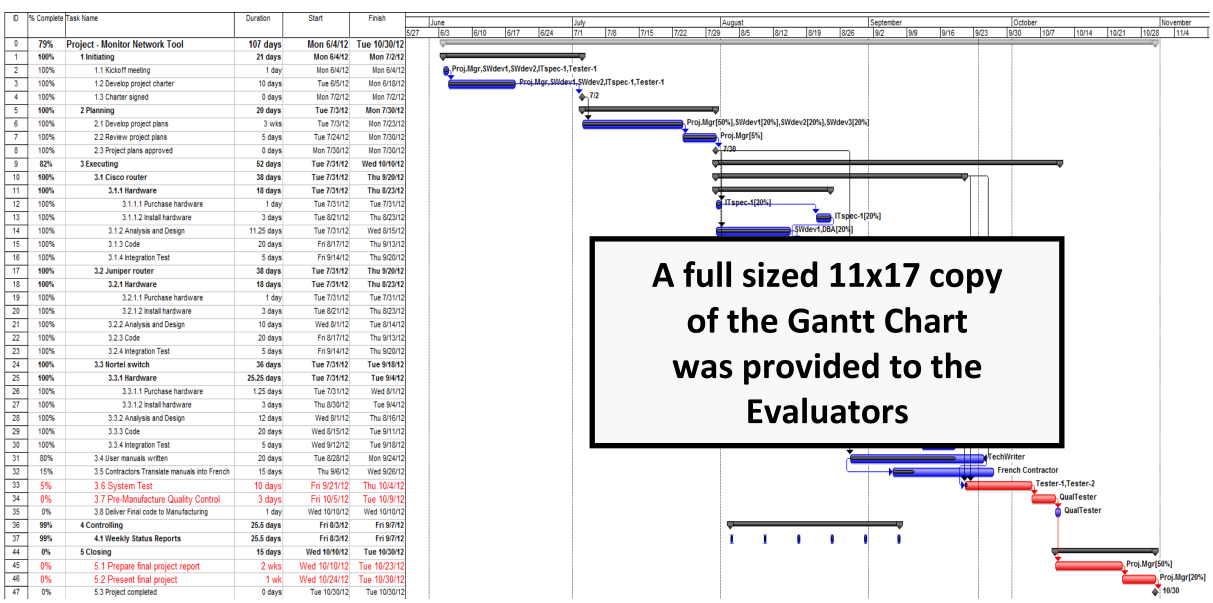
Activities completed since last status report: Coding tasks 15, 22, & 29 are complete.

Current Activities: Writing manuals on schedule. System tests started one day ago.

Status of Activities: Cost & Schedule status for these activities was determined to be:

Activity ID	Task Name	Cost Status	Schedule Status	Comments
15	3.1.3 Code Cisco	Done	Done	
22	3.2.3 Code Juniper	Done	Done	
29	3.3.3 Code Nortel	Done	Done	
31	3.4 Write User Manuals	Green	Green	
32	3.5 Write User Manuals in French	Green	Green	Green because not on Critical Path
33	3.6 System Test	Yellow	Yellow	Tests not going well.

The Gantt chart:



Sc6-Q.1. With the information provided in the Gantt and Earned Value reports, would you change the status of activities 31, 32, or 33?

The Earned Value metrics:

	Earned Value as of 24 September								
ID	Task Name	PV Planned Value	EV Earned Value	CV	CV %	CPI	BAC	EAC	VAC
0	Project - Monitor Networl	$56,335.80	$54,398.46	($453.09)	-1%	0.99	$62,591.80	$63,113.13	($521.33)
1	**Initiating**	$8,836.00	$8,836.00	$0.00	0%	1.00	$8,836.00	$8,836.00	$0.00
5	**Planning**	$6,050.00	$6,050.00	$0.00	0%	1.00	$6,050.00	$6,050.00	$0.00
9	**Executing**	$40,104.00	$38,872.00	($634.00)	-2%	0.98	$43,720.00	$44,433.07	($713.07)
10	***Cisco router***	$12,800.00	$12,800.00	$2,500.00	20%	1.24	$12,800.00	$10,300.00	$2,500.00
11	Hardware	$880.00	$880.00	$540.00	61%	2.59	$880.00	$340.00	$540.00
12	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
13	Install hardware	$120.00	$120.00	$0.00	0%	1.00	$120.00	$120.00	$0.00
14	Analysis and Design	$3,920.00	$3,920.00	$1,960.00	50%	2.00	$3,920.00	$1,960.00	$1,960.00
15	Code	$6,400.00	$6,400.00	$0.00	0%	1.00	$6,400.00	$6,400.00	$0.00
16	Integration Test	$1,600.00	$1,600.00	$0.00	0%	1.00	$1,600.00	$1,600.00	$0.00
17	***Juniper router***	$12,800.00	$12,800.00	$2,500.00	20%	1.24	$12,800.00	$10,300.00	$2,500.00
18	Hardware	$880.00	$880.00	$540.00	61%	2.59	$880.00	$340.00	$540.00
19	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
20	Install hardware	$120.00	$120.00	$0.00	0%	1.00	$120.00	$120.00	$0.00
21	Analysis and Design	$3,920.00	$3,920.00	$1,960.00	50%	2.00	$3,920.00	$1,960.00	$1,960.00
22	Code	$6,400.00	$6,400.00	$0.00	0%	1.00	$6,400.00	$6,400.00	$0.00
23	Integration Test	$1,600.00	$1,600.00	$0.00	0%	1.00	$1,600.00	$1,600.00	$0.00
24	***Nortel switch***	$10,280.00	$10,280.00	($810.00)	-8%	0.93	$10,280.00	$11,090.00	($810.00)
25	Hardware	$1,160.00	$1,160.00	$750.00	65%	2.83	$1,160.00	$410.00	$750.00
26	Purchase hardware	$40.00	$40.00	$0.00	0%	1.00	$40.00	$40.00	$0.00
27	Install hardware	$120.00	$120.00	$0.00	0%	1.00	$120.00	$120.00	$0.00
28	Analysis and Design	$3,120.00	$3,120.00	($1,560.00)	-50%	0.67	$3,120.00	$4,680.00	($1,560.00)
29	Code	$4,800.00	$4,800.00	$0.00	0%	1.00	$4,800.00	$4,800.00	$0.00
30	Integration Test	$1,200.00	$1,200.00	$0.00	0%	1.00	$1,200.00	$1,200.00	$0.00
31	***User manuals written***	$3,520.00	$2,816.00	$0.00	0%	1.00	$3,520.00	$3,520.00	$0.00
32	***Translate manuals to Frer***	$0.00	$0.00	($4,000.00)	0%	0.00	$0.00	$0.00	$0.00
33	***System Test***	$704.00	$176.00	($824.00)	-468%	0.18	$3,520.00	$20,000.00	($16,480.00)
34	***Pre-Manufac. Quality Cor***	$0.00	$0.00	$0.00	0%	0.00	$600.00	$600.00	$0.00
35	***Deliver to Manufacturing***	$0.00	$0.00	$0.00	0%	0.00	$200.00	$200.00	$0.00
36	**Controlling**	$1,345.80	$640.46	$180.91	28%	1.39	$1,345.80	$965.65	$380.15
37	Weekly Status Reports	$1,345.80	$640.46	$180.91	28%	1.39	$1,345.80	$965.65	$380.15
44	**Closing**	$0.00	$0.00	$0.00	0%	0.00	$2,640.00	$2,640.00	$0.00
45	Prepare final project repo	$0.00	$0.00	$0.00	0%	0.00	$2,200.00	$2,200.00	$0.00
46	Present final project	$0.00	$0.00	$0.00	0%	0.00	$440.00	$440.00	$0.00
47	Project completed	$0.00	$0.00	$0.00	0%	0.00	$0.00	$0.00	$0.00

Note: ***CPI is the ratio of Earned Value to Actual cost.***
CPI less than one is over budget.
CPI greater than one is under budget.

Scenario 6 - Additional Scope Information

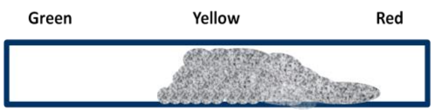
The computational intelligence status monitoring method gathered information for the activities below. This reflects subjective opinions as to the status of the scope constraint.

Scope Data for Activity ID=15

Scope Data for Activity ID=22

Scope Data for Activity ID=29

Scope Data for Activity ID=31



Scope Data for Activity ID=33

Activity ID	Task Name	Fuzzy Scope Status	Comments
15	3.1.3 Code Cisco	Green	Done
22	3.2.3 Code Juniper	Green	Done
29	3.3.3 Code Nortel	Green	Done
31	3.4 Write User manuals	Green	
32	3.5 Translate User manuals to French	n/a	Contractor
33	3.6 System Test	Red	

Sc6-Q.2. With the new information provided, does this change your determination of the status of activities 15, 22, 29, 31, or 32 ?

Sc6-Q.3. What might a project manager do with information provided by the scope data for activity 33 ?

Explanation for Activity 33 Scope status
Task 33 - System Test Manager had Test team work all weekend in anticipation of reassignment of Testers to new urgent project from a different department. System Testers only installed test equipment, but did not start testing because of lack of adequate test data. Testers believe that they need software "tools" written in order to generate enough test data.
Fuzzy visual input was YELLOW, but CI methods escalated to RED because it is a Critical Path activity.

7 - Conclusion

The computational intelligence status monitoring method was intended to gather information to help improve the project manager's understanding of the status of scope.

Q.7-1. **How important is the monitoring of scope?**

Q.7-2. **What is important about that?**

Q.7-3. **When managing projects, have you had issues with scope creep?**

Q.7-4. **What options for monitoring scope status do you have?**

Q.7-5. **If you could change one thing about the CI method, what would it be?**

Q.7-6. **What will that give you?**

Q.7-7. **What would the implementation of the CI method cost you?**

Q.7-8. **What can be improved?**

Q.7-9. **What would you hope for when monitoring scope?**

Thank you for your time and support of this research.

Appendix C - Towson University IRB Approval

TOWSON
UNIVERSITY

EXEMPTION NUMBER: 13-0X01

To: Joseph McGuighan

From: Institutional Review Board for the Protection of Human Subjects, Debi Gartland,Chair DG/wmp

Date: Tuesday, July 24, 2012

RE: Application for Approval of Research Involving the Use of Human Participants

Office of University Research Services
Towson University
8000 York Road
Towson, MD 21252-0001
t. 410 704-2236
f. 410 704-4494

Thank you for submitting an application for approval of the research titled, *A Fuzzy Computational Intelligence Approach to Monitoring Information Tehcnology Project Scope*

to the Institutional Review Board for the Protection of Human Participants (IRB) at Towson University.

Your research is exempt from general Human Participants requirements according to 45 CFR 46.101(b)(2). No further review of this project is required from year to year provided it does not deviate from the submitted research design.

If you substantially change your research project or your survey instrument, please notify the Board immediately.

We wish you every success in your research project.

CC: R. Hammell
File

Bibliography

ACM. (2012). *Herbert A. Simon - A. M. Turning Award Winner*. Retrieved July 18, 2012, from ACM A. M. Turing Award: http://amturing.acm.org/award_winners/simon_1031467.cfm

APM.org. (n.d.). *Membership of APM*. Retrieved May 16, 2012, from http://www.apm.org.uk/Individual

Asnar, Y., Giorgini, P., & Mylopoulos, J. (2011). Goal-driven risk assessment in requirements engineering. *Requirements Engineering*, 101-116.

Azzeh, M., Neagu, D., & Cowling, P. (2010). Fuzzy grey relational analysis for software effort estimation. *Empirical Sotware Engineering*, 60-90, 81.

Barnes, A., & Hammell, R. (2008). Determining Information Technology Project Status using Recognition-primed Decision Making Enabled Collaborative Agents for Simulating Teamwork (R-CAST). *Proceedings of CONISAR.* Phoenix, AZ: Conference on Information Systems Applied Research 2008.

Barranquero, T., & Guadarrama, S. (2010). Collecting Fuzzy Perceptions from Non-expert Users. *IEEE World Congress on Computational Intelligence July 18-23 2010.* Barcelona, Spain: IEEE.

Benson, R., Bugnitz, T., & Walton, W. (2004). *From Business Strategy to IT Action.* Hoboken, NJ: John Wiley & Sons.

Bergin, C. (2007, Jan 28). *Remembering the Mistakes of Challenger.* Retrieved December 1, 2011, from http://www.nasaspaceflight.com/2007/01/remembering-the-mistakes-of-challenger/

Boehm, B. (2010, April 5). *Constructive Cost Model 2.* Retrieved November 26, 2011, from http://csse.usc.edu/csse/TECHRPTS/2010/usc-csse-2010-503/usc-csse-2010-503.pdf

Boness, K., Finkelstein, A., & Harrison, R. (2011). A method for assessing confidence in requirements analysis. *Information and Software Technology*, 1084-1096.

Brooks, F. P. (1995). *The Mythical Man-Month: Essays on Software Engineering Anniversary Edition.* Crawfordsville, IN: Addison Wesley Longman, Inc.

Bulkeley, W. M. (1996, Nov 18). *Technology (A Special Report): Working Together - When Things Go Wrong: FoxMeyer Drug took a huge High-tech gamble: It didn't work.* Retrieved July 16, 2011, from Wall Street Journal (Eastern Edition): http://homepage.cs.uri.edu/courses/fall2007/csc305/Schedule/FoxMeyer3.pdf

Calleam, C. (2011). *Why IT Projects Fail*. Retrieved November 27, 2011, from http://calleam.com/WTPF/?page_id=3

Chen, H. (2011, March). Editorial: Design Science, Grand Challenges, and Societal Impacts. *ACM Transactions on Management Information Systems, 2*(1), 1:1-1:10.

Chen, H., Wu, Z., & Cudre-Maurox, P. (2012, May). Semantic Web Meets Computational Intelligence: State of the Art and Perspectives. *IEEE Computational Intelligence Magazine*, pp. 67-74.

Cook, J. (2007, July 17). *How to Spot a Failing Project.* Retrieved May 5, 2011, from CIO Magazine: http://www.cio.com/article/124309

Corbin, J., & Strauss, A. (2008). *Basics of Qualitative Research, 3e, Techniques and Procedures for Developing Grounded Theory.* Thousand Oaks, CA: Sage Publications, Inc.

COSMIC Method Publications. (2011, Mar 22). *COSMIC Functional Size Measurement Method*. Retrieved Nov 26, 2011, from http://www.cosmicon.com/methodV3.asp

Cox, E. (1999). *The Fuzzy Systems Handbook, Second Edition.* New York, NY: AP Professional.

DeMarco, A. (2011). *How to Estimate Software Projects*. Retrieved Nov 26, 2011, from http://www.pricesystems.com/products/popup_software.htm

DeMarco, T., & Lister, T. (2003). *Waltzing with Bears: Managing Risk on Software Projects.* New York, NY: Dorset House Publishing Co., Inc.

Deming, W. (1994). *The New Economics: For Industry, Government, Education.* Boston, MA: MIT Press.

Denning, P. J. (1997). A New Social Contract for Research. *Communications of the ACM*, 132-134.

Dhanji, Z. (2009, Nov 16). Retrieved Nov 17, 2009, from Linkedin.com: http://www.linkedin.com/groupAnswers?viewQuestionAndAnswers=&discussionID=9814221&gid=40431&commentID=8438167&trk=view_disc&ut=1ZLKW3iJ4_AB01

Dow, W., & Taylor, B. (2008). *Project Management Communications.* Indianapolis, IN: Wiley Publishing, Inc.

Drucker, P. F. (1993). *Management: Tasks, Responsibilities, Practices.* New York, NY: HarperCollins Publishers.

Eggen, D., & Witte, G. (2006, Aug 18). *Washington Post.* Retrieved April 13, 2011, from http://pqasb.pqarchiver.com/washingtonpost/access/1096595151.html

Feldman, J. (2012, February 12). Project Management Gets Lean. *InformationWeek*, pp. 29-32.

Ferrari, R., & Madhavji, N. (2008). Architecting-problems rooted in requirements. *Information and Software Technology*, 53-66.

Fleming, Q., & Koppelman, J. M. (2010). *Earned Value Project Management, Fourth Edition.* Newtown Square, PA: Project Management Institute, Inc.

Flyvbjerg, B., & Budzier, A. (2011, September). Why Your IT Project May Be Riskier Than You Think. *Harvard Business Review*. Retrieved from hbr.org/2011/09

Geerts, G. L. (2011). A design science research methodology and its application to accounting information systems research. *International Journal of Accounting Information Systems*, 142-151.

Gerosa, S. (2003). *Earned Value Management: How to avoid the 90 percent complete syndrome*. Retrieved July 29, 2011

Gido, J., & Clements, J. P. (2009). *Successful Project Management.* Mason, OH: South-Western Cengage Learning.

Hancock, D. (2010). *Tame, Messy, and Wicked Risk Leadership -- Advances in Project Management.* Burlington, VT: Gower Publishing Company.

Hardy-Vallee, B. (2012, February 12). *Most Projects Fail Because The Employees Working On Them Just Don't Care.* Retrieved July 30, 2012, from Business Insider: articles.businessinsider.com/2012-02-07/strategy/31032893

Haynes, S. R., Carroll, J. M., Kannampallil, T. G., Xiao, L., & Bach, P. M. (2009). Design Research as Explanation: Perceptions in the Field. *CHI 2009 - Proceedings of the 27th International Conference on Human Factors in*

Computing Systems (pp. 1121-1130). Boston, MA: Association for Computing Machinery (ACM).

Heller, R. (2002). *An Introduction to Function Point Analysis.* Retrieved Nov 26, 2011, from http://www.qpmg.com/fp-intro.htm

Hevner, A. R., & March, S. T. (2003, November). The Information Systems Research Cycle. *IEEE Computer*, pp. 111-113.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 75-105.

Hobbs, B. (2007). *The Multi-Project PMO: A Global Analysis of the Current State of Practice.* Newtown Square, PA: Project Management Institute.

Hulett, D. (2009). *Practical Schedule Risk Analysis.* Farnham, United Kingdom : Gower Publishing.

IEEE. (2011, November 28). *About the IEEE*. Retrieved August 2012, from http://cis.ieee.org/scope.html

ISBSG, The International Software Benchmarking Standards Group. (2011, June 16). *Repositories of IT history data (software metrics)*. Retrieved Nov 26, 2011, from Software Benchmarking Standards: www.isbsg.org

ISO/IEC. (2007, 02 15). Information Technology - Software Measurement - Function size measurement. *International Standard*. Geneva, Switzerland: International Organization for Standardization (ISO).

Janesick, V. (2000). The Choreography of Qualititive Research Design. In N. K. Denzin, & Y. S. Lincoln, *Handbook of Qualitative Research* (pp. 379-385). Thousands Oaks, CA: Sage Publications, Inc.

Kangari, R., & Riggs, L. S. (1989). Construction Risk Assessment by Linguistics. *IEEE Transactions on Engineering Management*, 126-131.

Kaplan, R. S., & Norton, D. P. (1996). *The Balanced Scorecard.* Boston, MA: Harvard Business School Press.

Kerzner, H. (2011). *Project Management Metrics, KPIs, and Dashboards.* Hoboken, NJ: John Wiley & Sons, Inc.

Keung, J., & Kitchenham, B. (2008). Analogy-X: Providing Statistical Inference to Analogy-Based Software Cost Eestimation. *IEEE Transactions on Software Engineering*, 471.

Keyes, J. (2011). *Implementing the Project Management Balanced Scorecard.* Boca Raton, FL: Taylor & Francis Group, LLC.

Khedr, M. (2006). Project Risk Management Using Monte Carlo Simulation. *AACE International Transactions*, 02.01-02.10.

Klakegg, O. J., Williams, T., Walker, D., Andersen, B., & Magnussen, O. M. (2010). *Early Warning Signs in Complex Projects.* Newtown Square, PA: Project Management Institute, Inc.

Klir, G. J., St.Clair, U., & Yuan, B. (1997). *Fuzzy Set Theory Foundations and Applications.* Upper Saddle River, NJ: Prentice Hall.

Kosko, B. (1993). *Fuzzy Thinking.* New York, NY: Hyperion.

Kuechler, W., & Vaishnavi, V. (2012, June). A Framework for Theory Development in Design Science Research: Multiple Perspectives. *Journal of the Association for Information Systems, 13*(6), 395-423.

Kwak, Y. H., & Anbari, F. T. (2012). History, Practices, and Future of Earned Value Management in Government: Perspectives from NASA. *Project Management Journal*, 77-90.

Levinson, M. (2008a, July 23). *Project Management: The 14 Most Common Mistakes IT Departments Make.* Retrieved July 5, 2011, from CIO Magazine: http://www.cio.com/article/438930

Levinson, M. (2008b, August 1). *Common Project Management Metrics Doom IT Departments to Failure.* Retrieved July 5, 2011, from CIO Magazine: http://www.cio.com/article/440721

Levinson, M. (2009, June 18). *Recession Causes Rising I.T. Project Failure Rates.* Retrieved July 5, 2011, from CIO Magazine: http://www.cio.com/article/495306

Lewis, J. P. (2001). *Industrial Project Management.* New York, NY: The McGraw-Hill Companies, Inc.

Li, J., Moselhi, O., & Alkas, S. (2006, November). Forecasting Project Status by Using Fuzzy Logic. *Journal of Construction Engineering and Management*, Vol. 132, No. 11.

Li, Y.-H., & Hu, Y.-Q. (2006). A Model of Multilevel Fuzzy Comprehensive Evaluation for Investment Risk. *International Conference on Machine Learning and Cybernetics*, 1942-1947.

Loch, C., De Meyer, A., & Pich, M. (2006). *Managing the Unknown: a New Approach to Managing High Uncertainty and Risk in Projects.* Hoboken, NJ: John Wiley & Sons.

Lucas, G. (1999, Feb 18). *Computer Bumbling Costs State $1 Billion.* Retrieved 02 17, 2011, from The San Francisco Chronicle: http://www.sfgate.com/cgi-bin/article.cgi?f=/c/a/1999/02/18/MN86384.DTL

March, S. T., & Smith, G. F. (1995). Design and natural science research on Information Technology. *Decision Support Systems*, 251-266.

Martin, C. L., Leboeuf, J., Yanez, C., & Gutierrez, A. (2005). Software Development Effort Estimation Using Fuzzy Logic: A Case Study. *Proceedings of the Sixth Mexican International Conference on Computer Science.* Puebla, Mexico: IEEE Computer Society.

Maude, F. (2011, Sept 22). *Dismantling the NHS National Programme for IT*. Retrieved Nov 27, 2011, from http://mediacentre.dh.gov.uk/2011/09/22

McDougall, P. (2006, Oct 16). *Information Week.* Retrieved January 19, 2011, from http://www.informationweek.com/story/showArticle.jhtml?articleID=193302693

McManus, J. (2004). *Risk Management in Software Development Projects.* Burlington, MA: Elsevieer Butterworth-Heinemann.

McNeill, D., & Freiberger, P. (1993). *Fuzzy Logic.* New York: Simon & Schuster.

McNeill, F. M., & Thro, E. (1994). *Fuzzy Logic A Practical Approach.* Chestnut Hill, MA: AP Professional.

Mendel, J., & Wu, D. (2010). *Perceptual Computing: Aiding People in Making Subjective Judgments.* Hoboken, NJ: John Wiley & Sons.

Meredith, J. R., & Mantel, S. J. (2009). *Project Management, A Managerial Approach, Seventh Edition.* Hoboken, NJ: John Wiley & Sons.

Microsoft. (2003, June). *Microsoft Solutions Framework White Paper.* Retrieved May 22, 2012, from http://www.microsoft.com/msf

Microsoft. (2005, April 27). *Introduction to the Microsoft Solutions Framework.* Retrieved May 22, 2012, from http://technet.microsoft.com/en-us/library/bb497060.aspx

Moore, S. (2010, April 5). *StrategicPPM.* Retrieved May 11, 2011, from http://strategicppm.wordpress.com/2010/04/05/the-fbis-virtual-case-file-project-and-project-failure/

Neuman, W. L. (2011). *Social Research Methods: Qualitative and Quantitative Approaches, 7th ed.* Boston, MA: Pearson Education, Inc.

Patton, M. Q. (2001). *Qualitative Evaluation Methods, 3rd Edition.* Beverly Hills, CA: SAGE Publications.

Peffers, K., Tuunanen, T., Rothenberger, M., & Chatterjee, S. (2008). A Design Science Research Methodology for IS Research. *Journal of Management Information Systems, 24*(3), 45-77.

Perry, G. (2008, 07 28). *Levi Strauss blames profit drop on SAP enterprise software system*. Retrieved July 30, 2012, from Accounting Web: www.accountingweb.com/topic/technology/levi-strauss-blames-profit-drop-sap-enterprise-software-system

PMI. (2008). *A Guide to the Project Management Body of Knowledge (PMBOK Guide), Fourth Edition.* Newtown Square, PA: Project Management Institute, Inc.

PMI. (2011, April). What is Agile? *PMI Today*, p. 8.

PMI. (2012, May). PMI Fact File. *PMI Today*, p. 4.

PMI. (2012, July). Project Perspectives. *PM Network*, p. 24.

PMI. (2012, May). What is the No. 1 reason projects fail? *PM Network*, p. 7.

Pries-Heje, J., Baskerville, R., & Venable, J. (2008). Strategies for Design Science Research Evaluation. *European Conference on Information Systems.* Galway, Ireland: AIS Electronic Library.

Richardson, G., & Butler, C. (2006). *Readings in Information Technology Project Management.* Boston, MA: Course Technology Thomson Learning.

Robertson, S., & Robertson, J. (2006). *Mastering the Requirements Process, Second Edition.* Boston, MA: Pearson Education, Inc.

Rogers Commision. (1986, Feb 3). *Report of the Presidential Commission on the Space Shuttle Challenger Accident.* Retrieved Dec 1, 2011, from NASA: http://science.ksc.nasa.gov/shuttle/missions/51-l/docs/rogers-commission/Chapter-4.txt

Saiedian, H., & Dale, R. (2000). Requirements engineering: making the connection between the software developer and customer. *Information and Software Technology*, 419-428.

Schwalbe, K. (2010a). *Information Technology Project Management, 6e.* Boston, MA: Course Technology.

Schwalbe, K. (2010b). *Revised An Introduction to Project Management.* Minneapolis, MN: Kathy Schwalbe, LLC.

Scott, L. (1996, Vol. 26 Issue 36). *Modern Healthcare.* Retrieved January 19, 2011, from http://home.modernhealthcare.com

Simon, H. A. (1996). *The Sciences of the Artificial, 3rd edition.* Cambridge, MA: The MIT Press.

Snow, A., & Keil, M. (2001). The Challenge of Accurate Software Project Status Reporting: A Two Stage Model Incorporating Status Errors and Reporting Bias. *34th Hawaii International Conference on System Sciences.* Honolulu, Hawaii.

Snow, A., & Keil, M. (2002). A Framework for Assessing the Reliability of Software Project Status Reports. *Engineering Management Journal, vol. 14*, 20-26.

Snowden, D., & Boone, M. (Nov 2007). A Leader's Framework for Decision Making. *Harvard Business Review*, 70-76.

Sommerville, I. (2011). *Software Engineering.* Boston, MA: Pearson Education, Inc.

Standish Group. (2011). *Standish Group Reports.* Retrieved October 22, 2011, from Standish Group: https://secure.standishgroup.com/reports/reports.php

Stewart, W. (2001). Balanced Scorecard for Projects. *Project Management Journal*, 38-53.

Stone, A. (2011, September 20). *D.C. Evacuation Plans: No One Was in Charge of Capital Emergency Response.* Retrieved September 9, 2012, from Huff Post DC: http://www.huffingtonpost.com/2011/09/16/dc-crisis-emergency-response-plan-management_n_966263.html

Trochim, W. (2011). *Social Research Methods*. Retrieved April 2, 2011, from http://www.socialresearch-methods.net/kb/scallik.php

Wang, J., & Hao, J. (2007). Fuzzy Linguistic PERT. *IEEE Transactions on Fuzzy Systems*, vol 15, 2.

Wat, F., & Ngai, E. (2001). Risk Analysis in Electronic Commerce Development Using Fuzzy Sets. *20th NAFIPS International Conference*, 807-811.

Weill, P., & Broadbent, M. (1998). *Leveraging the New Infrastructure.* Boston, MA: Harvard Business School Press.

Whitehouse.gov. (2010, April 7). *Open Government Initiative*. Retrieved August 02, 2012, from Around the Government, The White House: http://www.whitehouse.gov/open/around

Wi, H., Oh, S., Mun, J., & Jung, M. (2012, March). A Team Formation Model Based on Knowledge and Collaboration. *IEEE Engineering Management Review, 40*(1), 44-57.

Wieringa, R. (2010). Design Science Methodology: Principles and Practice. *ICSE'10 Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering.* Cape Town, South Africa: Association for Computing Machinery (ACM).

Woodhead, B. (2007, March 13). *$40m Qantas parts system flop*. Retrieved Nov 27, 2011, from http://www.theaustralian.com.au/business/aviation/m-qantas-parts-system-flop

Xu, Z., Khoshgoftaar, T., & Allen, E. (2002). Early Operational Risk Assessment of Software using Fuzzy Expert Systems. *Proceedings of the 5th Biannual World Automation Congress*, 435-442.

Yen, J., & Langari, R. (1999). *Fuzzy Logic: Intelligence, Control, and Information.* Upper Saddle River, NJ: Prentice Hall.

Youngkuk, Y. (2008, Nov 3). *McDonalds' "Innovate Project" Failure.* Retrieved July 6, 2011, from PMINIT: http://pminit.blogspot.com/2008/11/mcdonalds-innovate-project-failure.html

Yourdon, E. (1997). *Death March: Managing "Mission Impossible" Projects.* Upper Saddle River, NJ: Prentice Hall PTR.

Zadeh, L. (1965). Fuzzy Sets. *Information and Control*, 338-353.

Zadeh, L. (1973). Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. *IEEE Transactions on Systems, Man, and Cybernetics*, 28-44.

Zadeh, L. (1976). A fuzzy-algorithmic approach to the definition of complex or imprecise concepts. *International Journal of Man-Machine Studies*, 249-291.

Zadeh, L. (1999, January). From Computing with Numbers to Computing with Words -- from Manipulation of Measurements to Manipulation of Perceptions. *IEEE Transactions on Circuits and Systems, 45*(1), pp. 105-119.

Zadeh, L. (2009, April 14). Fuzzy Logic and Beyond - A New Look. *Acceptance Speech at Ben Franklin Award Ceremony*. Villanova University, PA, USA.

Zadeh, L. (2010). Precisiation of Meaning - Toward Computation with Natural Language. *International Conference on Information Reuse & Integration.* Berkeley, CA: IEEE.

Zier, B., & Inoue, A. (2011). Fuzzy Relational Visualization for Decision Support. *Proceedings of the Twenty-second Midwest Artificial Intelligence and Cognitive Science Conference* (pp. 8-15). Cincinnati, OH: Omnipress. Retrieved from http://ceur-ws.org/Vol-710/

Zimmermann, H.-J. (1996). *Fuzzy Set Theory and Its application, Third Edition.* Dordrecht, NL: Kluwer Academic Publishers.

Zurada, J. (1992). *Introduction to Artificial Neural Systems.* New York, NY: West Publishing Co.

Curriculum Vita

NAME: Joseph M. McQuighan

PERMANENT ADDRESS: 11155 Yellow Leaf Way, Germantown, MD 20876

PROGRAM OF STUDY: Information Technology

DEGREE AND DATE TO BE CONFERRED: Doctor of Science, 2013

Collegiate institutions attended	Dates	Degree	Date of Degree
TOWSON UNIVERSITY	August 2008	Doctor of Science	May 2013
UNIVERSITY OF SOUTHERN CALIFORNIA	August 1980	Master of Science	August 1983
UNIVERSITY OF NOTRE DAME	August 1972	Bachelor of Science	May 1976

Professional publications and conference proceedings:

J. McQuighan and R.J. Hammell II, "Applying the Design Science Research Methodology to IT Project Management", *Proceedings of the IADIS International Conference on Information Systems (IS 2013)*, pp. 191-198, 13 – 15 March, 2013, Lisbon, Portugal.

J. McQuighan and R.J. Hammell II, "Computational Linguistic Hedges Applied to a Project Critical Path", *Proceedings of the Conference on Information Systems Applied Research (CONISAR'11) v4 n1840*, pp. 1-9, 3-6 November 2011, Wilmington, NC.

J. McQuighan and R.J. Hammell II, "Scope as a Leading Indicator for Managing Software Development,", *Proceedings of the 9th ACIS International Conference on Software Engineering, Research, Management, & Applications (SERA 2011)*, pp. 235-241, 10-12 August 2011, Baltimore, MD.

J. McQuighan and R.J. Hammell II., "Computational Intelligence for Project Scope", *Proceedings of the 22nd Midwest Artificial Intelligence and Cognitive Science Conference (MAICS 2011)*, pp. 47-53, 16-17 April 2011, Cincinnati, Ohio.

Peer reviewed presentations and posters:

J. McQuighan and R.J. Hammell II, "Monitoring Project Scope Using Computational Intelligence", *Poster presented at the 10th International Research Network on Organizing by Project (IRNOP 2011)*, 19-22 June 2011, Montreal, Canada.

J. McQuighan and R.J. Hammell II, "Computational Intelligence Applied to Information Security Risk Management", *Presentation at the Short Topic Session, 14th Colloquium for Information Systems Security Education (CISSE 2010)*, 6-9 June 2010, Baltimore, MD.