Construction Workflow and Document Diagnostics

by

Antony Joseph Antony Chettupuzha

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Civil Engineering

Waterloo, Ontario, Canada, 2014

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This research aims to determine if there is sufficient information encoded within construction workflow histories and document meta-data that may be exploited for the development of knowledge constructs such as diagnosis, thereby contributing to the body of knowledge of automation in construction, with a focus on advanced construction information systems. Electronic Product and Process Management [EPPM] systems provide the capability to establish and map information flow between different parties in a construction project as well as to model project processes. The wealth of information contained in an EPPM system can be exploited to extract knowledge that can provide significant benefits to construction companies. Much of the information relating to processes and their structure, the actors (people and machines) that operate them, and the data associated with each instance of a process is encapsulated within workflows. Workflows, therefore, provide an ideal medium for the capture of knowledge over the course of a project lifecycle.

Project managers have recognized that workflows provide greater visibility and help enforce stricter compliance standards for project processes. While workflows do facilitate process compliance by ensuring constituent tasks are executed as per ordered definitions, the compliance of these individual tasks and their impact on the compliance of the workflow has not been explored. A framework has been developed to address stricter quality control by capturing knowledge of the execution times of work-items, which was then used as a basis for filtering workflows that may violate compliance norms. This significantly reduces the number of workflow instances that would need to be analyzed in detail during an audit. The framework was applied to a case study of a construction project located in British Columbia and validated.

In an ideal EPPM system, the workflow engine would operate silently and seamlessly in the background, automating structured information exchange from the start to the end of a project. In reality, most workflows used in construction projects are of a semi-automated nature requiring manual involvement for tasks ranging from selection of participants to delegation of actors. An adaptive algorithm that is able to recognize and incorporate emergent patterns from prior executed workflow instances and also determine the relative availability of resources can greatly improve the performance of a workflow implementation by reducing its semi-automated nature. An algorithm was developed to demonstrate how a self-adapting workflow methodology could be applied to construction workflows, and two specific cases based on data from a construction project were analyzed showing promising results in terms of time savings.

During a construction project, it is important to ensure that accurate and pertinent knowledge is delivered on time to appropriate personnel. Determining the criticality of documents at different stages of the project can aid companies with managing the flow of information in an organized manner, while providing for the detection of potentially disruptive, erroneous material that could result in delays and costs. An algorithm was designed based upon the meta-data and access interaction logs associated with documents in an EPPM system to identify critical documents. A scenario based on a real event and real data was developed on an EPPM system implementation and a simulation was conducted to determine the applicability of the algorithm and demonstrate its effectiveness.

It is concluded that there is sufficient information encoded within construction workflow histories and document meta-data that may be exploited for the development of knowledge constructs such as diagnosis. Diagnosis based knowledge was used to discriminate between executed behavior and planned behavior to aid compliance checking. Analysis of workflow histories resulted in the development of patterns in workflows which demonstrated time savings if implemented as self-adapting workflows.

Acknowledgements

It is with the utmost sincerity and gratitude that I would like to thank Dr Carl T. Haas for his exceptional support, guidance and mentorship during my time at the University of Waterloo. I feel truly blessed to have been supervised by Dr. Haas, and his kindness, advice and leadership, I am sure will have a lasting influence in the way I will approach life. I am also grateful for all the constructive feedback and advice from my committee members Dr. Ralph Haas, Dr. Mahesh Pandey and Dr. Keith Hipel. I would like to thank Dr. Vineet Kamat for agreeing to be the external examiner for my PhD defense.

I am indebted to Joel Gray of Coreworx Inc., for his continual support over the course of this project. Joel always made time to discuss our research despite his busy schedule and many of these discussions had a significant impact on the direction of my research. I would also like to thank Ray Simonson for all of his support of this project. I would like to express my appreciation for all the help extended by Kelly Maloney, Peter Walker, Michael Tenet, and everybody else at Coreworx Inc. I must also express my thanks to Angie Outzen for all of her help. I would also like to acknowledge NSERC-CRSNG for funding this research initiative.

I am most grateful to Anthony Schmeider for all of his help developing the simulation software program. I would like to thank Paulina for helping me with formatting sections of this thesis. I would also like to put in a note of appreciation for the effort put in by David Birnbaum in populating data for the model project simulation. I would like to acknowledge and thank my colleagues for their help over my time here at Waterloo. A warm thank you to all of my friends here – thanks for all the wonderful memories that I will cherish, of our times together here.

Finally I would like to thank my family for their constant encouragement and loving support.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

For large complex construction projects, there is a high volume of documentation and associated electronic files generated that must be transferred among all parties involved in the project in an efficient and timely manner. It is not unusual for hundreds of thousands of documents to be created and transferred over the lifecycle of mega-capital construction project (Hignett, 2014). Delays in the transfer of documents can affect procurement, operations and schedules and therefore adversely affect project costs. A central repository of the entire collection of documents generated and related transactional records can provide a valuable resource to construction project owners.

A structured document management system which imbibes process oriented workflows further extends the capability of managing the massive flow of information over a project lifecycle. At a meta level, an Electronic Product and Process Management [EPPM] system accomplishes these goals by allowing for the establishment and mapping of information flow, the implementation and modeling of project processes, while recording important participatory information pertaining to a RACI (responsible, accountable, consulted and informed) matrix from contractual and informal relationships developed between project parties. Recognizing the utility of EPPM systems, large corporations such as Fluor, Chevron and BHP Billiton have begun implementing them for large scale global projects.

Central to the functioning of any business is the execution of *tasks* or fundamental units of work. Van der Aalst defines a *work item* as the task which is executed for a specific case (van der Aalst 1998). An ordered collection of work items that facilitate the completion of a service or product is a *process*. Tasks are performed by *actors* or *resources*, which in general refers to humans or machines. In an organization, actors are differentiated by roles, giving rise to hierarchical and other organizational structures. A *workflow model* may be defined as the encapsulation of processes within an organizational context. Cerovsek and Katranuschkov (2006) note that a *workflow instance* is a specification of tasks in the process assigned to actors who work within the requirements and constraints of the organization. Van der Aalst (1998) offers a simpler definition of a workflow process, "it specifies which tasks need to be executed and in what order." A *workflow management system*, as defined by the Workflow Management Coalition, is one which, "completely defines, manages, and executes workflows through the execution of software whose order of execution is driven by a computer representation of the workflow logic."

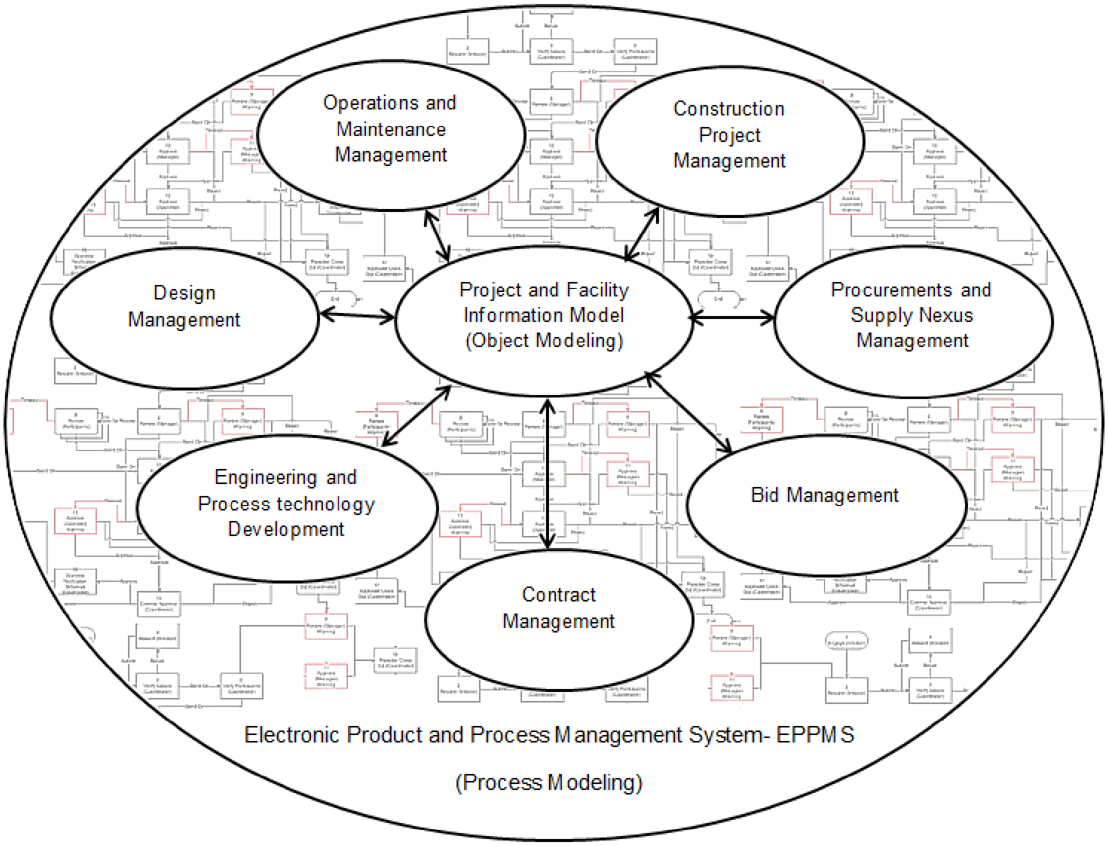


Figure 1.1: Diagram illustrating how an EPPMS links all construction spheres seamlessly

The nature of an EPPM system, as a repository for process oriented information flow, puts it in a unique position to exploit the extraction and storage of inherent knowledge encapsulated within documents and their movement over the project life cycle. An EPPM system might implement standard versions of workflows for various processes in projects, however these may be modified and customized depending on a variety of factors namely, company culture, type of project and change orders. Further, as users become acquainted with the workflow, changes may be made to reflect an improved understanding of the functioning of the work process. Implementing these changes can be expensive, both in terms of time taken to detect that a change is necessary, design the changes and then modify the workflow, as well as the costs incurred by clients for additional vendor services. However, every time a workflow is executed, data are recorded about every facet of the execution of the workflow, from the time taken to complete constituent tasks to the users who interacted with the workflow, primarily for auditing purposes. This valuable source of data may be processed into information about the behavior of workflows and their component entities. The knowledge captured from this analysis can be used to automate adaptive changes within the workflow, thereby saving significant time and costs.

A primary purpose for the adoption of workflow management systems in mega capital construction projects is to ensure work processes adhere to regulatory or company policies. Workflows ensure that constituent tasks are only executed as per a predefined order thereby lending rigorous structure to process flow. It should be noted however that an electronic system is limited to the interactions that occur within its confines, and therefore extraneous non-compliant activities cannot be captured. Nor can some parallel modes of communication such as face-to-face conversations be captured. While the structure of a workflow cannot be modified, the execution of individual constituent tasks can compromise the compliance of the process if they are not executed as specified. Developing enhanced workflow compliance detection at the task level would improve the overall compliance of the EPPM system. Knowledge would however have to be captured from the information system’s logs that can provide an analytical basis for machine interpretation of the compliance of a workflow task.

While it is possible to mine the documents stored in an EPPMS for knowledge, critical information about the overall functioning of the construction firm over the project’s lifecycle might not always be uncovered directly from them. Much of the information relating to processes and their structure, the actors (people and machines) that operate them, and the data associated with each instance of a process is encapsulated within workflows. Workflows, therefore, provide an ideal medium for the capture of knowledge over the course of a project lifecycle.

This provides the opportunity to identify key processes and personnel based on the flow of information over the duration of a project or different stages of a project. However, the actual information or project documents that are directly associated with this flow and could impact the efficiency may also be analyzed. In some instances, a workflow's design may not contain any inherent flaw, but referencing an incorrect version of a specification could cause a delay, and in the worst case result in costly repairs or redoing a section of work. This is particularly relevant during the construction phase of a project (as against the design phase when documents may be frequently updated or modified without directly affecting material or labor costs), as an incorrect reference of a revision of a document or a document with ambiguous or confusing design information directly affects an activity on the field. Identifying the characteristics of such potentially problematic documents would be useful for risk mitigation as well as improve the efficiency of the project.

To summarize, it has been illustrated that the effective functioning of an EPPMS depends heavily on the effectiveness of its workhorse, the workflow. Workflows contain information about the process control flow, the actors and the documents that are involved with a particular task. Knowledge constructs based upon this information may be used to enhance compliance checks of constituent workflow tasks, automate the adaptation of workflows and assist with risk mitigation by identifying critical documents.

1.2 Research Objectives

Improved process execution benefits project performance (O'Brien, Thomas, CII RT252). Successful process execution is dependent on the compliant execution of constituent tasks. As projects are carried out and completed across the world, there is the opportunity to capture legacy information from similar recurring process operations and use this information to incorporate automated changes to the workflow that reflect participant behavior, thereby reducing delays and costs associated with modifying workflow structure.

1.2.1 Research Proposition 1

It is this research's first proposition that construction workflows provide the opportunity to automate process diagnostic detection and evaluation which leads to efficient and improved compliance determination while also providing a basis for automated adaptation.

For every construction project, timely and accurate flow of information over the project's lifecycle is critical to the success of the venture. If misinformation, in the form of problematic documents pertaining to older versions, infiltrates the system it can disrupt the project and result in increased costs and delays. Identifying such potentially problematic documents would be useful for risk mitigation as well as improve the efficiency of the project.

1.2.2 Research Proposition 2

An EPPMS can aid with risk mitigation by identifying potentially critical documents that can affect the flow of information in a construction project.

1.2.3 Objectives

The objective of this research project is to automate the extraction of diagnostic knowledge from construction workflows in an EPPM system. Sub-objectives include:

a) Developing a methodology for efficient and enhanced compliance detection based on knowledge obtained from executed workflow instances

b) Developing a self-adaptive framework for construction workflows that builds on behavioral knowledge detection within an EPPM system

c) Determining characteristics of critically problematic documents or electronic products and developing an algorithm to detect such documents for risk mitigation

Analyses conducted based on the functioning of workflows from the above objectives can be used to extract valuable knowledge about the impact of the evolution of workflows and their effect on project delivery and performance.

1.3 Scope of Research

The overall objective of the research project, of which this thesis is a part, is to augment the capabilities of an EPPMS to automate and facilitate the improved execution of capital projects. The project comprised four distinct research thrusts: (i) Construction Supply Nexus Management (ii) Construction Project Risk Management (iii) Knowledge Management and (iv) Automating Process Management Functions with an EPPMS.

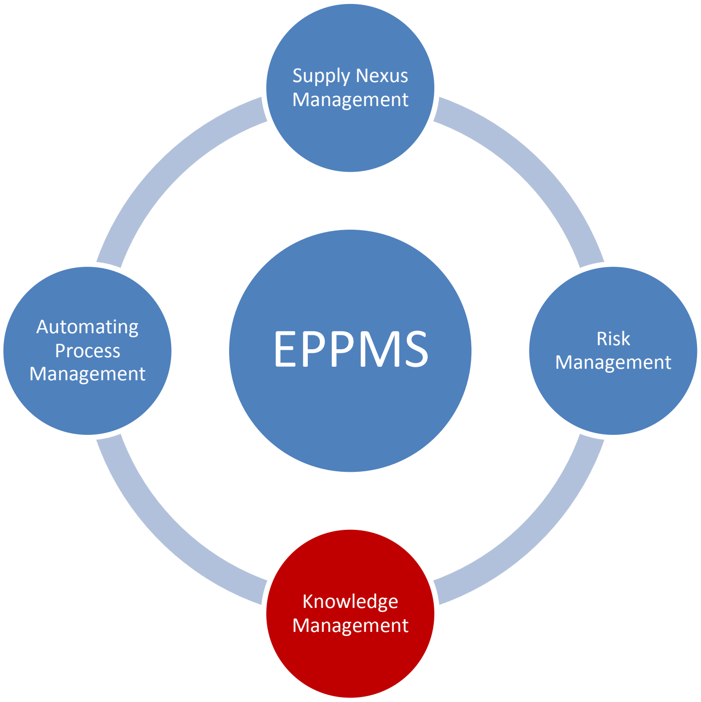


Figure 1.2: Research Thrusts for Augmenting Capabilities of an EPPM System

This thesis focused on the third thrust and hence the scope of this research was limited to developing tools to measure and extract knowledge from ongoing capital projects utilizing an electronic product and processing management system such as the Coreworx system, developed by the project's industrial partner, Coreworx Incorporated. As all of the projects currently using the Coreworx system are of a large-scale industrial nature, this research project was limited to studying the characteristic effects of workflow analysis for this sector, and in particular to data that were made available from the implementation of one such system at a client location.

1.4 Research Methodology

Figure 1.3 on the following page outlines the methodology that was employed for this research initiative.

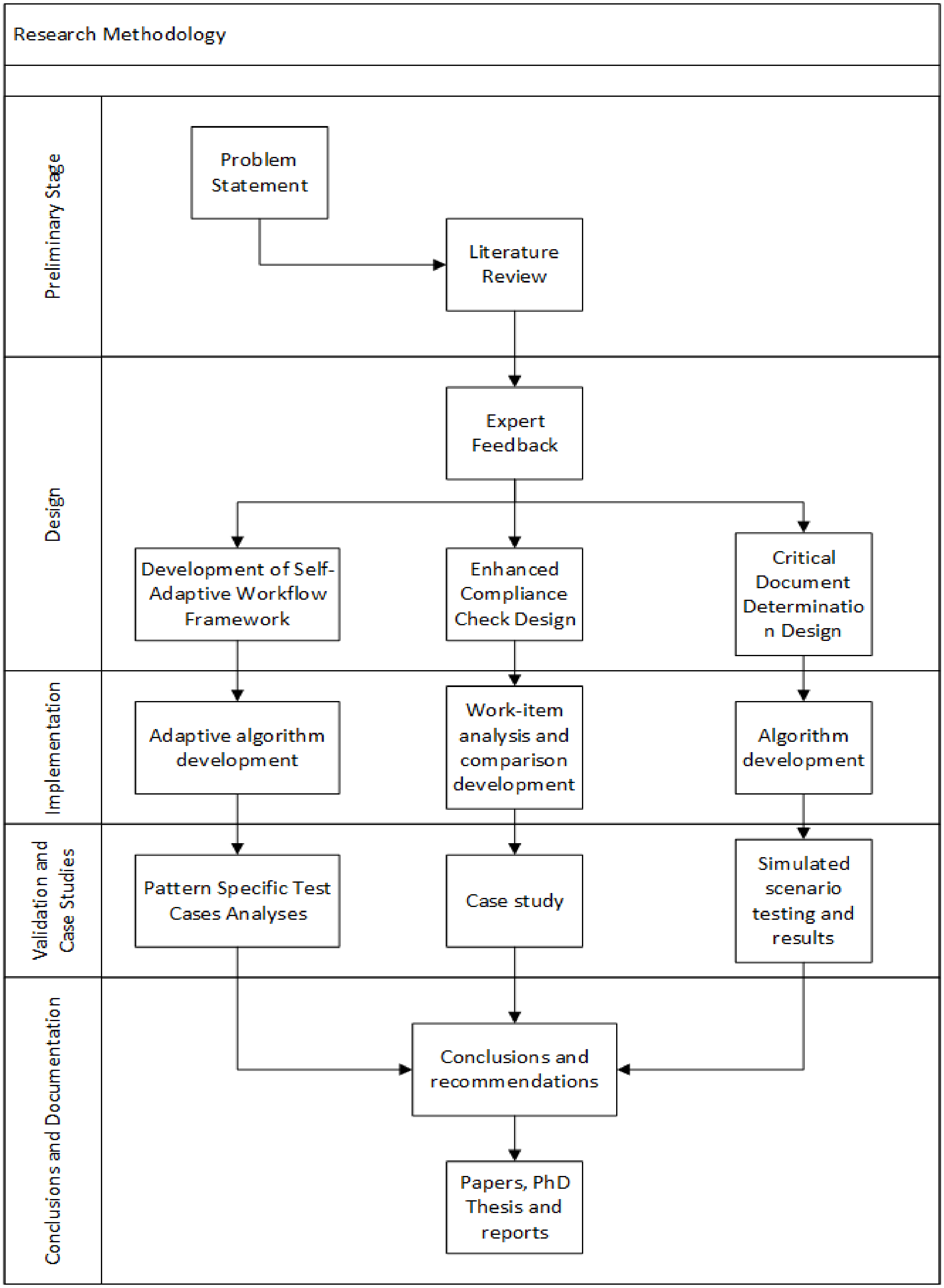

Research Methodology
Preliminary Stage
Design
Implementation
Validation and Case Studies
Conclusions and Documentation
Problem Statement
Literature Review
Expert Feedback
Development of Self-Adaptive Workflow Framework
Enhanced Compliance Check Design
Critical Document Determinatio n Design
Adaptive algorithm development
Work-item analysis and comparison development
Algorithm development
Pattern Specific Test Cases Analyses
Case study
Simulated scenario testing and results
Conclusions and recommendations
Papers, PhD Thesis and reports

Figure 1.3: Research Methodology

In order to make a meaningful contribution to both the stakeholders and the body of knowledge, this research initiative began by identifying areas wherein innovative knowledge capturing techniques could extend the capabilities of an EPPMS. Therefore, a study of the EPPMS system designed by Coreworx was carried out in conjunction with a literature review that covered knowledge management, its applicability to process oriented models and construction. It was determined that there was significant scope to extract relevant knowledge directly from workflow instance history logs, to provide for beneficial and effective measuring of compliance. Compliance may be defined as the adherence of processes to defined regulations or policies. A review on advances in the relatively new field of workflow analysis was also conducted, from which it was established that knowledge discovery from client databases often contributed to redesigning workflows. This was complimented by a thorough study of the database schema used in the Coreworx system. It was therefore proposed that automated detection of such behavioral knowledge may be used for implementing self-adapting workflows.

Interviews were then conducted with industry experts to determine why they specifically chose to adopt workflow management systems, what factors they believed affected the effectiveness of a workflow and its impact on project performance. Experts included project managers from the construction industry as well as implementation consultants and product managers of the Coreworx EPPM system in order to benefit from both the clients' and vendors' perspectives. Based on the experts' feedback, it was determined that the primary objective in implementing a workflow driven system was to ensure work processes were completed compliantly.

However the transition from paper-based compliant processes to software driven workflow models often resulted in templates that required substantial changes as users became acquainted with the system. These changes were often expensive in terms of time and additional modification costs.

Feedback on criteria that could influence the importance of documents within an EPPM system based upon meta-data was also noted.

Frameworks and algorithms were then developed for each particular sub-research focus. For the compliance checking algorithm, a case study was conducted on data made available for the first stage of a construction project in British Columbia. Two specific cases were explored from the same data set to determine the benefits that self-adapting workflows would have contributed to that project. A simulated model project was developed to test the critical documents algorithm in the absence of suitable data for analysis and validation. The results of all three systems were promising, and have been documented in this thesis and are being submitted to peer-reviewed journals for publication.

1.5 Structure of Thesis

This thesis is organized into seven chapters. Chapter One provides an introductory overview of the research including the background and motivation, objectives, scope and methodology of the research. Chapter Two consists of the literature review conducted on knowledge management, workflows and workflow mining, knowledge discovery and capture from process oriented systems with a focus on similar initiatives in the construction realm. Chapter Three describes diagnosis as a knowledge construct and defines its role in the realm of construction workflows and documents, and further outlines the challenges faced in data acquisition for this project. Chapter Four elaborates upon the framework developed to aid compliance checks in construction workflow instances. Chapter Five details the methodology employed to develop self-adapting construction workflows. Chapter Six describes how critical documents may be determined from construction documents. Finally Chapter Seven summarizes the conclusions and contributions of this research initiative.

Chapter 2

Background and Literature Review

2.1 Introduction

In diagnosing a condition, we tend to study symptoms and determine if behavior is deviant to established theory. Symptomatic study typically involves examination of a subject and drawing inferences. The examination requires the analysis of information, which Davenport and Prusak (1998) consider to be a distinguishing feature of knowledge in the data, information and knowledge hierarchy. Therefore in diagnosing construction workflows and documents, we are immediately concerned with the extraction of knowledge from the system that hosts both entities.

The extraction of knowledge from construction systems has received widespread attention in recent times. Most of these studies have tended to focus on the extraction of tacit knowledge from information systems. Several of these studies have proposed the creation of web-based portals or knowledge repositories based on extraction of knowledge from inherent process oriented models (El-Gohary et al., 2005, Woo, Jeong-Han et al., 2004, Schapke, S.e et al., 2002). The exploration of extraction of knowledge from construction workflows has however been largely unexplored, although Cerovsek and Katranuschkov proposed analyzing conceptualized workflow patterns as the basis for a real-time collaboration framework (Cerovsek and Katranuschkov, 2006). There have also been several efforts to discover knowledge from within construction databases (L. Soibelman, H. Kim, 2002) as well as from unstructured data generated during construction projects (Soibelman, L., Wu, J., Caldas, 2008).

This literature review begins with a brief introduction of knowledge management followed by broader knowledge management research and applications in civil engineering. An introduction to workflows follows, along with leading and contemporary research related to this emerging domain, particularly with a focus on knowledge discovery from workflows given its pertinence to this research initiative, and investigation of workflows within the architectural, engineering and construction sphere. Compliance is then addressed from a workflow standpoint, followed by a review of workflow evolution and adaptation. The focus then moves on towards document diagnosis and processing, and research initiatives conducted on construction documents. The next section examines different approaches that have been developed for weighting criteria, given the relevance to the methodology developed that follows in subsequent chapters of this thesis. Finally, the literature review concludes by establishing the knowledge gap that exists and contributions that may be made to the body of knowledge by undertaking this research initiative.

2.2 A Brief Background on Knowledge Management

The creation and role of knowledge has been discussed and debated throughout the ages, right from Plato and Aristotle to the current day. The business process industry took a keen interest in managing the knowledge inherent in their companies in the mid-1990s. In their book, “The Knowledge Creating Company” Nonaka and Takeuchi described how Japanese companies leveraged the management of knowledge within their firms to establish firm business practices that gave them a competitive edge. They postulated that among the two important types of knowledge in companies, explicit and tacit, the capture and transfer of tacit knowledge was the key to the success of a firm. Davenport and Prusak (1998) further emphasized the importance of using a firm’s inherent knowledge of the way that it functions in their widely acclaimed book, “Working Knowledge.” They defined knowledge as “a fluid mix of framed experiences, values, contextual information, and expert insight that provides a framework for evaluating and incorporating new experiences and information. It originates in the minds of knowers. In organizations, it often becomes embedded not only in documents or repositories but also in organizational routines, processes, practices and norms."

Several research studies have sought to classify the various types of knowledge. Alavi and Leidner (2001) identified the following types: Explicit, Implicit, Tacit, Individual, Social, Declarative, Procedural, Causal, Conditional, Relational and Pragmatic. Most researchers, including Nonaka and Takeuchi and Davenport and Prusak, however identify explicit and tacit knowledge as the key types of knowledge found in firms. Explicit knowledge is often structured and recorded in company documents, and is hence easily captured and transferred. Tacit knowledge is however more difficult to capture, Hart describes it as context specific residing in the heads of individuals (Hart, 1992). Polanyi draws an analogy to facial recognition, being able to distinguish a person's face among thousands but not being able to describe how a particular face is recognized (Polanyi, 1967). Tacit knowledge is often acquired from years of experience and transferred through direct face to face communication (Tiwana, 2000). Lam further noted a subclass of tacit knowledge; embedded knowledge found in an organization's routines (Lam, 2000) and as such is imbibed in the processes of the company.

As the era of globalization lead to businesses establishing branches of their firms in a far flung locations and increased levels of competition, firms began to realize that their competitive advantage lay in how effectively they were able to, "know what they knew" and disseminate this knowledge effectively to their employees (Davenport and Prusak, 1998). In other words, they began to realize how critical it was to manage the knowledge inherent in their organizations. Von Krogh defined knowledge management as the ability to identify and leverage collective knowledge in an organization to help it compete (von Krogh, 1998). Dalkir further expanded this definition to "KM is the deliberate and systematic coordination of an organization's people, technology, processes and organizational structure in order to add value through reuse and innovation. This coordination is achieved by creating, sharing and applying knowledge as well as through feeding the valuable lessons learnt and incorporating the best practices into corporate memory in order to foster continued organizational learning" (Dalkir, 2005).

The role of information technology [IT] in knowledge management has been debated by many. Several researchers (Davenport and Prusak, 1998, Nonaka and Takeuchi, 1995, Woo et al, 2004, Carillo et al, 2000) believe that IT systems should act solely as facilitators of knowledge storage and transfer, rather than knowledge discovery systems, due to technological limitations. They believe that the human element is far more intuitive at capturing knowledge and that the role of IT systems should provide people with the ability to communicate useful knowledge, and act as a "knowledge enabler". This has led to the socio-technical view of knowledge management, which emphasizes the interaction between the social and technical resources of a firm, i.e. the interplay between people, processes and technology, for the successful implementation of a system to leverage knowledge (Pan and Scarborough, 1999). As such Laudon and Laudon, 1998, categorize the role of IT systems into four broad groups;

- Knowledge work systems that create or capture knowledge from highly skilled workers
- Automation systems that assist with the dissemination of knowledge
- Group collaboration systems that promote knowledge sharing
- Artificial intelligence systems that attempt to capture and codify knowledge

As the focus of this research initiative is on capturing knowledge ingrained in work processes, we are interested in the techniques employed by the fourth system as defined by Laudon and Laudon. A stream of research has investigated the automated extraction of knowledge from existing information systems and sought creative ways to represent and transfer this knowledge (Davies, Duke et al, 2005, Lausen, Ding et al, 2005, Vrandecic et al, 2005, Grobelnik and Mladenic, 2005). It is widely recognized that machine-readable knowledge is best achieved through semantic process representation. This is often achieved by creating ontologies, or taxonomies and the interrelationship between their constituents (El Gohary, 2008). The Web Ontology Language (OWL) was developed for the semantic web, wherein metadata is used for facilitating machine understandable relational knowledge of webpages, and has been used extensively in several of the above listed research initiatives for knowledge management.

2.3 Knowledge Management in Civil Engineering

Construction differs from most other businesses in that it is project driven, and project environments are dynamically complex (Whelton, Pennanen and Ballard, 2005). Knowledge management for firms that operate on a project-to-project basis is typically hindered by the exclusivity of activities for a given project which may not be repeated in similar future projects and by assigning different sets of staff for projects (Brusoni et al., 1998). Prencipe and Tell (2001), note that "the pure project-based firms lack the organizational mechanisms for the knowledge acquired in one project to be transferred and used by other projects." Construction project sites are often rapidly mobilized and source different labor and material providers usually based on proximity. As such it is not unusual for a construction company to employ a completely different set of subcontractors for similar types of projects. Egbu and Botterill (2002) note that this can have detrimental effects on the flow of specialist technical knowledge for the growth of a construction organization as continuous learning may be stifled.

Kamara et al illustrated that the project-driven nature of construction made it difficult for companies to capture, transfer and reuse project specific knowledge (Kamara et al., 2002). Conroy and Soltan identify the organization, project management and project specific spheres as knowledge bases in project driven environments. They further postulate that knowledge evolves in the first two bases while it is created in the project specific base (Conroy and Soltan, 1998). De Krester and Wilkinson (2004) contend that this knowledge resides with project members and express the need for it to be captured and for the organization's benefit. In other words, most project generated knowledge is contextual, not easily captured and held by individuals and therefore fits the definition of tacit knowledge from the preceding section.

Indeed, the importance of tacit knowledge in the construction industry has been recognized by several researchers. Woo et al., (2004) assert that while AEC firms are often able to capture and store explicit knowledge, they have not been particularly successful at doing the same for tacit knowledge, or "core knowledge for highly intensive AEC activities". They further note that this has particularly been the case for larger, geographically dispersed firms and propose using a dynamic knowledge map to overcome this barrier. Several researchers recognize that the construction industry is characterized by a concentration of small firms, and have conducted studies that confirm that this fragmented nature inhibits the capture of tacit knowledge (Woo et al., 2004, Green et al., 2004, Pathirage et al., 2007, Carillo et al., 2000, Teerajetgul and Chareonngam, 2008).

While organizational learning techniques based on establishing communities of practice have been proposed (Brusoni et al., 1998) and implemented in organizations such as the construction industry institute (Liao P.C., Thomas S.R., O'Brien W.J., et al, 2012), as Carillo et al. note, the use of artificial intelligence systems to automate capture of tacit knowledge has not been as warmly received in construction (Carillo et al., 2000). The role of information technology systems has largely been utilized in storing explicit knowledge in the form of documents, and in proposing collaboration platforms to encourage the flow of tacit information (CII RS 230-1, 2007). A couple of notable examples of collaboration portals include the ones proposed by Cerovsek and Katranuschkov based on an active process model to support collaboration, while Dave and Koskela developed a social web application to foster collaboration (Dave and Koskela, 2009). The strong influence of the social aspect of knowledge management has led to attempts to represent knowledge through social analysis techniques as attempted by Chinowsky et al. (Chinowsky, Diekmann and O'Brien, 2010).

However, in recent times there has been keen interest in discovering tacit knowledge encapsulated within processes in the construction industry. Several leading research initiatives have followed on the methods used for capturing process oriented knowledge in automated systems by developing ontologies that categorize entities and define the interrelationships between them. El-Diraby et al developed a prototype formal ontology for construction knowledge (El-Diraby, Lima and Feis, 2005), which has since been adopted as a framework by several other researchers who have built new systems around it. El-Gohary and El-Diraby extended this system by developing an integration portal for collaborative construction, that at its foundation relied upon extracting formal semantics of process-oriented systems. The above authors developed ontologies for infrastructure products (IPD-Onto) (El-Diraby, 2006), construction process (IC-Pro-Onto) (El-Gohary, 2008), actors (Actor-Onto) (Zhang and El-Diraby, 2009). Zeb and Froese proposed a formal transactional ontology to bridge the gap between the aforementioned ontologies in their Trans_Dom_Onto (Zeb and Froese, 2011).

As the above review demonstrates, the field of knowledge management in construction is active with several initiatives investigating and developing a plethora of applications to implement from among a wide breadth of knowledge management techniques, measures that should advance the field while benefitting contemporary construction companies in the process. There is however, little literature on the extraction of knowledge directly from workflows, although Cerovsek and Katranuschkov (2006) do identify workflows as the "knowledge layer" of their collaborative active process model, and propose their utility as teaching tools for organizational learning. However, refinement of the workflow process itself based on the extracted knowledge has not been explored.

2.4 Workflows

The Workflow Management Coalition [WfMC] Specification, which was founded in 1993 to establish workflow standards based on processes, defines a workflow as “the computerized facilitation or automation of a business process, in whole or part” and further note that workflows are “concerned with the automation of procedures where documents, information or tasks are passed among participants according to a defined set of rules to achieve, or contribute to, an overall business goal.” As such, workflows are essentially a product of process oriented modeling of business activities.

A workflow management system (WFMS), as defined by the WfMC, is “a system that completely defines, manages and executes ‘workflows’through the execution of software whose order of execution is driven by a computer representation of the workflow logic.” An EPPM system such as the Coreworx system imbibes the core functionalities of a workflow management system in addition to various other tools for analysis and document management. For an excellent resource on definitions [briefly described in Section 1.1 of this thesis] and descriptions of workflow components, the reader is directed to W. M. P. van der Aalst’s paper titled “The Application of Petri Nets to Workflow Management” (Aalst, 1998).

While an abstract view of a workflow may imply that it is a virtual representation of the flow of work within an organization or project, within the context of a workflow management system, workflows assume far more importance as, in addition to providing the blueprint for the accomplishment of a task, they encapsulate critical information about a task’s status, such as when it was started or completed. As such they have the powerful capability to act as an auditing tool (Weijters and van der Aalst, 2004). The following section describes how this property of a workflow management system, in addition to its inherent capacity to act as an archive of process information, can be exploited to provide knowledge learning benefits and act as a valuable resource to continuous improvement of processes.

2.4.1 Knowledge Discovery in Workflows

As workflows automate processes, they contain information about who works on a task, – and therefore who likely has the experience and knowledge to complete such a task – when the task must be executed in the overall scheme of a particular project, what additional resources are required and the flow of documents and electronic products that form part of the task. All of this, as recognized by Stohr and Zhao (Stohr and Zhao, 2001), make workflow management systems “a repository of valuable process knowledge.” Zhao (Zhao, 1998) further expands on the types of knowledge which may be found in a workflow management system;

(i) Process Knowledge: that describes work items, their formal structure and rules

(ii) Institutional Knowledge: pertaining to employees and their responsibilities, organizational culture and regulation

(iii) Environmental Knowledge: that describes the external influences related to competitors, industrial affiliation, customers and regulations imposed by overseeing authorities.

Zhao (1998) contends that this knowledge exists within the workflow models, the history of executed instances of a workflow as well as external vaults such as databases and decision support systems. Both Zhao and Stohr (2001) point out that workflows can effectively disseminate knowledge as there are precise indications of what knowledge must be transferred at particular stages. Zhao also proposes the creation of a simulation module based on workflow history as a training tool for employees. Thus, workflows offer remarkable potential towards the capture and transfer of knowledge within the lifecycle of a project.

However, as Aalst et al hypothesize (van der Aalst et al., 2003), events may not always flow as smoothly as designed in the workflow. Should a worker choose to act in a manner contrary to that specified in the workflow, a deviation occurs and it becomes difficult to ascertain whether the deviation is to the benefit or detriment of the process or higher order entities such as the company or society. Aalst argues that there is therefore a need to distinguish between what has occurred against what should have occurred, and proposes mining workflow logs to arrive at this difference, describing workflow mining as the comparison between "posteriori" and "priori" models. In the context of construction information systems, there is no literature on mining workflow logs to distinguish between what was planned to occur against what eventually occurred.

As several authors including Aalst, Stohr and Zhao have noted, the capture and analysis of dynamically changing or evolving workflows can be used to augment continuous process improvement. Argyris proposed a "double-loop" system (Argyris, 1994), wherein two teams are involved; an inner team that provide feedback and suggestions based on execution of work and an outer analysis team that study the suggestions and incorporate changes as required back into the workflow, which Wargitsch et al implemented in their evolutionary workflow management system, Organizational Memory Information System (Wargitsch, Wewers and Theisinger, 1998).

While user based feedback is critical to understand the actor's perspective, it is also crucial to have independent information that can either be used to verify such feedback or provide additional information of events that have actually occurred but might not have been captured. There is active research in the field of process mining. Interested readers are directed to an excellent outline of attempted approaches provided by Aalst et al in their paper "Workflow Mining: A survey of issues and approaches." (van der Aalst et al., 2003).

In the same paper, Aalst et al expound upon the potential of workflow mining for continuous improvement from the analysis of workflow transactions. Data mining techniques may be applied to a database containing the transactions of workflow instances to assist with analysis of workflow functioning. Leymann and Roller (Leymann and Roller, 2000) also observe that data mining permits meticulous analysis of processes which could lead to the recognition factors inducing bottlenecks besides aiding continuous improvement. Thus contemporary literature is very clear on the plethora of knowledge available within workflows and there are several leading research initiatives that seek to capture this knowledge based both on manual feedback and computer aided analysis of data, albeit in domains other than construction.

It is important however to acknowledge that while workflows encapsulate much of the knowledge related to a process, they are constrained by the information and data that are directly inputted into them by actors involved in their creation and execution. There exists the likelihood of actors circumventing the workflow management system, or using other means of communication such as email or phone calls, when interacting with other actors. As such this information may not be recorded in the workflow instance. Davenport and Prusak (1998) distinguish between the hard network, through which company protocol established for official communication is used by actors, and the soft network, such as phone calls, emails or face-to-face interactions, and note that often critical knowledge may not be recorded in the latter case. This could even include changes in project or corporate strategic direction that are manifested in changing workflow behavior. The capture of knowledge from soft networks is outside the scope of this research initiative as the focus is limited to the analysis of workflows.

2.4.2 Workflows Applied to the AEC domain

Given their utility to enabling improved execution of business processes, workflows have begun to be used extensively in the construction industry by several leading companies such as FLUOR, Husky and Chevron, as a core component of EPPMS systems provided by Coreworx Inc. The use of workflows to facilitate improved construction process execution has also recently been recognized and investigated by several researchers in the academia.

Zhu and Augenbroe (2002) proposed using a "project window" to systematically reuse sub-processes from existing workflows for new processes in the AEC industry. Verheij and Augenbroe (2006) further extended this system to demonstrate how the above workflow definition and enactment can be used for collaborative planning of AEC projects. Katranuschkov et al., (2007) attempted a similar objective utilizing a slightly different approach that employed workflow patterns, proposed by van der Aalst, to re-use process patterns for collaborative work environments in the AEC sector. Tang and Akinci (2011) proposed the use of automated workflow construction to assist with bridge surveying the employs laser scanning as the primary inspection modus operandi.

2.5 Compliance

Given the immense value that workflows can impart to construction projects, interviews were conducted with project managers of mega-capital projects in order to ascertain and prioritize the advantages of workflow systems. Compliance was universally accepted and unequivocally voted as the single most important advantage of employing workflows to manage construction processes. Compliance to industry and governmental regulations, as well as specific policies that a corporation defines are factored into the initial design of a workflow template. However, it is possible that new laws or changes in company policy may occur during the deployment of a workflow implementation. As such it may be necessary to incorporate these changes to a new workflow implementation. Lui et al., (2007), developed a comprehensive system to address adherence of workflow templates to changing compliance policies. Several researchers have since developed similar work related to compliance of workflow templates.

Barrett, Cason et al (2000) have patented the development of a policy checker that contains a knowledge based system. Chung, Cheung and Machin (2008) developed a Compliance Flow system which matched user-defined processes against a standard model. Rinderle et al., (2004) proposed efficient compliance checks by observing control flow changes and data flow changes. Another rule based framework was proposed by Kumar and Liu (2008), with a focus on organizational role patterns in determining business process compliance. Lim, Kerschbaum and Wang introduced "workflow signatures" to try and protect the integrity of workflow data as a compliance aid.

Compliance seems to be viewed strictly as the adherence of the workflow to externally defined rules, either in company policy or government regulations. Most efforts seem to be directed at detecting abnormal behavior as it occurs in a dynamic environment. The compliance of individual executed workflow instances has not received as much emphasis, perhaps because it is generally accepted that deployed workflows follow a rigid structure.

Aalst (2003) has however elaborated on the need to distinguish between what has occurred against what should have occurred, and proposes mining workflow logs to arrive at this difference, describing workflow mining as the comparison between "posteriori" and "priori" models. Agarwal et al. (2006), proposed a system to understand compliance of workflow instances in the financial sector using OLAP queries on historical workflow databases. The system however cannot be applied to the construction sector given how the operations in construction projects do not follow the routine transactions in the financial sector.

2.6 Workflow Adaptation

Several researchers have explored the continuous improvement of workflows as noted in Section 2.4.1. However it is important to note that many of these research efforts were devoted towards workflow evolution, such as Wargitsch et al's memory based approach for evolutionary workflow management systems (Wargitsch et al., 1998). The goal of such initiatives was to continuously monitor, analyze, and modify workflows so as to change one implementation of a workflow over the course of a project. However for the vast majority of research initiatives, adaptation has been explored in the context of reactions to dynamic and unforeseen circumstances.

Adaptation of workflows may occur for a plethora of reasons and in variety of implementations. Van der Aalst et al., (2000) noted that workflow management systems had to be flexible to be able to accommodate changes, and went on to define the different kind of changes that workflows had to be able to consider. Han and Sheth (1998) describe several of the motivations for workflow adaptation and attempted development of a generic adaptation taxonomy to be applied to workflow management systems. Kammer et al (2000) however define the necessity of adaptation based on detecting, avoiding and handling exceptions and note the importance of flexibility and partial execution as approaches to functionalize adaptation.

It is unsurprising that a primary motivation for incorporating adaptive capabilities is to improve the robustness of a workflow in the event of an unforeseen failure (Muller, Greiner and Rahm, 2004). Typical approaches to workflow adaptation, such as the "agent work" system proposed by the above authors, tend to be rule-based driven as explored by Shuzhou and Soong (2002). However case specific adaptation has also been explored. Minor et al (2010) also attempted the automated adaptation of workflows based on workflow history data, but their focus was on using case based reasoning for when a deviation occurred to previous workflows requiring a change, rather than the project specific adaptations that were undertaken in this research thrust.

Adaptation, as considered in this research initiative, differs from workflow evolution and reactions to changing environments in that several different workflow implementations may be derived from a workflow template and adopted based upon the recognition of suitable conditions. In this context, knowledge from extracted workflow histories enables intelligent adaptation of a workflow so as to improve the efficiency of tasks without compromising compliance. Wang, Wang and Xu (2005) have proposed a monitoring agent, among other types of intelligent agents, that can introduce flexibility to a workflow template and tested their system on securities trading workflow platform. Agents are software constructs designed to operate autonomously (Bradshaw 1997) and their assimilation in workflow management systems has been proposed by Ehrler et al to assist a human administrator. While agents may be able to facilitate the adoption of pattern specific workflow implementations, there is a paucity of literature to address this in the context of improving workflow efficiency.

Several studies have also been conducted to support specific cases of adaptation as defined in this research thrust. Delegations, a case study that was conducted in this research project, received considerable interest a few years ago, most notably when Wainer, Kumar and Barthelmess (2007) developed a formal model permitting delegations and revocation. Atluri and Warner (2005) attempted to assign conditional delegations while adhering to typical workflow constraints such as appropriate authorization. Crampton and Khambhammettu (2008) developed algorithms to ascertain the satisfiability (which they define as the completion of a workflow by a set of users) based on various delegation requests. However all the above research initiatives propose methodology for permitting delegations within constraints rather than automating delegation based on workflow history.

To conclude, adaptation of workflows in literature has been confined to reacting to unforeseen circumstances either by following rule-based or case-based reasoning while incorporating flexibility, or by using agents in a semi-automated process in conjunction with a workflow administrator. Self-adapting workflows that recognize object parameters based on inheritance of various workflow patterns has not been explored. Within the construction sector in particular, the adoption of such algorithms would be of immense use, given the project specific nature of this field.

2.7 Document Analysis

In analyzing information systems, there are several different sources that may be mined depending on the type of knowledge that is being sought. Often in web-based systems, there are three main sources; (i) the content of a webpage, (ii) the structure of a web-page, and (iii) the usage and system logs pertaining to that webpage (Dunham, 2003). Suitable mining methods are adopted based upon the source of data. Mining knowledge from content often employs techniques such as text data mining and natural language processing so as to be able to associate keywords with intelligent contextual understanding of what the document describes.

Semantic analysis of documents is, and has been, an active research area for many years now. Semantic analysis, which is related to understanding the content of a document, typically relies on the creation of ontologies and natural language processing (Maedche and Staab, 2001). Such approaches tend to match similar documents and classify documents using relevance rankings (White, Ruthven and Jose, 2002). Indeed, there are at least a couple of patents (Bharat, Henzinger, 2000), (Kirsch, 1997) etc., that have been registered which tend to look for relevant documents (Pirolliet al., 2001).

Most servers protect information related to files that are not directly accessed by the public for security reasons. Therefore study of logs on the internet, which is where the largest mining of documents takes place, is constrained. However many enterprising researchers have proposed interesting ways of still finding out web access patterns, like Zaiane, Xin and Han (1998) and Nasraaoui, Krishnapuram and Joshi (1999).

The article at the forefront of mining document structure, or the links and metadata of a document, is undoubtedly the article that "brought order to the web" (Page, Brin, Motwani and Winograd, 1999) and introduced PageRank, which went on to become the largest search engine in the world, Google. Several other seminal papers on link structure followed, including Haveliwala's topic sensitive pagerank (2003).

2.7.1 Document Processing in the AEC Sector

In the AEC sector, the common approach to analyzing documents has built upon these techniques. El Gohary (2008) developed ontologies to enable semantic interpretation of infrastructure documents. Caldas and Soibelman (2003) automated the classification of construction documents hierarchically. Research in this area has usually been confined to understanding the content of documents, which pertains to the first source of data mentioned above.

However research about the importance of a document and the evolutionary characteristics that determine its criticality as the project progresses has not been explored in the context of mining. The content of a document in this regard ceases to be as important as the structure, or the relations and links between documents in the overall context of an information system in a construction project, and the usage characteristics associated with the document. This research initiative seeks to fill this knowledge gap by employing data mining techniques to structural and usage sources as described in the preceding paragraph.

2.8 Approaches to Determining Weighting Criteria

The research initiatives described in this dissertation often consist of a set of factors or diagnostic determining criteria. To obtain an objective function that would take into consideration each factor and its influence on the overall diagnosis, it becomes important to ascertain suitable weighting criteria. This section describes the various weighting methodologies that were studied so as to ascertain which method would be most applicable to this research initiative.

Wang and Stanley (1970) define the broad weighting methods they observed from education research as natural weights, priori or subjective weights, multiple regression, equal contributions to total variance, equal correlations with the composite, minimum generalized variance, minimum variation, weighting by length, weighting by validity, and weighting by difficulty. Theoretical methods, they contend, are often used when there are no composite factors that rely on human intuition or experience, and often rely on statistical measures such as the mean or variance of data.

Several methods have also been developed that take advantage of both subjective ranking as well as theoretical interpretation. Examples of these methods include the Rank Order Centroid, which Barron and Barrett (1996) contend are the most accurate of the rank based weighting methods, the ratio method and pairwise comparison. The Delphi method has seen widespread adoption after being developed by the U.S Air Force (Robinson, 1991). The method is an iterative processes wherein opinions are sought from experts and refined using statistical procedures until expert weights converge.

In recent times, artificial intelligence has influenced weighting methodology. Fuzzy pairwise comparison was proposed by Deng for multi-criteria analysis (1999). Jensen expounded upon the utility of using Bayesian networks to determine weights (1996). Neural networks have been used extensively to determine weights in virtually every discipline in science, notably in construction by Ghaboussi, Garrett and Wu (1991) and Kim et al., (2004).

Multi-attribute utility theory has received widespread attention and Dyer et al., (1992) believe, it will continue to do so given its application to management science. Genetic algorithms have also been utilized, sometimes in conjunction with neural networks (Montana and Davis, 1989) but often on their own (Fonseca and Fleming, 1993). Principal component analysis (Jolliffe, 2005) has found applicability to weighting by virtue of reducing factors and analyzing relationships based on variance maximization.

To summarize, literature is rich with techniques and approaches to determine appropriate weighting, however the selection of the method depends largely on the nature of the problem being investigated. Judicious selection of an appropriate technique is often influenced by expert feedback or experience and familiarity with commonly used methods in a domain.

2.9 Knowledge Gap

Even though the study of workflow management systems is relatively new, already several researchers have established different methodologies for the design and analysis of workflows. The value of workflow history logs has received considerable attention, and in several cases these data sources have been used for facilitating feeder information for continuous improvement programs. Several researchers have also noted the value that can be gained from mining workflow history logs for the effective dissemination of knowledge and development of social knowledge constructs. However, particularly in the construction sector, using this knowledge to facilitate smarter evolving workflows has not been explored.

Compliance of workflows has received a lot of attention, specifically in the banking sector after the financial crash of 2008. As such methodologies that were developed to ensure comparative analysis of workflow compliance was dynamic for changing governmental regulations. These efforts measure compliance at a workflow template level, and not at the work-item level. This is particularly important for construction, wherein there is no published literature for establishing the compliance of a workflow, wherein the compliance of the entire workflow is dependent on the compliance of individual components that make up the workflow as well as the overall workflow structure.

The analysis of documents in the construction sector has received a lot of attention in recent times, however this has usually been confined to intelligent categorization of documents to assist with organization, often for legacy data systems. Modern construction information systems however consist of many meta-data elements and access statistics, which present alternative methods for organizing and analyzing documents. Further, there has been no attempt to determine critical documents within construction information systems.

All of the above paragraphs demonstrate that there is sufficient scope to develop new methodology which can be applied to each of these domains. Such methodology can lay the foundation for further exploration and contributions to the body of knowledge in construction knowledge management.

Chapter 3

Data Sources and Diagnostic Development

Crucial to the development of diagnostics is the availability of suitable data upon which models may be developed and wherein hypotheses may be tested. This chapter elucidates upon the challenges faced in data acquisition, the need to generate some data and the methodology employed to ensure such data conformed as closely as possible to real life data when it was simulated. Finally, this chapter concludes with an introduction to diagnostic development for construction workflows and documents and outlines the structure of research in subsequent chapters of this thesis.

3.1 Data Requirements for Diagnostics

Dictionaries define diagnosis as “an investigation or analysis of the cause or nature of a condition, situation, or problem.” In the field of artificial intelligence, diagnosis refers to the development of algorithms that are able to verify that executed system behavior is in line with planned system behavior, and if it is not in line, discerning the reasons. Diagnosis therefore, is usually the result of analyzing information and forming an appropriate conclusion, and as such is a knowledge construct. The knowledge pyramid hierarchy - the original composition of which is commonly attributed to Ackoff (1989) – is depicted in Figure 3.1 to demonstrate how diagnosis depends on information which in turn depends on data.

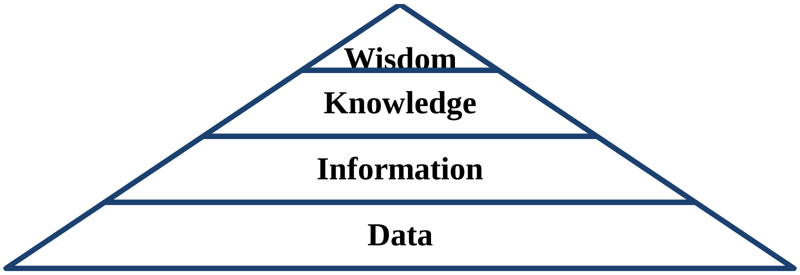


Figure 3.1Ackoff's Knowledge Pyramid

Davenport and Prusak (1998) have elaborated on the relationship between the different layers. Knowledge, they contend, is created when information is compared, when consequences are inferred from it and when connections are created as a result of analyzing information. Information on the other hand, is formed by categorizing data, performing corrections and calculations upon it, forming context and condensing pertinent data. Data, on the lowest tier, refers to observations or facts. Wisdom has often been described as the ability to make good judgments based on knowledge. In the case of construction information systems, an automatically adapting workflow can be described as a wisdom construct, provided knowledge exists for determining when a workflow may adapt.

In the interests of clarity, the following terminology that shall be used frequently through this dissertation has been defined:

(a) A work-item refers to a task or a fundamental unit of work.

(b) A process refers to an ordered collection of work items that facilitate the completion of a service.

(c) A workflow model or workflow template may be defined as the encapsulation of processes for a type of work. For example, workflow templates may be defined for RFIs, Change Management, and Approvals.

(d) A workflow implementation refers to a workflow template derived from a workflow model but customized for a specific activity. For example, if the workflow template is modified to tailor specific policies for a company. Again, if after being operational the new template is modified perhaps for optimization, this would result in the creation of a new implementation.

(e) Workflow instance refers to an executed workflow implementation for an activity. In other words, a specific case that has been completed.

With respect to the first two hypotheses in this thesis, to diagnose whether a workflow instance is compliant or whether a workflow may adapt to improve efficiency, it becomes necessary to collate observations on construction workflows. These observations are usually available in the form of workflow histories or logs, and several authors have already stated the importance of such logs for knowledge discovery as noted in the literature review. Criticality of a document is to a great extent a matter of human judgment. Some measures to aid in this judgment may be automated however. Automating measurement of the criticality of a document however (which is related to the third objective of this thesis), relies on information stored in that document's meta-data, its characteristics within an executed workflow as well as the access and usage statistics of the document. Storing and retrieving all of this data represents a major challenge for conventional construction information systems, however performing such operations on large volumes of data would be integral to the functioning of an ideal EPPM system.

3.2 Data Availability and Third Party Restrictions

Having a reasonable sample size of observations upon which to establish knowledge is therefore imperative if a system has to be designed for real-time diagnosis. In an ideal world, these observations would be drawn from a multitude of information systems that represent the plethora of construction project data that has ever been recorded. In the real world, the privacy of such data is heavily guarded and the transformation of all such data to a uniformly accessible format would be impractical for several reasons, not least of all because of time considerations. It would not be impractical however to form a diagnosis based on a subset of available data that represents a class of construction projects.

The sponsoring partner for this research initiative, Coreworx Incorporated, is a software vendor catering to Engineering, Procurement and Construction (EPC) companies that undertake projects with a capital of at least $200 million. Companies usually maintain their own servers and database systems for all data generated over the course of the project lifecycle. Given the value of such information to the company's functioning – and therefore also to competitors – and continuous improvement, the confidentiality of data is of the utmost importance and is usually covered in non-disclosure agreements prior to the installation of an EPPM system.

Several companies expressed reluctance to the sponsoring partner regarding sharing of their data for the purposes of this research initiative. This data pertained to the first, and to a smaller extent the beginning of the second stage, of a series of repetitive construction projects located in British Columbia. Table 3.1 below, depicts the types of workflows employed at the project site as well as the number of executed workflow instances recorded for the first project stage.

Workflow Type	Instances
Request for Information	1070
Change Request	615
Trend Impacts	1036
Trend Processes	924
Non-Conformance Reports	174
Approvals	78

Table 3.1: Data sources: Workflow type and number of instances

The Trend Impact and Trend Processes workflows were created while the project was operational and were introduced as replacements for the Change Request workflow. Therefore, the bulk of workflows associated with this project were related to changes that occurred. It is not uncommon however for a Change Request workflow to have been triggered by a Request for Information (RFI) workflow. Change Request and Trend workflows were analyzed by one of my colleagues in this project. The number of instances of Approval and Non-Conformance Reports are considerably low and so were excluded from this study. While all the above reasons would have been sufficient for selecting RFI workflows for this research initiative, there are several other important reasons why RFIs were chosen amongst the other workflow instances.

3.3 A Brief Description of Request for Information Workflows

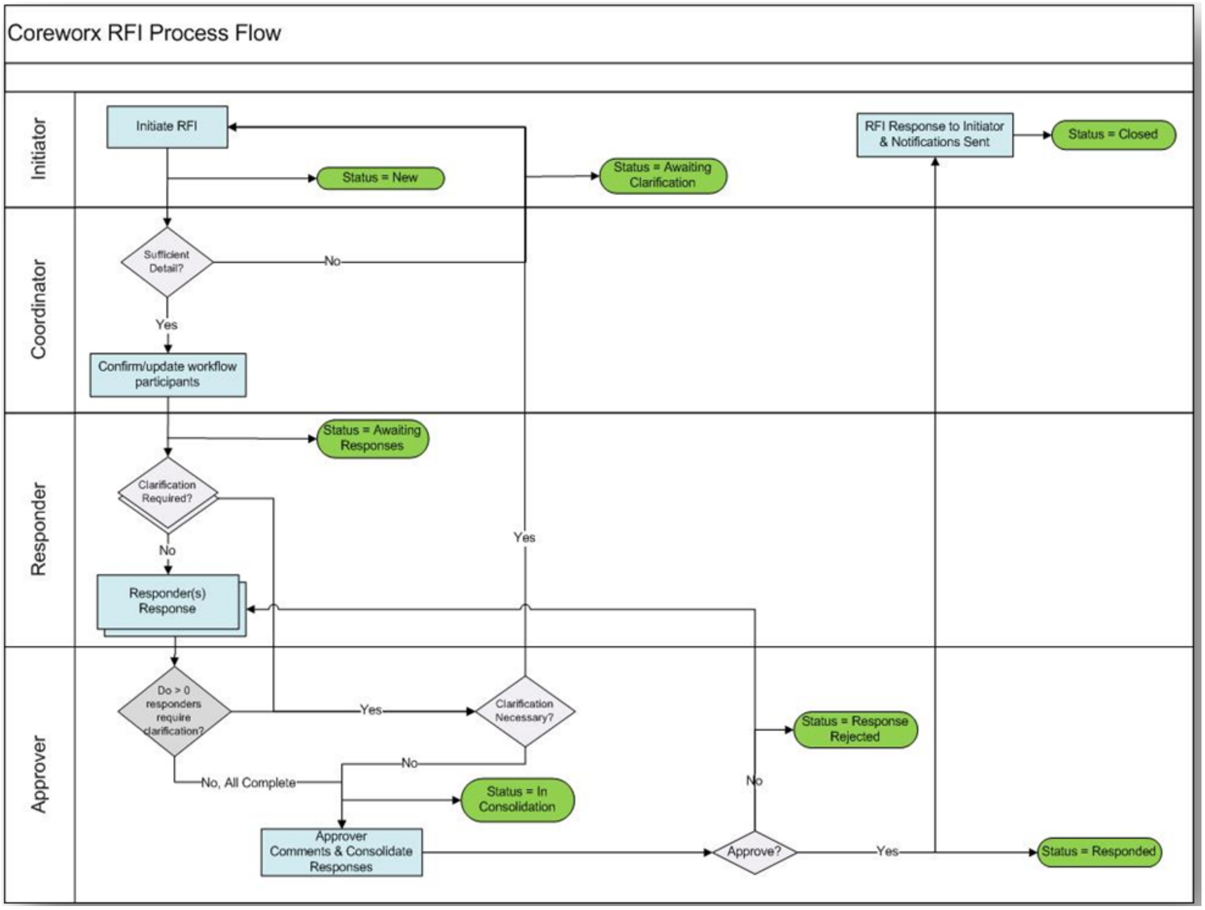

Coreworx RFI Process Flow
Initiator
Coordinator
Responder
Approver
Initiate RFI
Status = New
Status = Awaiting Clarification
RFI Response to Initiator & Notifications Sent
Status = Closed
Sufficient Detail?
No
Yes
Confirm/update workflow participants
Status = Awaiting Responses
Clarification Required?
No
Yes
Responder(s) Response
Do > 0 responders require clarification?
Yes
Clarification Necessary?
No
No, All Complete
Status = In Consolidation
Status = Response Rejected
No
Approver Comments & Consolidate Responses
Approve?
Yes
Status = Responded

Figure 3.2 The Coreworx RFI Process Flow © Coreworx Inc.

An RFI is issued when further information is required from a project participant. The RFI workflow implemented at the project site by Coreworx Inc has been shown in Figure 3.2. Once an RFI has been issued, the details that have been requested have to be verified by a project coordinator. The project coordinator must then select an appropriate list of participants who may respond to the request. These participants who respond to the request may then choose to ask for further clarification of the details of the request in question. Different RFIs have different requirements for how many responses are required before the RFI may be considered complete. Often the number of responses depends on the importance of the workflow and time constraints. RFI Workflows are therefore prioritized as High Priority, Medium Priority and Low Priority at the discretion of the initiator. This level of importance accorded to the workflow may be adjusted by the coordinator if required.

Once the appropriate number of responses has been satisfied, an approver reviews the comments and consolidates the replies if necessary. The approver may also request clarification to ensure that the responses satisfy the original query. Once the approver is satisfied with the response, he may send the response back to the initiator. At this stage the workflow is considered to have been completed. It should be noted that at every stage a permissible time limit is set for participants to respond. For this project, the time limit for every stage was set at 3 days. A notification is sent to a participant if they have not completed their task in the workflow close to the end of the 3 day time limit. The time limit varies depending on the priority of the workflow.

By their very nature, RFIs require a response to a query and the time needed for this response can vary considerably. The time taken to respond to an RFI depends on a number of factors such as concurrent activities the participant may be involved in, determining a suitable response based on sourcing information, the department or type of party that has initiated or is responding to the request etc, many of which are external to the EPPM system.

In the simplest cases, the project coordinator will be available to provide a suitable response to the initiator thereby satisfactorily completing the RFI within a few stages. For more complicated queries, a specific personnel group may have to be identified, this group may have to be modified by a coordinator depending on availability or expertise in the area and the RFI might require a consolidated response from more than one responder. Some of these responses may then require clarification prior to being consolidated by a manager before being sent back to the initiator. As such, there is no simple way of deducing an accurate average time of completion for an RFI.

While individual tasks, hereafter referred to as work-items, are often restricted by time constraints, such as the three day limit described above, these often constitute a maximum allowable time for that task to have not been completed. Adding all such times in series on the critical path for the process will provide the maximum allowable time for the RFI to have been completed except when there are iterations allowed. The true time will be reflected by adding the execution times for each work-item. The true times for two exactly similar RFIs can again vary significantly depending on the availability of personnel who are involved in that workflow instance. It is therefore quite a challenge to estimate what a reasonable timeframe would be for the average execution of an RFI instance, but it is possible to determine reasonable constraints within which most RFIs are completed.

3.4 Data for Process Oriented Diagnosis

One of the primary reasons companies employ workflow-oriented information systems is to ensure that work processes adopt a suitable level of compliance. Whilst providing structure so as to ensure work items were executed as planned, workflow information systems are also highly beneficial for auditing purposes. Records relating to whom all were involved in a task, when they began and finished working on the task, the status of the document after the interaction etc. must all be stored in a database.

These records are stored as logs representing workflow history data. Table 3.2 on the next page, is the result of a database query wherein data has been assembled showing the document ID, workflow ID, the priority of the said workflow, the work item, user who worked on the work item and the start and end times for that work item.

RFI ID	WF ID	Priority	Work Item	Start Time	End Time	User
RFI-00045-1003	38	High	Verify Details	7:51:24 PM	10:36:48 PM	John Smith
RFI-00045-1003	38	High	Verify Participants	10:36:51 PM	11:40:30 PM	John Smith
RFI-00045-1003	38	High	Respond (Responsible Engineer)	11:40:30 PM	3:45:49 PM	Sally Jane
RFI-00045-1003	38	High	Approve (Approver)	3:45:52 PM	8:36:31 PM	Sally Jane
RFI-00045-1003	38	High	Respond (Responsible Engineer)	8:36:34 PM	8:37:30 PM	Sally Jane
RFI-00045-1003	38	High	Verify Responders	8:37:30 PM	8:37:53 PM	Sally Jane
RFI-00045-1003	38	High	Respond (Responders)	8:37:53 PM	6:27:36 PM	James Brown
RFI-00045-1003	38	High	Consolidate (Engineer)	6:27:51 PM	6:56:41 PM	Sally Jane
RFI-00045-1003	38	High	Approve (Approver)	6:56:41 PM	9:24:45 PM	Matthew Paul
RFI-00045-1003	38	High	Approved Close Out	9:24:50 PM	9:22:01 PM	William Wallace

Table 3.2 RFI workflow instance data

It is not uncommon for a construction project to consist of several thousand executed workflow instances. Such massive volumes of information provide enormous scope for determining patterns among similar classes of workflow implementations which help define behavior for that class of process implementations. For instance, by analyzing thousands of workflows which terminated successfully, it may be possible to determine a time distribution that fits the execution times for all such workflows in that class, where their behavior is stationary (from a signal perspective) over a period of time.

If a new workflow instance executes in a much shorter period of time than expected based on the generated time-signature, a notification may be generated to warn of a possible compliance violation. Similarly, if a user delegates a series of tasks to another user, perhaps because he or she is too busy at that instant, then perhaps the next series of activities for the user may automatically be delegated for a few days. Therefore the diagnosis of process oriented data can be used to account for both reactive and predictive behavioral constructs.

3.5 Generation of Data for Determining Critical Documents

In an ideal EPPM system, data will be recorded for every interaction a user has with the system. Whenever a user accesses a file attached to a document such as an RFI, a record will be created in the database detailing when the user accessed the attached file and for how long. In some conventional EPPM systems however, this level detail is not recorded. Instead common file system records such as the last time the file was modified rather than when it was viewed, and who made the modification are stored. Perhaps this can be explained by the fact that information related to who may have changed a file rather than how often the file was viewed by other participants may be important for auditing purposes.

However, the meta-data related to how often an attached file was accessed, the number of different participants who accessed it and when they accessed it during the project are useful indicators of the relative importance of that file. The data that was available for this project did not have these access logs, and in addition there were insufficient relationships maintained about files that were attached to the documents. To demonstrate the utility of maintaining this meta-data for aiding in the identification of critical documents, a model project based project based on real data was developed within the current system. The developed system, which has been described in detail in chapter 6, was supplemented by appropriate access related logs based on user interaction on other information systems.

3.6 Diagnostic Development and Research Structure

The underlying objectives of this research initiative are to develop diagnostics pertinent to construction workflows and documents and as such to determine behavior which is beneficial to the industry and contributes to the body of knowledge. The previous sections in this chapter have elucidated upon how crucial data is and where it was sourced for this project for the development of diagnostics. It is just as important to identify conditions or behavior that must be studied. Knowledge discovery may be categorized as supervised or unsupervised depending on the algorithm employed. Unsupervised algorithms, such as clustering, typically search for new patterns, and should the new patterns establish a correlation of practical utility, an application based tool may be developed. Supervised learning differs due to the availability of class labels that aid in training a system to differentiate between behaviors based on outcomes. Such system learning is based upon a pre-determined target or objective.

Expert feedback is frequently employed in knowledge discovery to provide direction towards identifying objectives and model parameters. Project managers who have employed construction workflows to manage their projects are in an ideal position to describe why they chose a workflow implementation, what parameters they would like to measure better and how the workflow system can improve. Interviews were conducted with client project managers in Calgary from October 9th to 11th 2012, to help establish suitable characteristics of workflows that required further understanding. Interviews were also conducted with vendor consultants, so as to gain perspectives from both sides and arrive at mutually beneficial improvements for construction workflows.

The interviews yielded the following unanimous conclusion: the primary reason for employing a workflow management system for a construction project is to ensure processes adhere to acceptable compliance levels. As such, compliance has emerged as the single, most critical feature of construction workflows. However, while a workflow might ensure that the sequences of tasks abides by a predefined compliant enforcing structure, measuring the compliance of each of the individual work items has not received as much attention. The underlying assumption that a workflow was compliant might not hold weight if a few of the component work items did not execute as planned. Individual workflow instance oversight would be impractical considering the large number of instances that occur over a construction project. Chapter 4 provides a framework wherein it is possible to isolate potentially problematic workflows from a compliant point of view while reducing the number of workflows that may require additional scrutiny.

Client project managers further noted that often the rigid structure of a workflow, which is essential for ensuring compliance, was also a hindrance particularly when repetitive and apparently mundane tasks required manual confirmation. Therefore it would be useful to develop workflows that can adapt automatically based on predictive indicators from previously established behavior while maintaining expected levels of compliance. Chapter 5 outlines a general algorithm to accommodate such adaptation and further elaborates upon two specific cases for the implementation of algorithms to facilitate adaptation.

Finally Chapter 6 describes the development of a simulated model project that was created to demonstrate the utility of the algorithm developed to determine critical documents over the project’s lifecycle.

Chapter 4

Compliance in Construction Workflow Instances

Construction workflows differ from workflows employed in a number of other sectors such as finance and manufacturing. Workflows employed in the latter sectors are typically of a transactional nature – for example the authorization of a credit card transaction – and in many cases for manufacturing, workflows are fully automated with little or no tasks scheduled for humans except in the event of a breakdown. Due to the unique and unpredictable nature of construction projects, workflows employed in this sector tend to be facilitators of structured communication and analysis. Typical examples of workflows used in an construction EPPM system project are Request for Information (RFI), Change Management (CM) and Interface Management (IM) workflows.

The primary purpose of employing workflows to manage project processes is to ensure that the same structure and quality standards are adhered to in the execution of classes of processes. Workflows therefore automate the functioning of tasks strictly with regard to established rules, which may be defined by regulatory bodies or by policies instituted by the company. Given the immense value that workflows can impart to construction projects, interviews were conducted with project managers of mega-capital projects in order to ascertain and prioritize the advantages of workflow systems. Compliance was universally accepted and unequivocally voted as the single most important factor in employing workflows to manage construction processes.

Project managers noted that since compliance to industry and governmental regulations, as well as specific policies that a corporation defines were factored into the initial design of a workflow template, they had faith that the execution of processes was compliant with established policy. It is possible however, that new laws or changes in company policy may occur during the deployment of a workflow implementation. As such it may be necessary to incorporate these changes to a new workflow implementation. Lui et al., (2007), developed a comprehensive system to address adherence of workflow templates to changing compliance policies. Several researchers have since developed similar work related to compliance of workflow templates.

The focus has been on such implementations at the workflow template level, under the premise that all workflow instances inherit these compliance policies and as such they are enforced rigorously. The compliance of individual workflow instances is often either not monitored or has to be checked manually by a coordinator assigned to administer the workflow. In most cases, an individual instance may only be analyzed well after the tasks in that process instance have been completed as part of an audit.

The compliance of individual executed workflow instances has not received as much emphasis, perhaps because it is generally accepted that deployed workflows follow a rigid structure. Aalst (2003) has however elaborated on the need to distinguish between what has occurred against what should have occurred. He further proposes mining workflow logs to arrive at this difference, describing workflow mining as the comparison between "*posteriori*" and "*priori*" models. Agarwal et al. (2006), proposed a system to understand compliance of workflow instances in the financial sector using OLAP queries on historical workflow databases. The system however cannot be applied to the construction sector given how the operations in construction projects do not follow the routine transactions in the financial sector.

If a workflow instance's compliance is suspected of being overridden then the assumption is that a task or series of tasks have been subverted outside of the system. A process may be considered to be subverted if the tasks are not executed as expected; for example an actor approves a task without actually checking the task due to time constraints, or an actor does not respond to a task and the workflow proceeds after a time-out, or if unauthorized (for example, insider knowledge leak or in a conflict of interest scenario) communication occurs outside the scope of the workflow management system and is therefore not recorded by the system. At present there are a lack of tools for recognizing and addressing the compliance of individual workflow instances that may have been undermined by external subversion.

Further, on EPPM systems, executing instances aren't monitored automatically in real time. In the event that a task is not completed before the initially determined time, an alert is sent to a project coordinator. Often the project manager can only monitor the progress of an executing workflow on "a dashboard" that provides an overall view of all workflows in progress. In addition, there are no tools to analyze workflows that have been completed to see if they were compliant. Perhaps most importantly, the criterion for which a data analyst can determine if an executed workflow was compliant has not been expounded. Therefore a framework which clearly addresses the compliance of executed and executing workflows would be of great value both as an auditing tool as well as a monitoring tool.

4.1 Framework for Compliance Checks

Since compliance has been recognized as the key factor construction firms consider while employing workflows for mega capital construction projects, it is important to differentiate between the two stages of enforcing compliance. We may divide compliance checks into two broad categories: (i) Compliance of the workflow template to ensure that established governmental regulations are followed and (ii) Compliance of the individual workflow instances to the created workflow template. As literature is rich in addressing compliance at the design phase, i.e. the development of the workflow template and modifications to workflow implementations, the focus of this research initiative will be on the compliance of individually executed workflow instances in a construction project.

In the case of workflow instances that have already been completed, there is an obvious resource that could provide an indication of whether the instance was performed compliantly. Workflow histories possess a wealth of knowledge about project processes. By analyzing the logs of individual tasks of the workflow, it is possible to determine with sufficient accuracy the actual intervals of time that are spent on performing the task as against the time allotted for the task during the design phase of the workflow template. Tasks may have been completed in less than the allotted time, at exactly the allotted time and in some cases they may have required additional time.

While the time that is taken to perform a task is key to completing a process in accordance to schedule and therefore potentially reducing costs that may be incurred from delays, several project managers specified that they did not consider finishing a task quicker to be an indicator of efficiency. In fact one of the reasons for employing a rigorous workflow is to ensure that due process has been followed while completing a task. Naturally taking too long to complete a task is not ideal, but in some cases these circumstances are unavoidable due to external (i.e., of the EPPM system) factors such as an employee taking sick leave, etc.

As mentioned in Section 3.3, the allotted time is often fixed at a constant during the creation of the workflow template, and as such does not accurately reflect the time that the task requires but rather the permissible time for the task to be completed so that the workflow instance is completed in a timely manner. By their very nature, RFI’s tend to have varying execution times. A request may be answered immediately or may require considerable research or confirmation from concerned parties and as such may take much longer. To complicate matters further, it is not entirely possible to estimate just how many RFIs may be generated over the course of project nor is it accurate to establish that an average time taken to perform tasks related to an RFI in one project may be the same for another project or even for subsidiary projects within the umbrella of a large mega-capital project.

As mentioned in Chapter 3, several firms were reluctant to share data related to construction projects and as such it is difficult to conduct a comparative analysis of a large enough sample set to establish fundamental volumes and execution times for RFI workflow instances. Another factor to consider is that currently there is no industry standard process for RFIs, and different firms employ different workflow implementations with added or reduced tasks, and so the overall execution times can vary further still.

It is possible however, within the environs of a single construction project, to determine with reasonable confidence how long various tasks within an RFI workflow should take, given there exists a reasonably sized sample set for analysis. Indeed, constraints such as the use of a uniform workflow implementation, a limited workforce from varying construction parties and the establishment of specific communication protocols (i.e., firms either use interface management and/or change requests in addition to RFIs rather than mixing these protocols along with RFIs) make such an approach attractive.

It is quite likely that an analysis of the individual work-items for several executed workflow instances will yield distributions upon which the time required for these tasks can be modelled. These time signatures are crucial indicators of the average time that actors utilized when working on the work-items and as such demonstrate the time that was required for a task in the past to have been performed with adherence to the compliance policies inherent in that workflow. Further if such time signatures have been determined from a training sample of data, all future executing workflows may be monitored in real-time and for unusual time stamps an alert may be sent to a coordinator. This will greatly reduce the total number of workflows that are analyzed in an audit and can help with risk mitigation for an event as it is unfolding. The time signatures are a critical component of the framework developed, which are described in the subsequent sections of this chapter.

4.2 Methodology

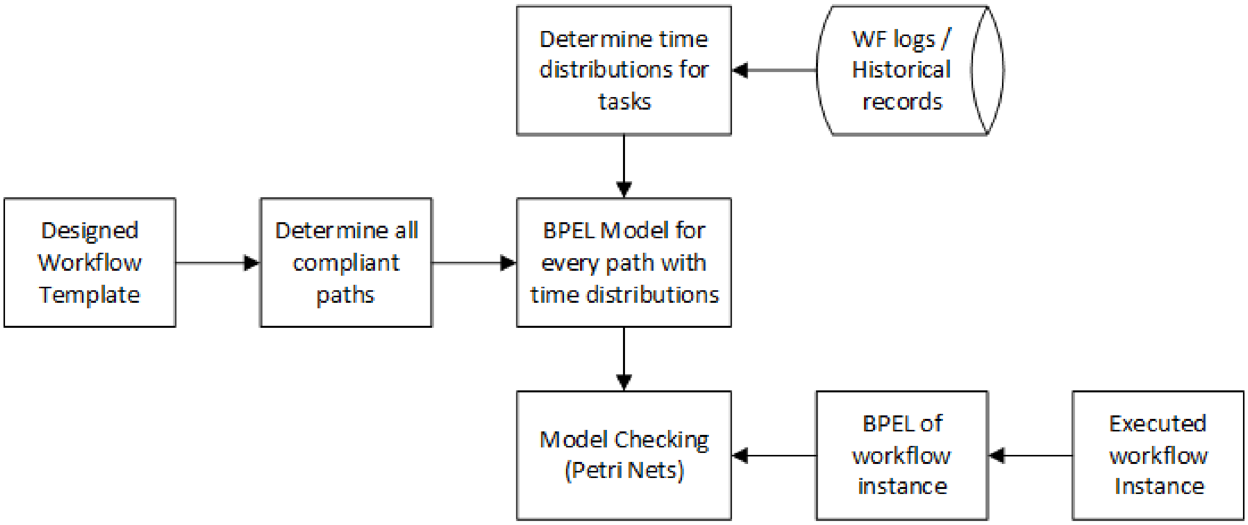

Determine time distributions for tasks
WF logs / Historical records
Designed Workflow Template
Determine all compliant paths
BPEL Model for every path with time distributions
Model Checking (Petri Nets)
BPEL of workflow instance
Executed workflow Instance

Figure 4.1: Framework to aid compliance checks

Figure 4.1 above, illustrates the methodology that has been developed to effectively determine if a workflow instance was executed compliantly. The framework consists of three different stages which occur independently. The first phase occurs when the workflow template has been designed and deployed for a construction project. The second phase consists of analyzing a training sample of workflow history logs to determine time distributions for each work item, and developing a composite Business Process Execution Language (BPEL) template with this additional information. The final stage is comparing a workflow instance, which has either executed or is in the process of execution for real-time monitoring, by model checking the two generated BPEL files or derived higher order implementations based on these BPEL templates for cases where additional complexity may be involved.

4.2.1 Compliance Check Framework Algorithm

While workflows automate the execution of work-items on an EPPM system as per a defined sequence so as to adopt pre-defined compliance policies, in certain cases the manner in which those tasks were completed externally, with respect to the EPPM system, might undermine those very policies. There is currently no way to detect this from within the system, and the sheer volume of workflow incidents make manual supervision of all cases impractical. An algorithm has been developed to aid efficient compliant checks based upon expected execution times for tasks within reasonable confidence limits. We acknowledge that this is one approach and others may be developed in the future.

Steps 1 through 3 occur during the creation or first examination of a workflow template. Steps 4 to 6 come into effect when there is a suitable sample size for analysis and can occur on a recurrent basis. Steps 7 through 9 can either occur during an audit or as a real-time monitoring process as an indicator for instances which may have violated compliance principles.

1. Design the workflow (The underlying assumption is that the template has been designed compliantly by the construction firm and vendor)
2. Develop the BPEL template
3. Determine all possible compliant paths (based on workflow history - either past projects / established templates or after a training set)
4. Determine time distributions for each work-item
5. Determine confidence limits for permissible execution times of each work-item
6. Develop consolidated workflow path with time distribution and integrate with BPEL or another appropriate derived model
7. For each executed instance develop the corresponding BPEL output
8. Find the actual time stamps based on the workflow history logs
9. Perform model comparison to see if the executed workflow was compliant

4.2.2 Determining All Compliant Paths for a Class of Workflows

Once a workflow template has been designed and deployed for a class of processes, such as Requests for Information or Change Requests or Interface Management, the first step is to determine all the possible compliant paths that an instance can undertake. In general, for a sound workflow, every single possible path from the first node to a concluding node may be considered a compliant path. The design of a workflow and determining the soundness of a workflow are beyond the scope of this research initiative, but there is sufficient literature that covers these subjects including van der Aalst's ground breaking paper on the "Application of Petri Nets to Workflow Management" (1998).

Assuming that the designed workflow is sound, one may immediately define all cases where a workflow terminates unexpectedly or with an error as non-compliant. These are usually cases wherein a task or series of tasks have timed-out or terminated with another labelled state denoting erroneous execution, and the workflow has been automatically closed or cancelled via manual intervention. Table A-2 in Appendix A presents the list of states created for the project that was analyzed for this research initiative. Having a pre-defined set of compliant paths mapped out and stored for a particular template eases the comparison and matching process for compliant and non-compliant instances, rather than having special cases written for every exception.

The out-of-the-box workflow developed by Coreworx Inc., for Requests for Information is presented on the next page. If one accounts for only the human activities – an unexpected termination of any one of the automated activities automatically triggers an error and requires manual intervention, nor does such a case indicate non-compliance as a result of task execution – there are only three possible compliant paths that every single workflow instance may undertake.

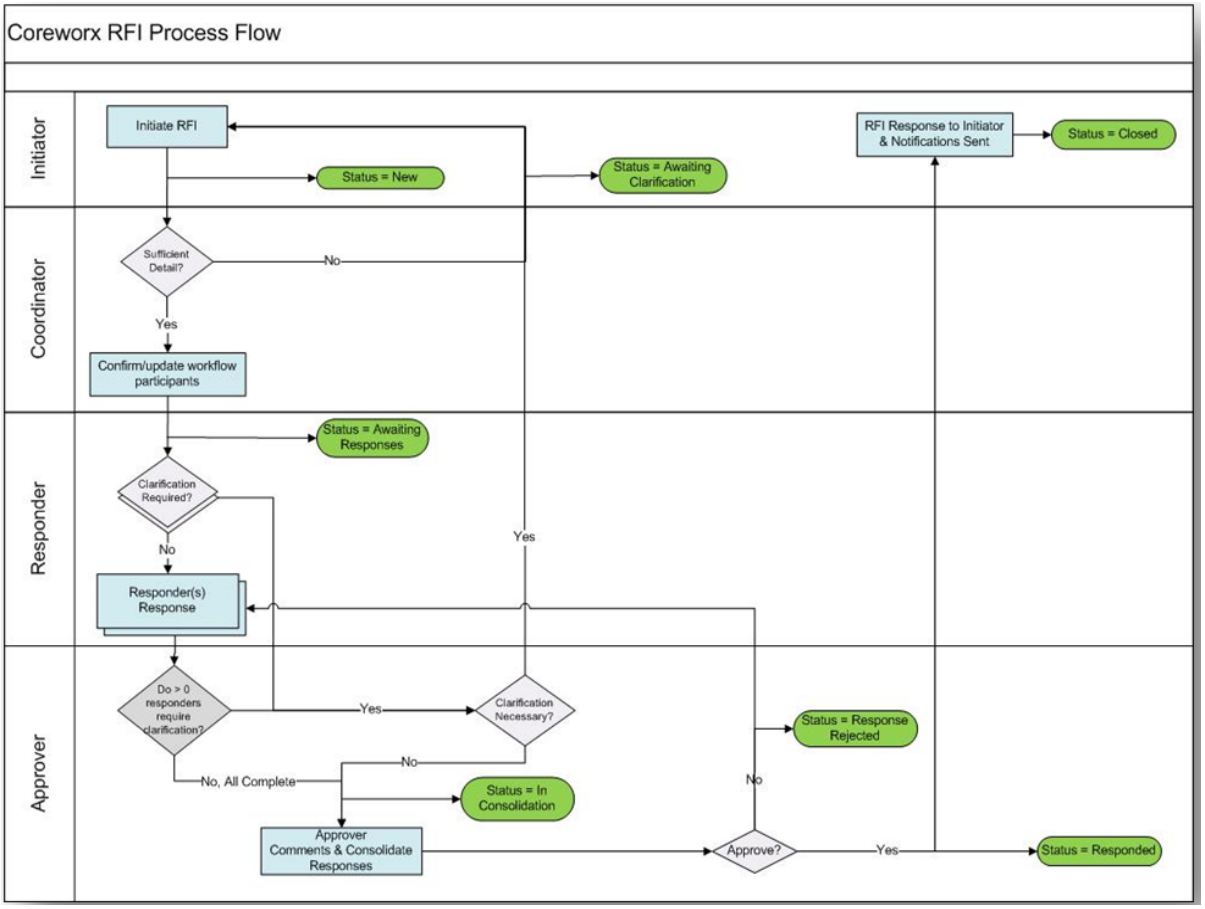

Coreworx RFI Process Flow
Initiator
Coordinator
Responder
Approver
Initiate RFI
Status = New
Status = Awaiting Clarification
RFI Response to Initiator & Notifications Sent
Status = Closed
Sufficient Detail?
No
Yes
Confirm/update workflow participants
Status = Awaiting Responses
Clarification Required?
No
Yes
Responder(s) Response
Do > 0 responders require clarification?
Yes
Clarification Necessary?
Status = Response Rejected
No
No, All Complete
Status = In Consolidation
No
Approver Comments & Consolidate Responses
Approve?
Yes
Status = Responded

Figure 4.2: Request For Information Workflow flowchart developed by Coreworx Inc©

The following three compliant paths may be considered for the RFI workflow pictured in figure 4.2 above:

1) Initiate RFI -> Verify Details -> Approve

2) Initiate RFI -> Verify Details -> Verify Participants -> Respond -> Review and Consolidate -> Approve

3) Initiate RFI -> Verify Details -> Verify Participants -> Respond -> Request Clarification -> Respond -> Review and Consolidate -> Approve

4.2.3 Development of Time Distributions for Workflow Activities

The compliant workflow paths above represent an entire workflow process case and the primary purpose for developing them is for a preliminary check to determine if the structure of a workflow instance matches one of these paths. In addition, as will be described later, these paths will be integrated with permissible time limits and converted to BPEL code for efficient matching with executed workflow instance logs. While the structure of all compliant permutations is a vital first step, the mere matching of structure does not indicate that a workflow instance was completed satisfactorily from a compliance point of view. Workflows consist of both a routing construct as well as individual work-items that constitute tasks that will be carried out by resources or actors.

Developing time distributions for workflow implementations, or the entire path, might be useful for a quick estimation of workflow instances that do and do not fit within the expected execution time for that particular path. Again this may not always indicate that all the component tasks were not completed compliantly. For instance, the overall time that an instance has taken to be completed might lie within the range expected for workflow implementation, however there may be one or more tasks which were not completed in accordance with established compliance policy. In such cases, the entire workflow instance can and should be considered to have not been successfully completed.

As such, developing an overall time-distribution for all workflow implementations has the potential to mislead an analyst, if all executed workflow instances are compared directly to it. Therefore it is important to determine the time distributions for all the component work-items and then conduct matches for work-items in conjunction with matches for the structure. This has the further advantage of raising an alert before, during or immediately after a work-item has been completed so as to permit corrective action before the workflow is closed, if real-time monitoring is implemented for the EPPM system.

It is assumed that completion time for work-items follows a probability distribution; i.e. for the same work item, different specific instances will be completed at different times which can be represented by a probability density function. To develop such a time probability distribution model it is important to first find a representative sample of data. There are several methods that can be employed to determine the best-fitting probability distribution including the method of moments, the maximum likelihood method and probability paper plotting. In addition there are several software packages such as “Easy Fit”, “StatGraphics” and various libraries for software packages which are able to determine the best fitting distribution for a sample of data. For all practical implementations such an application will have to be integrated within the EPPM system to determine the probability distribution. Figure 4.3 below illustrates the probability plotting method employed for determining the suitability of the Weibull distribution for a work item from available data:

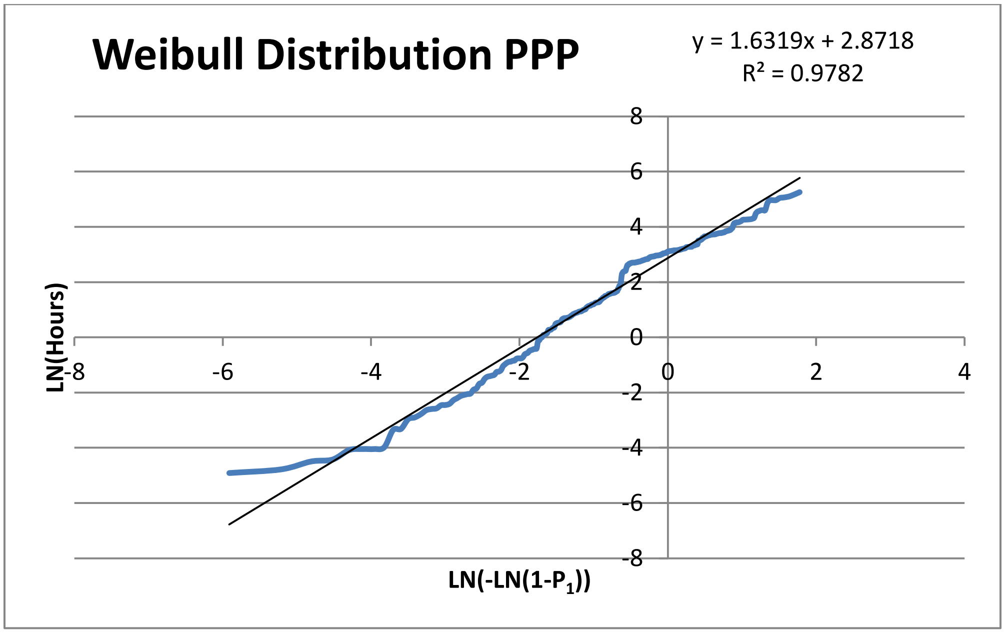


Figure 4.3: Probability Paper Plot for Weibull Distribution Based Verify Details Work Item

Once a distribution has been found and deemed to sufficiently fit the data, goodness of fit tests may be conducted to determine the accuracy of fit and if the probability distribution may be adopted for that work-item. The Anderson Darling test has been used to determine the goodness of fit for this research initiative, due its sensitivity to the tails of distribution, which are important for the considerations of this approach. The software tool EasyFit was used for determining the distribution and goodness of fit test results for the case study section of this chapter. It should however be noted that there is the potential for distributions to be multi-modal in nature, reflecting different response classes (for example, an immediate response versus a response after a period of time that may be attributed to the gathering of necessary information). As described in Section 4.3.11, such distributions were approximated to the nearest resembling uni-modal distribution as we are primarily interested in the tails of the entire data set and the added complexity of analyzing multimodal distributions would also be difficult to implement in practice.

Complete snapshots of relevant data sample sets and selection of distribution based on comparisons of the goodness of fit tests from EasyFit are presented in Appendix A. For a detailed reference of the employed methodology and goodness of fit tests, the reader is directed to the course resource by Pandey and Jyrkama's "CivE 601 Engineering Risk and Reliability", which was this author's primary reference for this dissertation on distribution selection and tests for goodness of fit.

Once a probability distribution has been identified and confirmed as a suitable fit to describe the time distribution of a work-item, it is possible to define confidence in the execution times for the vast majority of instances of that work item. However the "tails" of these distributions represent the polar extreme probabilities of times taken to complete tasks. In other words, one may attribute these cases to be the least likely of cases to occur. One may say with confidence that the probability of a work-item being completed within the range of times represented by the inverse cumulative density function from the 2.5th percentile to the 97.5th percentile is 95%, given by the following equation:

$$P(x_{0.025} \leq x \ \leq x_{0.0975}) = 0.95$$

Confidence intervals may thus be established for the range of times that are considered normal for the completion of a work-item, which represent the usual amount of time that can be allocated for the work-item to have been completed compliantly. Times which are extremely small or extremely long on the other hand indicate that a task may have been subverted and that perhaps the work-item in question needs to be examined further (for example, flagged for higher managerial attention). Establishing appropriate confidence limits for what constitutes the compliant completion of a work-item is largely at the discretion of the firm that chooses to employ the compliance checking algorithm, but standard confidence limits employed in engineering range from 5 to 10 percent, or the 2.5th and 97.5th percentiles and 5th and 95th percentiles respectively. This provides for more precise checking based on both the minimum and maximum expected time for the completion of a work-item based on workflow history from the same project. This represents an improvement over conventional implementations, wherein a notification is sent only if a task is not completed by an arbitrary defined constant time, such as 3 days per task in the case study. Of course this time limit may directly reflect contract requirements and thus addresses another aspect of compliance.

4.2.4 Consolidated Business Process Execution Language Representation

The analysis conducted on workflow history logs and the subsequent time distributions that have been determined have to be modelled for comparative analysis of executed workflow instances. It is important to consider the extensibility and scalability of such a model, so as not to limit the comparative analysis to the confines of conventional technology developed for one particular research thrust. It is also important to consider current industry trends and models that have been adopted by the workflow management community at large. While graphical front end languages are useful for human interpretation, machine readable code is far more efficient for rapid transformation to other language constructs and comparative analysis.

As such, language constructs such as the Business Process Model and Notation (BPMN), while representative of workflow notation, are process semantics and would require extensive transformation for analysis considerations. The Business Process Execution Language (BPEL) on the other hand is a universally accepted executable orchestration language that is recognized by OASIS (The Organization for the Advancement of Structured Information Standards). An example of BPEL code generated for one of the compliant paths in this project is presented below:

```
<bpel:process xmlns:bpel="http://docs.oasis-
open.org/wsbpel/2.0/process/executable">
  <bpel:sequence>
    <bpel:empty name="RFI Draft Initialization"/>
    <bpel:if name="Verify Details">
      <bpel:sequence>
        <bpel:empty name="Verify Participants"/>
        <bpel:if name="Respond">
        <bpel:empty name="Review and Consolidate"/>
      </bpel:sequence>
      <bpel:elseif/>
    </bpel:if>
    <bpel:empty name="Approve"/>
</bpel:sequence>
</bpel:process>
```

BPEL (formally renamed to WS-BPEL; the prefix WS is an acronym for Web Services) is widely supported in the industry by several leading software companies such as Microsoft, IBM, SAP and Seibel Systems. An orchestration executable language is one wherein communication with other systems is controlled by orchestration engines. For example, in the code above, the presence of "if-else" constructs provides for structured programming. BPEL is also XML compliant, meaning that it is both human and machine readable, consisting of markup and content. This also allows for the creation of elements and attributes to extend the capabilities of the language. While there are a few other languages that may also be used to aptly represent workflows in an XML format, such as Yet Another Workflow Language (YAWL), the widespread adoption of BPEL in industry and therefore its portability were important factors in adopting it for this research initiative.

It is proposed to create additional attributes to augment the capabilities of standard BPEL for the purpose of integrating time-distribution related data. The additional attributes include the upper and lower limit confidence limits for anticipated workflow executed times. Each attribute will hold the corresponding value in a unit of time. For example, if it has been determined that the execution times for work-item Approve follows a normal distribution, and if the 5th and 95th percentiles of this normal distribution are five minutes and one hour respectively, then the corresponding additions to the BPEL code would be:

```
<bpel:empty name="Approve" upper_conf="3600" lower_conf="300"/>
```

4.2.5 Workflow Instance and Model Comparison

After the model template for a workflow implementation that defines the 95% confidence interval for the durations that work-items take has been defined, subsequent executed workflow instances may be compared with it. The data associated with each workflow instance is stored in the workflow history logs. A corresponding BPEL representation of the workflow instance may be reconstructed from the workflow's logs. As with the model, additional attributes that represent the time taken for the completion of each activity need to be specified for each instance. An example of how the relevant BPEL code would appear for an executed workflow instance is provided below:

```
<bpel:process xmlns:bpel="http://docs.oasis-
open.org/wsbpel/2.0/process/executable">
  <bpel:sequence>
    <bpel:empty name="RFI Draft Initialization" exec_time=”100”/>
    <bpel:if name="Verify Details" exec_time=”400”>
```

```
      <bpel:sequence>
        <bpel:empty name="Verify Participants" exec_time=”50”/>
        <bpel:if name="Respond" exec_time=“600”>
        <bpel:empty name="Review and Consolidate" exec_time=”500”/>
      </bpel:sequence>
      <bpel:elseif/>
    </bpel:if>
    <bpel:empty name="Approve" exec_time=”100”/>
 </bpel:sequence>
</bpel:process>
```

Care should be taken to ensure that the time calculated matches both the format and the manner in which the model time was calculated. For example, if the model’s time was estimated taking into consideration only working man hours, rather than calendar hours, and displayed in seconds, then the workflow instance attributes should be calculated identically. For practical purposes wherein just executed times have to be compared, it may well be sufficient to simply create a script that checks if the executed times for all work-items fall within the established 95% confidence intervals for every corresponding work-item in the developed model.

However, it is also important to consider extensibility of the model. Workflow petri-nets, proposed by van der Aalst (1998), are implementations of workflows that allow for thorough analysis since a workflow is provided with a strong underlying mathematical basis. Aalst argues that workflow petri nets are better suited for concepts like soundness, than other methods such as Pi calculus (Aalst 2003). A Petri-net workflow based on an RFI instance, presented below, would be able to consider additional features such as resource allocation etc., if the model is developed to accommodate these concepts later.

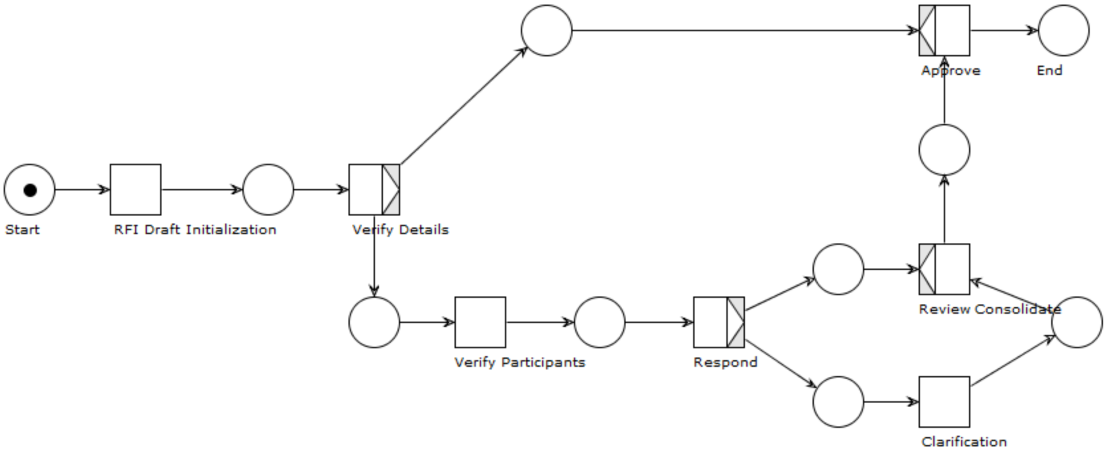


Figure 4.4: Petri Net diagram of the RFI template (developed using WoPeD© tool)

4.3 Validation Case Studies

4.3.1 Overview

To validate the methodology developed in the preceding sections, a case study was conducted on data obtained from one of Coreworx's clients. The client, an EPC firm, was developing a series of construction projects in British Columbia. Workflow related data from the development of the first construction project was provided and integrated into a virtual machine with Coreworx's system installed on it. The data consisted of 1070 RFI workflows and 925 "Trends" workflows. The "Trend" workflow was developed specifically for a client as an advanced change request workflow which accommodated budget changes and the impacts these changes would have on the project. The RFI case study is described first in the subsequent sections, followed by the Trend case study and conclusions.

4.3.2 RFI Workflow Case Study

When the system was installed at the client's site, the first few workflow instances were developed for testing the system as well for training imparted to the clients by Coreworx. Therefore the first 25 workflow instances for all workflow classes were ignored as they did not accurately reflect real project data. It should also be noted that workflow identification numbers (id) represented all classes of workflows that were executed on the system, such as Change Requests, Transmittals etc., in addition to Request for Information workflows. As such the unique workflow ids created for each RFI workflow instance may not match the consecutive RFI ids developed for each RFI document, as reflected in the training data snapshots in Appendix A.

The RFI template that was deployed for the project is displayed in the next section along with all possible compliant paths that could have been generated. The data was then cleaned as described in the subsequent sections and divided into training and testing data sets for the development of time distributions.

4.3.3 RFI Template and Compliant Paths

Figure 4.5 below displays the RFI template that was employed at the site. All textual information related to the workflow schema has been removed from Figure 4.5 as per the non-disclosure agreement between industry and this research initiative. Therefore, the names of the activities and the conditions for the execution of some of these activities or selection of a workflow path may not be immediately apparent; however the process has been explained in the following pages. Although there are a large number of work items, most of these represent automated system actions, such as saving a pdf file, or recording data and storing it within the system's database.

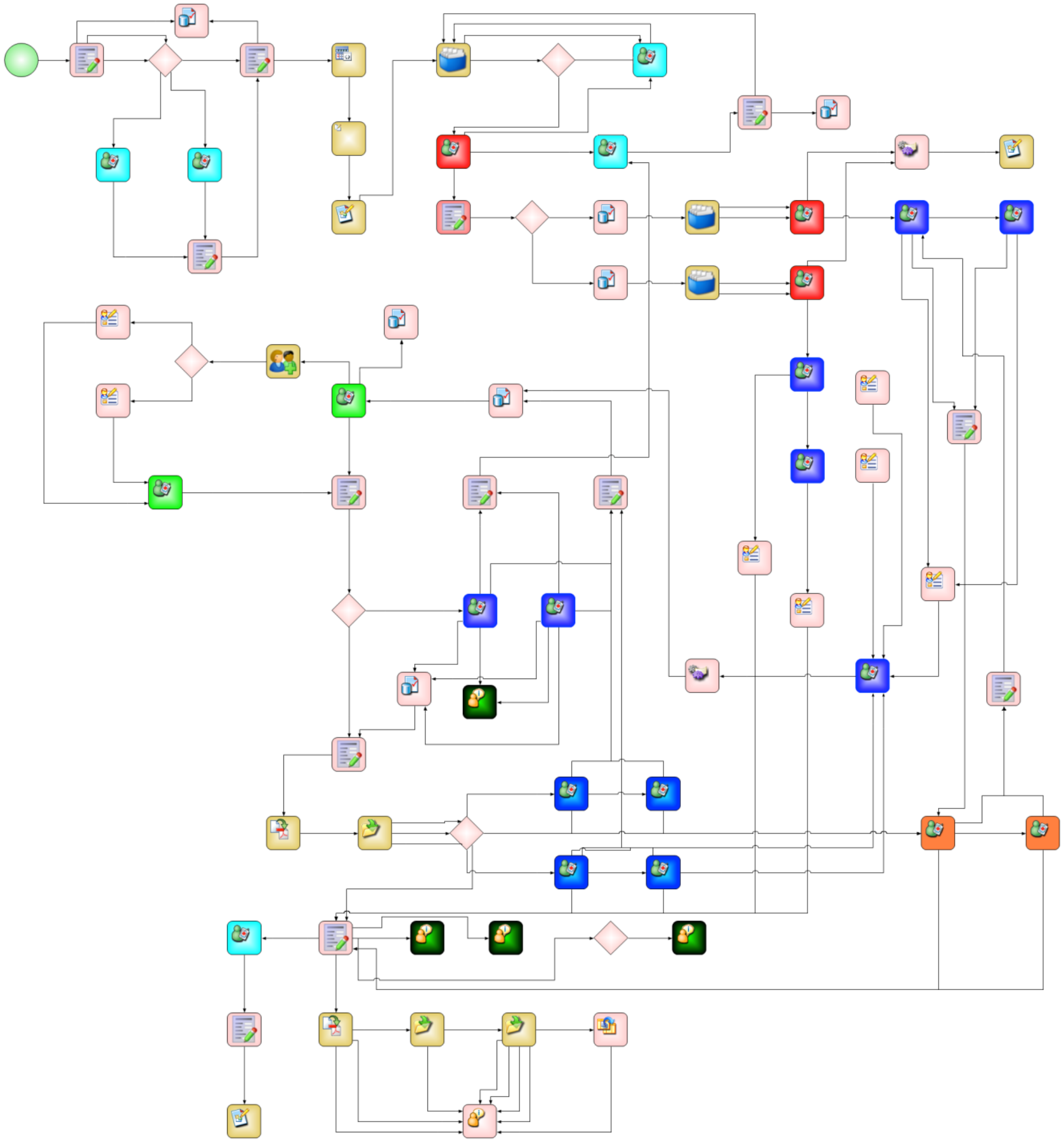


Figure 4.5: RFI Workflow Representation Used in Case Study (© Coreworx Inc.)

All the icons in Figure 4.5 which have a green figurine transcribed within them represent actions that involved humans. However many of these constitute work-items that do not comprise a human completing a task but rather being involved, such as an automated warning sent to a user informing that he or she has not completed his or her task prior to the time limit set for it.

It is important to therefore decompose the above RFI template into component tasks which are completed by users rather than the system and to combine similar categories of tasks in order to develop accurate time distributions and compare these with future workflow instances. On combining classes of similar tasks, we arrive at the same basic RFI flowchart illustrated in figure 4.2, however given the paucity of clarification instances in the available data set, these cases were not considered for analysis. The two compliant paths may be inferred from the Petri Net workflow of the RFI displayed below:

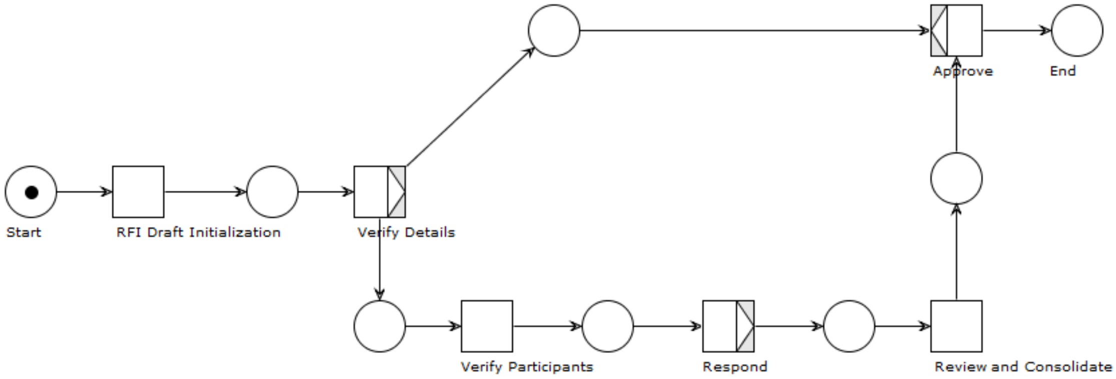


Figure 4.6: Petri Net Workflow Developed for Case Study Analysis

4.3.4 RFI Workflow Data Cleaning and Analysis

The raw data consisted of 1070 Request for Information workflow instances. These workflow instances contained a total of 22840 work items that involved a user. An extremely high number of these work items had to be rejected as the completion time was recorded as a 'NULL' value and therefore could not meaningfully be used for analysis. Of the remaining 6536 work-items, when warning and notification work-items – which do not constitute a task but rather an alert sent to a user that they have not completed a task within an established timeframe which was set at a default of 3 days – were removed, the number of work-items that remained was 5605.

These work-items were distributed over 996 workflow instances, ranging from workflow id '25' to workflow id '3102.' Establishing a sufficient sample size to obtain representative time distributions is a critical factor, and so the data was divided into two sets – one for developing time-distributions and the other to test the number of instances that would require an additional compliance check.

In order to simulate a system which would incorporate the methodology to the greatest extent possible, it was decided to pick the first 'n' workflow instances rather than pick random instances from the entire data, and since the workflows ranged from 25th to 3102nd, the first 2000 workflow instances were set as the cut off for analysis. This corresponded to 655 workflow instances, leaving a total of 341 for testing, which approximately works out to two thirds for analysis and one third for testing.

It should be noted however, that each workflow instance does not contain every work-item element, nor do all workflow instances have an equal number of work item instances. The number of constituent work-items was largely dependent on the route that was employed for that specific RFI case. Further some workflow instances employed the same work-item more than once depending on whether the same class of work-items was required to be completed by several users, as in the case of the work-item "Respond", or in other cases when a clarification was required and a loop was initiated. The number of such clarifications within the available data set was very low and not sufficient to conduct analysis. Therefore the number of work-items across these 996 workflow instances varied greatly. Care was taken however to ensure that there sufficient representative samples were selected in each case.

Table 4.1 on the next page shows the number of work-items that were considered for each of the stages as well as the percentages of testing and training instances against the entire set of instances. In general about 70% of the instances in the sample were used for training and around 30% for testing. There is the possibility that some task time distributions may exhibit co-variances (for example, pattern recognition on sequences of tasks may have revealed relations to the execution times), but analyzing these were beyond the scope of this thesis.

Work Item	Training Size	Testing Size	Total	Training Percentage	Testing Percentage
Verify Details	759	395	1154	68.89%	31.11%
Verify Participants	1105	472	1577	70.07%	29.93%
Respond	1066	387	1453	73.37%	26.63%
Review and Consolidate	423	122	545	77.61%	22.39%
Approve	637	239	876	72.72%	26.28%
Total	3990	1615	5605	71.19%	28.81%

Table 4.1: Distribution of Work Items For Training and Testing

It may be noted that the first work-item in every RFI, “Request for Information Draft Initialization”, does not appear in the above table. This is unfortunately the result of every single such work item instance’s completion time recorded as “NULL” in the software implementation. Further, there were only 124 instances wherein a clarification work-item was initiated, and the completion time for 25 of those instances was recorded as “NULL.” In all such cases where there are an insufficient number of work-items to represent a sufficiently sized sample set, the work item has to be ignored from the analysis for both the model and the instances for comparison. While this does compromise the compliance of the workflow to an extent for a real system, for the exploratory purposes of this dissertation, the analysis proceeded with the available data. As such, the workflow route that included the clarification element was discarded, however for all other workflow instances, the RFI Initialization work item was omitted. In an ideal EPPM system, there will be well defined start and end times and cases where a NULL value has been recorded will have to be investigated by an auditor.

The start and end times for a work item represent the times that the work-item was issued to a user and the time that work item was completed, as registered on the system. These times do not, however, accurately reflect the time in work hours that was spent directly on the task. Consider for example, if a work item was issued at Friday at 10 p.m. and the user completed the work item at Monday around 8am. Merely calculating the time between those two hours would yield 58 hours, without giving consideration to the work hours put in by the user, which could have in reality been as low as a few minutes.

Therefore, all times had to be corrected to reflect as accurately as possible the work hours put in to complete a task. Weekend hours were deducted from all times. Although vendors recollected that two shifts had been created for the project, these definitions were not part of the available database and so a reasonable assumption was made for shifts to be from 7am to 3pm and 3pm to 11pm, after studying the start and completion times for work items. The work items stripped of these non-working hours were then ready for the first stage of data analysis. Data snapshots for each work item have been presented in Appendix A.

Upon analyzing the data, it was immediately apparent that several of the data points were noisy and could not have accurately reflected the time taken to complete a task. For instance, in the workflow template design, if a user had not completed a task within 3 days of it being issued, an alert notification was sent requiring immediate action. It is therefore safe to assume that it was expected that most tasks would be completed within 3 working days, or given two 8 hour shifts, a total of 48 hours, with some leeway to account for finishing a task after an alert notification was sent out. However, in some cases a work item's completed and corrected time was higher than 3000 hours, which seems to suggest a faulty (as in a software recording error, or a workflow instance that was supposed to be cancelled but escaped the attention of the coordinator as it was no longer of an direct importance to the project) workflow instance. It should be noted that these tasks may represent a dropped task and should be flagged for review by higher management. This range of values would surely fall outside the extreme outliers in a box plot equation, $x > Q_3 + 30 \times IQR$

Where Q_3 represents the 3rd quartile and IQR represents the interquartile range. This became more apparent when the histograms for each work item were mapped, as displayed in the figures 4.7 to 4.11 below:

4.3.4.1 Verify Details

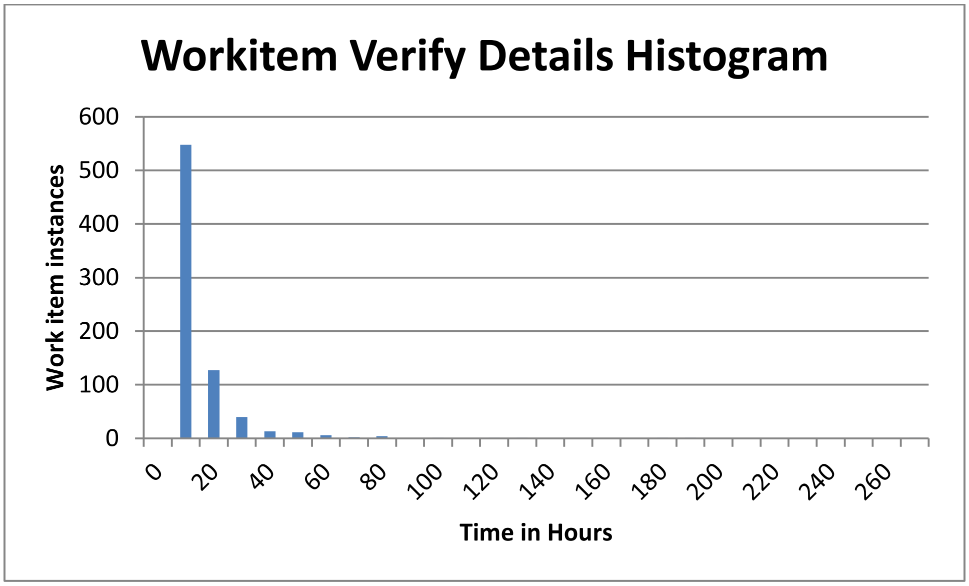


Figure 4.7 Histogram of Hours vs Instances for Work Item Verify Details

The Verify Details work item instances v Time in Hours histogram is similar to all the other work item histograms plotted against time. The vast majority of instances are completed within the first bin, or the first few bins, giving all the histograms symbolic right skews. It is safe to assume that the outlying instances are the result of a workflow that has violated the initially established timeframe for work item execution, a default of 3 days being defined in the original template. The following pages display the histograms for each of the other activities. It should also be noted that often the time limits such as 3 days to complete a task are part of contractual agreements between construction parties.

4.3.4.2 Verify Participants

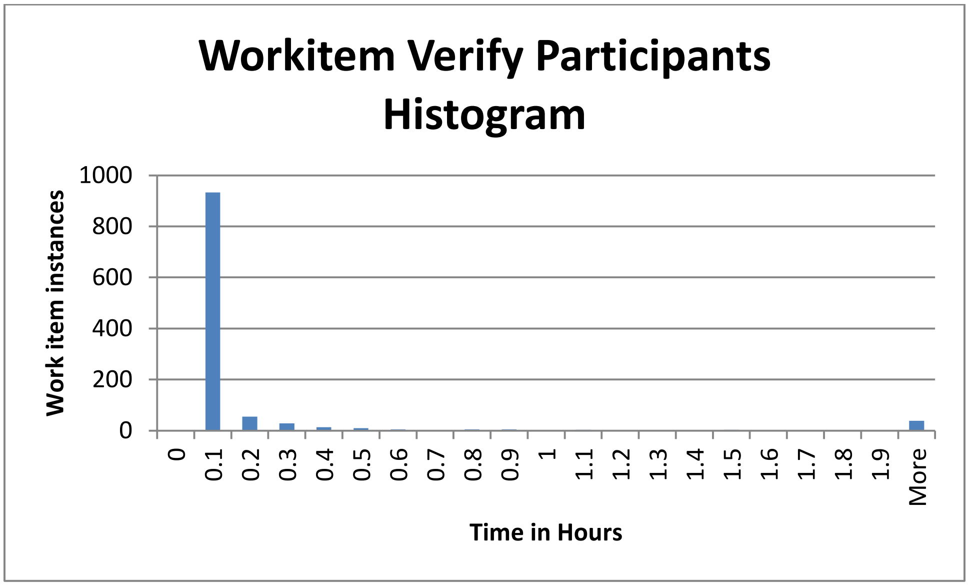


Figure 4.8 Histogram of Hours vs Instances for Work-item Verify Participants

4.3.4.3 Respond

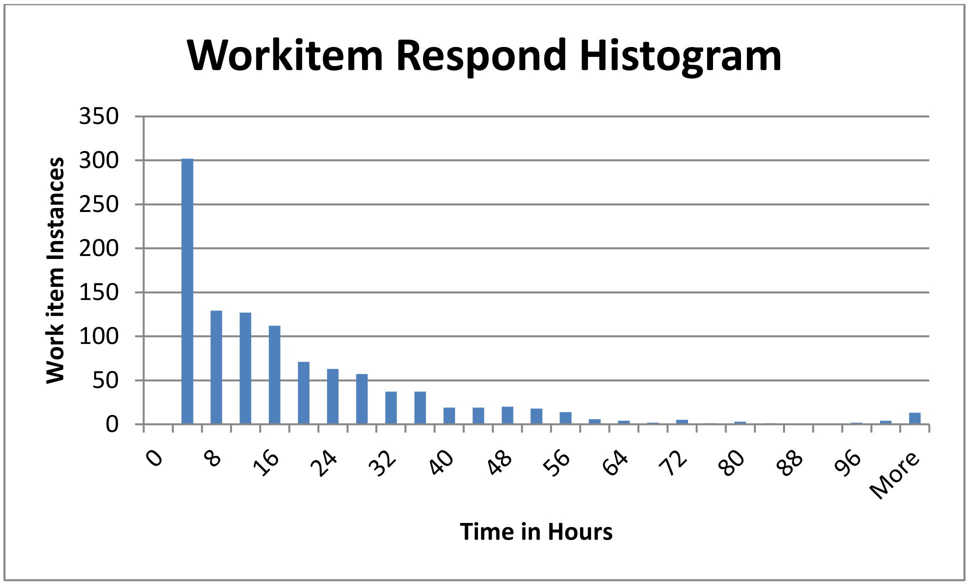


Figure 4.9: Histogram of Hours vs Instances for Work-item Respond

4.3.4.4 Review and Consolidate

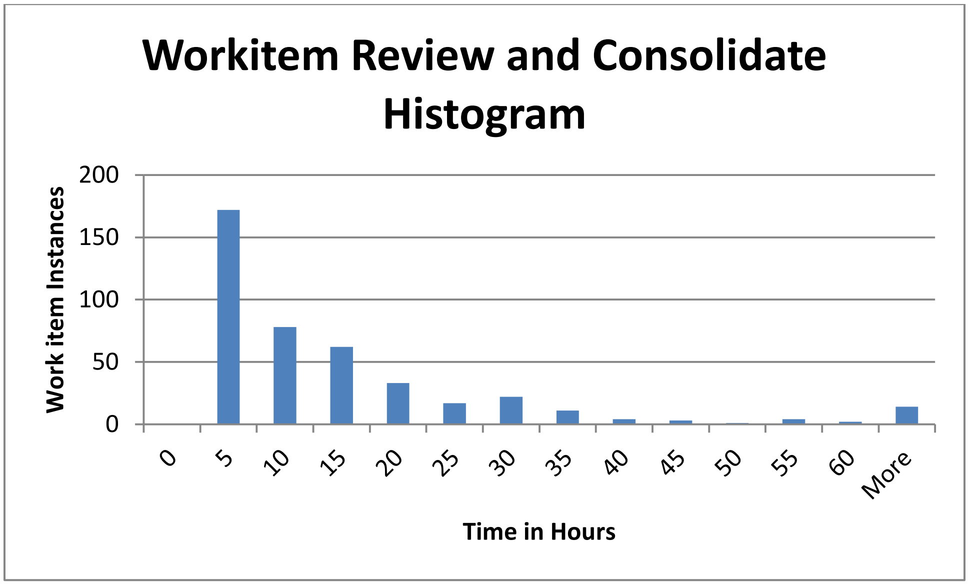


Figure 4.10: Histogram of Hours vs Instances for Work-item Review and Consolidate

4.3.4.5 Approve

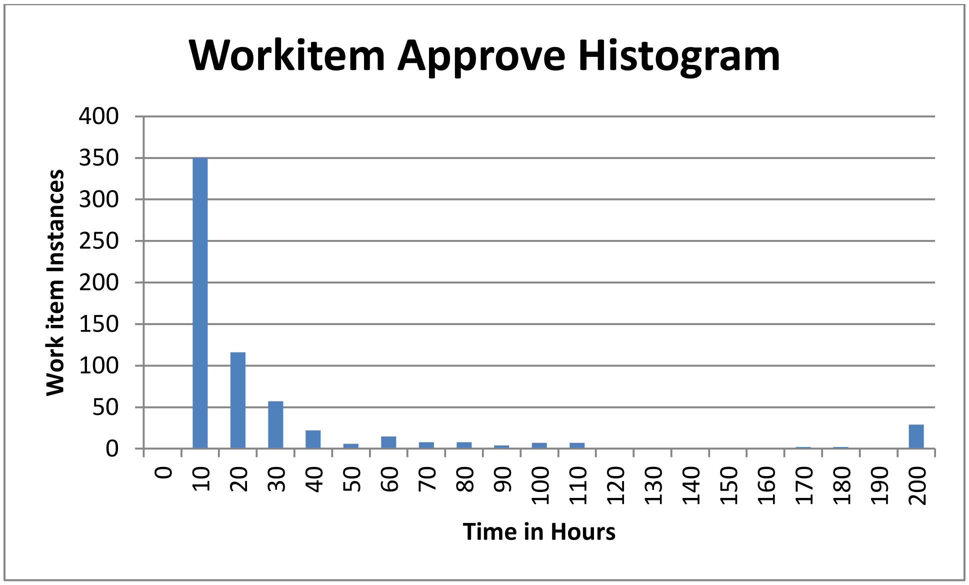


Figure 4.11: Histogram of Hours vs Instances for Work-item Approve

Maintaining these extreme values in the analysis would provide misleading probability time distributions for each work item and defeat the purpose of determining an effective time for the completion of the work item. Therefore the data was further cleaned to remove all such values and the time distributions for each work item were developed.

4.3.5 Time Distributions for RFI Work Items

The cleaned training data sets were analyzed using the software package EasyFit©. The time distributions that fitted the data best as per the Anderson Darling Goodness of Fit test were accepted for each work-item. The Probability Density Functions as well as the goodness of fit test results are displayed for each work item below:

4.3.5.1 Verify Details

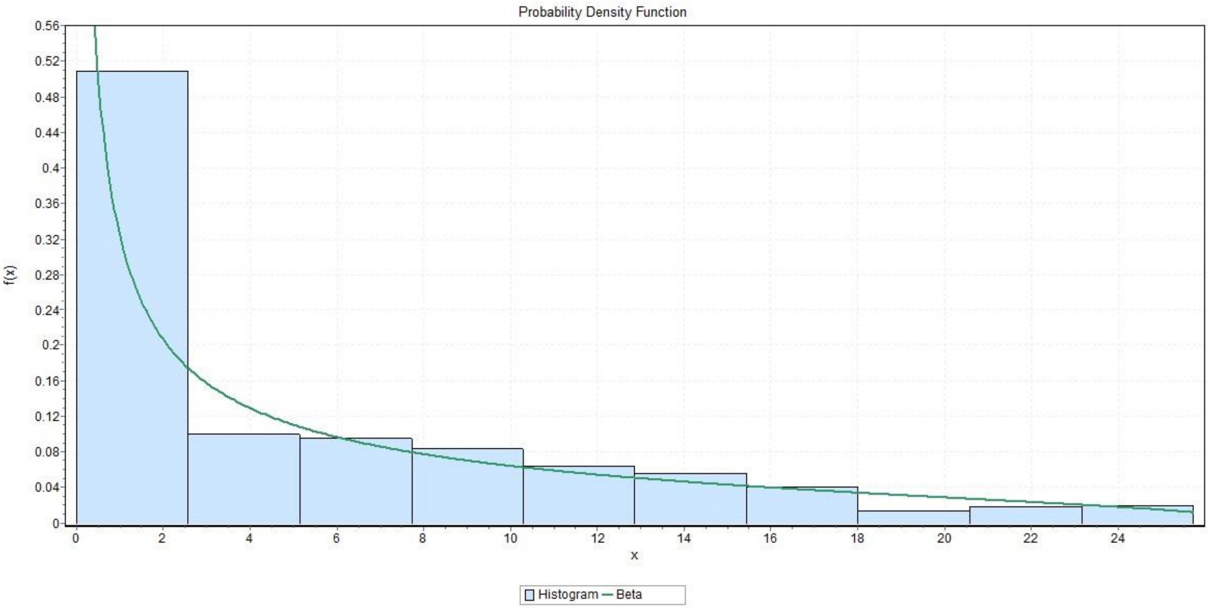


Figure 4.12: Probability Density Function For Verify Details Beta Distribution

<table>
<tr><td>Beta Distribution</td><td>α₁=0.37509</td><td>α₂=1.4019</td><td>a=0.00597</td><td>b=27.007</td></tr>
<tr><td colspan="5">Anderson-Darling Goodness of Fit Results</td></tr>
<tr><td>Sample Size</td><td colspan="4">707</td></tr>
<tr><td>Statistic</td><td colspan="4">1.966</td></tr>
<tr><td>Rank</td><td colspan="4">1</td></tr>
<tr><td>α</td><td>0.1</td><td>0.05</td><td>0.02</td><td>0.01</td></tr>
<tr><td>Critical Value</td><td>1.9286</td><td>2.5018</td><td>3.2892</td><td>3.9074</td></tr>
<tr><td>Reject?</td><td>Yes</td><td>No</td><td>No</td><td>No</td></tr>
</table>

Table 4.2: Beta Distribution for Verify Details: Parameters and Goodness of Fit Results

4.3.5.2 Verify Participants

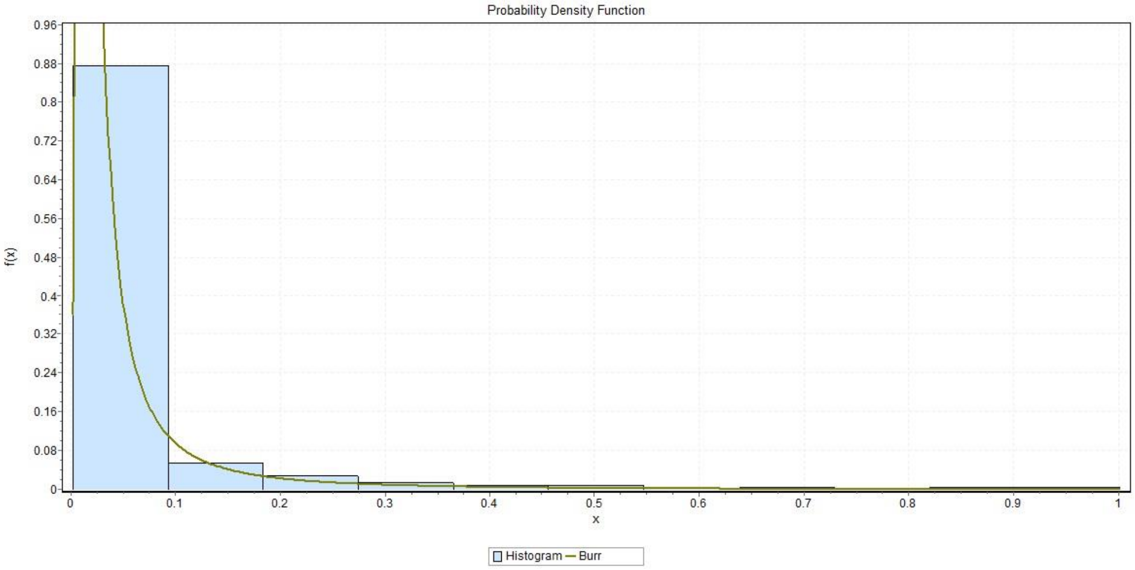


Figure 4.13: Probability Density Function For Verify Participants Burr Distribution

Burr Distribution Parameters	k=0.36014		α=2.8941		β=0.01105
Anderson-Darling Goodness of Fit Results					
Sample Size	1057				
Statistic	0.80391				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.3: Burr Distribution for Verify Participants: Parameters and Goodness of Fit Results

4.3.5.3 Respond

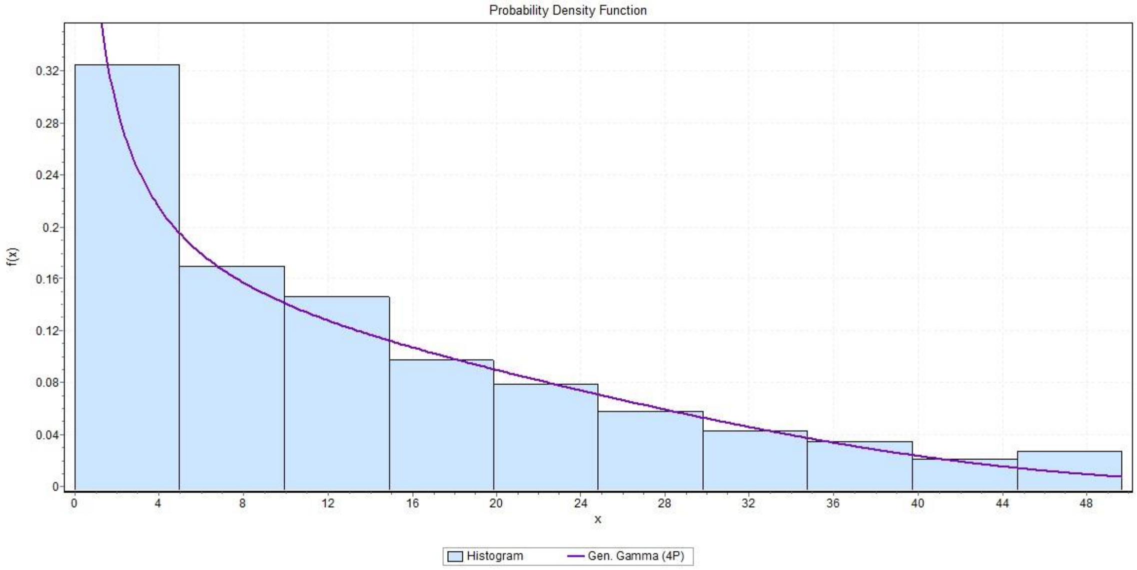


Figure 4.14: PDF For Respond Generalized Gamma 4 Parameter Distribution

Gen. Gamma (4P) Distribution Parameters	k=2.8169	α=0.19982	β=37.662	γ=0.00696	
Anderson-Darling Goodness of Fit Results					
Sample Size	1002				
Statistic	1.2374				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.4: Generalized Gamma 4 Parameter Distribution for Respond: Parameters and Goodness of Fit Results

4.3.5.4 Review and Consolidate

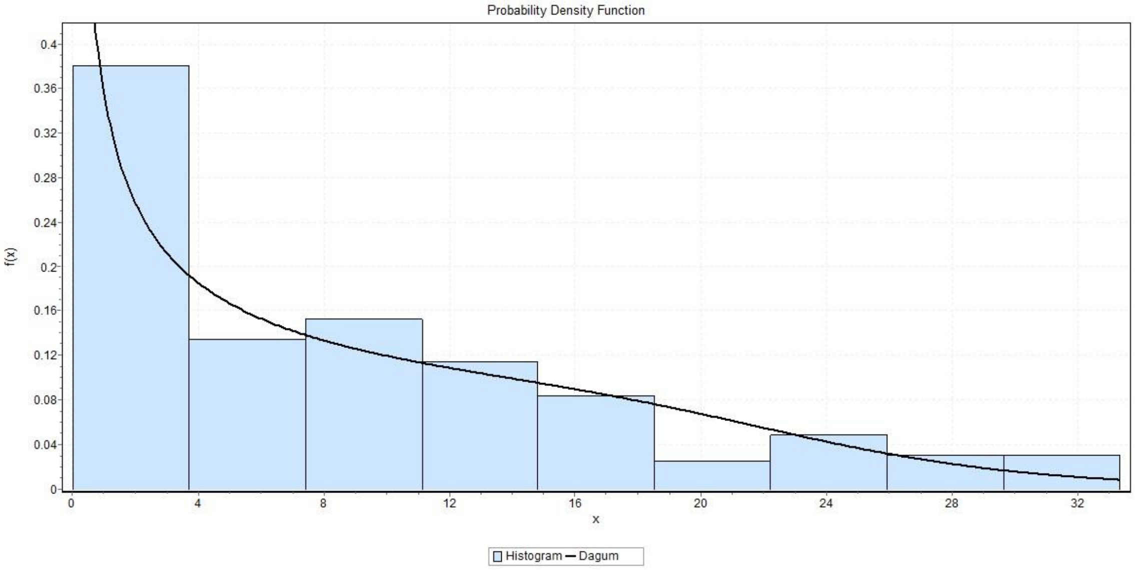


Figure 4.15: Probability Density Function For Review and Consolidate Dagum Distribution

Dagum Distribution Parameters	k=0.08728		α=6.0443		β=25.049
Anderson-Darling Goodness of Fit Results					
Sample Size	394				
Statistic	0.98349				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.5: Dagum Distribution for Review and Consolidate: Parameters and Goodness of Fit Results

4.3.5.5 Approve

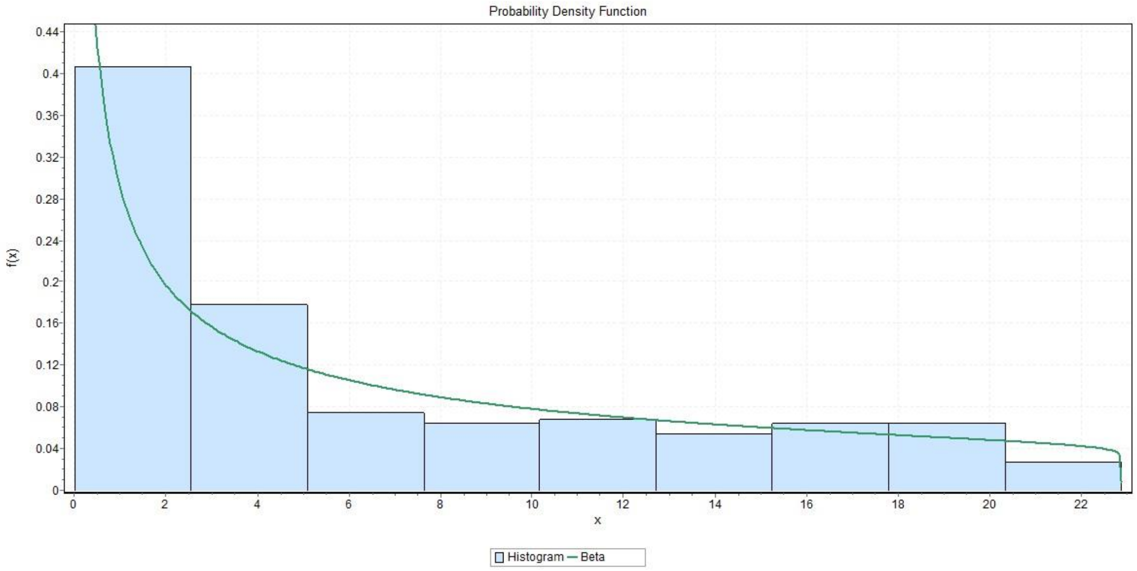


Figure 4.16: Probability Density Function For Approve Beta Distribution

Beta Distribution Parameters	α_1=0.4462	α_2=1.0642	a=0.01115	b=22.869	
Anderson-Darling					
Sample Size	484				
Statistic	1.6926				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	Yes	No	No	No	No

Table 4.6: Beta Distribution for Approve: Parameters and Goodness of Fit Results

4.3.5.6 Consolidated Work Item Time Distribution Confidence Intervals

Task	Time Distribution	2.5th Percentile time (Hours)	97.5th Percentile time (Hours)
Verify Details	Beta	0.00691	22.756
Verify Participants	Burr	0.00447	0.381
Respond	Generalized Gamma 4 Parameter	0.05308	43.606
Review & Consolidate	Dagum	0.02301	29.994
Approve	Beta	0.01657	21.334

Table 4.7: 2.5th and 97.5th confidence limits for work item time distribution

Table 4.7 above illustrates the derived distributions for the tasks in the workflow as well as the 95% interval of times for which these tasks were executed.

4.3.6 Integrated BPEL Representation of RFI Workflow Instance

The following is the BPEL generated code from the WoPeD application for the workflow implementation that was considered for this analysis, with the integrated confidence limits:

```
<bpel:process xmlns:bpel="http://docs.oasis-
open.org/wsbpel/2.0/process/executable">
  <bpel:sequence>
    <bpel:empty name="RFI Draft Initialization"/>
    <bpel:if name="Verify Details" upper_conf=“22.756” lower_conf=“0.0069”/>
```

```
      <bpel:sequence>
        <bpel:empty name="Verify Participants" upper_conf=“0.381”
lower_conf=“0.0045”/>
        <bpel:if name="Respond upper_conf=“43.606” lower_conf=“0.0531”/>
        <bpel:empty name="Review and Consolidate"  upper_conf=“29.994”
lower_conf=“0.023”/>
      </bpel:sequence>
      <bpel:elseif/>
    </bpel:if>
    <bpel:empty name="Approve"  upper_conf=“21.334” lower_conf=“0.01657”/>
 </bpel:sequence>
</bpel:process>
```

It should be noted that the above BPEL code does not contain confidence limits for the “RFI draft initialization” work-item as these values could not be obtained since the end times for all cases was saved as ‘NULL’ in the database. However the model has to include the node in order to display the correct workflow implementation path, so that if applications generated higher order representations such as workflow petri-nets, these work-items are represented appropriately.

4.3.7 Matching Test RFI Instance Data with the Model

The BPEL of all testing sample instances would have to be generated from the workflow history logs and compared with the BPEL of the integrated confidence limit implementation model developed above. The BPEL generated output for one such instance (workflow ID = 2734), including the executed times, is displayed below:

```
<bpel:process xmlns:bpel="http://docs.oasis-
open.org/wsbpel/2.0/process/executable">
  <bpel:sequence>
    <bpel:empty name="RFI Draft Initialization"/>
    <bpel:if name="Verify Details" exec_time=”0.1829”>
      <bpel:sequence>
        <bpel:empty name="Verify Participants" exec_time=”0.0014”/>
        <bpel:if name="Respond" exec_time=“0.486”>
        <bpel:empty name="Review and Consolidate" exec_time=“0.0243”>
       </bpel:sequence>
      <bpel:elseif/>
    </bpel:if>
    <bpel:empty name="Approve" exec_time=”0.0232”/>
 </bpel:sequence>
</bpel:process>
```

A simple script, rather than workflow Petri-nets which are not required for this level of analysis and would employ unnecessary additional computing expense, compared the values for each corresponding work-item. For example, the workflow instance above would require an additional check as the execution time for the work-item “Verify Participants” is lower than the established lower confidence limit from the developed time distribution.

4.3.8 Summary and Interpretation of RFI Case Study Results

Table 4.8 on the next page summarizes the resulting number and percentage of testing work item instances that did not fall within the confidence interval of time distributions for each class of work-item during the comparison between the testing samples and the integrated model.

Work-Item	Testing Sample Size	Cases Outside Confidence Interval Times	Cases over 3 working days (48 hours)	Percentage requiring additional check	Percentage determined solely by framework (excluding > 3 day duration)
Verify Details	395	54	12	13.67%	10.63%
Verify Participants	472	31	2	6.57%	6.14%
Respond	387	25	15	6.46%	2.58%
Review and Consolidate	122	17	3	13.93%	11.48%
Approve	239	73	43	30.54%	12.55%

Table 4.8 Percentage of compliant instances for each work item

Cases that fall out of the 95% confidence interval have been categorized into two types: (i) The total number of cases that fell outside of the range and (ii) all cases wherein the work-item time was over 3 working days, assuming two 8-hour working shifts in a day. The latter case has been separated because it represents the number of cases that would have triggered an alert notification for the current default setting in the EPPM system, and therefore cases which would have been brought to the notice of a supervisor without the implementation of the compliance checking framework. The remaining cases reflect those which would not have been checked by a supervisor in the current EPPM system implementation.

The above table shows that approximately 125 (out of a total of 1615) work-items were executed in times outside the 95% confidence interval established by the time distributions for each work item, which would otherwise have not been detected by an auditor. While this does not directly indicate that these cases were non-compliant, it does provide a significantly smaller subset for closer inspection based on statistics of standard reference ranges, than searching the entire list of workflow instances. As the reference range is not one-sided, cases wherein the work-item has been completed in a relatively short duration can also be brought under scrutiny. The table below shows the number of cases that were completed before the 2.5th confidence interval of the established time distributions.

Work Item	Total number of cases executed under the 2.5th established confidence time	Percentage
Verify Details	5	1.27%
Verify Participants	5	1.06%
Respond	4	1.03%
Review and Consolidate	2	1.64%
Approve	3	1.26%

Table 4.9: Testing Work-Item Samples with Durations Under 2.5th Percentile

Both the above tables are about specific work-item instances. Work-items however are merely constituents of workflows, and when an audit is conducted, it is performed on workflow instances so as to understand the context behind why a certain amount of time was utilized for component work-items. It is quite possible that multiple work-items appear within the same workflow, even for the case of work-items that belong to the same activity class. The 125 work-items that were outside the confidence interval and also shorter than 48 hour durations, belonged to 107 unique workflows. The testing sample set consisted of 341 workflow instances, meaning that the 125 work-items were distributed over 31.37% of the testing sample instances. Multiple work-item instances in this range appeared in 14 workflow instances, indicating that these workflow instances in particular seemed to execute abnormally. So depending on how an auditor wishes to analyze the data set, the number of workflow instances that may come under additional scrutiny can be varied.

4.3.9 Trend Workflow Case Study: Template and Compliant Paths

A second case study was undertaken to examine the applicability of this methodology to a different workflow template class at the same construction project in British Columbia. The trend workflow was designed to be an advanced change management workflow which incorporated specific additional features at the request of a client. This case study explored the potential of using an additional dimension to aid compliance checks while also noting the results of focusing solely on time signatures of work items.

Figure 4.17 on the next page, displays the Trend workflow schema. Textual information has been removed from the workflow schema as per the non-disclosure agreement between industry and this research initiative. However the potential execution path and important human tasks have been listed and described in the following pages.

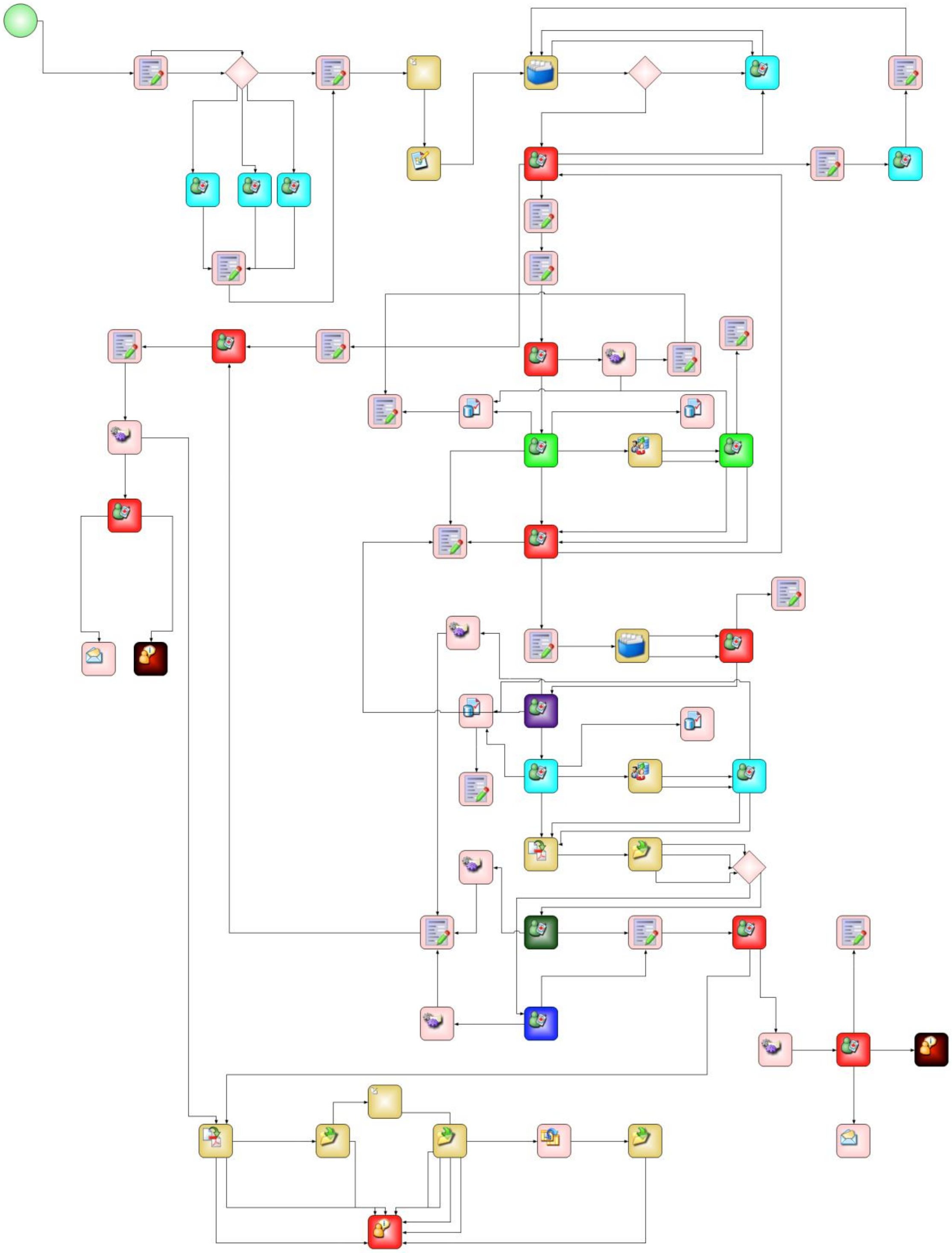
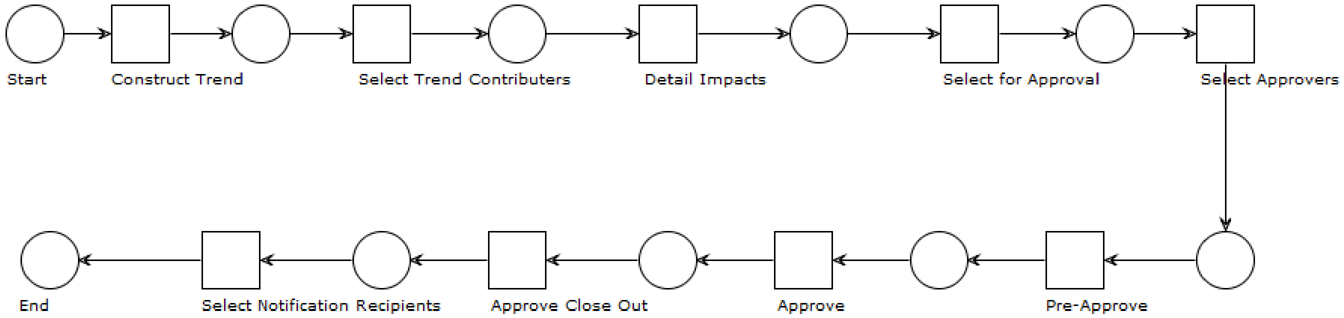


Figure 4.17: Trend Workflow Representation Used in Case Study (© Coreworx Inc.)

When the machine related and minor activities are removed, there are nine primary work-items that remain and which were analyzed. These activities are:

1) Construct Trend
2) Select Trend Contributors
3) Detail Impacts
4) Send for Approval
5) Select Approvers
6) Pre-Approve
7) Approve
8) Approve Close Out
9) Select Notification Recipients

The trend workflow exhibits a simple linear relationship between work-items as represented in the petri-net below:



Therefore there is only one compliant path that can exist, which is the sequence of work-items from Construct Trend to Select Notification Recipients.

4.3.10 Trend Workflow Data Cleaning and Analysis

The raw data consisted of 925 Trend workflow instances. These workflow instances contained a total of 30921 work items that involved a user. An extremely high number of these work items had to be rejected as the completion time was recorded as a 'NULL' value and therefore could not meaningfully be used for analysis. Of the remaining 17956 work-items, when warning and notification work-items – which do not constitute a task but rather an alert sent to a user that they have not completed a task within an established timeframe which was set at a default of 3 days – were removed, the number of work-items that remained was 15168.

These work-items were distributed over workflow instances, ranging from workflow id '1118' to workflow id '3108.' Similar to Section 4.3.4, a training sample set was selected for all workflows up to workflow id '2400.' This corresponded to 607 workflow instances, leaving a total of 318 for testing, which again approximately works out to two thirds for analysis and one third for testing. Table 4.1 on the next page shows the number of work-items that were considered for each of the stages as well as the percentages of testing and training instances against the entire set of instances. In general about 65% of the instances in the sample were used for training and around 35% for testing.

Again, similar to Section 4.3.4, work items times were corrected to reflect actual working hours based on two assumed work shifts, prior to data analysis, rather than reflect the times when a work item was issued to a user before being completed and sent on to the next user as recorded in the EPPM system. Data snapshots for each work item have been presented in Appendix A.

Work Item	Training Size	Testing Size	Total	Training Percentage	Testing Percentage
Construct Trend	1351	867	2218	60.91	39.09
Select Trend Contributors	1227	723	1950	62.92	37.08
Detail Impacts	2515	1229	3744	67.17	32.83
Send For Approval	1253	727	1980	63.28	26.72
Select Approvers	551	304	855	64.44	35.56
Pre-Approve	1402	947	2349	59.68	40.32
Approve	445	226	671	66.32	33.68
Approve Close Out	417	206	623	66.93	33.07
Select Notification Recipients	534	244	778	68.64	31.36
Total	9695	5473	15168	63.91	36.09

Table 4.10: Separation of Trend Work Items for Training and Testing

Upon analyzing the data, it was immediately apparent that several of the data points were noisy and could not have accurately reflected the time taken to complete a task. For instance, in the workflow template design, if a user had not completed a task within 3 days of it being issued, an alert notification was sent requiring immediate action. Again, an assumption was made that most tasks would be completed within 3 working days, or given two 8 hour shifts, a total of 48 hours, with some leeway to account for finishing a task after an alert notification was sent out. This 3 day time out period was coded into the Trend system as well, and often such time periods are indicative of contractual obligations between construction parties. Histograms for each work item were mapped as in the RFI case study, and are displayed in the Figures 4.18 to 4.26 below:

4.3.10.1.1 Construct Trend

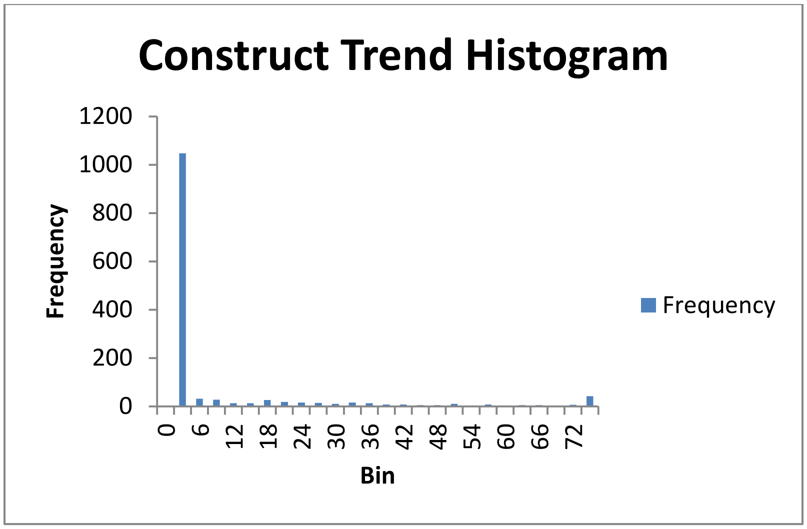


Figure 4.18: Histogram of Hours vs Instances for Work-item Construct Trend

4.3.10.1.2 Select Trend Contributors

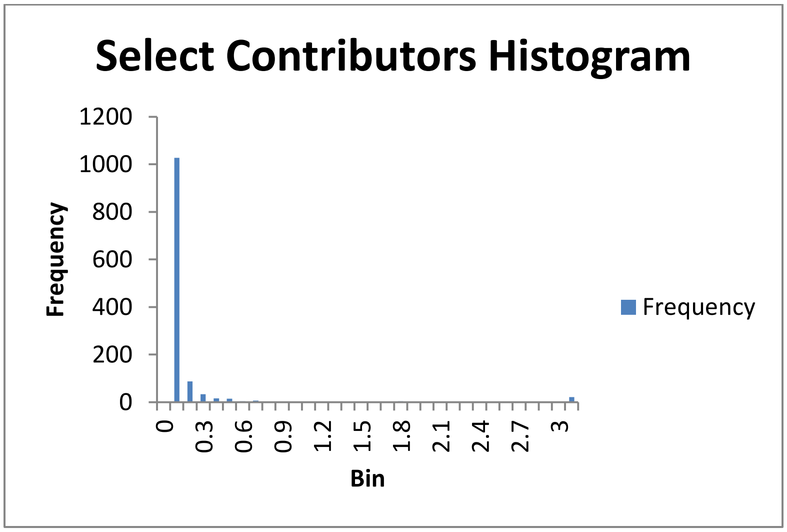


Figure 4.19: Histogram of Hours vs Instances for Work-item Select Contributors

4.3.10.1.3 Detail Impacts

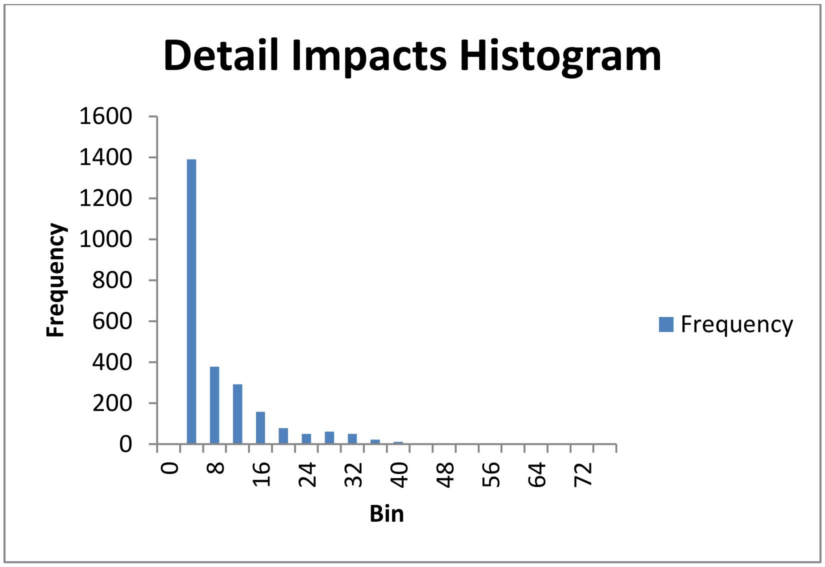


Figure 4.20: Histogram of Hours vs Instances for Work-item Detail Impacts

4.3.10.1.4 Send for Approval

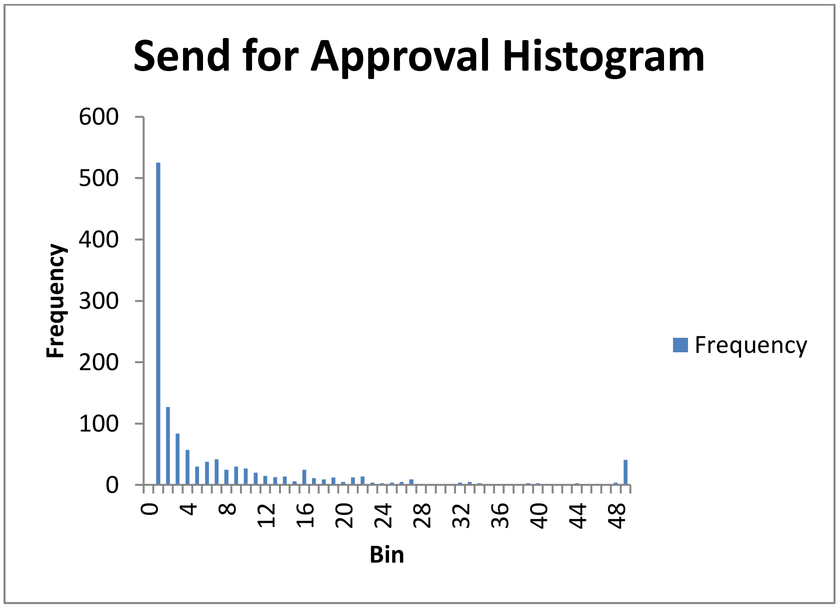


Figure 4.21: Histogram of Hours vs Instances for Work-item Send for Approval

4.3.10.1.5 Select Approvers

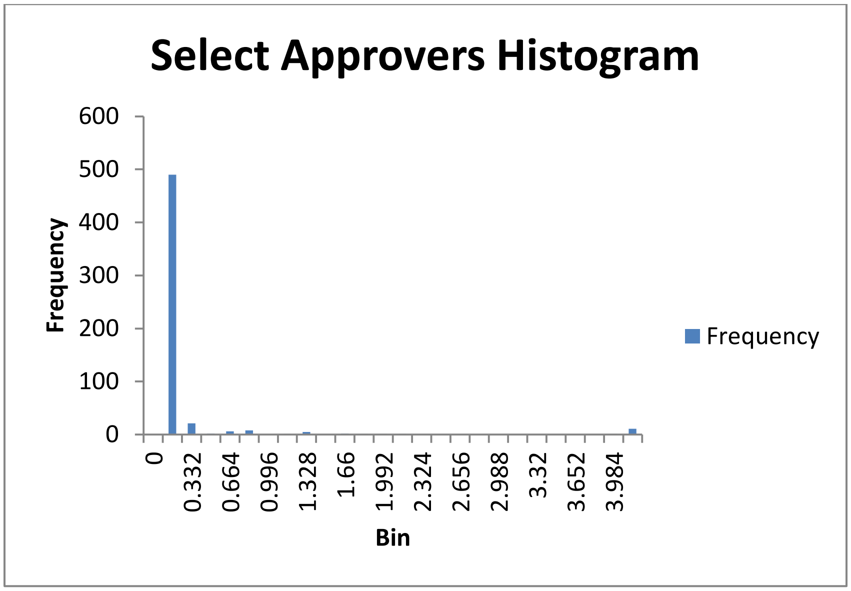


Figure 4.22: Histogram of Hours vs Instances for Work-item Select Approvers

4.3.10.1.6 Pre-Approve

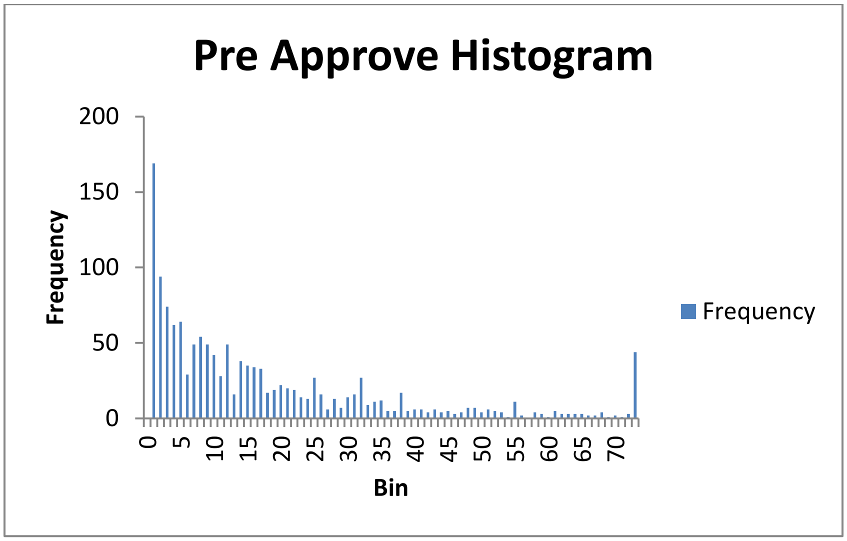


Figure 4.23: Histogram of Hours vs Instances for Work-item Pre-Approve

4.3.10.1.7 Approve

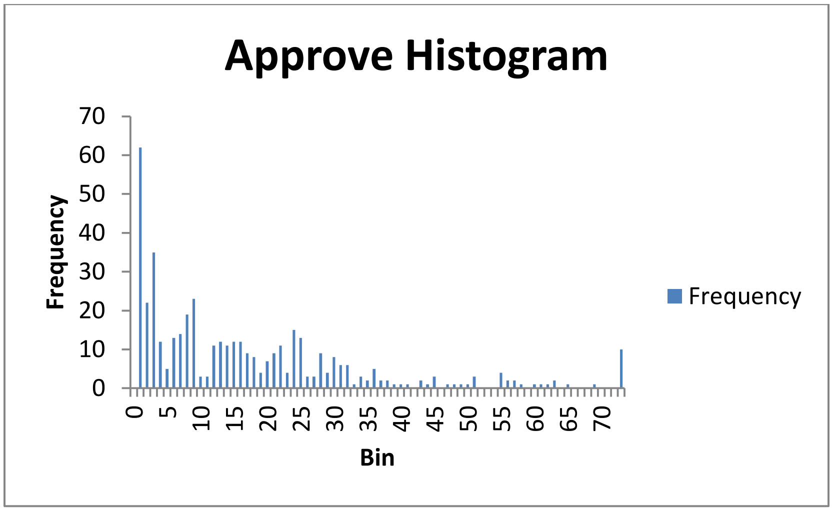


Figure 4.24: Histogram of Hours vs Instances for Work-item Approve

4.3.10.1.8 Approve Close Out

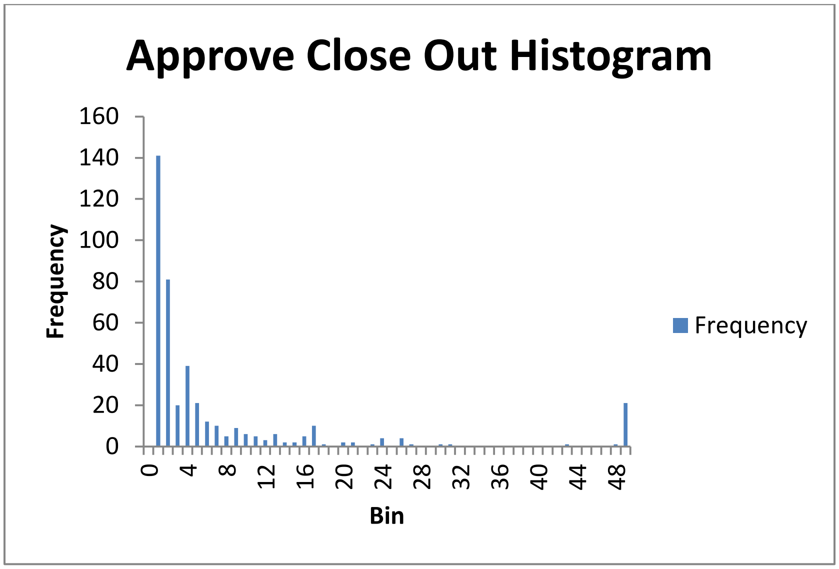


Figure 4.25: Histogram of Hours vs Instances for Work-item Approve Close Out

4.3.10.1.9 Select Notification Recipients

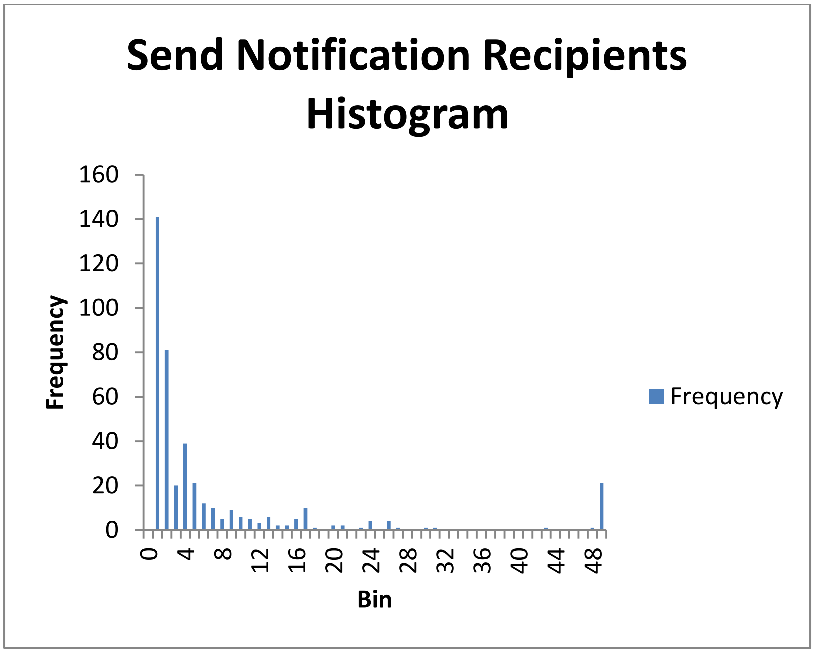


Figure 4.26: Histogram of Hours vs Instances for Work-item Send Notification Recipients

Again, as in section 4.3.4, all values that corresponded to outliers were removed from further analysis and the cleaned data was then used to determine time distributions of each constituent work items.

4.3.11 Time Distributions for Trend Work Items

The cleaned training data sets were analyzed using the software package EasyFit©. The time distributions that fitted the data best as per the Anderson Darling Goodness of Fit test were accepted for each work-item. The Probability Density Functions as well as the goodness of fit test results are displayed for each work item in the following pages. As the Trend workflow instances consisted of more work items, there was a lot more data available for analysis. This revealed some interesting phenomena that had been suspected previously, such as the presence of different classes of responses yielding multimodal distributions.

4.3.11.1.1 Construct Trend

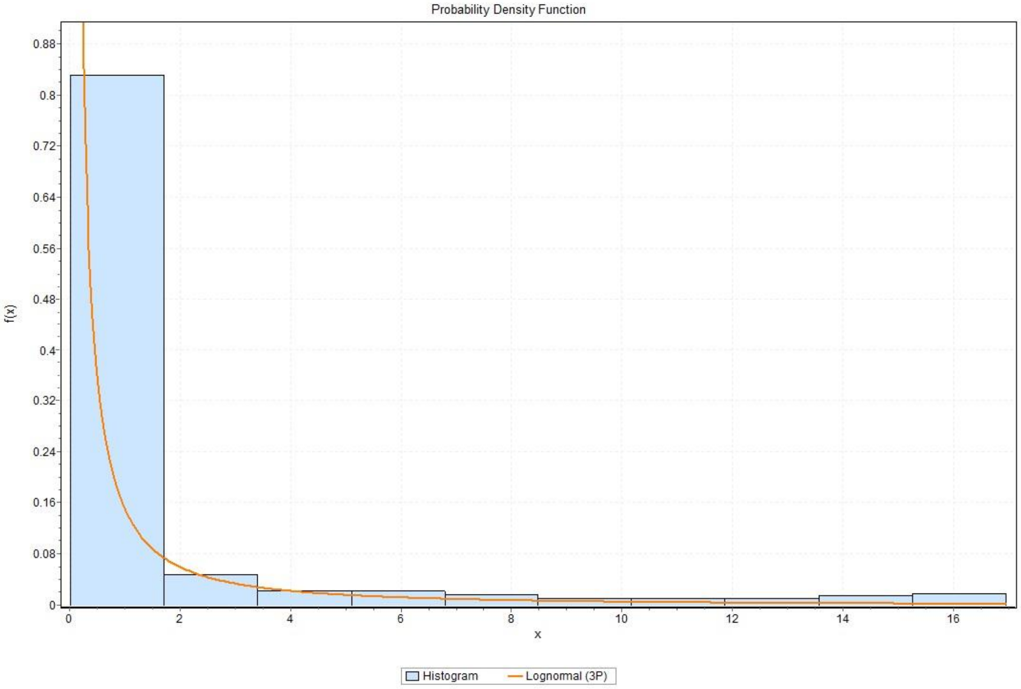


Figure 4.27: Probability Density Function For Construct Trend Lognormal Distribution

Lognormal Distribution Parameters	σ = 3.0756		μ = -2.7334	ϒ = 0.0084	
Anderson-Darling Goodness of Fit Results					
Sample Size	1226				
Statistic	2.3977				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	Yes	Yes	No	No	No

Table 4.11: Lognormal Distribution for Construct Trend: Parameters and Goodness of Fit Results

4.3.11.1.2 Select Trend Contributor

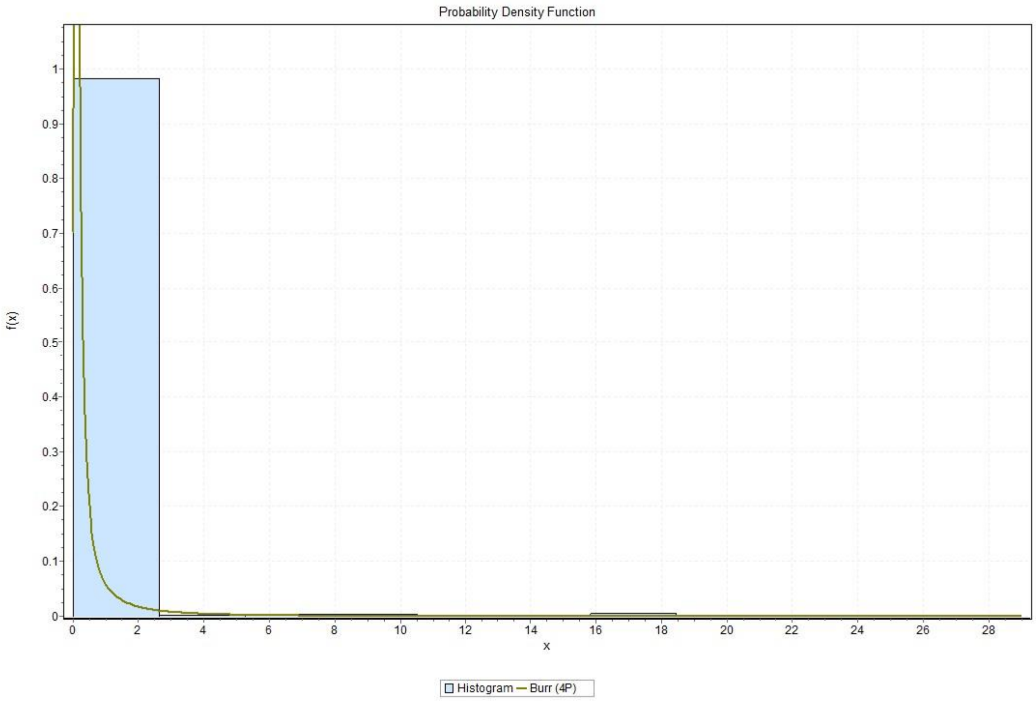


Figure 4.28: Probability Density Function For Select Contributors Burr Distribution

Burr Distribution Parameters	k = 0.0574	α = 13.581	β = 0.01049	ϒ = -0.0033	
Anderson-Darling Goodness of Fit Results					
Sample Size	1226				
Statistic	2.3977				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	Yes	Yes	No	No	No

Table 4.12: Burr Distribution for Select Contributors: Parameters and Goodness of Fit Results

4.3.11.1.3 Detail Impacts

The detail impacts work item was interesting because it exhibited characteristics of a multimodal distribution. It appeared as though there were two distinct peaks between durations extending from 0-5 hours and 5 to 48 hours, although the second peak appeared to be significantly lower than the first peak. Figure 4.29 below, illustrates the multimodal nature of the graph of work item instances v time:

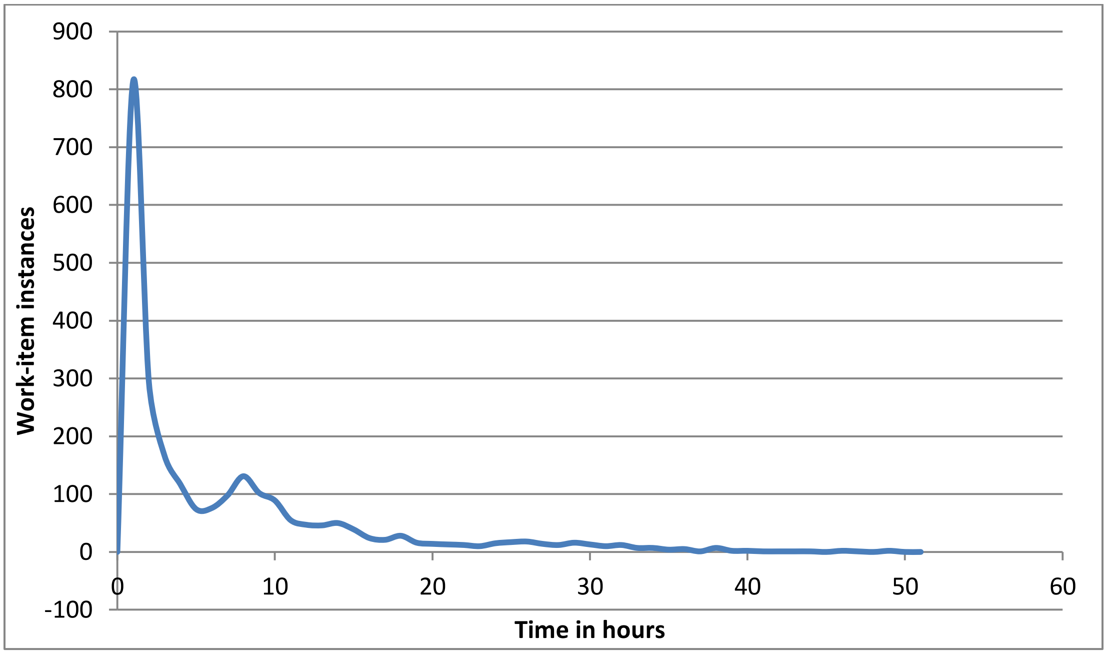


Figure 4.29: Multimodal nature of Detail Impacts Work item v Hours Graph

What this could indicate is that different personnel or perhaps different cases or in some cases both, require different levels of response. Although the task is defined the same, the amount of work required for it varies on a case by case basis, and we may begin to observe classes of responses for individual work items. Consider for example a simple task such as responding to an email. If the answer is known an immediate response may be delivered, however if the information is required from a colleague, we can expect the answer to be delivered a little later. An effort was made to identify two distributions with regard to the durations noted above and two distributions were obtained by separating the time intervals into two classes: (i) 0 to 5 hours and (ii) more than 5 hours

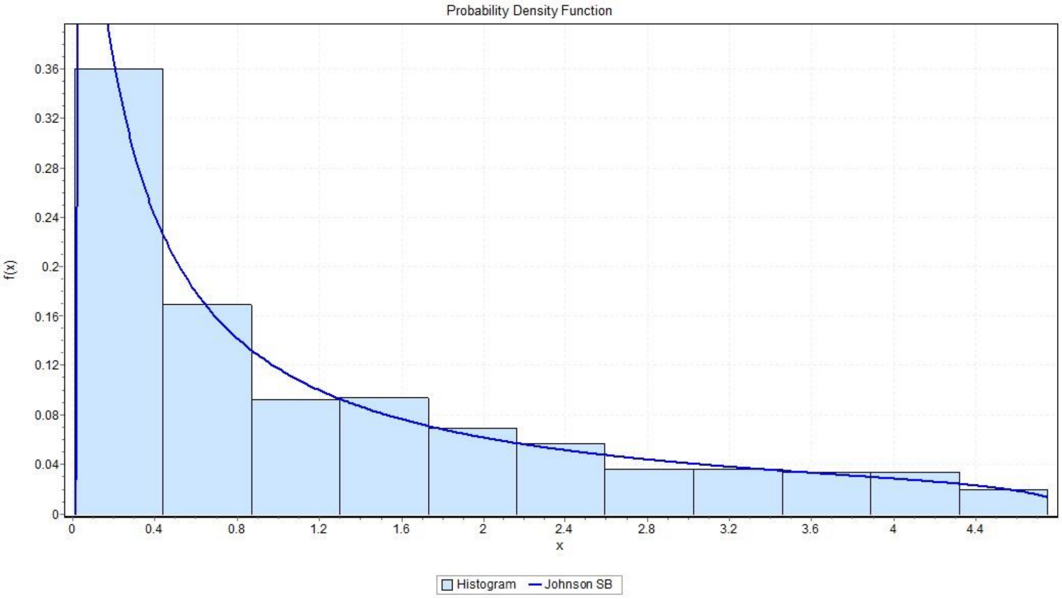


Figure 4.30 Probability Density Function for first 5 hours of Detail Impacts Johnson Distribution

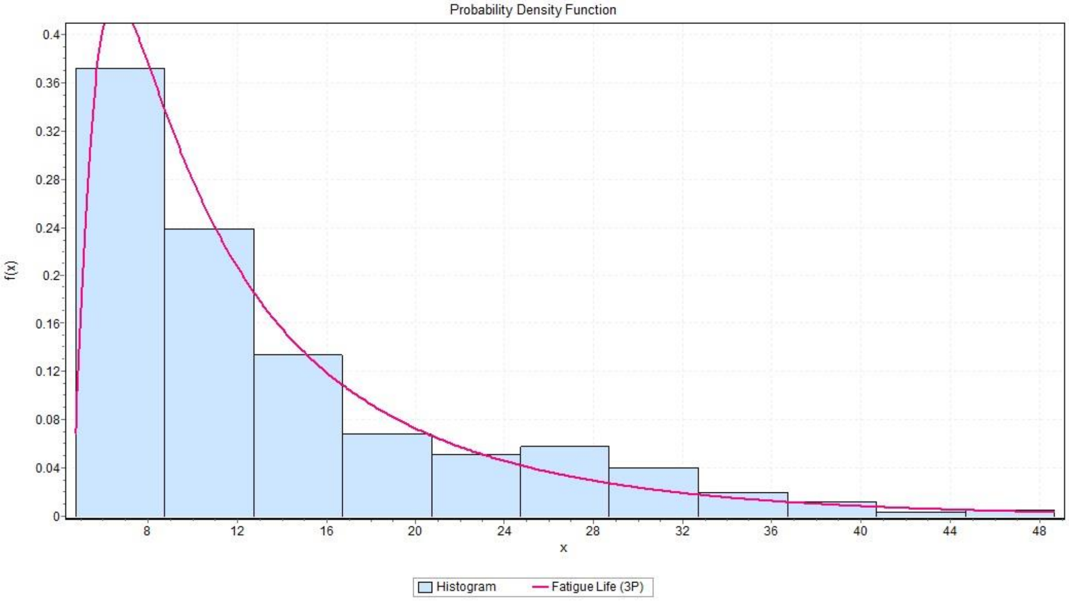


Figure 4.31 Probability Density Function for durations over 5 hours of Detail Impact Fatigue Life Distribution

It should be noted that the method of separation is suspect. It is not possible to ascertain with any certainty which instances between durations of 3 to 5 hours belong to a specific class of response. In other words, we may never be able to ascribe where one distribution ends and the other begins. Perhaps this separation is inconsequential in the context of identifying confidence intervals, since it becomes increasingly difficult to establish reasonable confidence limits based on two separate distributions for the same set of data. Analyzing a multimodal distribution is however, also extremely complicated.

Considering that one of the long term objectives of this research is to augment the capabilities of an EPPM system, and the paucity of software packages capable of identifying and analyzing multimodal distributions, that may be integrated into such a system, it is perhaps better to opt for simplicity and select the closest fitting uni-modal distribution in such circumstances. However this is certainly an area of interest which may be pursued in future research initiatives.

Considering that the second peak is significantly lower than the first peak (in figure 4.29) – which allows for an approximate distribution considering all durations – and accepting that while the phenomenon is interesting, the additional complexity involved in determining multimodal distributions makes adopting them for this research initiative, and perhaps in general in practice, unfeasible, a uni-modal distribution that best fitted the data was adopted for this research, as depicted in Figure 4.32 on the next page.

The uni-modal distribution from the EasyFit software package that best fitted the distribution was the Kumaraswamy distribution. However, as is probably apparent in Figure 4.32 above, the fit was not perfect and this was reflected in the goodness of fit results for the Anderson Darlington test in table 4.13 below. This limitation should be taken into consideration when obtaining and utilizing the confidence limits for auditing purposes.

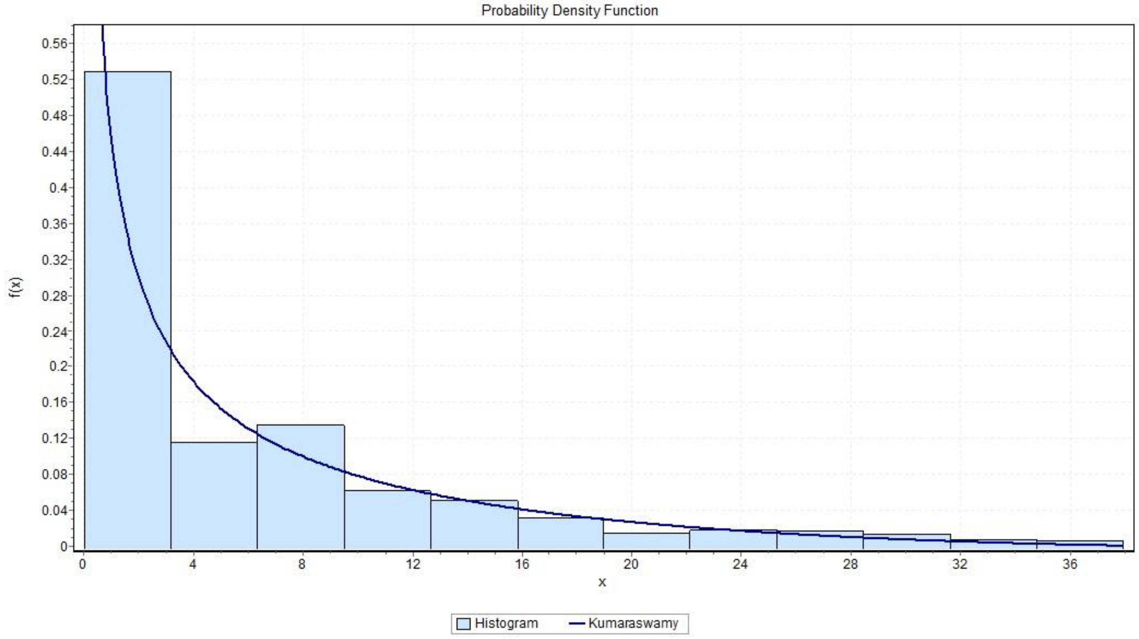


Figure 4.32: Probability Density Function For Detail Impacts Approximated Kumaraswamy Distribution

.

Kumaraswamy Distribution Parameters	α1 = 0.6173	α2 = 5.1078	a = 0.00758	b = 89.329	
Anderson-Darling Goodness of Fit Results					
Sample Size	2507				
Statistic	12.217				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	Yes	Yes	Yes	Yes	Yes

Table 4.13: Kumaraswamy Distribution for Detail Impacts: Parameters and Goodness of Fit Results

4.3.11.1.4 Send For Approval

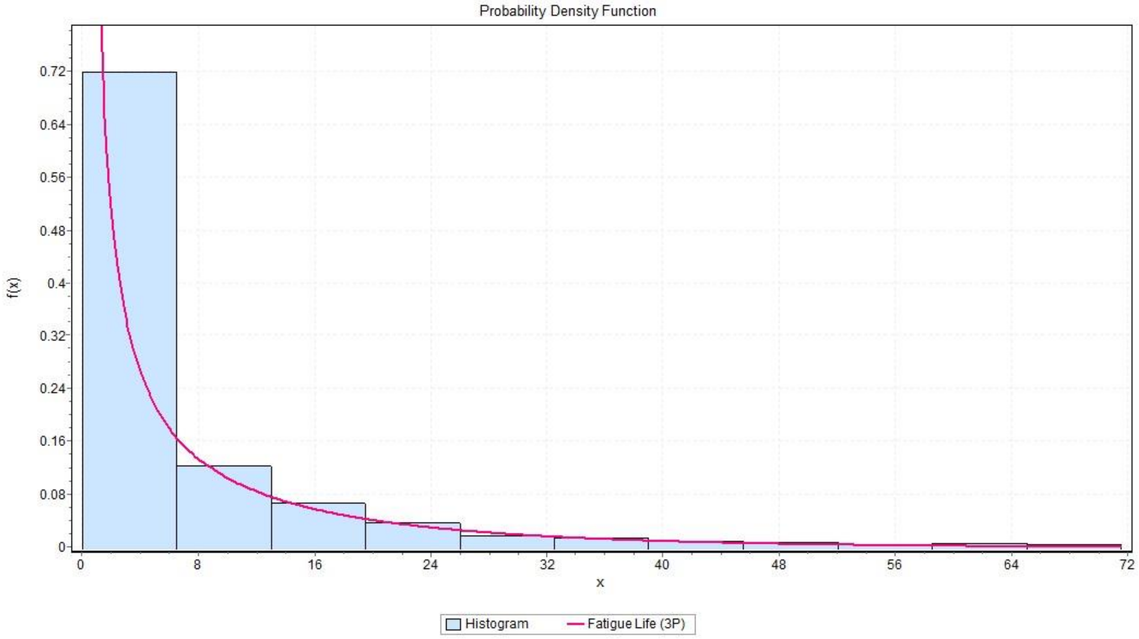


Figure 4.33: Probability Density Function for Send for Approval Fatigue Life Distribution

Fatigue Life Distribution Parameters	α = 2.4925		β = 1.5447		ϒ = -0.0203
Anderson-Darling Goodness of Fit Results					
Sample Size	1228				
Statistic	0.02841				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.14: Fatigue Life Distribution for Send for Approval: Parameters and Goodness of Fit Results

4.3.11.1.5 Select Approvers

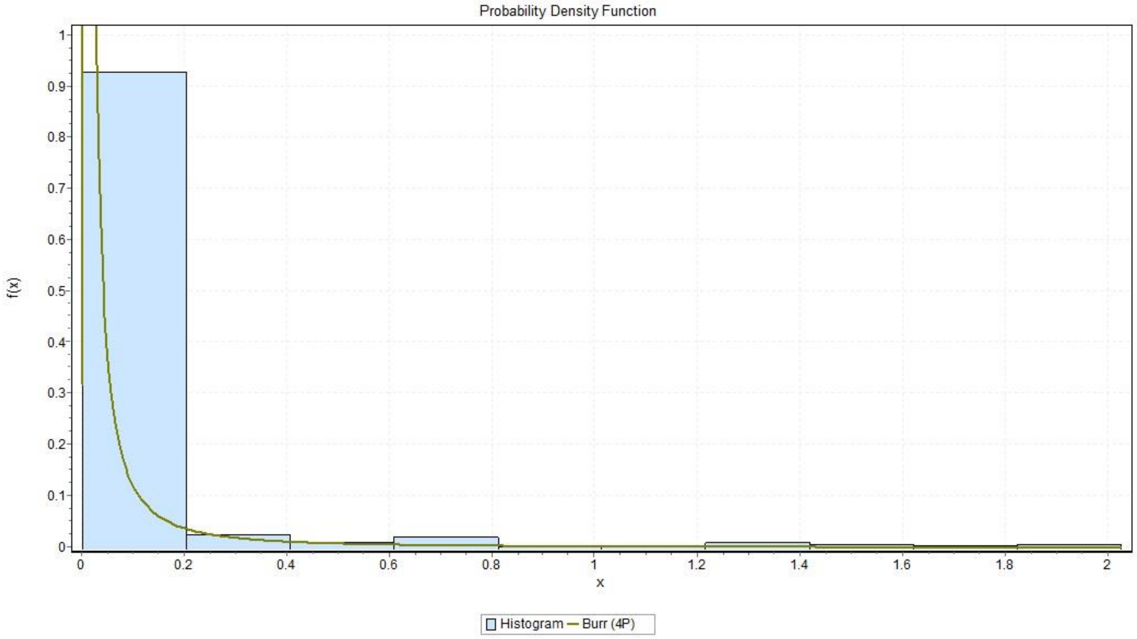


Figure 4.34: Probability Density Function for Select Approvers Burr Distribution

Burr Distribution Parameters	k = 0.1793	α = 3.7441	β = 0.0026	ϒ = 0.0015	
Anderson-Darling Goodness of Fit Results					
Sample Size	538				
Statistic	0.60333				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.15: Burr Distribution for Select Approvers: Parameters and Goodness of Fit Results

4.3.11.1.6 Pre-Approve

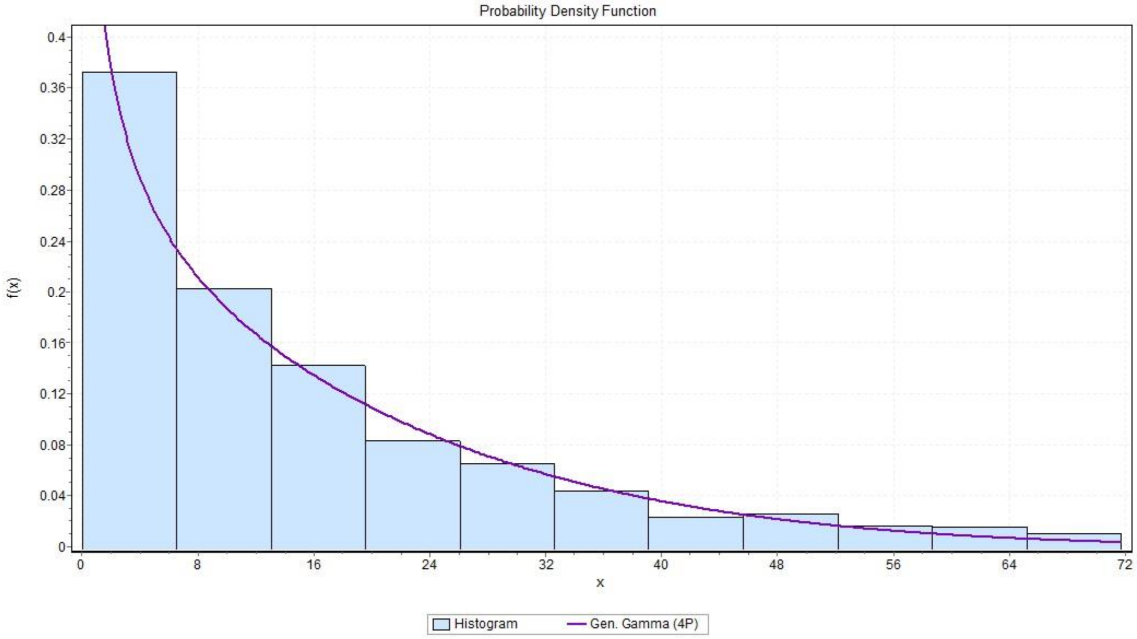


Figure 4.35: Probability Density Function for Pre Approve General Gamma 4 Parameter Distribution

Generalized Gamma 4 Parameter Distribution	k = 1.535	α = 0.4267	β = 33.321	ϒ = 0.00654	
Anderson-Darling Goodness of Fit Results					
Sample Size	1358				
Statistic	1.2006				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.16: Gen. Gamma Distribution for Pre-Approve: Parameters and Goodness of Fit Results

4.3.11.1.7 Approve

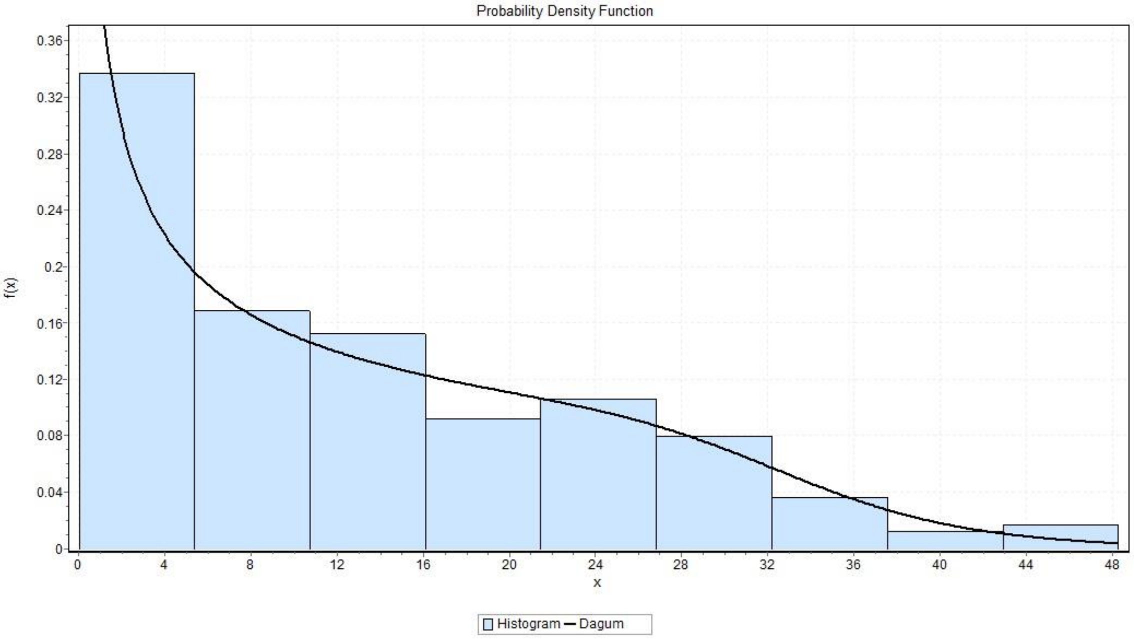


Figure 4.36: Probability Density Function for Approve Dagum Distribution

Dagum Distribution Parameters	k = 0.0855	α = 6.3654	β = 34.161	ϒ = 0.0185	
Anderson-Darling Goodness of Fit Results					
Sample Size	415				
Statistic	0.55931				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.17: Dagum Distribution for Trend Approve: Parameters and Goodness of Fit Results

4.3.11.1.8 Approve Close Out

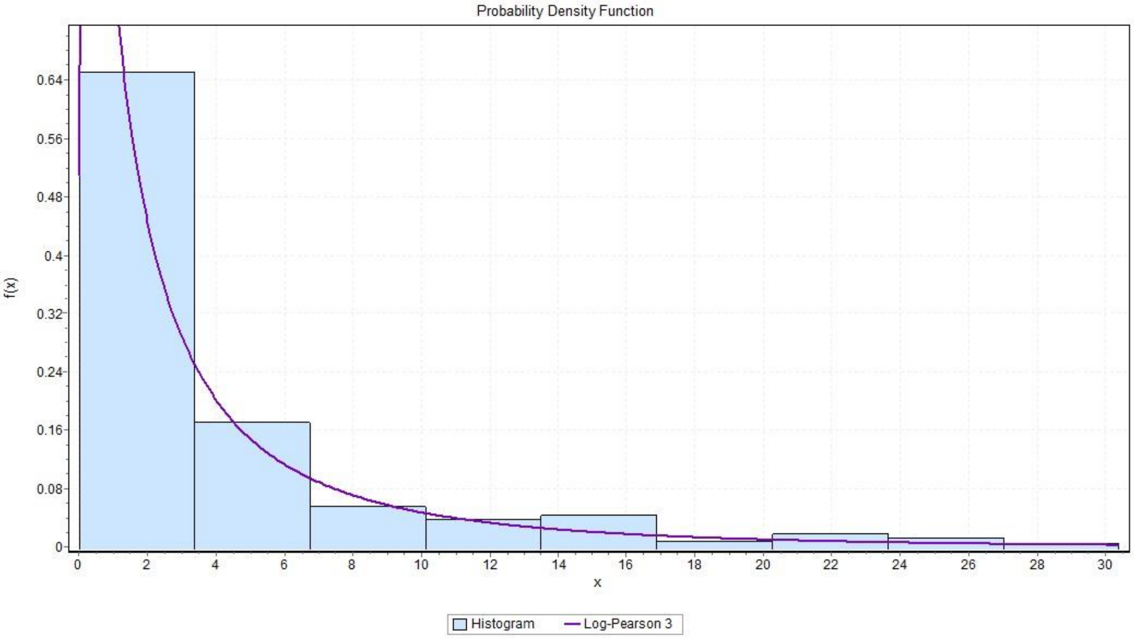


Figure 4.37: Probability Density Function for Approve Close Out Log Pearson Distribution

Log-Pearson Distribution Parameters	α = 29.929		β = -0.2803		ϒ = 8.8482
Anderson-Darling Goodness of Fit Results					
Sample Size	394				
Statistic	0.71445				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.18: Log Pearson Distribution for Approve Close Out: Parameters and Goodness of Fit Results

4.3.11.1.9 Select Notification Recipients

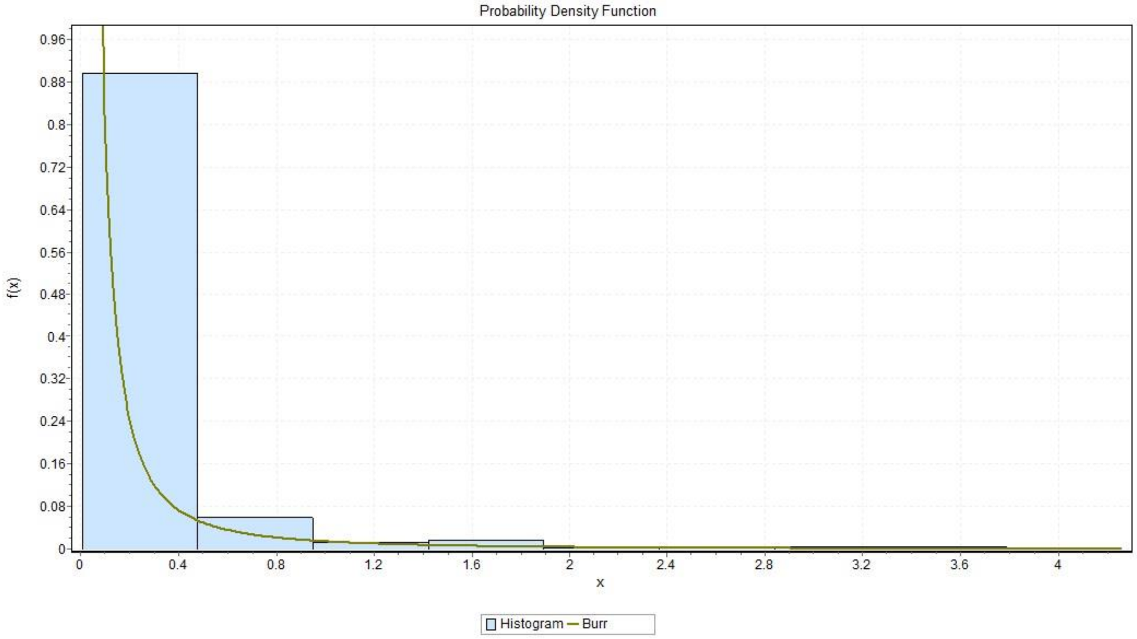


Figure 4.38: Probability Density Function for Send Notification Recipients Burr Distribution

Burr Distribution Parameters	K = 0.221	α = 3.3181		β = 0.0138	
Anderson-Darling Goodness of Fit Results					
Sample Size	498				
Statistic	1.1155				
Rank	1				
α	0.2	0.1	0.05	0.02	0.01
Critical Value	1.3749	1.9286	2.5018	3.2892	3.9074
Reject?	No	No	No	No	No

Table 4.19: Burr Distribution for Send Notification Recipients: Parameters and Goodness of Fit Results

4.3.11.2 Consolidated Trend Work Item Time Distribution Confidence Intervals

Task	Time Distribution	2.5th Percentile time (Hours)	97.5th Percentile time (Hours)
Construct Trend	Lognormal	0.0086	26.978
Select Contributors	Burr (4 Parameter)	0.0068	1.1928
Detail Impacts	Kumaraswamy	0.024	30.428
Send for Approval	Fatigue Life	0.0394	39.873
Select Approvers	Burr	0.0031	0.6448
Pre Approve	General Gamma 4 parameter	0.1058	57.408
Approve	Dagum	0.0575	40.404
Approve Close Out	Log Pearson	0.0606	24.3
Send for Notification	Burr	0.0073	2.1009

Table 4.20: 2.5th and 97.5th confidence limits for work item time distribution

Table 4.20 above summarizes the derived distributions for each work item in the workflow as well as the 95% confidence intervals of times for which these tasks were executed.

4.3.12 Integrated BPEL Representation of Trend Workflow Instance

The following is the BPEL generated code from the WoPeD application for the workflow implementation that was considered for this analysis, with the integrated confidence limits:

```
<bpel:process xmlns:bpel="http://docs.oasis-
open.org/wsbpel/2.0/process/executable">
  <bpel:sequence>
    <bpel:empty name="Construct Trend" upper_conf="0.0086" lower_conf="26.978"/>
    <bpel:empty name="Select Trend Contributers" upper_conf="0.0068"
lower_conf="1.1928"/>
    <bpel:empty name="Detail Impacts" upper_conf="0.024" lower_conf="30.428"/>
    <bpel:empty name="Select for Approval" upper_conf="0.0394"
lower_conf="39.873"/>
    <bpel:empty name="Select Approvers" upper_conf="0.0031" lower_conf="0.6448"/>
    <bpel:empty name="Pre-Approve" upper_conf="0.1058" lower_conf="57.408"/>
    <bpel:empty name="Approve" upper_conf="0.0575" lower_conf="40.404"/>
    <bpel:empty name="Approve Close Out" upper_conf="0.0606" lower_conf="24.3"/>
    <bpel:empty name="Select Notification Recipients" upper_conf="0.0073"
lower_conf="2.1009"/>
  </bpel:sequence>
</bpel:process>
```

This BPEL output for the Trend workflow implementation may now be used for model checking purposes, employing complex models such as petri nets where required, or simple parsing comparative algorithms to determine whether executing workflow instances contain work items that lie outside of the established confidence intervals. A simple comparison was conducted with the testing sample from the case study and the results are presented in the next section.

4.3.13 Summary and Interpretation of Trend Workflow Case Study

Table 4.21 on the following page summarizes the resulting number and percentage of testing work item instances that did not fall within the confidence interval of time distributions for each class of work-item during the comparison between the testing samples and the integrated model. Similar to section 4.3.8, all cases that fall out of the 95% confidence interval have been categorized into two types: (i) The total number of cases that fell outside of the range and (ii) all cases wherein the work-item time was over 3 working days, assuming two 8-hour working shifts in a day. In analyzing the results and comparing them to the RFI workflow case study, it should be noted that there were significantly higher testing sample instances for the Trends workflow as compared to the RFI workflow.

Work-Item	Testing Sample Size	Cases Outside Confidence Interval Times	Cases over 3 working days (48 hours)	Percentage requiring additional check	Additional Percentage determined by framework
Construct Trend	867	168	32	19.38%	15.69%
Select Trend Contributors	723	14	1	1.94%	1.79%
Detail Impacts	1229	24	3	1.95%	1.71%
Send For Approval	727	30	9	4.13%	2.89%
Select Approvers	304	29	0	9.54%	9.54%
Pre-Approve	947	116	32*	12.25%	9.50%**
Approve	226	40	13	17.70%	11.95%
Approve Close Out	206	25	2	12.14%	11.17%
Select Notification Recipients	244	13	0	5.33%	5.33%

Table 4.21 Percentage of compliant instances for each trend work item

The above table shows that approximately 367 (out of a total of 5473) work-items were executed in times outside the 95% confidence interval established by the time distributions for each work item, which would otherwise have not been detected by an auditor. A significant proportion of these cases belonged to the construct trend work item, wherein the workflow instance was created. This indicates that a large number of workflow instances may not have been defined very well. However some leeway may be attributed to the fact that the trend workflow was a completely new class of workflows and therefore training time for users could have accounted for a few of these cases. Another interesting work item was the pre-approve work item, wherein the 97.5th percentile was greater than the 48 hour time period assumed for an automatic notification, at 57 hours and 40 minutes. The high number of cases violating the period were primarily cases that were conducted in durations shorter than the 2.5th percentile.

As in Section 4.3.8, cases wherein the work-item has been completed in a relatively short duration were also be brought under scrutiny. Table 4.22 below, shows the number of cases that were completed before the 2.5th confidence interval of the established time distributions.

Work Item	Total number of cases executed under the 2.5th established confidence time	Percentage
Construct Trend	113	13.03%
Select Contributors	11	1.52%
Detail Impacts	5	0.41%
Send For Approval	18	2.48%
Select Approvers	2	0.66%
Pre-Approve	90	9.50%
Approve	4	1.77%
Approve Close Out	13	6.31%
Select Notification Recipients	4	1.64%

Table 4.22: Testing Work-Item Samples with Durations Under 2.5th Percentile

An audit would typically be performed on workflow instances, so as to understand the context behind why a certain amount of time was utilized for component work-items. Similar to section 4.3.8, multiple work-items could appear within the same workflow. The 367 work-items that were outside the confidence interval and also shorter than 48 hour durations, belonged to 189 unique workflows. The testing sample set consisted of 318 workflow instances, meaning that the 367 work-items were distributed over an astounding 59.43% of the testing sample instances.

Even allowing for training cases and poor definitions of the workflow instances because of the associated novelty of this new class of workflows, when the “construct trend” work item is disregarded (so as to enable a fair comparison with the RFI workflow case study where there was insufficient data for the “RFI draft initialization” work item), 153 workflows, or 48% of the testing sample would have been highlighted for a closer look by an auditor. In other words, the compliance checking methodology that is solely based upon time distributions would have only removed half of the cases from the initial data set. This may be explained by the fact that there were some work items such as “pre-approve” and “construct trend” that heavily influenced the final results, and an auditor may choose to adjust his level by including or excluding these work items or conducting a deeper study into why these work items seemed to include a high number of cases on the tails of the distribution.

4.4 Discussion

Narrowing down the subset of workflows to be studied in an audit by 50% is no trivial achievement and may still provide significant time savings for an auditor. The approach however may be further refined based upon unique characteristics of a workflow class. The methodology applied has been restricted to one variable, the execution time of a work item. While the time dimension will exist for every work item, the feature rich nature of newer workflows that utilize the full potential of an EPPM system, allows for analysis along other dimensions. Compliance can then be measured on multiple criteria which will allow for deeper insight into the execution of tasks.

The Trend workflow implementation was developed primarily to associate budget impacts with change request workflows. This would allow for prioritization of change requests based on the impact the change request would have on the project's overall budget. This characteristic however provides an opportunity to associate budget impacts with time taken to complete a work item. Is there, for example, a relationship between the time spent on a work item and the associated budget impact for that workflow instance (budget impacts were applied to the change request workflow – and therefore each constituent work-item may be considered to have the same impact value)? Were so many "pre-approve" and "construct trend" executed so quickly because the budget impact associated with those workflows was low? Did it take longer to approve a change request because the budget impact was high and could affect the overall project? And if such relationships exist, would it not be possible then to isolate cases wherein tasks which had an associated smaller budget ended up taking far more time than expected or tasks which had high budget impacts were executed within short durations?

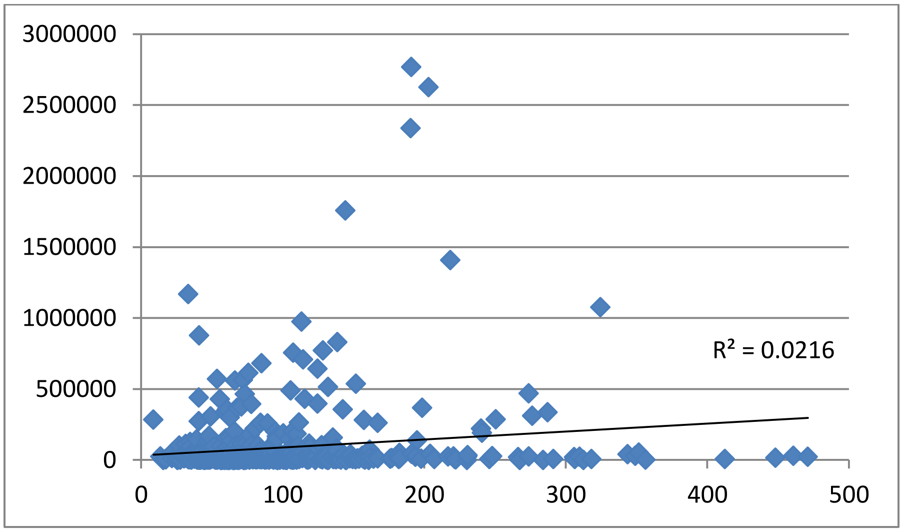


Figure 4.39: Time spent on a workflow instance versus cost impact of the workflow instance

To determine whether such a relationship did indeed exist, a second study was conducted on the Trend workflow data set wherein budget costs were compared to execution times for individual work items. No relation was found between budget impact and execution time for any of the nine Trend work items. Figure 4.39 on the preceding page depicts the graph between the overall execution time of the workflow instance and the cost impact of that instance for all 925 Trend workflow instances. As can be seen from the extremely low R^2 value and the general scatter of points, from this data set such a relationship did not exist. While the analyses result was disappointing in this case, incorporating as many dimensions as possible in trying to efficiently determine cases that could be non-compliant can provide richer analysis and deeper insight.

4.5 Conclusion

A framework was developed to establish methodology upon which executed workflow instances may be compared with an integrated time-distribution work-item based workflow implementation to efficiently determine a subset of workflow instances for compliance auditing purposes. A case study was conducted based upon data available from the first stage of a construction project in British Columbia. The results of this case study for RFIs indicate that the model would have reduced the set of workflow instances that would have required examination in the event of a compliance audit to 31% of a testing sample, the total work-item testing set containing on average 7.78% of instances which would not have been flagged under a current EPPM system. For the Trend workflow case study, the model would have reduced the sample set of workflows to be analyzed by an auditor by 40%, although the percentage of work items that would have required a closer examination would have been 6.7%. Finally it was concluded that compliance checks may also be made richer by analyzing along additional dimensions, when they are available.

Chapter 5

Automating Self-Adapting Construction Workflows

Companies typically expect a designed workflow template to operate silently and seamlessly in the background, automating structured information exchange from the start to the end of a project. In reality, most construction workflows are still of a semi-automated nature, and require a human coordinator to confirm participants and specific details for every workflow instance. Project managers noted that this often slowed down what should effectively have been the automation of a process that had been pre-defined for classes of activities. For example, one project manager commented on the pre-population of data entry fields and wondered if similar technology could be applied to workflows. It is also not unusual for repetitive sub-tasks that were included in the original template to be later deemed as unnecessary after a workflow has been deployed based upon the feedback of users. While it is natural for a workflow template to undergo changes after it has been deployed, such changes are often expensive both in terms of time and cost.

During interviews conducted to assess the functionality of workflows, business analysts and project managers ranked the automation of these adaptations for workflows as the most desirable change they would like to see in their EPPM systems. An adaptive framework that is able to recognize which actors have expertise in an area and also determine the relative availability of resources can greatly improve the performance of a workflow by reducing the semi-automated nature of the workflow.

Knowledge from executed workflow history logs may be exploited to identify such changes early and develop workflow implementations without costly manual supervision and intervention. Self-adapting workflows could greatly reduce the costs and delays involved for every customization of a workflow template that an off-site vendor would have to make. Ideally such adaptations would occur even before users of the system have identified that the workflow template requires a change. There are different types of modifications that may be incorporated into a workflow affecting either the structure of the workflow or the actors or both. A generalized framework has been developed which outlines how self-adapting construction workflows may be created. Specific cases wherein adaptation may be employed as well as the algorithms that have been developed for each case have been described in the sections below.

5.1 Methodology For Enabling Self Adaptation of Construction Workflows

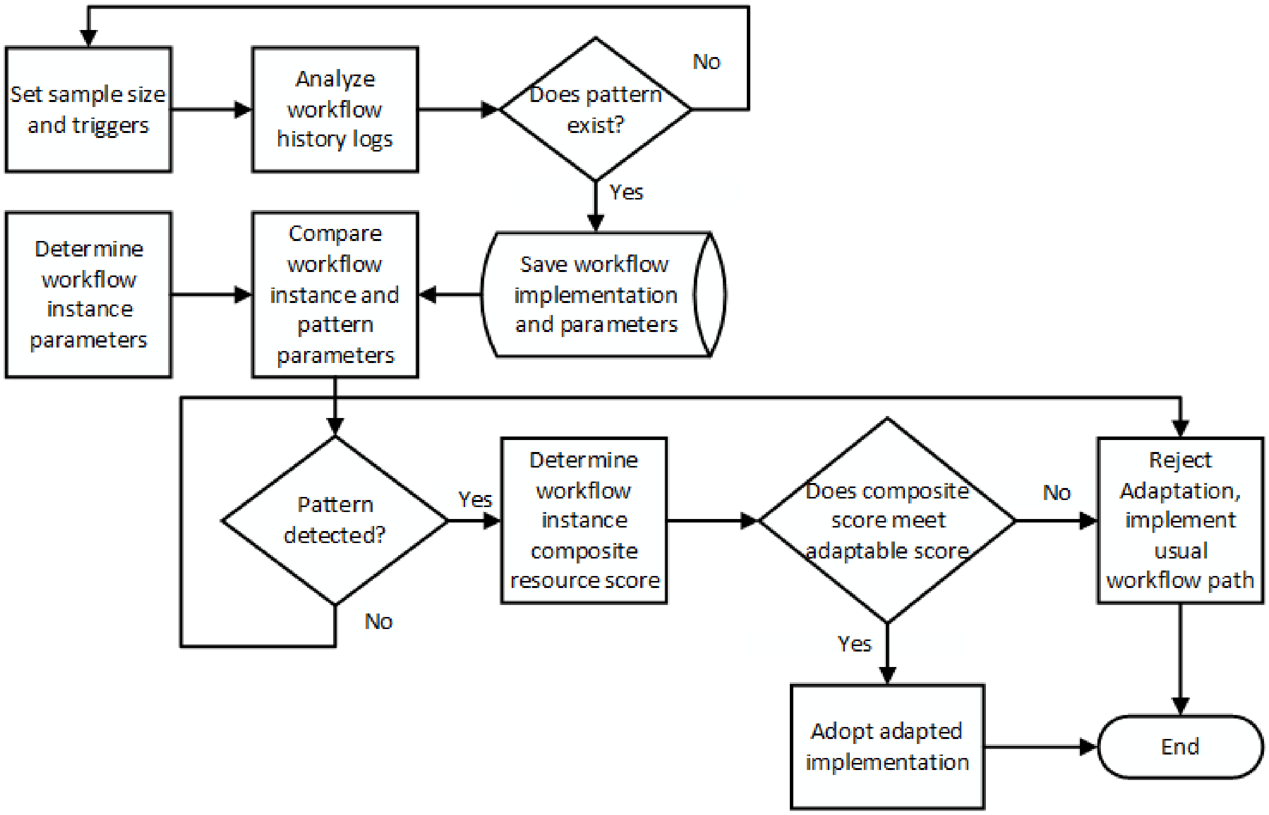

Set sample size and triggers
Analyze workflow history logs
Does pattern exist?
No
Yes
Save workflow implementation and parameters
Compare workflow instance and pattern parameters
Determine workflow instance parameters
Pattern detected?
Yes
No
Determine workflow instance composite resource score
Does composite score meet adaptable score
No
Yes
Reject Adaptation, implement usual workflow path
Adopt adapted implementation
End

Figure 5.1: Methodology for self-adaptation of workflows

Figure 5.1 above, demonstrates the generalized methodology developed for enabling self-adapting workflows. The process is divided into two stages, the first of which is an iterative analysis stage, while the second consists of a real-time continuous monitoring phase. In the first stage, analysis is conducted on a training sample size of data to determine adaptation parameters. In the second stage, a workflow instance is analyzed in real-time to determine if it meets the conditions to undergo adaptation.

5.1.1 Generalized Adaptive Framework Methodology Algorithm

Intelligent self-adaptation of workflows is possible based on exploiting knowledge from history logs of executed instances. When a pattern has been identified as recurring on a frequent basis, future workflow instances may adapt to a new implementation provided they satisfy criteria that permits such an adaptation. One of the assumptions in this research initiative is that user feedback is inferred from workflow logs rather than from surveys so as to remove elements of subjectivity. The following algorithm was designed for the self-adaptive framework. Steps 1 to 4 represent the analysis stage while steps 5 to 7 represent the implementation stage:

1. Set initial sample size and pattern identification triggers
2. Analyze workflow history logs to determine if pattern exists
3. If pattern exists, save workflow implementation as self-adaptable workflow implementation
4. Determine attributes for triggering adaptation associated to pattern and save
5. Compare current workflow instance's parameters with list of saved adaptable patterns
6. If parameters match, determine current workflow's composite score
7. If composite score is acceptable, modify current workflow instance to adaptive implementation

It is important to establish two factors before searching for patterns:

1. What number of instances constitutes a representative sample for future adaptations?

2. What thresholds of recurrence trigger the creation of an implementation?

5.1.2 Representative Sample Sets

There is no real way to accurately predict the amount of information exchange that can occur for classes of communication such as Requests for Information or Change Requests over the course of a construction project for several reasons. Construction projects are unique and vary greatly from one to another in terms of scope, length, parties involved, policies established etc., and elements outside reasonable control can further add to the unpredictability of estimating information transactions. Further, for proprietary reasons most companies do not disclose such information publicly, and there are no data sources available for the industry as a whole to establish an average number of transactions from which a suitable number of instances may be considered to be a representative sample. This is further complicated when new communication protocols, such as interface management are established.

Until such data sources become available, it is recommended that the firm implementing the workflow provides an estimate or defines an initial number of instances they are willing to consider as a suitable training sample. After the initial sample set has been arrived at for an initial analysis, analysis can occur either on a continuous basis at the completion of every workflow instance – a computationally expensive but more accurate method – or after predefined incremental completions.

It should be noted that the sample set for analysis may not constitute a fixed number of completed workflow instances. Rather the sample set is representative of a number of completed instances that fit a specific identified pattern. For example, say it has been established that after 200 RFIs have satisfactorily completed the initial analysis that a pattern has been noted for 8 instances. Establishing a permanent pattern based upon 4% of completed instances of a sample set might be premature, and therefore it becomes important to establish a sample set size for the number of workflow instances analyzed in the identification of a pattern.

5.1.3 Establishing Thresholds

Thresholds that may be used for triggering the implementation of a pattern depend to some degree on the sample size. However, unlike the indeterminacy that plagues selecting suitably sized samples, thresholds can be selected based upon user intuition and the level of accuracy demanded. This flexibility provides users with a powerful option of deciding exactly when the frequency of occurrence demands that a pattern may be incorporated into the creation of a new workflow implementation.

There are several different types of thresholds that may be selected depending on the pattern that is likely to emerge. In considering the types of thresholds it is important to note that workflows primarily consist of three dimensions as defined by van der Aalst (1998); the case dimension, the resource dimension and the process dimension. The case dimension refers to specific instances, the resource dimension refers to the actors interacting with the workflow while the process dimension refers to the work-items and routing constructs between them. Emergent patterns could be for a specific dimension or a combination of dimensions.

Threshold triggers for the pattern might be of a temporal nature, i.e. several similar actions occurring within a short span of time, or of a proportional nature, i.e. when the frequency of incidents exceeds a factor. The following are examples of thresholds that may be defined for an EPPM system:

1) For cases which begin with exactly the same set of users, is the frequency of only one subset of users who actually completed the workflow instances greater than X%?
2) Is the percentage of clarification workflow routes initiated whenever user A completes an activity greater than X%?
3) For a set of consecutive workflows, have the time intervals for an activity been within the range Y%?

4) Has a user B completed X% of activities with a status Q within a range of time T?

5.1.4 Saving and Comparing Emergent Patterns and Parameters

The above examples demonstrate how the dimensions as well as different kinds of triggers can be used in various permutations. The full range of expected patterns that can be expected to develop depends on the workflow template; let this be denoted by the set S. It should be noted that while a few users openly embraced the concept of self-adapting workflows, a key consideration in employing workflows has always been to ensure a suitable level of compliance. For optimal operation of the self-adapting workflow framework, both from the points of view of computational expense as well as maintaining a desirable level of compliance, an EPPM system vendor would present a list of potential patterns to a client and a selected subset $P, where\ P \subseteq S$, would be saved as pattern templates associated to a workflow template.

During the analysis of the workflow instances in the sample set, if a particular pattern corresponds to one of the pattern templates of the set P, it is saved as a recurrent pattern object, and each of the attributes of the pattern shall be saved along with it thus making it a unique instance. For example, let us assume that a particular route being triggered by a user pairing as has been saved as one of the pattern templates. During analysis, if this route happens to be the clarification route, and the pairing is for users A and B, wherein B has worked on an activity named respond, then the pattern attributes may be stored as P1(trigger_activity='respond', resultant_route='clarification', user1='A', user2='B').

During the operation of future workflows wherein users A and B are selected, if user B happens to be selected in the pre-executed implementation setting as working on the activity "respond", then a comparison between the pattern template objects and the workflow instance objects will yield a match, after which resource scores may be tabulated and then the adapted workflow implementation may be deployed, replacing the original workflow implementation. The result would be to expedite the process by automating a manual task automatically.

5.2 Resource Availability

The mere identification of a pattern for a given instance does not necessarily make it immediately implementable. Adaptation should only occur under specific conditions that ensure the pattern-modified implementation would create an improvement over the current workflow implementation. These conditions are related to the resources or actors who will be involved in the new adapted implementation.

It is important to ensure that the correct personnel are selected and that they are suitably available to partake in the adapted instance. Given that the pattern attribute is based on both historical workflow log data and the current instance's attributes, in most cases a safe assumption can be made that all users that shall be involved in the adapted instance shall be appropriately selected. A potential example of where an incorrect user may be selected is in the specific case of delegations, which shall be explained in Section 5.3.2.

The availability of a user is also critical. It is not uncommon for a user to be involved in multiple workflow instances at the same time. Care must be taken to ensure that the user selected in the workflow is not overburdened or pulled from other tasks unless the executing instance is of relatively greater importance. If the user is involved in multiple concurrent workflow instances, statistics related to how much time the user generally requires to complete a work-item and how many tasks in which he or she is involved become important in deciding whether to allocate the work-item to the user in question and therefore adopt the pattern identified.

To take all these factors into consideration, a composite resource availability score has been developed which consists of the following factors:

(a) Priority of the current workflow instance

(b) Concurrent workflow association

(c) Average Work-item Execution Time

Other factors, if identified can be added using the general approach proposed here.

5.2.1 Priority of the Current Workflow Instance

Workflow instances have varying degrees of priority depending upon the urgency with which they must be completed. For example, when a request for information is being created, the initiator has the responsibility of ensuring that an appropriate priority is assigned to the instance. This may be confirmed or modified later by a coordinator. In the Coreworx© EPPM system, there are three different kinds of priorities that may be assigned to a workflow instance: high, medium and low.

If a workflow instance has a relatively higher priority, this implies that a response is sought at the earliest (provided of course, that a user of suitable authority is able to respond while adhering to compliant protocol). Consider the case where multiple users are able to respond, but only one response is deemed sufficient as indicated during the creation of the instance. An adaptable implementation which recognizes that a particular user responds more often successfully and correctly – this is described in detail in section 5.5 – can reduce time by automatically selecting that user who should respond rather than a selection of users.

As such, a proportional factor reflecting the priority of the workflow should be considered while deciding whether the user should be drafted into the current workflow instance. The priority score, P_s may be determined by the following formula

$$P_s = 1 - \frac{i_{rank}}{n+1}$$

Where i_{rank}, is the rank of the given priority from higher to lower, and n describes the full range of priorities that can be attributed to a workflow instance. For example, if there are three priorities; high, medium and low, then the priority score for a workflow instance with low priority will be 0.25.

5.2.2 Concurrent Workflows

Document	Workflow ID	Work-Item	Date	Time	Actor
RFI-ECA-INT-1025	1882	Respond	2/17/2012	9:12:29 PM	User_123
RFI-SO-00060-0005	1904	Respond	2/21/2012	6:52:13 PM	User_123
RFI-SO-00060-0006	1905	Respond	2/21/2012	6:51:21 PM	User_123
RFI-SO-00060-0007	1906	Respond	2/21/2012	6:50:46 PM	User_123
RFI-SO-00060-0008	1907	Respond	2/21/2012	6:50:06 PM	User_123
RFI-SO-00060-0009	1914	Respond	2/21/2012	6:43:29 PM	User_123
RFI-CON-0040S-1358	2063	Respond	3/12/2012	4:26:05 PM	User_123

Table 5.1 Illustration of User Active in Concurrent Workflows

Table 5.1 provides an example based on data from a construction project in British Columbia wherein the Coreworx© system was employed, depicting how a user was assigned to the same activity for five of eleven workflows in under 10 minutes. This demonstrates both the pace at which RFIs may be generated during a project as well as how quickly a user may be inundated with requests from different workflows and their respective coordinators. Of interest in this scenario is that the user was assigned to the work-items for workflows in the reverse order in which they were created, as indicated by the workflow ID.

This is not an uncommon case and can be explained as such; at some point in the preceding workflows, perhaps another user was assigned to multiple workflows and proceeded to work on them in the order in which they showed up on his dashboard. The order on the dashboard of the Coreworx© system is based on a Last-In-First-Out format. What this case demonstrates is that the complexity with which determining the number of concurrent workflows that a user is assigned to will vary with the EPPM system used, and therefore care will have to be taken while selecting a time-frame within which the adaptation algorithm is activated.

Clearly, as the number of simultaneous workflows to which a user is assigned increases, the ability to specifically be chosen on an adapted instance diminishes. Given that factors such as the kind of activity – the time it demands – and the priority of a workflow may be considered independently, the value ascribed to the number of concurrent workflows factor may be considered to be equally distributed to all the workflows, as described in the following equation:

$$C_s = \frac{1}{n_c}$$

Where C_s is the concurrent workflow score for the user for that workflow, and n_c represents the total number of simultaneous workflows the user is assigned to, including the adapted workflow.

5.2.3 Average Work-item Execution Time

In chapter four, time distributions were derived for each of the work-items in the request for information workflow template. These were derived for all users who worked on a particular work-item, so as to be able to arrive at expected confidence intervals for which the majority of tasks should be completed compliantly. However the response times of individual users varies significantly.

The time taken by a user to complete a work-item could be an important indicator for why that user has been recognized in an emergent pattern. It is also important to distinguish between the time a user takes to complete specific work-items and the average time the user requires for any work-item. This is important because if a user happens to be working on concurrent activities, if there is a significant difference in the time taken to complete various tasks, then this must be reflected in the final resource score in order to make an accurate assessment for whether the adaptation may be selected.

For the user in question, $t_{work\text{-}item}$ represents the median time taken for that particular work-item, provided that there are sufficient representative samples for the work item in question. If a user has not undertaken a representative sample of activities for one of the concurrent tasks, then the average time that the user required for all work-items shall be used instead.

While it might appear to be computationally expensive to determine the average times for all users over all tasks at regular intervals, there are several advantages that make such an analysis compelling. The data related to the times taken by users is already recorded in the EPPM database system. Performing operations such as aggregation, averaging etc., process this data into information. Comparing and deriving connections from information leads to the creation of knowledge and the development of such business intelligent tools can provide a company with insight into how operations are running and how they can be further improved. These are the same sorts of tools and metrics used by journal article editors to rank the performance of associate editors and reviewers.

5.2.4 Composite Resource Availability Index

The composite resource availability score provides an indication about the user's relative availability to partake in the work-item(s) in the adapted workflow implementation. The score is calculated by considering the criticality of the work-item in the adapted implementation, the anticipated time it will require and whether employing the user in the adapted instance will have a significantly detrimental effect to other workflow instances that the user may be involved in concurrently.

The composite resource availability score, given by A_{cr} is given by the following equation:

$$A_{cr} = \frac{C_s \times P_s \times t_{work-item}}{\sum_{i=1}^{n-1} P_{s-i} \times t_i}$$

Where: C_s is the concurrent workflow score,

$t_{work\text{-}item}$ is the anticipated time that will be required for the adapted work-item

P_s is the priority score for the adapted workflow instance

n is the number of concurrent workflows the user is involved in,

t_i is the average time taken for the i^{th} work-item in the list of concurrent workflows,

$P_{s\text{-}i}$ is the priority score for the i^{th} workflow in the list of concurrent workflows.

If the resulting composite resource availability score is greater than a threshold, the adapted workflow may be implemented.

5.3 Case Study: Observed Repetitive Workflow Patterns

In an ideal EPPM system, an adaptive framework would be implemented prior to the system being deployed for a construction project. Such a system would search for specific patterns based on thresholds and samples sets established during discussions between the vendor and client(s). This would have permitted detailed analysis of the adaptation patterns identified as well as the adapted workflow instances that were implemented upon parameters and resource scores meeting decided upon thresholds.

Installing self-adapting workflows on an EPPM system for this research initiative would not have been practical however for the following reasons. Before a research application can be developed as product for an EPPM system several steps have to be completed. First, a proof of concept model must be developed demonstrating the utility and feasibility of incorporating the technology. The software product must then be designed taking into consideration the existing system's architecture and platform. After the product has been designed, initial prototypes must be developed and tested extensively. The finished product would further require extensive discussions between the vendor and client prior to being implemented on an existing system or as part of a system being deployed for a new construction project.

5.3.1 Data Source for Analysis of Workflow Histories

Clients who were interviewed in Calgary during October 2012, expressed enthusiasm about self-adapting workflows, and went so far as to rate their incorporation as the most desirable change they would like to see in their EPPM systems. To demonstrate the utility and feasibility of implementing such a system, a case study was conducted on data that was made available for analysis by a client. The purpose of the case study was to identify specific adaptable patterns and then to assess and demonstrate the effectiveness self-adapting workflows would have contributed towards the project.

The study was conducted on an EPPM system developed by Coreworx Inc., that was deployed for the construction of cabin gas plants for a leading EPC client in British Columbia. The study focused on 1070 RFI instances that were created in the first of six stages of the project. The originally deployed RFI template was modified eight times, resulting in nine implementations being used for the first stage of the project alone. The high number of modifications indicates that as the project proceeded, clients began to understand their system and work processes better and began modifying the template to reflect this improved understanding. An autonomous adaptive system that would have been able to have inferred these changes based on user interaction and behavior, and seamlessly implemented them would automated, expedited and have reduced the number of modifications that were made to the primary workflow template, and therefore would have saved time and associated costs.

5.3.2 Determining Patterns from Workflow History Logs

The Coreworx EPPM system consists of four primary components: (a) a document management system, (b) a workflow engine, (c) a compatibility viewer for rich media display, and (d) a portal shell for user interaction. The workflow engine that was employed for the construction project was a third party application that was integrated into the Coreworx EPPM system. The workflow engine application maintained logs of executed workflow instances within an internal database that was independent in nature to the rest of the Coreworx system apart from a few modifications to enable connections to other parts of the system. Analysis to determine patterns relating to workflow history were therefore primarily confined to this component database.

This database maintained data about who worked on a work-item, when they were assigned to the work item, when they started and finished working on it, the name of the activity, the version of the workflow employed and the status of work items. Information relating to the roles and responsibilities of users, and therefore the permissions they were entitled to, were maintained in a RACI matrix outside of this system as this concerned interactions between the user and the portal. In addition, information relating to files that were attached, and even to the actual RFI document and associated meta-data, were stored in the document management system. As such, information strictly related to workflow history was analyzed.

The following patterns were observed to have occurred with particular frequency:

(1) Delegation of tasks by an actor consistently to a subset of available actors

(2) Frequent path selection at forks in a consistent choice of path

In the following sections, each pattern has been described in detail, followed by adaptation-specific methodology that was developed to address the case. Finally the validity and benefits for implementing these patterns are discussed.

5.4 Delegation of Tasks by an Actor Consistently to a Subset of Available Actors

5.4.1 Introduction

Delegations are factored into the design of workflows to accommodate flexibility for an actor. Delegations are typically made when the primary actor originally assigned to a task wishes to hand the task over to another actor as a subordinate, alternate or proxy. There are a variety of reasons for why an actor may choose to delegate a task to another, ranging from whether the actor is on leave to whether the actor believes another actor is in a better position to handle the work-item.

It might not always be possible, if at all, for an algorithm to determine why an actor has chosen to delegate a task – because in many cases external factors (at least to the system) may influence the decision – but at the very least, an algorithm can detect when a delegation has occurred. In many instances this knowledge alone will be sufficient for an adaptation pattern to be recognized and implemented should conditions favor it.

5.4.2 Analysis of Workflow History Logs

Delegations are stored in two database tables; one named “delegated_by” which stores data related to the identity of the user who delegated the task, when he or she delegated the task, the time it took to delegate the task and the work-item id that the user delegated, and a complimentary table named “delegated_to” which stores similar data about the user whom the task was delegated to. Figure 5.2

below presents a snapshot of the number of instances over time recorded in the "delegated_by" database table for the period March 30th 2012 to May 21st 2012. Raw data (names changed to protect privacy) from a query which combined data from both the "delegated_by" and "delegated_to" tables which was used for the figures below is included in Appendix B.

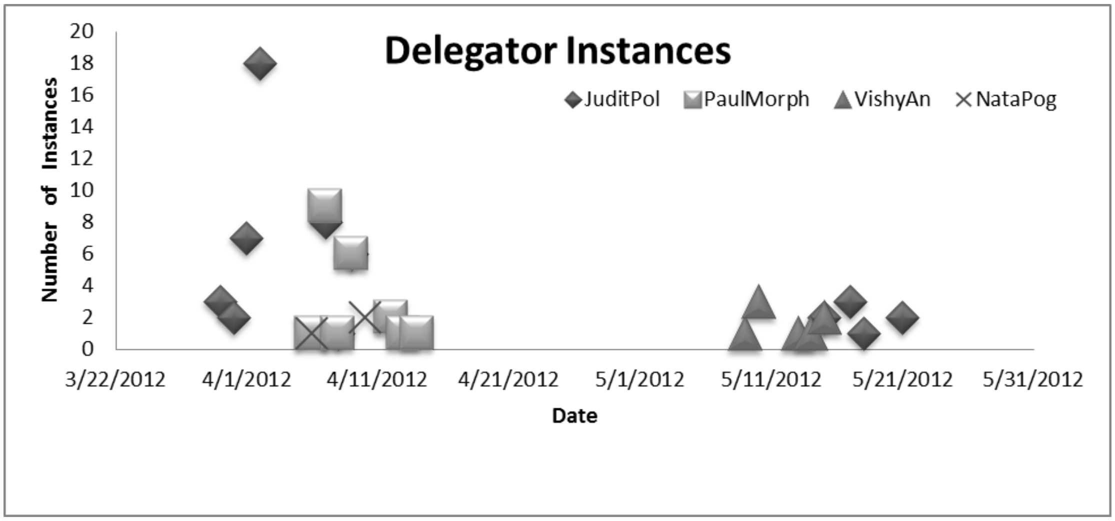


Figure 5.2 Delegator Instances by Date

.Figure 5.3 below presents the same graph over the same time period but for the "delegated_to" table:

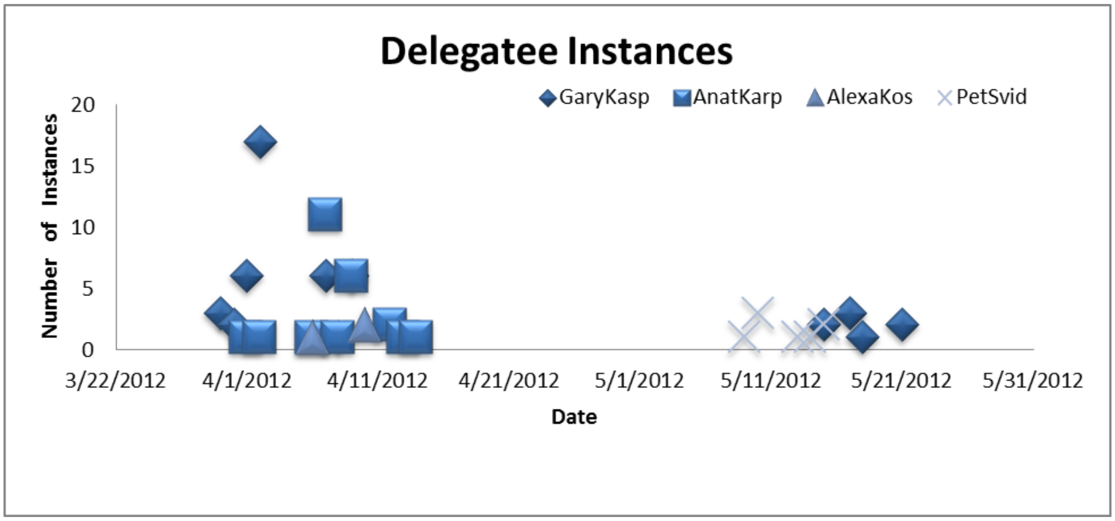


Figure 5.3 Delegatee Instances By Date

We note in Figure 5.2 that there are four delegators, and that there are clusters of delegations made at different times. Again in Figure 5.3 there are four delegatees, and on comparing both figures we can see that clusters of users from one graph corresponds with the cluster of users in the second graph, thereby implying that delegations often happen between the same delegator-delegatee pair in short intervals of time. Indeed, this is a common occurrence that was noted for delegations over the entire project's duration. Table 5.2 below provides a full list of delegator-delegatee pairings that were observed from the data. (All names have been changed to protect privacy as part of a non-disclosure agreement that was signed with the client who made data available for this project).

Pairing	**Instances**	**Percentage**
JuditPol—GaryKasp	52	12.04
AronNimzo—GaryKasp	49	11.34
VishyAn—GaryKasp	44	10.19
AnatKarp—PaulMorph	28	6.48
GaryKasp—AronNimzo	25	5.79
AlexAlk—BobFisch	23	5.32
AnatKarp—MagnusCarl	22	5.09
HikaNaka--GaryKasp	21	4.86
JoseCapa--BobFisch	17	3.94
Dan Groza--VishyAn	14	3.24
AnatKarp--GaryKasp	13	3.01
VishyAn--EmLask	12	2.78
BobFisch--BobFisch	11	2.55
LevAron--VladKram	9	2.08
AronNimzo--BobFisch	8	1.85
4 to 6 pairing instances	42	9.72
2 to 3 pairing instances	26	6.02
Single paired instances	16	3.70
Total	**432**	**100**

Table 5.2: Delegation Pairing Instances

The data source contained 1070 RFI workflow instances which contained a total of 22,986 work-items. The first 116 of these work-items have not been included for analysis as they were part of a training program, comprising 15 trial workflow instance examples, that Coreworx Inc., offers clients to help them become acquainted with the system. There were 432 delegations made in total excluding the training data. Delegations, considering that each delegation consists of two work-items, therefore represent a total of 3.78% of the total number of work-items.

More than two thirds of the delegations consisted of pairings with more than 10 delegation instances. Studying Figures 5.2 and 5.3 again, we note that in general there are short bursts of delegation pairing instances that occur over intervals of time. For example, the delegation pair JuditPol-GaryKasp occurred over two short intervals of time and each interval of time consisted of more than 10 instances. Given this data, we can postulate that if a certain number of consecutive work-item instances are delegated by one user to another within a short period of time, then it is likely that the next several work-items may be delegated to the same user. This constitutes the adaptation pattern that maybe then be automated for the next set of work-items until a stopping criteria has been identified, or an actor manually stops adaptation.

5.4.3 Adaptation Specific Methodology

The following is the adaptation specific methodology that has been developed for delegations. It is evident from the above workflow analysis that the number of delegations as a percentage of the total number of work-items is not high. Further, delegations occur in short intervals of time between pairs of actors. Therefore the delegation pattern may be considered to be temporal in nature. As such, while pairs of actors may be determined from a training set, it is likely that new pairs will be created over the duration of the project. Therefore detection of the delegation patterns must be an ongoing process.

The pattern itself should be detected after a sample size of *n* consecutive work-items are delegated from one actor to another, within a time period, *t*. Both *n* and *t* may be specified by the client unless they are willing to accept default values that the vendor recommends. An important parameter that must be determined immediately is the stopping criteria, since otherwise delegations may occur infinitely without the delegating authority being aware that they are occurring and missing crucial tasks that they might be available to perform. It is not possible to estimate an exact number of tasks that may be delegated by analyzing available data. This information must be provided by the delegating authority.

Once the pattern has been registered, it is recommended that immediately a prompt is sent to the delegating authority requesting a stopping criteria; either a fixed number of following work items or all work item instances until a particular date. Additionally a secondary delegate may be requested or proposed in the event that the primary delegatee's availability does not meet the composite resource availability score.

The pattern is then automatically initiated for all such tasks within the window selected by the delegator, and the parameters that are evaluated at the initiation of the pattern implementation are the delegator's ID and the stopping criteria. This is followed by determining the primary delegatee's composite resource availability score. If the score threshold is met, then the pattern is implemented. If not, then if a secondary delegatee is available the process is repeated. In the event that there is no secondary delegatee or that the threshold is not met, then the task is sent to the primary delegatee and shall remain dormant until such time as he is available, as would be the case if the adaptation implementation did not exist.

5.4.4 Validation

Implementing these algorithms on the existing data set has no value. Implementation on a real project would validate the efficacy of the algorithms but was outside the scope and resources available for this research effort. As the adaptation algorithms have not been implemented for real data, it was proposed to evaluate the benefits of utilizing the adaptation by studying the effect they would have had if they were implemented for the available system. The average time that was required per delegation by a delegator from the available data was 1 hour and 8 minutes. Effectively we may say that for a large number of tasks, this was 1 hour and eight minutes that was lost when the task may have been sent to a delegate immediately. It should be noted that in some cases a delegator received a task towards the end of a week and was only able to delegate the task at the start of the next week. However, for the purpose of simplicity, the average time shall be considered.

Considering that 231 work-items consisted of 10 or more sets of 13 delegation pairs, we may approximate that if the pattern was recognized after three consecutive work-item instances that the time required for 231 – 39 = 192 delegation cases may have been saved. This equates to 218.5 hours or effectively over five and a half man-hour weeks of time that was lost over the course of the project. At a typical professional rate of $150/hour, this is a cost of over $30,000.00 for the project. Hence there would definitely have been time and therefore costs savings for the project had the delegation adaptive workflow algorithm been implemented. There may also be small schedule advantages.

5.5 Frequent Similar Path Selection at Forks in a Consistent Choice of Path

5.5.1 Introduction

Often within an RFI, a work-item will be assigned to multiple actors. The criteria for the completion of that stage and moving to the next stage of the workflow varies depending on the RFI implementation definitions, but can require just one actor or any other subset of actors to complete the work item, or require all the actors to complete the task. If the task has to be completed by any combination other than all the personnel it was assigned to, then there are number of combinations of subsequent paths that the workflow implementation can follow for the completion of the task, depending on which actors complete the work item.

These points at which such combinations are possible are called forks in the workflow implementation. A common stage at which they occur in an RFI is during the “Respond” stage, which follows the “Verify Participants” work-item, and is followed by the “Review and Consolidate” work-item wherein the responses are collated and organized for approval. Figure 5.4 below illustrates one such case which appeared in an RFI implementation wherein two responders were assigned the "Respond" work-item and only one response was required for the workflow to continue to the "Review and Consolidate" phase.

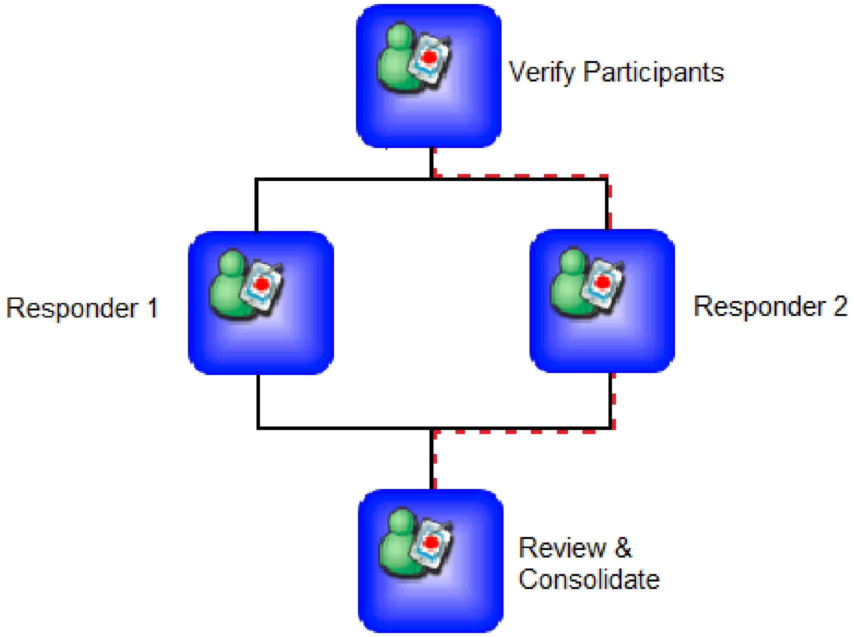


Figure 5.4: Frequent Path at a Fork in an RFI Workflow

Over the course of several workflow instance executions of these RFI implementations, familiar patterns may emerge showing that one actor, or set of actors, responds more frequently and successfully than others who were also assigned to these types of work-items, as indicated by the red-dashed line in Figure 5.4 above.

If such cases are identified then for future instances, rather than assign the task to all actors in the work-item it might be more efficient to send the work-item to the specific subset that are more likely to complete the task. The determination of whether such an adaptation can occur however should take into consideration not just the frequency of the path selection but also resource availability. It might also consider the desirability of rewarding responsiveness with more work, although it does correspond with aphorism “If you want a job done, give it to a busy person!”

5.5.2 Analysis of Workflow History Logs

When a work-item is issued to an actor, a record is stored in the database noting the time that the work-item was issued as the “Start Time.” This is not always indicative of the time the actor first noticed that the work-item was issued to him or her, a record or state change which is not currently saved in the database. If the actor does not complete the work-item before a predefined duration – in the British Columbia project, this was set to 3 days, which is a default setting for the Coreworx system – the item “times out” and a notification is sent to the actor informing that he must complete the work-item at the earliest possible time. The time that a work item is completed is noted as the “End time”, although this does not always represent the time that the actor completed it but rather the time that the work-item moved to the next phase. The status of the work-item at the “end-time” is also recorded. The different statuses that were defined for work-items in the project by Coreworx Inc are displayed below in Appendix A.

Considering the list of status definitions, the only statuses which may represent successful completion of a work-item by an actor are CO – denoting work item completion, AC – denoting the work-item was completed as the min-max ownership count was satisfied and WC – denoting an external decision was taken to end the work-item. If the work-item was issued to two actors and only one was required to complete it, then the status recorded for the work-item as issued to the actor who completed it would be CO while it would be AC for the other actor. Statuses represented by WC are usually reserved for “Warning” items, which in the Coreworx system are saved as work items rather than an attribute of a work item. Warnings are sent when the predefined time-limit nears and the work-item requires immediate work by an actor. Therefore, discriminating criteria exists to determine which actors completed a work-item and which actors did not.

For the project at British Columbia, the “Respond” work-item frequently could have been completed by two or more actors. Raw data related to two of these actors is presented in Appendix B. If the first 1000 workflows that were completed on the system are considered as a training base to determine patterns based on user interaction, the number of work items that were assigned and the eventual status of these work items after being completed by the actor provides a basis for that actor’s successful completion for that stage of the project, and could act as an indicator for later stages. Based on the raw data issued from the project, the following summarizations could be made for each actor (as in the delegation case, all names have been changed to protect privacy as part of non-disclosure agreement).

Actor	Total Work items	Completed with status CO	Time out, Abort & other Statuses	Warnings issued
SunnyGav	47	43	4	4
KrisSrik	53	29	24	15

Table 5.3: "Respond" Work Item Completion Records For Actors

Table 5.3 above shows a stark difference in the successful completion of work-items between both actors. While the actor SunnyGav completed over 90% of the work items issued to him successfully, the actor KrisSrik only completed 54.7% of his corresponding activities. Perhaps more indicative of who was more prompt in completing a work item are the number of time-outs that were issued. In addition, warnings that were issued (which were not counted as specific work-items in this analysis are they do not constitute the work-item “Respond” being completed) are significantly higher for actor KrisSrik. Of course, quality of the work cannot be measured and depending on the work item, this may have significant value, in addition to prompt and nominally successful completion.

It may thus be summarized that from this training data, actor SunnyGav was likely to complete over 90% of his work-items within the pre-defined time duration, while the corresponding probability of actor KrisSrik is 0.547. Similarly we can store information related to other statuses and warnings as attributes to each actor. The values of these attributes may then be used as discriminating factors in the first stage of automated selection of the user at a fork in the workflow, as descriptors of the emergent pattern. These metrics might also be issued in an advisory capacity to senior managers as well.

5.5.3 Adaptation Specific Methodology

Figure 5.5 below illustrates the specific methodology developed for this case, inherited from the general methodology in section 5.1:

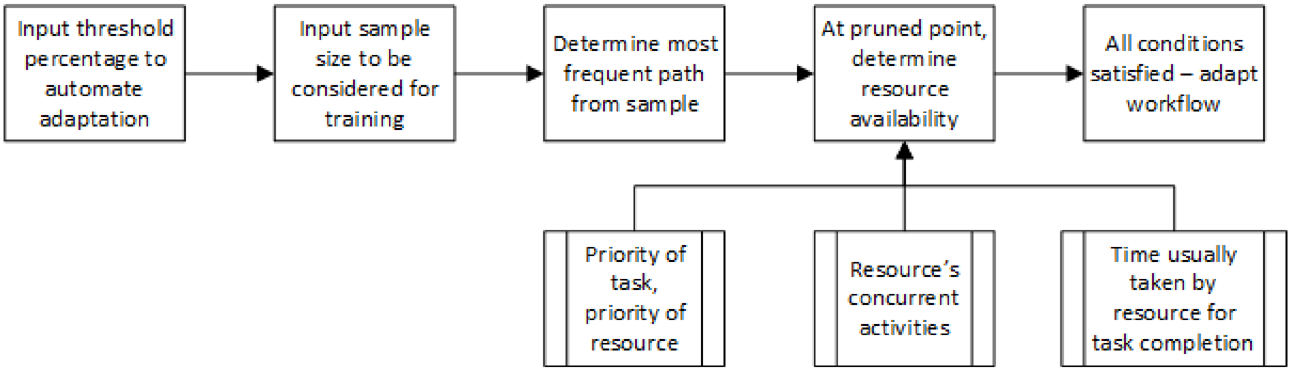

Input threshold percentage to automate adaptation
Input sample size to be considered for training
Determine most frequent path from sample
At pruned point, determine resource availability
All conditions satisfied – adapt workflow
Priority of task, priority of resource
Resource's concurrent activities
Time usually taken by resource for task completion

Figure 5.5: Frequent Path Self Adapting Methodology

The first two steps have been described in detail in the general methodology and have been preserved for this specific case. The most important step in this analysis is the determination of the frequent path from the sample. In an ideal situation, a significant number of cases would exist wherein both actors work exclusively together on work-items in the sample set, and based on which actor completed the task first each time, it would be possible to establish branch possibilities. When a paucity of such data exists, as was the case for all actors in this project over the entire sample set and not just the first 1000 workflows discussed in the previous section, comparisons may be established based on independent task completion statistics. Comparisons between the various attributes associated with each actor may be used as discriminating criteria to establish branch probabilities for each actor completing the work-item. In the case described above, there would be a clear preference for selecting actor SunnyGav over actor KrisSrik. However, before the pattern can be adopted, the actor SunnyGav's resource score must be calculated and if it turns out that his composite resource index score does not permit the adaptation, then the adaptation is abandoned. Resource scores have been described in detail in Section 5.2..

5.5.4 Validation

Table 5.4 below, presents how each SunnyGav and KrisSrik performed on tasks they were both involved in for the testing data set:

Work item Status	SunnyGav	KrisSrik	Total
Time-out	6	15	21
Warning	6	17	23

On analyzing the testing data set, it was determined that both SunnyGav and KrisSrik were assigned to the same "Respond" work-item together for a total of 31 workflow instances. In these instances, there were a total of 21 time-outs, and KrisSrik was responsible for 15 of these time-outs. However, out of the 6 instances that SunnyGav timed-out, KrisSrik timed-out in 4 as well, indicating perhaps that both actors were busy with another aspect of the project. In such cases, the resource score would not have allowed the adaptation to occur in any case. Further, a total of 23 warnings were issued, and

17 of these warnings were issued to KrisSrik, indicating that there were two additional warnings sent to KrisSrik on workflows that were assigned to both actors.

The above records indicate that the pattern persisted throughout the project and that an adaptation with a preference for the RFI route including SunnyGav over KrisSrik would not have affected the overall outcome of the workflow detrimentally. Further, if one considers that the time-outs and warnings consisted of 72 hours durations, we already note that a potential 792 hours or 88 man working hours were lost waiting for KrisSrik to respond, which equates to about 4 schedule weeks. Further the average time taken by KrisSrik to complete work-items in the testing set was 44.75 hours, while the corresponding time taken by SunnyGav was 17.57 hours. Therefore, significant time savings would have been achieved if the frequent path pattern had been adopted with the respond work-item being issued to SunnyGav instead of to both participants.

5.6 Conclusion

A general methodology was developed for the implementation of self-adapting workflows in construction EPPM systems. The methodology considered the implementation of an adaptation based upon the recognition of emergent patterns from workflow history logs and the availability of resources that are required if the workflow was to adopt the pattern. Two specific cases were analyzed, on data that was obtained from the first stage of a construction project in British Columbia. In one case the automation of task delegation was analyzed and in the second case intelligent path selection at forks in a workflow implementation was studied. The analysis showed that the adoption of self-adapting workflows would have resulted in significant time savings for the project. Projected over other workflows, such as CR (change requests), IM (interface management) and emerging process workflows such as scaffolding requests, the impact of these automated adaptations could be in the hundreds of thousands of dollars for projects in the capital range of hundreds of millions of dollars. As a percent of the total project cost, this benefit is small, but it benefit to cost ratio with respect to incremental implementation cost over the existing EPPM system customer base would be extremely high.

Chapter 6

Determining Critical Construction Documents in EPPM Systems

In analyzing workflows to determine their compliance or the potential for automating self-adaptation, the focus tends to be upon the resources, actors and functioning of the workflow based upon its design. However, a crucial aspect of workflows pertains to the flow or the information – in the form of documents or electronic products such as BIM models – that is passed between different participants during the execution of a task. We often hear the phrase “missing the woods for the trees” to describe circumstances where the “big picture” is overlooked in favor of minute details, however in certain cases it is also true that it is possible to “miss the trees for the woods.”

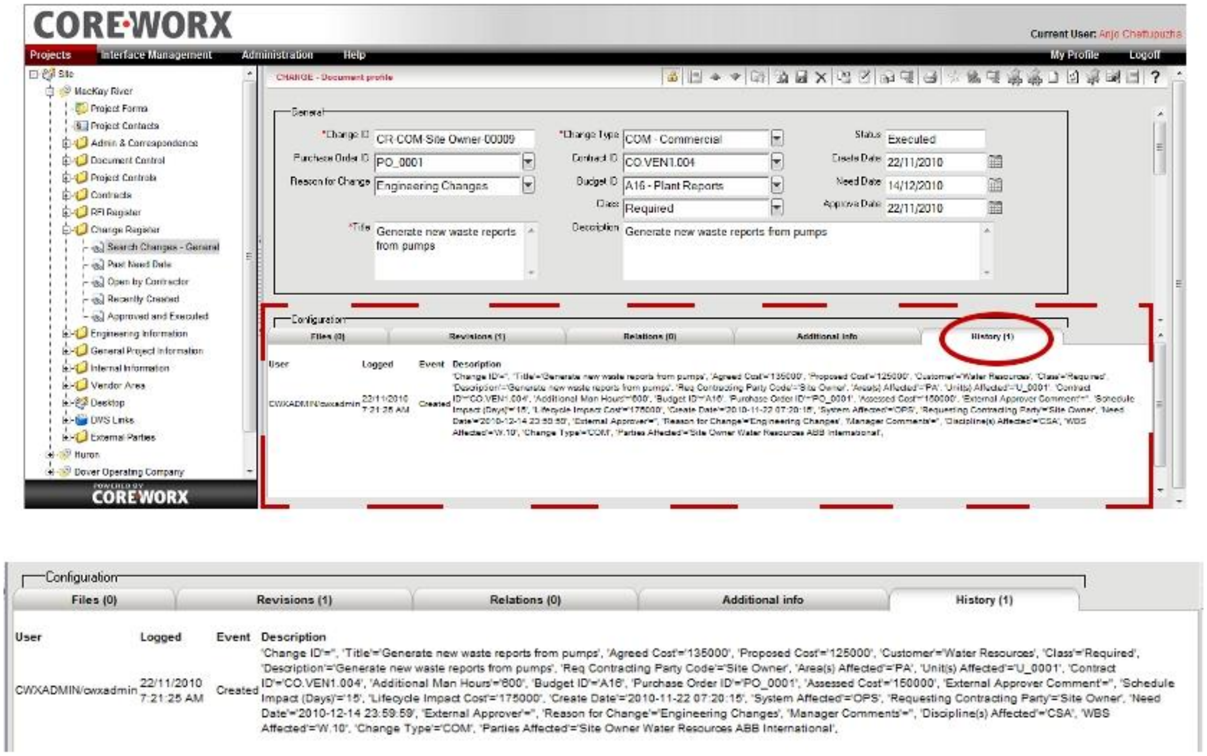
As an analogy, consider the macroscopic nature of analyzing transportation networks, wherein the structure, behavior, and status of traffic in a network is often studied treating the entities that flow through it, vehicles, as a class of similar if not identical entities that share the same properties. This generalization assists the analysis by reducing the number of complexities and variables whose impact may be negligible to the final outcome. However, all vehicles are not the same and assuming homogeneous behavior discounts the impact of certain specific cases, which leads to either blanket assumptions that a “black swan” event cannot occur or unnecessarily assumes every entity to be a potential black swan.

It is not unreasonable to apply the same paradigm to workflows and the flow of their constituent entities; documents or other electronic products. It becomes important then, to consider whether the actual information or project documents that are directly associated with this flow could impact the efficiency of a workflow. In some instances, a workflow’s design may not contain any inherent flaw, but referencing an incorrect or incomplete version of a specification could cause a delay and in the worst case result in costly repairs or redoing a section of work. This is particularly relevant during the design phase of a project, as it may signal conflict of confusion among stakeholders. Even during the construction phase of a project (as against the design phase when documents may be frequently updated or modified without directly affecting material or labor costs), as an incorrect reference of a revision of a document directly affects an activity in the field. This in turn intuitively establishes when a particular revision of a document has become a key document that shall be referenced over the rest of the project.

Identifying the characteristics of such potentially problematic documents would be useful for risk mitigation as well as for improving the efficiency of the project. Real time monitoring systems may be employed to catch such exceptions and raise a warning or provide for other appropriate action if the document is deemed an important resource. Purely from a knowledge management perspective, a core tenet of the discipline is to ensure that accurate and pertinent knowledge is delivered at the appropriate time, and therefore it is imperative to ensure that counteractive or pathogenic knowledge does not enter the workflow information flow.

6.1 Leveraging an EPPM System’s Inherent Advantages For Document Criticality Determination

Traditional construction information systems do not explicitly store information related to how documents are involved in construction activities. However in an EPPM system, workflows are employed to automate construction project management and engineering design process activities. Workflows include information relating to the order in which process tasks must be executed, the actors who must execute them and the human resource requirements for a particular task. Information about the processes includes information about when documents are to be attached to an activity. Further, EPPM systems typically maintain a structured document management system for improved organization and retrieval of documents. The nature of an EPPM system, as a repository for process oriented information flow, puts it in a unique position to be used to exploit the extraction and storage of inherent knowledge encapsulated within documents and their movement over the project life cycle. In an ideal EPPM system, information relating to when a document should be accessed to which other documents it is linked and every single instance that it is accessed is also recorded.



*Change ID	CR-COM-Site Owner-00009	*Change Type	COM - Commercial	Status	Executed
Purchase Order ID	PO_0001	Contract ID	CO.VEN1.004	Create Date	22/11/2010
Reason for Change	Engineering Changes	Budget ID	A16 - Plant Reports	Need Date	14/12/2010
		Class	Required	Approval Date	22/11/2010
*Title	Generate new waste reports from pumps	Description	Generate new waste reports from pumps		

User	Logged	Event	Description
CWXADMIN/cwxadmin	22/11/2010 7:21:25 AM	Created	'Change ID'='', 'Title'='Generate new waste reports from pumps', 'Agreed Cost'='135000', 'Proposed Cost'='125000', 'Customer'='Water Resources', 'Class'='Required', 'Description'='Generate new waste reports from pumps', 'Req Contracting Party Code'='Site Owner', 'Area(s) Affected'='PA', 'Unit(s) Affected'='U_0001', 'Contract ID'='CO.VEN1.004', 'Additional Man Hours'='600', 'Budget ID'='A16', 'Purchase Order ID'='PO_0001', 'Assessed Cost'='150000', 'External Approver Comment'='', 'Schedule Impact (Days)'='15', 'Lifecycle Impact Cost'='175000', 'Create Date'='2010-11-22 07:20:15', 'System Affected'='OPS', 'Requesting Contracting Party'='Site Owner', 'Need Date'='2010-12-14 23:59:59', 'External Approver'='', 'Reason for Change'='Engineering Changes', 'Manager Comments'='', 'Discipline(s) Affected'='CSA', 'WBS Affected'='W.10', 'Change Type'='COM', 'Parties Affected'='Site Owner Water Resources ABB International',

Figure 6.1: Coreworx (c) System History Tab Displaying Document History

In EPPM systems, documents are divided into distinct classes based upon the process they are modeled around, such as RFIs and Change Requests. Customized templates are designed for each class containing specific meta-data fields that all documents within a class will require. These customized document templates are referred to as document profiles. In addition, usage characteristics such as when the document was changed recently, who made the changes etc., are recorded. Storing usage characteristics are important for auditing and litigation purposes and are usually tracked in history revisions of document profile instances. Anyone familiar with article review histories in an editing system will have some idea of the type of information stored in an EPPM system for a construction project. Figure 6.1 on the previous page, displays the history tab in the Coreworx system where this information may be viewed from within a web browser.

In analyzing information systems, there are several different sources that may be mined depending on the type of knowledge that is being sought. Often in web-based systems, there are three main sources; (i) the content of a webpage, (ii) the structure of a web-page and (iii) the usage and system logs pertaining to that webpage (Dunham, 2003). Suitable mining methods are adopted based upon the source of data. Mining knowledge from content often employs techniques such as text data mining and natural language processing so as to be able to associate keywords with intelligent contextual understanding of what the document describes.

An instance, or an executed implementation, of a document profile may contain attachments pertinent to that particular activity. For example, an RFI may contain a reference document or a design drawing. Due to confidentiality agreements between contracted parties in this research project, the content of these attachments were not accessible, and therefore intelligent contextual and semantic information from these attachments was not considered for this research initiative. This might have been a blessing in disguise, as semantic interpretation may well be subjective, and often privacy laws prevent access of copyrighted content.

Mining knowledge from the structure, usage and systems logs may be attained from workflow histories. The content of a document in this regard ceases to be as important as the structure, or the relations and links between documents in the overall context of an information system in a construction project, and the usage characteristics associated with the document. Such information will enable rapid assessment of wasteful access and distribution of a document without computationally expensive mining of the content of a document. As such, in an ideal EPPM system, it should be possible to obtain critical insight into a document's importance by analyzing associated workflow and usage statistic histories from stored document meta-data fields.

Interviews were conducted with EPPM system development consultants and construction clients to determine criteria that establish the criticality of a document at a particular phase of a construction project. Based on the feedback obtained from these experts, the following factors were identified as necessary for the determination of the criticality of documents:

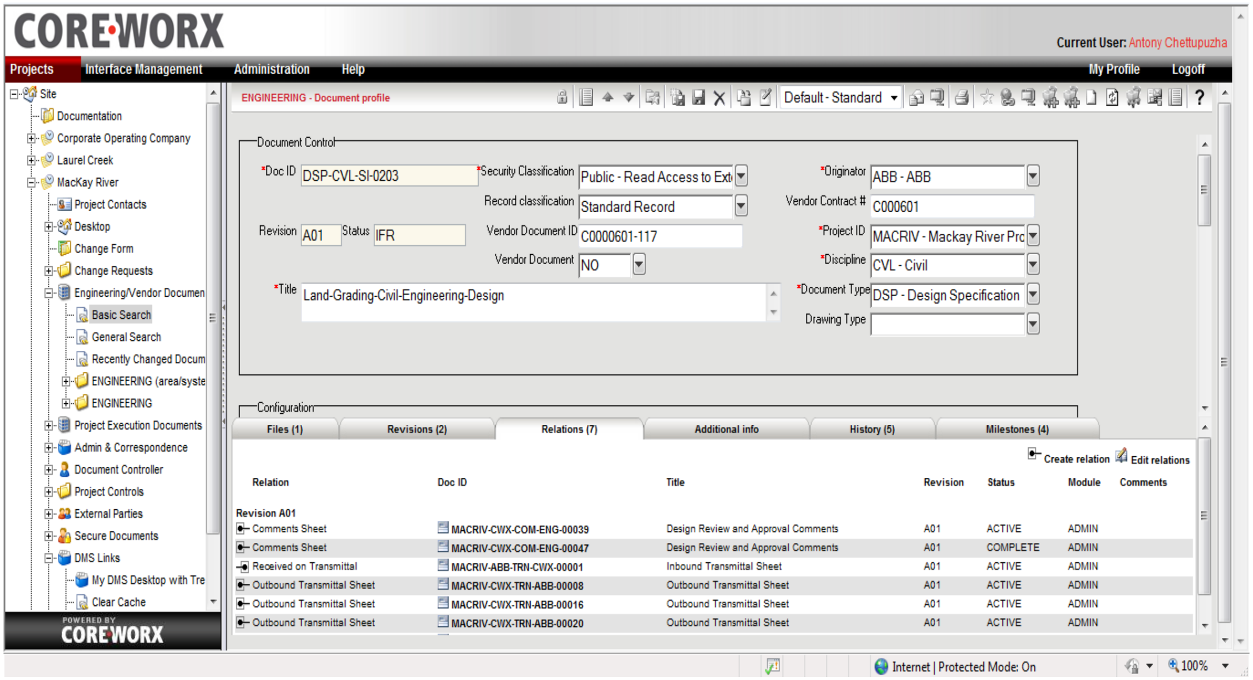
1) Degree centrality of a document
2) Ratio of actual to expected idle time of a document
3) Average access of a document
4) Revisions that a document has undergone

Each of these factors is described in the following subsections followed by methodology which may be employed to determine the criticality of a document based on each factor and based upon a combined score for all factors.

6.1.1 Degree Centrality of a Document

The Coreworx system maintains an internal tracking of one-to-one relationships between documents that are linked to the same workflow. For example, if an RFI workflow instance contains a specification, say A, attached by the initial issuer of the RFI and if another participant adds another specification, say B, to assist with the completion of the RFI, a link is created internally in the Coreworx system associating specification A with specification B.

This feature permits searches across the document management system for inter-related document pairs. When multiple documents are attached to the same workflow, many such pairs are created. The list of files attached to a document profile instance is displayed in the "Relations" tab, as part of the configuration and history parameters for that instance. Figure 6.1 below shows the “relations” tab relating seven different documents in a workflow instance:



Document Control

*Doc ID: DSP-CVL-SI-0203
*Security Classification: Public - Read Access to Ext
Record classification: Standard Record
Revision: A01 Status: IFR
Vendor Document ID: C0000601-117
Vendor Document: NO
*Title: Land-Grading-Civil-Engineering-Design
*Originator: ABB - ABB
Vendor Contract #: C000601
*Project ID: MACRIV - Mackay River Prc
*Discipline: CVL - Civil
*Document Type: DSP - Design Specification
Drawing Type:

Configuration

Relation	Doc ID	Title	Revision	Status	Module	Comments
Revision A01						
Comments Sheet	MACRIV-CWX-COM-ENG-00039	Design Review and Approval Comments	A01	ACTIVE	ADMIN	
Comments Sheet	MACRIV-CWX-COM-ENG-00047	Design Review and Approval Comments	A01	COMPLETE	ADMIN	
Received on Transmittal	MACRIV-ABB-TRN-CWX-00001	Inbound Transmittal Sheet	A01	ACTIVE	ADMIN	
Outbound Transmittal Sheet	MACRIV-CWX-TRN-ABB-00008	Outbound Transmittal Sheet	A01	ACTIVE	ADMIN	
Outbound Transmittal Sheet	MACRIV-CWX-TRN-ABB-00016	Outbound Transmittal Sheet	A01	ACTIVE	ADMIN	
Outbound Transmittal Sheet	MACRIV-CWX-TRN-ABB-00020	Outbound Transmittal Sheet	A01	ACTIVE	ADMIN	

Figure 6.2: The Relations tab demonstrating other documents related to the current document

Consider the case for when two files, ‘A’ and ‘B’ are attached to one workflow and one of these files, say ‘A’, is attached to another workflow but is now paired with another file, ‘C’. One may consider that the relationships between ‘A’ and ‘B’ and ‘A’ and ‘C’ to be first degree relationships. While ‘B’ and ‘C’ might not be directly related, one may however derive an implicit secondary relationship between them through their relationship with ‘A’.

Now consider the multitude of files that are attached to the thousands of workflows instances corresponding to a particular document profile. A network of the inter-relationships and their order for all such related documents may be created, thereby demonstrating how all the project documents are related to each other. This type of network may be visually represented by a cluster map. Further, if documents are linked to one another on more than one workflow, the strength of the bond between these documents may further be highlighted by increasing the density of that bond.

Figure 6.3 below displays a typical representation of a density cluster map, obtained from Google's network graph analysis tool for trial data. In this example, the relative strength of the bonds is highlighted by the thickness of the bond edges between nodes.

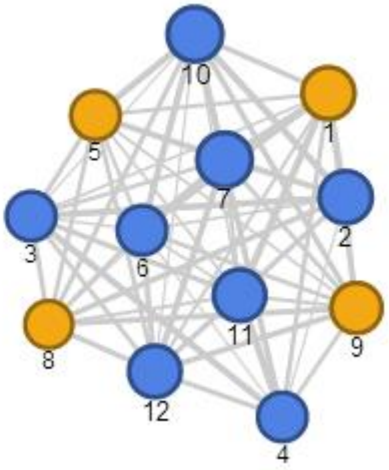


Figure 6.3: An Example of a Density Cluster Map

These types of relationships between entities have been studied extensively in network analysis and graph theory and are most commonly found in the analysis of social networks to determine the relative influence of entities. A common measure that has been used to determine the criticality of a node in a network graph is its centrality. There are several kinds of centralities that are used to measure different influences that the node has on the network. As we are primarily interested in determining documents that are deemed to be relatively more important than other documents by virtue of the number of other documents that they are connected to, the centrality measure selected for this factor is degree centrality. Strictly speaking, degree centrality refers to all the links that are connected to a specific node.

This relative importance of a document via its degree centrality may be demonstrated by an illustrative example. Consider if a document has a very high centrality measure, indicating it is directly connected to several other documents within the system. If the document happens to contain an error, then it is quite likely that several of the other documents that have been tied to it may in some way have related information that is either independent or in worse cases derived from the error in this document. Whether the document has an error or not, its linkage to so many other documents indicates that it is a key document that is usually used as a reference with other documents. In either case, the degree centrality of the document is an excellent indicator of the relative importance of the document in the network compared to all the other documents.

6.1.2 Idle Time of Document in a Workflow versus Expected Idle Time

The flow of documents or electronic products within an EPPMS might not necessarily be continuous, but instead depends on when certain workflow instances are active, or in other words, during the execution of activities or tasks. There are several instances where a document is created for a one-off instance of a workflow and is not expected to be accessed again. Then again, there are sets of documents which might be accessed at regular intervals over the lifecycle of a project. As such there are periods of time wherein the state of document can be assumed to be idle, and other times where the state of document or electronic product is active or in-use.

This in turn reveals that there are periods during which a document is not expected to be used, and if it is being accessed during one of these periods this could indicate a change or deviation – either hard coded or unrecorded – in the workflow, potential access of the wrong document or unauthorized access. Further, there could be documents which are deemed relatively unimportant at the start of a project, but which become increasingly referenced as the project progresses and are hence far less idle than initially anticipated. These documents might then be recognized as critical to the functioning of a certain task. Thus, understanding how often a document's state changes from an expected idle state can be an important indicator of either a potential disruption or the increasing importance of the document in question.

An important consideration in the calculation of idle times is that over lapping durations are only counted once, with the preference going to the workflow duration if access outside of the workflow to which the document is attached overlaps with the workflow duration. This is because we are strictly considering the amount of time the document was considered to be idle to how much time it actually was idle, and do not want to count the same duration multiple times. Hence if a workflow instance duration was from days 5 to 8 and the file was accessed outside of the workflow between days 7 to 9, the access external to the file would only be considered for day 9. Overlapping workflow durations are merely merged – or mathematically speaking, we only consider the union of two workflows excluding the intersection of each – so as to reflect the overall duration of both workflows, and this is similar for accesses. Multiple accesses and overlapping workflows are handled in the next factor.

We may calculate the expected idle time of a document at any time 't' as:

$$Expected\ Idle\ Time = PJ_t - \sum_{i=1,j=i+1}^{n-1,n} (WF_i \cup WF_j)$$

Where PJ_t represents the total project duration up to time 't and WF$_i$ represents the duration of a workflow to which the document is attached, and 'n' represents the number of workflows that the document is attached to.

The actual idle time of the document at time 't' may then be defined as:

$$Actual\ Idle\ Time = PJ_t - \left(\sum_{i=1,j=i+1}^{n-1,n} (WF_i \cup WF_j) + \sum_{p=1,q=p+1}^{m-1,m} (AE_p \cup AE_q) \right)$$

Where, AE$_j$ represents durations of access that are external to workflows that contain the document.

The Idle Time Factor may then be calculated as

$$Idle\ Time\ Factor = 1 - \left(\frac{Actual\ Idle\ Time}{Expected\ Idle\ Time}\right)$$

Which reduces to:

$$Idle\ Time\ Factor\ = \frac{\sum_{p=1,q=p+1}^{m-1,m}(AE_p \cup AE_q)}{PJ_t - \sum_{i=1,j=i+1}^{n-1,n}(WF_i \cup WF_j)}$$

The purpose of subtracting the ratio of actual idle time to expected idle time from 1, is to ensure that increasing values of the factor reflect importance of decreased idle times, as the ratio will always be less than 1 (since the actual idle time will always be either equal to or lower than the expected idle time).

6.1.3 Average Access of a Document

Related to the idle time of a document is the access of a document. While the change of state of a document from idle to active implies access to the document, there are several additional facets of document access that make incorporating it as an additional distinct factor for identifying critically problematic documents essential. For example, when a document's state changes, a record is made noting that the document has been accessed outside an expected workflow duration, but does not take into account who has accessed the document, why they have accessed it or how they accessed it.

An EPPM system allows users to access a document through a variety of ways. For example, in Coreworx's system, formal access is in the form of a transmittal workflow, but users may also use intuitive search pages to locate a document by keyword or meta-data filtering and save customized searches based on a set of filtering criteria for quicker access. If a user finds a particular document useful, there is also an option to "favorite" or "bookmark" it for easy and frequent access. Similar to the bookmarking of a document, is the document cart functionality which provides users with the ability to mark a set of interesting documents and then “check” them out, or download them to their personal computer.

Understanding how a document has been accessed from the above described available methods can also help in understanding why a document was accessed, or at least indicate the importance of document to a particular user. Favorites and checked out documents indicate that a particular document might be useful as a frequent reference. Further extrapolations about the importance of a document to a particular process can be made by creating associations to when it is accessed by multiple users working on the same project. Saved searches based on limiting filter criteria could indicate that a set of documents related to a process need to be checked periodically. The relevance of a document may thus be established by analyzing how it was accessed.

An ideal EPPM system would have an inbuilt capability to track when a document is accessed and who accesses it, as noted in the preceding section. Also, the number of times a particular document is accessed by different parties may be obtained from workflow logs. Determining how the document has been accessed can be achieved by keeping records of the specifically built uniform resource locators [url] that led to document access. While the number of times a document has been accessed can be directly taken from database records, when a document is downloaded to a participant’s personal computer, there is no longer any capability to measure how often the participant has accessed the document. In all cases, it is assumed that we are able to glean the document’s importance based on the logs and meta-data describing its interaction on the EPPM system.

Representing the number of instances the document has been accessed might provide a useful reference index, but the duration of time that the document was in an active state is an acceptable criteria upon which to scale different durations of time that the document might have been accessed for. It is important however to include every access instance of the document and to treat it independent of other accesses so as to arrive at accurate average access duration of the document. This includes both accesses outside of workflow activities and those comprised within the functioning of a workflow. The average access of a document may then be calculated over a time, ‘t’ of the project duration as described in the following equation:

$$Average\ Access\ Ratio = \frac{\left(\sum_{i=0}^{n} WF_i + \sum_{j=0}^{m} AE_j\right)}{PJ_t}$$

Where PJ_t represents the total project duration up to time ‘t and WF_i represents the duration of a workflow to which the document is attached, and ‘n’ represents the number of workflows that the document is attached to and , AE_j represents durations of access that are external to workflows that contain the document. As all accesses are treated independently, it is possible that in some cases the access duration of a document will exceed the project duration under consideration, resulting in an average access duration value that is greater than 1.

6.1.4 Document Revisions

It is natural for documents to undergo revisions over the course of a project. Changes to design or requirements, updating the completion of tasks, and reporting of on-site activities etc., all constitute updates made to documents. However it is crucial to ensure that only the most up-to-date or relevant document is accessed by a participant at all stages of a project. Using outdated requirements or specifications can have serious implications during the construction and execution stage of a project potentially resulting in costly delays and damages. Keeping track of version changes and ensuring that only the most recent accepted revision is accessed is therefore a vital requirement for an EPPMS. Older versions are usually preserved for auditing purposes, and therefore are usually accessed when there is sufficient reason to investigate the evolution of a document. The availability of older versions is also useful for maintaining references to activities that might not have been important enough to be recorded.

A key component of the Coreworx system is the underlying document management system [DMS].The DMS is responsible for maintaining revision history of documents. Every document or class of documents in Coreworx’s DMS has an associated “Document Profile” which contains relevant meta-data about the document, which may then be used to quickly locate the document and others in its class. The most recent version of a document is automatically appended to a workflow along with records of changes maintained in the “history tab”. New versions of documents are stored as separate associated files, with different nomenclature used to distinguish a revised version. For example, the first version of a file on implemented building specifications might be named Build_Specs_A01, and an updated version might be saved as Build_Specs_A02.

While the links to the associated files within the document management system are different, the document profile will contain near identical data about the revised version, since the vast majority, if not all, of the meta-data fields remain the same. This has serious implications for how versions might be accessed, since when a search is conducted based on filtering meta-data tags, all versions of a document will show up. Accidental access of the wrong version can easily occur if the sorting of search results isn’t optimal or if a user does not notice a newer version, perhaps for as simple a reason as not noticing pagination of search results.

A solution for this problem could be alerting the participant that they are accessing an older version of the document and providing a direct link to the most recent version. By default, this most recent version might be identified as the “standard” reference or version of the document that is meant to be accessed by all participants. There may be instances however wherein a newer version is deliberately not accessed by the participant, say if outside of the EPPM system, a group of participants have mutually agreed upon using a previous version for the completion of a task, but wish to preserve the newest version as a draft for the next stage of a project. Or perhaps a previous version contains a set of instructions or notes which might be applicable in similar repetitive activities in a project, but which was left out in the final version of the submitted document.

Hence there should be a mechanism that allows a user to override referencing the most recent version, and also to record that an override has occurred. An increasing number of overrides by multiple users, particularly during the same temporal span, then implies that the older version of a document is identified as a standard, if not for that particular process, at least as a reference for new processes. A user with appropriate responsibility may then be given a notification alerting to the status of the previous version – which also acts as an important check to note that participants are accessing the correct version – and may then officially classify the document as an acceptable standard reference.

Regardless of the whether the document being accessed is the standard reference version, all the versions essentially have the same meta-data information and this creates a situation where each version needs to be highlighted as a potentially problematic document. The magnitude of the potential error increases with the increase in the number of versions. If say there are 14 versions of a document and a search for the file brings up 10 instances in the first page, there is an increased probability in selection of an incorrect document version. All files which are not modified will obviously not share their criticality and can be disregarded from the analysis. However the degree of version changes needs to be emphasized, and hence the following equation was developed to highlight the increased inherent risk in documents with multiple versions relative to documents with multiple revisions:

$$Document\ Revision\ Factor = \left(\frac{number\ of\ revisions}{Maximum\ number\ of\ revisions\ for\ all\ documents}\right)^{\frac{1}{j}}$$

Where, the value of 'j' varies depending on the maximum number of revisions, estimated at 3 for less than 10 revisions and adjusted accordingly to account for larger values.

The graph below demonstrates how this variability increases depending on the number of versions within the document management systems, penalizing increasing revisions while scaling down lower version numbers:

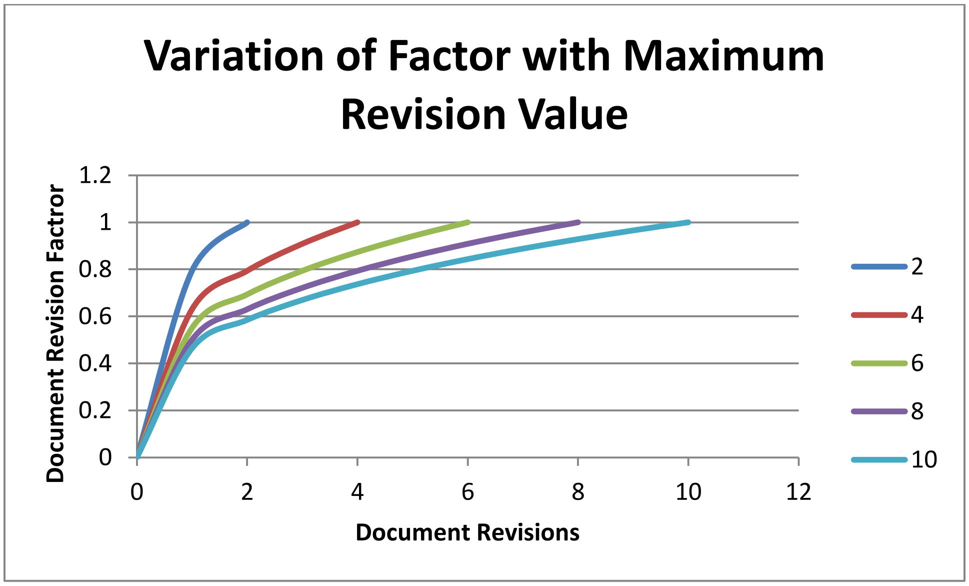


Figure 6.4: Variation of Factor for Varying Maximum Revision Values

6.1.5 Criterion for Determining Criticality of a Construction Document

The identified factors in the preceding section that impact the criticality of a document are not all encompassing. Rather, based on the available meta-data tied to each document profile and workflow history logs, they provide a basis upon which a separation may be made between documents that are more likely to have an impact on the project or be a risk to the project and those that will not. This in turn allows for early notification, remedial and risk mitigation measures to be put into operation. It should be noted however that not all documents will exhibit each factor. For example, it is quite possible that several documents will not be related to any other documents and it is more than likely that the vast majority of attached files will not be revised.

If there were only one criterion based on a weighted combination of these factors, the results could be misleading as a vast majority of the documents may not come into consideration simply because of the absence of a factor or two, even if the values of the other two factors favor rating the document as critical. Each factor in isolation does provide a strong enough basis for a document to be regarded as critical. Again, there might be cases where all the factors individually have just sub-critical values, but since each factor exists, the combination of all the factors shows that the document exhibits several features that display criticality.

It is therefore proposed that the criticality be determined both on the basis of each factor individually as well as the combination of all factors. Care must be taken to ensure that when the factors are combined that proper normalization techniques are applied to ensure each factor compares favorably with another. Min-Max normalization was applied wherever appropriate to factors in the analysis conducted for this research project. The average of all the factors was then calculated, thereby implying that all factors were treated equally.

In the absence of verification data, determination of appropriate weighting criteria is heavily dependent on user intuition and may be largely subjective, particularly for simulated environments. However, for demonstrative purposes, weighting via some of the methods discussed in the literature review Section 2.8, has been conducted as an example of how such weighting may occur in practice. A questionnaire was created for industry and academic experts (the questionnaire is displayed in Appendix E) as the basis for ranked or rating factors used in the algorithm for weight determination.

It becomes important therefore to establish thresholds for each factor (including the combined factor) which separates the status of a document from critical to non-critical. Given that EPPM systems are relatively new and that the importance of construction documents has not been examined by this approach, determining these threshold values scientifically is not feasible. Classification mechanisms do not exist simply because there has been no ranking method for the criticality of documents in an information system to test the suitability against verified critical documents. Expert intuition will be required in determining when a document crosses over from the realm of being sub-critical to critical, and this will most likely be the case on a project-to-project basis given the unique and unpredictable nature of construction projects.

6.2 UW Engineering 6 Building Simulated Model Project

In order to validate the utility of the above methodology, it is necessary to test the factors with data, which would ideally originate from a real construction project. As mentioned in Chapter 3, obtaining data was a major challenge for this research initiative primarily because the owners of project data were clients of the project sponsors who by and large expressed reluctance at sharing project data. In fact, one of the reasons for the initial foray into critical determination from meta-data rather than semantic interpretation of document content was based on early discussions with a leading EPC company who were adamant that data if shared would not include document content. The data that was eventually made available to this project unfortunately did not meet the requirements to carry out analysis.

Two of the four factors (or three out of five, if one considers the combined factor) rely heavily on file access data. This access data relates to every single time a file has been accessed and the duration of the access, at the very least. More specific data relating to the mode of access, the user or at least the role of the user who accessed the data etc. may have resulted in more exact analyses. The data from the British Columbia project however only recorded the last time a file was modified. This is a limitation of current EPPM implementations that it is hoped will be addressed in the future. Further usage statistics from user logs were not recorded, as these largely were believed to be available from universal resource locator logs which would include time-stamps, but this data had been over-written on server logs after the completion of the first stage of the project.

It may have been possible to simulate the access of files and incorporate this into the available data but a further examination revealed that of the 7000 files in the system, only 52 relationships were explicitly stored in the database and these covered only a handful of document pairs. A possible reason for the low number of relationships maintained internally is that the feature was not taken advantage of by the client during the first stage of the project as they were acclimatizing to using an EPPM system for the first time. Simulation based on partial and non-existent data for three of the four factors did not seem practical. Therefore there was a need to develop an alternate model project upon which the methodology could be tested.

6.2.1 Alternate Model Project Development

In developing an alternate scenario, it was important to ensure that the model project conformed to a real world scenario as much as possible. This scenario has typically been termed a “model project” in past studies (Youngsoo and Sungkwon , 2004), and most CPM algorithm validation over the last few decades has occurred on “model schedules” (Hegazy, et al., 2004), so while not ideal, it was judged to be an acceptable validation approach. This would include a realistic model project based on actual events that had taken place, real world data (as available) from the project in question, simulated data based on actual data for similar projects and the development of the model project on an EPPM system so as to simulate to the maximum extent possible how the project would have occurred had such a system been in place.

The University of Waterloo’s Engineering 6 Building is a $42 million, 115,000 sq. ft. building that was constructed in 2011. Previous research initiatives at the University of Waterloo involving mapping the building over different construction phases using laser scans were completed by this time. Fortunately, real world data pertaining to this construction project was made available in the form of building drawings and specifications documents. Building drawings and specification are an important subset of the kinds of documents that are usually attached to workflows and were deemed to be sufficient for the purposes of identifying critical documents within a simulated environment.

Coreworx Inc. facilitated development of the project on one of their EPPM system implementations, employing RFIs, Project Information Control and Workflow component products. Therefore all the project's RFIs were created on the Coreworx system employing the standard out-of-the-box RFI workflow template, as they would have been for a real project. This would permit validation of analysis methods for quantifying relationships between documents and generation of different versions of documents.

The one major difference between a real world project and the developed model project is the time in which the project would be developed on the system. As the RFIs were all inputted into the system in the space of a few weeks, extracting work-item times from the system's database would provide misleading durations for the project, which was assumed to have taken place over a duration of 8 months. Therefore work-item durations were generated randomly based on the work-item time distributions developed in Chapter 3.

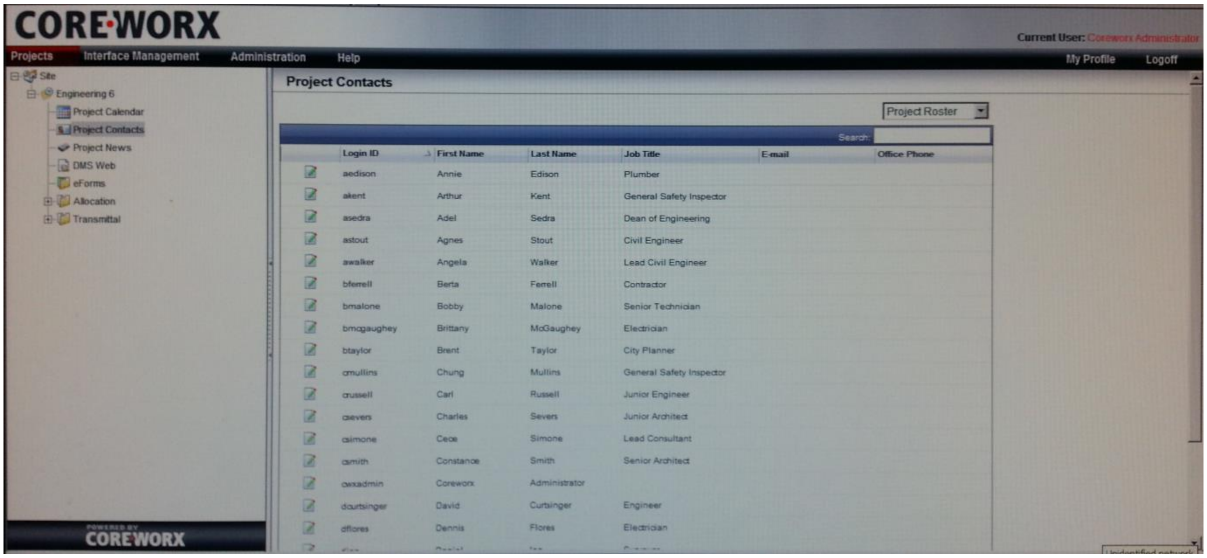
The version of the Coreworx Inc. EPPM implementation used for the development of the model project did not measure every instance a file was accessed nor the duration of an access. Therefore these values were generated for each simulated run randomly based upon an exponential function with random values of λ per simulated run. Finally a scenario was created with the model project and based on a real life event that required changes to be made. This scenario is explained in the next section.

6.2.2 Model Project Description

The developed model project begins in the middle of the model construction project, and is initiated by a change request made by the client (the University of Waterloo). The change request involved changes in a user-facing portion of the HVAC system. This prompted various other changes to the HVAC system, as well as other systems within the building, in order to accommodate user-facing changes.

The model project begins with a request for information from the University to see if the desired change is possible. A change request is then initiated and all of the sub-contractors and hired consulting firms are notified of the change. This then prompts a flurry of communication between various sub-contractors and the general contractor. As the problem becomes more apparent, it is noted that changes made to one floor are propagated and affect the HVAC systems on all of the other floors. After the changes to all of the HVAC system have been confirmed to all parties, the other systems that are also affected and which will require modifications are addressed by the construction participants.

The first step of the model project creation involved creating three main data tables. The first was a list of all relevant parties to the project. This table was used to import the workgroups and contracting parties into the EPPM system as users. The table consisted of 20 companies, each with between three and six employees involved with the electronic product and process management system, for a total of 90 people to be entered into the Coreworx system.



Project Contacts

Login ID	First Name	Last Name	Job Title	E-mail	Office Phone
aedison	Annie	Edison	Plumber		
akent	Arthur	Kent	General Safety Inspector		
asedra	Adel	Sedra	Dean of Engineering		
astout	Agnes	Stout	Civil Engineer		
awalker	Angela	Walker	Lead Civil Engineer		
bferrell	Berta	Ferrell	Contractor		
bmalone	Bobby	Malone	Senior Technician		
bmcgaughey	Brittany	McGaughey	Electrician		
btaylor	Brent	Taylor	City Planner		
cmullins	Chung	Mullins	General Safety Inspector		
crussell	Carl	Russell	Junior Engineer		
csevers	Charles	Severs	Junior Architect		
csimone	Cece	Simone	Lead Consultant		
csmith	Constance	Smith	Senior Architect		
cwxadmin	Coreworx	Administrator			
dcurtsinger	David	Curtsinger	Engineer		
dflores	Dennis	Flores	Electrician		

Figure 6.5: List of Users Created for Model project on Coreworx System

The second table contained all of the files that were relevant to the model project (and the construction project in general), that would be uploaded onto the Document Management System (DMS). In all, 214 files were created each representing an actual drawing or specification file from the data obtained from AECON Inc. As the content of these files would not affect the algorithm, almost all the files apart from two for demonstration purposes that were developed for uploading were dummy or blank files representing real drawings and specifications in our possession. However all relevant meta-data about the files was included in the relevant fields on the Coreworx system.

Finally, a list of 200 RFIs related to this model project was created. These RFIs were created referencing real world examples obtained from the British Columbia project that was provided to Coreworx Inc. The RFIs were developed in two stages; (i) 80 core RFIs that were directly related to the changes required because of the request to modify the HVAC system and an additional 120 RFIs were added that were likely to occur in parallel in the normal course of project events while the change and RFIs related to the HVAC change were taking place. Throughout the model project, other RFIs are issued for varying reasons, as is expected during any construction project.

6.2.3 Model Project Components

The model project consisted of several different project components, each of which had to be uploaded on to the Coreworx system in order for the project to be initialized prior to the generation of RFIs from the workflow template. The creation of some components were fictitious in nature, such as the names of personnel and companies, however in all cases adherence to a real world scenario was maintained by studying real RFIs created for the British Columbia project and with guidance from experienced consultants at Coreworx Inc. Other components such as the names of files and the workflow items were based upon real-world data and implemented workflow templates.

Each of the participant companies were created on the system along with the users who belonged to those firms. A “Responsible, Accountable, Consulted and Informed (RACI),” matrix representing the permissions to which users were entitled was created for the subset of users from the contracting parties who would be interacting with the system. This was to simulate to the maximum extent possible how the work-items would have been completed, so that users who did not have access to certain files would not be able to see or change them within the RFI.

The 218 files that were created were uploaded to the system via a bulk one-time upload at the start of the project. As the access durations would be generated randomly this would not affect the simulation of the system. The Coreworx system allows for tagging (or attaching) a file to an RFI from the list of files that have been uploaded by this process. Each of the files was renamed to reflect the naming convention adopted when a project is created on the Coreworx system. This feature was useful when creating files which consisted of multiple versions or revisions.

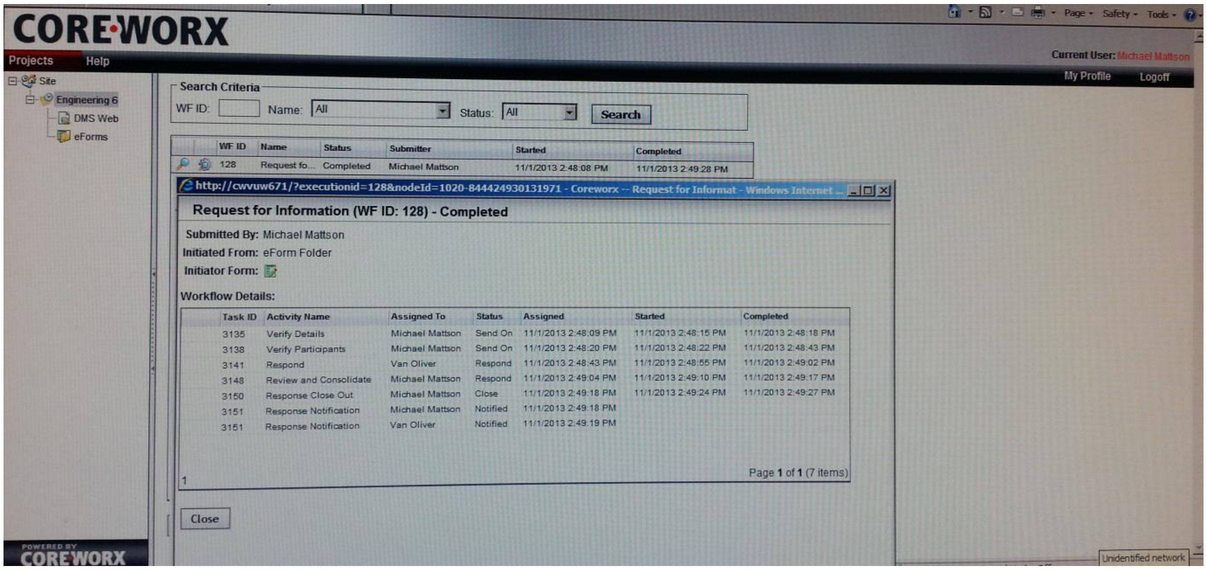
The most important component of the system was the RFI workflow template, that has been shown earlier in this thesis in section 3.3. The RFIs created in the list described in section 6.2.2 had to be generated on to the Coreworx system, and each RFI was created with strict adherence to the method with which it would have been created if it had been created on a real project.

6.2.3.1 Process of Generating RFIs

Referencing the list of 200 RFIs, the designated “initiator” account for a given RFI was logged into. The RFI was then created, filling in the title, description, requesting party (which company that person worked for) and a discipline (to feed into the RACI matrix and allow for auto population of the responder & consolidator roles). Any documents required were linked to through the DMS system, and the RFI was submitted.

The initiator user was then logged out, and the coordinator account was logged in to the system. At this point, the coordinator has the option to “Verify Details”, and then either moves forward the RFI to the “Verify Participants” step, or “Respond” directly. A direct response included a text response as well as attaching any documents necessary. Approximately 25% - 35% of the RFIs were responded to directly. If this option was chosen, the next step was for the initiator to “Close” the workflow. When the path required moving forward the RFI, the “Verify Participants” step was activated.

The “Verify Participants” step was carried out by the same user, the coordinator for that RFI. This step typically contained auto populated fields for responders and consolidators, which the Coreworx system retrieved from the RACI Matrix and the workgroups. These fields were reviewed and modified as necessary. Typically one user was selected to act as a responder and one for a consolidator, and then the “Verify Participants” work-item was closed. Both the responder and the consolidator were noted, as those accounts would need to get accessed. After the participants were verified, the coordinator logged out.



Search Criteria

WF ID: Name: All Status: All

	WF ID	Name	Status	Submitter	Started	Completed
	128	Request fo...	Completed	Michael Mattson	11/1/2013 2:48:08 PM	11/1/2013 2:49:28 PM

http://cwvuw671/?executionid=128&nodeId=1020-844424930131971 - Coreworx -- Request for Informat - Windows Internet ...

Request for Information (WF ID: 128) - Completed

Submitted By: Michael Mattson
Initiated From: eForm Folder
Initiator Form:

Workflow Details:

Task ID	Activity Name	Assigned To	Status	Assigned	Started	Completed
3135	Verify Details	Michael Mattson	Send On	11/1/2013 2:48:09 PM	11/1/2013 2:48:15 PM	11/1/2013 2:48:18 PM
3138	Verify Participants	Michael Mattson	Send On	11/1/2013 2:48:20 PM	11/1/2013 2:48:22 PM	11/1/2013 2:48:43 PM
3141	Respond	Van Oliver	Respond	11/1/2013 2:48:43 PM	11/1/2013 2:48:55 PM	11/1/2013 2:49:02 PM
3148	Review and Consolidate	Michael Mattson	Respond	11/1/2013 2:49:04 PM	11/1/2013 2:49:10 PM	11/1/2013 2:49:17 PM
3150	Response Close Out	Michael Mattson	Close	11/1/2013 2:49:18 PM	11/1/2013 2:49:24 PM	11/1/2013 2:49:27 PM
3151	Response Notification	Michael Mattson	Notified	11/1/2013 2:49:18 PM		
3151	Response Notification	Van Oliver	Notified	11/1/2013 2:49:19 PM		

1

Page 1 of 1 (7 items)

Figure 6.6 An Example of a Completed RFI in the E6 Building Model project on the Coreworx System

The responder, as noted, then logged in and accessed the RFI. A written response to the RFI was provided, and any documents that were deemed necessary supplicants were attached through the DMS system. The respond command was then executed and this user logged out. If more than one responder was listed, this same process was followed for all responders.

The consolidator account was then used to log in and access the RFI. The consolidator was able to see all responses, selected the best text response and inserted it into the final response text box. If there were any files attached incorrectly, or missing, this step would allow for that to be rectified, although that was not utilized during the simulation. Once the final response was copied, the consolidator would submit the data and then log out of the system.

Finally, the user that initiated the workflow would be logged back in, and they could see the completed RFI. At this stage it could be reviewed by the user, and closed out to verify that they had seen the response. All RFIs were closed using this process terminating the workflow instance. This allowed for the work-item related logs to be saved in the database as would be expected for a real construction project.

In addition, relationships between documents could be explicitly defined in the “relations tab” of the document profile. The one component that could not be transferred “as is” from this system was the time stamps for the project, as these were inputted directly from the list developed prior to the EPPM system initialization and merely involved copying and pasting relevant text. These work-item durations and the other factors that were required for a simulation are described in the next section.

6.2.4 Simulated Environment

The inputted real world data were insufficient for the determination of all the defined critical factors. Data would have to be generated to appropriately represent real data or provide a suitable reference sample for analysis. Most of this data would have to be generated randomly, particularly in the case of accesses outside of the workflow, which were not logged on the Coreworx system implementation used for this model project. The method of choice for analysis would therefore be a simulation, where each run would give an output for individual factors and the combined factor.

Not all the inputs are random however, as some of these were modeled on real world data as described in the previous section. For example, as the relationships had been explicitly defined in the workflows for all the attached documents, there was no requirement to randomize relationships. The centrality factor therefore remained the same irrespective of the simulation. Figure 6.7 on the following page displays the degree -centrality based network graph that was developed for this simulation. The thickness of the connections indicates the number of relationships between a pair of documents. (Higher resolution breakdowns of this figure are available in Appendix D).

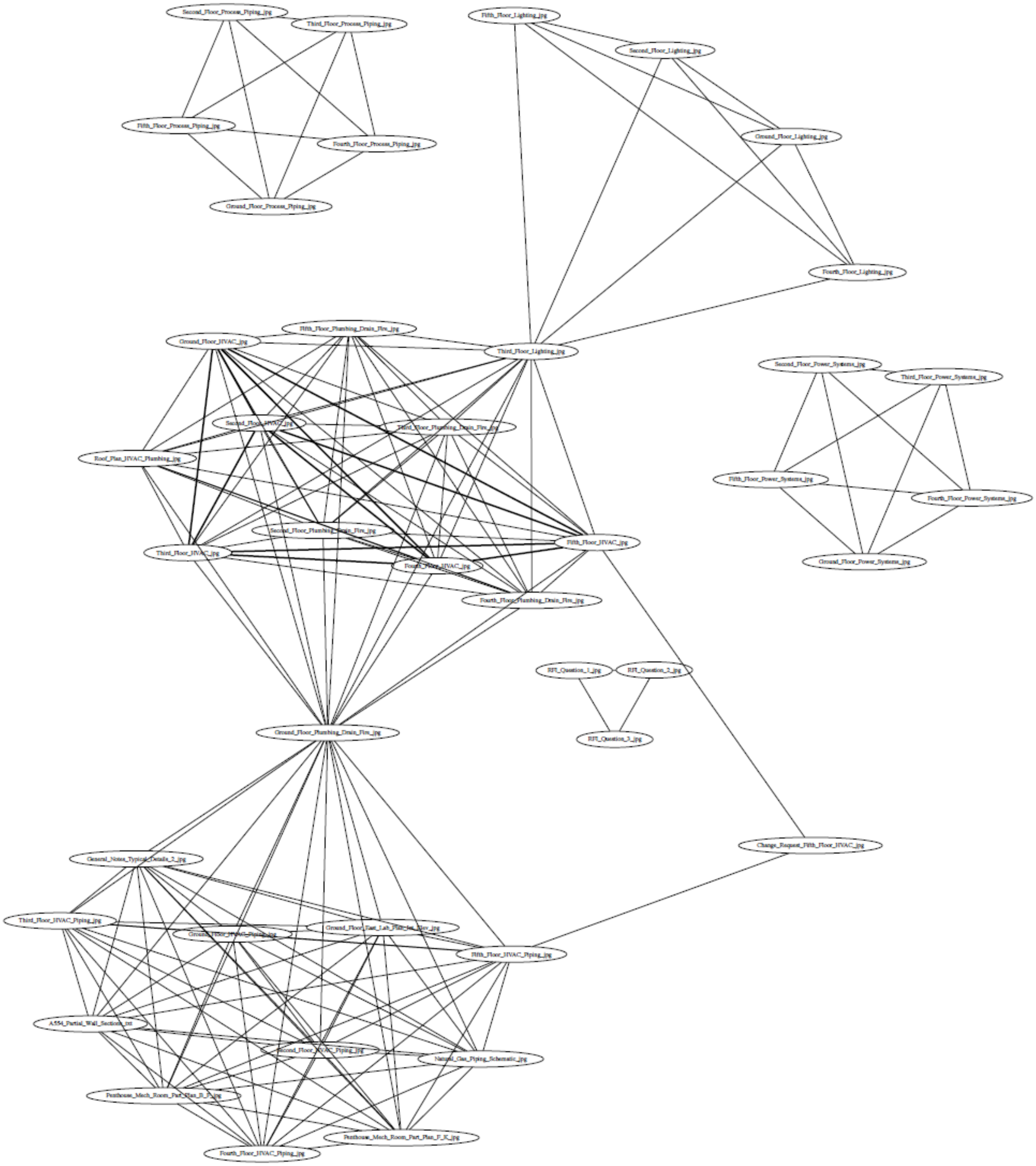

Second_Floor_Process_Piping_jpg
Third_Floor_Process_Piping_jpg
Fifth_Floor_Process_Piping_jpg
Fourth_Floor_Process_Piping_jpg
Ground_Floor_Process_Piping_jpg
Fifth_Floor_Lighting_jpg
Second_Floor_Lighting_jpg
Ground_Floor_Lighting_jpg
Fourth_Floor_Lighting_jpg
Ground_Floor_HVAC_jpg
Fifth_Floor_Plumbing_Drain_Fire_jpg
Third_Floor_Lighting_jpg
Second_Floor_HVAC_jpg
Third_Floor_Plumbing_Drain_Fire_jpg
Roof_Plan_HVAC_Plumbing_jpg
Second_Floor_Plumbing_Drain_Fire_jpg
Third_Floor_HVAC_jpg
Fifth_Floor_HVAC_jpg
Fourth_Floor_HVAC_jpg
Fourth_Floor_Plumbing_Drain_Fire_jpg
Second_Floor_Power_Systems_jpg
Third_Floor_Power_Systems_jpg
Fifth_Floor_Power_Systems_jpg
Fourth_Floor_Power_Systems_jpg
Ground_Floor_Power_Systems_jpg
RFI_Question_1_jpg
RFI_Question_2_jpg
RFI_Question_3_jpg
Ground_Floor_Plumbing_Drain_Fire_jpg
Change_Request_Fifth_Floor_HVAC_jpg
General_Notes_Typical_Details_2_jpg
Third_Floor_HVAC_Piping_jpg
Ground_Floor_HVAC_Piping_jpg
Ground_Floor_East_Lab_Plan_for_Elev_jpg
Fifth_Floor_HVAC_Piping_jpg
A554_Partial_Wall_Sections_txt
Second_Floor_HVAC_Piping_jpg
Natural_Gas_Piping_Schematic_jpg
Penthouse_Mech_Room_Part_Plan_B_F_jpg
Penthouse_Mech_Room_Part_Plan_F_K_jpg
Fourth_Floor_HVAC_Piping_jpg

Figure 6.7: Document Network Graph Developed From Centrality Algorithm

This was also the case for the file versions, as a finite set of files was uploaded to the EPPM system and it was important to keep track of these files and maintain homogeneity for each simulation. The remaining inputs related to the idle time ratio and the average access.

For the idle time ratio, both the times that the document was within a workflow and the time it was accessed outside of all workflows had to be generated. Since base data relating to a construction project was developed for the exactly similar and corresponding work-items in Chapter 4, this method of generating durations for work-items was adopted for the workflow durations. These times were however randomized prior to the simulation using random probabilities of occurrence for the distributions and parameters. Since all the workflow related data such as the relations were not randomized for every run, the randomly generated work-item times were generated only once and then used consistently. By varying only the access external to the workflows for each simulation run, it becomes possible to understand the influence of these accesses on the criticality of the document.

The accesses outside of the workflows were modeled using an exponential distribution. The underlying assumption made here is that accesses tend to follow a Poisson distribution, such that each access occurs independently and continuously at a constant rate. The time between accesses was modelled around an exponential distribution with values of λ randomized at the start of each simulated run. The memory-less factor λ was obtained randomly from a uniform distribution with a mean of 1 hour and a standard deviation of 30 minutes.

Each of the factors was calculated as described in Section 6.1. The simulation was written in the programming language “python” and executed on the UNIX Bourne-Again Shell (bash). For the combined factor, min-max normalization was applied as described in the following equation to ensure that all factors returned output within the same range [0,1]:

$$Normalized\ (x_i) = \frac{x_i\ - x_{min}}{x_{max} - x_{min}}$$

Where, x_i represents the data point for that particular factor.

The source code for the algorithm and the bash shell scripts that were used to run the simulation are presented in Appendix C.

6.2.5 Thresholds for Individual and Combined Factors

As has been described in Section 6.1.5, the purpose of establishing thresholds is to delineate those documents that are critical as per the factor selected to those that are less likely to have a detrimental impact on the project. Establishing these thresholds without class labels that would otherwise form a discriminating basis would have to be achieved over an iterative trial and error process over several different construction projects, data for which was not available to this project.

For demonstrative purposes in the simulation, four different thresholds were selected and all the documents that crossed a particular threshold were saved per simulation run. The thresholds that were selected were 0.75, 0.83, 0.9 and 0.9999. The threshold that is selected for individual factors should be higher than the factor that is used for the combined threshold, in order to account for cases where all factors may be present but just under the critical threshold for an individual factor, given the equal weighting that was applied to the calculation of the combined factor. If the threshold for individual factors was assumed at 0.9, then for the combined factor the threshold would be 0.83, thus accounting for cases where the four cases were present together but within 92% of the threshold for each factor.

6.2.6 Simulation Output

In total, 1000 simulated runs were conducted. Each simulated run began with creating randomized values of λ for the exponential distribution for the access time intervals of each of the 218 files. These values of λ were stored in an output csv (comma separated value) file for purposes of checking the results to ensure accuracy. Table 6.1 below provides a snapshot of the output for a simulation run generating the value of λ for each document

Document Name	λ
A401_Building_Sections.txt	1.416851
A501_Wall_Sections.txt	0.663226
A502_Wall_Sections.txt	1.242594
A551_Partial_Wall_Sections.txt	1.34653
A552_Partial_Wall_Sections.txt	1.081718
A553_Partial_Wall_Sections.txt	0.670115
A554_Partial_Wall_Sections.txt	1.063348
Air_Schematic.txt	0.818119
Atrium_Plan_Section_Details.txt	0.729176
Beam_Elevations_1.txt	0.987918
Beam_Elevations_2.txt	0.935284
Building_Lighting_Specs.jpg	0.685466
Building_Mechanical_Specs.jpg	0.515787
Change_Request_Fifth_Floor_HVAC.jpg	1.158154
Curtain_Wall_Elevation.txt	1.340398
Details_1.txt	0.692923
…	…
…	…
…	…
West_East_Elevations.jpg	0.773006
West_Parking_Specs.jpg	0.585261
Womens_Washroom_Typical_Elev.jpg	1.40021

Table 6.1: Output of λ Values Assigned to Each Document for a Simulation Run

Once a suitable value of λ was established for each simulated run, it was possible to calculate all the access times. As mentioned in Section 6.2.4, each simulation run generated random access statistics which affected both the average access as well as the actual idle time calculations. The centrality and document revision factors remained constant throughout the simulation. For every simulated run, an output csv file was generated that displayed the values calculated for each factor (including the combined factor) for a particular document in that simulation. Table 6.2 on the following page provides a snapshot of this output obtained from a single simulated run:

Document Name	Idle Time Ratio	Average Access	Degree Centrality	Revisions	Combined
Fifth_Floor_Plumbing_Drain_Fire.jpg	0.583894	0.613079	0.691589	0.793701	2.850786
RFI_Question_1.jpg	0.968014	0.174814	0.018692	0.629961	2.087395
Ground_Floor-Power_Systems.jpg	0.797199	0.125099	0.037383	0.629961	1.885556
Email_from_Tom_Dean.jpg	0.786495	0.08752	0	0.629961	1.79989
Email_from_Professor.jpg	0.723575	0.063624	0	0.629961	1.713074
Details_2.txt	0.597736	0.051805	0	0.629961	1.575416
Typical_Curtain_Wall_Section_A611.jpg	0.943794	0.06875	0	0.629961	1.938419
Fifth_Floor_HVAC.jpg	0.736329	0.924653	1	0.793701	3.623206
Fourth_Floor_East_Lab_Int_Elev.jpg	0.849414	0.07548	0	0.629961	1.850769
Third_Floor_Process_Piping.jpg	0.738912	0.144707	0.037383	0.629961	1.846877
Second_Floor_HVAC.jpg	0.092792	0.940112	0.990654	0.90856	3.007703
Second_Floor_Plumbing_Drain_Fire.jpg	0.495279	0.797475	0.691589	1	2.984343
Section_And_Details_1.jpg	0.597736	0.046385	0	0.629961	1.569996
…	…	…	…	…	…
…	…	…	…	…	…
Details_And_Notes.txt	0.692115	0.067381	0	0.629961	1.685371
Site_Grading_Servicing_Plan.jpg	0.786495	0.068463	0	0.629961	1.780833
HVAC_Test_Results.jpg	0.949774	0.083103	0	0.629961	1.958752
West_East_Elevations.jpg	0.597736	0.050342	0	0.629961	1.573953
Fifth_Floor_West_Lab_Int_Elev.jpg	0.346058	0.026993	0	0.629961	1.298926
Ground_Floor_HVAC_Piping.jpg	0.24168	0.211073	0.102804	0.793701	1.517781

Table 6.2: Output of a Simulated Run

It may be noted that in the above output table that there are values of 1 for degree centrality as well as revision. This is due to the application of min-max normalization so as to allow for equal comparison for the combined factor, wherein the assumption in the absence of suitable validating data is that the weights shall be given equal consideration. The combined factor in this case was calculated by adding the values of the other 4 factors. Values of zero for degree centrality indicate that the document had no relationships to any other in the EPPM system.

At the completion of all 1000 runs, an aggregation script was run which collated the results from all the simulated runs per factor in terms of threshold for each of the 218 documents. Table 6.3 on the next page shows a selected portion of the output csv file with the aggregated results of the simulation.

Document Name	**Factor**	**<0.75**	**0.75 < x < 0.83**	**0.83 < x <0.9**	**0.9 <x <0.9999**
Fifth_Floor_Plumbing_Drain_Fire.jpg	Idle Time	660	178	102	60
Fifth_Floor_Plumbing_Drain_Fire.jpg	Average Access	1000	0	0	0
Fifth_Floor_Plumbing_Drain_Fire.jpg	Centrality	1000	0	0	0
Fifth_Floor_Plumbing_Drain_Fire.jpg	Revisions	0	1000	0	0
Fifth_Floor_Plumbing_Drain_Fire.jpg	Combined	1000	0	0	0
Ground_Floor-Power_Systems.jpg	Idle Time	617	201	123	59
Ground_Floor-Power_Systems.jpg	Average Access	1000	0	0	0
Ground_Floor-Power_Systems.jpg	Centrality	1000	0	0	0
Ground_Floor-Power_Systems.jpg	Revisions	1000	0	0	0
Ground_Floor-Power_Systems.jpg	Combined	1000	0	0	0
…	…	…	…	…	…
…	…	…	…	…	…
HVAC_Piping_Test_Results.jpg	Idle Time	629	216	102	53
HVAC_Piping_Test_Results.jpg	Average Access	1000	0	0	0
HVAC_Piping_Test_Results.jpg	Centrality	1000	0	0	0
HVAC_Piping_Test_Results.jpg	Revisions	1000	0	0	0
HVAC_Piping_Test_Results.jpg	Combined	1000	0	0	0
Ground_Floor_HVAC_Piping.jpg	Idle Time	612	199	122	67
Ground_Floor_HVAC_Piping.jpg	Average Access	1000	0	0	0
Ground_Floor_HVAC_Piping.jpg	Centrality	1000	0	0	0
Ground_Floor_HVAC_Piping.jpg	Revisions	0	1000	0	0
Ground_Floor_HVAC_Piping.jpg	Combined	1000	0	0	0

Table 6.3: Aggregated Output Over All 1000 Simulated Runs

The numbers in each of the threshold bins for each factor indicates the number of times document appeared in that threshold bin over the 1000 simulations. For example, the Ground floow HVAC Piping drawing appeared 612 times out of the 1000 simulations with a threshold value under 0.75 and

189 times with a threshold value greater than 0.9 for the Idle Time factor. Again it may be noted that since the centrality and revisions do not change in each simulated run, they appear in all 1000 runs with the same threshold.

6.2.7 Summary of Results

The output has been separated by factor and then sorted in descending order of the number of documents, from the largest to the smallest threshold. The following tables list the critical documents ranked from most critical to least critical found over the entire simulation for each factor:

6.2.7.1 Degree Centrality:

Document Name	Rank	Simulation Instances within Threshold Percentage Bin			
		<0.75	0.75<x<0.83	0.83<x<0.9	0.9 < x
Fifth_Floor_HVAC.jpg	1	0	0	0	1000
Second_Floor_HVAC.jpg	2	0	0	0	1000
Fourth_Floor_HVAC.jpg	3	0	0	0	1000
Ground_Floor_HVAC.jpg	4	0	0	0	1000
Third_Floor_HVAC.jpg	5	0	0	0	1000
Ground_Floor_Plumbing_Drain_Fire.jpg	6	0	1000	0	0
Fifth_Floor_Plumbing_Drain_Fire.jpg	7	1000	0	0	0
Ground_Floor-Power_Systems.jpg	8	1000	0	0	0
Email_from_Tom_Dean.jpg	9	1000	0	0	0
Email_from_Professor.jpg	10	1000	0	0	0
Details_2.txt	11	1000	0	0	0
Typical_Curtain_Wall_Section_A611.jpg	12	1000	0	0	0
…	…	…	…	…	…

Table 6.4: Critical Documents over Varying Thresholds for Degree Centrality

Given the variation of the degree centrality, the separation between documents that were connected to several other documents as compared to those that were connected to fewer nodes was pronounced as shown in Figure 6.4 above. Five documents were clearly extremely critical compared to the other 218. This is not very surprising when Figure 6.7 is examined in close detail. Degree centrality remained static over the course of the simulation, as document relationships had to be preserved based on the development of the scenario. It was largely included in the simulation so that it could be incorporated in the combined factor, though it’s individual impact can be significant to require risk mitigation.

6.2.7.2 Idle Time Ratio:

The idle time ratio was calculated based on a fixed project duration of 8 months. As described in Section 6.1.2, when the equation essentially reduces to the summation of the union of overlapping accesses divided by the difference between the project duration and the union of overlapping work-items that the document was in. The times for each work-item were determined based upon the distributions in Chapter 4 and were constant throughout the simulation. Over 1000 simulations, the idle time factor thresholds were distributed as one might expect for a randomized generation as displayed in Table 6.5 below:

Document Name	Rank	Simulation Instances within Threshold Percentage Bin			
		<0.75	0.75<x<0.83	0.83<x<0.9	0.9<x
Roof_Plan_HVAC_Plumbing.jpg	1	575	183	143	99
Ground_Floor_Plumbing_Drain_Fire.jpg	2	624	173	111	92
Fourth_Floor_HVAC.jpg	3	637	191	84	88
Fourth_Floor_Plumbing_Drain_Fire.jpg	4	609	184	122	85
Third_Floor_Plumbing_Drain_Fire.jpg	5	582	191	144	83
Second_Floor_HVAC_Piping.jpg	6	608	191	119	82
Stairs_Fire_Specs.jpg	7	593	192	135	80
Third_Floor-Lighting.jpg	8	601	181	139	79
Third_Floor_HVAC.jpg	9	627	181	115	77
Site_Plan_Electrical.jpg	10	625	187	111	77
Fifth_Floor_HVAC_Piping.jpg	11	603	202	121	74
…		…	…	…	…
…		…	…	…	…
Sprinkler_Test_Results.jpg	212	657	198	108	37
Sprinkler_Specs.jpg	213	670	194	101	35
Room_Finished_Schedule.jpg	214	620	219	127	34

Table 6.5: Critical Documents over Varying Thresholds for Idle Time Factor

6.2.7.3 Average Access

The average access factor was significantly different in that considered every single instance that a document was accessed, whether within the workflow or externally. Instances with overlapping durations were treated independently. Given the fact that some documents were attached to a workflow instance more often than others, this contributed significantly to the eventual score for that document. This would appear to explain the significant difference in results obtained for the average access times and the idle time times, since the workflow durations were more dominant and would appear to have occurred more frequently than the access statistics. Table 6.6 below provides a snapshot of this sorted output and separates files very clearly between thresholds based on the criticality of access registered for each document per simulated run.

Document Name	**Rank**	**Simulation Instances within Threshold Percentage Bin**			
		<0.75	**0.75<x<0.83**	**0.83<x<0.9**	**0.9<x**
Second_Floor_HVAC.jpg	1	0	0	0	1000
Roof_Plan_HVAC_Plumbing.jpg	2	0	0	0	1000
Fourth_Floor_HVAC.jpg	3	0	0	0	1000
Ground_Floor_HVAC.jpg	4	0	0	33	967
Fifth_Floor_HVAC.jpg	5	0	0	272	728
Third_Floor_HVAC.jpg	6	0	199	799	2
Second_Floor_Plumbing_Drain_Fire.jpg	7	0	553	447	0
Fifth_Floor_Plumbing_Drain_Fire.jpg	8	1000	0	0	0
Ground_Floor-Power_Systems.jpg	9	1000	0	0	0
…	…	…	…	…	…
…	…	…	…	…	…

Table 6.6: Critical Documents over Varying Thresholds for Average Access

6.2.7.4 Document Revisions

Table 6.7 on the next page shows that only five documents were deemed to be critical when the revisions factor was taken into consideration. This implies that there were relatively few documents with many versions. From Table 6.2 it is possible to estimate that the maximum revision for any file in the scenario was four revisions.

Document Name	Rank	Simulation Instances within Threshold Percentage Bin			
		<0.75	0.75<x<0.83	0.83<x<0.9	0.9<x
Second_Floor_Plumbing_Drain_Fire.jpg	1	0	0	0	1000
Second_Floor_HVAC.jpg	2	0	0	0	1000
Ground_Floor_Plumbing_Drain_Fire.jpg	3	0	0	0	1000
Roof_Plan_HVAC_Plumbing.jpg	4	0	0	0	1000
Third_Floor-Lighting.jpg	5	0	0	0	1000
Fifth_Floor_Plumbing_Drain_Fire.jpg	6	0	1000	0	0
Fifth_Floor_HVAC.jpg	7	0	1000	0	0
First_Floor_Outlets.jpg	8	0	1000	0	0
…		…	…	…	…
…		…	…	…	…
HVAC_Test_Results.jpg	212	1000	0	0	0
West_East_Elevations.jpg	213	1000	0	0	0
Fifth_Floor_West_Lab_Int_Elev.jpg	214	1000	0	0	0

Table 6.7: Critical Documents over Varying Thresholds for Document Revisions

The factor is designed specifically to take into consideration such cases when the maximum number of revisions is low. Ideally a document may be considered critical due to a high number of revisions when there are multiple versions that may lead to a previous version being mistaken for being the most current. Given that most searches return 5 documents at a time, files that at the very least consist of 5 versions should be treated cautiously.

6.2.7.5 Combined Equally Weighted Factors

The average of all the above factors, thereby implying equal weightage given to each factor in the absence of validating criterion for different weight distribution, resulted in the combined factor.

Document Name	Rank	Simulation Instances within Threshold Percentage Bin			
		<0.75	0.75<x<0.83	0.83<x<0.9	0.9<x
Second_Floor_Plumbing_Drain_Fire.jpg	1	278	471	251	0
Second_Floor_HVAC.jpg	2	259	494	247	0
Fourth_Floor_HVAC.jpg	3	682	318	0	0
Roof_Plan_HVAC_Plumbing.jpg	4	736	264	0	0
Ground_Floor_HVAC.jpg	5	749	251	0	0
Fifth_Floor_HVAC.jpg	6	824	176	0	0
Third_Floor_HVAC.jpg	7	914	86	0	0
Fifth_Floor_Plumbing_Drain_Fire.jpg	8	1000	0	0	0
Ground_Floor-Power_Systems.jpg	9	1000	0	0	0
…	…	…	…	…	…
…	…	…	…	…	…

Table 6.8: Critical Documents over Varying Thresholds for Combined Equally Weighted Factors

As can be noted from table 6.8 above, the combination of factors for each simulation never resulted in a breach of the 90% threshold. But as stated in Section 6.1.5, setting a slightly lower threshold is necessary for this factor, and the results indicate two files which would have been determined to be critical by this method.

Equally weighted criteria were selected in the absence of verification data for this simulation to avoid subjectivity in the adjudication of weights to each factor. If the system is to be deployed however, it would be prudent for weights to be determined either after analyzing an EPPM system after the conclusion of a project and matching critically selected documents with factors and arriving at suitable weights. If a more empirical approach is sought, a Delphi method may be conducted based on the questionnaire in Appendix E. If say, 5 project managing experts evaluate the form and provide their feedback, a second and then further successive rounds will focus on incremental changes based on converging weights for the factors.

If the Delphi method is not adopted, then feedback from the questionnaire may be implemented using any of the methods described in Section 2.8 of this thesis. The questionnaire was sent to a construction software vendor and an expert provided her input on rating the factors. Table 6.9 below shows the weights obtained by the Ratio method and Rank order centroid based on the expert's feedback:

Factor	Rank: 1 to 4 (where 1 is most important)	Relevance Rating	Weighting by Rank Order Centroid Method	Weighting by Ratio Method (normalized)
Degree Centrality	3	6	0.27083	0.2
Document Idle Time	4	3	0.0625	0.1
Document Access Statistics	1	7	0.52083	0.3
Document Versions	2	7	0.145833	0.4

Table 6.9: Illustrative weighting based on feedback of questionnaire

6.3 Discussion

The model project was based upon a real world scenario that occurred during the construction of the Engineering 6 Building at the University of Waterloo. A change to the HVAC layout on one of the floors initiated by a request from a faculty member resulted in floor plan changes that propagated throughout the building. As the scope of changes to the overall layout began to become apparent, and as contractors realized how these changes would affect their operations, a flurry of information exchange began to occur. Most of this exchange occurred in the form of requests for information and change requests, however neither the owner nor general contractor had implemented an EPPM system. Hence communication exchanges were typically made via email and were not captured or stored within a document management system. As a result, extracting relevant data in the event of an audit would have been a challenge. Informal exchange of information of this nature makes keeping track of the documents that were transferred between all concerned parties a near impossible task.

As such it is difficult to estimate which documents were most relevant to the changes and which documents posed the greatest risk to the project. The model project attempted to recreate this scenario and demonstrate the value of using an EPPMS to structure and capture all information exchanged. While the letter from the faculty member initiated all the changes, after the initial approval by the owner, it would not have been as critical a document during construction as, say, updated drawings for each floor.

Consider the case of a plumbing contractor having not received an updated drawing of the HVAC system, who upon arriving at the site is confused by the changes in layout. Typical inquiries would perhaps consist of an RFI to the general contractor asking if the changes had occurred along with a request for updated floor plans. The general contractor may then send an updated drawing, along with a request for cost estimates for changes that have to be made. We can expect that drawings for every floor with the proposed changes may be sent along with queries regarding whether the changes are possible within the allocated budget.

As multiple contractors across various disciplines begin communicating with the general contractor and various other parties, the information exchange can quickly become voluminous but there may be a few common themes for all these exchanges. As a result, we can expect that a small subset of documents are exchanged with high frequency between all parties. Consider if the changes cannot be made by one contractor and therefore an additional change has to be made, and the ripple effect this will have on all the other parties and future construction operations. Newer versions of documents such as drawings will have to be created, and these may or may not be sent to all relevant parties.

Again as a new contractor begins to work at the site, they may have the incorrect version and in the worst case scenario complete their work based on the old plan, which may require extensive and potentially expensive corrections and associated delays to the project schedule. Employing an EPPM system would not only ensure that an audit would take less time and cost less, but could result in avoiding the situation completely since the access of potentially problematic documents would immediately result in a notification being sent to a project manager for risk mitigation.

The model project was based around such situations and documents that were anticipated to receive the most attention were attached to RFIs with greater frequency than other documents used in the project. As the project required extensive changes to the HVAC system, it was natural to expect that these documents would undergo several revisions along with associated floor plan and elevation drawings. These documents were also more likely to be viewed by contractors more often at different phases of the project.

The results from the simulation tend to demonstrate that it was these documents that were most critical to the project in terms of the risk they posed based on the factors designed in this research project. While the simulation tended to focus with greater emphasis on varying the access of documents, the links between documents and versions remained constant, since the model project was developed upon the basis that the RFIs were static and stored within the database. However, the impact of the durations of the workflow instances containing these documents (as well as the fact that such documents were likely to be attached to multiple RFI workflow instances) was also important particularly when the average access factor was taken into consideration, and also played an important role in the determination of the idle time factor.

This explains why the same set of documents appeared consistently at the top of the list for all the four factors. The results of the simulation therefore correspond closely to an expected subset of documents that were likely to be transferred between all parties during the duration of the model project.

6.4 Conclusion

An algorithm was designed to determine documents that could adversely affect a construction project based on file logs and meta-data associated with each file stored in an EPPM system. The algorithm was tested on a scenario based on a real event that required substantial changes, over 1000 simulations to account for a lack of available data due to limitations of current EPPM implementations. Table 6.10 on the following page displays the files that appeared over thresholds that were selected for a set of factors for at least 50 of 1000 simulations (or at least 5% of the time).

The results indicate that 9 of a total of 218 files could be considered critical enough to be potentially have a detrimental effect on the project and in an ideal EPPM system would have been brought to the notice of an administrator to ensure that risk mitigation action may be taken as appropriate.

Document Name	Factor appears in 50 or more Simulations				
	Threshold = 0.9				Threshold=0.83
	Degree Centrality	Idle Time	Average Access	Revisions	Combined
Fifth_Floor_HVAC.jpg	✓	✓	✓		
Second_Floor_HVAC.jpg	✓	✓	✓	✓	✓
Fourth_Floor_HVAC.jpg	✓	✓	✓		
Ground_Floor_HVAC.jpg	✓	✓	✓		
Third_Floor_HVAC.jpg	✓	✓			
Second_Floor_Plumbing_Drain_Fire.jpg		✓	✓	✓	✓
Roof_Plan_HVAC_Plumbing.jpg		✓	✓	✓	
Third_Floor-Lighting.jpg		✓		✓	
Ground_Floor_Plumbing_Drain_Fire.jpg		✓		✓	

Table 6.10: Critical Documents appearing in 50 simulations or more

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

This thesis presented frameworks that diagnosed construction workflows and documents based on inherent data retrieved from execution history and logs. Diagnosis, as a knowledge construct, was used to aid compliance checks of executed workflow instances, by developing criteria upon which executed instances could be compared to established models to see if they deviated from planned behavior. While workflows do ensure the compliance of the order of execution of work-items, the effect of extraneous factors on the compliance of constituent work-items has been difficult to identify and manage, and methodology to effectively determine potential non-compliant cases during an audit from a much smaller subset based on statistical foundations of confidence intervals was developed.

The design of workflows is a complex and difficult task, often relying on the experience and intuition of experts. The fragmented nature of the firms in the construction industry, together with the large volume of information that is generated and must be delivered in a timely manner between various parties during the execution of a project inhibits the collation and effective dissemination of knowledge. While an increasing number of large capital construction companies have begun to appreciate the benefits of using electronic product and process models to manage projects, the dynamic and often singular nature of projects brings unique challenges to designing workflows.

Often software workflow implementations are designed directly from paper-based work process definitions. As users become acquainted with workflows systems along with an increased visibility of work processes, the inherent advantages of using an automated workflow management system become clearer. As such they often seek to implement changes in the workflow to reflect their increased understanding of the work process itself, but implementing these changes can be time consuming and costly. A silent integrated intelligent algorithm that is able to detect behavioral pattern constructs and seamlessly adapt workflows to incorporate such behavior can save significant costs and time, while providing flexibility to a workflow instead of developing several different implementations.

The meta-data that is an essential component of the document management system for organization and retrieval of files, encapsulates information that can be exploited to determine the criticality of a document. This provides a computationally less expensive monitoring alternative to semantic driven natural language processing algorithms. Workflow history and access logs further provide a basis to discriminate upon the criticality of a document based upon user interaction and deviation of expected behavior.

In summary, the following conclusions can be stated from this research undertaking:

1) There is sufficient information encoded within construction workflow histories and document meta-data that may be exploited for the development of knowledge constructs such as diagnosis

2) Diagnosis based knowledge may be used to discriminate between executed behavior and planned behavior, thereby providing a basis for early risk mitigation if significant deviations are detected

3) Analysis of workflow histories may also result in the storage and retrieval of emergent behavioral patterns in workflows thereby facilitating the implementation of self-adapting workflows

4) Diagnosis may be measured in documents as deviation in expected idle time, and together with other factors such as a document’s revision history and relationship to other documents in the

system, can be a powerful tool in determining the potential for that document to have disruptive effect on a construction project

5) Thus the knowledge extracted from workflows, besides assisting owners with business analysis tools, can facilitate real-time monitoring of workflows and documents

It is hoped that this research effort will lay the foundation for further efforts to analyze and capture tacit knowledge from workflows, which may then be used to identify key social constructs within a construction project while facilitating the injection of knowledge on a by-demand basis

7.2 Contributions

This research has made significant contributions to the following areas: (1) Augmenting the capabilities of construction information systems, (2) Effective determination of compliance of construction workflows, (3) The body of knowledge of automated process-oriented construction, and (4) Construction document classification. These contributions are described in brief below:

1) By tying together the evaluation of workflows with knowledge capture and diagnosis, this research project attempts to augment the capabilities of EPPM systems equipping them with silent, automatically evolving processes that occur in the background and do not require manual oversight or intervention. Construction information systems are able to exploit inherent knowledge encapsulated within database transaction logs and document management system meta-data to alert a supervisor about deviations from expected behavior. These additional features make EPPM systems more robust and primed for early risk mitigation of detected anomalies and as such can provide cost savings to owners and contractors.

2) Criteria have been established to efficiently determine a subset of constituent work-item instances that may have violated compliance extraneously. This builds upon research that has

been developed for the compliance of workflow templates and delves down to the work-item level. Further it has been demonstrated that posteriori knowledge from a construction project is an effective source for auditing future executing instances.

3) The adaptive framework provides significant contributions to the body of knowledge in the automation of process oriented construction, namely in considering the effect of resource based availability and establishing methodology for effective pattern detection, storage and case-based retrieval. Resource availability was expounded based upon the priority of a workflow, a user's concurrent tasks and the user history for work-item completion.

4) While most construction document classification research tends to focus on semantic interpretation of document content, this research established criteria to determine the potential detrimental effect of a construction document based upon the meta-data and transaction logs of the document, thereby classifying documents based upon their potential risk to a project. This allows computationally inexpensive monitoring for rapid assessment of a document's criticality to a construction project.

7.3 Limitations

The frameworks developed and tested in this system were influenced by expert intuition as there was a paucity of published literature for construction workflow and document analysis. The frameworks also depended heavily on a single EPPM implementation, and data from that were made available from a single construction project. The EPPM implementation was provided by Coreworx, who primarily cater to mega-capital construction projects. Therefore, the developed frameworks are restricted both by scale and scope of a project until similar EPPM implementations are designed for smaller construction undertakings.

Coreworx, as software vendors, were restricted to what data they could make available for this research and unless a client explicitly agreed to make data available, it could not be used for validating the frameworks in this project. Over the duration of this project only one EPC firm made data available thereby limiting all the analysis to one specific construction project. This severely constrained the research and led to the development of specific case studies for validation and the development of a demonstrable simulated scenario. It is suggested that for future work, these frameworks will have to be tested across multiple construction projects with real data before they can be adopted.

Requests for Information was the only class of workflows analyzed in this research as they constituted one of only two types of workflows with sufficient workflow instances for analysis, the other major class was related to Change Requests, which was analyzed in parallel by a fellow research colleague. Further, several methodologies such as adaptation based upon triggered clarification loops and factors such as criticality of a document based upon number of interface points etc., had to be discarded due to the paucity of suitable data. Therefore, the developed methodologies are not all encompassing, and there is definitely scope for advancement as more data becomes available.

7.4 Recommendations for Future Work

The research presented in this thesis resulted in the development of frameworks that diagnose construction workflows and documents based on posteriori knowledge inherent in EPPM systems. These approaches are novel to the construction industry as it begins to embrace workflow oriented systems for the management of construction projects. While some of the research lays the foundation for further exploration, some of the methodologies may be extended to make them more robust and inclusive of factors that were not available to this project due to data restrictions. Recommendations for future work are listed below:

1) As knowledge has been extracted from workflow histories for diagnosis and development of adapted workflow implementations, it may be possible to analyze this knowledge and the results of the adaptations can be fed into a continuous improvement cycle that is aimed at identifying industry standard workflows that are ideally suited to classes of construction activities.

2) The comparison of the BPEL integrated workflow implementation and the BPEL of the executed instance provides a check in the time dimension. Workflows however consist of several other dimensions which more robust models such as Petri Nets will be able accommodate for higher levels of analysis. The compliance framework may be extended to include these dimensions incorporating resource and case specific factors for the comparison as the workflow implementations vary in complexity

3) A general methodology has been developed for self-adapting workflows, and two specific cases were created based on observed patterns. It was suggested in this thesis that a vendor may provide a list of potential patterns to a client. Algorithms that specifically search for patterns such as clustering algorithms may also be developed so as to search for an even wider array of patterns that may not intuitively be apparent to a human. Therefore there is scope for additional research to aid vendors with the determination of patterns based on data-mining techniques.

4) Depending on available data, several factors may be incorporated to improve the detection of critical documents in a construction EPPM system. If the system consists of modules incorporating other communication protocols such as interface management, then additional

factors such as criticality based upon the number of critical interface points that a document is connected to may be added to the algorithm.

5) The methodologies require extensive and rigorous testing with preferably labelled result data before they can be adopted or implemented. For example, if a construction firm with an EPPM system independently created a list of documents that were observed for critical impact, then the algorithm could be run and the results compared with the list for validation purposes

6) The weighting of the factors for the combined factor in the critical documents could not be determined based on the simulated data output. Machine learning algorithms for determining weights such as neural networks may be developed after the validation of the algorithm can be achieved as described in the previous step

7) Studies may be carried out on the frameworks for other EPPM systems, a series of mega capital construction projects and for projects of varying scale and scope to determine the applicability and conduct comparability assessments of the frameworks across the construction industry.

8) As newer classes of construction workflows emerge and are assimilated within EPPM systems, it may be possible to identify and associate dimensions other than time to individual work items. This will facilitate compliance checks based on criteria along other dimensions such as cost, schedule etc., which will enhance and enrich the compliance checking methodology.

Bibliography

Ackoff, R. L. “From Data to Wisdom.” Journal of Applied Systems Analysis, 16, 3-9, 1989

Agrawal, R., Johnson, C., Kiernan, J., Leymann, F.,”Taming Compliance with Sarbanes-Oxley Internal Controls Using Database Technology”, In: ICDE 2006. Proceedings of the 22nd International Conference on Data Engineering, p. 92. IEEE Computer Society, Los Alamitos (2006)

Alavi, M and Leidner, D.E (2001) “Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues”, MIS Quarterly, Vol. 25, No. 1 (Mar., 2001), pp. 107-136

Argyris C., (1994) “Good communication that blocks learning”, Harvard Business Review, July-Aug 1994

Atluri, V., & Warner, J. (2005, June) “Supporting conditional delegation in secure workflow management systems”, Proceedings of the tenth ACM symposium on Access control models and technologies (pp. 49-58). ACM.

Barrett, M. J., Cason, S. P., D'andria, K. M., Gearing, M. W., Ho, K. K. T., Miller, H. E., ... & Woisard, E. (2000). U.S. Patent No. 6,029,144. Washington, DC: U.S. Patent and Trademark Office.

Barron, F. H., & Barrett, B. E. (1996). “Decision quality using ranked attribute weights”. Management Science, 42(11), 1515-1523.

Bharat, K. A., & Henzinger, M. R. (2000). U.S. Patent No. 6,112,203. Washington, DC: U.S. Patent and Trademark Office.

Bradshaw, J. M. (1997). Software agents. MIT press.

Bruggemann, B.M., Holz, K., Molkenthin, F., “Semantic documentation in engineering”, Proceedings of the ICCCBE VIII, Palo Alto, CA, ASCE, Reston, VA, August, 2000, pp. 828–835.

Brusoni, S., Prencipe, A., Salter, A., (1998) "Mapping and measuring innovation in project-based firms", CoPS Working Paper No. 46, SPRU, University of Sussex.

Burges, C.J.C., “A tutorial on support vector machines for pattern recognition, Data Mining and Knowledge Discovery” 2 (2) (1998) 121– 167.

Caldas, C.H. and Soibelman, L. (2003) “Automating hierarchical document classification for construction management information systems”, Automation in Construction 12, pp. 395–406.

Carrillo, P.M., Anumba, C.J. and Kamara, J.M. (2000), ‘‘Knowledge management for construction: key IT and contextual issues’’, in Gudnason, G. (Ed.), Proceedings of the Inter. Conf. on Construction IT, Icelandic Building Research Institute, Reykjavik, Iceland, 28-30 June, pp. 155-65.

Cerovsek, T. and Katranuschkov, P., “Active Process Reuse Model For Collaboration”, ITcon 2006, Vol 11, pp. 467-489

Chinowsky, P., Diekmann, J., and O’Brien, J., (2010) “Project Organizations as Social Networks,” Journal of Construction Engineering and Management, Vol. 136, no 4, pp 452-458.

Chung, P. W., Cheung, L. Y., & Machin, C. H. (2008). “Compliance Flow–Managing the compliance of dynamic and complex processes” Knowledge-Based Systems, 21(4), 332-354

Cohen, P., “Empirical Methods for Artificial Intelligence”, MIT Press, 1995.

Conroy, G., Soltan, H. (1998) "ConSERV, as a continual audit concept to provide traceability and accountability over the project life cycle" International Journal of Project Management Volume 16, Issue 3, pp 185-197

Construction Industry Institute (CII), 2007. "Effective Management Practices and Technologies for Lessons Learned Programs" Construction Industry Institute, Research Summary 230-1, The Univ. of Texas at Austin, Austin, Texas.

Crampton, J., & Khambhammettu, H. (2008, June). “Delegation and satisfiability in workflow systems” Proceedings of the 13th ACM symposium on Access control models and technologies (pp. 31-40). ACM

Dalkir K., “Knowledge Management in Theory and Practice”. Elsevier Butterworth-Heinemann., Burlington., 2005.

Dave, B., Koskela, L., (2009) “Collaborative knowledge management – A construction case study”, Automation in Construction, Vol 18, pp 894-902

Davenport, T. H., and Prusak, L. (1998), “Working Knowledge: How Organizations Manage What They Know”, Harvard Business School Press

Davies J., Duke A., et al, “Next generation knowledge access” Journal of Knowledge Management, Vol. 9 No. 5 2005, pp. 64-84

De Krester, S., Wilkinson, S., (2004) “Strategies for managing project generated knowledge: A New Zealand case study” Knowledge management in the construction, Idea group publishing pp 1-17

Deng, H. (1999). “Multicriteria analysis with fuzzy pairwise comparison”. International Journal of Approximate Reasoning, 21(3), 215-231

Duda, R. O., Hart P. E. and Stork, D. G., Pattern Classification (2nd ed.), John Wiley and Sons, 2001.

Dunham, M., “Data Mining Introductory and Advanced Topics”, Prentice Hall, 2003.

Dyer, J. S., Fishburn, P. C., Steuer, R. E., Wallenius, J., & Zionts, S. (1992). “Multiple criteria decision making, multiattribute utility theory: the next ten years”. Management science, 38(5), 645-654.

Egbu, C.O., and Botterill, C. (2002) "Information technologies for knowledge management: their usage and effectiveness", ITcon Vol. 7, Special Issue ICT for Knowledge Management in Construction , pp. 125-137

Ehrler, L., Fleurke, M., Purvis, M., & Savarimuthu, B. T. R. (2006). Agent-based workflow management systems (WfMSs). Information Systems and E-Business Management, 4(1), 5-23.

El-Diraby, T. E. (2006). “Infrastructure Development in the Knowledge City” Lecture Notes in Computer Science, v 4200 LNAI, Intelligent Computing in Engineering and Architecture - 13th EG-ICE Workshop, ASCONA, Switzerland: 175-185.

El-Diraby, T.A., Lima, C., Feis, B., (2005) “Domain taxonomy for construction concepts: Toward a formal ontology for construction knowledge”, Journal of Computing in Civil Engineering, Vol 19, No.4, ASCE pp. 394-406

El-Gohary, N. 2008. “Semantic Process Modelling and Integration for Collaborative Construction and Infrastructure Development”, PhD Thesis, Department of Civil Engineering, University of Toronto, Canada.

El-Gohary, N.M. and El-Diraby, T.E. (2010), “Dynamic Knowledge-Based Process Integration Portal for Collaborative Construction”, J. Constr. Engrg. and Mgmt. Volume 136, Issue 3, pp. 316-328.

Fayyad, U., Piatetsky-Shapiro G., and Smyth P., “From Data Mining to Knowledge Discovery in Databases” AI Magazine, Fall 1996 pp. 37-54

Fonseca, C. M., & Fleming, P. J. (1993, June). “Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization”. In ICGA (Vol. 93, pp. 416-423)

Ghaboussi, J., Garrett Jr, J. H., & Wu, X. (1991). “Knowledge-based modeling of material behavior with neural networks”. Journal of Engineering Mechanics, 117(1), 132-153

Grant, R. M., (1996) "Toward a knowledge based theory of a firm”, Strategic Management Journal Vol 17, pp 109-122

Green, S., Newcombe, R., Fernie, S. and Weller, S. (2004), "Learning across business sector: knowledge sharing between aerospace and construction", University of Reading, Reading.

Grobelnik M. and Mladenic D., “Automated knowledge discovery in advanced knowledge management” Journal of Knowledge Management, Vol. 9 No. 5 2005, pp. 132-149

Han, J., Kamber, M., (2001) “Data Mining: Concepts and techniques”, Academic press, U.S.A

Han, Y., Sheth, A., & Bussler, C. (1998, November). “A taxonomy of adaptive workflow management” , Workshop of the 1998 ACM Conference on Computer Supported Cooperative Work.

Hart, A. (1992) “Knowledge acquisition for expert systems” 2nd edition, McGraw Hill

Haveliwala, T. H. (2003). Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. Knowledge and Data Engineering, IEEE Transactions on, 15(4), 784-796

T. Hegazy, A. Elhakeem, E. Elbeltagi "Distributed scheduling model for infrastructure networks", Journal of Construction Engineering and Management, 130 (2) (2004), pp. 160–167

Hignett, T. (2014) “Capital Projects – A Whole New Ball Game”, http://www.coreworx.com/capital-projects-a-whole-new-ball-game-blog/

Jain, A. K. and Dubes, R.C., Algorithms for Clustering Data, Prentice Hall, 1988.

Jain, A.K., Robert, P.W. Mao, D.J., "Statistical Pattern Recognition: A Review," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 1, pp. 4-37, Jan. 2000, doi:10.1109/34.824819

Jensen, F. V. (1996). “An introduction to Bayesian networks” (Vol. 210). London: UCL press.

Jolliffe, I. (2005). “Principal component analysis”. John Wiley & Sons, Ltd.

Kamara, J. M., Augenbroe, G., Anumba, C. J., Carrillo, P.M., (2002) "Knowledge management in the architecture, engineering and construction industry", Construction Innovation: Information, Process, Management, Vol. 2 Iss: 1, pp.53 – 67

Kammer, P. J., Bolcer, G. A., Taylor, R. N., Hitomi, A. S., & Bergman, M. (2000). “Techniques for supporting dynamic and adaptive workflow”. Computer Supported Cooperative Work (CSCW), 9(3-4), 269-292

Katranuschkov, P., Gehre, A., Scherer, R.J., (2007) "Reusable Process Patterns for Collaborative Work Environments", Proceedings of the 13th International Conference on Concurrent Enterprising, Centre of Concurrent Enterprising, Nottingham, UK, pp 87-96

Kim, H., Rauch, A. F., & Haas, C. T. (2004). “Automated quality assessment of stone aggregates based on laser imaging and a neural network”. Journal of Computing in Civil Engineering, 18(1), 58-64

Kirsch, S. T. (1997). U.S. Patent No. 5,659,732. Washington, DC: U.S. Patent and Trademark Office

Kumar, A., & Liu, R. (2008). A rule-based framework using role patterns for business process compliance. In Rule Representation, Interchange and Reasoning on the Web (pp. 58-72). Springer Berlin Heidelberg.

Lam, A. (2000). “Tacit knowledge, organizational learning and societal institutions: An integrated framework” Organization Studies, 21(3), pp. 487–513

Laudon, K. C. & P. L. Laudon (1998) “Management Information Systems”, Fourth edition, Prentice-Hall, New Jersey, USA.

Lausen H, Ding Y, Stollberg M, Fensel D, Hernandez R.L., and Han S-K, "Semantic web portals: state-of-the-art survey" Journal of Knowledge Management Vol. 9 No. 5 2005, pp. 40-49

Leymann F, Roller D., (2000) "Production Workflow: Concepts and Techniques" Prentice Hall

Liao P.C., Thomas S.R., O'Brien W.J., Dai,J. Mulva, S.P., Kim, I. (2012) "Benchmarking Project Level Engineering Productivity", Journal of Civil Engineering and Management, Vol. 18, Iss. 2, 2012

Lim, H. W., Kerschbaum, F., & Wang, H. (2012). Workflow Signatures for Business Process Compliance. Dependable and Secure Computing, IEEE Transactions on, 9(5), 756-769.

Lui, Y., Müller, S., Xu, K. “A static compliance-checking framework for business process models”, IBM Systems Journal 46(2), 335–362 (2007)

Maedche, A., & Staab, S. (2001). “Ontology learning for the semantic web”. IEEE Intelligent systems, 16(2), 72-79.

Marsan M.A, Bobbio A., Donatelli S., “Petri Nets in Performance Analysis: An Introduction”, Lecture Notes in Computer Science, 1998, Volume 1491/1998, 211-256

Minor, M., Bergmann, R., Görg, S., & Walter, K. (2010). “Towards case-based adaptation of workflows”, Case-Based Reasoning Research and Development (pp. 421-435). Springer Berlin Heidelberg

Montana, D. J., & Davis, L. (1989, August). “Training Feedforward Neural Networks Using Genetic Algorithms.” In IJCAI (Vol. 89, pp. 762-767).

Müller, R., Greiner, U., & Rahm, E. (2004). “Agentwork: a workflow system supporting rule-based workflow adaptation” Data & Knowledge Engineering, 51(2), 223-256

Nasraoui, O., Krishnapuram, R., & Joshi, A. (1999, August). Mining web access logs using a fuzzy relational clustering algorithm based on a robust estimator. In Proc. the Eighth Int’l World Wide Web Conference, Toronto, Canada.

Nonaka, I., and Takeuchi, H. (1995) “The Knowledge- Creating Company: How Japanese Companies Create the Dynamics of Innovation”, Oxford University Press

Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web, Technical report, Stanford University, 1998

Pan, S.L., Scarborough, H., (1999) “Knowledge management in practice: an exploratory case study”, Technology Analysis and Strategic Management, Vol 11,No 3 pp 359-375

Pathirage, C., Amaratunga., D. G., Haigh, R.P., (2007) "Tacit knowledge and organizational performance: construction industry perspective", Journal of Knowledge Management Vol 11, No.1, pp 115-126

Pirolli, P. L., Pitkow, J. E., & Huberman, B. A. (2001). U.S. Patent No. 6,272,507. Washington, DC: U.S. Patent and Trademark Office

Polanyi, M. (1967) “The tacit dimension” Routledge and Kegan Paul

Prencipe, A., Tell, F., "Inter-project learning: processes and outcomes of knowledge codification in project-based firms" (2001) Research Policy, Volume 30, Issue 9, Pages 1373-1394

Rinderle, S., Reichert, M., & Dadam, P. (2004). “Correctness criteria for dynamic changes in workflow systems—a survey”. Data & Knowledge Engineering, 50(1), 9-34

Robinson, J. B. (1991). “Delphi methodology for economic impact assessment” Journal of transportation engineering, 117(3), 335-349

Roweis, S.T., and Saul, L.K, (2000) "Nonlinear Dimensionality Reduction by Locally Linear Embedding", Science 22 December 2000: 290 (5500), 2323-2326 DOI:10.1126/science.290.5500.2323]

Schapke, S.E., Menzel K., Scherer, R.J., “Towards Organisational Memory Systems in the Construction Industry, CISEMIC 2002

Shuzhou, L., & Soong, A. G. E. (2002). “A formal framework to support workflow adaptation.” International Journal of Software Engineering and Knowledge Engineering, 12(03), 245-267

Skibniewski, M.J., and Ghosh, M. (2009), “Determination of Key Performance Indicators with Enterprise Resource Planning Systems in Engineering Construction Firms”, J. Constr. Engrg. And Mgmt. Volume 135, Issue 10, pp. 965-978 (October 2009).

Soibelman L., Kim H., “Generating construction knowledge with knowledge discovery in databases”, Journal of Computing in Civil Engineering, vol. 16 (1), ASCE, 2002, pp. 39– 48.

Soibelman L., Wu, J., Caldas, C., Brilakis, I. and Lin, K.Y. (2008), “Management and analysis of unstructured construction data types”, Advanced Engineering Informatics 22 (2008), pp. 15–27.

Stohr, E. A., Zhao, J. L., (2001) “Workflow Automation: Overview and Research Issues”, Information Systems Frontiers 3:3, pp 281-296

Tang P., Akinci B., "Formalization of workflows for extracting bridge surveying goals from laser-scanned data", Automation in Construction (2011), doi:10.1016/j.autcon.2011.09.006

Teerajetgul, W., Chareonngam, C., (2008) "Tacit knowledge utilization in Thai construction projects", Journal of Knowledge Management Vol 12, No.1, pp 164-174

Tiwana, A. (2000) “The knowledge management toolkit – Practical techniques for building a knowledge management system” Englewood Cliffs, NJ: Prentice-Hall

van der Aalst, Wil MP, et al. "Adaptive workflow." Enterprise Information Systems. Springer Netherlands, 2000. 63-70

van der Aalst, W.M.P., van Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., Weijters, A.J.M.M, (2003), “Workflow Mining: A Survey of Issues and Approaches”, Data and Knowledge Engineering 47(2), 237–267

van der Aalst, WMP, “The Application of Petri Nets to Workflow Management”, Journal of Circuit Systems and Computers, 1998, Vol 8, pp. 21-66

van der Aalst, W. M. P. (2004). “Pi calculus versus Petri nets: Let us eat humble pie rather than further inflate the Pi hype”, 2004. DOI= http://is. tm. tue. nl/research/patterns/download/pihype. pdf.

Verheij H., Augenbroe G. (2006) "Collaborative planning of AEC projects and partnerships", Automation in Construction, 15 (4), pp. 428-437

Viswanadham N., Narahari Y., “Performance Modeling of Automated Manufacturing Systems”, Prentice-Hall Inc., 1992

von Krogh, G. "Care in Knowledge Creation, “California Management Review (40:3), 1998, pp. 133-153.

Vrandecic D., Pinto S., Tempich C., and Sure Y. “The DILIGENT knowledge process” Journal of Knowledge Management Vol. 9 No. 5 2005, pp. 85-96

Wainer, J., Kumar, A., & Barthelmess, P. (2007). DW-RBAC: “A formal security model of delegation and revocation in workflow systems”. Information Systems, 32(3), 365-384.

Wang, M. W., & Stanley, J. C. (1970). “Differential weighting: A review of methods and empirical studies” Review of Educational Research, 663-705.

Wang, M., Wang, H., & Xu, D. (2005). “The design of intelligent workflow monitoring with agent technology”. Knowledge-Based Systems, 18(6), 257-266

Wargitsch, C., Wewers, T., Theisinger, F. (1998) “An organizational memory-based approach for an evolutionary workflow management system” HICCS-31, pp. 174–183

Weijters, A.J.M.M., van der Aalst, W.M.P, (2004) “Workflow Mining: Discovering workflow models from event-based data”, in: C. Dousson, F. Höppner, R. Quiniou (Eds.), Proceedings of the ECAI Workshop on Knowledge Discovery and Spatial Data, 2002, pp. 78–84

WFMC [Hollingsworth, D.,] (1995) “The Workflow Management Coalition Specification”, (WFMC – TC00 – 1003), Issue 1.1

Whelton, M., Pennanen, A., Ballard, G. (2005) "Knowledge emergence and adaptive management: An exploration on the co-production of project needs and requirements by client-specialist groups" Knowledge management in the construction, Idea group publishing pp 251-275

White, R. W., Ruthven, I., & Jose, J. M. (2002, August). Finding relevant documents using top ranking sentences: an evaluation of two alternative schemes. In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 57-64). ACM.

Woo, J.H., Clayton, M.J., Johnson, R.E., Flores, B.E., Ellis C., “Dynamic Knowledge Map: Reusing Expert’s Tacit Knowledge in the AEC Industry”, Automation in Construction 2004, Volume 13, pp. 203-207

Youngsoo J., Sungkwon W., (2004). ”Flexible Work Breakdown Structure for Integrated Cost and Schedule Control.” J. Constr. Eng. Manage., 130(5), 616–625.

Zaiane, O. R., Xin, M., & Han, J. (1998, April). Discovering web access patterns and trends by applying OLAP and data mining technology on web logs. In Research and Technology Advances in Digital Libraries, 1998. ADL 98. Proceedings. IEEE International Forum on (pp. 19-29). IEEE

Zeb, J., and Froese, T., (2011) “Design and management of transactions in the AEC/FM industry using an ontological approach”, Proceedings of the 3rd International/ 9th Construction Specialty Conference, CSCE, Ottawa, Canada

Zhang, J. and El-Diraby, T. E. 2009. “SSWP: A Social Semantic Web Portal for Effective Communication in Construction”, Journal of Computers, Vol. 4(4), © 2009 Academy Publisher.

Zhao, J.L., “Knowledge Management and Organizational Learning in Workflow Systems”, Proceedings of AIS 1998

Y. Zhu, G. Augenbroe (2002), “Workflow process views”, Proceedings of the European Conference on Product and Process Modeling, University of Ljubljana, Slovenia, (September).

Appendix A-1

Time Distribution Data for Workflow Activities

RFI Workflow Data from British Columbia Construction project

Activity Name: Verify Details

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
RFI-CON-00045-1002	25	0.517	…	…	…
RFI-CON-00045-1002	25	0.030	…	…	…
RFI-CON-00045-1002	25	15.620	…	…	…
RFI-SO-00049-1001	31	0.035	RFI-CON-00040S-1307	1922	22.824
RFI-SO-00027-1001	32	1.026	RFI-CON-00040S-1308	1923	16.000
RFI-SO-00049-1002	33	0.029	RFI-CON-00040S-1309	1924	16.000
RFI-SO-00049-1003	34	15.868	RFI-CON-00040S-1310	1925	16.000
RFI-SO-00049-1004	37	21.851	RFI-CON-00040S-1311	1926	16.000
RFI-CON-00045-1003	38	2.757	RFI-CON-00040S-1312	1927	16.000
RFI-CON-00045-1004	39	2.729	RFI-CON-00040S-1313	1928	16.000
RFI-SO-00049-1005	40	0.101	RFI-CON-00040S-1314	1929	12.882
RFI-SO-00049-1006	47	3.043	RFI-CON-00040S-1314	1929	12.882
RFI-CON-00040S-1002	50	0.010	RFI-CON-00040S-1315	1930	16.000
RFI-SO-00049-1007	51	1.214	RFI-CON-00040S-1316	1931	16.000
RFI-SO-00049-1008	52	1.163	RFI-CON-00061-1007	1933	0.114
RFI-SO-00049-1009	55	7.124	RFI-CON-00061-1007	1933	0.114
RFI-CON-00032-1001	75	11.471	RFI-CON-00040S-1317	1936	11.232
RFI-CON-00032-1002	77	10.864	RFI-CON-00040S-1318	1937	0.720
RFI-SO-00049-1010	80	0.818	RFI-CON-00040S-1318	1937	0.720
RFI-SO-00049-1010	80	1.102	RFI-ECA-INT-1028	1941	0.280
RFI-SO-00049-1011	81	0.573	RFI-CON-00040S-1319	1950	8.691
RFI-SO-00049-1012	82	1.151	RFI-ECA-INT-1029	1953	20.272
RFI-SO-00049-1013	83	0.887	RFI-CON-00040S-1320	1954	63.364
RFI-SO-00049-1014	84	0.045	RFI-CON-00040S-1321	1955	52.078
RFI-SO-00049-1015	87	4.583	RFI-CON-00040S-1322	1956	8.932
RFI-PO-60045-1002	89	2.657	RFI-CON-00001-1012	1958	8.740
RFI-SO-00049-1016	94	0.018	RFI-CON-00040S-1323	1959	6.921
RFI-PO-60045-1001	95	2.746	RFI-CON-00040S-1324	1966	10.026
…	…	…	RFI-CON-00040S-1326	1985	0.282
…	…	…	RFI-CON-00040S-1327	1996	16.000
…	…	…	RFI-CON-00040S-1328	1997	7.476

…	…	…	RFI-PO-60006-1022	1999	8.975

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Beta	0.05208	2	1.966	1	39.645	2
Gen. Gamma (4P)	0.06106	3	1.9806	2	35.139	1
Dagum	0.0764	5	3.7861	3	43.63	3
Pearson 6	0.08655	9	6.0167	4	54.398	5
Kumaraswamy	0.04983	1	6.1959	5	N/A	
Gamma (3P)	0.09755	10	6.3674	6	60.725	8
Weibull	0.08619	8	6.8214	7	51.989	4
Weibull (3P)	0.08585	7	7.4072	8	63.581	10
Log-Pearson 3	0.08555	6	7.4862	9	59.709	7
Gen. Gamma	0.10608	12	9.0516	10	57.474	6
Burr	0.10102	11	9.117	11	61.792	9
Power Function	0.07069	4	10.003	12	N/A	
Johnson SB	0.1726	29	15.678	13	80.892	11
Log-Logistic (3P)	0.12382	17	16.555	14	104.18	16
Pearson 6 (4P)	0.13221	19	18.423	15	N/A	
Gen. Pareto	0.15003	25	18.564	16	89.665	13
Log-Logistic	0.11838	16	18.766	17	85.695	12
Lognormal	0.11053	14	19.544	18	104.77	17
Frechet (3P)	0.11567	15	19.924	19	100.44	14
Levy (2P)	0.1365	20	21.041	20	102.77	15
Lognormal (3P)	0.10944	13	21.348	21	109.45	18
Burr (4P)	0.14242	23	23.698	22	N/A	
Inv. Gaussian (3P)	0.15983	28	25.258	23	145.15	21
Dagum (4P)	0.14142	21	27.004	24	N/A	
Gen. Extreme Value	0.15463	27	27.842	25	156.66	24
Fatigue Life (3P)	0.14189	22	30.912	26	132.18	20
Gumbel Max	0.18341	31	31.502	27	114.36	19
Pareto 2	0.12839	18	34.801	28	174.57	26
Gamma	0.14367	24	35.627	29	155.09	23
Normal	0.19709	32	45.479	30	167.32	25

Activity Name: Verify Participants

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
RFI-CON-00045-1002	25	0.018	…	…	…
RFI-CON-00045-1002	25	0.016	…	…	…
RFI-SO-00049-1001	31	0.116	RFI-CON-00040S-1310	1925	0.518
RFI-SO-00049-1001	31	0.012	RFI-CON-00040S-1311	1926	0.500
RFI-SO-00027-1001	32	0.031	RFI-CON-00040S-1312	1927	0.370
RFI-SO-00027-1001	32	7.936	RFI-CON-00040S-1313	1928	0.238
RFI-SO-00049-1002	33	0.010	RFI-CON-00040S-1314	1929	0.072
RFI-SO-00049-1002	33	0.009	RFI-CON-00040S-1314	1929	0.119
RFI-SO-00049-1003	34	0.130	RFI-CON-00040S-1315	1930	0.232
RFI-SO-00049-1003	34	0.021	RFI-CON-00040S-1316	1931	0.224
RFI-SO-00049-1004	37	0.018	RFI-CON-00061-1007	1933	0.050
RFI-SO-00049-1004	37	0.043	RFI-CON-00040S-1317	1936	0.013
RFI-CON-00045-1003	38	1.061	RFI-CON-00040S-1318	1937	0.034
RFI-CON-00045-1003	38	0.006	RFI-ECA-INT-1028	1941	0.018
RFI-CON-00045-1004	39	0.961	RFI-CON-00040S-1319	1950	0.015
RFI-CON-00045-1004	39	0.051	RFI-CON-00040S-1319	1950	0.011
RFI-SO-00049-1005	40	0.025	RFI-ECA-INT-1029	1953	0.014
RFI-SO-00049-1005	40	0.005	RFI-CON-00040S-1321	1955	0.033
RFI-SO-00049-1005	40	0.002	RFI-CON-00040S-1321	1955	0.010
RFI-SO-00049-1006	47	0.063	RFI-CON-00040S-1322	1956	0.038
RFI-SO-00049-1006	47	0.036	RFI-CON-00001-1012	1958	0.090
RFI-CON-00040S-1002	50	0.006	RFI-CON-00040S-1323	1959	1.915
RFI-CON-00040S-1002	50	0.010	RFI-CON-00040S-1324	1966	0.033
RFI-SO-00049-1007	51	0.044	RFI-CON-00040S-1325	1971	0.078
RFI-SO-00049-1007	51	0.010	RFI-CON-00040S-1325	1971	0.060
RFI-SO-00049-1008	52	0.050	RFI-CON-00061-1008	1979	0.038
RFI-SO-00049-1008	52	0.010	RFI-CON-00061-1008	1979	0.014
RFI-SO-00049-1008	52	0.007	RFI-CON-00061-1009	1980	0.017
RFI-SO-00049-1009	55	0.016	RFI-CON-00061-1009	1980	0.028
RFI-CON-00032-1001	75	0.056	RFI-CON-00040S-1327	1996	0.006
RFI-CON-00032-1002	77	0.331	RFI-CON-00040S-1328	1997	0.046
RFI-SO-00049-1010	80	0.138	RFI-CON-00040S-1328	1997	0.707
RFI-SO-00049-1011	81	1.210	RFI-CON-00040S-1328	1997	0.015
RFI-SO-00049-1012	82	0.155	RFI-PO-60006-1022	1999	0.029
RFI-SO-00049-1013	83	0.008	RFI-PO-60006-1022	1999	0.075

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Burr	0.05681	1	3.2797	1	34.397	1
Frechet (3P)	0.07667	4	8.3931	2	79.374	4
Dagum	0.07634	3	8.7598	3	72.448	2
Pearson 5 (3P)	0.08162	7	8.7822	4	79.415	5
Dagum (4P)	0.07833	5	8.7862	5	75.679	3
Pearson 6 (4P)	0.08205	8	8.89	6	80.666	6
Pearson 5	0.08534	9	9.6244	7	82.444	7
Pearson 6	0.08681	10	9.9205	8	82.696	8
Log-Logistic (3P)	0.07452	2	17.365	9	139.97	9
Frechet	0.09401	11	18.29	10	156.4	10
Lognormal (3P)	0.1444	14	36.173	11	246.24	12
Inv. Gaussian (3P)	0.15706	16	36.938	12	205.8	11
Inv. Gaussian	0.14073	13	41.514	13	293.96	14
Log-Logistic	0.15942	17	44.193	14	324.53	15
Pareto 2	0.13627	12	44.281	15	362.43	18
Levy (2P)	0.16976	20	47.515	16	249.92	13
Lognormal	0.1634	19	48.642	17	346.46	17
Burr (4P)	0.15031	15	58.241	18	N/A	
Gen. Pareto	0.16142	18	58.28	19	478.4	21
Gen. Extreme Value	0.1745	21	60.994	20	464.86	20
Levy	0.20633	25	71.339	21	336.28	16
Gen. Gamma (4P)	0.18182	22	83.598	22	724.22	26
Weibull (3P)	0.19043	23	93.063	23	820.44	27
Kumaraswamy	0.22251	26	111.48	24	N/A	
Cauchy	0.19191	24	126.19	25	404.41	19
Weibull	0.23862	27	126.25	26	717.69	25
Pareto	0.2875	28	142.02	27	852.48	31
Log-Pearson 3	0.07949	6	185.44	28	N/A	
Beta	0.34108	30	214.51	29	N/A	
Fatigue Life (3P)	0.36186	31	236.48	30	839.25	29

Activity Name: Respond (responders)

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
RFI-CON-00045-1002	25	13.096	…	…	…
RFI-CON-00045-1002	25	3.269	…	…	…
RFI-CON-00045-1002	25	5.786	RFI-CON-00040S-1305	1920	11.582
RFI-CON-00045-1002	25	50.237	RFI-CON-00040S-1306	1921	0.233
RFI-CON-00045-1002	25	49.652	RFI-CON-00040S-1307	1922	5.808
RFI-SO-00049-1001	31	3.705	RFI-CON-00040S-1308	1923	12.626
RFI-SO-00049-1001	31	9.290	RFI-CON-00040S-1309	1924	12.566
RFI-SO-00049-1001	31	0.407	RFI-CON-00040S-1310	1925	12.553
RFI-SO-00027-1001	32	4.245	RFI-CON-00040S-1311	1926	12.453
RFI-SO-00027-1001	32	4.155	RFI-CON-00040S-1312	1927	12.352
RFI-SO-00027-1001	32	17.905	RFI-CON-00040S-1313	1928	12.332
RFI-SO-00049-1002	33	12.169	RFI-CON-00040S-1314	1929	0.223
RFI-SO-00049-1002	33	12.144	RFI-CON-00040S-1314	1929	34.703
RFI-SO-00049-1002	33	19.264	RFI-CON-00040S-1315	1930	12.303
RFI-SO-00049-1003	34	13.702	RFI-CON-00040S-1316	1931	3.108
RFI-SO-00049-1003	34	28.141	RFI-CON-00061-1007	1933	0.334
RFI-SO-00049-1004	37	13.695	RFI-CON-00040S-1318	1937	11.976
RFI-SO-00049-1004	37	13.529	RFI-ECA-INT-1028	1941	0.520
RFI-CON-00045-1003	38	8.764	RFI-CON-00040S-1319	1950	15.187
RFI-CON-00045-1003	38	0.016	RFI-CON-00040S-1321	1955	25.769
RFI-CON-00045-1003	38	93.829	RFI-CON-00040S-1321	1955	14.879
RFI-CON-00045-1004	39	8.895	RFI-CON-00040S-1322	1956	1.219
RFI-CON-00045-1004	39	98.440	RFI-CON-00001-1012	1958	1.257
RFI-CON-00045-1004	39	31.642	RFI-CON-00040S-1323	1959	1.276
RFI-CON-00045-1004	39	29.264	RFI-CON-00040S-1324	1966	7.355
RFI-SO-00049-1005	40	43.022	RFI-CON-00040S-1325	1971	10.211
RFI-SO-00049-1005	40	48.965	RFI-CON-00040S-1325	1971	27.931
RFI-SO-00049-1005	40	0.204	RFI-CON-00040S-1325	1971	31.763
RFI-CON-00040S-1002	50	25.701	RFI-CON-00061-1008	1979	15.045
RFI-CON-00040S-1002	50	16.697	RFI-CON-00061-1009	1980	14.984
RFI-SO-00049-1007	51	13.814	RFI-CON-00061-1009	1980	34.018
RFI-SO-00049-1007	51	15.022	RFI-CON-00061-1009	1980	36.473
RFI-SO-00049-1008	52	13.785	RFI-CON-00040S-1327	1996	15.559
RFI-SO-00049-1008	52	0.234	RFI-CON-00040S-1328	1997	1.528
RFI-SO-00049-1009	55	35.371	RFI-CON-00040S-1328	1997	2.532

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Gen. Gamma (4P)	0.03767	1	1.2374	1	33.452	2
Kumaraswamy	0.0505	4	1.9413	2	36.151	4
Dagum	0.05202	5	1.9657	3	34.225	3
Johnson SB	0.0676	7	4.5459	4	29.541	1
Beta	0.04219	2	5.4977	5	N/A	
Dagum (4P)	0.04273	3	5.5518	6		
Gen. Pareto	0.0814	14	5.6622	7	39.291	5
Log-Pearson 3	0.06626	6	6.0121	8	62.273	8
Pearson 6	0.07501	11	8.9864	9	60.046	6
Burr	0.07046	8	11.412	10	79.864	12
Weibull	0.08429	15	11.426	11	78.388	11
Gen. Extreme Value	0.08842	17	12.064	12	71.198	10
Gamma (3P)	0.07458	10	12.839	13	N/A	
Burr (4P)	0.07414	9	12.839	14	84.289	13
Power Function	0.09187	20	13.617	15	93.96	15
Gen. Gamma	0.08898	18	14.498	16	90.482	14
Gumbel Max	0.10916	27	14.612	17	60.344	7
Frechet (3P)	0.07794	13	15.662	18	105.63	17
Weibull (3P)	0.07581	12	16.286	19	N/A	
Pearson 6 (4P)	0.09129	19	19.05	20	N/A	
Pareto 2	0.09344	24	19.502	21	129.99	20
Exponential	0.0926	21	19.696	22	134.05	22
Lognormal (3P)	0.09593	25	19.998	23	137.58	25
Erlang (3P)	0.09299	22	20.135	24	135.64	23
Exponential (2P)	0.09303	23	20.213	25	135.78	24
Fatigue Life (3P)	0.1317	33	25.289	26	139.58	26
Log-Logistic (3P)	0.11135	28	26.532	27	205.47	33
Erlang	0.08834	16	28.354	28	158.01	29
Pert	0.10885	26	28.488	29	151.64	28
Chi-Squared (2P)	0.11373	29	28.571	30	116.8	19

Activity Name: Review and Consolidate

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
RFI-CON-00045-1002	25	27.023	…	…	…
RFI-SO-00049-1001	31	0.560	…	…	…
RFI-SO-00049-1001	31	0.048	RFI-CON-00040S-1264	1754	33.323
RFI-SO-00027-1001	32	3.722	RFI-CON-00040S-1265	1759	12.471
RFI-SO-00049-1002	33	3.721	RFI-CON-00040S-1266	1760	8.603
RFI-SO-00049-1003	34	14.946	RFI-CON-00040S-1267	1764	11.576
RFI-SO-00049-1004	37	28.570	RFI-CON-00040S-1268	1773	1.962
RFI-CON-00045-1003	38	0.481	RFI-CON-00040S-1269	1778	19.460
RFI-CON-00045-1004	39	0.533	RFI-CON-00040S-1269	1778	12.423
RFI-CON-00045-1004	39	61.616	RFI-CON-00040S-1271	1780	14.419
RFI-SO-00049-1005	40	4.715	RFI-CON-00040S-1272	1787	1.117
RFI-SO-00049-1005	40	0.053	RFI-CON-00040S-1273	1790	57.159
RFI-SO-00049-1006	47	1.050	RFI-CON-00040S-1276	1797	40.800
RFI-SO-00049-1006	47	1.050	RFI-CON-00040S-1282	1823	8.726
RFI-CON-00040S-1002	50	0.566	RFI-CON-00040S-1286	1836	10.595
RFI-SO-00049-1007	51	0.109	RFI-CON-00040S-1287	1839	1.040
RFI-SO-00049-1008	52	4.911	RFI-CON-00040S-1290	1849	28.284
RFI-SO-00049-1015	87	18.866	RFI-CON-00040S-1291	1857	12.909
RFI-PO-60045-1002	89	6.846	RFI-CON-00040S-1295	1871	5.728
RFI-SO-00049-1016	94	2.548	RFI-ECA-INT-1025	1882	27.832
RFI-SO-00049-1016	94	0.467	RFI-CON-00040S-1296	1887	2.414
RFI-SO-00049-1016	94	0.732	RFI-CON-00040S-1297	1891	6.266
RFI-SO-00049-1016	94	0.323	RFI-CON-00040S-1298	1892	11.555
RFI-SO-00049-1016	94	1.003	RFI-SO-00060-0005	1904	0.214
RFI-PO-60045-1001	95	6.928	RFI-SO-00060-0006	1905	0.192
RFI-SO-00035-1001	109	0.017	RFI-SO-00060-0007	1906	0.139
RFI-CON-00040P-1004	110	2.135	RFI-SO-00060-0008	1907	0.168
RFI-CON-00040P-1004	110	0.567	RFI-SO-00060-0009	1914	0.098
RFI-CON-00040P-1005	111	12.177	RFI-CON-00040S-1314	1929	7.441
RFI-CON-00040P-1006	113	0.842	RFI-CON-00040S-1319	1950	0.379
RFI-CON-00040P-1007	114	0.759	RFI-CON-00040S-1321	1955	1.374
RFI-CON-00040S-1003	121	2.065	RFI-CON-00061-1008	1979	16.247
RFI-SO-00024-1002	122	39.949	RFI-CON-00061-1009	1980	15.288

RFI-SO-00024-1002	122	59.514	RFI-CON-00040S-1328	1997	65.181
RFI-CON-00040P-1008	130	2.086	RFI-PO-60006-1022	1999	172.170

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Dagum	0.03753	1	0.63511	1	8.8391	1
Dagum (4P)	0.04091	2	0.63747	2	10.43	2
Pearson 6	0.05812	6	1.1886	3	18.705	5
Gen. Gamma	0.05577	5	1.3813	4	13.495	3
Burr	0.06579	8	1.6363	5	24.858	6
Weibull	0.0686	10	1.7725	6	27.767	9
Weibull (3P)	0.07209	12	1.8496	7	28.018	10
Log-Pearson 3	0.06707	9	2.0571	8	27.255	8
Gen. Pareto	0.07234	13	2.2563	9	15.653	4
Gamma	0.07791	14	3.4319	10	28.203	11
Gen. Extreme Value	0.08985	17	4.3078	11	26.148	7
Gamma (3P)	0.04564	3	5.1859	12	N/A	
Gen. Gamma (4P)	0.0518	4	5.2984	13	N/A	
Pearson 6 (4P)	0.06508	7	5.664	14	N/A	
Beta	0.07106	11	6.1493	15	30.128	12
Inv. Gaussian (3P)	0.1159	22	6.3202	16	50.086	18
Burr (4P)	0.08168	16	6.374	17	N/A	
Fatigue Life (3P)	0.12157	23	6.6181	18	49.896	17
Lognormal (3P)	0.11351	21	6.6216	19	64.29	23
Frechet (3P)	0.1099	20	6.826	20	65.822	26
Johnson SB	0.09213	19	7.0068	21	56.175	19
Pareto 2	0.08064	15	7.0107	22	40.625	15
Log-Logistic	0.13561	27	9.7407	23	72.026	28
Lognormal	0.12677	24	9.8396	24	74.851	29
Log-Logistic (3P)	0.09081	18	11.365	25	N/A	
Levy (2P)	0.14819	28	14.632	26	142.72	32
Exponential	0.13	25	18.047	27	64.788	24
Exponential (2P)	0.13076	26	18.802	28	66.085	27
Gumbel Max	0.24768	43	24.521	29	41.128	16
Cauchy	0.23583	40	25.184	30	38.104	13

Activity Name: Approve

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
RFI-CON-00045-1002	25	37.104	…	…	…
RFI-SO-00049-1001	31	4.291	…	…	…
RFI-SO-00027-1001	32	6.444	…	…	…
RFI-SO-00049-1002	33	6.436	RFI-PO-60006-1020	1878	2.505
RFI-SO-00049-1003	34	2.052	RFI-PO-60006-1021	1879	10.393
RFI-SO-00049-1004	37	1.995	RFI-PO-60006-1021	1879	0.552
RFI-CON-00045-1003	38	4.844	RFI-ECA-INT-1025	1882	21.626
RFI-CON-00045-1003	38	2.468	RFI-SO-00060-0005	1904	0.034
RFI-CON-00045-1003	38	31.953	RFI-SO-00060-0006	1905	0.038
RFI-CON-00045-1004	39	2.766	RFI-SO-00060-0007	1906	0.070
RFI-CON-00045-1004	39	66.525	RFI-SO-00060-0008	1907	0.025
RFI-SO-00049-1005	40	1.193	RFI-CON-00061-1006	1911	4.177
RFI-SO-00049-1005	40	1.193	RFI-SO-00060-0009	1914	0.025
RFI-SO-00049-1006	47	0.353	RFI-ECA-INT-1026	1918	0.539
RFI-CON-00040S-1002	50	3.957	RFI-ECA-INT-1026	1918	126.846
RFI-SO-00049-1007	51	2.101	RFI-ECA-INT-1027	1919	0.035
RFI-SO-00049-1008	52	0.032	RFI-ECA-INT-1027	1919	11.672
RFI-SO-00049-1008	52	1.172	RFI-ECA-INT-1027	1919	11.672
RFI-SO-00049-1008	52	1.172	RFI-CON-00040S-1315	1930	51.431
RFI-SO-00049-1009	55	3.602	RFI-CON-00040S-1316	1931	51.444
RFI-CON-00032-1001	75	4.472	RFI-CON-00061-1007	1933	2.835
RFI-CON-00032-1001	75	13.782	RFI-CON-00040S-1317	1936	24.316
RFI-CON-00032-1002	77	0.018	RFI-ECA-INT-1028	1941	2.439
RFI-CON-00032-1002	77	0.017	RFI-ECA-INT-1028	1941	178.661
RFI-CON-00032-1002	77	19.750	RFI-CON-00040S-1319	1950	10.873
RFI-SO-00049-1010	80	0.346	RFI-ECA-INT-1029	1953	4.559
RFI-SO-00049-1010	80	0.346	RFI-ECA-INT-1029	1953	4.559
RFI-SO-00049-1011	81	6.250	RFI-CON-00040S-1322	1956	5.645
RFI-SO-00049-1011	81	6.250	RFI-CON-00001-1012	1958	5.614
RFI-SO-00049-1012	82	1.869	RFI-CON-00040S-1323	1959	5.407
RFI-SO-00049-1012	82	1.869	RFI-CON-00040S-1325	1971	8.919
RFI-SO-00049-1013	83	5.927	RFI-CON-00040S-1325	1971	117.437
RFI-SO-00049-1014	84	0.323	RFI-CON-00061-1008	1979	0.328
RFI-SO-00049-1014	84	0.323	RFI-CON-00061-1009	1980	0.037
RFI-SO-00049-1015	87	55.831	RFI-PO-60006-1022	1999	52.811

RFI-PO-60045-1002	89	5.208	RFI-PO-60006-1022	1999	9.170

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Beta	0.06617	5	1.6926	1	14.831	1
Dagum	0.08326	13	2.8685	2	19.448	3
Weibull	0.07119	7	4.1744	3	17.097	2
Log-Pearson 3	0.07404	8	4.3117	4	21.904	4
Pearson 6	0.07968	11	4.5485	5	31.027	6
Burr	0.07832	10	5.581	6	30.079	5
Dagum (4P)	0.04388	2	5.8909	7	N/A	
Gen. Gamma	0.08795	15	6.0061	8	32.267	7
Kumaraswamy	0.05785	3	6.1774	9	N/A	
Gen. Gamma (4P)	0.06699	6	6.2687	10	N/A	
Power Function	0.08436	14	7.1619	11	N/A	
Gamma (3P)	0.06296	4	7.2271	12	N/A	
Gen. Pareto	0.12146	23	8.1831	13	42.516	10
Weibull (3P)	0.0752	9	9.0475	14	N/A	
Burr (4P)	0.08255	12	10.254	15	N/A	
Lognormal (3P)	0.09656	16	10.738	16	53.717	12
Frechet (3P)	0.09988	17	11.356	17	74.668	15
Log-Logistic	0.11499	22	12.432	18	42.119	9
Lognormal	0.11133	19	12.463	19	40.518	8
Inv. Gaussian (3P)	0.10891	18	12.677	20	90.75	18
Gen. Extreme Value	0.11243	21	13.384	21	68.859	14
Log-Logistic (3P)	0.12456	25	14.058	22	N/A	
Fatigue Life (3P)	0.12813	26	14.112	23	58.28	13
Pearson 6 (4P)	0.11214	20	15.628	24	N/A	
Gumbel Max	0.13756	29	16.381	25	81.57	16
Levy (2P)	0.18805	36	16.981	26	46.673	11
Johnson SB	0.03225	1	20.156	27	N/A	
Gamma	0.13631	28	20.729	28	127.42	24
Pareto 2	0.1362	27	20.779	29	131.17	25
Exponential	0.14138	30	22.994	30	133.82	26

Trend Workflow Data from British Columbia Construction project

Activity Name: Construct Trend

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00001	1118	1.746	…	…	…
TRD-00-00063	1129	29.887	…	…	…
TRD-00-00063	1129	0.023	TRD-00-00913	3088	0.011
TRD-00-00063	1129	0.007	TRD-00-00914	3089	2.831
TRD-00-00063	1129	0.005	TRD-00-00914	3089	0.011
TRD-00-00063	1129	0.008	TRD-00-00914	3089	0.010
TRD-00-00063	1129	0.007	TRD-00-00919	3091	0.011
TRD-00-00063	1129	0.007	TRD-00-00919	3091	0.018
TRD-00-00063	1129	0.010	TRD-00-00915	3092	0.714
TRD-00-00063	1129	0.011	TRD-00-00915	3092	0.010
TRD-00-00063	1129	0.006	TRD-00-00915	3092	0.011
TRD-00-00002	1130	15.218	TRD-00-00916	3095	1.491
TRD-00-00002	1130	0.011	TRD-00-00916	3095	0.011
TRD-00-00003	1131	0.995	TRD-00-00916	3095	0.020
TRD-00-00005	1133	2.345	TRD-00-00917	3096	0.293
TRD-00-00006	1134	4.147	TRD-00-00917	3096	0.274
TRD-00-00007	1135	19.443	TRD-00-00917	3096	0.012
TRD-00-00008	1136	19.925	TRD-00-00918	3097	0.391
TRD-00-00009	1137	8.695	TRD-00-00918	3097	10.618
TRD-00-00010	1138	16.876	TRD-00-00918	3097	1.297
TRD-00-00012	1141	0.305	TRD-00-00918	3097	0.010
TRD-00-00013	1142	112.432	TRD-00-00920	3098	0.299
TRD-00-00013	1142	0.006	TRD-00-00922	3101	1.527
TRD-00-00014	1143	112.303	TRD-00-00922	3101	0.012
TRD-00-00015	1144	109.702	TRD-00-00921	3103	0.196
TRD-00-00016	1145	112.315	TRD-00-00921	3103	0.019
TRD-00-00018	1146	0.538	TRD-00-00923	3105	1.893
TRD-00-00017	1147	0.472	TRD-00-00923	3105	0.429
…	…	…	TRD-00-00924	3107	0.386
…	…	…	TRD-00-00925	3108	2.240

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Lognormal (3P)	0.04298	1	2.8238	1	59.983	2
Frechet (3P)	0.08472	7	8.4332	2	46.902	1
Fatigue Life (3P)	0.11572	16	9.5828	3	120.82	9
Log-Logistic (3P)	0.05399	3	12.026	4	N/A	
Burr (4P)	0.05364	2	12.064	5	N/A	
Pearson 6 (4P)	0.08668	8	12.579	6	140.92	15
Dagum (4P)	0.05714	5	14.334	7	N/A	
Frechet	0.09992	10	14.578	8	103.61	5
Pareto 2	0.12426	19	15.033	9	75.08	3
Weibull (3P)	0.06501	6	15.427	10	N/A	
Pearson 5	0.10545	13	15.951	11	135.09	13
Pearson 6	0.10708	14	15.976	12	134.14	12
Log-Pearson 3	0.10492	11	16.102	13	98.927	4
Dagum	0.1087	15	17.654	14	136.4	14
Pearson 5 (3P)	0.12972	21	17.901	15	105.9	6
Pareto	0.09109	9	22.278	16	108.14	7
Lognormal	0.1188	17	22.952	17	111.07	8
Log-Logistic	0.12119	18	23.022	18	122.37	10
Levy	0.10533	12	23.14	19	179.68	17
Kumaraswamy	0.12691	20	26.528	20	N/A	
Gamma (3P)	0.15897	24	29.234	21	226.48	22
Gen. Gamma (4P)	0.14361	23	31.363	22	N/A	
Weibull	0.14162	22	45.168	23	168.29	16
Fatigue Life	0.18195	25	53.577	24	182.87	18
Levy (2P)	0.1841	26	55.932	25	250.95	24
Inv. Gaussian (3P)	0.18665	27	62.239	26	274.92	25
Gamma	0.32073	35	67.536	27	130.68	11
Gen. Gamma	0.22429	28	71.285	28	183.99	19
Gen. Pareto	0.24953	31	75.321	29	221.5	21
Gen. Extreme Value	0.23788	30	84.394	30	282.31	26

Activity Name: Select Contributors

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00001	1118	0.064	…	…	…
TRD-00-00063	1129	0.013	…	…	…
TRD-00-00063	1129	0.014	TRD-00-00911	3083	0.043
TRD-00-00063	1129	0.015	TRD-00-00912	3085	0.027
TRD-00-00063	1129	0.008	TRD-00-00912	3085	0.009
TRD-00-00063	1129	0.009	TRD-00-00912	3085	0.043
TRD-00-00063	1129	0.006	TRD-00-00913	3088	0.032
TRD-00-00063	1129	0.009	TRD-00-00913	3088	0.011
TRD-00-00063	1129	0.031	TRD-00-00913	3088	0.076
TRD-00-00063	1129	0.018	TRD-00-00914	3089	0.009
TRD-00-00063	1129	0.009	TRD-00-00914	3089	0.011
TRD-00-00002	1130	0.052	TRD-00-00914	3089	0.034
TRD-00-00002	1130	0.014	TRD-00-00919	3091	0.247
TRD-00-00003	1131	0.511	TRD-00-00919	3091	0.027
TRD-00-00005	1133	1.768	TRD-00-00915	3092	0.030
TRD-00-00006	1134	1.211	TRD-00-00915	3092	0.012
TRD-00-00007	1135	0.405	TRD-00-00915	3092	0.024
TRD-00-00008	1136	0.192	TRD-00-00916	3095	0.010
TRD-00-00009	1137	0.094	TRD-00-00916	3095	0.014
TRD-00-00010	1138	0.091	TRD-00-00916	3095	0.032
TRD-00-00012	1141	0.111	TRD-00-00917	3096	0.009
TRD-00-00013	1142	0.190	TRD-00-00917	3096	0.060
TRD-00-00013	1142	0.016	TRD-00-00918	3097	0.014
TRD-00-00014	1143	0.106	TRD-00-00918	3097	0.029
TRD-00-00015	1144	0.104	TRD-00-00922	3101	0.033
TRD-00-00016	1145	0.089	TRD-00-00922	3101	0.045
TRD-00-00018	1146	0.036	TRD-00-00921	3103	0.012
TRD-00-00017	1147	0.205	TRD-00-00921	3103	0.041
…	…	…	TRD-00-00923	3105	0.009
…	…	…	TRD-00-00925	3108	0.008

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Burr (4P)	0.0398	2	2.3977	1	25.951	1
Burr	0.05053	3	2.9411	2	34.795	2
Pearson 5 (3P)	0.05326	5	4.2236	3	58.209	3
Dagum (4P)	0.05117	4	4.5788	4	62.391	4
Frechet (3P)	0.0558	6	4.7192	5	64.445	5
Pearson 5	0.08654	11	11.534	6	143.46	8
Pearson 6	0.0863	10	11.59	7	143.08	7
Frechet	0.08064	9	11.913	8	174.75	9
Log-Logistic (3P)	0.07604	7	13.106	9	203.09	11
Lognormal (3P)	0.07749	8	20.29	10	225.35	12
Inv. Gaussian (3P)	0.09674	13	26.357	11	142.25	6
Gen. Pareto	0.13309	18	28.321	12	358.22	14
Pearson 6 (4P)	0.09207	12	30.022	13	314.7	13
Gen. Extreme Value	0.13247	17	30.184	14	442.87	15
Log-Logistic	0.11501	15	34.015	15	445.85	16
Lognormal	0.1136	14	36.226	16	447.55	17
Levy (2P)	0.12121	16	38.455	17	186.97	10
Pareto 2	0.16892	20	38.754	18	524.18	19
Log-Pearson 3	0.03348	1	48.926	19	N/A	
Weibull (3P)	0.20179	22	79.18	20	687.64	24
Levy	0.17345	21	86.58	21	463.07	18
Kumaraswamy	0.21867	24	99.504	22	N/A	
Pareto	0.24444	25	110.2	23	559.88	20
Weibull	0.1634	19	113.17	24	778.19	28
Inv. Gaussian	0.28317	27	125.95	25	612.08	23
Dagum	0.21193	23	152.39	26	592.34	22
Cauchy	0.27049	26	191.28	27	932.47	30
Beta	0.309	28	218.8	28	N/A	
Fatigue Life (3P)	0.3308	30	253.62	29	701.6	26
Gen. Gamma (4P)	0.32875	29	259.1	30	N/A	

Activity Name: Detail Impacts

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00063	1129	0.149	..	…	…
TRD-00-00063	1129	0.312	…	…	…
TRD-00-00063	1129	1.670	TRD-00-00913	3088	1.317
TRD-00-00063	1129	0.192	TRD-00-00913	3088	1.182
TRD-00-00063	1129	0.713	TRD-00-00913	3088	8.198
TRD-00-00002	1130	9.533	TRD-00-00913	3088	6.705
TRD-00-00002	1130	32.468	TRD-00-00914	3089	1.257
TRD-00-00002	1130	0.556	TRD-00-00914	3089	22.882
TRD-00-00002	1130	18.380	TRD-00-00914	3089	1.954
TRD-00-00002	1130	8.861	TRD-00-00914	3089	3.251
TRD-00-00002	1130	13.793	TRD-00-00919	3091	0.038
TRD-00-00002	1130	1.436	TRD-00-00919	3091	4.402
TRD-00-00002	1130	1.230	TRD-00-00915	3092	1.112
TRD-00-00005	1133	16.000	TRD-00-00915	3092	22.978
TRD-00-00005	1133	9.590	TRD-00-00915	3092	4.909
TRD-00-00005	1133	11.747	TRD-00-00915	3092	2.497
TRD-00-00005	1133	0.292	TRD-00-00916	3095	0.088
TRD-00-00005	1133	0.145	TRD-00-00916	3095	23.148
TRD-00-00006	1134	4.720	TRD-00-00916	3095	3.711
TRD-00-00006	1134	37.274	TRD-00-00916	3095	3.032
TRD-00-00006	1134	6.379	TRD-00-00917	3096	0.174
TRD-00-00006	1134	0.697	TRD-00-00917	3096	0.066
TRD-00-00006	1134	23.183	TRD-00-00917	3096	0.253
TRD-00-00006	1134	0.098	TRD-00-00917	3096	6.648
TRD-00-00006	1134	1.432	TRD-00-00918	3097	7.622
TRD-00-00007	1135	18.256	TRD-00-00918	3097	2.084
TRD-00-00007	1135	37.758	TRD-00-00922	3101	1.056
TRD-00-00007	1135	0.733	TRD-00-00922	3101	0.164
…	…	…	TRD-00-00921	3103	0.464
…	…	…	TRD-00-00921	3103	0.202

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Kumaraswamy	0.05424	1	12.217	1	187.66	2
Gen. Gamma (4P)	0.05699	2	13.126	2	199.36	7
Gamma (3P)	0.06148	3	15.442	3	201.81	9
Gen. Gamma	0.06352	5	16.469	4	197.21	4
Burr	0.06212	4	16.897	5	200.53	8
Weibull (3P)	0.06891	8	17.113	6	197.84	6
Pearson 6	0.06486	6	17.306	7	197.02	3
Beta	0.06603	7	18.534	8	218.68	13
Log-Pearson 3	0.08515	13	18.963	9	248.91	14
Weibull	0.07017	9	21.204	10	207.84	11
Dagum	0.07605	11	23.177	11	250.86	15
Gamma	0.07779	12	23.245	12	208.89	12
Dagum (4P)	0.07249	10	24.62	13	N/A	
Pearson 6 (4P)	0.08649	14	25.18	14	N/A	
Fatigue Life (3P)	0.1019	18	27.514	15	206.65	10
Lognormal (3P)	0.10268	19	30.734	16	346.13	18
Lognormal	0.10269	20	30.738	17	346.18	19
Log-Logistic (3P)	0.09341	15	31.746	18	393.2	22
Burr (4P)	0.09962	17	34.912	19	N/A	
Johnson SB	0.12656	25	36.581	20	165.66	1
Log-Logistic	0.11741	22	41.017	21	388.15	21
Gen. Pareto	0.12704	26	46.042	22	197.62	5
Frechet (3P)	0.1125	21	49.996	23	529.65	27
Pareto 2	0.09842	16	56.287	24	441.06	23
Inv. Gaussian (3P)	0.14252	28	63.376	25	449.62	24
Gen. Extreme Value	0.12093	23	69.443	26	348.85	20
Fatigue Life	0.12214	24	73.079	27	301.08	16
Frechet	0.13785	27	90.739	28	762.82	31
Gumbel Max	0.2162	38	119.87	29	320.04	17
Pearson 5	0.15518	29	126.73	30	1108.8	40

Activity Name: Send for Approval

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00001	1118	8.757	…	…	…
TRD-00-00063	1129	6.155	…	…	…
TRD-00-00063	1129	6.316	TRD-00-00907	3078	0.224
TRD-00-00063	1129	0.063	TRD-00-00907	3078	0.704
TRD-00-00063	1129	24.967	TRD-00-00908	3079	0.164
TRD-00-00063	1129	23.168	TRD-00-00908	3079	0.228
TRD-00-00063	1129	73.485	TRD-00-00908	3079	0.650
TRD-00-00063	1129	30.726	TRD-00-00909	3081	0.129
TRD-00-00063	1129	18.349	TRD-00-00909	3081	0.444
TRD-00-00063	1129	16.000	TRD-00-00909	3081	0.098
TRD-00-00063	1129	0.164	TRD-00-00910	3082	0.087
TRD-00-00002	1130	0.373	TRD-00-00910	3082	0.281
TRD-00-00002	1130	1.114	TRD-00-00910	3082	0.630
TRD-00-00003	1131	0.379	TRD-00-00912	3085	0.194
TRD-00-00005	1133	0.030	TRD-00-00912	3085	1.445
TRD-00-00006	1134	0.062	TRD-00-00913	3088	0.228
TRD-00-00007	1135	1.775	TRD-00-00913	3088	0.414
TRD-00-00008	1136	2.163	TRD-00-00913	3088	0.087
TRD-00-00009	1137	2.019	TRD-00-00914	3089	0.747
TRD-00-00010	1138	11.323	TRD-00-00914	3089	0.577
TRD-00-00012	1141	0.898	TRD-00-00919	3091	0.151
TRD-00-00013	1142	2.330	TRD-00-00915	3092	0.904
TRD-00-00013	1142	0.068	TRD-00-00915	3092	1.084
TRD-00-00013	1142	0.068	TRD-00-00916	3095	0.914
TRD-00-00014	1143	8.244	TRD-00-00916	3095	0.340
TRD-00-00015	1144	3.726	TRD-00-00917	3096	0.972
TRD-00-00015	1144	3.726	TRD-00-00917	3096	1.622
TRD-00-00016	1145	19.065	TRD-00-00918	3097	0.857
…	…	…	TRD-00-00922	3101	0.425
…	…	…	TRD-00-00921	3103	1.264

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Fatigue Life (3P)	0.02841	1	0.50446	1	8.8602	1
Lognormal (3P)	0.068	10	7.2075	2	64.029	3
Log-Pearson 3	0.06409	6	7.3033	3	70.547	4
Gen. Gamma (4P)	0.06078	4	7.5358	4	75.165	8
Lognormal	0.06864	11	7.7153	5	72.937	5
Fatigue Life	0.05232	2	8.1507	6	32.822	2
Log-Logistic (3P)	0.06627	9	8.7098	7	73.18	6
Burr (4P)	0.06603	8	8.8571	8	93.001	10
Burr	0.06507	7	9.6737	9	99.08	12
Dagum	0.06962	12	10.138	10	82.653	9
Weibull (3P)	0.05397	3	12.094	11	N/A	
Pearson 6 (4P)	0.08072	15	12.399	12	117.89	14
Pearson 6	0.07791	14	13.118	13	133.55	17
Dagum (4P)	0.06238	5	13.374	14	N/A	
Log-Logistic	0.08516	18	13.636	15	103.82	13
Frechet (3P)	0.07585	13	14.257	16	98.605	11
Pareto 2	0.0877	19	15.744	17	118	15
Levy (2P)	0.08509	17	17.706	18	134.14	18
Weibull	0.08307	16	17.829	19	159.88	20
Gamma	0.11677	25	19.645	20	220.33	26
Gen. Gamma	0.10373	22	23.733	21	197.05	24
Gamma (3P)	0.0998	21	23.991	22	N/A	
Frechet	0.09946	20	24.904	23	158.34	19
Beta	0.10534	23	27.29	24	N/A	
Inv. Gaussian (3P)	0.1223	27	28.645	25	164.77	21
Johnson SB	0.18028	31	32.195	26	73.215	7
Gen. Pareto	0.16362	30	37.858	27	119.17	16
Pearson 5	0.12207	26	40.401	28	181.82	23
Gen. Extreme Value	0.15035	29	48.712	29	207.78	25
Kumaraswamy	0.11334	24	52.791	30	N/A	

Activity Name: Select Approvers

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00001	1118	0.021	…	…	…
TRD-00-00063	1129	0.005	…	…	…
TRD-00-00002	1130	0.007	TRD-00-00881	2994	0.007
TRD-00-00003	1131	0.013	TRD-00-00882	2995	0.007
TRD-00-00005	1133	0.012	TRD-00-00883	2996	0.008
TRD-00-00006	1134	0.009	TRD-00-00884	3000	0.004
TRD-00-00007	1135	0.021	TRD-00-00885	3009	0.006
TRD-00-00008	1136	0.013	TRD-00-00886	3014	0.031
TRD-00-00009	1137	0.028	TRD-00-00888	3027	0.007
TRD-00-00010	1138	0.015	TRD-00-00889	3028	0.008
TRD-00-00012	1141	0.024	TRD-00-00890	3031	0.005
TRD-00-00013	1142	1.504	TRD-00-00891	3032	0.012
TRD-00-00013	1142	1.504	TRD-00-00892	3036	0.007
TRD-00-00014	1143	0.010	TRD-00-00893	3041	0.004
TRD-00-00015	1144	0.008	TRD-00-00894	3050	0.007
TRD-00-00015	1144	0.008	TRD-00-00895	3054	0.030
TRD-00-00016	1145	1.242	TRD-00-00896	3055	0.006
TRD-00-00018	1146	0.019	TRD-00-00897	3060	0.008
TRD-00-00017	1147	0.038	TRD-00-00899	3067	0.007
TRD-00-00019	1149	0.007	TRD-00-00900	3068	0.352
TRD-00-00019	1149	0.007	TRD-00-00902	3073	0.009
TRD-00-00020	1153	0.013	TRD-00-00903	3074	0.021
TRD-00-00044	1155	1.290	TRD-00-00904	3075	0.006
TRD-00-00044	1155	1.290	TRD-00-00905	3076	0.012
TRD-00-00021	1157	0.007	TRD-00-00906	3077	0.007
TRD-00-00023	1159	0.032	TRD-00-00908	3079	0.007
TRD-00-00024	1160	0.019	TRD-00-00909	3081	0.008
TRD-00-00026	1162	0.009	TRD-00-00910	3082	0.034
…	…	…	TRD-00-00913	3088	0.010
…	…	…	TRD-00-00917	3096	0.009

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Burr (4P)	0.02986	1	0.60333	1	9.0447	1
Burr	0.04307	2	1.2462	2	12.317	2
Pearson 5 (3P)	0.06754	4	3.3483	3	25.82	3
Frechet (3P)	0.067	3	3.6531	4	27.233	4
Pearson 6 (4P)	0.07121	6	3.8232	5	33.679	5
Dagum (4P)	0.06828	5	4.4365	6	34.937	6
Log-Pearson 3	0.07362	7	9.2404	7	N/A	
Log-Logistic (3P)	0.07722	8	10.34	8	87.077	10
Pearson 5	0.11633	10	11.614	9	77.987	8
Pearson 6	0.11642	11	11.681	10	78.207	9
Inv. Gaussian (3P)	0.11808	12	12.467	11	69.293	7
Frechet	0.10489	9	13.19	12	115.79	12
Lognormal (3P)	0.1196	13	16.994	13	121.87	13
Levy (2P)	0.14564	15	19.987	14	97.52	11
Inv. Gaussian	0.13683	14	24.36	15	205.02	16
Gen. Pareto	0.15777	19	27.317	16	274.3	23
Log-Logistic	0.15461	17	27.835	17	242.78	19
Pareto 2	0.18725	21	28.328	18	294.47	25
Gen. Extreme Value	0.15176	16	29.292	19	275.21	24
Lognormal	0.15518	18	30.182	20	246.3	20
Pareto	0.21881	23	39.484	21	206.18	17
Weibull (3P)	0.18759	22	42.109	22	N/A	
Levy	0.22406	24	50.128	23	263.03	21
Dagum	0.16654	20	55.646	24	146.51	14
Weibull	0.22616	25	59.721	25	446.36	33
Gen. Gamma (4P)	0.26877	27	71.074	26	N/A	
Gamma (3P)	0.28467	28	76.534	27	N/A	
Fatigue Life (3P)	0.29424	29	77.619	28	267.6	22
Cauchy	0.22786	26	89.082	29	306.21	26
Kumaraswamy	0.29932	30	92.309	30	N/A	

Activity Name: Pre Approve

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00001	1118	32.80456	…	…	…
TRD-00-00063	1129	13.74586	…	…	…
TRD-00-00063	1129	0.735994	TRD-00-00897	3060	10.24691
TRD-00-00002	1130	29.71712	TRD-00-00899	3067	1.233053
TRD-00-00002	1130	13.28415	TRD-00-00899	3067	0.427902
TRD-00-00002	1130	4.015947	TRD-00-00899	3067	24.90025
TRD-00-00002	1130	3.857544	TRD-00-00899	3067	8.012482
TRD-00-00003	1131	106.7441	TRD-00-00900	3068	43.89292
TRD-00-00003	1131	58.83833	TRD-00-00900	3068	0.080642
TRD-00-00003	1131	16.43219	TRD-00-00900	3068	3.839993
TRD-00-00005	1133	11.13255	TRD-00-00902	3073	5.874973
TRD-00-00005	1133	30.35377	TRD-00-00902	3073	1.666468
TRD-00-00006	1134	86.05909	TRD-00-00902	3073	6.413439
TRD-00-00007	1135	11.38625	TRD-00-00903	3074	8.370376
TRD-00-00007	1135	30.85409	TRD-00-00903	3074	0.649661
TRD-00-00008	1136	11.38148	TRD-00-00903	3074	20.82739
TRD-00-00008	1136	30.84994	TRD-00-00904	3075	0.239384
TRD-00-00009	1137	11.16036	TRD-00-00904	3075	0.110281
TRD-00-00009	1137	31.02863	TRD-00-00904	3075	36.47081
TRD-00-00010	1138	160.279	TRD-00-00906	3077	1.454341
TRD-00-00012	1141	17.99454	TRD-00-00906	3077	0.381215
TRD-00-00012	1141	1.113407	TRD-00-00906	3077	19.94977
TRD-00-00012	1141	10.89685	TRD-00-00909	3081	5.78924
TRD-00-00013	1142	33.64111	TRD-00-00909	3081	0.082705
TRD-00-00013	1142	49.66239	TRD-00-00909	3081	7.408461
TRD-00-00013	1142	58.89534	TRD-00-00913	3088	0.206458
TRD-00-00013	1142	62.64373	TRD-00-00913	3088	0.378388
…	…	…	TRD-00-00913	3088	24.81236
…	…	…	TRD-00-00913	3088	7.9491

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Gen. Gamma (4P)	0.02555	1	1.2006	1	10.899	2
Log-Pearson 3	0.04064	9	1.7934	2	20.421	6
Pearson 6	0.03097	2	2.002	3	12.495	3
Dagum	0.03537	6	2.5092	4	24.354	8
Johnson SB	0.0426	12	2.602	5	32.143	11
Weibull	0.04202	11	2.8131	6	19.157	4
Dagum (4P)	0.0351	4	2.9252	7	8.9131	1
Burr	0.03535	5	3.5003	8	25.228	9
Gen. Gamma	0.03917	8	3.5407	9	21.371	7
Weibull (3P)	0.03673	7	3.6646	10	25.487	10
Gamma (3P)	0.03222	3	5.9585	11	N/A	
Gen. Pareto	0.07134	19	6.1761	12	19.5	5
Kumaraswamy	0.04071	10	6.3377	13	N/A	
Burr (4P)	0.04365	13	6.784	14	41.724	12
Pearson 6 (4P)	0.04487	14	8.8523	15	N/A	
Beta	0.05383	15	8.8978	16	N/A	
Gamma	0.06265	16	11.162	17	68.339	14
Lognormal (3P)	0.07133	18	14.134	18	98.788	21
Pareto 2	0.06819	17	14.154	19	83.506	16
Fatigue Life (3P)	0.08883	25	14.422	20	88.824	18
Gen. Extreme Value	0.08571	24	15.499	21	78.573	15
Frechet (3P)	0.07339	20	15.584	22	98.2	20
Exponential	0.07579	22	17.521	23	88.169	17
Exponential (2P)	0.07609	23	17.855	24	89.088	19
Log-Logistic (3P)	0.07433	21	18.058	25	148.81	22
Log-Logistic	0.11065	26	26.187	26	156.27	23
Lognormal	0.11141	27	26.823	27	175.16	27
Gumbel Max	0.14307	29	30.318	28	62.941	13
Pert	0.11698	28	47.66	29	169.92	25
Chi-Squared (2P)	0.14711	30	52.111	30	159.62	24

Activity Name: Approve

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00063	1129	61.42	…	…	…
TRD-00-00002	1130	16.64	…	…	…
TRD-00-00003	1131	116.31	TRD-00-00857	2934	0.059
TRD-00-00005	1133	0.41	TRD-00-00858	2943	0.282
TRD-00-00007	1135	0.28	TRD-00-00859	2945	0.751
TRD-00-00008	1136	0.29	TRD-00-00860	2946	18.279
TRD-00-00009	1137	0.20	TRD-00-00861	2947	18.158
TRD-00-00010	1138	20.95	TRD-00-00862	2948	42.305
TRD-00-00012	1141	18.55	TRD-00-00867	2969	0.094
TRD-00-00013	1142	1.40	TRD-00-00868	2970	0.722
TRD-00-00014	1143	21.16	TRD-00-00869	2975	17.944
TRD-00-00015	1144	23.40	TRD-00-00871	2978	42.406
TRD-00-00018	1146	113.40	TRD-00-00872	2979	20.154
TRD-00-00017	1147	12.19	TRD-00-00873	2981	0.666
TRD-00-00019	1149	1.53	TRD-00-00874	2984	0.689
TRD-00-00020	1153	3.65	TRD-00-00875	2986	42.901
TRD-00-00021	1157	0.37	TRD-00-00876	2987	0.409
TRD-00-00023	1159	28.88	TRD-00-00877	2990	42.384
TRD-00-00024	1160	0.40	TRD-00-00878	2991	42.338
TRD-00-00026	1162	0.41	TRD-00-00879	2992	42.319
TRD-00-00029	1170	27.25	TRD-00-00880	2993	42.293
TRD-00-00030	1171	31.02	TRD-00-00881	2994	42.276
TRD-00-00031	1173	21.66	TRD-00-00882	2995	42.246
TRD-00-00037	1180	6.69	TRD-00-00883	2996	42.215
TRD-00-00110	1182	10.26	TRD-00-00884	3000	42.954
TRD-00-00034	1183	22.68	TRD-00-00885	3009	42.445
TRD-00-00032	1186	44.44	TRD-00-00886	3014	42.462
TRD-00-00038	1196	19.22	TRD-00-00889	3028	18.485
…	…	…	TRD-00-00890	3031	32.395
…	…	…	TRD-00-00891	3032	26.185

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Dagum	0.04377	2	0.55931	1	12.758	1
Gen. Pareto	0.08861	14	3.1383	2	13.274	2
Johnson SB	0.08976	15	3.155	3	14.447	3
Log-Pearson 3	0.06324	6	3.6432	4	38.505	6
Dagum (4P)	0.036	1	4.6414	5	N/A	
Gen. Gamma (4P)	0.04847	5	4.6854	6	N/A	
Beta	0.04682	3	5.1175	7	N/A	
Kumaraswamy	0.04785	4	5.2198	8	N/A	
Weibull	0.07978	7	5.6072	9	43.228	8
Gamma (3P)	0.0959	18	5.6939	10	37.318	5
Weibull (3P)	0.08579	9	5.9183	11	48.319	10
Gen. Extreme Value	0.10324	19	6.7664	12	57.147	21
Gumbel Max	0.10915	21	7.4636	13	50.101	12
Power Function	0.1135	27	7.5304	14	44.808	9
Gen. Gamma	0.10717	20	7.8129	15	48.878	11
Frechet (3P)	0.09509	17	8.3749	16	57.517	22
Exponential	0.08653	11	8.4273	17	53.431	16
Burr	0.08629	10	8.6068	18	54.243	19
Pearson 6	0.11151	25	8.7876	19	60.509	25
Erlang (3P)	0.08737	12	8.8154	20	54.132	17
Pareto 2	0.08544	8	8.8586	21	56.261	20
Exponential (2P)	0.08746	13	8.9284	22	54.222	18
Lognormal (3P)	0.10946	22	9.4704	23	66.808	27
Chi-Squared (2P)	0.11128	24	9.9489	24	60.774	26
Pert	0.11197	26	10.624	25	51.883	15
Error	0.13253	30	10.789	26	34.175	4
Pearson 6 (4P)	0.12547	28	10.938	27	51.833	14
Normal	0.13111	29	11.171	28	43.132	7
Fatigue Life (3P)	0.13337	31	11.265	29	59.589	24
Log-Logistic (3P)	0.11126	23	11.468	30	87.354	32

Activity Name: Approve Close Out

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00063	1129	0.056	…	…	…
TRD-00-00003	1131	16.027	…	…	…
TRD-00-00005	1133	26.113	TRD-00-00855	2931	0.485
TRD-00-00007	1135	16.243	TRD-00-00857	2934	18.394
TRD-00-00008	1136	16.476	TRD-00-00858	2943	18.376
TRD-00-00009	1137	16.612	TRD-00-00859	2945	18.111
TRD-00-00010	1138	1.270	TRD-00-00860	2946	0.303
TRD-00-00012	1141	4.906	TRD-00-00861	2947	0.398
TRD-00-00013	1142	4.448	TRD-00-00862	2948	19.607
TRD-00-00014	1143	1.268	TRD-00-00867	2969	18.074
TRD-00-00015	1144	42.075	TRD-00-00868	2970	17.803
TRD-00-00018	1146	9.914	TRD-00-00869	2975	0.734
TRD-00-00017	1147	5.462	TRD-00-00871	2978	2.043
TRD-00-00020	1153	4.148	TRD-00-00872	2979	0.057
TRD-00-00021	1157	2.825	TRD-00-00873	2981	17.675
TRD-00-00023	1159	2.249	TRD-00-00874	2984	16.470
TRD-00-00024	1160	2.887	TRD-00-00875	2986	0.257
TRD-00-00026	1162	2.890	TRD-00-00876	2987	16.499
TRD-00-00029	1170	1.198	TRD-00-00877	2990	1.962
TRD-00-00031	1173	1.172	TRD-00-00878	2991	0.456
TRD-00-00037	1180	7.248	TRD-00-00879	2992	1.934
TRD-00-00110	1182	0.353	TRD-00-00880	2993	1.872
TRD-00-00034	1183	47.813	TRD-00-00881	2994	1.797
TRD-00-00032	1186	0.350	TRD-00-00882	2995	1.721
TRD-00-00038	1196	0.924	TRD-00-00883	2996	1.529
TRD-00-00040	1199	1.502	TRD-00-00884	3000	0.198
TRD-00-00043	1202	10.571	TRD-00-00885	3009	0.541
TRD-00-00045	1207	2.307	TRD-00-00886	3014	0.419
…	…	…	TRD-00-00889	3028	0.233
…	…	…	TRD-00-00890	3031	0.106

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Log-Pearson 3	0.03854	1	0.71445	1	30.625	12
Fatigue Life (3P)	0.04993	4	0.76981	2	23.274	5
Lognormal (3P)	0.0554	9	1.0361	3	28.97	10
Lognormal	0.0558	11	1.0706	4	25.412	7
Dagum	0.05039	5	1.23	5	22.412	4
Burr	0.05564	10	1.2767	6	20.21	1
Log-Logistic (3P)	0.05668	12	1.291	7	27.376	9
Pearson 6	0.05257	7	1.3308	8	32.694	13
Pareto 2	0.04847	3	1.3965	9	35.276	14
Log-Logistic	0.07162	16	1.6268	10	20.67	2
Kumaraswamy	0.07086	15	2.911	11	21.497	3
Weibull	0.08077	17	3.8797	12	26.76	8
Gen. Pareto	0.0907	19	4.4919	13	24.714	6
Gamma (3P)	0.10153	20	4.832	14	29.018	11
Burr (4P)	0.04845	2	5.0591	15	N/A	
Dagum (4P)	0.05049	6	5.2561	16	N/A	
Gamma	0.08375	18	5.263	17	51.643	18
Pearson 6 (4P)	0.05263	8	5.281	18	N/A	
Gen. Gamma (4P)	0.05782	13	5.5625	19	N/A	
Gen. Gamma	0.10647	25	5.9667	20	42.945	15
Weibull (3P)	0.07056	14	6.2615	21	N/A	
Gen. Extreme Value	0.10711	26	7.2496	22	46.746	17
Frechet	0.10452	22	9.2774	23	44.069	16
Beta	0.10534	23	9.4074	24	N/A	
Frechet (3P)	0.10366	21	10.92	25	N/A	
Levy (2P)	0.13365	27	13.314	26	56.863	20
Fatigue Life	0.1658	31	14.468	27	56.221	19
Pearson 5 (3P)	0.15031	28	14.793	28	70.919	22
Inv. Gaussian (3P)	0.16197	30	18.618	29	65.941	21
Exponential	0.19289	34	26.002	30	89.769	23

Activity Name: Send Notification Recipients

Document ID	WF ID	Time (hrs)	Document ID	WF ID	Time (hrs)
TRD-00-00001	1118	0.138	…	…	…
TRD-00-00063	1129	0.835	…	…	…
TRD-00-00002	1130	0.401	TRD-00-00865	2959	0.084
TRD-00-00003	1131	9.026	TRD-00-00867	2969	0.077
TRD-00-00005	1133	0.022	TRD-00-00867	2969	0.077
TRD-00-00006	1134	0.064	TRD-00-00868	2970	0.031
TRD-00-00007	1135	0.032	TRD-00-00868	2970	0.031
TRD-00-00008	1136	0.016	TRD-00-00869	2975	0.081
TRD-00-00009	1137	0.018	TRD-00-00871	2978	0.059
TRD-00-00010	1138	0.013	TRD-00-00872	2979	0.029
TRD-00-00010	1138	0.013	TRD-00-00873	2981	0.033
TRD-00-00012	1141	0.319	TRD-00-00874	2984	0.045
TRD-00-00013	1142	0.062	TRD-00-00874	2984	0.045
TRD-00-00013	1142	0.062	TRD-00-00875	2986	0.026
TRD-00-00014	1143	0.005	TRD-00-00876	2987	0.042
TRD-00-00014	1143	0.005	TRD-00-00877	2990	0.035
TRD-00-00015	1144	0.019	TRD-00-00878	2991	0.035
TRD-00-00015	1144	0.019	TRD-00-00879	2992	0.024
TRD-00-00016	1145	0.009	TRD-00-00880	2993	0.024
TRD-00-00016	1145	0.009	TRD-00-00881	2994	0.036
TRD-00-00018	1146	0.036	TRD-00-00882	2995	0.022
TRD-00-00017	1147	1.664	TRD-00-00883	2996	0.031
TRD-00-00019	1149	0.259	TRD-00-00884	3000	0.022
TRD-00-00019	1149	0.259	TRD-00-00885	3009	0.042
TRD-00-00020	1153	0.051	TRD-00-00886	3014	0.040
TRD-00-00020	1153	0.051	TRD-00-00889	3028	0.060
TRD-00-00044	1155	0.009	TRD-00-00890	3031	0.047
TRD-00-00044	1155	0.009	TRD-00-00891	3032	1.292
…	…	…	TRD-00-00892	3036	0.903
…	…	…	TRD-00-00893	3041	0.943

Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Burr	0.05321	5	1.1155	1	27.773	8
Pearson 5 (3P)	0.04865	1	1.1535	2	23.172	4
Pearson 6 (4P)	0.05173	3	1.2359	3	20.612	2
Frechet (3P)	0.05107	2	1.2366	4	21.326	3
Dagum (4P)	0.05183	4	1.2702	5	20.414	1
Pearson 5	0.06126	7	1.9025	6	27.815	9
Pearson 6	0.05856	6	1.9115	7	27.628	7
Log-Pearson 3	0.06549	10	1.9998	8	25.216	6
Frechet	0.06457	9	2.165	9	24.217	5
Inv. Gaussian (3P)	0.08799	11	3.2804	10	33.605	10
Log-Logistic (3P)	0.06255	8	3.3788	11	35.547	11
Lognormal (3P)	0.10191	13	5.8462	12	52.9	12
Inv. Gaussian	0.11172	14	7.7236	13	62.947	14
Levy (2P)	0.12205	16	10.644	14	59.142	13
Log-Logistic	0.12818	18	11.815	15	67.79	15
Burr (4P)	0.09912	12	12.155	16	N/A	
Gen. Pareto	0.11943	15	13.274	17	91.51	18
Lognormal	0.13463	20	13.411	18	75.787	16
Pareto 2	0.12466	17	14.034	19	97.334	20
Gen. Extreme Value	0.13355	19	15.397	20	89.139	17
Gen. Gamma (4P)	0.15033	21	17.266	21	136.41	23
Weibull (3P)	0.16297	22	25.187	22	N/A	
Levy	0.18606	23	27.082	23	96.684	19
Fatigue Life (3P)	0.21359	26	28.61	24	114.74	21
Weibull	0.20497	24	37.965	25	177.61	29
Gamma (3P)	0.22985	29	38.657	26	294.32	33
Pareto	0.20575	25	38.889	27	156.51	27
Fatigue Life	0.25571	30	45.725	28	153.79	26
Beta	0.26571	31	54.428	29	N/A	
Cauchy	0.22299	28	57.443	30	147.67	25

Appendix A-2
Work-item Status Definitions

Code	Name
AN	Assign
OW	Ownership Taken
RA	Reassign
DE	Delegate
HO	Hold
SP	Skip
UP	Update
CS	Change Status and Save Data
AB	Abort
PA	Pause
RS	Resume
RH	Release Hold
QC	Queue Closed
CO	WorkItem Completed
TO	Timeout, WorkItem closed when timeout happened
CR	WorkItem Created
AC	WorkItem Closed As Min/Max Ownership Count Satisfied
WC	WorkItem Closed as decision was taken based on properties
AW	WorkItem waiting for user action
IO	Information
ID	Information Deleted
CP	Change Priority
AA	Action taken by Assignee
AD	Action taken by Delegate
NQ	New Queue
QA	Queue Allocated
QW	Queue Item Allocated
IC	Queue WorkItem closed
CL	Queue Closed
QR	Queue Removed
QT	Queue Timeout

Appendix B
Self-Adapting Workflow Data

Delegation related data

Note: Names have been changed to protect privacy as per the non-confidentiality agreement

WF ID	Activity	Delegated by	Date	Time	Delegate To	Date
2173	Verify Details	JuditPol	3/30/2012	0:42:53	GaryKasp	3/30/2012
2174	Verify Details	JuditPol	3/30/2012	8:31:58	GaryKasp	3/30/2012
2180	Verify Details	JuditPol	3/30/2012	20:21:05	GaryKasp	3/30/2012
2186	Verify Details	JuditPol	3/31/2012	21:27:30	GaryKasp	3/31/2012
2189	Verify Details	JuditPol	4/1/2012	14:56:06	GaryKasp	4/1/2012
2190	Verify Details	JuditPol	4/1/2012	15:59:44	AnatKarp	4/1/2012
2191	Verify Details	JuditPol	4/1/2012	10:07:11	GaryKasp	4/1/2012
2192	Verify Details	JuditPol	4/1/2012	16:21:42	GaryKasp	4/1/2012
2193	Verify Details	JuditPol	4/1/2012	17:12:07	GaryKasp	4/1/2012
2194	Verify Details	JuditPol	4/1/2012	17:20:55	GaryKasp	4/1/2012
2195	Verify Details	JuditPol	4/1/2012	18:01:36	GaryKasp	4/1/2012
2201	Verify Details	JuditPol	4/2/2012	10:33:40	GaryKasp	4/2/2012
2202	Verify Details	JuditPol	4/2/2012	11:35:10	GaryKasp	4/2/2012
2202	Respond	NataPog	4/10/2012	15:30:08	AlexaKos	4/10/2012
2203	Verify Details	JuditPol	4/2/2012	12:13:54	GaryKasp	4/2/2012
2204	Verify Details	JuditPol	4/2/2012	12:22:37	GaryKasp	4/2/2012
2205	Verify Details	JuditPol	4/2/2012	12:42:18	GaryKasp	4/2/2012
2207	Verify Details	JuditPol	4/2/2012	12:52:10	GaryKasp	4/2/2012
2207	Respond	NataPog	4/6/2012	15:24:25	AlexaKos	4/6/2012
2211	Verify Details	JuditPol	4/2/2012	13:28:23	GaryKasp	4/2/2012
2212	Verify Details	JuditPol	4/2/2012	19:50:10	AnatKarp	4/2/2012
2213	Verify Details	JuditPol	4/2/2012	14:05:58	GaryKasp	4/2/2012
2214	Verify Details	JuditPol	4/2/2012	14:44:06	GaryKasp	4/2/2012
2215	Verify Details	JuditPol	4/2/2012	14:54:21	GaryKasp	4/2/2012
2216	Verify Details	JuditPol	4/2/2012	15:15:29	GaryKasp	4/2/2012
2218	Verify Details	JuditPol	4/2/2012	15:30:11	GaryKasp	4/2/2012
2220	Verify Details	JuditPol	4/2/2012	15:57:36	GaryKasp	4/2/2012
2221	Verify Details	JuditPol	4/2/2012	16:13:40	GaryKasp	4/2/2012
2222	Verify Details	JuditPol	4/2/2012	16:27:12	GaryKasp	4/2/2012

2223	Verify Details	JuditPol	4/2/2012	17:49:51	GaryKasp	4/2/2012
2224	Verify Details	JuditPol	4/2/2012	18:09:47	GaryKasp	4/2/2012
2240	Verify Details	PaulMorph	4/6/2012	16:42:11	AnatKarp	4/6/2012
2240	Verify Details	JuditPol	4/6/2012	16:42:11	GaryKasp	4/6/2012
2241	Verify Details	JuditPol	4/6/2012	18:40:47	GaryKasp	4/6/2012
2241	Verify Details	PaulMorph	4/7/2012	0:40:47	AnatKarp	4/7/2012
2241	Verify Details	PaulMorph	4/7/2012	15:03:35	AnatKarp	4/7/2012
2241	Verify Details	JuditPol	4/7/2012	15:03:35	GaryKasp	4/7/2012
2242	Verify Details	PaulMorph	4/7/2012	17:26:07	AnatKarp	4/7/2012
2242	Verify Details	JuditPol	4/7/2012	17:26:07	GaryKasp	4/7/2012
2242	Respond	NataPog	4/10/2012	16:27:28	AlexaKos	4/10/2012
2243	Verify Details	PaulMorph	4/7/2012	17:04:14	AnatKarp	4/7/2012
2243	Verify Details	JuditPol	4/7/2012	17:04:14	GaryKasp	4/7/2012
2245	Verify Details	PaulMorph	4/7/2012	19:27:47	AnatKarp	4/7/2012
2245	Verify Details	JuditPol	4/7/2012	19:27:47	AnatKarp	4/7/2012
2246	Verify Details	JuditPol	4/7/2012	15:20:53	GaryKasp	4/7/2012
2246	Verify Details	PaulMorph	4/7/2012	21:20:53	AnatKarp	4/7/2012
2247	Verify Details	PaulMorph	4/7/2012	16:21:11	AnatKarp	4/7/2012
2247	Verify Details	JuditPol	4/7/2012	16:21:11	GaryKasp	4/7/2012
2248	Verify Details	JuditPol	4/7/2012	16:56:01	GaryKasp	4/7/2012
2248	Verify Details	PaulMorph	4/7/2012	22:56:01	AnatKarp	4/7/2012
2251	Verify Details	PaulMorph	4/7/2012	18:28:33	AnatKarp	4/7/2012
2251	Verify Details	JuditPol	4/7/2012	18:28:33	GaryKasp	4/7/2012
2252	Verify Details	PaulMorph	4/8/2012	16:28:04	AnatKarp	4/8/2012
2252	Verify Details	JuditPol	4/8/2012	16:28:04	GaryKasp	4/8/2012
2254	Verify Details	JuditPol	4/9/2012	8:42:40	GaryKasp	4/9/2012
2254	Verify Details	PaulMorph	4/9/2012	14:42:40	AnatKarp	4/9/2012
2255	Verify Details	PaulMorph	4/9/2012	10:08:06	AnatKarp	4/9/2012
2255	Verify Details	JuditPol	4/9/2012	10:08:06	GaryKasp	4/9/2012
2256	Verify Details	PaulMorph	4/9/2012	14:04:07	AnatKarp	4/9/2012
2256	Verify Details	JuditPol	4/9/2012	14:04:07	GaryKasp	4/9/2012
2257	Verify Details	PaulMorph	4/9/2012	14:18:17	AnatKarp	4/9/2012
2257	Verify Details	JuditPol	4/9/2012	14:18:17	GaryKasp	4/9/2012
2258	Verify Details	JuditPol	4/9/2012	14:34:47	GaryKasp	4/9/2012
2258	Verify Details	PaulMorph	4/9/2012	20:34:47	AnatKarp	4/9/2012
2264	Verify Details	PaulMorph	4/9/2012	23:42:55	AnatKarp	4/9/2012
2264	Verify Details	JuditPol	4/9/2012	23:42:55	GaryKasp	4/9/2012
2264	Verify Details	PaulMorph	4/10/2012	13:59:30	AnatKarp	4/10/2012
2264	Verify Details	JuditPol	4/10/2012	13:59:30	GaryKasp	4/10/2012

2265	Approved	VishyAn	5/9/2012	14:05:37	PetSvid	5/9/2012
2275	Verify Details	PaulMorph	4/12/2012	22:30:44	AnatKarp	4/12/2012
2276	Verify Details	PaulMorph	4/12/2012	23:52:19	AnatKarp	4/12/2012
2279	Verify Details	PaulMorph	4/13/2012	22:38:51	AnatKarp	4/13/2012
2281	Verify Details	PaulMorph	4/14/2012	16:33:02	AnatKarp	4/14/2012
2403	Verify Details	VishyAn	5/15/2012	16:51:55	PetSvid	5/15/2012
2403	Verify Details	JuditPol	5/15/2012	16:51:55	GaryKasp	5/15/2012
2403	Respond	JuditPol	5/18/2012	19:11:08	GaryKasp	5/18/2012
2422	Verify Details	VishyAn	5/10/2012	7:51:31	PetSvid	5/10/2012
2422	Verify Details	VishyAn	5/10/2012	12:28:54	PetSvid	5/10/2012
2423	Verify Details	VishyAn	5/10/2012	14:00:05	PetSvid	5/10/2012
2423	Verify Details	JuditPol	5/17/2012	7:39:31	GaryKasp	5/17/2012
2441	Verify Details	VishyAn	5/13/2012	14:28:02	PetSvid	5/13/2012
2446	Verify Details	VishyAn	5/14/2012	13:45:30	PetSvid	5/14/2012
2449	Approved	JuditPol	5/17/2012	19:44:04	GaryKasp	5/17/2012
2460	Verify Details	VishyAn	5/15/2012	18:03:12	PetSvid	5/15/2012
2460	Verify Details	JuditPol	5/15/2012	18:03:12	GaryKasp	5/15/2012
2469	Verify Details	JuditPol	5/17/2012	7:49:42	GaryKasp	5/17/2012
2488	Verify Details	JuditPol	5/21/2012	8:54:57	GaryKasp	5/21/2012
2488	Verify Details	JuditPol	5/21/2012	15:47:08	GaryKasp	5/21/2012

Frequent Path Related Data

The following table presents the raw output of respond activities for the two actors considered in Chapter 5, Section 5.5

Actor Name	Work Item	WF ID	Status	Start Time	End Time
SunnyGav	Respond (Responders)	25	CO	40605.68	40605.81
SunnyGav	Respond (Responsible Engineer)	38	CO	40608.99	40609.66
SunnyGav	Respond (Responsible Engineer)	38	CO	40609.86	40609.86
SunnyGav	Respond (Responsible Engineer)	39	CO	40608.99	40609.66
KrisSrik	Respond (Responders)	39	CO	40617.89	40619.77
SunnyGav	Respond (Responsible Engineer)	175	CO	40628.18	40632.67
SunnyGav	Respond (Responders)	195	CO	40630.92	40637.82
SunnyGav	Respond (Responders)	201	CO	40633.63	40633.74
SunnyGav	Respond (Responders)	228	CO	40633.92	40633.92

SunnyGav	Respond (Responders)	228	CO	40634.62	40634.64
KrisSrik	Respond (Responders)	228	TO	40634.88	NULL
KrisSrik	Respond (Responders) Warning	228	CO	40637.96	40639.76
SunnyGav	Respond (Responders)	231	CO	40637.58	40637.84
KrisSrik	Respond (Responders)	231	TO	40638.87	NULL
KrisSrik	Respond (Responders) Warning	231	CO	40639.96	40654.86
SunnyGav	Respond (Responders)	231	CO	40654.88	40654.88
SunnyGav	Respond (Responsible Engineer)	245	CO	40639.6	40639.62
SunnyGav	Respond (Responsible Engineer)	246	CO	40639.6	40639.62
SunnyGav	Respond (Responders)	246	CO	40639.62	40639.62
SunnyGav	Respond (Responders) Warning	246	CO	40641.96	40642.64
KrisSrik	Respond (Responders)	246	TO	40644.7	NULL
KrisSrik	Respond (Responders) Warning	246	CO	40645.06	40645.68
KrisSrik	Respond (Responders)	246	CO	40645.75	40645.75
SunnyGav	Respond (Responsible Engineer)	247	CO	40639.6	40639.61
SunnyGav	Respond (Responders)	326	TO	40652.72	NULL
SunnyGav	Respond (Responders) Warning	326	CO	40653.96	40654.6
KrisSrik	Respond (Responders)	349	TO	40665.69	NULL
KrisSrik	Respond (Responders) Warning	349	CO	40667.96	40669.81
KrisSrik	Respond (Responders)	477	CO	40682.61	40682.79
SunnyGav	Respond (Responders)	477	CO	40687.68	40687.7
SunnyGav	Respond (Responsible Engineer)	553	TO	40690.57	NULL
SunnyGav	Respond (Responsible Engineer) Warning	553	CO	40691.96	40693.58
KrisSrik	Respond (Responders) Warning	635	AB	40708.83	NULL
KrisSrik	Respond (Responders)	640	TO	40708.96	NULL
KrisSrik	Respond (Responders) Warning	640	CO	40710.83	40711.85
KrisSrik	Respond (Responders)	641	TO	40706.89	NULL
KrisSrik	Respond (Responders)	641	TO	40706.89	NULL
KrisSrik	Respond (Responders) Warning	641	WC	40709.58	NULL
KrisSrik	Respond (Responders)	641	CO	40710.6	40710.95
KrisSrik	Respond (Responders)	642	TO	40706.89	NULL
KrisSrik	Respond (Responders)	642	TO	40706.89	NULL
KrisSrik	Respond (Responders)	642	TO	40706.89	NULL
KrisSrik	Respond (Responders) Warning	642	WC	40709.58	NULL
KrisSrik	Respond (Responders)	642	CO	40710.6	40710.79
KrisSrik	Respond (Responders)	647	TO	40708.96	NULL
KrisSrik	Respond (Responders) Warning	647	CO	40710.83	40711.75
KrisSrik	Respond (Responders)	654	TO	40709.86	NULL
KrisSrik	Respond (Responders) Warning	654	CO	40710.96	40721.92

SunnyGav	Respond (Responders)	662	CO	40711.81	40711.81
KrisSrik	Respond (Responders)	664	TO	40710.62	NULL
KrisSrik	Respond (Responders) Warning	664	CO	40711.96	40724.77
KrisSrik	Respond (Responders)	667	CO	40710.6	40715.8
SunnyGav	Respond (Responders)	692	CO	40714.91	40715.99
KrisSrik	Respond (Responders)	693	TO	40715.8	NULL
KrisSrik	Respond (Responders) Warning	693	CO	40716.96	40737.93
KrisSrik	Respond (Responders)	735	TO	40728.59	NULL
SunnyGav	Respond (Responders) Warning	735	CO	40729.96	40760.8
KrisSrik	Respond (Responders)	740	CO	40729.55	40729.71
KrisSrik	Respond (Responders)	741	CO	40729.55	40730.87
SunnyGav	Respond (Responders)	756	CO	40731.57	40731.62
SunnyGav	Respond (Responders)	758	CO	40731.57	40731.62
SunnyGav	Respond (Responders)	760	CO	40731.56	40731.62
KrisSrik	Respond (Responders)	762	TO	40732.72	NULL
KrisSrik	Respond (Responders) Warning	762	CO	40737.96	40739.84
SunnyGav	Respond (Responders)	765	CO	40735.69	40736.64
SunnyGav	Respond (Responders)	765	TO	40735.69	NULL
KrisSrik	Respond (Responders)	765	CO	40735.69	40737.76
KrisSrik	Respond (Responders)	779	CO	40735.74	40736.01
KrisSrik	Respond (Responders)	779	TO	40735.74	NULL
KrisSrik	Respond (Responders)	780	CO	40736.6	40737.78
SunnyGav	Respond (Responders)	781	CO	40736.6	40736.64
KrisSrik	Respond (Responders)	781	TO	40736.6	NULL
KrisSrik	Respond (Responders)	781	CO	40736.6	40736.94
KrisSrik	Respond (Responders) Warning	781	CO	40739.96	40745.98
KrisSrik	Respond (Responders)	783	CO	40736.62	40737.99
SunnyGav	Respond (Responders)	784	CO	40736.6	40736.64
KrisSrik	Respond (Responders)	784	TO	40736.73	NULL
KrisSrik	Respond (Responders) Warning	784	CO	40737.96	40739.83
SunnyGav	Respond (Responders)	785	CO	40736.6	40736.64
KrisSrik	Respond (Responders)	785	TO	40736.6	NULL
SunnyGav	Respond (Responders)	790	CO	40736.62	40736.64
KrisSrik	Respond (Responders)	790	CO	40736.62	40738
KrisSrik	Respond (Responders)	790	TO	40736.62	NULL
SunnyGav	Respond (Responders)	793	CO	40752.74	40753.92
SunnyGav	Respond (Responders)	794	CO	40736.74	40736.8
SunnyGav	Respond (Responders)	794	TO	40736.74	NULL
SunnyGav	Respond (Responders)	836	CO	40749.57	40749.59

SunnyGav	Respond (Responders)	867	CO	40752.7	40753.92
KrisSrik	Respond (Responders)	867	CO	40758.77	40767.8
SunnyGav	Respond (Responders)	921	CO	40765.87	40766.58
KrisSrik	Respond (Responders)	921	CO	40765.87	40766.67
SunnyGav	Respond (Responders)	940	CO	40770.61	40770.62
SunnyGav	Respond (Responders)	941	CO	40770.61	40770.62
SunnyGav	Respond (Responders)	949	CO	40770.57	40770.57
SunnyGav	Respond (Responders)	954	CO	40782.87	40786.6
SunnyGav	Respond (Responders)	954	CO	40782.87	40786.6
SunnyGav	Respond (Responders)	972	CO	40776.66	40777.62
SunnyGav	Respond (Responders)	972	CO	40776.66	40777.79
SunnyGav	Respond (Responders)	996	CO	40778.73	40778.73
KrisSrik	Respond (Responders)	996	CO	40778.74	40781.65
SunnyGav	Respond (Responders)	1099	TO	40804.67	NULL
SunnyGav	Respond (Responders) Warning	1099	CO	40805.96	40807.94
SunnyGav	Respond (Responders)	1099	CO	40812.69	40813.54
SunnyGav	Respond (Responsible Engineer)	1113	TO	40802.94	NULL
SunnyGav	Respond (Responders)	1123	TO	40808.63	NULL
SunnyGav	Respond (Responders) Warning	1123	WC	40809.71	NULL
SunnyGav	Respond (Responders)	1127	CO	40807.92	40807.95
SunnyGav	Respond (Responders)	1163	CO	40821.45	40821.5
KrisSrik	Respond (Responders) Warning	1178	CO	40824.96	40835.92
KrisSrik	Respond (Responders)	1184	CO	40821.46	40823.64
SunnyGav	Respond (Responders)	1184	TO	40823.81	NULL
SunnyGav	Respond (Responders) Warning	1184	CO	40824.96	40830.59
SunnyGav	Respond (Responders)	1192	CO	40820.58	40820.58
SunnyGav	Respond (Responders)	1229	CO	40827.81	40828.47
KrisSrik	Respond (Responders)	1229	TO	40830.5	NULL
KrisSrik	Respond (Responders)	1229	TO	40830.5	NULL
KrisSrik	Respond (Responders) Warning	1229	CO	40833.96	40837.63
SunnyGav	Respond (Responders)	1294	CO	40843.72	40844.62
KrisSrik	Respond (Responders)	1294	CO	40844.67	40847.58
SunnyGav	Respond (Responders)	1326	CO	40850.47	40850.6
KrisSrik	Respond (Responders)	1326	TO	40851.89	NULL
KrisSrik	Respond (Responders) Warning	1326	CO	40856	40856.89
SunnyGav	Respond (Responsible Engineer)	1377	TO	40861.71	NULL
SunnyGav	Respond (Responsible Engineer) Warning	1377	CO	40863.04	40863.57
SunnyGav	Respond (Responsible Engineer)	1381	TO	40861.46	NULL
SunnyGav	Respond (Responsible Engineer) Warning	1381	CO	40863.04	40863.57

SunnyGav	Respond (Responsible Engineer)	1382	TO	40861.71	NULL
SunnyGav	Respond (Responsible Engineer) Warning	1382	CO	40863.04	40863.57
KrisSrik	Respond (Responsible Engineer)	1382	CO	40864.67	40871.17
SunnyGav	Respond (Responsible Engineer)	1382	CO	40871.82	40871.9
SunnyGav	Respond (Responders)	1473	CO	40888.64	40890.7
KrisSrik	Respond (Responders)	1473	CO	40888.64	40890.68
SunnyGav	Respond (Responders)	1499	CO	40884.62	40885.69
KrisSrik	Respond (Responders)	1499	CO	40884.62	40886.05
SunnyGav	Respond (Responders)	1500	CO	40884.62	40884.97
KrisSrik	Respond (Responders)	1500	CO	40884.62	40886.05
SunnyGav	Respond (Responders)	1502	CO	40884.62	40884.97
KrisSrik	Respond (Responders)	1502	CO	40884.62	40886.05
KrisSrik	Respond (Responders)	1504	CO	40886.6	40886.77
SunnyGav	Respond (Responders)	1523	TO	40884.6	NULL
KrisSrik	Respond (Responders)	1523	TO	40884.6	NULL
KrisSrik	Respond (Responders) Warning	1523	WC	40889.71	NULL
SunnyGav	Respond (Responders) Warning	1523	CO	40890	40890.7
KrisSrik	Respond (Responders)	1523	CO	40896.99	40897.01
SunnyGav	Respond (Responders)	1525	CO	40884.6	40884.87
KrisSrik	Respond (Responders)	1525	CO	40884.6	40885.83
SunnyGav	Respond (Responders)	1526	CO	40883.76	40883.77
KrisSrik	Respond (Responders)	1526	CO	40883.76	40886.73
SunnyGav	Respond (Responders)	1527	TO	40884.59	NULL
KrisSrik	Respond (Responders)	1527	TO	40884.59	NULL
KrisSrik	Respond (Responders) Warning	1527	WC	40889.71	NULL
SunnyGav	Respond (Responders) Warning	1527	CO	40890	40890.69
SunnyGav	Respond (Responders)	1528	CO	40883.75	40883.77
KrisSrik	Respond (Responders)	1528	CO	40883.75	40885.78
SunnyGav	Respond (Responders)	1528	TO	40885.82	NULL
SunnyGav	Respond (Responders) Warning	1528	CO	40887	40890.7
KrisSrik	Respond (Responders)	1529	TO	40884.59	NULL
KrisSrik	Respond (Responders) Warning	1529	CO	40890	40890.77
KrisSrik	Respond (Responders)	1529	TO	40896.99	NULL
KrisSrik	Respond (Responders) Warning	1529	CO	40899	40921.82
KrisSrik	Respond (Responders)	1530	TO	40884.3	NULL
KrisSrik	Respond (Responders)	1530	CO	40884.59	40885.82
KrisSrik	Respond (Responders) Warning	1530	CO	40890	40890.78
SunnyGav	Respond (Responders)	1541	CO	40884.66	40884.76
KrisSrik	Respond (Responders)	1541	CO	40884.66	40884.92

SunnyGav	Respond (Responders)	1542	CO	40884.79	40884.97
KrisSrik	Respond (Responders)	1542	CO	40884.79	40885.74
SunnyGav	Respond (Responders)	1546	TO	40885.23	NULL
KrisSrik	Respond (Responders)	1546	TO	40885.23	NULL
KrisSrik	Respond (Responders) Warning	1546	WC	40886.71	NULL
SunnyGav	Respond (Responders) Warning	1546	CO	40887	40890.69
SunnyGav	Respond (Responders)	1563	CO	40892.96	40892.96
KrisSrik	Respond (Responders)	1563	TO	40892.96	NULL
KrisSrik	Respond (Responders) Warning	1563	CO	40894	40896.69
SunnyGav	Respond (Responders)	1563	CO	40896.88	40896.88
SunnyGav	Respond (Responders)	1566	CO	40892.96	40892.96
KrisSrik	Respond (Responders)	1566	TO	40892.96	NULL
KrisSrik	Respond (Responders) Warning	1566	CO	40894	40897
SunnyGav	Respond (Responders)	1567	CO	40892.96	40892.96
KrisSrik	Respond (Responders)	1567	TO	40892.96	NULL
KrisSrik	Respond (Responders) Warning	1567	CO	40894	40925.73
SunnyGav	Respond (Responsible Engineer)	1592	CO	40891.73	40891.89
SunnyGav	Respond (Responders)	1639	CO	40921.72	40921.93
KrisSrik	Respond (Responders) Warning	1647	CO	40922	40924.95
KrisSrik	Respond (Responders) Warning	1648	CO	40922	40925.77
KrisSrik	Respond (Responders)	1650	TO	40921.74	NULL
KrisSrik	Respond (Responders) Warning	1650	CO	40925	40927.03
SunnyGav	Respond (Responders)	1658	CO	40921.75	40921.94
KrisSrik	Respond (Responders)	1658	TO	40925.62	NULL
KrisSrik	Respond (Responders) Warning	1658	CO	40926	40926.97
SunnyGav	Respond (Responders)	1664	CO	40921.91	40921.94
KrisSrik	Respond (Responders) Warning	1664	CO	40927	40943
SunnyGav	Respond (Responders)	1666	CO	40921.91	40921.93
SunnyGav	Respond (Responders)	1677	CO	40921.88	40921.94
SunnyGav	Respond (Responsible Engineer)	1697	CO	40923.8	40924.7
KrisSrik	Respond (Responders)	1704	TO	40933.88	NULL
KrisSrik	Respond (Responders) Warning	1704	WC	40938.71	NULL
KrisSrik	Respond (Responders) Warning	1704	CO	40939	40940.63
SunnyGav	Respond (Responders)	1708	CO	40932.08	40934.73
SunnyGav	Respond (Responsible Engineer)	1718	TO	40925.82	NULL
SunnyGav	Respond (Responsible Engineer) Warning	1718	CO	40927	40927.9
KrisSrik	Respond (Responders) Warning	1722	CO	40932	40933.99
KrisSrik	Respond (Responders)	1752	CO	40931.88	40934.01
KrisSrik	Respond (Responders)	1754	TO	40936.64	NULL

KrisSrik	Respond (Responders) Warning	1754	CO	40943	40946.58
KrisSrik	Respond (Responders)	1759	TO	40932.67	NULL
KrisSrik	Respond (Responders) Warning	1759	CO	40939	40940.03
KrisSrik	Respond (Responders)	1780	TO	40935.6	NULL
KrisSrik	Respond (Responders)	1780	TO	40935.6	NULL
KrisSrik	Respond (Responders) Warning	1780	CO	40941	40947.65
SunnyGav	Respond (Responders)	1790	CO	40938.81	40938.97
KrisSrik	Respond (Responders)	1796	AB	40938.78	NULL
KrisSrik	Respond (Responders)	1796	AB	40938.78	NULL
KrisSrik	Respond (Responders)	1797	TO	40938.78	NULL
KrisSrik	Respond (Responders) Warning	1797	CO	40941	40945.94
SunnyGav	Respond (Responsible Engineer) Warning	1801	CO	40939	40946.89
SunnyGav	Respond (Responsible Engineer)	1803	CO	40942.67	40942.76
SunnyGav	Respond (Responsible Engineer)	1882	TO	40952.62	NULL
SunnyGav	Respond (Responsible Engineer) Warning	1882	CO	40954	40956.88
KrisSrik	Respond (Responders)	1882	CO	40956.88	40959.86
KrisSrik	Respond (Responders)	1891	CO	40955.68	40955.7
KrisSrik	Respond (Responders)	1892	TO	40959.81	NULL
KrisSrik	Respond (Responders) Warning	1892	CO	40961	40973.82
SunnyGav	Respond (Responsible Engineer)	1919	CO	40960.65	40960.9
SunnyGav	Respond (Responsible Engineer)	1919	TO	40960.9	NULL
SunnyGav	Respond (Responsible Engineer) Warning	1919	CO	40962	40966.64
SunnyGav	Respond (Responsible Engineer)	1953	TO	40963.91	NULL
SunnyGav	Respond (Responsible Engineer) Warning	1953	CO	40965	40966.63
KrisSrik	Respond (Responders)	2002	TO	40970.95	NULL
KrisSrik	Respond (Responders) Warning	2002	CO	40976	40991.82
SunnyGav	Respond (Responders)	2013	CO	40974.65	40977.92
KrisSrik	Respond (Responders)	2013	CO	40974.65	40977.03
KrisSrik	Respond (Responders)	2015	TO	40975.95	NULL
KrisSrik	Respond (Responders) Warning	2015	CO	40980.96	40983.9
KrisSrik	Respond (Responders)	2016	TO	40975.95	NULL
KrisSrik	Respond (Responders) Warning	2016	CO	40980.96	40981.16
KrisSrik	Respond (Responders)	2044	CO	40978.73	40983.88
KrisSrik	Respond (Responders)	2055	TO	40978.71	NULL
KrisSrik	Respond (Responders) Warning	2055	CO	40983.96	40984.74
KrisSrik	Respond (Responders)	2061	CO	40978.71	40983.89
KrisSrik	Respond (Responders)	2077	TO	40981.61	NULL
KrisSrik	Respond (Responders) Warning	2077	CO	40986.96	40988.73
KrisSrik	Respond (Responders)	2078	TO	40981.61	NULL

KrisSrik	Respond (Responders) Warning	2078	CO	40986.96	40988.71
SunnyGav	Respond (Responsible Engineer)	2109	CO	40987.7	40987.74
KrisSrik	Respond (Responders)	2131	CO	40991.78	40994.87
KrisSrik	Respond (Responders)	2135	CO	40991.78	40991.86
KrisSrik	Respond (Responders)	2136	TO	40998.78	NULL
KrisSrik	Respond (Responders) Warning	2136	CO	40998.96	41001.9
KrisSrik	Respond (Responders) Warning	2138	CO	40997.96	41003.66
KrisSrik	Respond (Responders)	2142	TO	40998.78	NULL
KrisSrik	Respond (Responders) Warning	2142	CO	41001.96	41011.93
SunnyGav	Respond (Responders)	2144	CO	40998.77	41001.61
SunnyGav	Respond (Responders)	2150	CO	40998.64	41001.62
KrisSrik	Respond (Responders) Warning	2150	CO	41003.96	41019.8
SunnyGav	Respond (Responders)	2162	CO	40998.78	41001.63
KrisSrik	Respond (Responders)	2162	TO	41001.8	NULL
KrisSrik	Respond (Responders) Warning	2162	CO	41004.96	41010.96
SunnyGav	Respond (Responders)	2189	CO	41001.57	41001.65
KrisSrik	Respond (Responders)	2189	TO	41001.78	NULL
KrisSrik	Respond (Responders) Warning	2189	CO	41006.96	41010.76
SunnyGav	Respond (Responders)	2189	CO	41012.62	41012.63
KrisSrik	Respond (Responders)	2189	TO	41013.41	NULL
KrisSrik	Respond (Responders) Warning	2189	CO	41015.96	41015.98
KrisSrik	Respond (Responders)	2195	CO	41001.78	41001.92
SunnyGav	Respond (Responders)	2202	CO	41009.65	41009.81
SunnyGav	Respond (Responders)	2203	CO	41003.65	41003.87
SunnyGav	Respond (Responsible Engineer)	2206	CO	41002.62	41003.87
SunnyGav	Respond (Responsible Engineer)	2209	CO	41002.62	41003.87
KrisSrik	Respond (Responders)	2215	TO	41005.62	NULL
KrisSrik	Respond (Responders) Warning	2215	CO	41009.96	41012.81
SunnyGav	Respond (Responders)	2242	CO	41012.6	41012.61
SunnyGav	Respond (Responders)	2246	CO	41009.76	41009.81
KrisSrik	Respond (Responders)	2248	CO	41008.6	41010.96
KrisSrik	Respond (Responders)	2255	TO	41009.77	NULL
KrisSrik	Respond (Responders) Warning	2255	CO	41011.96	41012.63
SunnyGav	Respond (Responders)	2258	CO	41009.8	41009.81
KrisSrik	Respond (Responders)	2258	CO	41009.8	41012.8
SunnyGav	Respond (Responsible Engineer)	2282	CO	41015.57	41015.8
SunnyGav	Respond (Responsible Engineer)	2283	CO	41015.76	41015.81
KrisSrik	Respond (Responders)	2305	TO	41018.78	NULL
KrisSrik	Respond (Responders) Warning	2305	CO	41024.96	41033.66

KrisSrik	Respond (Responders)	2305	TO	41036.72	NULL
KrisSrik	Respond (Responders) Warning	2305	CO	41040.96	41043.66
SunnyGav	Respond (Responders)	2339	CO	41029.97	41030.7
KrisSrik	Respond (Responders)	2339	CO	41029.97	41030.89
SunnyGav	Respond (Responders)	2352	CO	41031.96	41032.57
KrisSrik	Respond (Responders) Warning	2403	WC	41060.71	NULL
SunnyGav	Respond (Responders)	2410	CO	41039.95	41040.6
KrisSrik	Respond (Responders)	2410	TO	41039.95	NULL
KrisSrik	Respond (Responders) Warning	2410	CO	41042.96	41051.66
SunnyGav	Respond (Responsible Engineer)	2438	CO	41043.66	41043.68
SunnyGav	Respond (Responsible Engineer)	2438	TO	41044.77	NULL
SunnyGav	Respond (Responsible Engineer) Warning	2438	CO	41044.96	41045.57
SunnyGav	Respond (Responders)	2447	CO	41043.92	41043.93
SunnyGav	Respond (Responders)	2449	CO	41044.72	41044.73
SunnyGav	Respond (Responders)	2461	TO	41047.84	NULL
SunnyGav	Respond (Responders) Warning	2461	CO	41051.96	41052
KrisSrik	Respond (Responders)	2496	CO	41058.16	41060.64
KrisSrik	Respond (Responders)	2520	CO	41058.16	41060.63
KrisSrik	Respond (Responders)	2521	CO	41058.16	41060.63
SunnyGav	Respond (Responsible Engineer)	2581	CO	41066.77	41067.94
SunnyGav	Respond (Responsible Engineer)	2592	CO	41067.98	41068.71
SunnyGav	Respond (Responsible Engineer)	2627	CO	41078.13	41078.65
SunnyGav	Respond (Responsible Engineer)	2653	CO	41081.72	41081.76
SunnyGav	Respond (Responders)	2753	CO	41106.16	41106.75
SunnyGav	Respond (Responsible Engineer)	2871	TO	41140.82	NULL
SunnyGav	Respond (Responsible Engineer) Warning	2871	CO	41141.96	41142.62
KrisSrik	Respond (Responders) Warning	2871	CO	41142.96	41157.89
SunnyGav	Respond (Responsible Engineer)	2878	CO	41141.99	41142.62
SunnyGav	Respond (Responsible Engineer)	3086	CO	41199.63	41199.85

Appendix C
Critical Documents Algorithm Source Code

Compute_factors.py

```
#!/usr/local/bin/bash

"""
Computes factors for a given set of input
filess and writes them to a given output file
"""

import argparse
from contextlib import closing
import loaders



class Normalized(object):
  def __init__(self, values_iterable):
    self._min = min(values_iterable)
    self._max = max(values_iterable)

  def normalize(self, value):
    return (value - self._min) / (self._max - self._min)



def parse_args():
  parser = argparse.ArgumentParser(description="Compute and write
critical document factors")
  parser.add_argument("--fvf", type=str, required=True, help="File
vs file Excel file path")
  parser.add_argument("--rev", type=str, required=True,
help="Revisions Excel file path")
```

```
  parser.add_argument("--rfi", type=str, required=True, help="RFIs
vs files Excel file path")
  parser.add_argument("--times", type=str, required=True,
help="Workflow times Excel file path")
  parser.add_argument("--acc", type=str, required=True,
help="Accesses CSV file path")
  parser.add_argument("--out", type=str, required=True, help="Output
file")
  return parser.parse_args()


def main():
  args = parse_args()
  files = loaders.load_files(args)

  norm_idle_access_percent = Normalized([f.idle_access_percent() for
f in files.itervalues()])
  norm_use_percent = Normalized([f.use_percent() for f in
files.itervalues()])
  max_relations = float(max([f.total_file_relations() for f in
files.itervalues()]))
  centrality = lambda relations: (relations/max_relations)
  max_revisions = float(max([f.num_revisions() for f in
files.itervalues()]))
  variability = lambda revisions: (revisions/max_revisions) ** 2

  with closing(open(args.out, "w")) as fout:
    header = "file_name, idle_access_percent, use_percent,
centrality, variability, combined\n"
    fout.write(header)
    for f in files.itervalues():
      n = f._name
      i =
norm_idle_access_percent.normalize(f.idle_access_percent())
      u = norm_use_percent.normalize(f.use_percent())
      c = centrality(f.total_file_relations())
      v = variability(f.num_revisions())
      combined = sum([i, u, c, v])
      fout.write("{0}, {1}, {2}, {3}, {4}, {5}\n".format(n, i, u, c,
v, combined))


if __name__ == "__main__":
  main()
```

----

count_thresholds.py

```
#!/usr/local/bin/bash
from contextlib import closing
import glob

factor_thresholds = [0.75, 0.83, 0.9, 9999999]
factor_names = ["idle_access", "average_access", "centrality",
"variability", "combined"]

def get_bucket(value, buckets):
  bigger = map(lambda b: value >= b, buckets)
  return bigger.index(False)

def main():
  counts_for_files = {}

  for fname in glob.glob("../out/t*.csv"):
```

```
  with closing(open(fname)) as f:
    lines = f.readlines()

  for line in lines[1:]:
    cells = line.split(",")
    filename = cells[0].strip()

    if filename not in counts_for_files:
      empty_buckets = [0, 0, 0, 0]
      counts_for_files[filename] = [list(empty_buckets),
                                    list(empty_buckets),
                                    list(empty_buckets),
                                    list(empty_buckets),
                                    list(empty_buckets)]

    factors = [float(c.strip()) for c in cells[1:]]
    for (factor_ix, value) in enumerate(factors):
      # scale the combined factor so we can use the same buckets
      if factor_ix == 4:
        value /= 4.0
      bucket_ix = get_bucket(value, factor_thresholds)
      counts_for_files[filename][factor_ix][bucket_ix] += 1

header = "file, factor, <0.75, <0.83, <0.9, <9999999"
print header
for (name, factors) in counts_for_files.iteritems():
  for (factor_ix, buckets) in enumerate(factors):
    line = "{0}, {1}".format(name, factor_names[factor_ix])
    for bucket in buckets:
      line += ", {0}".format(bucket)
    print line
```

```
if __name__ == "__main__":
  main()

--------------------------------------------------------------------
```

generate_accesses.py

```
import argparse
from contextlib import closing
import datetime
import globs
import random
import scipy.stats

"""
This scrip generates access times and durations for a list of files.
The number of access is determined based on project duration and
mean_days_between_accesses.
"""

mean_days_between_accesses = 3.0
access_duration_mean = 1.0/24 # one hour
access_duration_stddev = 0.5/24 # 30 minutes


def parse_args():
  parser = argparse.ArgumentParser(description="Simulate file
accesses")
  parser.add_argument("--num-trials", type=int, required=True,
      help="Number of trials to generate")
  parser.add_argument("--fnames-file", type=str, required=True,
      help="File containing the list of file names")
  parser.add_argument("--out-dir", type=str, required=True,
      help="Directory where outputs will be stored")
  parser.add_argument("--parms-dir", type=str, required=True,
      help="Directory where distribution parms will be written")
  return parser.parse_args()


def generate_interaccess_times(lambda_parm):
  """
  Interaccess times are drawn from an exponential distribution with
shape
  parameter lambda generated randomly (see comment in main() below)
and
  scaled by mean_days_between_accesses
  """
  num_values = int(globs.project_days / mean_days_between_accesses)
  seq = scipy.stats.expon.rvs(scale=1/lambda_parm, size=num_values)
  return [mean_days_between_accesses * s for s in seq]


def generate_access_durations(num_values):
  """
  Access durations come from a normal distribution with mean and
  standard deviation defined parameters as defined in the constants
above
  """
  return list(scipy.stats.norm.rvs(loc=access_duration_mean,

scale=access_duration_stddev,size=num_values))


def main():
  args = parse_args()
  with closing(open(args.fnames_file)) as f:
    lines = f.readlines()
    filenames = [line.strip() for line in lines]

  for trial in range(args.num_trials):
    if trial % 20 == 0:
      print "Generating data for trial {0}".format(trial)

    # Lambda is the "spread" parameter in the exponential
distribution.
    # Compute a random lambda parameters for each file and write
    # them to disk. They are drawn from a uniform distribution on
    # [0.5, 1.5].
    lambdas = []
    parmsfile = "{0}/t{1}.csv".format(args.parms_dir, trial)
    with closing(open(parmsfile, "w")) as parmsout:
      header = "filename,
lambda_for_exponential_distribution_of_interfile_access_times"
      parmsout.write(header)
      for fname in filenames:
        l = random.uniform(0.5, 1.5)
        lambdas.append(l)
        parmsout.write("{0}, {1}\n".format(fname, l))

    # Generate interaccess times and access durations; write them to
disk
    ofname = "{0}/t{1}.csv".format(args.out_dir, trial)
    with closing(open(ofname, "w")) as fout:
      for (fname, l) in zip(filenames, lambdas):
        interaccess_times = generate_interaccess_times(l)
        access_durations =
generate_access_durations(len(interaccess_times))

        outline = ""
        outline += fname
        access_time = globs.project_start
        for (t, d) in zip(interaccess_times, access_durations):
          access_time += datetime.timedelta(days=t)
          end_time = access_time + datetime.timedelta(days=d)
          if end_time > globs.project_end:
            break
          outline += ", {0},
{1}".format(access_time.strftime(globs.date_format), d)
          access_time = end_time
        outline += "\n"
        fout.write(outline)


if __name__ == "__main__":
  main()
--------------------------------------------------------------------
```

gen.sh (shell script to generate the accesses from the above python program)

```
#!/bin/bash

python generate_accesses.py --num-trials 1000 --fnames-file
../in/fnames --out-dir ../in/gen --parms-dir ../out/parms/

--------------------------------------------------------------------
```

Globs.py (Global definitions)

```
import datetime
```

```
date_format = "%Y-%m-%d %H:%M"

project_start = datetime.datetime(2013, 1, 1, 0, 0, 0)
project_end = datetime.datetime(2013, 9, 1, 0, 0, 0)
project_delta = project_end - project_start
seconds_per_day = float(24 * 60 * 60)
project_days = project_delta.days + (float(project_delta.seconds) /
seconds_per_day)
workflow_start_days_interval = 1
--------------------------------------------------------------------
```

Loaders.py

```
from contextlib import closing
import datetime
import globs
import sys
import xlrd


# --- Classes ------------------------------------------------------
----------

class Workflow(object):
  def __init__(self):
    Workflow._time_names = None
    self._workflow_id = None
    self._workflow_name = None
    self._durations_days = None
    self._start_time = None

  def __repr__(self):
    return "(({0}, {1}, {2}, {3}))".format(
        self._workflow_id, self._workflow_name,
self._durations_days, self._start_time)

  def start(self):
    return self._start_time

  def end(self):
    return self._start_time + datetime.timedelta(days=self.days())

  def days(self):
    return sum(self._durations_days)

  def is_time_during_workflow(self, time):
    end_time = self._start_time +
datetime.timedelta(days=self.days())
    return self._start_time <= time and time <= end_time


class File(object):
  def __init__(self):
    self._name = None
    self._accesses = None
    self._workflows = []
    self._revisions = None
    self._file_relations = [] # [(filename, num_times_related), ...]
    self._time_factors_computed = False

  def __repr__(self):
    return "(({0}, {1}, {2}, {3}, {4}, {5}))".format(self._name,
self._accesses,
        self._workflows, self._revisions, self._file_relations,
        self._time_factors_computed)


# workflows may overlap.



# expected idle time = time in the project duration outside of
workflows
# note: workflows may overlap
# actual idle time = expected - access times outside of workflows
# factor = 1 - actual / expected
# if an access is completely inside a workflow it is ignored
# if it is partially inside a workflow

# NORMALIZE these into [0, 1] before computing the combined score
#
# average access takes overlaps as different time
# idle only consider time not idle (overlaps do not count)
  def _compute_time_factors(self):

    # use percentage (average use) = total workflow time / project
duration
    # (overlapping workflows are considered distinct so this factor
could be greater than 1)
    days_in_flows = sum([w.days() for w in self._workflows])
    days_in_flows += sum([a[1] for a in self._accesses])
    self._use_percent = float(days_in_flows) / globs.project_days

    # Compute expected idle time = duration of the project in
    # which the file was not associated with any workflow.
    # This algorithm works because at the moment a workflow starts
we can
```

```
    # compute the gap after the previous workflow by considering
only the
    # most recently ended workflow
    expected_idle_days = 0
    previous_end_time = globs.project_start
    self._workflows.sort(key=lambda workflow: workflow.start())
    for w in self._workflows:
      gap_between_workflows = w.start() - previous_end_time
      if gap_between_workflows > datetime.timedelta(days=0):
        expected_idle_days += gap_between_workflows.days
      previous_end_time = max(previous_end_time, w.end())
    gap_before_project_end = globs.project_end - previous_end_time
    if gap_before_project_end > datetime.timedelta(days=0):
      expected_idle_days += gap_before_project_end.days

    # Compute actual idle time = duration of the project in which
the file
    # was not being accessed and was not associated with any
workflow
    all_accesses = list(self._accesses)
    all_accesses.extend([(w.start(), w.days()) for w in
self._workflows])

    idle_days = 0
    previous_end_time = globs.project_start
    all_accesses.sort(key=lambda access: access[0])
    for a in all_accesses:
      a_start = a[0]
      a_days = a[1]
      a_end = a_start + datetime.timedelta(days=a_days)
      gap_between_accesses = a_start - previous_end_time
      if gap_between_accesses > datetime.timedelta(days=0):
        idle_days += gap_between_accesses.days
      previous_end_time = max(previous_end_time, a_end)
    gap_before_project_end = globs.project_end - previous_end_time
    if gap_before_project_end > datetime.timedelta(days=0):
      idle_days += gap_before_project_end.days

    self._idle_access_percent = 1.0 - (float(idle_days) /
expected_idle_days)

  def idle_access_percent(self):
    if not self._time_factors_computed:
      self._compute_time_factors()
    return self._idle_access_percent

  def use_percent(self):
    if not self._time_factors_computed:
      self._compute_time_factors()
    return self._use_percent

  def total_file_relations(self):
    return sum([n for (f, n) in self._file_relations])

  def num_revisions(self):
    return self._revisions


# --- Helper functions ---------------------------------------------
----------

def assert_interval_in_project(start, days):
  if start < globs.project_start:
    print "Interval start time {0} is before the project start date
{1}".format(
        start, globs.project_start)
    print "You might need to regenerate the inputs"
    sys.exit(1)
  end_time = start + datetime.timedelta(days=days)
  if end_time > globs.project_end:
    print "Interval end time {0} is after the project end date
{1}".format(
        end_time, globs.project_end)
    print "You might need to regenerate the inputs"
    sys.exit(1)


def get_rows_from_sheet(sheet):
  """
  Converts an xlrd sheet to a list of rows like:
  [ row1, row2, ..., rowm]
  where rowi = [celli1, celli2, ..., cellin] for i  = 1, 2, ..., m
  """
  rows = sheet.nrows
  cols = sheet.ncols
  out_rows = []
  for row in range(rows):
    out_row = []
    for col in range(cols):
      out_row.append(sheet.cell(row, col).value)
    out_rows.append(out_row)
  return out_rows


def get_first_sheet_from_path(path):
  """
  Returns the first xlrd sheet from the Excel sheet at path
  """
  book = xlrd.open_workbook(path)
  sheet = book.sheet_by_index(0)
  return sheet


def file_adjacency_matrix_to_edge_list(rows):
  """
  Converts the file adjacency matrix to an edge list

  Input:
  The adjacency matrix in the form:
  [ ["", "file name 1", ..., "file name n"],
    ["file name 1", links(1,1), ..., links(1,n)],
    ...
    ["file name n", links(n,1), ..., links(n,n)] ]

  Output:
  File adjacency edge list in the form:
  [(file name i, file name j, links(i, j)), ...]
  """
  header_row = rows[0]
  header_col = [row[0] for row in rows]
  links = []
  added_edges = {}
  for (row_ix, row) in enumerate(rows):
    for (col_ix, val) in enumerate(row):
      # Ignore cells containing non-integer values
      try:
        weight = int(rows[row_ix][col_ix])
```

```
   except ValueError:
    continue

   # Minimum non-zero weight is 1
   if weight > 0:
    v1 = str(header_col[row_ix])
    v2 = str(header_row[col_ix])

    # Don't add the same edge twice; validate the input file
    if (v1, v2) in added_edges:
     w = added_edges[(v1, v2)]
     if w != weight:
      print "link {0}--{1} appears in adjacency list with 
different weights: {2} and {3}".format(v1, v2, weight, w)
      sys.exit(1)
     continue
    if (v2, v1) in added_edges:
     w = added_edges[(v2, v1)]
     if w != weight:
      print "link {0}--{1} appears in adjacency list with 
different weights: {2} and {3}".format(v1, v2, weight, w)
      sys.exit(1)
     continue

    # Add the edge
    link = (v1, v2, weight)
    links.append(link)
    added_edges[(v1, v2)] = weight

 return links
```

```
def related_file_edge_list(path):
  """
  Returns a file adjacency edge list in the form:
  [(file name i, file name j, links(i, j)), ...]
  Each edge is included only once, so if edge (i, j) is in the list
then
  edge (j, i) will not be.
  Assumes the matrix is in the first sheet of the Excel sheet at
path.
  """
  sheet = get_first_sheet_from_path(path)
  rows = get_rows_from_sheet(sheet)
  return file_adjacency_matrix_to_edge_list(rows)

# --- Loader functions -------------------------------------------
----------

def load_accesses(files, path):
  """
  Loads File access lists from the CSV at path, allocates a new File
for each
  list and adds them to the files dict
  """
  with closing(open(path, "r")) as f:
    lines = f.readlines()
    for line in lines:
      cells = line.split(",")
      file_name = str(cells[0])
      access_list = []
      for cell_ix in range(1, len(cells), 2):
        access_time =
datetime.datetime.strptime(cells[cell_ix].strip(),
globs.date_format)
        access_days = float(cells[cell_ix + 1].strip())
        assert_interval_in_project(access_time, access_days)
        access_list.append((access_time, access_days))
      if file_name in files:
        print "Error: {0} appears more than once in
{1}".format(file_name, path)
        sys.exit(1)
      files[file_name] = File()
      files[file_name]._name = file_name
      files[file_name]._accesses = access_list


def load_revisions(files, path):
  """
  Adds revisions from the Excel sheet at path to the Files in dict
files
  """
  sheet = get_first_sheet_from_path(path)
  rows = get_rows_from_sheet(sheet)
  header = rows[0]
  if header[0] != "" or header[1] != "" or header[2] != "# of
revisions":
    print "Found unexpected header row in the docs and revisions
file: {0}".format(header)
    sys.exit(1)
  for row in rows[1:]:
    file_name = str(row[1])
    revisions = int(row[2])
    if file_name not in files:
      files[file_name] = File()
      files[file_name]._name = file_name
    if files[file_name]._revisions is not None:
      print "Error: {0} appears more than once in the revisions file
{1}".format(file_name, path)
      sys.exit(1)
    files[file_name]._revisions = revisions


def load_related_files(files, path):
  edges = related_file_edge_list(path)
  for (f1, f2, times_related) in edges:
    if f1 not in files:
      files[f1] = File()
      files[f1]._name = f1
    files[f1]._file_relations.append((f2, times_related))
    if f2 not in files:
      files[f2] = File()
      files[f2]._name = f2
    files[f2]._file_relations.append((f1, times_related))


def load_workflows(path):
  """
  Loads Workflows from the first sheet of the Excell book at path
  """
  sheet = get_first_sheet_from_path(path)
  rows = get_rows_from_sheet(sheet)
  if rows[0][0] != "requestforinformation_id" or rows[0][1] !=
"title":
    print "Found unexpected header rows in activity time sheet"
    print rows[0]
    sys.exit(1)
  time_names = [str(c) for c in rows[0][2:7]]
  Workflow._time_names = time_names

  workflows = {}
  start_time = globs.project_start
  for row in rows[1:]:
    try:
      workflow_id = int(row[0])
    except ValueError:
      # Ignore the footer rows at the bottom of the sheet
      continue
    hours_per_day = 24.0
    workflow_name = str(row[1])
    workflow_durations_days = [float(t) / hours_per_day for t in
row[2:7]]

    if workflow_id in workflows:
      print "Error: workflow_id {0} appears more than once in the
activity time sheet".format(
          workflow_id)
      sys.exit(1)

    workflow = Workflow()
    workflow._workflow_id = workflow_id
    workflow._workflow_name = workflow_name
    workflow._durations_days = workflow_durations_days
    workflow._start_time = start_time
    # We assume that a new workflow starts every
workflow_start_days_interval days
    start_time +=
datetime.timedelta(days=globs.workflow_start_days_interval)
```

```
    workflows[workflow_id] = workflow

  return workflows


def load_workflows_for_files(files, workflows, path):
  """
  Appends to workflows containing the File to the _workflows list
for each File
  in files. Relationships are loaded from the RFI vs file Excel
sheet at path
  """
  sheet = get_first_sheet_from_path(path)
  rows = get_rows_from_sheet(sheet)
  if rows[0][1] != "Document" or rows[1][0] !=
"requestforinformation_id" or rows[1][1] != "title":
    print "Found unexpected header rows in RFIs vs Docs sheet"
    print rows[0]
    print rows[1]
    sys.exit(1)
  filenames = rows[0][2:]
  for row in rows[2:]:
    workflow_id = int(row[0])
    #workflow_title = str(row[1]) ignore this. We get it from time
times for activities sheet
    attached_file_counts = [int(count) for count in row[2:]]
    attached_filename_indexes = [ix for (ix, count) in
enumerate(attached_file_counts)
                                 if count > 0]
    attached_filenames = [str(filenames[ix]) for ix in
attached_filename_indexes]
    for f in attached_filenames:
      if f not in files:
        files[f] = File()
        files[f]._name = f
      files[f]._workflows.append(workflows[workflow_id])


def load_files(args):
  """
  Constructs File objects for input data. Returns a dict of Files:
  {file_name: file_object, ...}
  """
  files = {}
  load_accesses(files, args.acc)
  load_revisions(files, args.rev)
  load_related_files(files, args.fvf)
  workflows = load_workflows(args.times)
  load_workflows_for_files(files, workflows, args.rfi)
  return files


if __name__ == "__main__":
  # for testing
  files = {}
  #load_accesses(files, "../in/gen/t0000001")
  #load_revisions(files, "../in/Docs_&_Revisions.xlsx")
  #load_related_files(files, "../in/Docs_vs_Docs.xlsx")
  workflows = load_workflows("../in/times_for_activities.xlsx")
  load_workflows_for_files(files, workflows,
"../in/RFIs_vs_Docs.xlsx")
  print files

-------------------------------------------------------------------
```

Make_adjacency.py

```
#!/usr/local/bin/python
from contextlib import closing
import loaders

def strip_filename(filename):
  """
  Sanitize strings for systems we output data to.
  E.g. GraphVis doesn't like node identifiers with . in them
  """
  bad_chars = ['-', '.']
  lofc = [(c if c not in bad_chars else "_") for c in filename]
  return ''.join(lofc)

def write_adjacency_graph(adjacency_list, filename):
  """
  Writes a "neato" file to be converted by GraphViz using a command
like:
  neato -T pdf adj.neato -o adj.pdf
  """
  with closing(open(filename, "w")) as f:
    f.write("graph {")
    f.write("overlap = scale;")
    for edge in adjacency_list:
      thickness = max(1, float(edge[2]) / 4)
      line = "{0} -- {1}
[penwidth={2}];\n".format(strip_filename(edge[0]),

strip_filename(edge[1]), thickness)
      f.write(line)
    f.write("}")
```

```
def main():
  adj_list =
loaders.related_file_edge_list('../in/Docs_vs_Docs.xlsx')
  write_adjacency_graph(adj_list, "adj.neato")

if __name__ == "__main__":
  main()
```

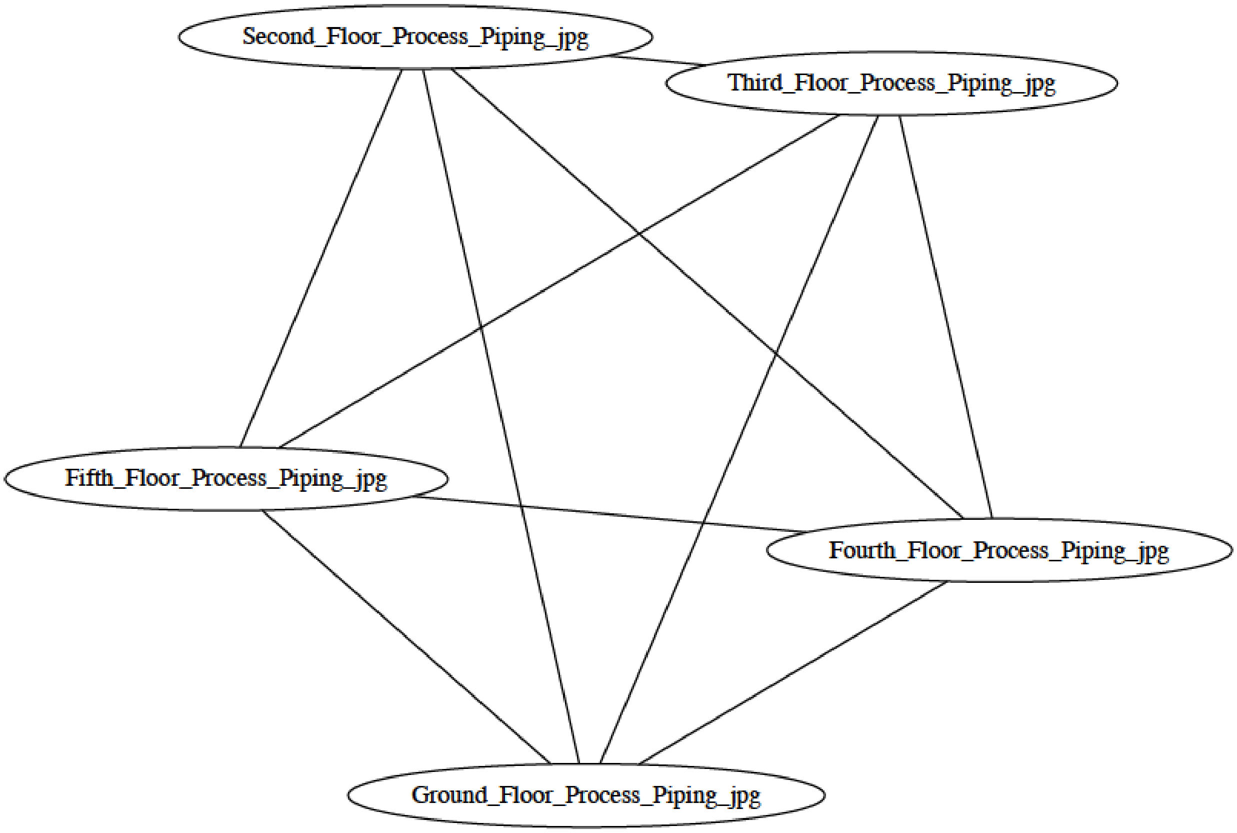
Run.sh

```
#!/bin/bash
# Runs the compute factors script once for each access times file

for infile in ../in/gen/t*.csv ; do

     echo Computing factors for $infile

     outfile=../out/$(basename $infile)

     python compute_factors.py \
           --fvf "../in/Docs_vs_Docs.xlsx" \
           --rev "../in/Docs_&_Revisions.xlsx" \
           --rfi "../in/RFIs_vs_Docs.xlsx" \
           --times "../in/times_for_activities.xlsx" \
           --acc "$infile" \
           --out "$outfile"

done
```

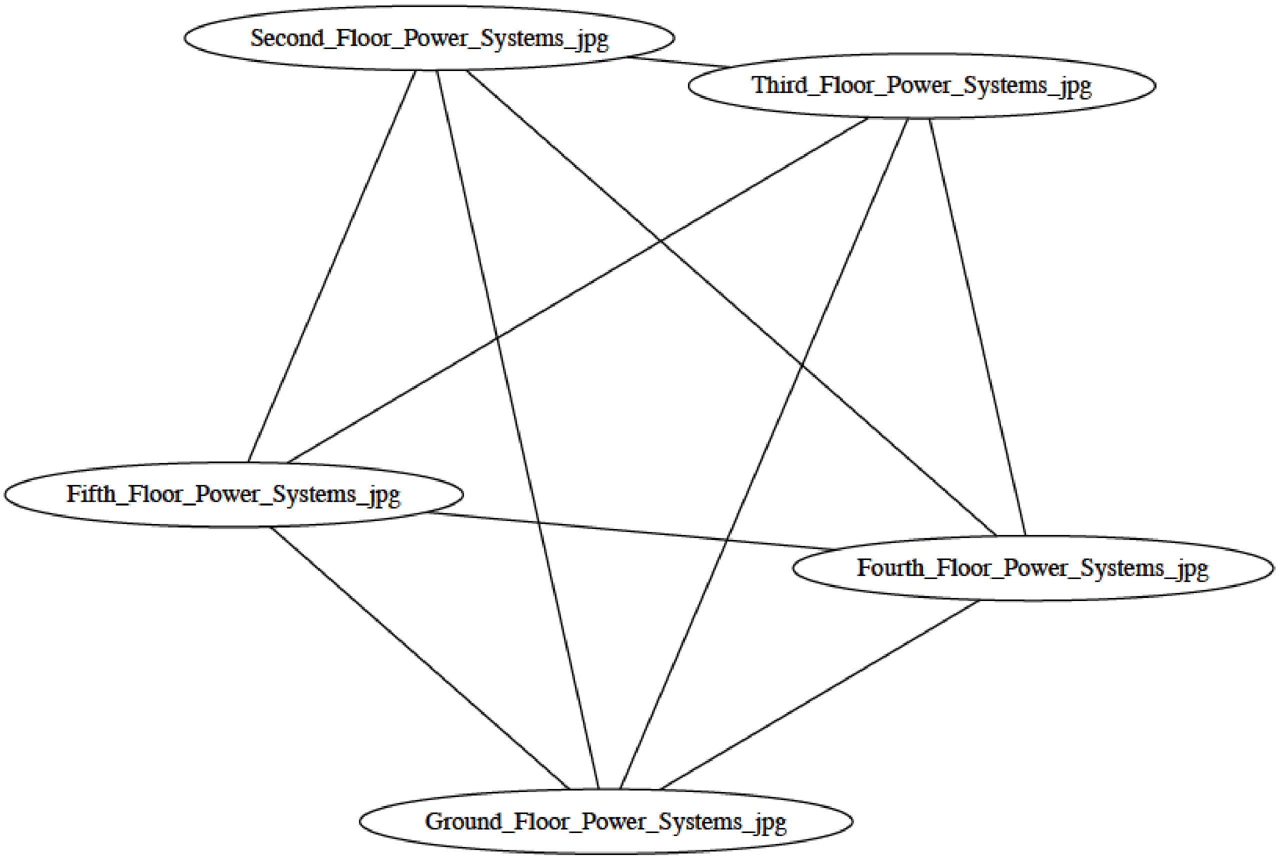
Appendix D

Degree Centrality Derived Document Network Graph From Model Project Scenario in High Resolution

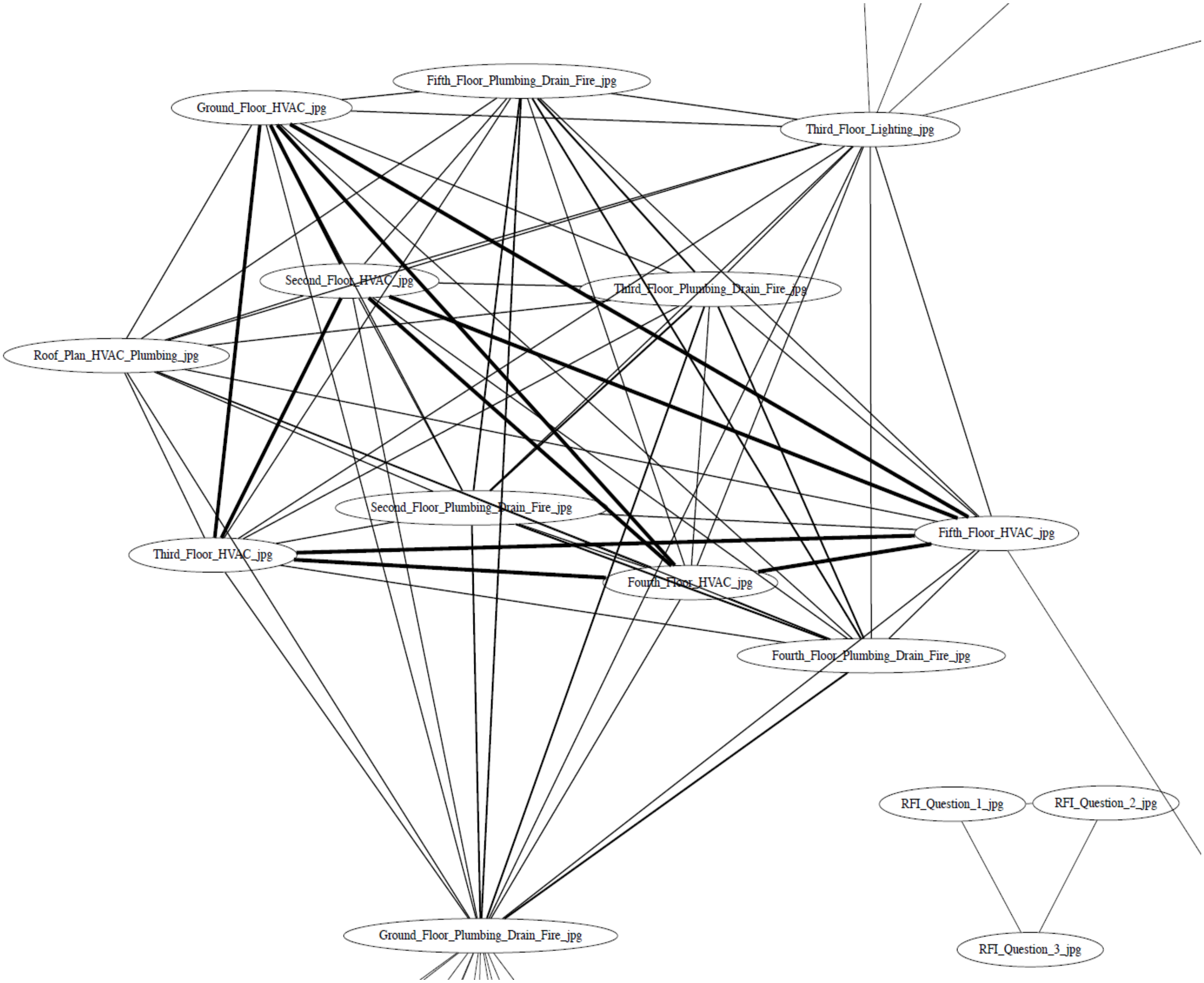
First Set of Five Isolated Interlinked Documents

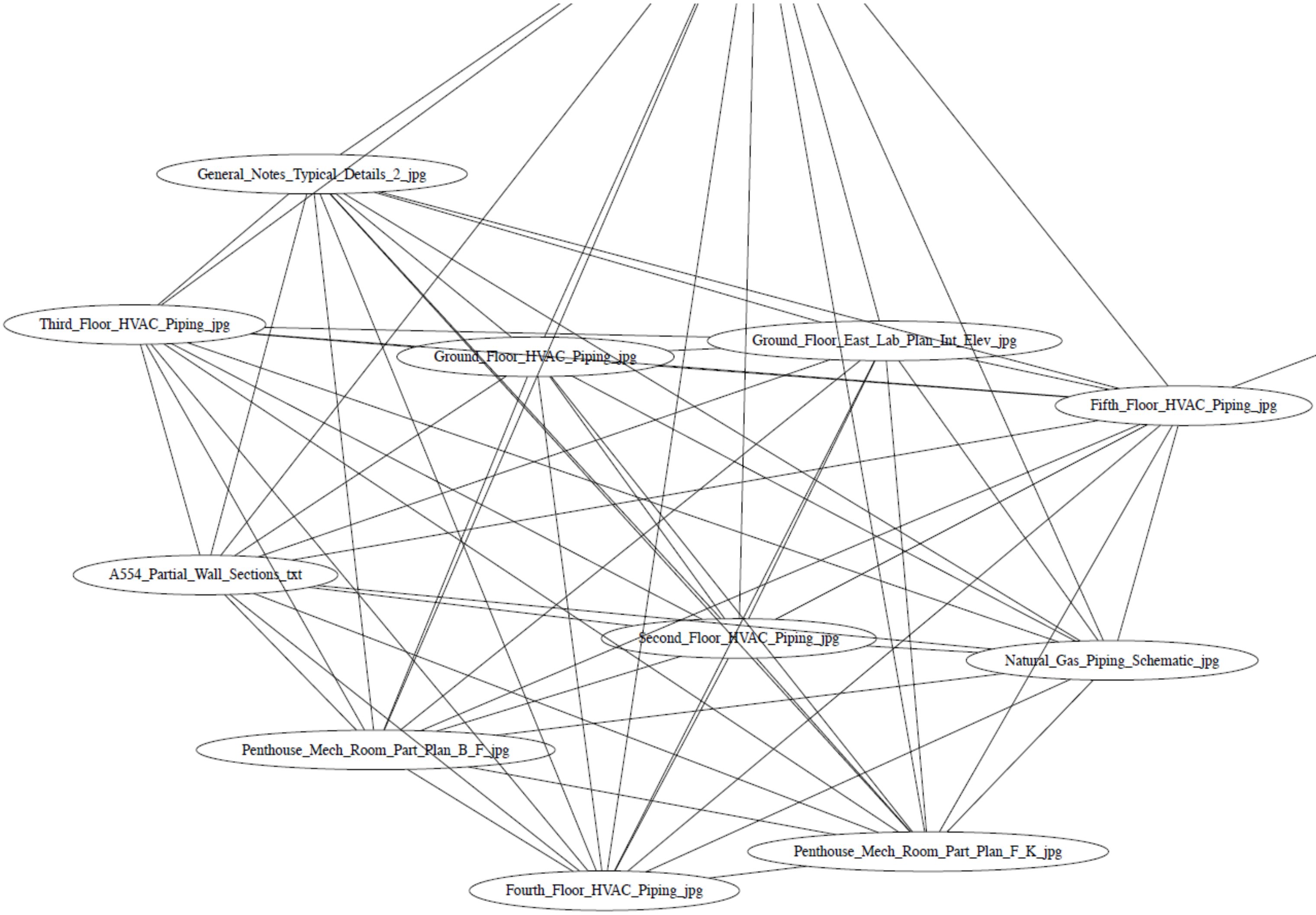

Second_Floor_Process_Piping_jpg
Third_Floor_Process_Piping_jpg
Fifth_Floor_Process_Piping_jpg
Fourth_Floor_Process_Piping_jpg
Ground_Floor_Process_Piping_jpg

Second Set of Five Isolated Interlinked Documents


Second_Floor_Power_Systems_jpg
Third_Floor_Power_Systems_jpg
Fifth_Floor_Power_Systems_jpg
Fourth_Floor_Power_Systems_jpg
Ground_Floor_Power_Systems_jpg

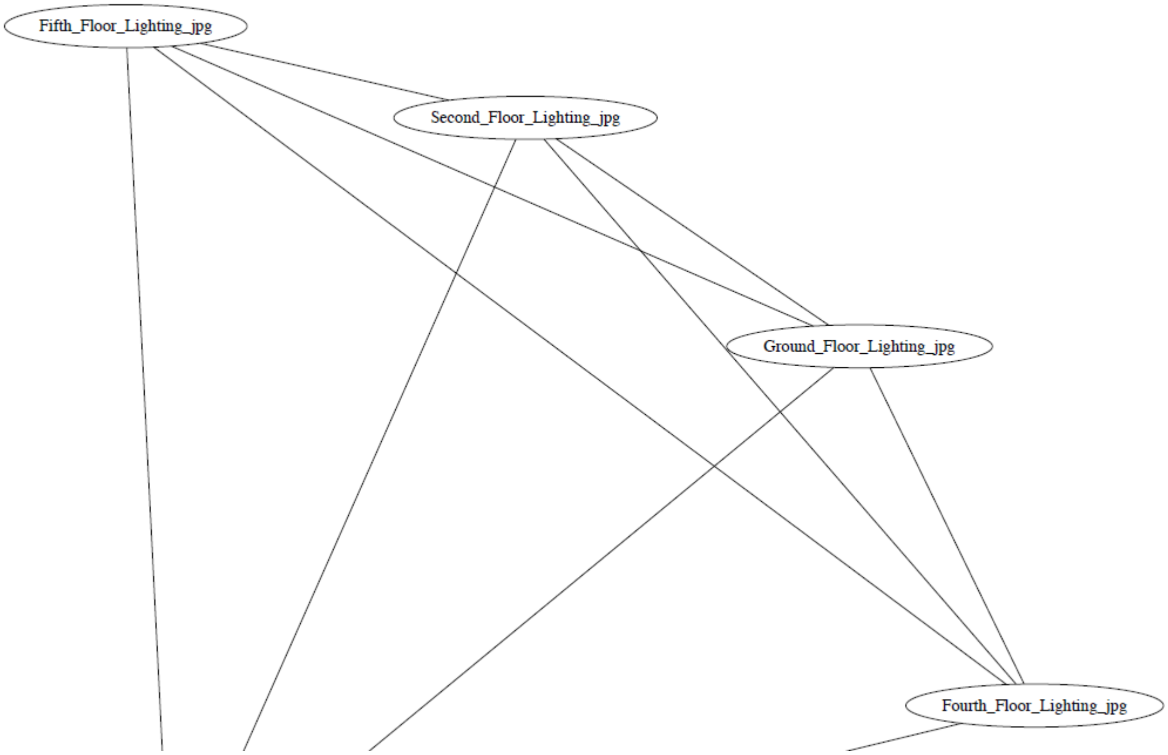
A densely packed cluster of interrelated documents


Fifth_Floor_Plumbing_Drain_Fire_jpg
Ground_Floor_HVAC_jpg
Third_Floor_Lighting_jpg
Second_Floor_HVAC_jpg
Third_Floor_Plumbing_Drain_Fire_jpg
Roof_Plan_HVAC_Plumbing_jpg
Second_Floor_Plumbing_Drain_Fire_jpg
Fifth_Floor_HVAC_jpg
Third_Floor_HVAC_jpg
Fourth_Floor_HVAC_jpg
Fourth_Floor_Plumbing_Drain_Fire_jpg
RFI_Question_1_jpg
RFI_Question_2_jpg
Ground_Floor_Plumbing_Drain_Fire_jpg
RFI_Question_3_jpg

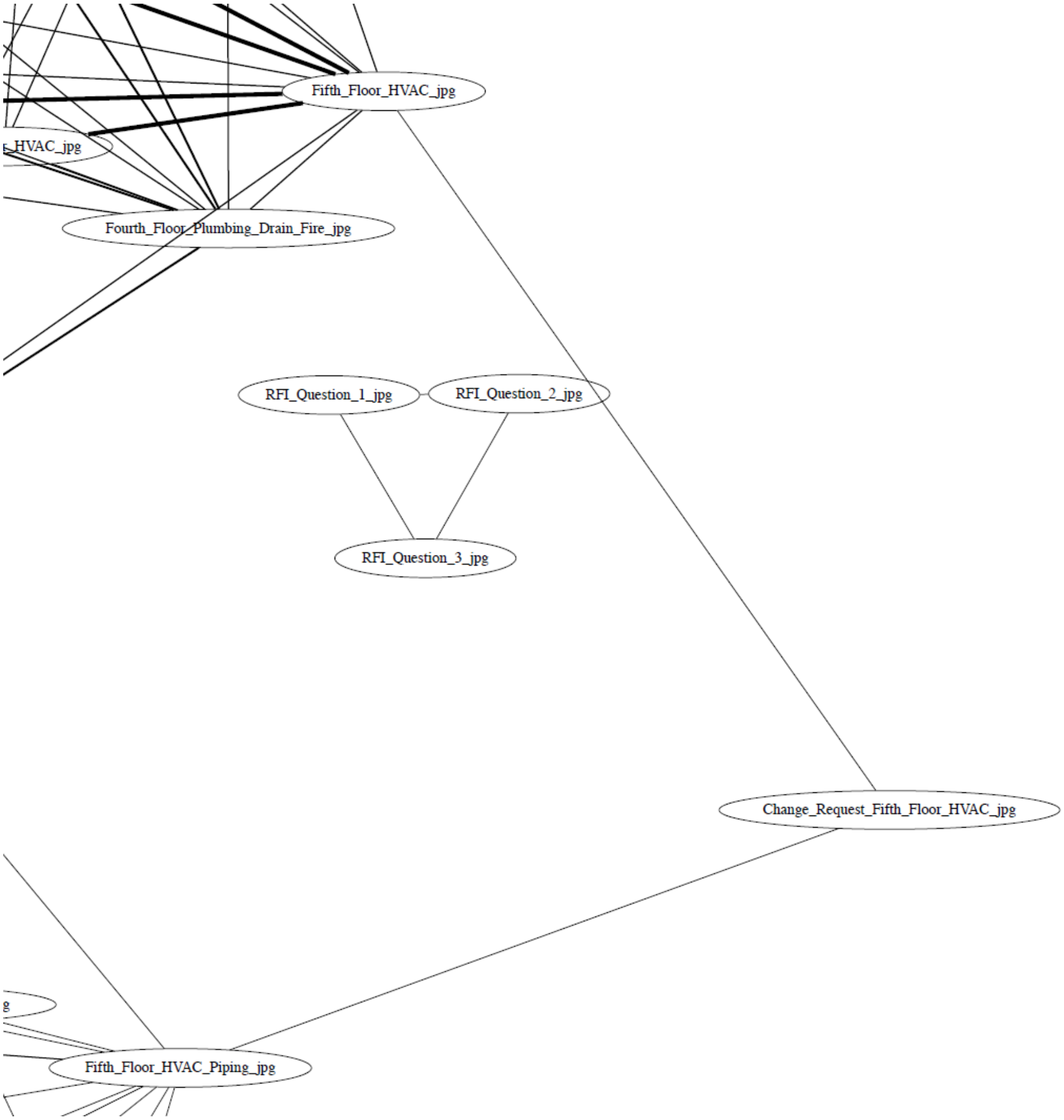

General_Notes_Typical_Details_2_jpg
Third_Floor_HVAC_Piping_jpg
Ground_Floor_East_Lab_Plan_Int_Elev_jpg
Ground_Floor_HVAC_Piping_jpg
Fifth_Floor_HVAC_Piping_jpg
A554_Partial_Wall_Sections_txt
Second_Floor_HVAC_Piping_jpg
Natural_Gas_Piping_Schematic_jpg
Penthouse_Mech_Room_Part_Plan_B_F_jpg
Penthouse_Mech_Room_Part_Plan_F_K_jpg
Fourth_Floor_HVAC_Piping_jpg

The second packed cluster of interrelated documents, though thinner lines indicate less links between document pairs.

Four satellite documents attached to Third_Floor_Lighting in the main cluster:


Fifth_Floor_Lighting_jpg
Second_Floor_Lighting_jpg
Ground_Floor_Lighting_jpg
Fourth_Floor_Lighting_jpg

And finally the connection between Fifth_Floor_HVAC_Piping and Fifth_Floor_HVAC through a change request. There is also an isolated cluster of three interrelated documents


Fifth_Floor_HVAC_jpg
_HVAC_jpg
Fourth_Floor_Plumbing_Drain_Fire_jpg
RFI_Question_1_jpg
RFI_Question_2_jpg
RFI_Question_3_jpg
Change_Request_Fifth_Floor_HVAC_jpg
g
Fifth_Floor_HVAC_Piping_jpg

Appendix E
Questionnaire For Weight Determination

Relative Importance of Factors Used in Determining Critical Documents

We have identified four factors which were used to determine critical documents within a model project scenario. Below we list these factors and provide a brief explanation for each factor:

1) **Document inter-relation score**: This score represents how documents are related to one another within an information system. For example, if two documents are attached to the same workflow, a relation is created between each. A network can then be created showing all the relationships between documents. We evaluate a score based on the network analysis method, degree centrality, which measures all the links coming in to a node within a network.
2) **Document idle time**: We measure the time a document was actually not in use to the time that it was expected to be idle when a project was created. The underlying philosophy here is that if the document was accessed outside of expected access times, then perhaps the document is being accidentally or incorrectly referenced or it is a document that has greater relevance to a project. The overall duration is recorded for each document rather than the number of accesses.
3) **Document Access Statistics**: This refers to the overall access that a document experiences, perhaps from multiple users within and outside of a workflow. Increased access instances indicate that a document is relatively more important to documents that may only be accessed for a specific workflow.
4) **Document Versions**: Every time a file is modified and saved, it is saved as a newer version. The number of versions that a file has can be problematic as incorrect versions may be accessed accidentally. An increased number of versions may also indicate that a particular document evolved as part of a collaborative process between different project participants.

Based on your understanding of the above factors, how would you rank the factors? In addition, give each document a relevant rating score from 1 to 10, where 1 indicates that the factor has no importance while 10 indicates that a factor is extremely relevant to determining the criticality of a document:

Factor	Rank: 1 to 4 (where 1 is most important)	Relevance Rating
Document Inter-Relation Score		
Document Idle Time		
Document Access Statistics		
Document Versions		