A CONTEXT-BASED REPRESENTATION AND REASONING FORMALISM TO SUPPORT CONSTRUCTION SAFETY PLANNING

BY

HAN-HSIANG WANG

DISСЕRNАTION

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Civil Engineering in the Graduate College of the University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Doctoral Committee:

Senior Lecturer Frank Boukamp, Chair, Royal Melbourne Institute of Technology
Associate Professor Liang Y. Liu
Associate Professor Khaled El-Rayes
Associate Professor Allen H. Renear
ABSTRACT

Construction industry has the highest potential for occupational hazard events among all United States’ industries. One of the major causes is construction employers or employees’ negative attitudes toward safety requirements imposed by different construction safety documents, such as construction safety regulations and project safety plans. Such attitudes towards safety requirements include ignorance, negligence and disobedience and the first two can be dealt with by raising construction project participants’ awareness of safety requirements through better construction safety planning.

The huge number of safety requirements from different construction safety documents may hinder project participants from carefully searching through them for identifying applicable safety requirements. In addition, current approaches and tools for raising awareness of safety requirements are not sufficient. For example, traditional keyword-based search through a large number of construction safety specifications can help find safety requirements which contain the search keywords. However, there will be other requirements which are semantically relevant but are not found just because they do not contain these specific keywords.

Hence, there is a need for a formalized approach to automating the identification of applicable construction safety requirements. This approach should enable identifying safety requirements more efficiently (i.e. saving time in the identification process) and effectively (i.e. better identify both directly and inferentially relevant requirements), and make it feasible to more easily raise project participants’ awareness of safety requirements.

To address the above need, a Construction Safety Documents Management Framework is developed in this research. The developed Framework comprises the following components: (1) a computer interpretable model for representing safety requirements of construction safety documents to enable automated reasoning about them; (2) a semantically-rich model for representing concepts acquired from
construction safety documents which describe contextual information to which the imposed requirements apply; and (3) reasoning mechanisms to reason about the above two models for evaluating concepts and safety requirements’ applicability to given project contexts.

For the representation of construction safety requirements, the Ordered Hierarchy of Content Object (OHCO) approach is adopted to build the representation model; Extensible Markup Language (XML) is used in this research to implement the OHCO-based model. Ontological modeling, on the other hand, is leveraged to model semantically-rich concepts that describe construction contexts. In addition, the developed reasoning mechanisms utilize the ontological relationships between modeled concepts to automatically evaluate each concept’s applicability. Construction safety requirements’ applicability then can be evaluated by reasoning about the requirements’ applicability conditions and exceptions, which are represented using concepts defined in the concept ontologies. Safety requirements can be classified according to their evaluated applicability.

The approach is mainly validated for Job Hazard Analysis documents through computational experiments in multiple representation and reasoning test cases. The validation results show that the developed Framework can successfully evaluate the applicability of safety requirements imposed by these JHA documents and identify safety requirements applicable to given contexts.

During the validation, the advantages, limitations and practical implications of the developed Framework are discovered. The potential directions of future research on improving the Framework to reinforce its capability and to remove its limitations are lastly presented.
To My Parents and My Wife
ACKNOWLEDGEMENTS

I want to thank many people for their help and support throughout my PhD study to make this dissertation possible.

First and foremost, I would like to sincerely thank my academic adviser, Dr. Frank Boukamp. Without his instruction, patience, encouragement, research and personal support, my PhD study would not have been such an enjoyable experience. Working with him has been an invaluable experience, not only in learning how to be a capable researcher, but also in learning how to be a person with a warm heart. Also, his advice and constructive criticism regarding my research tremendously help me in developing this thesis. I am grateful to have had the chance to work with Frank.

I want to thank my PhD committee members, Professor Allen H. Renear, Professor Liang Y. Liu, and Professor Khaled El-Rayes, for their service and constructive advice. The support of my PhD committee helped improve my research and I feel very fortunate to have had the opportunity to work with them.

I also would like to thank Patrick McGowan, Daniel Ruane and James Smith from W.E. O’Neil Construction of Chicago for sharing their valuable practical experiences, knowledge and feedback about this research.

Many thanks also go to my colleagues and friends in Urbana-Champaign. They comprise the fourth part of my PhD life, in addition to my family, research and entertainment, and make my PhD study complete and colorful.

I would like to thank Department of Civil and Environmental Engineering of University of Illinois at Urbana-Champaign for the enjoyable atmosphere. Especially, I’d like to thank Susan Hale and Mary Pearson for their help and support in managing the official paperwork and administrative matters required
throughout my PhD studies.

I also want to thank Professor Wei-Chih Wang and Professor Ren-Jye Dzeng of the National Chiao Tung University and Professor Wu-Ting Tsai of the National Central University, all in Taiwan, for recommending me to go to University of Illinois at Urbana-Champaign to pursue a PhD. I would not have come to study in the United States of America and even would not have earned this degree without their advice and help.

I’d like to thank my entire family in Taiwan for their unconditional and everlasting love and support. I am grateful to my parents, Li-Chu Liu and Meng-Lin Wang for all their support, love, and belief in me. I am also grateful to my elder brother, Han-Tao Wang, and sister-in-law, Wen-Hui Guan, for their care, support and for bringing an adorable angel, my niece Cian-Yi Wang to this world when I was in the United States. Whenever I feel depressed, seeing her simple-hearted smiles in pictures or through web-cam can always alleviate my depression. I’d also like to thank my parent in law, Audrey Chang and Joe Ku for their support.

The greatest appreciation is extended to my wife, Yu-Ping, for her endless love and care for me. She is always there cheering me up and stands by me through the good times and bad. I would never have accomplished the dream of PhD study without her sacrifice and unconditional support.

Finally, thanks also go to everyone whom I have not mentioned here and who support me through advice and/or friendship during my time as a PhD student at University of Illinois at Urbana-Champaign.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>RESEARCH BACKGROUND</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>CONSTRUCTION SAFETY DOCUMENT MANAGEMENT FRAMEWORK —</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>REPRESENTATION MODEL</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>CONSTRUCTION SAFETY DOCUMENT MANAGEMENT FRAMEWORK —</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>REASONING ENGINE</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>IMPLEMENTATION AND VALIDATION</td>
<td>99</td>
</tr>
<tr>
<td>6</td>
<td>DISCUSSIONS AND CONCLUSIONS</td>
<td>167</td>
</tr>
<tr>
<td></td>
<td>REFERENCES</td>
<td>189</td>
</tr>
<tr>
<td>A</td>
<td>ILLUSTRATION ON BASICS STEPS OF USING JHA ADVISOR</td>
<td>195</td>
</tr>
<tr>
<td>B</td>
<td>TEST CASES OF REPRESENTATION OF THE EXTRACTED CONCEPTS</td>
<td>201</td>
</tr>
<tr>
<td>C</td>
<td>TEST CASES OF REASONING ABOUT THE EXTRACTED CONCEPTS</td>
<td>208</td>
</tr>
<tr>
<td>D</td>
<td>ANALYSIS OF REPRESENTING APPLICABILITY CONDITIONS FOR SAFETY</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RULES WITH MULTI-TYPE CONCEPTS</td>
<td>209</td>
</tr>
<tr>
<td>E</td>
<td>EXAMPLES OF JHA DOCUMENT REPRESENTATION</td>
<td>210</td>
</tr>
</tbody>
</table>
CHAPTER 1: INTRODUCTION

1.1 MOTIVATION

Construction is the industry that has the highest potential for occupational hazard events among all the different industries in the United States. It accounts for only around 5% of United States’ workforce but claims around 20% of all occupational fatalities (Abdelhamid and Everett 2000). According to the statistics of the U.S. Bureau of Labor Statistics, construction industry, including private sector, government and self-employed workers, accounted for 1,282 out of 5,703 fatal work injuries recorded in 2006 – the most of any industry sector (U.S. Bureau of Labor Statistics 2007). These facts imply that improvement efforts are needed to provide a more secure working environment for all project participants on construction sites.

In order to prevent occupational hazards, many research efforts have been conducted to understand, identify and analyze the trends and root causes of construction accidents (Abdelhamid and Everett 2000; Arboleda and Abraham 2004; Hinze et al. 1998; Suraji et al. 2001). They focused on hazards in general construction industry or in specific trades, and discussed the general or specific types of accidents. These research efforts provide important management information through qualitative post-hazard analysis such as understanding what kinds of accidents are likely to occur in what types of construction work. This allows project participants to take precautions accordingly against eventual accident reoccurrence. Instead of relying on post-hazard analysis, other research focused on analyzing construction activities themselves or activity-related factors which may result in potential hazards, such as equipment operation or spatial conflicts of activities or activities which expose workers to fall hazards, with the goal of accident prevention.

Deatherage et al. (2004) presented that fatalities can be prevented if project participants on sites follow Occupational Safety and Health Administration (OSHA) regulations. Benner (1983) pointed out that the most explicit technique to achieve construction safety is to abide by safety requirements, which aim to improve safety by clearly regulating what an employer or employee should or should not do without requiring personal judgment. Hudson (1999) also indicated that safety requirements of regulations stipulating preventive measures can help prevent many potential accidents. However, the safety-related research mentioned in the last paragraph seldom took into account safety requirements and their importance in the approaches. The research efforts conducting post-hazard analyses, for instance, rarely discussed in depth accidents which result from the ignorance or negligence of construction safety requirements; however, such a discussion is important because it can help safety engineers and supervisors come up with solutions, such as raising employees’ awareness of safety requirements, or identifying potential problems resulting from employees’ poor attitudes. The proactive research efforts, on the other hand, attempted to remove the root factors of accidents from the activities. However, the research seldom mentioned explicitly the relations between the preventive measures of the potential hazards associated with the activities and the specific safety requirements imposed by construction safety documents. The discussion on such relations is essential because they can best interpret the rationale of why to adopt the preventive measures, allowing project participants to become more familiar with the situations to which they are exposed as well as allowing them to better react to similar conditions in the future.

Furthermore, Teo et al. (2005) argued that insufficient safety knowledge of workers is one of the major
causes of site accidents. Safety requirements imposed by construction safety documents, such as safety specifications or project construction safety plans, are good sources of safety knowledge. Therefore, increasing workers’ awareness of applicable safety requirements imposed by construction safety documents is one solution to the issue of workers’ insufficient safety knowledge and, hence, should allow preventing accidents. Laurence (2005) also claimed that if safety requirements are not properly identified, project participants may be unaware of safe behaviors, further resulting in occupational hazards and possibly leading to fatalities, injuries and illnesses. However, there are a huge amount of sources of safety requirements, such as federal government, state and city governments, the employer or employee’s own company, professional and technical societies, or trade associations (Brauer 1990). Due to the volume of different safety requirements stipulated in different sources, identification of safety requirements applicable to an activity becomes cumbersome and time-consuming (Wang and Boukamp 2007) and at times may hinder project participants from carefully reviewing and identifying applicable safety requirements.

Although some safety specification databases, such as the Canadian enviroOSH Legislation plus Standards (Canadian Centre for Occupational Health and Safety 2008), exist and provide functionality for searching safety requirements based on keywords, it is difficult for users to efficiently determine applicable safety requirements comprehensively by simply using keywords to do the search. Keyword-based searches are often either too specific to exclude documents that would have been of interest or they are too general to include documents which are not of interest (Boukamp 2006; Soibelman and Caldas 2006). That is, keyword-based searches performed on a large number of construction safety documents can help generate lists of safety requirements which contain the search keywords; however, there will be other requirements
which do not contain these keywords and will therefore not be listed even though they are inferentially relevant.

Research efforts using expert systems for code compliance checking, such as the Health and Safety Expert System (HASES) developed by Gowri et al. (1993), also provide ways to support the compliance checking for safety requirements and are found to be especially beneficial for non-experts who do not have the knowledge of relevant safety requirements. These efforts rely on predefined IF-THEN rules. Users have to provide conditional information as the IF parts to the systems to allow it to reason to identify applicable codes defined in the THEN parts of the rules. However, this type of systems has a similar reasoning limitation as keyword-based searches: only the safety requirements which literally contain the provided conditional IF information will be identified while those requirements which are only inferentially relevant to the provided conditional information will not be identified.

In addition, a lot of information systems have been developed in the construction industry; these systems rely on classifications to organize information in structured or unstructured format. However, even though classifications deploy superclass-subclass relations among information that provide a better semantics than keyword-based search, classifications are still insufficient for representing other semantics among information, e.g. one piece of information which applies to a project may indicate another one which definitely does not apply to the project.

Therefore, these issues and discussions above suggest that there is a need to have a new safety management approach which focuses on an improved identification of safety requirements. This approach
should improve the currently unsatisfactory search functionality provided by the keyword-based search mechanism and current construction information systems, and facilitate automated identification and classification of safety requirements in order to help raise project participants’ awareness of the requirements.

1.2 OVERVIEW OF THE VISION

In this research, I envision to improve construction site safety by supporting automated identification and classification of safety requirements through automated reasoning about construction safety documents, and to present a prototype system that could be used to implement the identification and classification process. To achieve this goal, the information of safety requirements from construction safety documents needs to be well represented, structured and processed. Figure 1.1 shows an overview of the developed approach that highlights the process to reason about the construction safety documents. The process will incorporate document modeling to model safety documents in a computer readable and interpretable format. Additionally, concepts extracted from construction safety documents and contexts identified from within a project will be processed to obtain a list of safety requirements classified according to their applicability. To organize the extracted concepts, a topology representation will be required.
In the proposed process depicted in Figure 1.1, several sub-processes need to be implemented. One, illustrated in Figure 1.2, is to extract concepts from construction safety documents and to organize the extracted concepts in a topology representation. The concept (or contextual concept) in this research is defined as follows:

A (contextual) concept is a unique term\(^1\) abstracting a domain phenomenon.

In the context of construction safety documents, terms abstracting jobs performed, components built,\(^1\) actions taken or resources used are concepts. A topology representation of concepts, hence, is a tool that can model domain concept knowledge, i.e. concepts and their relationships, in order to leverage this knowledge for later reasoning work. To achieve automated identification of safety requirements, a topology representation of concepts for the construction safety domain is required; in this regard, terms which are used in the safety documents and fit in with the definition of concept become the sources of concepts and should be extracted. Semi-automatic approaches to extract concepts from documents, such as Text2Onto

---

\(^1\) A term usually describes a string of characters or a word with a specific meaning while a concept usually represents an abstract principle. I do not specifically distinguish their difference in the definition of a contextual concept in this research.
(Cimiano and Völker 2005) and TerMine (The National Centre for Text Mining 2008), exist; however, because these approaches are not designed specifically for the construction domain, their capability of concept extraction cannot fully satisfy the herein proposed research’s needs (Wang and Boukamp 2008). Therefore, this research mainly relies on manual concept extraction from construction safety documents. Once the concepts are extracted, they are modeled as classes in a topology representation, and relationships are then defined to connect these classes. After this sub-process, a topology with extracted concepts is output.

![Figure 1.2: Representing concepts extracted from construction safety documents](image)

Another sub-process (shown in Figure 1.3) is to model the safety documents, from which safety requirements shall be identified, in a computer readable and interpretable format. The purposes of this sub-process are to (1) transform the original safety documents from plain text format into a computer readable and interpretable format, which can be easier accessed and analyzed, to benefit the automation process; and to (2) structurally arrange the documents so that applicability conditions and applicability exceptions for each safety requirement can be specified to facilitate the later reasoning process. At the end of this sub-process, safety documents modeled in a computer interpretable format are generated.
The next sub-process, shown in Figure 1.4, is to reason about the concepts modeled in the topology to determine each concept’s applicability. First of all, context identification approaches are adopted to see what contexts the current project is in. The context (or domain context) in this research is defined as follows:

A context is described through a concept or a conjunction of concepts selected from the topology to abstractly define a condition or situation in a domain.

In other words, a project’s current conditions or situations that are described through concepts are identified through context identification approaches. If certain contexts are identified (i.e., certain situations take place), the concepts or conjunction of concepts which describes the identified contexts should be flagged *applicable* in the topology. The reasoning mechanism will automatically reason about the topology and propagate the identified contexts’ applicability throughout the topology. If the concepts describing the identified contexts could not be found in the topology, project participants should check whether the new concepts are in the same domain as the topology and whether they are essential. Project participants then decide whether or not to model them into the topology. Different context identification approaches could be adopted, such as reality capture technologies (such as Radio Frequency Identification, RFID), Building Information Model, or manual observation and identification. In this research, I adopt the approach of
manual observation and identification by users. As shown in Figure 1.4, an updated topology with propagated applicability information is output after this sub-process.

Figure 1.4: Reasoning about identified concepts to obtain an updated topology

Figure 1.5 depicts the final sub-process for reasoning about the modeled safety documents. Both the modeled safety documents and updated topology from previous sub-processes become the input in the sub-process. Unlike the one adopted in previous sub-process, the reasoning mechanism here is performed on the modeled documents and it will reason about safety requirements’ applicability conditions and applicability exceptions which are specified in the sub-process of modeling safety documents to determine the safety requirements’ applicability (e.g. applicable, possibly applicable or not applicable). For example, if a safety requirement’s applicability condition contains a single concept that is found not applicable in the topology, the safety requirement is regarded as not applicable (details are discussed in chapter 4). After this sub-process, safety requirements that are respectively classified as applicable, possibly applicable or not applicable are identified and output.
By automatically generating classified safety requirements, I expect the envisioned approach can enable improvements in site safety planning and management by raising awareness of these requirements throughout different phases of a construction project. For example, the approach could be deployed during the design and planning phases of a project to make designers and schedulers aware of safety implications resulting from specific designs and schedules. Also, the approach could support safety planning and evaluation of safety impacts of different construction methods deployed in construction activities. Finally, field engineers and inspectors could receive a safety checklist that is tailored towards the specific project they work on, rather than relying on general safety checklists.

1.3 DOMAIN OBJECTIVE

The principal domain objective of this research is: improving the search for safety requirements from construction safety documents to raise project participants’ awareness of safety requirements.

1.4 RESEARCH QUESTIONS AND OBJECTIVES

The specific research questions to be addressed are:
How should safety requirements in construction safety documents be modeled to support automated identification and classification of these safety requirements?

How should topological concepts that are extracted from construction safety documents be modeled?

What reasoning mechanisms are necessary to process the modeled topological concepts for identifying construction safety requirements?

What reasoning mechanisms are necessary to process the modeled construction safety documents for identifying construction safety requirements?

In order to achieve the domain objective and address these research questions, the specific research objectives include:

1. Establishing a topology representation structure that can be used to model the concepts of safety requirements and relationships between them.

2. Developing a modeling framework for modeling construction safety documents to enable automated reasoning about the documents.

3. Developing reasoning mechanisms that can process the modeled concepts and construction safety documents in order to identify and classify safety requirements according to their applicability.

4. Developing a computer system that can implement the reasoning mechanisms to search through safety requirements and output requirements grouped by their applicability.

1.5 RESEARCH CONTRIBUTIONS

The contributions of this research are:
(1) Representation models are developed, which can be used to model concepts and relationships between the concepts and to model construction safety documents.

(2) A topology reasoning engine is developed, which allows automatically reasoning about modeled concepts to evaluate their applicability.

(3) A safety requirement reasoning engine is developed, which allows reasoning about modeled construction safety documents to evaluate safety requirements’ applicability and also allows classifying safety requirements according to their applicability.

1.6 RESEARCH NEEDS

In the course of fulfilling the research objectives, five research needs shall be taken into account in this research. The research needs thereby guide the development of the representation and modeling framework as well as the reasoning mechanisms.

(1) Need for flexibility in the modeling of concepts

A construction company needs to be able to define safety rules in construction safety documents to regulate necessary safety actions for each project in order to assure a safe construction environment. A new project may adopt construction methods which are different from those of previous projects; in this regard, new safety rules may have to be specifically defined for the new project. In addition, each rule is described through multiple concepts and these concepts need to be structured to formally represent the conceptual knowledge. Therefore, the approach developed to structure concepts has to be flexible in consideration of the need of defining new safety rules.
(2) Need for flexibility in the modeling of construction safety documents

As aforementioned, a construction company has a need to define new safety rules for new projects. These new safety rules together with their applicability conditions and exceptions will comprise safety requirements of construction safety documents. Thus, the proposed approach must allow flexibly defining new safety rules and comprising safety requirements. In other words, the document modeling approach chosen in this research should ensure that the construction safety documents are maintainable and human readable so that engineers have no difficulty in adding and editing safety rules for these documents.

(3) Need for modeling construction safety documents in hierarchies

Slava (1985) and Boukamp (2006) identified that construction specifications have a hierarchical structure. According to my observations, construction safety documents are also structured in hierarchies in order to organize safety requirements in the documents. Hierarchies are established through sections and subsections of safety documents which are used to define applicability conditions and group parts which fall under similar applicability conditions. Since construction safety documents have to be modeled in a computer readable and interpretable format for automatically reasoning purpose, the format adopted in the proposed approach should also be capable of addressing the need of modeling construction safety documents in hierarchies.

(4) Need for reasoning about concepts

To identify applicable safety requirements, engineers have to first identify the current project context and then use it to find the safety requirements which apply to the context. Since a context is described through concepts, engineers need to observe as many concepts related to the project context as possible to
make the context description complete. However, it is unavoidable that engineers may observe some concepts while neglecting others. Neglecting concepts can either lead to applicable safety requirements being not identified or lessen the chance to filter out safety requirements that are irrelevant, i.e. not applicable, to a given project context. Engineers cannot be expected to identify all of the positively and negatively relevant concepts. Therefore, there is a need for helping engineers identify concepts that are relevant to the observed concepts. This research need shall be addressed in the proposed approach by providing a reasoning mechanism that can reason about the modeled concepts and identify concepts related to the observed ones.

(5) Need for reasoning about construction safety documents

Once the context (i.e., the concepts describing the context) is identified, engineers would like to know which safety requirements apply and which do not apply to the identified context. Also, engineers may also want to be assured that no safety requirements are filtered out until these requirements are fully evaluated to be not applicable to the context. Therefore, there is a need to have a reasoning mechanism that can reason about the construction safety documents to evaluate each safety requirement’s applicability. This mechanism should be robust: not only will applicable and not applicable safety requirements be identified, but also safety requirements whose applicability is unknown will not be filtered out. This need should also be addressed in the proposed approach.

1.7 RESEARCH METHODOLOGY

The research methodologies adopted in this research can be summarized as follows:
Literature review: Previous research is reviewed in the following areas: construction safety theories and practices, specification and document modeling approaches, construction information systems, classification representations, and ontological modeling. The applications of these approaches as well as their advantages and disadvantages are studied to find out how to best address the needs of this research.

Framework development: Based on the reviews and evaluations of previous research, a representation and reasoning framework for managing construction safety documents is developed.

System development: A prototype system is developed in this research to realize the developed framework.

Validation: The developed framework and prototype system are validated to determine how effective it is to use them to identify applicable safety requirements. The validation is performed through the use of synthetic test cases, application of scientific measurements and interviewing professionals who test the prototype system and give their feedback. In addition, comparison between the approach proposed in this research and keyword-based document searches is also conducted.

1.8 SCOPE

Construction safety requirements are imposed by construction safety documents that are prepared according to the needs of different construction safety management approaches. Among such many safety management approaches, Certified Safety Professionals and experienced safety engineers (Boukamp and Wang 2008; Roughton and Crutchfield 2007; Swartz 2001) claimed the Job Hazard Analysis (JHA) is one of the most prevalent and effective safety management practices in the industry. JHA is also a safety
planning approach recommended by OSHA (U.S. Department of Labor 2002). Therefore, I focus on JHA
documents in this research and research how safety requirements can automatically be identified from them
in order to benefit construction safety planning. Although other construction safety documents, such as the
OSHA safety regulation for construction safety (U.S. Department of Labor 2003) or proprietary safety
manuals, are also important for construction safety planning, they are not in the scope of this research and I
only discuss them to evaluate the generality of the developed approach.

1.9 DISSERTATION OVERVIEW

This dissertation describes the developed Construction Safety Document Management Framework in
which representation models and corresponding reasoning engines are developed. First, chapter 2
summarizes the literature review in which related research works are discussed. Chapters 3 and 4 discuss
the proposed representation models and reasoning engines respectively. In chapter 5, this dissertation
provides a detailed discussion on the validation performed for both the representation models and reasoning
engines. Finally, conclusions and directions for future research are presented in chapter 6.
CHAPTER 2: RESEARCH BACKGROUND

2.1 INTRODUCTION

This research builds on, integrates and extends research in the areas of (1) representation of specifications, (2) document modeling, (3) construction information systems, (4) topology representation and (5) Job Hazard Analysis. Exploring the first two areas would help me identify potential approaches to address the research needs for flexibility and hierarchies in the modeling of construction safety documents. The third and fourth research areas are studied in order to identify eligible approaches which allow flexibly modeling concepts and provide ways to connect concepts upon which the reasoning mechanisms can build. I also looked into the last research area as I mainly use the documents from this area for demonstration in this research. The following sections will describe these different research areas and point out how the previous research efforts relate to this research.

2.2 REPRESENTATION OF SPECIFICATIONS

Automated reasoning about specifications has been researched for different types of specifications in the construction industry, such as federal government regulations, design specifications, construction specifications, and building envelope codes. Therefore, different types of representation and reasoning approaches have been developed aiming at different types of specifications and different reasoning tools. These approaches are summarized as follows:
2.2.1 Decision Table-based Approaches

Decision table-based approaches (Fenves et al. 1969; Garrett and Fenves 1987) are the initial research efforts of design specification representations. In these approaches, the provisions of the specifications are represented in the form of decision tables, which display all the possible combinations of applicability conditions and related actions of each provision. Figure 2.1: An example of using a decision table to represent design specifications

shows an example of using a decision table to represent a design specification for determining the minimum slab thickness given various fire-resistance ratings and different aggregate types of normal-weight concrete (Hoffman and Gustafson 2000). In this example, the upper part of the decision table represents conditions, i.e. various fire-resistance ratings and aggregate types while the lower part represents the actions corresponding to each different condition. For example, if a designer decides to use normal weight concrete of Carbonate aggregate in a project and requires 2-hour fire-resistive rating, the slab thickness in the design should not be less than 4.6 inches, according to the last column in the decision table.

| Normal weight concrete: Siliceous aggregate | Y | Y | N | N |
| Normal weight concrete: Carbonate aggregate | N | N | Y | Y |
| Fire-resistant rating = 1 hour | Y | N | Y | N |
| Fire-resistant rating = 2 hour | N | Y | N | Y |
| Minimum Slab Thickness = 3.5 in. | Y | N | N | N |
| Minimum Slab Thickness = 5.0 in. | N | Y | N | N |
| Minimum Slab Thickness = 3.2 in. | N | N | Y | N |
| Minimum Slab Thickness = 4.6 in. | N | N | N | Y |

Note: Y = Yes, N = No

Figure 2.1: An example of using a decision table to represent design specifications
The advantages of these approaches are that they provide a precise interpretation of relevant design specifications to better support design code compliance checking (Fenves et al. 1969) and automatic design of structural components (Garrett and Fenves 1987). However, decision table-based approaches usually have to establish links among different decision tables because provisions are often connected with one another in a specification and each decision table represents only one provision. Such interlinks among decision tables make it difficult to automatically reason about the tables and therefore, the decision table-based approaches are difficult to be automatically processed.

2.2.2 Rule-based Approaches

Rule-based approaches (Ding et al. 2004; Gowri et al. 1993; U.S. Department of Energy 2010) define IF-THEN rules for specifications. The IF parts of the rules describe applicability conditions that need to be satisfied for the specifications to be applicable; the THEN parts describe required actions to be taken or applicable codes that have to be abided by. These approaches are usually applied to check code compliance, such as checking energy code compliance by U.S. Department of Energy (2010) and checking safety code compliance by Gowri et al. (1993).

Following the example shown in section 2.2.1, instead of using a decision table to represent the applicability conditions and required actions of a specification, rule-based approaches simply address them in several IF-THEN rules, such as:

IF Normal Weight Concrete of Siliceous Aggregate is used AND Fire-resistive Rating = 1 hour
**THEN** Check whether *Slab Thickness* is equal to or greater than 3.5 inches.

In other words, a rule-based system has to ask system users what type of normal weight concrete is designed and what fire-resistive rating is preferred in order to return the required slab thickness to users. A system may also ask users to input current design thickness for slabs in addition to the previous two questions, then the system can directly return “specification passed” or “specification violated” rather than returning the required minimum thickness to users.

The advantages of rule-based approaches include their flexibility of representing specifications (Ding et al. 2004) and easy deployment in expert systems (Rasdorf and Wang 1988). However, in order for a provision to be automatically processed, the IF and THEN parts of the provision have to been modeled in specific programming or modeling languages, which makes the provision less understandable and accessible by specification users.

2.2.3 Logic-based Approaches

Logic-based approaches (Hakim and Garrett 1993; Rasdorf and Lakmazaheri 1990) adopt formal logics to model the provisions of design specifications. Since logic is a formal, systematic knowledge representation and reasoning approach, the provisions of design specifications modeled in the form of logic can be formally represented and reasoned about through the inference capability of logic. Hakim and Garrett (1993) claimed that these approaches enable the evaluation of the consistency, completeness, and clarity of design specification models as well as support the reasoning about incomplete knowledge.
Basically, logic-based approaches also use IF-THEN logics to represent specifications. The difference is that logic-based approaches adopt formal logics and rule-based ones do not. For example, the example IF-THEN rule shown in section 2.2.2 can be represented in first order logic as follows:

$$\forall x [\text{NormalWeightConcreteSiliceousAggregate}(x) \land \text{FireResistiveRating}(x) = 1 \rightarrow \exists y \exists z [\text{Slab}(y) \land \text{IsComposedOf}(y, x) \land \text{Thickness}(z) \geq 3.5 \land \text{HasThickness}(y, z)]$$

In this logical statement, the applicability condition is specified on the left side of the arrow while the action is on the right. Additionally, in order to reason about such a logical statement, users are asked to provide the same information as that in the rule-based approach. This information then is used to test whether this logical statement holds or not. If the statement holds, this means that the design specification is complied with; otherwise, the design specification is violated.

Logic-based approaches require specification authors and users to have knowledge of logic to be capable of modeling the specifications in logic and understanding them. However, specifications modeled in logic usually include user-defined predicates, logical operators (connectors) and quantifiers. It is very difficult and impractical to require designers and engineers to have such knowledge in order to access them. Therefore, the application of logic-based approaches in the construction industry is hindered.

2.2.4 Object-oriented Approaches

Object-oriented approaches adopt object-oriented models to represent both design and construction specifications (Boukamp 2006; Garrett and Hakim 1992; Kiliccote 1994; 1996). Object-oriented models use class hierarchies to represent specifications, including their applicability conditions and requirements, so that the applicability of one section of a specification can be passed down to its subsections. Figure 2.2
shows an example of a specification modeling structure using object-oriented models developed by Boukamp (2006). This applicability inheritance feature can reduce the efforts of determining whether one part of the specifications is applicable or not, and thereby, facilitate the reasoning process.

Moreover, object-oriented models can go a step further to model the detail of the applicability conditions of a specification, such as properties or behaviors prescribed in that section (Boukamp 2006), which enables the comparison between requirements imposed by specifications and deviations found in the as-built conditions, thus facilitating the evaluation of the adherence to the imposed requirements.

How to use object-oriented models to model specifications and their applicability conditions can be illustrated using the following example, which is based on the specification modeling structure developed by Boukamp (2006). Suppose a construction specification from ACI 117 shown below is focused in this example:

<table>
<thead>
<tr>
<th>ACI 117 Standard Specifications for Tolerances for Concrete Construction and Materials</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section 4 – Cast-In-Place Concrete for Building</td>
</tr>
<tr>
<td>4.1 Vertical Alignment</td>
</tr>
<tr>
<td>4.1.1 For heights 100ft or less:</td>
</tr>
<tr>
<td>Lines, surfaces, arises ........................................ 1 in.</td>
</tr>
<tr>
<td>Outside corner of exposed corner columns and control joint grooves in concrete exposed to view .................... 1/2 in.</td>
</tr>
</tbody>
</table>

Figure 2.2: A specification modeling example using object-oriented approaches (Boukamp 2006)
To model, the specification 4.1 in this example, this specification is represented as an object of the class “GeneralSpecification”. Then, this object is associated with an object of the class “GeneralApplicabilityCondition”, which describes the circumstances under which this specification applies. This applicability condition object can store specific information imposed by the specification. For example, it stores context information, i.e. “Concrete Construction and Materials”, “Cast-In-Place Concrete” and “Building”, which are imposed by the general specification and section headers; it also stores behavior information, i.e. “Vertical Alignment”, which is imposed by the sub-section header. In other words, the specification 4.1 will be applicable when all the information contained in its associated applicability condition object is found applicable to given project situations.

Although object-oriented approaches are the prevalent specification representation approaches due to their flexibility and extensibility (Boukamp 2006) as compared with the aforementioned approaches, two main difficulties hinder the broad application of such approaches to the construction industry. The first difficulty is to maintain the modeled specification libraries. They are only editable by ones who understand the underlying modeling mechanisms and formalism (i.e. the object-oriented representation language and the modeling specification). However, it is impractical in the construction industry to ask specification authors or users to have such programming knowledge. The second difficulty is to model the specifications in a human readable manner. Object-oriented approaches use classes and attributes to represent key concepts of specifications, such as representing the concept “Column” as a class and the concept’s “Height” as the class’s attribute. However, once specifications are represented in this way, the specifications become less human readable than originally in their textual format and the understandability of the specifications is lessened accordingly. Additionally, users cannot be sure whether the objects of the classes and their
attributes are processed in compliance with the textual descriptions of the specifications when the reasoning mechanisms are performed as the textual descriptions are separate from the actual reasoning performed in an object’s method.

2.2.5 Hybrid Approaches

Hybrid approaches (Kerrigan and Law 2003; Neilson et al. 1998; Rasdorf and Wang 1988; Yabuki and Law 1993) combine some of the aforementioned approaches. For example, Yabuki and Law (1993) combined object-oriented models and logic to develop an object-logic approach to represent design specifications. Rasdorf and Wang (1988) combined rule-based and decision-table approaches to develop a new approach to check design code compliance in an expert system. While hybrid approaches leverages advantages of the combined approaches, they inevitably inherit drawbacks from the combined approaches. For example, the object-logic approach by Yabuki and Law (1993) has similar maintainability issues as the object-oriented approach has, which make working with specifications cumbersome (Kiliccote 1994).

2.2.6 Summary

Although representations, such as decision table-based, logic-based and rule-based approaches, provide formal and straightforward ways to represent specifications and can clearly define which requirements of the specifications should apply when their related conditions are satisfied, they are less flexible and manageable than the object-oriented approaches (Boukamp 2006; Fenves et al. 1995). In addition, when the second research need for modeling flexibility and maintainability discussed in
section 1.6 is taken into consideration, these approaches to modeling design and construction specifications are not be well suited to model construction safety documents. This finding guides me to another research domain, document modeling, which is discussed in the next section.

2.3 DOCUMENT MODELING

A document is composed of the content part that stores the information, and the format part that specifies the way in which the information is arranged and presented. The domain of document modeling is only concerned with the content part and not with how the content is formatted. Renear (2008) defined document modeling as “a principled systematic representation of textual information in order to improve the efficiency, functionality and interoperability of the creation, management, and exploitation of documents and document-like content.” In other words, document modeling is a process to support document users in the interaction with the content of documents.

DeRose et al. (1990) summarized document modeling approaches that deal with textual content of a document in different ways. These approaches include treating text as bitmap, as a stream of characters, as formatting instructions, as page layout, as a stream of content objects, and as an ordered hierarchy of content objects (OHCO). The OHCO approach was claimed to prevail over other modeling approaches due to its capabilities to support data integrity, information retrieval functions, and even special processing on non-textual data (DeRose et al. 1990). The OHCO approach views the texts of a document as an ordered hierarchy of content objects, which can be described by defining its constituent terms separately and explaining their potential role in this specific research:
• **Content objects** represent structural components in a document. For example, the Job Hazard Analysis (JHA) document (U.S. Department of Labor 2002) targeted by this research could have content objects represented as *Activity-*-, *Job Step-*-, and *Potential Hazard-*objects.

• **Hierarchy** indicates the content objects are organized into a system with different hierarchical levels. For instance, one *Activity* object may comprise multiple *Job Step* object, and one *Job Step* object may be associated with multiple *Potential Hazard* object in a JHA document.

• **Ordered** states the arrangement of content objects, i.e. which type of objects may follow or precede another. For instance, an *Activity* object may also contain one *Applicability Condition* object for each *Job Step* object, which specifies the condition in which that *Job Step* object applies. The *Applicability Condition* object shall precede its related *Job Step* object to let document readers understand at once what condition the following *Job Step* object applies to.

The content objects of the OHCO approach can be best represented through descriptive markup by putting “markup tags” around them. For example, the notion that a book named “Logic” has three chapters “Propositional Calculus”, “Quantifiers”, and “Predicate Logic” may be represented in the OHCO approach as follows:

```
<Book>
  <BookName>Logic</BookName>
  <Chapter>
    <ChapterTitle>Propositional Calculus</ChapterTitle>
  </Chapter>
  <Chapter>
    <ChapterTitle>Quantifiers</ChapterTitle>
  </Chapter>
  <Chapter>
    <ChapterTitle>Predicate Logic</ChapterTitle>
  </Chapter>
</Book>
```
The markup tag <Book>, with its respective closing tag </Book>, is the root tag representing the content object of “Book” in which the content objects “Book Name” and “Chapter” are defined through markup tags respectively. This example also shows the meaning of ordered (BookName is followed by Chapter) and hierarchy (Book, Chapter, and ChapterTitle comprise the three-level hierarchy).

To model documents in OHCO through a descriptive markup approach, it is necessary to have a standard which defines the required descriptive markup tags and specifies the rules of using the defined descriptive markup tags to model the content objects of the documents. There are two widely known, common standards which can serve this purpose: Standard Generalized Markup Language (SGML) and Extensible Markup Language (XML).

SGML was developed in the 1970s and then became the Standard 8879 of the International Organization for Standards (ISO) in 1986 (ISO 1986). SGML itself does not specify a particular set of markup tags that are used to describe content objects of a document, but rather specifies a way for defining a customized markup language with markup tags for a document. In other words, SGML is a meta-language which aims at defining specific markup languages that can be used to describe a document in OHCO structure. On the other hand, the standard XML was initially drafted in 1996 in order to streamline SGML by removing SGML’s redundant, complex and confusing features, and became a Recommendation of the World Wide Web Consortium (W3C) in 1998 (Harold and Means 2004). Since XML is just a simplified version of SGML, it is also a meta-language; that is, it also aims at providing a means of defining markup tags to describe content objects of a document. Deploying SGML or XML enables using the OHCO approach for modeling documents, and document authors, therefore, can flexibly define their own markup
language with specific markup tags that allow structuring documents’ content according to their needs. In addition, since XML tags are human-readable texts and the entire modeled document thus is in a human-readable text format, users’ can easily revise the document models as long as they abide by the rules specified by the defined markup language.

The OHCO approach is regarded as an eligible alternative to the object-oriented approach and is adopted in this research for modeling construction safety documents due to the following main advantages (DeRose et al. 1990). First, it benefits document authoring by allowing editing documents in text, which make documents understandable by readers, and also allowing collaborative work among different authors. Second, documents described in OHCO structure are system and application-independent; thereby, they can be transferred freely. Third, the OHCO approach views documents as a database of text elements so that it facilitates information retrieval from documents. XML is then chosen as the modeling language to realize the modeling process as XML is currently in a wide range of application and XML also has been successfully applied to related research on modeling government specifications (Kerrigan and Law 2003; Lau et al. 2005). In this research, all the details of deploying the OHCO approach and XML technique are presented in chapter 3.

2.4 CONSTRUCTION INFORMATION SYSTEMS

Construction information systems store information in an organized manner to facilitate classification, analysis, retrieval and reuse of the information. For over a decade, a multitude of information systems has been deployed in a wide range of application areas in the construction industry. Among these are systems
for collecting and processing site information according to traditional superintendents’ daily site reports (Russell 1993); systems for improving constructability by analyzing and classifying lessons learned according to the MasterFormat of the Construction Specification Institute (Kartam 1996); and systems for supporting the management of large number of documents and their integration into a model-based information system (Caldas et al. 2005).

Most construction information systems rely on classifications to categorize information. For example, Caldas et al. (2002) adopted a classification with 13 divisions to classify construction documents; El-Diraby and Zhang (2005) developed a classification called BCtaxo with 5 major root division in order to classify concepts representing knowledge of the building construction industry. This characteristic indicates that construction information systems can deal with the often encountered difficulty resulting from searching for specific information: information may be stored in various sources and represented in structured or unstructured format (Caldas et al. 2002). Caldas et al. (2002) defined structured information as the information stored in database systems or in specific applications, and unstructured information as the information represented in text-based documents, graphical or multimedia files. Such a difficulty can be tackled by incorporating classifications into construction information systems which gather structured and/or unstructured information and organize it in accordance with the classifying configurations. Not only do classifications enable unstructured information to be systematically organized to facilitate reuse of the information and retrieval of the documents with the information, but they also allow structured information or documents to be rearranged in different dimensions in which users are interested. While many data mining tools have been developed to analyze structured information, more and more research efforts of construction information systems were made to focus on unstructured information (Soibelman et al. 2008;
In addition, information stored in the systems is context-specific; i.e. the information applies or relates to a specific situation (Caldas et al. 2002). Classifications used in construction information systems help cluster information of the same or similar contexts together. Then when a piece of information in a cluster is found to be useful, it is likely that other information in the same cluster may be useful as well.

While this reasoning feature plays an important role in managing information of construction information systems, the relationship between information (more specific, contextual information) is not limited to specialization-generalization relationship imposed by classifications, as discussed in chapter 1. Other relationships, such as associations, exist between them and should also be taken into account in construction information systems to allow establishing a more comprehensive network of contextual information. Such a network can then benefit the reasoning about the information through the identified relationships, e.g. what information should be applicable when it is associated with other applicable information.

Therefore, the herein proposed research builds upon classifications and goes beyond it by taking into account more types of relationships between contextual information. In the next section, topology representation is discussed as a means to model the information of contextual concepts and their relationships.
2.5 TOPOLOGY REPRESENTATION

Topology representation consists of two parts: representation of concepts and representation of relationships between concepts. In this research, concepts are the textual information describing application contexts for construction safety documents (the formal definition of concept for this research is given in section 1.2), and relationships are the semantic relationships between the concepts in the application domain. Therefore, topology representation approaches aim at addressing the first research need discussed in section 1.6 by modeling the context-describing concepts and representing required relationships between them.

2.5.1 Concept Representations and Classifications

In general, concepts can be represented in dictionaries or classifications. If represented in dictionaries, concepts are alphabetically arranged, such as the glossary of terms provided for construction safety and health (U.S. Department of Labor 1996) or that for metal building systems (MetalBuilding.com 2009). Dictionaries are a straightforward way for concept representations as they impose an ordered arrangement on the concepts. Classifications, on the other hand, provide hierarchical structures to organize concepts in class hierarchies. That is, the class hierarchies allow defining super-subclass relationships\(^1\) between concepts.

There have already been many classifications developed for the construction industry. For example, MasterFormat is a standard for specification-writing to hierarchically organize data for construction.

\(^{1}\) Superclass-subclass relationships are also called subsumption relationships.
requirements, products, and activities (Construction Specifications Institute and Construction Specifications Canada 2005). The OmniClass Construction Classification System (OCCS) provides a classification for purposes ranging from organizing documents and project information to providing a structure for electronic databases (OCCS 2008). BARBi, a project initiated by the Norwegian construction industry, establish a reference classification with a complete collection of concepts and objects from the building and construction industry as well as associated properties and relationships (BARBi 2004). Industry Foundation Classes (IFC) provides a classification with abundant classes to represent product and process information for the Architecture, Engineering, and Construction (AEC) industry (IAI 2008). In the safety domain, the Occupational Injury and Illness Classification Manual (OI&ICM) provides a coding system to classify injuries and illnesses in accordance with their nature, their source/secondary source, part of body affected, and event or exposure in which the injury or illness was produced (U.S. Department of Labor 1992). It should be noted that all the aforementioned classifications are regarded as *elaborate classifications* in this research because they have detailed hierarchical structures in which not only generic categories but also specific sub-categories can be specified. These comprehensive classifications allow concepts to be thoroughly organized as all concepts shall be able to find appropriate categories from the classifications to fit in.

As opposed to the elaborate classifications, there are other *simple classifications* that only specify generic categories without specific sub-categories. For example, the Components, Actions, and Resources (CAR) classification is a simple classification for representing construction method information (Aalami et al. 1998; Akinci 2000). The CAR classification has also been successfully adopted to represent contextual information (Wang and Boukamp 2008).
Although the simple classifications only provide generic categories for classification and may not organize concepts as systematically as the elaborate classifications do, they allow a more straightforward concept representation. For example, if the CAR classification is adopted, one only has to consider three categories into which each concept can be classified. Therefore, they are suitable for modeling a small number of less complicated concepts while elaborate classifications are favorable to any number of concepts of different complexities.

2.5.2 Other Relationships between Concepts

The other important aspect of topology representation is to represent relationships between concepts. If one wants to access the represented concepts for reasoning purposes, such as reasoning about concepts to determine whether a concept can be deduced by knowing another concept exists, semantic relationships between the concepts are needed to enable the deduction from known concepts to others that are not directly considered relevant.

When concepts are represented in class hierarchies, superclass-subclass relationships between the concepts can be defined. In practices, there are other types of relationships existing between concepts. The followings are some examples implying such relationships: A wall connects to another wall, and a crane is operated by a worker; engineers use “pour concrete” or “place concrete” alternately to describe concreting activity; and only truck cranes are used to hoist materials throughout this project. The first description indicates relatedness (i.e. “connect to” and “is operated by” respectively) between those concepts. The second one indicates another kind of associations which exists between concepts with identical meaning.
The last description also shows another kind of associations: one concept which applies in a project may exclude other concepts’ applicability (i.e. no other types of cranes are used in the project). Therefore, these associations indicating the identicalness of meaning, exclusiveness (together called logical associations) and relatedness (called non-logical associations) of concepts should also be taken into account as relationships in this research.

Approaches for modeling relationships of non-logical associations can be found in some classifications which provide instructions not only for classifying concepts into hierarchies but also for defining relationships for connecting concepts. For example, IFC provides many relationships to connect product and process information, such as the relationship “IfcRelConnectsElements” for connecting building products (IAI 2008). However, since relationships to be represented are highly concept-driven, i.e. strictly relying on the concepts they connect, the current relationship representations, including that of IFC, have to be general enough in order to maximize the capability of representing concepts. For example, the relationship “IfcRelConnectsElements” is used to connect different products of a building, but in IFC there is no relationship “IfcRelConnectsToWall” that aims specifically to connect specific building product “Wall”. That is, when users need to model relationships that are to be used for specific concepts, it is inevitable that they have to create specific relationships by themselves (Wang and Boukamp 2008). Specific relationships are important and necessary as they can enhance the semantics of concept representations and thus benefit the reasoning mechanisms. For example, suppose that the relationship “IfcRelConnectsToWall” exists; once this relationship is processed by the reasoning mechanism, one can be sure that the concept to which this relationship connects is “Wall” without having to check the concept connected. Moreover, the relationship “IfcRelConnectsToWall” could be specialized into more specific relationships, such as
“IfcRelSupportsWall” which specifies “Support” relation to “Wall” concept. This specialization provides more specific semantics and hence enables more precise reasoning about the relationships between concepts.

On the other hand, relationships of logical associations include disjoint and equivalent relationships. Disjoint relationships are used to model the exclusiveness between concepts, i.e. what concepts exclude one another. For instance, concepts “precast” and concept “cast-in-place” are disjoint. Equivalent relationships are used to model the identicalness of meaning of concepts, i.e. what concepts have the same semantic meaning. For instance, concept “pour” used in describing activity “pouring concrete” can be replaced with concept “place” without changing its meaning.

2.5.3 Ontological Modeling: An Approach to Topology Representation

A modeling approach to topology representation should be capable of fulfilling the representation of concepts and their relationships (superclass-subclass, logical associations and non-logical associations). However, classifications and other data modeling approaches are not fully sufficient for addressing this need. This fact directs me to another modeling approach which is the most promising one to topology representation: ontological modeling.

Ontological modeling is a systematic approach for representing knowledge in ontologies. Gruber (2008) defined an ontology as “an explicit and formal specification of a conceptualization”. Specifically, an ontology can model a set of concepts within a knowledge domain and relationships between these concepts;
ontological modeling, therefore, is a process to model concepts and relationships into ontologies. This feature makes ontological modeling an eligible approach to topology representations in this research.

Ontological modeling originates from the philosophy domain and is widely adopted in research efforts in the artificial intelligence and computer science domain in the recent two decades (Gruber 2008). The main areas in which ontological modeling is applied include communication and knowledge sharing, logic inference and reasoning, and knowledge reuse (Feruzan 2007).

Ontological modeling is a well-suited approach to topology representation for two main reasons. First, concepts and their semantic relationships can be easily modeled in the form of classes and properties in an ontology. Second, an ontology provides ontological inference mechanisms which allow reasoning about concepts modeled in the ontology. That is, implicit knowledge of concepts can be deduced by reasoning about explicitly declared facts of concepts through pre-defined reasoning axioms.

There has been much research work in computer science domain adopting ontological modeling for representing and reasoning about concepts. For example, Panu Korpipää et al. (2004) utilized an ontology to offer scalable representation and easy navigation of concepts for personalizing mobile device applications. Souza et al. (2006) proposed using an ontology to formally represent concepts to improve geospatial data integration and query processing. Kim and Choi (2006) and Wang et al. (2004) proposed ontology-based frameworks for modeling and reasoning about concepts in pervasive computing environments.

In the AEC domain, however, related research efforts which apply ontological modeling to concept
modeling and reasoning is relatively fewer. The e-COGNOS project was the first project to deploy ontological modeling for knowledge management and concept modeling and reasoning in the construction industry (Lima et al. 2005). Aziz et al. (2005) used an ontology to represent and deliver the information of contextual concepts, such as location, time and profile.

It should be noted that ontologies are independent of any classification even though concepts are usually modeled in class hierarchies in ontologies. In other words, ontological modeling does not specify what classifications should be used to classify concepts. When ontological modeling is adopted for representing the topology, users can simply categorize the identified concepts according to their attributes/features or incorporate external classifications which are found appropriate for classifying them into the ontology.

2.5.4 Languages for Ontological Modeling

To create an ontology, an ontology language is required to provide formal syntactic structure and modeling rules with which contexts can be represented. Ontology languages allow users to explicitly formalize and conceptualize their domain knowledge (El-Diraby et al. 2005; Lima et al. 2005; Rezgui 2006). Antoniou and van Harmelen (2004) pointed out that ontology languages should equip with the following features: a well-defined syntax, efficient reasoning support, a formal semantics, sufficient expressive power and convenience of expressions.

The most common ontology languages include Resource Description Framework (RDF) and RDF
Schema (RDFS), DAML+OIL (DARPA Agent Markup Language + Ontology Inference Layer), and Web Ontology Language (OWL). RDF provides data model specifications and XML-based serialization syntax for ontological modeling; RDFS provides specifications of class and property hierarchies for RDF.

DAML+OIL is a combined ontology language effort of DAML of the United States and OIL of Europe in 2000. It builds upon RDF/RDFS and provides more powerful modeling capability. OWL is a specification by the World Wide Web Consortium (W3C) (W3C 2004) and serves as a fundamental component of the Semantic Web initiative (Antoniou and van Harmelen 2004). OWL is based on the DAML+OIL language and therefore, has many features of DAML+OIL, such as adopting RDF as the modeling language to define ontology vocabularies and using XML-based RDF syntax for representing information (Bechhofer et al. 2004). OWL is divided into three expressiveness-increasing sublanguages: OWL-Lite, OWL-DL (Description Logic), and OWL-Full. OWL-DL is most often used as it provides strong expressiveness without losing computational reasoning efficiency and can exploit the considerable existing body of description logic reasoning (Wang et al. 2004).

The main modeling primitives of RDF/RDFS concern the organization of vocabularies in typed hierarchies: superclass-subclass relationships of classes and properties, domain and range restrictions, and classes’ instances. However, a number of important features are missing. Antoniou and van Harmelen (2004) listed the following expressiveness discrepancies: range restrictions, disjointedness of classes, combinations of classes, and cardinality restrictions. Ideally, OWL would be an extension of RDFS, in the sense that OWL would use the RDF syntax, such as rdfs:Class and rdfs:subClassOf, and would add language primitives to support the richer expressiveness required and overcome the previous discrepancies.
Chen et al. (2003) pointed out three advantages brought about by using OWL to define ontologies in developing ontological applications: (1) better expressiveness than other ontology languages, (2) backing of a well known and regarded standards organization, and (3) promising opportunities for expanding current applications due to the emergence of ontology inference engines in support of OWL.

In this research, OWL is adopted as the ontological modeling language to model concepts extracted from construction safety documents and relationships specified for the extracted concepts. The mechanism of reasoning about concepts developed in this research builds upon the abovementioned ontological inference mechanisms, and its major notion is: the applicability of a concept used to describe a specific project situation can imply applicability or inapplicability of other concepts that are semantically related to the first concept. The details of how to deploy OWL in representing and reasoning about concepts are discussed in the following two chapters.

2.6 JOB HAZARD ANALYSIS (JHA)

In this research, I use JHA documents as the targeted documents for demonstrating the proposed approach. As discussed in the previous chapter, JHA documents are selected as JHA is claimed to be one of the most prevalent and effective safety management practices in the industry (Roughton and Crutchfield 2007; Swartz 2001). The Occupational Safety and Health Administration (OSHA) recommends conducting JHAs in projects to prevent hazards in workplaces and to reduce worker injuries and illnesses (U.S. Department of Labor 2002).
JHA is a process of identifying potential hazards for each step of an activity and proposing safety rules to prevent these hazards. The basic procedure for conducting a JHA includes: (1) analyzing the activity to list all steps needed to perform the activity; (2) identifying potential hazards which may occur in the different job steps listed; and (3) propose safety rules (a.k.a. safety procedures or precautions) which can be adopted to eliminate or prevent each hazard.

In practices, JHA is usually conducted by construction safety engineers due to their expertise in knowing what potential hazards are associated with what tasks and what safety requirements regulate necessary actions for these hazards. If safety engineers conduct JHA for certain activities which they are not familiar with, they will consult the foremen in charge of the activities in order to understand the details of the activities and proceed with the analyses accordingly. In addition, they may also search through the collection of previous JHA documents to refer to similar or relevant ones.

Although previously identified information about job steps involved in an activity, hazards associated with each job step, and safety rules developed to avoid these hazards can be leveraged when performing a new JHA, reviewing previous JHAs is time-consuming when the number of activities and their respective steps and associated hazards is large. Also, useful information about potential hazards and associated safety rules from job steps of previous JHAs sometimes may be ignored if steps of the previous activity are identical to steps of the new activity but the previous activities fall into different categories than the new one.

Due to the complexity and the time-consuming nature of JHAs, safety engineers have to perform
JHAs often weeks, sometimes even months before the activity actually is scheduled to be performed. This makes it difficult to quickly react to changes in the construction plans and schedules as well as to appropriately identify the resulting safety concerns.

Therefore, this research aims to develop an approach that helps address the problems encountered in the JHAs. The proposed approach allows quickly identifying safety rules from a database storing previous JHA documents, which not only achieves the goal of raising project participants’ awareness of safety requirements but also assists safety engineers in efficiently conducting new JHAs.
CHAPTER 3: CONSTRUCTION SAFETY DOCUMENT MANAGEMENT FRAMEWORK — REPRESENTATION MODEL

3.1 OVERVIEW OF THE PROPOSED APPROACH

The principal goal of this research is to provide an approach that can benefit the search for safety requirements from construction safety documents and develop a system to demonstrate this approach. In this research, I develop a Construction Safety Document Management Framework. The Framework consists of two parts, the Representation Model and Reasoning Engine, which respectively deal with the representation and reasoning issues.

Figure 3.1 illustrates how the Construction Safety Document Management Framework and its parts are formed. In the Representation Model that is discussed in this chapter, both construction safety documents and ontology of concepts extracted from these documents will be modeled. The Reasoning Engine, discussed in chapter 4, will reason about the contextual concept ontology that model the concepts and subsequently evaluate the applicability of the safety documents. The Reasoning Engine, thereby, will leverage information available in the Representation Model. All the information flows are shown in Figure 6 through the dotted lines with arrowheads indicating the directions of the flows.

In section 1.2, I gave an overview of this research and briefly discuss what methodologies shall be adopted and how they cooperate with each other for the aforementioned representation and reasoning parts. In the following sections of this chapter, I will explain the details of the Representation Model and how it
should be implemented. In chapter 4, I will discuss the Reasoning Engine and elaborate the reasoning rules and principles guiding the Reasoning Engine.

Figure 3.1: Overview of the Construction Safety Document Management Framework

3.2 INTRODUCTION TO THE REPRESENTATION MODEL

The Representation Model aims to provide a systematic structure for modeling contextual concepts, construction safety documents, and the requirements they impose in a computer readable and interpretable format. The concepts stored in the documents may scatter over the documents. A concept that appears in one section of a document may appear multiple times in others. Since the concepts represent the application contexts of a safety requirement, it would be better to have them specifically organized elsewhere rather than scattered over the documents in order to effectively utilize them. Also, there are many different construction safety documents that specify diverse safety requirements. Only when they are modeled in a computer readable and interpretable format, can they be efficiently processed.

The Representation Model consists of two sub-models: the concept ontology representation model,
which models the concepts in an ontology, and the textual document representation model, which models documents’ texts in a structured format. The two sub-models are discussed in the following sections.

3.3 CONCEPT ONTOLOGY REPRESENTATION MODEL

The concept ontology representation model leverages ontological modeling to model concepts extracted from construction safety documents and to model relationships between these concepts. When tied to construction safety documents, the concepts can be used to describe contexts representing the applicability conditions and applicability exceptions of safety requirements. As input to the reasoning mechanism, the concepts can be selected to represent a list of potential situations that may occur in a project, i.e. the project specific contexts. Hence, the concepts can be viewed as a set of topological concepts of a domain knowledge base, and these topological concepts as well as the relationships between them should be well organized in order to benefit the reasoning process for identifying the applicability of construction safety requirements.

The concept ontology representation model requires (1) classifications to model concepts in hierarchies, and (2) relationships to connect the represented concepts, which are discussed in the following subsections.

3.3.1 Classification of Concepts

The first step of deploying concept ontology representation model is to use classifications to model
concepts in hierarchies. There are three principles guiding the hierarchical representation:

(1)  *Determine primary grouping concepts.*

(2)  *Use the primary grouping concepts as the main classes and all other concepts belonging to the primary grouping concepts are made subclasses.*

(3)  *Intermediate classifications (that is, used for secondary and tertiary grouping concepts and so on) may be necessary for the primary grouping concepts when some of the concepts can in nature be subdivision of the intermediate classifications.*

The first principle requires users to determine a set of concepts which are the most representative and general for all the concepts. Primary grouping concepts can be determined in different ways. If a safety document’s structure comprises constituent elements that store contextual information, these constituent elements can be considered as primary grouping concepts. For example, OSHA recommended three constituent elements for a JHA document, Activity, Job Step, and Potential Hazard (as shown in Figure 3.2), to specify the concepts describing the contexts to which the safety rules apply (U.S. Department of Labor 2002). Therefore, three primary grouping concepts, “Activity”, “Job Step”, and “Potential Hazard” are defined and used to group JHA concepts in this research.

In addition, one also can determine the primary grouping concepts empirically or consulting experienced and knowledgeable professionals. Another way is to refer to those classification systems discussed in chapter 2, such as MasterFormat (Construction Specifications Institute and Construction Specifications Canada 2005), to find appropriate ones as primary grouping concepts.
<table>
<thead>
<tr>
<th>Job Steps</th>
<th>Potential Hazards</th>
<th>Recommended Safety Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fly forms to area to be installed</td>
<td>Material dislodgement</td>
<td>● Inspect rigging to be/being used</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Ensure proper rigging method</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Ensure direct contact with crane operator</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Clear area to land material</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Use taglines</td>
</tr>
<tr>
<td>Take forms off cart/ blocking</td>
<td>Sprain/Strain of back</td>
<td>● Get assistance; Work with a partner.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Discuss lifting and moving process with partner.</td>
</tr>
<tr>
<td></td>
<td>Slip/Trip/Fall</td>
<td>● Clear path from cart to work location.</td>
</tr>
<tr>
<td>Stand forms into place</td>
<td>Sprain/Strain of back</td>
<td>● Use proper lifting technique.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Get assistance; Work with a partner.</td>
</tr>
<tr>
<td></td>
<td>Pinched fingers</td>
<td>● Wear slip resistant gloves.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Set form on ground away from adjacent form then grab form in a place where your fingers will not get pinched.</td>
</tr>
<tr>
<td></td>
<td>Sharp edges</td>
<td>● Review rebars conditions for sharp edges or tie wire hazards.</td>
</tr>
<tr>
<td></td>
<td>Fall</td>
<td>● Use ladder or scaffold; do not use top 2 rungs of ladder.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Ensure area around ladder/ scaffold is clear of debris and flat.</td>
</tr>
<tr>
<td>Set pins</td>
<td>Fall</td>
<td>● Use a portable ladder in the proper manner.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Get a partner to hold the form when needed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Use scaffolding where possible; Scaffold must be erected under supervision of a competent person; All guardrails must be installed and pins used; No substitute materials!</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● If climbing form, must use retractable lanyard anchored to top of form when feet are higher than 6’ off working surface.</td>
</tr>
</tbody>
</table>

Figure 3.2: An example JHA document

For the second principle, the JHA’s three primary grouping concepts play the role of the main classes that store the respective concepts as subclasses in this research. For instance, the concept “Frame Column” becomes a subclass of the primary grouping concept “Activity”.

The third principle indicates that other classifications can be incorporated to further classify the concepts of the primary grouping concepts. For example, suppose we have the following concepts which can be categorized to the primary grouping concept “Building Component” in the classification: “Bearing
Wall”, “Collar Beam”, “Drywall”, “Floating Wall”, “Joist”, “Nonbearing Wall”, “Parapet”, “Retaining Wall”, and “Tail Beam”. To structure the classification, it is useful to define two new classes “Beam” and “Wall” (i.e. they act as the secondary grouping concepts) as subclasses of the main class “Building Component” instead of representing all the concepts as direct subclasses of the main class “Building Component” in the classification. The new class “Beam” can act as a superclass of the classes “Collar Beam”, “Joist”, and “Tail Beam”, and the new class “Wall” can act as a superclass of the others.

I use a heuristic approach to illustrate how new classifications can be incorporated into existing ones in section 3.3.4. In addition, whether new classifications should be incorporated should consider if they can benefit the representation of and reasoning about concepts. Such considerations are also discussed in section 3.3.4.

3.3.2 Relationships between Classified Concepts

From a reasoning perspective, only modeling a concept classification without specifying appropriate additional semantic relationships between concepts is insufficient. The developed reasoning mechanism for automated identification of construction safety requirements (discussed in the next chapter) strongly relies on semantic relationships due to their capability of stringing related concepts together. Deploying semantic relationships enables navigation through sets of concepts and thereby prevents important concepts from being unattended. For example, if an engineer forgets to identify a Potential Hazard concept “Hazardous Atmospheres” for a Job Step concept “Excavation Using Sloping and Benching as Protective Measure”, the recommended safety procedure for preventing this hazard will be neglected.
Using a semantic relationship to connect the Job Step concept to the Potential Hazard concept can ensure the Potential Hazard will not be ignored once the Job Step concept is identified.

When concepts are modeled in a classification, superclass-subclass relationships between the concepts are specified. Additional semantic relationships, such as associations, need to be specified between the concepts as well. Associations are connections between concepts which do not have superclass-subclass relationships between them in order to specify the relatedness, exclusiveness or identicalness of meaning between them. Two main types of associations are used in this research: non-logical associations (called “association relationships”) and logical associations (called “logical relationships”).

Modeling associations is a highly concept-driven process. Thus, it is necessary to thoroughly review the construction safety documents from which the concepts are extracted in order to determine what associations would benefit the reasoning mechanisms and how they should be defined and established.

3.3.2.1 Association relationships

When defining association relationships to connect concepts, one should give them semantically-rich names to facilitate the understanding of how the two connected concepts are related (Wang and Boukamp 2008). In this research, the primary concept “Activity” has an association relationship “hasStep” connecting the primary concept “Job Step” in representing the concept of “An Activity hasStep Job Step”. An inverse relationship, “isStepOf”, for the relationship “hasStep” is also defined in order to represent the inverse relation between the concepts, i.e. “A Job Step isStepOf an Activity”. Similarly, another association
relationship “hasHazard” and its inverse relationship “isHazardOf” are defined to enable the connection of the primary concepts “Job Step” and “Potential Hazard”.

To facilitate the determination and presentation of association relationships, I use semantic relationship matrix that provides a straightforward means to structure concepts to be connected and then to help define association relationships for the concepts. Figure 3.3 shows the matrix that outlines the association relationships used in this research. First, concepts which are planned to be linked are printed in the cells of both the heading row and column of the matrix to respectively represent their connecting and connected roles, i.e. the subject and object of the association relationships to be defined. Thereby, the lower triangular area in the matrix is for representing association relationships which link the connecting concepts to the connected ones whereas the upper triangular area is for representing their respective inverse relationships. For instance, the association relationship “hasStep” shown in Figure 3.3 is for linking the connecting concept “Activity” to the connected context “Job Step”, and its inverse relationship “isStepOf” works reversely. Finally, the diagonal cells in the matrix could be used to represent association relationships linking concepts of the same type - but this is not required in the Concept Ontology Representation Model in this research.

Figure 3.3: Semantic relationship matrix for association relationship determination
Another occasion in which association relationships can be used is when combined concepts exist in the concept dictionary. A combined concept is a concept which is composed of multiple single concepts, which are called constituent concepts in this research. For example, an Activity concept “Frame Column” is a combined concept as it consists of two single concepts, an Action concept “Frame” and a Component concept “Column”. Combined concepts are important and useful as their applicability can imply the applicability of their constituent concepts (discussed in chapter 4). Therefore, if one wants to explore such an implication, association relationships can be used to connect a combined concept to its constituent concepts.

Following the previous example, association relationships “comprisesAction” and “comprisesComponent” can be defined to connect the “Frame Column” to the “Frame” and “Column” respectively. In addition, a variant of the semantic relationship matrix can be used to facilitate the determination of the association relationships for combined concepts. Figure 3.4 shows the matrix variant for the “Frame Column” as well as other two combined concepts. The heading row and column of the matrix in Figure 3.4 represent combined and constituent concepts respectively; the body of the matrix lists the association relationships connecting the combined concepts to the constituents. In this research, I am concerned with how a combined concept can be formed by other concepts; thus, I focus on uni-directional relations for combined concepts, i.e. the relations from the combined to the constituent concepts, and no inverse association relationships need to be defined. Hence, they will not be shown in the matrix variant.
3.3.2.2 Logical relationships

Logical relationships include disjoint and equivalent relationships, which respectively specify the exclusiveness and identicalness of meaning between concepts. Disjoint relationships connect concepts which exclude each other. An example is that concepts “Cast-In-Place” and “Precast” should be connected through a disjoint relationship if only one of them can apply in a given context. On the other hand, equivalent relationships connect concepts which are of the same meaning. For instance, a potential hazard “Slip” can be declared to be equivalent to “Trip” because in the context of construction safety they both mean that someone accidently slides or falls and loses his/her balance.

Association relationships together with their inverse relationships and logical relationships are important to help string semantically related context descriptors together bi-directionally to enable propagation of applicability values among concepts. That is, these relationships will be crucial for the effective knowledge inference from given facts about concepts’ applicability. The interpretation of all the relationships for the purpose of ontological reasoning is discussed in detail in section 4.2.
### 3.3.3 Ontological Modeling of the Classified Concepts and Relationships

Ontological modeling is adopted in this research to model the class hierarchies of concepts and the relationships between them. OWL was chosen as the ontological modeling language used to develop ontologies for two reasons. First, the developed concept reasoning mechanism (discussed in the next chapter) can exploit and build upon the powerful expressiveness of OWL. In addition, OWL is currently the most prevalent language in the ontological engineering domain and Semantic Web applications.

OWL provides a well-defined syntax to model the concepts represented in the class hierarchies and to model the relationships defined between the concepts. For instance, Figure 3.5 shows a snippet, which is modeled in OWL, of a classification representation.

```xml
<owl:Class rdf:ID="Frame_Column">
  <rdfs:subClassOf rdf:resource="#Activity"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasStep"/>
      <owl:allValuesFrom>
        <owl:Class>
          <owl:unionOf rdf:parseType="Collection">
            <owl:Class rdf:about="#Fly_Forms_to_Area_to_Be_Installed"/>
            <owl:Class rdf:about="#Set_Pins"/>
            <owl:Class rdf:about="#Stand_Forms_Into_Place"/>
            <owl:Class rdf:about="#Take_Forms_Off_Cart/Blocking"/>
          </owl:unionOf>
          <owl:Class/>
        </owl:allValuesFrom>
      </owl:Restriction>
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
```

Figure 3.5: Snippet of an OWL model for the example classification

The representation elements used in the snippet in Figure 3.5 can be interpreted as follows:

- **Class hierarchy/Superclass-subclass relationship representation:** OWL’s syntax uses the tag
<owl:Class> to define a new class “Frame Column” and the tag <rdfs:subClassOf> to specify the new
class as a subclass of the class “Activity”.

- **Association relationship representation:** OWL’s syntax first uses the tag <rdfs:subClassOf> to specify
the subject class, i.e. class “Frame Column”, to which the association relationship plans to connect.
Then it uses the tags <owl:Restriction> and <owl:onProperty> to define the association relationship
“hasStep”. This specifically means the relationship is regarded as a property in OWL and the property
imposes a restriction on the subject class to which the property connects. The restriction means that
the subject class, i.e. “Frame Column”, is restricted to must have the connection with the specified
property “hasStep”. Finally, OWL’s syntax uses the tag <owl:allValuesFrom> and its nested tags to
specify the objects of the relationship. As shown in Figure 3.5, tag <owl:allValuesFrom> specifies that
the object class of the association relationship “hasStep” is a class that is the union of four other
classes, i.e. “Fly Forms to Area to Be Installed”, “Set Pins”, “Stand Forms Into Place” and “Take
Forms Off Cart/Blocking”, all of which are related job steps required by the class “Frame Column” as
shown in Figure 3.2 and need to be represented in OWL as classes as well.

Although the syntax of OWL is well-defined so that it can be used to explicitly model the classified
concepts and their relationships in an ontology, as shown in Figure 3.5, it is cumbersome and not
user-friendly because of the syntax’s complexity (Antoniou and van Harmelen 2004; Wang and Boukamp
2008). To deal with this issue, an OWL-based ontology authoring tool that provides a graphical user
interface for editing the concepts and relationships can be used to help develop an ontology without directly
dealing with the complicated OWL syntax. In this research, an ontology authoring tool, Protégé (Protégé
2008), is adopted to model and edit the concepts and relationships. Protégé is developed by the Stanford
Center for Biomedical Informatics Research at the Stanford University School of Medicine and has become a popular ontology authoring tool for different research domains.

To illustrate how Protégé can help develop an ontology, the following lists the steps that are performed within Protégé to model the concepts and relationships:

**Step 1: Create classes for the primary grouping concepts of classification.** Protégé uses class as the basic unit to stand for concepts (as the OWL syntax <owl:Class> implies). The process of modeling concepts cannot be started until the primary grouping concepts have been created as primary classes in Protégé. In this research, three primary classes for the primary grouping concepts “Activity”, “Job Step”, and “Potential Hazard” are defined to help model all the JHA domain concepts. All these three primary classes are automatically classified as subclasses of the class “owl:Thing”, which is the root class of any ontology in Protégé (see rectangle “1 & 2” in Figure 3.6).

**Step 2: Classify concepts into primary classes.** Once primary classes have been defined, the concepts can then be assigned to the appropriate primary classes. If intermediate classes are required, they should first be defined as subclasses of the primary classes and are used to classify other concepts. For instance, the concept “Frame Column” is added below an intermediate class “Concrete Activity”, which is defined as a direct subclass of the primary class “Activity” (also shown in rectangle “1 & 2” in Figure 3.6).

**Step 3: Define disjoint relationships between classes.** In this step, concept classes should be checked for mutual exclusiveness with other concept classes. For instance, the concept “Excavation Using Slopes” and “Excavation Using Support Systems” are declared disjointed in the research (rectangle 3 in Figure 3.6).

**Step 4: Create properties for relationships between concepts.** Protégé classifies a relationship as a property of related concepts. Thus, relationships are defined in the property editor in Protégé (rectangle 4a in Figure
The properties defined at the primary class level can be used to connect classes below these primary classes. That is, the properties defined in this step are available to all subclasses of the so-called domain and range classes of the properties; in this case, the domain and range classes are the connecting and connected primary classes. For example, if we want to represent “An activity ‘Frame Columns’ has a job step ‘Set Pins’”, primary classes Activity and Job Step should be specified as the domain class and range class respectively for the “hasStep” relationship in Protégé (rectangle 4b in Figure 3.7). Also, the relationship “isStepOf” is also defined as the inverse relationship of “hasStep”. As shown in the rectangle 4a in Figure 3.7, each relationship is paired with its inverse relationship to form a property pair in Protégé.

Figure 3.6: Screenshot of the concept classification editor in Protégé
Step 5: Link classes using properties of relationships. After the types of properties have been defined in Step 4, relationships between the concept classes can now be established using these semantically rich properties. In Protégé, a class to be connected is highlighted first, and a property is then selected as well as a proper quantifier restriction (i.e. the existential quantifier $\exists$, which means “at least one” of the instances of the connected class, or the universal quantifier $\forall$, which means “for all” of the instances of the connected class) and cardinality restriction. For example, an Activity class “Excavation Using Slope” is assigned a property “hasStep” that leads to one Job Step class “Steps of Excavation Using Sloping and Benching as Protective Measure” through both existential quantifier $\exists$ and universal quantifier $\forall$ (rectangle 5 in Figure 3.6). These representations semantically mean that this Activity only comprises this
Job Step (i.e. does not comprise other Job Steps) and such a relation between the Activity and Job Step must exist.

Following this five-step process, a concept representation model, which represents all the concepts in a computer readable format, specifically OWL, can be created and used to support the reasoning process discussed in section 4.2 Concept Ontology Reasoning Engine.

3.3.4 Considerations for Preparing a Comprehensive Ontology

The concepts modeled in an ontology do not have to be limited to those extracted from the targeted construction safety documents, i.e. JHA documents in this research. The concepts with their definitions/explanations from other sources, such as federal safety regulations (U.S. Department of Labor 2003) or OmniClasses (OCCS 2008) can also be incorporated into the ontology as long as they are suitable for describing project situations in the context of construction safety. For example, the OSHA defines a term “overhand bricklaying” in the construction safety document 29 CFR 1926 (U.S. Department of Labor 2003). If this term currently does not exist in the ontology but is considered to be used as an activity in future JHA, it can be added into the ontology and grouped with, if any, other relevant terms, such as “bricklaying”, which already exist in the ontology.

3.3.4.1 Heuristics of introducing new classifications

Incorporating concepts from other construction safety document sources in an ontology can make the
ontology more comprehensive. However, incorporating concepts from different sources into an ontology may result in an unfavorable issue of concept integration. That is, existing classifications may or may not apply to new concepts extracted from other construction safety documents. If new classifications are required to structure the new concepts in the ontology, integrating the existing and new classifications may be a challenge. In this research, the following heuristics, together with illustrative examples, is proposed to address this issue of classification integration. The heuristics is also illustrated in the flowchart shown in Figure 3.8.

First, it has to be evaluated whether the new concepts can fit into the existing classifications. The so-called existing classifications include all the grouping concepts (primary, secondary, tertiary and so on) used in the classifications. Such an evaluation basically checks if the new concepts are proper specialization of the grouping concepts of the existing classifications in terms of their semantic meanings. For example, if a classification has three grouping concepts “Material”, “Labor”, and “Equipment and Machine” and a concept “Truck Crane” needs to be classified, one can quickly determine that this concept should be classified as a sub-concept below “Equipment and Machine” according to the meaning of “Truck Crane”.

For the new concepts which can fit into the existing classifications, it is then determined whether adding new classifications to further classify the new concepts can bring benefits for concept representation and reasoning (detail is discussed in the next subsection). If it can bring benefits, the new classifications are integrated into the existing classifications (i.e. the new ones are subsumed into the existing ones) and the new concepts are then assigned to the new classifications; if it cannot, the new
concepts are directly assigned to the existing classifications. For example, if dozens of concepts belong to
the “Equipment and Machine” group, adding two concepts “Equipment” and “Machine” as sub-concepts
of “Equipment and Machine” can help structure the concepts; if only a few concepts belong to the
“Equipment and Machine” group, there is no necessity for distinguishing equipment from machine in this
group.

For the new concepts which do not fit into the existing classifications, it has to be decided whether
providing new classifications for the new concepts can bring benefits for concept representation and
reasoning. If it can bring benefit, the new classifications are added in the class hierarchy parallel to the
existing classifications. Then, the new concepts are assigned to the new classifications. If, on the other hand,
it cannot, the new concepts are just added to the class hierarchy parallel to the existing classifications. For
example, if a concept “Electricity” needs to be classified in the classification of the previous example,
none of the existing concepts is a suitable super-concept for it and thus, this concept can be added on the
same hierarchical level as the concepts “Material”, “Labor” and “Equipment and Material” do. If more
other concepts, such as “Water”, “Wells”, “Sanitary Sewerage” and “Storm Drainage”, also are to be
classified, first adding another concept “Utility” parallel to the concept “Material” and then put these
concepts below the concept “Utility” would be better.

3.3.4.2 Evaluate benefits of adding new classifications

The evaluation of adding new classifications in an existing classification should consider whether
concept representation and reasoning can gain benefits from the new classifications. I discuss them
respectively in the following paragraphs.

From the representation viewpoint, adding new classifications is deemed to be beneficial to concept representation if adding the new classifications can improve the structure of the concepts and largely reduces the number of subclasses on one hierarchy level by distributing them into lower, more specific hierarchy levels. Adding new classifications usually becomes useful when the number of the subclasses in a class is too large to be easily maintained and manipulated. In other words, introducing an additional hierarchical structure would allow splitting the number of subclasses into different branches in the hierarchy, making the hierarchy easier to manage and maintain. An example of this discussion is given in section 3.3.1, which is about adding concepts “Beam” and “Wall” in an existing classification.

From the reasoning perspective, adding new classifications is basically beneficial to concept reasoning as some safety documents may contain contextual information only relevant to the introduced generalizations of existing concepts. Without the generalized concepts, those relevant documents cannot be retrieved and thereby, the reasoning about concepts will be detrimentally affected. Hence, to evaluate whether adding new classifications is more beneficial for concept reasoning, one should consider if it allows more associations to be established between the classes of the new classifications and existing classes in the ontology. That is, if the classes of the new classifications can be connected to other existing ones after they are added into the ontology, such new classifications are deemed to be more helpful. The reason is they allow more channels for propagating applicability between concepts and thus, enable the reduction of isolated clusters of concepts, which are concepts in class hierarchies without associations with other hierarchies.
3.4 TEXTUAL DOCUMENT REPRESENTATION MODEL

Textual document representation aims at modeling the construction safety documents in a computer readable and interpretable format. The representation has to be flexible because it may be used for diverse construction safety documents and it should be able to accommodate the differences between these documents’ skeletons. This representation also has to be maintainable since it should let document users edit the documents without difficulties. In addition, the computer readable and interpretable format should be capable of facilitating the process of accessing the information within the documents in order to
automate the identification and classification of safety requirements.

I decide to use Extensible Markup Language (XML) to model the construction safety documents in the developed document representation approach as it is a modeling language which can fulfill the aforementioned requirements. If a document is modeled in XML format, its content is well organized in an ordered hierarchy and marked up with human-readable, descriptive tags. Document users can easily edit the document in any text editor without requiring a specific software application by following the content hierarchy and self-explanatory tags.

In this section, I will discuss the developed textual document representation approach for modeling construction safety documents. To model a construction safety document in XML format, the steps to be taken consists of: (1) analyzing the document, (2) defining a schema, and (3) representing the document in XML format according to the specification of the schema. In the following sections, I first propose a general structure for construction safety documents and then discuss the three aforementioned steps in turn.

3.4.1 A General Structure of Construction Safety Documents

In this section, I first look into two construction safety documents to identify the common characteristics of these safety documents (in section 3.4.1.1), and propose a general structure for construction safety documents based on the identified characteristics (in section 3.4.1.2).
3.4.1.1 Study of construction safety documents’ characteristics

In the first study, I focus on an OSHA construction safety document. Figure 3.9 shows a snippet of the standard 1926.703 extracted from the OSHA construction safety document, 29 CFR 1926 Subpart Q. This standard snippet indicates some important characteristics of the example document.

(1) The document, subpart Q, is composed of multiple safety requirements, i.e. from 1926.703(a)(1) to 1926.703(e)(2).

(2) Each requirement can comprise sub-requirements, such as 1926.708(b)(8) having three sub-requirements from 1926.708(b)(8)(i) to (iii) (not shown in Figure 3.9).

(3) Each requirement contains the information of conditions, which specify the circumstances under which the specification is applicable or not applicable. For example, the standard 1926.703(a)(1) is applicable to “Cast-In-Place Concrete” and “Framework”, which are specified respectively in the title of the standard 1926.703 and 1926.703(a).

(4) In addition to the information of conditions, each requirement mostly contains the rules that describe necessary actions that should be taken when its conditions are satisfied. For example, the rules of the standard 1926.703(a)(1) are “Formwork shall be…to the formwork” and “Formwork which is designed,……of this paragraph”.

Additional, I focus on JHA documents in another document study. I use the JHA document shown in Figure 3.2 as an example of illustration and Figure 3.10 shows part of the JHA document. First, I found that the document is composed of multiple safety requirements, which are represented in 4-tuples of activity, job step, potential hazard and safety rules. Second, each safety requirement contains the
information of conditions, which are the activity, job step and potential hazard information. For example, the first safety requirement in Figure 3.10 applies to the activity “Frame Column”, job step “Fly forms to area to be installed” and potential hazard “Material dislodgement”. Third, each safety requirement also contains safety rules in addition to the information of conditions. For example, there are five safety rules in the first safety requirement.

OSHA 29 Code of Federal Regulation
Part 1926: Safety and Health Regulations for Construction
Subpart Q: Concrete and Masonry Construction
1926.703 Requirements for cast-in-place concrete
1926.703(a) General requirements for formwork
1926.703(a)(1)
Formwork shall be designed, fabricated, erected, supported, braced and maintained so that it will be capable of supporting without failure all vertical and lateral loads that may reasonably be anticipated to be applied to the formwork.

Formwork which is designed, fabricated, erected, supported, braced and maintained in conformance with the Appendix to this section will be deemed to meet the requirements of this paragraph.

1926.703(e)(2)
Reshoring shall not be removed until the concrete being supported has attained adequate strength to support its weight and all loads in place upon it.

Figure 3.9: Snippet of the OSHA construction safety document

<table>
<thead>
<tr>
<th>Job Steps</th>
<th>Potential Hazards</th>
<th>Recommended Safety Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fly forms to area to be installed</td>
<td>Material dislodgement</td>
<td>• Inspect rigging to be/being used</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Ensure proper rigging method</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Ensure direct contact with crane operator</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Clear area to land material</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Use taglines</td>
</tr>
<tr>
<td>Take forms off cart/ blocking</td>
<td>Sprain/Strain of back</td>
<td>• Get assistance; Work with a partner.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Discuss lifting and moving process with partner.</td>
</tr>
<tr>
<td></td>
<td>Slip/Trip/Fall</td>
<td>• Clear path from cart to work location.</td>
</tr>
</tbody>
</table>

Figure 3.10: Snippet of the example JHA document
3.4.1.2 A general structure for construction safety documents

In addition to the studies and observations discussed in the previous subsection, previous research on specification representation also identified that specifications can have applicability conditions and applicability exceptions and can impose different types of requirements (Slava 1985). While Slava’s research was mainly to support the writing of construction specifications, the general structure identified for construction specifications was indeed a good reference for the representation of construction safety documents discussed in this section. For example, each safety requirement should have applicability conditions and, if necessary, applicability exceptions. Therefore, a general structure for construction safety documents can be proposed. The general structure is represented in an object-oriented manner for the convenience of demonstration and understanding, which is based on the one shown in Figure 2.2.

In the document representation approach developed in this research, a construction safety document can be composed of multiple Safety Requirements, which specify the regulations project employees should abide by. Each Safety Requirement can be composed of Conditions, which specify the circumstances under which their corresponding rules apply or do not apply, and Rules, which are necessary actions that should be taken when their related conditions are satisfied. Each Safety Requirement also can be composed of other Safety Requirements to form a nesting structure of safety requirements. Figure 3.11 shows a UML diagram that illustrates the decomposition of a construction safety document.
The *Conditions* can be further classified into two parts: *Applicability Conditions*, which are the conditions in which their corresponding rules apply, and *Applicability Exceptions*, which are the exceptions in which their corresponding rules do not apply. In this research, each *Condition* is represented through one *Context* and the *Context* is described through *Concepts*, i.e. the concepts from the developed ontology or their logical concatenation. That is, the three primary grouping concepts, Activity, Job Step and Potential Hazard, discussed in the previous section and their sub-concepts can be used as *Concepts* to describe the *Contexts* for the applicable conditions and exceptions of the *Safety Requirements*. Figure 3.12 shows a UML diagram that illustrates the decomposition of the *Conditions* for JHA documents.

The proposed general structure of construction safety documents indicates that developed textual document representation model is context-based. Both the representation of applicability conditions/exceptions and the concepts extracted from documents associate with contexts closely, as can be seen in Figure 3.11 and Figure 3.12. In this research, I focus on the role of contexts and do not consider other characteristics in the representation of applicability conditions/exceptions which construction safety documents may contain in their safety requirements, such as concepts’ property values (examples are the
height of a working level or the rated capacity for a single leg sling). This strategy of focusing on contexts allows me to look further into how contexts’ semantics can be leveraged to help identify construction safety requirements. The evaluation of concepts’ property values is out of the scope of this research and is discussed in chapter 6 as one of the future research directions.

Figure 3.12: Decomposition of the condition for JHA documents

3.4.2 Document Analysis

The goal of analyzing a construction safety document is to understand what its constituent elements are and how the constituent elements form the document content. The constituent elements identified should be specific enough to enable appropriate structuring of the documents’ information in XML and to facilitate the process of reasoning about the document. For example, the constituent elements identified for JHA documents are Activity, Job Step, Potential Hazard and Recommended Safety Procedure, as discussed and pointed out in section 3.3, and these four elements can comprehend and classify all important
information of a JHA document.

In addition, different JHA sections of a JHA document can be defined using these four constituent elements. For example, a Potential Hazard section in a JHA document is formed with a Potential Hazard element and its corresponding Recommended Safety Procedure element. As shown in Figure 17, the example JHA document comprises three Potential Hazard sections, the first of which is led by the Potential Hazard element “Material dislodgement” followed by its five corresponding Recommended Safety Procedure elements. Similarly, a Job Step section is formed with a Job Step element, its associated Potential Hazard elements and Recommended Safety Procedure elements; an Activity section is formed with all the four constituent elements. Due to the nesting structure of JHA documents, an Activity section can contain multiple Job Step sections while a Job Step section also can contain multiple Potential Hazard sections.

The document analysis task should also take into account the structure of construction safety documents shown in Figure 3.11 and Figure 3.12 after the constituent elements are identified. That is, the analysis task has to identify which elements of a document provide concepts that act as context descriptors in order to represent Conditions and which elements have the information for Rules. For example, for a JHA document, the information of the constituent elements Activity, Job Step, and Potential Hazard are used as Conditions and Recommended Safety Procedure provides the information for Rules.
3.4.3 Schema Definition for Modeling Document

After the construction safety documents’ constituent elements are identified, the next step for textual document representation is to define a schema that can formalize the structure of the identified constituent elements. The purpose of defining the schema is to specify exactly what markup tags should be defined to represent the elements and how they are organized in the XML representation.

Document type definitions (DTDs) and XML Schema Definition (XSD) are two popular approaches which are often used to define such a schema. They can define the basic constituent elements as well as structural rules, i.e. what elements have to occur in which place in the XML document and what the contents and/or attributes of the elements should be. Because XSD has better control over the data types and format of elements and attribute values than DTDs (Harold and Means 2004), XSD is adopted in this research. In addition, there are many XML Schema languages capable of defining XML Schema. The one developed by World Wide Web Consortium, W3C XML schema language (W3C 2008), is selected to create XSD for construction safety documents due to its popularity over others.

Figure 3.13 illustrates the snippet of an XSD for the example JHA document shown in Figure 3.2. First of all, XSD uses <xs:element> to define a new tag named JHA as the root tag for a JHA document and assigns a user-defined data type “JHAType” to the JHA tag (line 1). A user-defined data type assigned to a tag specifies what other tags will be nested in that tag. For instance, “JHAType” is defined, through <xs:complexType> (line 2), as a data type which can in sequence present multiple Activity tags (line 3-6). In other words, a JHA tag can contain multiple Activity tags in itself; these definitions create the schema of a
JHA document as follows:

```xml
<JHA>
  <Activity>……</Activity>
  <Activity>……</Activity>
  ......
</JHA>
```

Because the tag `Activity` defined in line 4 is assigned another user-defined data type “ActType”, the rest of the snippet in Figure 3.13 continues on defining this new data type. Line 7 to 15 of the schema declare that “ActType” is a data type which can in sequence present five tags: `ActTitle` tag, which surrounds the activity’s name; `Ctxt_App_Condition` and `Ctxt_App_Exception` tags, which respectively surround the contexts representing the applicability conditions and applicability exceptions of the safety requirements; `ActDescription` tag, which surrounds the activity’s description; and `JobStep` tag, which groups the job step information of the activity.

```
1  <xs:element name="JHA" type="JHAType"/>
2  <xs:complexType name="JHAType">        
3     <xs:sequence>
4       <xs:element name="Activity" maxOccurs="unbounded" type="ActType"/>
5     </xs:sequence>
6  </xs:complexType>
7  <xs:complexType name="ActType">        
8     <xs:sequence>
9       <xs:element name="ActTitle" type="xs:string"/>
10      <xs:element ref="Ctxt_App_Condition" minOccurs="0"/>
11      <xs:element ref="Ctxt_App_Exception" minOccurs="0"/>
12       <xs:element name="ActDescription" type="xs:string" minOccurs="0"/>
13       <xs:element name="JobStep" maxOccurs="unbounded" type="StepType"/>
14     </xs:sequence>
15  </xs:complexType>
```

Figure 3.13: Snippet of an XSD for the example JHA document

In summary, following the schema definition process, I can in turn formalize JHA documents’ four constituent elements by giving them respective tags, such as the tag `<Activity/>`\(^1\) for the Activity element.

\(^1\) `<Activity/>` is symbolically equal to `<Activity></Activity>`, which means a pair of tags without content. However, I use this representation here and throughout the dissertation only for the purpose of indicating the notion of tag pairs and do not imply that there should be no content in them.
In addition, three other types of tags are defined in this research.

- **Descriptive and informative tags**: These tags are defined to further elaborate the four constituent elements, such as `<ActTitle/>` and `<ActDescription/>` for elaborating Activity title and description. For example, the activity in the example JHA document in Figure 3.10 can be represented in XML format as

```
<ActTitle>Frame Column</ActTitle>
<ActDescription>This activity aims at framing columns with formwork</ActDescription>
```

- **Concept tags**: These tags are `<Concept/>`, `<Concept_AND/>`, and `<Concept_OR/>`. They are defined for indicating the information of concepts. If a context is described through single concept, the tag `<Concept/>` should be used; otherwise, `<Concept_AND/>` or `<Concept_OR/>` should be used depending on whether the multiple concepts are conjunctively or disjunctively related. They are discussed in detail in section 4.3.1.

- **Reasoning tags**: These tags are `<Ctxt_App_Condition/>` and `<Ctxt_App_Exception/>`, which are defined to respectively describe the applicability conditions and exceptions of safety requirements. These tags will contain exactly one of the concept tags as the contexts for applicability conditions and exceptions are described through either a concept or a concept conjunction or a concept disjunction.

All the tags, which are defined in this research for representing JHA documents, are summarized in Table 3.1.
Table 3.1: Listing of the defined XML markup tags for JHA documents

<table>
<thead>
<tr>
<th>No</th>
<th>Markup Name</th>
<th>Markup Symbol</th>
<th>Markup Description</th>
<th>Inner Element</th>
<th>Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>JHA</td>
<td>&lt;JHA/&gt;</td>
<td>Root tag, starting and ending a JHA document</td>
<td>1</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>Activity</td>
<td>&lt;Activity/&gt;</td>
<td>Tag grouping Activity information</td>
<td>3, 4, 5, 13, 14</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>Activity Title</td>
<td>&lt;ActTitle/&gt;</td>
<td>Tag surrounding the Activity’s title</td>
<td>String</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>Activity Description</td>
<td>&lt;ActDescription/&gt;</td>
<td>Tag surrounding the Activity’s description</td>
<td>String</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Job Step</td>
<td>&lt;JobStep/&gt;</td>
<td>Tag grouping Job Step information</td>
<td>6, 7, 8, 13, 14</td>
<td>No</td>
</tr>
<tr>
<td>6</td>
<td>Job Step Title</td>
<td>&lt;StepTitle/&gt;</td>
<td>Tag surrounding the Job Step’s title</td>
<td>String</td>
<td>No</td>
</tr>
<tr>
<td>7</td>
<td>Job Step Description</td>
<td>&lt;StepDescription/&gt;</td>
<td>Tag surrounding the Job Step’s description</td>
<td>String</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>Potential Hazard</td>
<td>&lt;PotentialHazard/&gt;</td>
<td>Tag grouping Potential Hazard information</td>
<td>9, 10, 11, 13, 14</td>
<td>No</td>
</tr>
<tr>
<td>9</td>
<td>Potential Hazard Title</td>
<td>&lt;HazardTitle/&gt;</td>
<td>Tag surrounding the Potential Hazard’s title</td>
<td>String</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>Potential Hazard Description</td>
<td>&lt;HazardDescription/&gt;</td>
<td>Tag surrounding Potential Hazard’s description</td>
<td>String</td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>Recommended Safety Procedure</td>
<td>&lt;RecommendedProcedure/&gt;</td>
<td>Tag grouping Recommended Procedure information</td>
<td>12</td>
<td>No</td>
</tr>
<tr>
<td>12</td>
<td>Rule</td>
<td>&lt;Rule/&gt;</td>
<td>Tag surrounding the Rule for the Recommended Procedure</td>
<td>String</td>
<td>Yes</td>
</tr>
<tr>
<td>13</td>
<td>Applicability Condition</td>
<td>&lt;Ctxt_App_Condition/&gt;</td>
<td>Tag grouping the Applicable Conditions of the safety requirements</td>
<td>15 or 16 or 17</td>
<td>No</td>
</tr>
<tr>
<td>14</td>
<td>Applicability Exception</td>
<td>&lt;Ctxt_App_Exception/&gt;</td>
<td>Tag grouping the Applicable Exceptions of the safety requirements</td>
<td>15 or 16 or 17</td>
<td>Yes</td>
</tr>
<tr>
<td>15</td>
<td>Concept</td>
<td>&lt;Concept/&gt;</td>
<td>Tag surrounding exact one Concept</td>
<td>String</td>
<td>Yes</td>
</tr>
<tr>
<td>16</td>
<td>Concept Conjunction</td>
<td>&lt;Concept_AND/&gt;</td>
<td>Tag representing conjunction of multiple Concepts</td>
<td>15, 17</td>
<td>Yes</td>
</tr>
<tr>
<td>17</td>
<td>Concept Disjunction</td>
<td>&lt;Concept_OR/&gt;</td>
<td>Tag representing disjunction of multiple Concepts</td>
<td>15, 16</td>
<td>Yes</td>
</tr>
</tbody>
</table>

1 The number assigned to each tag here is only for the convenience of indicating nesting relations in “Inner Element” column.
2 Column “Inner Element” shows the content inside a tag, which is a tag or a set of tags or a string.
3.4.4 Modeling Documents in XML Format

Figure 3.14 illustrates a snippet of an XML document that is based on the example JHA document in Figure 3.2. The essential information of the JHA document is currently represented in XML format using the defined tags listed in Table 3.1 and following the schema definition. For example, the tag `<Activity/>` groups the tag `<ActTitle/>`, which surrounds the activity’s name “Frame Columns”; the tag `<Ctxt_App_Condition/>`, which specifies the context described through the concept “Frame Columns” as the applicability condition; and the tag `<JobStep/>`, which further groups other tags.

```
<Activity>
  <ActTitle>Frame Columns</ActTitle>
  <Ctxt_App_Condition>
    <Concept>Frame_Columns</Concept>
  </Ctxt_App_Condition>
  <JobStep>
    <StepTitle>Fly forms to area to be installed</StepTitle>
    <Ctxt_App_Condition>
      <Concept_AND>
        <Concept>Frame_Columns</Concept>
        <Concept>Fly_Forms_To_Area_To_Be_Installed</Concept>
      </Concept_AND>
    </Ctxt_App_Condition>
    <PotentialHazard>
      <HazardTitle>Material dislodgement</HazardTitle>
      <Ctxt_App_Condition>
        <Concept_AND>
          <Concept>Frame_Columns</Concept>
          <Concept>Fly_Forms_To_Area_To_Be_Installed</Concept>
          <Concept>Material_Dislodgement</Concept>
        </Concept_AND>
      </Ctxt_App_Condition>
      <RecommendedProcedure>
        <Rule>(1) Inspect rigging to be/being used.</Rule>
        <Rule>(2) Ensure proper rigging method.</Rule>
        <Rule>(3) Ensure direct contact with crane operator.</Rule>
        <Rule>(4) Clear area to land material.</Rule>
        <Rule>(5) Use taglines.</Rule>
      </RecommendedProcedure>
    </PotentialHazard>
  </JobStep>
</Activity>
```

Figure 3.14: Snippet of the example JHA document in XML format
As discussed in section 3.4.1, in this research, the contexts for the applicability conditions and applicability exceptions of a JHA document are described through the concepts representing the activities, job steps and potential hazards. In other words, these contexts for the applicability conditions and applicability exceptions shall expand with the development of the nesting structure from tag \textit{Activity} to tag \textit{PotentialHazard} of a JHA document. For instance, only one context “Frame Columns” is needed in the nesting level of tag \textit{<Activity/>} while another context “Fly Forms to Area to Be Installed” is added into the applicability condition and disjunctive to the first context in the nesting level of tag \textit{<JobStep/>}, as shown in Figure 3.14. This characteristic of context expansion not only allows for different applicability conditions that each nesting level possesses but also facilitates the reasoning process to faster identify the applicability of safety requirements, which will be discussed in section 4.3.1. In addition, as discussed in section 3.4.3, when the context is described through a single concept, tag \textit{<Concept/>} should be used, which is shown in line 4 of Figure 3.14 in this example. If there are two or more concepts describing the context, tag \textit{<Concept_OR/>} is used to surround these concepts, each of which is surrounded by tags \textit{<Concept/>} as shown in line 9 to 12 and line 17 to 21 of Figure 3.14.

3.5 SUMMARY OF THE REPRESENTATION MODEL

In this chapter, I start with the overview of the developed Construction Safety Document Management Framework, including brief introduction of its two constituent parts: Representation Model and Reasoning Engine, and then discuss the Representation Model in detail. The Representation Model consists of two sub-models: the concept ontology representation model and the textual document representation model.
The concept ontology representation model illustrates the steps for modeling concepts in an ontology, which are classification of concepts, definition of relationships between the classified concepts and ontological modeling of the concepts and relationships. I also discuss the considerations of developing a comprehensive concept ontology, in which a heuristic approach is proposed to help evaluate the integration of classifications. On the other hand, the textual document representation model portrays the steps of preparing construction safety documents in XML format, which are document analysis, schema definition for modeling documents. In addition to these steps, a general structure of construction safety documents demonstrated in an object-oriented manner is also proposed. The structure indicates the basic framework a construction safety document should have in order to be modeled in XML format and therefore allows facilitating the document modeling process.

The Representation Model presented in this chapter helps the development of the Reasoning Engine. In the next chapter, I discuss in detail how the Reasoning Engine is developed and works on the foundation of the Representation Model.
4.1 INTRODUCTION TO THE REASONING ENGINE

The Reasoning Engine is the second part of the proposed Construction Safety Document Management Framework. It consists of two sub-engines: the concept ontology reasoning engine, which reasons about the concept ontology prepared through the procedure introduced in section 3.3, and the safety requirement reasoning engine, which reasons about the JHA documents modeled in XML format by following the process described in section 3.4. These two sub-engines are discussed in the following sections.

4.2 CONCEPT ONTOLOGY REASONING ENGINE

The goal of the reasoning engine is to evaluate the applicability of each concept of the concept ontology according to a set of concepts for which the applicability values are known. The reasoning engine, therefore, should explore and navigate through the ontology by traversing the different relationships between concepts in the ontology whereby it should intelligently propagate the given concepts’ applicability values to other related concepts.

The concept ontology specifies the concepts of the construction safety documents and how these concepts relate to each other. However, if there is no reasoning engine defined that is able to leverage the concept ontology and the knowledge modeled as semantic relationships between the concepts, the concept ontology is at best only a knowledge representation and provides little benefit for the automated
identification of construction safety requirements. The proposed concept ontology reasoning engine utilizes the concepts and their semantic relationships modeled in the concept ontology is discussed in the following sub-sections.

4.2.1 Reasoning Premises

The goal of the reasoning engine, to evaluate the applicability of each concept of the concept ontology according to a set of concepts for which the applicability values are known, implies two reasoning premises for the reasoning engine:

(Premise 1) Concept applicability premise

Each concept has three options of applicability: applicable, not applicable and possibly applicable, and each concept must carry exactly one of these applicability options at any given time.

The concept applicability premise\(^1\) specifies the possible applicability a concept can carry: a concept is either applicable, which means the domain phenomenon described by the concept applies to the current situation; or not applicable, which means the phenomenon does not apply; or possibly applicable\(^2\), which means the information to determine whether the domain phenomenon described by the concept applies or not is insufficient so the applicability is unknown. The concept applicability premise is important as it lays a foundation for the reasoning engine.

---

\(^1\) This reasoning premise implies that the concept ontology reasoning engine takes the open-world assumption: a concept cannot be concluded to be not applicable just because it cannot be shown to be applicable, and therefore falls into the possibly applicable category.

\(^2\) One may also use possibly inapplicable to describe the applicability value of concepts falling into the possibly applicable category. Both possibly applicable and possibly inapplicable mean the applicability value of the concepts is unknown. I use possibly applicable in this research to represent engineers’ preference for having applicable concepts/safety requirements identified, and possibly applicable literally reflects this expectation more that possibly inapplicable does.
(Premise 2) Context availability premise

A project’s context, which defines a project's situation, is partially known before starting the engine’s reasoning process, and the context is described through concepts whose applicability is set as either applicable or not applicable.

The context availability premise requires that some information about the context defining a project’s situations is known before the reasoning process begins. The context can be hypothesized or identified through context identification techniques, such as manual observation and identification, which is adopted in this research. The context is then described through a selection of concepts from the concept ontology that describe the actual situation by setting their applicability value to either applicable or not applicable. This selection of concepts describing the identified context is used as the initial input into the reasoning process. Therefore, the context availability premise is significant in that it guarantees that initial input into the reasoning engine exists.

The concept applicability premise and context availability premise can be further formalized and elaborated using epistemic modal logic. Garson (2009) defined modal as “an expression, such as necessarily or possibly, that is used to qualify the truth of a judgment”. Therefore, modal logic is a type of formal logic which is concerned with statements which are qualified in different modalities, such as possibility, probability or necessity. For example, to say “It is possible that the last batch of rebar will arrive this week.” logically is to make a statement, with the modality of possibility, that the last batch of rebar could not arrive at site this week. Epistemic modal logic is a type of modal logic regarding knowledge and belief (Hendricks and Symons 2009). Specifically, a statement made with epistemic modality represents how well the person making the statement knows the information in the statement.
For example, a sentence “The construction manager knows that the last batch of rebar will arrive this week.” is made with epistemic modality. The difference between this and the last example is that the construction manager in this example has knowledge of (or certainty about) the described event while the speaker in the last example doesn’t.

Therefore, given the context availability premise, the three options of applicability, applicable, not applicable and possibly applicable can be interpreted using epistemic modality as follows: A concept is applicable or not applicable means that an engineer knows or is certain that the domain phenomenon described by the concept applies or does not apply to the current situation (i.e. the known context). They can be represented in unary predicates with the epistemic modal operator $\Box$ as “$\Box \text{Applicable}(x)$” and “$\Box \sim \text{Applicable}(x)$” respectively, where $x$ stands for any concept and $\text{Applicable}(x)$ stands for the concept $x$ is applicable. In additions, a concept is possibly applicable means that for all an engineer knows, the domain phenomenon described by the concept is not known to be applicable to the current situation and it also is not known to be not applicable. This can be represented in a unary predicate using the predicate $\text{Applicable}(x)$ and the epistemic modal operator $\Diamond$ as “$\Diamond \text{Applicable}(x) \& \Diamond \sim \text{Applicable}(x)$”.

Epistemic modal logic provides a formalization of the concept applicability premise and the context availability premise. This is especially important for elaborating the applicability of possibly applicable because its formalization reasonably explains and emphasizes that the indetermination of a concept’s applicability grounds on one’s available knowledge of the given context.
4.2.2 Concept Ontology Reasoning Principles

Starting with the two reasoning premises, there are six concept ontology reasoning principles that guide the reasoning process of the concept ontology reasoning engine. Most of the principles utilize the ontological relationships defined between concepts to facilitate the reasoning process.

(Principle 1) Initialization Principle

All concepts are initially set as possibly applicable before the reasoning engine is activated.

Before the reasoning engine is activated and before input about a context is provided, all the concepts of the concept ontology are initially set as possibly applicable since there is no information to help judge whether the concepts are applicable or not at this moment. Once updated information, such as the initial input of an identified context, is available, the applicability of the concepts of the concept ontology will be updated accordingly by adhering to the following reasoning principles.

(Principle 2) Hierarchical Propagation Principle

If a concept is applicable, its super-concepts must be applicable. If a concept is not applicable, its sub-concepts must be not applicable.

A concept’s super-concepts are the generalization of the concept. If the concept is applicable, its generalized super-concepts must have the same applicability. Similarly, a concept’s sub-concepts are the specialization of the concept. If the concept is not applicable, there is no chance for its specialized sub-concepts to be applicable, i.e. they must be not applicable. In other words, the applicability value

---

1 The proposed reasoning principles can also be symbolically formalized using Description Logic (DL) for evaluating the concept ontology reasoning engine’s computational complexity and decidability. In this research, I do not use DL to formalize the reasoning principles because it cannot represent the change of each concept’s applicability value just according to the principles.
applicable of a concept can be propagated upwards to higher level concepts of a concept ontology while the applicability value not applicable of a concept can be propagated downwards to lower level concepts of a concept ontology. For instance, if a concept “Retaining Wall” is applicable for a project, its super-concept “Wall” must be applicable; if the concept “Wall” is not applicable, its sub-concept “Retaining Wall” must be not applicable.

(Principle 3) Non-logical Association Propagation Principle

If a concept is applicable, another concept which is connected to the first concept through a non-logical association relationship must be applicable.

This principle only applies when a concept is found applicable. The non-logical association relationships are used to semantically connect concepts which are related with each other and, therefore, indicate that when the connecting concept is applicable, the connected concept should be applicable as well. For example, the Job Step concept “Set Pins” connects with the Potential Hazard concept “Fall” through a non-logical association relationship “hasHazard” (according to the semantic relationship matrix in Figure 3.3). Once the concept “Set Pins” is found applicable, the concept “Fall” will accordingly become applicable.

(Principle 4) Equivalent Association Propagation Principle

Two concepts connected through an equivalent association relationship must carry the same applicability value.

Two concepts which are connected through an equivalent association relationship have the same contextual meaning, and therefore, must share the same applicability value, no matter whether this value is
applicable, not applicable, or possibly applicable. For example, if the applicability of the Potential Hazard concept “Slip” is applicable, another Potential Hazard concept “Trip” must carry the applicability value applicable as well since these two concepts are equivalent as discussed in section 3.3.2.

(Principle 5) Disjoint Association Propagation Principle

If a concept is applicable, another concept which is connected to the first concept through a disjoint association relationship must be not applicable.

This principle only applies when a concept is found applicable. Concepts which are connected through disjoint association relationships semantically exclude one another. Therefore, when one concept is applicable, this concept will exclude other concepts to which it connects through disjoint relationships and require that those concepts be not applicable. For example, if the Activity concept “Excavation Using Support Systems” is applicable, another Activity concept “Excavation Using Slopes” must be not applicable since these two concepts are declared as disjointed as discussed in section 3.3.3.

(Principle 6) Concept Combination Propagation Principle

If a combined concept in the concept ontology is applicable, the applicability value is propagated to its constituent concepts of the concept combination.

This principle is a variant of the principle 3. A combined concept uses non-logical association relationships to represent the relations between it and the constituent concepts (as discussed in section 3.3.2.1). Therefore, if a combined concept is applicable, by applying principle 3, its constituent concepts must be applicable. Similarly, if a concept combination is not applicable, this applicability value will not be propagated to its constituent concepts according to principle 3.
These six reasoning principles guide the reasoning process when the reasoning engine is activated. After the reasoning process, the concept ontology is updated and some concepts in the concept ontology will carry an applicability value different from the initial *possibly applicable* value. These are the concepts that were assigned different values in the beginning to describe a certain project context and the concepts to which different applicability values were propagated following the above reasoning principles. However, during the reasoning process, it is possible to encounter applicability contradictions for a concept, e.g. a concept may first be found *applicable* but then assigned the value *not applicable* later which contradicts the first applicability value. In the next section, I discuss when and how the concept applicability contradictions may happen and how a contradiction can be resolved.

4.2.3 Concept Applicability Contradiction

4.2.3.1 Causes of concept applicability contradiction

In this research, I observed two main causes for concept applicability contradiction. The following section will discuss the causes in detail.

(1) *Flawed concept ontology*

The first cause for concept applicability contradiction can be that the concept ontology is flawed. As discussed in section 3.3, the most essential step of developing a concept ontology is to define relationships between concepts. If the required relationships are not defined or the defined relationships are not appropriately used to connect concepts, the concept ontology may represent the domain knowledge
incorrectly, which will result in applicability contradiction when the reasoning engine interprets the concept ontology. For example, Figure 4.1 shows a simple concept ontology, in which two main concepts are defined: the concept “Activity”, which has a sub-concept “Pour”, and the concept “Resource”, which has a sub-concept “Concrete Bucket”. “Pour” and “Concrete Bucket” are connected through an association relationship while Activity and Resource are connected through a disjoint relationship. If the concept “Pour” is found applicable in a project’s concrete pouring activity, the concepts “Activity” and “Concrete Bucket” should be applicable according to the principle 2 and 3 respectively. However, the applicability contradiction occurs on the concept “Resource” because it should be not applicable due to the disjoint relationship with the applicable concept “Activity” (principle 5) but should also be applicable due to its applicable sub-concept “Concrete” (principle 2). This contradiction, therefore, indicates that concept ontology is problematic due to the disjoint and the association relationship.

![Figure 4.1: Applicability contradiction example in a flawed concept ontology](image)

(2) Application of the reasoning principles to multiple input concepts

A context is described by setting multiple concepts’ applicability values, which act as input to the reasoning engine. These multiple input concepts will be reasoned about sequentially in the reasoning engine, i.e. the reasoning engine takes each concept’s applicability value and propagates it throughout the ontology following the reasoning principles. If the concept ontology is not flawed, an applicability
contradiction may still occur as a result of applying the reasoning principles to the multiple input concepts. For example, Figure 4.2 shows an illustrative concept ontology, in which the concept “Activity”, which has a sub-concept “Cast”, is defined. The concept “Cast” has two disjointed sub-concepts “Cast-In-Place” and “Precast”. If the identified context of a project includes both “Cast-In-Place” and “Precast”, e.g. a precast concrete column placed on cast-in-place concrete foundation, both of the concepts “Cast-In-Place” and “Precast” bare assigned the applicability value applicable that becomes the input for the reasoning engine. Starting with concept “Cast-In-Place”, the reasoning engine determines that concepts “Cast” and Activity are applicable according to the principle 2, and the concept “Precast” is not applicable according to the principle 5. However, when the engine processes the second input concept Precast, the concept’s initial applicability value, i.e. applicable, conflicts with its applicability value as determined by the reasoning engine, i.e. not applicable. This conflict, therefore, results in an applicability contradiction.

![Figure 4.2: Applicability contradiction example of multiple input concepts](image)

4.2.3.2 Resolutions of concept applicability contradiction

In order to address applicability contradictions, a contradiction detection mechanism had to be implemented in the reasoning engine. This contradiction detection mechanism should be able to not only send warning messages to users when a contradiction is detected, but also to provide users with options to
accept or decline the new, contradicting applicability values of concepts.

When warning messages are sent to users, e.g. in the form of a pop-up dialogue on the screen, users should first respond to the warning messages by determining to accept or decline the new applicability value that results in the contradiction. At this point, users get a chance to evaluate the alternatives of applicability given by the reasoning engine and from the initial input, and determine which applicability value better describes the current project situations. For instance, in the example concept ontology shown in Figure 4.2, one should accept the suggested applicability value, i.e. *applicable*, which conflicts with the computed value, i.e. *not applicable*, for the concept Precast because the concept Precast is an identified context.

On the other hand, users (or ontology developers, if users are not proficient in editing ontologies) need to inspect the concept ontology to see whether there is any flaw that results in a contradiction in the reasoning process. If any flaw is found, the concept ontology should be corrected accordingly, e.g. by revising the concept definitions or relationships assigned to concepts. For example, to resolve the applicability contradiction shown in Figure 4.1, one can simply remove the disjoint relationship between concept Activity and concept Resource to correct the flawed example concept ontology.

The contradiction detection mechanism with the user response function shall allow feedback on the concepts’ current applicability status from users. Therefore this mechanism facilitates the refinement of the concepts’ applicability of the concept ontology by resolving applicability contradictions. The mechanism also reinforces the reasoning engine by complementing the reasoning process, and will benefit the
application of the concept ontology to the safety requirement reasoning engine, which is discussed in the next section.

4.2.4 Forward and Backward Reasoning about Concepts

In this research, the concept ontology representation model allows an association relationship to be defined as an inverse relationship to another association relationship. When two concepts are connected together through such bi-directional association relationships, the developed concept ontology reasoning engine allows two ways to reason about the concepts by deploying the reasoning principle 3: forward reasoning and backward reasoning. Forward reasoning, for example, means to propagate applicability value applicable from Activity concepts to Job Step concepts through association relationship “hasStep”; on the other hand, backward reasoning means to propagate applicability value applicable the other way around through association relationship “isStepOf”.

The two-way reasoning will result in a situation in which some concepts can be evaluated to be applicable, but such information may be unnecessary. Figure 4.3 illustrates an example, in which both Activity A and Activity B connect to Job Step C bi-directionally through the association relationships. If Activity A is specified applicable, forward reasoning helps induce that Job Step C is applicable and backward reasoning follows to induce that Activity B is applicable as well. However, whether Activity B is applicable or not may not be a concern to engineers if they only focus on the other two concepts. Backward reasoning would make the situation more complicated when Potential Hazard concepts are connected to many Job Step concepts, and Job Step concepts to many Activity concepts.
Therefore, a control mechanism that allows only forward or backward reasoning or both was developed specifically for JHA concepts in the concept ontology reasoning engine and implemented through the prototype system. In this research, due to the nesting structure of JHA documents as discussed in section 3.4.4, I focus on reasoning about concepts from “Activity” to “Job Step” to “Potential Hazard”, i.e. only forward reasoning is allowed in the developed reasoning engine. This reasoning strategy is adopted in the validation of the developed concept ontology reasoning engine in section 5.3.2.

While backward reasoning is useful in some circumstances for providing a different reasoning aspect, such as knowing what activities should be applicable when a job step which is contained in those activities is known applicable, it is out of the scope of the validation and is recommended to be further studied in the future research.

4.3 SAFETY REQUIREMENT REASONING ENGINE

The goal of the safety requirement reasoning engine is to evaluate the applicability of safety requirements and to classify them based on their applicability. The safety requirement reasoning engine
uses the concepts of the concept ontology and relies on the availability of applicability values for the concepts, as determined by the concept ontology reasoning engine. The concepts are used to define applicability conditions and applicability exceptions of safety requirements and thus the concepts’ applicability has to be interpreted to determine whether those conditions and exceptions are satisfied.

Furthermore, the reasoning mechanism should be capable of addressing the complexity of the conditions that combine different concepts. Since a safety requirement’s condition is represented through a single context and that context may be described through multiple concepts, the applicability of this safety requirement cannot be determined until the reasoning process considers the applicability of all concepts and their combination. Additionally, the reasoning mechanism should be able to extract the related safety rules and group them according to their applicability, i.e. safety rules of the same applicability shall be organized in the same group.

4.3.1 Applicability Conditions and Exceptions and Their Representation

A safety requirement’s applicability condition is the condition describing the circumstances under which the safety requirement applies. Similarly, a safety requirement’s applicability exception is the condition describing the particular circumstances under which the safety requirement does not apply even when the applicability conditions apply. In this research, each safety requirement, except for general safety requirements, must have an applicability condition in order to specify the conditions in which the requirement applies while the definition of an applicability exception is not necessary. General safety requirements are always applicable no matter what the given contexts are; therefore, there is no need to
specify applicability conditions for them and in this research, the developed prototype system will automatically evaluate safety requirements without applicability conditions to be applicable.

The safety requirement reasoning engine reasons about applicability conditions and applicability exceptions of safety requirements to determine whether a requirement applies or does not apply. Applicability conditions and exceptions connect with the concept ontology since concepts from the ontology are used to represent and define a context in the applicability conditions or exceptions. In other words, the applicability conditions and applicability exceptions are a bridge to be traversed to determine safety requirements’ applicability based on applicability values determined for the concepts in the concept ontology. Therefore, it is necessary to understand how to represent applicability conditions and exceptions before the reasoning engine is discussed.

Applicability conditions and applicability exceptions of safety requirements are represented by the context, and this context is described through concepts. In order to properly represent them in the XML documents to support the latter reasoning process, tags that are used to identify contexts within the XML document need to be defined. There are three types of tags defined in this research for this purpose:

- **Concept Tag** (<Concept>…</Concept>): the tag is used to surround a concept, such as <Concept>Frame_Columns</Concept>.

- **Concept Conjunction Tag** (<Concept_AND>…</Concept_AND>): the tag is used to surround a set of concept tags (i.e. <Concept/>), or concept disjunction tags (i.e. <Concept_OR/>), or both. It represents the conjunction of concepts and concept disjunctions. For example, the context “frame and pour columns” (represented through the concept conjunction “Frame_Columns AND Pour
Columns”) is represented in a safety requirement XML file as:

```
<Concept_AND>
  <Concept>Frame_Columns</Concept>
  <Concept>Pour_Columns</Concept>
</Concept_AND>
```

- **Concept Disjunction Tag** (<Concept_OR>…</Concept_OR>): the tag is used to surround a set of concept tags (i.e. <Concept/>), concept conjunction tags (i.e. <Concept_AND/>), or both. It represents the disjunction of concepts and concept conjunctions. For example, the context “slip or trip” (represented through the concept disjunction “Slip OR Trip”) is represented in a safety requirement XML file as:

```
<Concept_OR>
  <Concept>Slip</Concept>
  <Concept>Trip</Concept>
</Concept_OR>
```

Moreover, applicability condition and applicability exception tags are defined and leverage the previous three tags for concept, concept conjunction and concept disjunction:

- **Applicability Condition Tag** (<Ctxt_App_Condition>…</Ctxt_App_Condition>): the tag is used to surround either a concept tag (i.e. <Concept/>), a concept conjunction tag (i.e. <Concept_AND/>), or a concept disjunction tag (i.e. <Concept_OR/>), all of which are used to define the applicability condition. For example, if a safety requirement’s applicability condition is “frame and pour columns”, this applicability condition is represented as:

```
<Ctxt_App_Condition>
  <Concept_AND>
    <Concept>Frame_Columns</Concept>
    <Concept>Pour_Columns</Concept>
  </Concept_AND>
</Ctxt_App_Condition>
```

- **Applicability Exception Tag** (<Ctxt_App_Exception>…</Ctxt_App_Exception>): the tag has the same modeling rule as the Applicability Condition Tag. For example, if a safety requirement’s
applicability exception is “slip or trip”, this applicability condition is represented as:

```
<Ctxt_App_Exception>
  <Concept_OR>
    <Concept>Slip</Concept>
    <Concept>Trip</Concept>
  </Concept_OR>
</Ctxt_App_Exception>
```

These five tags are summarized in Table 3.1 in section 3.4.3 with other tags defined in the XML schema. In the next section, I will present the reasoning rules that guide the reasoning process of the safety requirement reasoning engine.

4.3.2 Safety Requirement Reasoning Rules

The reasoning rules of the safety requirement reasoning engine are based on applicability conditions and applicability exceptions that are represented in the form of XML tags and based on the concepts of the concept ontology that have been reasoned about through the concept ontology reasoning engine. In this research, six reasoning rules are defined. The first four rules aim at evaluating the satisfaction value of applicability conditions and exceptions. The three satisfaction values: satisfied, possibly satisfied and not satisfied, are defined for applicability conditions and applicability exceptions to specify whether the condition or exception is satisfied or not. The remaining two rules aim at using the evaluation results to determine the safety requirements’ applicability.

4.3.2.1 Rules for evaluating applicability conditions and applicability exceptions

Every safety requirement has exactly one applicability condition and possibly one applicability
exception, which have to be evaluated according to the rules prescribed in this section. To facilitate the understanding of the rules, a term “evaluation unit” is adopted in the rules. An evaluation unit is defined as a part of an applicability condition or exception that is to be evaluated in order to determine whether the condition or exception is satisfied or not. The following are the definitions of evaluation units:

- An evaluation unit can be a single concept, a concept conjunction, or a concept disjunction. In case of a concept conjunction or disjunction, the evaluation unit has sub-evaluation units.

- In order to evaluate an evaluation unit, its sub-evaluation units should be evaluated first.

- An evaluation unit has exactly one of the three applicability values: *applicable*, *possibly applicable*, and *not applicable*.

- An applicability condition or exception can have only one **root evaluation unit**, meaning, while it may have sub-evaluation units, it is the outermost evaluation unit in an applicability condition or exception.

Figure 4.4 shows an example applicability condition for a JHA document, in which the root evaluation unit consists of two sub-evaluation units; one of the sub-evaluation units further contains two other sub-evaluation units.

![Figure 4.4: An example applicability condition demonstrating root and sub evaluation units](image-url)
Based on the definition of evaluation units, rule 1 defines the relations between applicability conditions and exceptions and their evaluation unit:

**Rule 1** Applicability Condition and Exception Evaluation Rule

The satisfaction values of applicability conditions and exceptions are determined by the applicability values of their root evaluation units. If the root evaluation unit is *applicable, possibly applicable or not applicable*, the conditions and exceptions will be *satisfied, possibly satisfied or not satisfied*, respectively.

To determine the applicability value of a condition or exception’s root evaluation unit, the following rules are applied to evaluate the root evaluation unit and its sub-evaluation units:

**Rule 2** Concept Evaluation Rule

The applicability value of an evaluation unit that is a single concept must be the same as the applicability value of the single concept.

**Rule 3** Concept Conjunction Evaluation Rule

The applicability value of an evaluation unit that is a concept conjunction can be determined as follows:

3a. If at least one of the sub-evaluation units of the evaluation unit is found *not applicable*, the evaluation unit must be *not applicable*.

3b. If at least one of the sub-evaluation units is *possibly applicable* and all the others are *applicable*, the evaluation unit must be *possibly applicable*.

3c. Otherwise, if all the sub-evaluation units are *applicable*, the evaluation unit must be *applicable*. 
**Rule 4** Concept Disjunction Evaluation Rule

The applicability value of an evaluation unit that is a concept disjunction can be determined as follows:

4a. If at least one of the sub-evaluation units in the evaluation unit of the applicability condition or exception is found *applicable*, the evaluation unit must be *applicable*.

4b. If at least one of the sub-evaluation units is *possibly applicable* and all the others are *not applicable*, the evaluation unit must be *possibly applicable*.

4c. Otherwise, if all the sub-evaluation units are *not applicable*, the evaluation unit must be *not applicable*.

4.3.2.2 Rules for determining safety requirements’ applicability

After a safety requirement’s applicability condition and exception are evaluated using the aforementioned reasoning rules, the applicability of this safety requirement can be evaluated using the following rules.

**Rule 5**

If a safety requirement has only an applicability condition, the safety requirement is *applicable*, *not applicable*, or *possibly applicable* only when the applicability condition is *satisfied*, *not satisfied*, or *possibly satisfied*, respectively.

**Rule 6**

If a safety requirement has both an applicability condition and an applicability exception:

6a. The safety requirement is *applicable* only when the applicability condition is *satisfied* and the
applicability exception is *not satisfied*.

6b. The safety requirement is *not applicable* only when either the applicability condition is *not satisfied* or the applicability exception is *satisfied*.

6c. Otherwise, the safety requirement is *possibly applicable*.

Rule 6 is summarized in Table 4.1 that lists all the possible combinations of the satisfaction evaluation values of the applicability condition (column 1) and applicability exception (column 2) as well as the corresponding applicability value of the safety requirement (column 3).

<table>
<thead>
<tr>
<th>IF</th>
<th>THEN</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Applicability Condition is</strong> &amp; <strong>Applicability Exception is</strong></td>
<td><strong>Safety Requirement is</strong></td>
</tr>
<tr>
<td>Satisfied &amp; Not Satisfied</td>
<td>Applicable</td>
</tr>
<tr>
<td>Satisfied &amp; Possibly Satisfied</td>
<td>Possibly Applicable</td>
</tr>
<tr>
<td>Satisfied &amp; Satisfied</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Possibly Satisfied &amp; Not Satisfied</td>
<td>Possibly Applicable</td>
</tr>
<tr>
<td>Possibly Satisfied &amp; Possibly Satisfied</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Possibly Satisfied &amp; Satisfied</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Not Satisfied &amp; Not Satisfied</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Not Satisfied &amp; Possibly Satisfied</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Not Satisfied &amp; Satisfied</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>

4.3.3 Summary of Safety Requirement Reasoning Engine

The safety requirement reasoning engine uses the concepts of the concept ontology and their applicability values to reason about the safety requirements. A safety requirement’s applicability depends on whether the safety requirement’s applicability condition and, if any, applicability exception are satisfied or not. For example, a safety requirement is found *applicable* only when its applicability condition is *satisfied* and the applicability exception is either absent or *not satisfied*.
In this research, six reasoning rules are defined for the safety requirement reasoning engine. The safety requirement reasoning process starts with evaluating applicability condition and applicability exceptions by processing the rules 1 to 4. Then the reasoning process applies the rules 5 and 6 to determine the applicability of safety requirements based on the satisfaction values of the applicability conditions and exceptions. For example, if the applicability condition and applicability exception of a safety requirement are both satisfied, the safety requirement is not applicable according to Table 4.1. Once each safety requirement’s applicability is obtained, the safety requirements can be classified according to their applicability, and the applicable safety requirements that apply to the project’s contexts become available.

4.4 FEEDBACK OF REASONING RATIONALE

Kiliccote (1996) pointed out that for a reasoning system, mechanisms for reporting explanations of the reasoning results should be provided to system users to improve their understanding of the reasoning process and to benefit their decision-making. Therefore, both reasoning mechanisms presented in this chapter should be capable of providing feedback of reasoning results to users. This functionality enables users to understand the rationale of the reasoning processes, i.e. how the applicability of concepts and safety requirements are determined. According to the reasoning rationale, users can evaluate whether the reasoning results are justifiable and whether the reasoning mechanisms need to be revised. In this research, the functionality of reasoning rationale feedback is implemented for the developed Reasoning Engine in an automated manner through developing a prototype system and incorporating this functionality into the system, which is discussed in the following chapter.
4.5 SUMMARY OF THE REASONING ENGINE

In this chapter, I discuss the Reasoning Engine in detail, which is the second part of the developed Construction Safety Document Management Framework. The Reasoning Engine consists of two sub-engines: the concept ontology reasoning engine and the safety requirement reasoning engine.

The concept ontology reasoning engine provides a reasoning mechanism that can be used to evaluate the applicability of concepts in a concept ontology given a set of concepts whose applicability values are known. I propose six concept ontology reasoning principles in this engine for reasoning about concepts. Based on the proposed reasoning principles, both forward and backward reasoning are allowed for reasoning about JHA concepts and forward reasoning is adopted in this research due to the nesting structure of JHA documents. I also discuss concept applicability contradiction, including what causes the contradiction and how to resolve it.

On the other hand, the safety requirement reasoning engine provides a reasoning mechanism that can be used to evaluate the applicability of safety requirements and to classify them according to their applicability values. I proposed six reasoning rules in this engine for reasoning about safety requirements and their applicability conditions and exceptions. I discuss in detail how to represent applicability conditions and exceptions in XML format and how to evaluate them based on the notion of evaluation units.

Lastly, I discuss the need to have a reasoning rationale feedback mechanism for both reasoning engines in order to improve users’ understanding of how reasoning is processed. This mechanism and the implementation and validation of the developed Framework will be discussed in the next chapter.
CHAPTER 5: IMPLEMENTATION AND VALIDATION

5.1 INTRODUCTION

In this chapter I describe the tests I performed to validate the Construction Safety Document Management Framework in this research. The motivation for the development of the proposed Construction Safety Document Management Framework is to enable automated reasoning about context-describing concepts and construction safety documents to identify applicable safety requirements imposed by the documents for the purpose of raising project participants’ awareness of these applicable safety requirements. Therefore, the validation strategy consists of four major parts:

1. Validation of the representation model for representing context-describing concepts in concept ontologies.
2. Validation of the reasoning mechanism implemented for reasoning about the represented concepts of the concept ontology to obtain their applicabilities.
3. Validation of the representation model for representing construction safety documents in XML format.
4. Validation of the reasoning mechanism implemented for identifying applicable safety requirements imposed by the represented construction safety documents.

The first two parts of the strategy are to validate the representation of and reasoning about concept ontology while the second two are to validate the representation of and reasoning about textual
In order to validate the Framework, I used Java programming language (JDK 1.6.0) to implement a prototype system that supports the use of the developed Representation Model and Reasoning Engine of the Framework. I deployed the prototype system within different test case scenarios, which were based on construction safety documents acquired from the industry. The prototype system was deployed to reason about the acquired documents and their context-describing concepts modeled for these test cases.

In the test cases, the construction safety documents I focused on in the research are JHA documents due to the significance of JHA discussed in chapter 2. There were 78 JHA documents acquired from a private construction company. These JHA documents covered job hazard analyses for different divisions of construction, such as concrete construction, excavation, electrical construction, masonry, mechanical construction, and plumbing. They together accounted for combined 678 concepts and 1121 recommended safety procedures/safety rules. The one shown in Figure 3.2 in section 3.4.2 is an example of one of the acquired JHA documents. In addition, according to safety professionals’ opinions, these documents can best represent the document writers’ safety knowledge related to the topics of these documents. In other words, it is reasonable to assume there is no missing safety knowledge related to these acquired documents’ topics and to assume the safety rules in these documents can be used as prior knowledge in the validations. The assumptions ensure that the validations of the developed safety requirement reasoning engine, discussed in section 5.4.2, can focus on these acquired JHA documents without the issues of incomplete safety knowledge related to these documents.
The following describes the validation steps. First, I will describe how the prototype system works and relates to other software tools in the research in order to demonstrate the validity of the developed Framework. Second, I will focus on the developed Framework to discuss the validation of the ability to represent context-describing concepts, followed by a discussion of the validation of the reasoning mechanism for reasoning about the represented concepts. Then, I will discuss the developed Framework’s validation of the ability to represent construction safety documents and then the validation of the reasoning mechanism for identifying applicable safety requirements. The validation of the concept ontology representation model and reasoning engine as well as the validation of the textual document representation model and reasoning engine will validate the overall Construction Safety Document Management Framework.

5.2 PROTOTYPE SYSTEM: JHA ADVISOR

5.2.1 JHA Advisor, Ontology Authoring Tools and Document Modeling Tools

Based on the previous discussion on the Representation Model and Reasoning Engine of the Construction Safety Document Management Framework, a prototype system, JHA Advisor, has been developed mainly to support the use of the developed Framework. JHA Advisor is designed not only to perform the developed reasoning mechanisms but also to provide functions of adding, deleting and editing concepts and different JHA sections for the input concept ontology (OWL file) and the JHA documents (XML file) respectively. However, JHA Advisor is not designed to generate the initial input information required by itself, i.e. the initial concept ontology for organizing concepts and relationships in OWL format and JHA documents for storing JHA information in XML format; therefore, the initial input information
need to be prepared through other ontology authoring tools and document modeling tools independent of the JHA Advisor. The ontology authoring tool adopted in this research, as discussed in section 3.3.3, is Protégé 3.3 beta. The document modeling tool can be just a basic text editor, such as Windows Notepad, or sophisticated software with integrated functionality of defining XML schema and editing XML documents. Figure 5.1 illustrates the interaction among JHA Advisor, ontology authoring tools and document modeling tools.

- **Interaction A**: JHA Advisor requires a concept ontology, which is initially prepared through ontology authoring tools, to perform the concept ontology reasoning engine. Later update on the concept ontology can be done either through the ontology authoring tools or through JHA Advisor since JHA Advisor provides the functionality for editing the concepts of the concept ontology.

- **Interaction B**: The JHA documents required by JHA Advisor are initially modeled into XML format through document modeling tools. If the modeled JHA documents need to be edited or are considered inappropriate, such as applicability condition is wrongly defined, one can either go back to the document modeling tool or use the editor provided by JHA Advisor to add, delete or revise different JHA sections and safety requirements.

Figure 5.1: Interactions among the JHA Advisor, ontology authoring tools and document modeling tools
• Interaction C: The concepts organized in the concept ontology are retrieved from the JHA documents while the applicability condition and applicability exception of the documents are described through the concepts stored in the concept ontology.

5.2.2 Use of JHA Advisor

Figure 5.2 shows the user-interface of JHA Advisor. On the left is the working area for the concept ontology in which three separate panes respectively show the ontology hierarchies of the three primary grouping concepts “Activity”, “Job Step” and “Potential Hazard”; on the right is the working area for JHA documents.

Once the concept ontology and JHA documents are prepared in OWL and XML in accordance with the instructions of chapter 3 through ontology authoring tools and document modeling tools respectively, the following summary steps can be adopted to run JHA Advisor (the detailed steps are illustrated in Appendix A):

(1) Load the concept ontology into JHA Advisor
(2) Load the modeled JHA documents into JHA Advisor
(3) Specify applicability values for concepts
(4) Propagate applicabilities among the concepts as well as the safety requirements
(5) Output the reasoning results as an MS Excel file¹ through JHA Advisor

¹ The MS Excel format is used because it is the most prevalent format adopted to organize JHA knowledge by the safety engineers of the cooperative construction company.
After the reasoning results are output, users can obtain lists of safety requirements, which are categorized according to their applicabilities, together with their related concept information (i.e. the Activity, Job Step and Potential Hazard concepts describing the contexts to which the safety requirements apply).

5.3 REPRESENTATION OF AND REASONING ABOUT JHA CONCEPTS

I used the developed concept ontology representation model to represent all the 678 concepts extracted from the acquired JHA documents in a concept ontology. I tested whether the model is capable of representing these concepts that had different semantic properties. I also tested how the model can be specialized by introducing new, intermediate classifications to classify the concepts. In addition, I used the developed concept ontology reasoning engine to reasoning about the 678 concepts represented in the concept ontology and to test whether the applicability values were correctly propagated and whether
applicable and/or not applicable concepts were properly identified.

5.3.1 Representation of Concepts

Of the extracted 678 JHA concepts, 78 were Activity concepts, 245 Job Step concepts, and 355 Potential Hazard concepts. When processing and arranging these extracted concepts, I found that the three types of primary grouping concepts, i.e. Activity, Job Step and Potential Hazard, have different semantic properties:

- Activity concepts usually are specified through a single action concept together with, if any, resource concepts. Examples are “Pour Columns” and “Welding Operation”. In addition, Activity concepts usually are division-oriented: they can be easily identified what construction divisions they belong to according to their titles. For example, Activity concepts “Pour Columns” and “Welding Operation” can be easily categorized to concrete division and to mechanical division respectively.

- Job Step concepts are specified through single or multiple action concepts usually together with single or multiple resource concepts. Examples are “Turn off Power”, “Drill and Secure Anchors and Hangers” and “Install Piping Overhead in Lateral Racks”.

- Potential Hazard concepts usually are unconditionally or conditionally specified through general or specific unsafe behavior/environments, injuries and illnesses. For example, a Potential Hazard concept “Injury to Personnel due to Hydraulic Jack Crushing” is a concept specified by a general injury on a condition while another Potential Hazard concept “Dermatitis” is one specified by a specific illness without any condition.
Therefore, when the extracted concepts were represented in class hierarchies, different considerations of representation should be taken into account in order to address the abovementioned differences of these concepts’ semantic properties.

5.3.1.1 Strategies for concept representation in hierarchies

The following strategies for representing the three types of JHA concepts were developed and adopted to take into account the aforementioned differences of the semantic properties when I represented the extracted concepts:

(1) Strategy for representing Activity concepts:

Instead of representing all of the Activity concepts directly under the primary grouping concept “Activity”, I first represented them in the concept ontology using the classification of MasterFormat (Construction Specifications Institute and Construction Specifications Canada 2005). MasterFormat provides a detailed classification system for classifying construction specifications in accordance with the divisions into which the specifications can be categorized. Thus, MasterFormat is well-suited for Activity concepts to represent their division-specific semantic properties and its classification provided the secondary grouping concepts for representing the extracted Activity concepts under the primary grouping concept “Activity”. These secondary grouping concepts were listed in the second column of Table B.1 in Appendix B.
(2) Strategy for representing Job Step concept:

I found that Job Step concepts can be most straightforwardly grouped based on the action concepts specified in these Job Step concepts’ titles since each Job Step concept must carry at least one action concept. Therefore, I created a set of secondary grouping concepts, which comprised those action concepts occurring in the extracted Job Step concepts, under the primary grouping concept “Job Step”. I then used these secondary grouping concepts to represent the extracted Job Step concepts. For those Job Step concepts comprising multiple action concepts, only one action concept which was the most representative of the Job Step concept was selected and incorporated in the set of secondary grouping concepts. These secondary grouping concepts were listed in the second column of Table B.2 in Appendix B.

(3) Strategy for representing Potential Hazard concept:

Not only are Potential Hazard concepts more diversified than the other two types of concepts (as aforementioned, they usually are specified unconditionally or conditionally, and generally or specifically), but the number of them usually is also much larger than that of the other two. Therefore, a well-organized classification is especially necessary for the representation of and reasoning about Potential Hazard concepts, which was discussed in detail in section 3.3.1. In this research, the classification for represent the extracted Potential Hazard concepts was adapted from the Occupational Injury and Illness Classification Manual (OI&ICM) (U.S. Department of Labor 1992) in which potential hazards are classified, based on their exposure, into “Contact With Objects and Equipment”, “Falls”, “Bodily Reaction and Exertion”, “Exposure to Harmful Substances or Environments”, “Transportation Accidents”, “Fires and Explosions”, “Other Events or Exposures”, “Assaults and Violent Acts” and “Non-classifiable”. These
classes and their subclasses, which may also have subclasses to further classify concepts, provided a detailed structure that is well-suited for representing diversified and considerable Potential Hazard concepts in this research and therefore, I adopted the first seven classes as the secondary grouping concepts and their subclasses as the tertiary grouping concepts to represent the extracted Potential Hazard concepts. These secondary and tertiary grouping concepts were respectively listed in the second and third columns of Table B.3 in Appendix B.

Those grouping concepts (secondary, tertiary or further) mentioned in these strategies together with the three primary grouping concepts (Activity, Job Step and Potential Hazard) are called fundamental grouping concepts in this research as they formed the backbone of the representation model in the test cases.

In addition to these strategies for the three types of JHA concepts, another strategy was also considered when there was a need to add new classes which do not belong to the aforementioned classifications (that is, not exist as any of the secondary, tertiary grouping concepts and so forth) in the representation of concepts. This strategy applies to all JHA concepts independent of their types.

(4) Strategy for adding new classes

Any new class which does not exist in the aforementioned classifications and can generalize the extracted concepts was first defined and added in the concept ontology as a subclass of the secondary or tertiary grouping concepts. Then, those extracted concepts which can be generalized by the new class were assigned to the new class as its subclasses.
The concepts defined according to this strategy and used to group other concepts were called
*extended grouping concepts* as opposed to fundamental grouping concepts. Moreover, another
circumstance in which the notion of extended grouping concepts applies without defining new classes is
when one of the extracted concepts was used to classify one or more of other extracted concepts. In this
regard, the extracted concept classifying others was viewed as an extended grouping concept as well.

According to the fourth strategy, a new class acted as an extended tertiary grouping concept if it was
added under the fundamental secondary grouping concepts (for all the cases of representing Activity and
Job Step concepts¹, and part of the cases of representing Potential Hazard concepts²) and as a extended
quaternary grouping concept if under the fundamental tertiary grouping concepts and so forth (only in the
cases of representing Potential Hazard concepts). In addition, when defining new classes according to the
fourth strategy in the representation of the extracted concepts, I also took into account the proposed
heuristic steps of integrating new concepts into existing classifications discussed in 3.3.4.

5.3.1.2 Validating cases of concept representation in hierarchies

The following are examples of concept representation I implemented to validate the concept
representation model and the reasoning engine later. These examples are categorized by either the
fundamental or extended grouping concepts being used to represent the 678 extracted concepts.

---
¹ The representations of Activity and Job Step concepts have only fundamental secondary grouping concepts
according to the first and second strategies.
² Two of the six fundamental secondary grouping concepts for Potential Hazard concepts do not have subclasses as
the fundamental tertiary grouping concepts in the case studies (i.e. “Hazard From Other Events Or Exposure” and
“Hazard From Transportation Accidents”).
**Category A: Concept representation using fundamental grouping concepts.** In this category, extracted concepts were represented as subclasses of the fundamental grouping concepts, which can be secondary and tertiary and further grouping concepts.

*A1: Add extracted concepts directly in fundamental secondary grouping concepts.* In this sub-category, extracted concepts were represented as subclasses of the fundamental secondary grouping concepts.

**Example 1:** The extracted concepts “Frame Column” and “Forklift Use” were represented as subclasses of the secondary grouping concept “Concrete Activity” and “Equipment Activity”, both of which were subclasses of the primary grouping concept “Activity” (as shown in test case no. 1 and 4 of Table B.1 in Appendix B).

**Example 2:** The extracted concept “Inspect Chains Straps And Hooks For Deficiencies” was represented as a subclass of the secondary grouping concept “Inspect Step”, which was a subclass of the primary grouping concept “Job Step” (as shown in test case no. 17 of Table B.2 in Appendix B).

*A2: Add extracted concepts directly in fundamental tertiary or further grouping concepts.* In this sub-category, extracted concepts were represented as subclasses of the fundamental tertiary or further grouping concepts.

**Example 3:** The extracted concept “Strain” was represented as a subclass of the tertiary grouping concept “Bodily Reaction”, which was a subclass of the secondary grouping concept “Hazard From Bodily Reaction And Exertion” that was a subclass of the primary grouping concept “Potential Hazard” (as shown in test case no. 1 of Table B.3 in Appendix B).

**Category B: Concept representation using newly defined classes as extended grouping concepts.** In this
category, new concepts were first defined as extended grouping concepts, which can be tertiary or further grouping concepts. Then extracted concepts were represented as subclasses of the defined classes.

**B1: Add extracted concepts in extended tertiary grouping concepts formed with new concepts.** In this sub-category, new concepts were first defined as extended tertiary grouping concepts and then extracted concepts were represented as subclasses of the new concepts.

**Example 4:** The extracted concept “Damage To Equipment” was represented as a subclasses of the concept “Equipment Event Or Exposure”, which is one of nine extended tertiary grouping concepts specifically defined for the fundamental secondary grouping concept “Hazard From Other Events Or Exposure” under the “Potential Hazard” concept (as shown in test case no. 22 of Table B.3 in Appendix B).

**B2: Add extracted concepts in extended quaternary or further grouping concepts formed with new concepts.** In this sub-category, new concepts were first defined as extended quaternary or further grouping concepts; then extracted concepts were represented as subclasses of the new concepts.

**Example 5:** A new extended quaternary grouping concept “Rays” was defined under the class hierarchy “Exposure To Radiation”, “Hazard From Exposure To Harmful Substances Or Environments” and “Potential Hazard”, in turn the fundamental tertiary, secondary and primary grouping concepts. It then represented the extracted concepts “Heat Rays” and “UV Rays” as its subclasses.

**Category C: Concept representation using extract concepts as extended grouping concepts.** In this category, extracted concepts were used as superclasses to represent other extracted concepts.

**C1: Add extracted concepts in extended tertiary grouping concepts formed with other extracted concepts.**
In this sub-category, some extracted concepts acted as extended tertiary grouping concepts, and other extracted concepts were represented as subclasses of them.

**Example 7:** The extracted concept “Turn Off Power At Circuit Breaker” was represented as a subclass of another extracted concept “Turn Off Power”, which acted as an extended tertiary grouping concept. The concept “Turn Off Power” was a subclass of the fundamental secondary grouping concept “Turn Off Step” that is a subclass of the primary grouping concept “Job Step”.

**C2: Add extracted concepts in extended quaternary or further grouping concepts formed by other extracted concepts.** In this sub-category, some extracted concepts acted as extended quaternary or further grouping concepts, and other extracted concepts were represented as subclasses of them.

**Example 8:** The extracted concept “Welding of Tube With Orbital Welder” was represented as a subclass of another extracted concept “Welding of Tube”, which acted as an extended quaternary grouping concept and had the concepts “Welding”, “Weld Step” and “Job Step” in turn upwardly in the hierarchy.

In addition to these examples, I also conducted representation cases for all other extracted concepts, part of which were listed in Table B.1, Table B.2 and Table B.3 in Appendix B. These tables showed the test cases of concept representation for every fundamental primary and secondary grouping concept.

5.3.1.3 Relationships of concept representation

Another essential work in concept representation is to represent relations between concepts through association and logical (equivalent and disjoint) relationships. In the test cases, I used both *JHA Advisor*...
and Protégé to establish such relations between the extracted concepts. In addition, the following representation considerations were taken into account in the test cases:

- For establishing relations through association relationships, while the relations were established bi-directionally, e.g. from an Activity concept to a Job Step concept through “hasStep” and the other way around through “isStepOf”, I focused on the relations only from Activity concepts to Job Step concepts and from Job Step concepts to Potential Hazard concepts due to the reasoning consideration in the following discussion in the test cases (detail is explained in section 3.3.2).

- Among the extracted concepts, disjoint relationships were only deployed to some Activity and Potential Hazard concepts. That is, there were no disjoint relationships deployed between the extracted Job Step concepts. There were 18 sets of multiple disjoint Potential Hazard concepts and one set of two disjoint Activity concepts in the test cases. For example, the extracted concepts “Fall From Rolling Scaffold” and “Fall From Suspended Scaffold” were declared disjoint and comprised one of the 18 sets of Potential Hazard concepts.

- Equivalent relationships were deployed in two sets of two Potential Hazard concepts, which were the set of “Slip” and “Trip”, and the set of “Sprain” and “Strain”.

According to these considerations and the requirements of representing relationships discussed in 3.3.2, I established all the required relations between the extracted concepts in the test cases through association and logical relationships. Part of the relationship representations were listed in Table B.1, Table B.2 and Table B.3 in Appendix B. For example, in the test case no. 2 in Table B.1Table B.1, the extracted concept “Excavation Using Support Systems” had its associated Job Step concept and disjoint concept defined.
It is noteworthy that Table B.1 and Table B.2 did not show equivalent concepts because there were no equivalent relationships deployed for the extracted Activity and Job Step concepts. In addition, Table B.2 did not show disjoint concepts because no disjoint relationships were deployed between the extracted Job Step concepts.

5.3.1.4 Summary of validations of concept ontology representation model

The developed concept representation model allowed me to model all the extracted JHA concepts according to the abovementioned strategies and representation categories. It also allowed me to establish necessary relations between the extracted concepts through the defined association relationships as well as logical relationships. Therefore, the concept representation model can be considered validated in its ability to represent the concepts of JHA documents.

The concept ontology representation model has a limitation: when the number of concepts becomes large, modeling these concepts in a concept ontology will become tedious since the modeling processes need to be proceeded manually. This limitation may be addressed through providing tools which allow a more straightforward process of modeling concepts, which can be a potential research direction in the future.

Another aspect related to the concept representation model that requires validation is the reasoning about the modeled concepts. The required reasoning mechanism aims to automatically evaluate concepts’ applicabilities to a given search contextual conditions. The validation of the reasoning mechanism
provided by the developed concept ontology reasoning engine is discussed in the following subsection.

5.3.2 Reasoning about Concepts

I validated the reasoning engine developed for reasoning about concepts and for evaluating concepts’ applicability to given contexts by conducting several synthetic test cases. In these test cases, I specified synthetic contexts to test whether the reasoning principles developed in 4.2.2 in the research can function correctly, namely, properly propagating applicability value between concepts as well as identifying concepts’ applicability to the specified synthetic contexts. Moreover, the concept reasoning engine was implemented in JHA Advisor so the test cases conducted here were also used to evaluate whether JHA Advisor functioned properly to implement the reasoning process.

Due to the huge number of concepts represented in the concept ontology, the following subsections only showed representative test cases on those extracted concepts which were listed in the Table B.1, Table B.2 and Table B.3 in Appendix B and had the required relationships defined for validating the developed reasoning principles. For example, to validate the fifth reasoning principle, Disjoint Association Propagation Principle, those concepts with relations defined through disjoint relationships were included in the synthetic tests.

5.3.2.1 Cases of validating concept ontology reasoning engine

In this subsection, I discussed three test cases, which were part of the tests I conducted to validate the
developed concept ontology reasoning engine and were representative of demonstrating the reasoning engine’s capability. The results of these test cases were summarized in Table C.1 in Appendix C.

(1) Case 1:

In this case, the following context was specified:

“A construction project is planned to use support systems during the upcoming excavation activity to prevent from cave-in.”

The concept “Excavation Using Support Systems” of the concept ontology can best describe the given context. Therefore, I assigned this concept the applicability value applicable. After that, the reasoning engine successfully evaluated the concept’s fundamental primary and secondary grouping concepts, “Activity” and “Excavation Activity” respectively, and its associated Job Step concept, “Excavation Using Shoring Manufactured Trench Boxes or Other Support Systems”, to be applicable. The reasoning engine also successfully evaluated the eleven Potential Hazard concepts associated with the abovementioned Job Step concept to be applicable as well. In addition, the engine evaluated this concept’s disjoint concept, “Excavation Using Sloping”, to be not applicable.

The result showed that the second, third and fifth concept ontology reasoning principles worked properly to propagate the applicability value from the concept “Excavation Using Support Systems” to its super-concepts and associated concepts as well. The result was summarized as the first test case in Table C.1 in Appendix C.
(2) Case 2:

In this case, the following context was specified:

“There is no HVAC activity but plumbing activity scheduled for today.”

According to this description, the concepts best describing the context were two Activity concepts: “HVAC Activity” and “Plumbing Activity”. Therefore, the applicability value not applicable was assigned to the former concept and the value applicable to the latter concept. Then, the reasoning engine successfully evaluated all the three sub-concepts of “HVAC Activity” to be not applicable, which are “Disassembly of HEPA Filter Bank Interior Ductwork and Fire Sprinkler System”, “Fibrous Insulation and Refractory Ceramic Fiber”, and “Removal of Exterior Duct Work”. Also, the reasoning engine successfully evaluated the fundamental primary grouping concept “Activity” of the concept “Plumbing Activity” as applicable.

Because both concepts “HVAC Activity” and “Plumbing Activity” were fundamental secondary grouping concepts and only used for classification, there were no association relationships between them and other Job Step concepts. The result showed that the second concept reasoning principle properly worked to propagate both the applicability values applicable and not applicable in class hierarchies. The result was summarized as the second test cast in Table C.1 in Appendix C.

(3) Case 3:

In this case, the following context was specified:

“Some accidents, including workers’ sprain, slip and fall due to poly moving, and being struck by a truck, happened yesterday on the construction site of this project.”
According to this description, the concepts best describing the context were four Potential Hazard concepts: “Sprain”, “Slip due to Poly Moving”, “Fall due to Poly Moving” and “Struck By”. Hence, the applicability value applicable was assigned to all of these concepts. After that, the reasoning engine successfully evaluated these concepts’ fundamental and extended grouping concepts to be applicable. For example, the grouping concepts of the concept “Slip Due to Poly Moving”: “Potential Hazard”, “Hazard from Bodily Reaction and Exertion”, “Bodily Reaction”, and “Slip”, were found applicable after the reasoning process. Moreover, the equivalent concept of the concept “Slip”, “Trip”, was then evaluated to be applicable. Similarly, the equivalent concept of the concept “Sprain”, “Strain”, was also successfully evaluated to be applicable.

Because Potential Hazard concepts were the last stop in forward reasoning process, there were no concepts associated with these four context-describing concepts to be evaluated. The test case result showed that the second and third concept reasoning principles worked properly to propagate the applicability values applicable in class hierarchies and to equivalent concepts. The result was summarized as the third test cast in Table C.1 in Appendix C.

5.3.2.2 Summary of and discussion on validations of concept ontology reasoning engine

The test cases discussed in the last subsection demonstrated that the developed concept ontology reasoning engine and prototype system JHA Advisor were capable of correctly evaluating concepts’ applicability in the given testing scenarios by propagating applicability values between concepts according to the proposed reasoning principles. In addition to those concepts describing the synthetic
contexts, I also tested other concepts modeled in the concept ontology and found that expected reasoning results were also obtained once the necessary association, disjoint and equivalent relationships were properly defined between the concepts. Therefore, the concept ontology reasoning engine can be considered validated in its ability to correctly reason about JHA concepts which are modeled in an ontology.

The validations of the concept ontology reasoning engine are conducted in given synthetic testing scenarios. However, I do not consider the computational complexity, decidability and completeness of the concept ontology reasoning engine in the validations. These issues will become critical when a reasoning process involves many applicability value assignments of concepts and especially results in many concept applicability contradictions. Therefore, there has a need to further study these issues in future research in order to prove the concept ontology reasoning engine’s applicability in large-scale reasoning scenarios.

To make reasoning about concepts manageable, I only allow single-inheritance in the concept ontology, namely a concept can have only one super-concept. Removing this limitation and allowing multiple-inheritance in the concept ontology reasoning engine are worth further studying in the future research.

In this research, the importance of successful representation of and reasoning about concepts lies in their assistance in the automated identification of construction safety requirements which are applicable to given contexts. Hence, the validation of the textual document representation model and safety requirement reasoning engine developed for automated identification of safety requirements is built upon
the validated concept ontology representation model and reasoning engine. In the following section, I first
discuss the validation conducted to prove the textual document representation model’s capability of
modeling JHA documents. Then I discuss the validation of the safety requirement reasoning engine’s
capability of reasoning about the modeled JHA documents, evaluating safety requirements’ applicability
and classifying the requirements according to their applicability.

5.4 REPRESENTATION OF AND REASONING ABOUT JHA DOCUMENTS

I used the developed textual document representation model to represent the acquired JHA
documents including 678 JHA concepts together with corresponding 1121 JHA safety rules in XML
format. I tested whether the model is capable of modeling the JHA concepts, safety rules, and applicability
conditions which were described through the JHA concepts in different representation situations. In
addition, I used the developed safety requirement reasoning engine to reason about the represented JHA
documents for testing whether the applicability values were correctly propagated from JHA concepts,
which were used to describe applicability conditions and were already modeled and validated as discussed
in section 5.3, to JHA documents and whether applicable and not applicable safety requirements were
properly identified and classified.

5.4.1 Representation of JHA Documents

As presented in section 3.4.4, one of the characteristics of JHA documents is the nesting structure in
which JHA information is organized in the documents. That is, a JHA Activity section can be composed
of multiple Job Step sections; a Job Step section can be composed of multiple Potential Hazard sections; and, a Potential Hazard sections can have multiple safety rules. As a result, contexts for applicability conditions of a JHA document shall expand with the development of the nesting structure from the tag Activity to the tag PotentialHazard (i.e., from Activity sections to Potential Hazard sections inwardly) in the JHA document. Specifically, a context for the applicability condition of an Activity section can be sufficiently described through a single Activity concept (called single-type description) while that of a Potential Hazard section should be described through Activity, Job Step and Potential Hazard concepts jointly (called multi-type description). From the evaluation unit perspective, single-type description means that the root evaluation unit of the applicability condition is a single concept while multi-type description means that the root evaluation unit of the applicability condition consists of concepts of different types, each of which further acts as a sub-evaluation unit. Therefore, this context expansion characteristic must be taken into account when the developed textual document representation model is used to represent JHA documents.

In the following section, I first demonstrated how multi-type description of contexts for applicability conditions should be done and why the decision was made. Then, I discussed the validation of using the developed textual document representation model to represent the acquired JHA documents. I discussed the representation model’s capability of representing JHA documents through several examples of different representation situations.
5.4.1.1 Multi-type description for applicability conditions

Contexts for applicability conditions of JHA Job Step and Potential Hazard sections can be described with multi-type concepts in two ways: **conjunctive description** and **disjunctive description**. Conjunctive description of contexts means that the multi-type concepts describing contexts are jointed through logical AND while disjunctive description means that those concepts are jointed through logical OR. Figure 5.3(A) shows an XML example for conjunctive description for which the tag `<Concept_AND/>` is used while Figure 5.3(B) is an example for disjunctive description for which the tag `<Concept_OR/>` is used. In addition, conjunctive and disjunctive descriptions also mean two different kinds of semantic formation of root evaluation units.

![XML examples](image)

(A) Conjunctive description  
(B) Disjunctive description

Table D.1 in Appendix D showed the result of an analysis of the applicability differences between conjunctive and disjunctive descriptions in different reasoning scenarios. In the analysis, I analyzed the conjunctive and disjunctive descriptions of contexts for applicability conditions of JHA safety rules through Activity, Job Step, and Potential Hazard-type concepts. For the simplification of illustration, these multi-type concepts were viewed in the form of evaluation units: the Activity, Job Step and Potential Hazard units were respectively composed of the Activity, Job Step and Potential Hazard concepts. The
three units that act as sub-evaluation units together comprised the root evaluation unit. Then, the
evaluation of each scenario of the analysis was based on the conjunction or disjunction of the three
sub-evaluation units that have different applicability values according to the developed textual document
reasoning rules.

The result of the analysis showed that if conjunctive description was deployed, only one out of the 27
scenarios produced applicable root evaluation unit, which implied a satisfied applicability condition
containing the root evaluation unit and applicable JHA safety rules to which the applicability condition
belonged. On the other hand, if disjunctive description was deployed, only one scenario generated not
applicable root evaluation unit, implying a not satisfied applicability condition and not applicable JHA
safety rules. Table 5.1 summarized the number of root evaluation units of different applicability values in
both conjunctive and disjunctive descriptions.

Table 5.1: Number of root evaluation units of different applicabilities in both description types

<table>
<thead>
<tr>
<th>Applicability of Root Evaluation Unit</th>
<th>Description Type</th>
<th>Disjunctive Description</th>
<th>Conjunctive Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>applicable</td>
<td></td>
<td>19</td>
<td>1</td>
</tr>
<tr>
<td>not applicable</td>
<td></td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>possibly applicable</td>
<td></td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>27</td>
<td>27</td>
</tr>
</tbody>
</table>

The analysis result shows that conjunctive description has a stricter mechanism to identify applicable
JHA information than disjunctive description does (i.e., the threshold of reasoning out applicable JHA
information with conjunctive description is much higher than with disjunctive description). As a result,
the reasoning about JHA information is so strictly controlled that JHA information can more easily be
evaluated to be not applicable and excluded accordingly, given some sub-evaluation units which are known not applicable. On the other hand, disjunctive description allows JHA information to easily become applicable as long as any sub-evaluation unit belonging to the root evaluation unit is applicable; hence, unnecessary JHA information still can be evaluated as applicable if one inadvertently specifies applicable value for certain concepts, which makes disjunctive description unfavorable to the reasoning about JHA information.

Therefore, conjunctive description was chosen in the representation of and reasoning about JHA documents in this research. Throughout the validation, logical AND was used for root evaluation units, as shown in the example of Figure 5.3(A), for multi-type sub-evaluation units to describe contexts for applicability conditions of different JHA document sections.

5.4.1.2 Cases of representation of JHA documents

I tested the developed textual document representation model using the model to represent the acquired JHA documents. The following examples were part of the implementation for the document representation with which I demonstrated the model’s capability of representing documents in different representation scenarios.

Scenario A: Representation of Activity sections. To represent Activity sections of the JHA documents, Activity title was first represented, which was followed by the representation of applicability condition and Job Step sections. The applicability condition was described through single Activity concept, i.e.
single-type description as discussed above; the representation of Job Step sections nested in Activity
sections was discussed in the next scenario.

**Example 1:** For the JHA document targeting “Frame Columns” activity, its Activity section can be
represented as follows:

```xml
<Activity>
  <ActTitle>Frame Columns</ActTitle>
  <Ctxt_App_Condition>
    <Concept>Frame_Columns</Concept>
  </Ctxt_App_Condition>
  ......
</Activity>
```

**Example 2:** For the JHA document targeting “Purge Gas Operation” activity, its Activity section can be
represented as follows:

```xml
<Activity>
  <ActTitle>Purge Gas Operation</ActTitle>
  <Ctxt_App_Condition>
    <Concept>Purge_Gas_Operation</Concept>
  </Ctxt_App_Condition>
  ......
</Activity>
```

**Scenario B: Representation of Job Step sections.** Job Step title was first represented, which was followed
by the representation of applicability condition and Potential Hazard sections. The applicability condition
was described through both Activity and Job Step concepts, i.e. multi-type description; the representation
of Potential Hazard sections nested in Job Step sections was discussed in the next scenario. Two
sub-scenarios were discussed below.

**B1: Represent single Job Step concept.** A Job Step section mostly contained a single step of an activity. In
this regard, a Job Step concept was sufficient, together with Activity concept, to describe the context of
this section’s applicability condition.

**Example 3:** A Job Step section “Fly forms to area to be installed” of the Activity section “Frame Columns”
was represented as followed:

```xml
<JobStep>
  <StepTitle>Fly forms to area to be installed</StepTitle>
  <Ctxt_App_Condition>
    <Concept_AND>
      <Concept>Frame_Columns</Concept>
      <Concept>Fly_Forms_To_Area_To_Be_Installed</Concept>
    </Concept_AND>
  </Ctxt_App_Condition>
  ...... 
</JobStep>
```

**Example 4:** A Job Step section “Introduction and removal of purge to POC” of the Activity section “Purge Gas Operation” was represented as followed:

```xml
<JobStep>
  <StepTitle>Introduction and removal of purge to POC</StepTitle>
  <Ctxt_App_Condition>
    <Concept_AND>
      <Concept>Purge_Gas_Operation</Concept>
      <Concept>Introduction_And_Removal_Of_Purge_To_POC</Concept>
    </Concept_AND>
  </Ctxt_App_Condition>
  ...... 
</JobStep>
```

**B2: Represent multiple Job Step concepts.** A Job Step section may also contain several steps of an activity which were better described together. In these cases, multiple Job Step concepts were represented together with the Activity concept to describe the context of this section’s applicability condition in a Job Step section.

**Example 5:** A Job Step section “Set up flow meters; Install purge lines” of the Activity section “Purge Gas Operation” was represented as followed:

```xml
<JobStep>
  <StepTitle>Set up flow meters; Install purge lines</StepTitle>
  <Ctxt_App_Condition>
    <Concept_AND>
      <Concept>Purge_Gas_Operation</Concept>
      <Concept>Set_Up_Flow_Meters</Concept>
      <Concept>Install_Purge_Lines</Concept>
    </Concept_AND>
  </Ctxt_App_Condition>
  ...... 
</JobStep>
```
Scenario C: Representation of Potential Hazard sections. Potential Hazard title was first represented, which was followed by the representation of applicability condition and Recommended Safety Procedure/Safety Rule section. The applicability condition was described through Activity, Job Step and Potential Hazard concepts, i.e. this is the other occasion for multi-type description; the Safety Rule section was represented through listing all safety rules corresponding to the Activity, Job Step, and Potential Hazard concepts.

C1: Represent single Potential Hazard concept. In this sub-scenario, a Potential Hazard section only had a single hazard contained in this section. Therefore, a Potential Hazard concept was sufficient, together with Activity and Job Step concepts, to describe the context of this section’s applicability condition.

Example 6: A Potential Hazard section “Material dislodgement” of the Job Step section “Fly forms to area to be installed” was represented together with five safety rules as followed:

```
<PotentialHazard>
  <HazardTitle>Material dislodgement</HazardTitle>
  <Ctx_App_Condition>
    <Concept_AND>
      <Concept>Frame_Columns</Concept>
      <Concept>Fly_Forms_To_Area_To_Be_Installed</Concept>
      <Concept>Material_Dislodgement</Concept>
    </Concept_AND>
  </Ctx_App_Condition>
  <RecommendedProcedure>
    <Rule>(1) Inspect rigging to be/being used.</Rule>
    <Rule>(2) Ensure proper rigging method.</Rule>
    <Rule>(3) Ensure direct contact with crane operator.</Rule>
    <Rule>(4) Clear area to land material.</Rule>
    <Rule>(5) Use taglines.</Rule>
  </RecommendedProcedure>
</PotentialHazard>
```
C2: Represent multiple Potential Hazard concepts. A Potential Hazard section may also contain several hazards at the same time. This occasion happened when these hazards were associated with the same Job Step and related to the same safety rules. In these cases, multiple Potential Hazard concepts were disjunctively represented and then this disjunction was represented together with the Activity and Job Step concepts conjunctively to describe the context of this section’s applicability condition. Logical OR was used for these multiple Potential Hazard concepts in this sub-scenario because the listed safety rules can be applied to any of them.

Example 7: A Potential Hazard section “Dropping of tools, trips, falls” of the Job Step section “Fly forms to area to be installed” was represented together with two safety rules as followed:

```xml
<PotentialHazard>
  <HazardTitle>Dropping of tools, trips, falls</HazardTitle>
  <Concept_App_Condition>
    <Concept_AND>
      <Concept>Purge_Gas_Operation</Concept>
      <Concept>Set_Up_Flow_Meters</Concept>
      <Concept>Install_Purge_Lines</Concept>
      <Concept_OR>
        <Concept>Dropping_Of_Tools</Concept>
        <Concept>Trip</Concept>
        <Concept>Fall</Concept>
      </Concept_OR>
    </Concept_AND>
  </Concept_App_Condition>
  <RecommendedProcedure>
    <Rule>(1) 100% fall protection.</Rule>
    <Rule>(2) Do not lay tools on equipment or grating. Keep tools in toolbox or similar container.</Rule>
  </RecommendedProcedure>
</PotentialHazard>
```

5.4.1.3 Summary of validation of document representation

The developed textual document representation model allowed me to model all the acquired JHA documents no matter what kinds of representation scenario they had. A JHA document can be represented
in a XML file by integrating the necessary steps discussed in the abovementioned scenarios. For example, the JHA document in XML format shown in Figure 3.14 in 3.4.4 can be obtained by integrating example 1, 3 and 6 of the three scenarios above. Integrating example 2, 4, 5 and 7 can generate another complete JHA document in XML format that was shown in Appendix E. Therefore, the capability and flexibility of the textual document representation model to represent JHA documents was successfully validated.

The textual document representation model has a representation limitation: modeling construction safety documents in XML format will become cumbersome when the size of the documents becomes huge. To address this limitation, using an editor which, for example, can automatically generate XML tags for users to fill in with required information is a better alternative than using a simple text editor, e.g. NotePad. In addition, having an editing tool which provides user interfaces for directly entering information, such as those in the prototype system JHA Advisor, rather than using markup tags also is an option to address the limitation.

Another aspect related to the textual document representation model that requires validation is the reasoning about the textual documents represented in XML format using the developed safety requirement reasoning engine. The required reasoning mechanism aims to automatically evaluate JHA sections’ applicabilities to a given search contextual conditions. The validation of the reasoning mechanism provided by the developed safety requirement reasoning engine is discussed in the following subsections.
5.4.2 Reasoning about JHA Documents

I validated the reasoning engine developed for reasoning about textual documents and for evaluating
applicability of different JHA sections to given contexts by conducting several test cases. In these test
cases, I specified contexts to test whether the reasoning rules developed in 4.3.2 can function correctly,
i.e., properly propagating applicability from concepts of the concept ontology to represented JHA
documents as well as evaluating safety rules’ applicability to the specified contexts. In addition, the safety
requirement reasoning engine was implemented in JHA Advisor so the test cases conducted here were also
used to evaluate whether JHA Advisor functioned properly to implement the reasoning process provided
by the reasoning engine.

5.4.2.1 Evaluation measure and criteria

A formal measure is necessary in the test cases to assess how effective the reasoning engine was in
terms of evaluating safety rules’ applicability, e.g. whether the identified safety requirements which are
evaluated to be applicable really apply to the specified contexts. In the document retrieval domain, the two
basic and most frequently used measures for evaluating information retrieval effectiveness are precision
and recall, which are defined as “the fraction of retrieved documents that are relevant” and “the fraction of
relevant documents that are retrieved” respectively (Manning et al. 2008). Precision and recall can be
represented in the following formulas:

\[
\text{Precision} = \frac{|\text{relevant documents} \cap \text{documents retrieved}|}{|\text{documents retrieved}|}
\]

\[
\text{Recall} = \frac{|\text{relevant documents} \cap \text{documents retrieved}|}{|\text{relevant documents}|}
\]
The higher the precision value is, the more relevant the retrieved documents are; on the other hand, the higher the recall value is, the more documents which are relevant are retrieved. Manning et al. (2008) also pointed out that it is ideal to get as high recall value as possible. This is because people prefer to know more documents which are potentially relevant are retrieved (high recall value) rather than knowing the retrieved documents are relevant but being unaware of how many relevant documents have not been retrieved (high precision value).

In the construction safety domain, engineers prefer that more applicable safety requirements which are relevant (i.e., relevant to one another and also relevant to the specified contexts) can be identified. This indicates that a higher recall value for the proposed approach is preferred. Sacrificing recall value for precision value may result in applicable safety requirements being ignored, which can lead to safety issues on the construction sites.

In the validation, I only use recall to evaluate the effectiveness of the safety requirement reasoning engine and the formula for calculating recall value was adapted to:

\[
\text{Recall} = \frac{|\text{relevant safety rules} \cap \text{safety rules retrieved}|}{|\text{relevant safety rules}|}
\]

The “relevant” in this formula means that when certain JHA concepts are assigned specific applicability value applicable or not applicable, the safety rules relating to these concepts (e.g. the rules were in the same JHA sections as the concepts) should be evaluated to be applicable or not applicable accordingly. Therefore, I use this measure in the validation to evaluate whether the safety rules relevant to the concepts describing the specified contexts are retrieved. Specifically, recall value is expected to be
equal to 1 in the validation since the developed safety requirement reasoning engine aims to identify all the safety rules which are relevant to the concepts describing the specified contexts and to evaluate them to be *applicable* or *not applicable*. This also explains why precision is not specifically considered in the validation: since higher recall value is preferred and recall value in the validation is expected to be 1, there is no need to be concerned with precision value is in this regard.

5.4.2.2 Concept neglect mechanism

Another important issue that needs to be pointed out is *concept neglect*, which means to neglect certain concepts describing applicability conditions in a reasoning process such that these concepts’ applicability values are ignored and not evaluated by the safety requirement reasoning engine. The ultimate goal of concept neglect is to adjust the reasoning strategies for JHA sections whose applicability conditions are fully or partially described through these ignored concepts. For the fully described conditions, concept neglect helps to ignore the whole JHA sections; for the partially described conditions, concept neglect allows the JHA sections’ applicability to be independent of the ignored concepts.

Intuitively, safety rules are tied to a combination of Activity, Job Step and Potential Hazard concepts while they also can be tied to potential hazard concepts alone. In addition, engineers may want to know what safety rules are applicable when certain job steps are performed regardless of the hazards associated with the job steps, or when certain potential hazards occur on sites no matter what steps are taken. Therefore, these facts are considered in this research by implementing the concept neglect mechanism in the safety requirement reasoning engine. Figure 5.4 is an example of how concept neglect works. Suppose
I want the Activity concept “Frame Columns” to be neglected during the reasoning process. The reasoning engine then will only reason about the concept “Fall”; in this case, the original applicability condition with conjunctive description turns to be equivalent to an applicability condition described through a single concept “Fall”.

The concept neglect mechanism allows engineers to search for safety rules which are associated with specific concepts or specific types of concepts while ignoring others. From reasoning perspective, it enhances the safety requirement reasoning engine by increasing the engine’s capability of reasoning about JHA documents under different user-specified conditions. As shown in the last two columns of Table F.1 in Appendix F, safety rules can have more possible results of applicability evaluation in different scenarios with concept neglect mechanism (the column with heading “w/ CN”) than without the mechanism (the column with heading “w/o CN”, which is the same as the last column of Table D.1). In other words, concept neglect mechanism brings multi-type conjunctive description reasoning flexibilities. In the next subsection, I discuss the validation of the safety requirement reasoning engine in which concept neglect mechanism is deployed in some scenarios and therefore, this concept neglect mechanism is also validated together with the validation of the engine.
5.4.2.3 Cases of validating safety requirement reasoning engine

The safety requirement reasoning engine is built upon the concept ontology reasoning engine by using concepts’ applicability to evaluate safety rules’ applicability; therefore, the test cases performed in this section is built upon the cases conducted in section 5.3.2.1.

I started the validation of the safety requirement reasoning engine by conducting single concept tests: I assigned applicability values to single concept and evaluated how well the engine worked. Then I conducted multiple concept tests, which were to assign applicability values to multiple concepts and observe the reasoning results.

*Test Scenario A: Single concept tests.* In this scenario, I tested how well the reasoning engine functioned when assigning *applicable* or *not applicable* value to single concept.

*A1: Assign applicable value to single concept*

I assigned *applicable* value to each of the represented concepts listed in column 2 of Table B.1, Table B.2 and Table B.3 in Appendix B, and used recall value to evaluate whether all the relevant *applicable* or *not applicable* safety rules were correctly evaluated and identified. The test case results were shown in Table G.1, Table G.2 and Table G.3 in Appendix G. These tables also listed whether equivalent and disjoint relationships were involved in the whole concept reasoning process, i.e. not only in reasoning about the test concepts but also in reasoning about other concepts associated with the test concepts.

Table G.1 demonstrated the tests on assigning *applicable* value to Activity concepts and the results
of these tests showed that the safety requirement reasoning engine successfully identified *applicable* safety rules (applicable recall values\(^1\) in the tests were equal to 1) and *not applicable* safety rules (not-applicable recall values\(^2\) in the tests where disjoint relationship was involved were equal to 1). It is noteworthy that *not applicable* safety rules were only available in the test cases when there were disjoint relationships involving in the tests.

Table G.2 and Table G.3 demonstrated the tests on assigning *applicable* value to Job Step and Potential Hazard concepts respectively. Since the assignment of applicability value was not on Activity concepts, concept neglect mechanism was involved in these test cases to ensure the reasoning engine’s proper functioning: Activity concepts were all neglected in the tests shown in Table G.2 while both Activity and Job Step concepts were neglected in the tests shown in Table G.3.

The results of these test cases in Table G.2 and Table G.3 also showed that the safety requirement reasoning engine successfully identified *applicable* and *not applicable* safety rules except for the test case number 30 in Table G.2. The reason why the applicable recall value for this test case was not equal to 1 was that multiple Job Step concepts were used to describe the context for the applicability condition, as the *Scenario B2* showed in section 5.4.1.2. Therefore, only assigning *applicable* value to one of the Job Step concepts was not sufficient to determine the corresponding safety rules’ applicability value. The test results of test scenario A1 were illustrated in Figure 5.5.

---

\(^1\) Applicable recall value is calculated for the identified *applicable* safety rules that are indexed through the test concept as well as through the inferred *applicable* concepts which are related to the test concept.

\(^2\) Not-applicable recall value is calculated for the identified *not applicable* safety rules that are indexed through the concepts which are disjoint with the inferred *applicable* concepts.
**A2: Assign not applicable value to single concept**

The evaluation of assigning not applicable value to single concept can be illustrated using those test cases of scenario A1 in which disjoint relationships were involved since disjoint relationship helped reason out not applicable concepts from applicable ones. For example, specifying the test concept in test case number 2 in Table G.1 applicable resulted in making another Activity concept “Excavation Using Slopes” not applicable due to the disjoint relationship. The developed safety requirement reasoning engine then successfully identified the not applicable safety rules for this not applicable Activity concept. As shown in Appendix G, when single concept was automatically given not applicable value through reasoning over disjoint relationships, not applicable safety rules can be properly evaluated and identified.

*Test Scenario B: Multiple concept tests.* In this scenario, I discussed how well the safety requirement reasoning engine worked when assigning applicable and/or not applicable value to multiple concepts. For
simplification, I used two Activity concepts in each test case in the following demonstration.

**B1: Assign applicable value to multiple concepts**

Two representative test cases were conducted to demonstrate this sub-scenario. In the first case, two Activity concepts “Frame Columns” and “Excavation Using Support Systems” were assigned *applicable* value. As shown in Table G.4, the applicable and not-applicable recall values were both equal to 1. In addition, the numbers of *applicable* and *not applicable* safety rules in the combined reasoning were equal to the sum of the numbers in the separate reasoning. This meant the processes of reasoning about the two concepts were independent, i.e. one process was not affected by the other.

In the second case, another two Activity concepts “Forklift Use” and “Decontamination of Windows” were assigned *applicable* value. The applicable and not-applicable recall values were both equal to 1 in this case. However, the number of *not applicable* safety rules in the combined reasoning was different from the sum of the numbers of those in the separate reasoning. This indicated that concept applicability contradiction occurred in the reasoning process and there were concepts whose applicability values were updated during the reasoning about the second concept.

**B2: Assign not applicable value to multiple concepts**

When concepts are assigned *not applicable* value, no concepts will be deduced *applicable* according to the defined concept ontology reasoning principles. Therefore, if the concepts which are assigned *not applicable* value are used to describe the context for applicability conditions and are incorporated in logical AND, the safety rules having the applicability conditions will be evaluated *not applicable* according to the safety requirement reasoning rules.
The test case shown in Table G.5 proved this argument. Two Activity concepts “Carcinogen Control” and “Spray Painting” were assigned not applicable value and then 27 not applicable safety rules were successfully identified. No safety rules were evaluated applicable as expected.

**B3: Assign both applicable and not applicable values to multiple concepts**

This sub-scenario is a combination of previous two sub-scenarios. I listed in Table G.6 the result of a test case conducted for this sub-scenario, in which an Activity concept “Frame Columns” and one of its Job Step concepts “Set Pins” were respectively assigned applicable and not applicable values. The result illustrated that both applicable and not applicable recall value were equal to 1. That is, the safety requirement reasoning engine was also proved to function properly in reasoning of this sub-scenario.

5.4.2.4 Summary of and discussion on validations of safety requirement reasoning

I introduce the notion of concept neglect mechanism in this section, which adds reasoning flexibility into the safety requirement reasoning engine. By conducting several test cases, some of which apply concept neglect mechanism, I prove that the safety requirement reasoning engine is capable of identifying safety rules which are evaluated applicable or not applicable. The reasoning engine also can successfully classify these safety rules according to their applicabilities, and the classified rules are output into an MS Excel file using JHA Advisor.

In the scenario of assigning applicability value to single concept, the safety requirement reasoning engine does not successfully identify applicable safety rules only when the concept being assigned value
is one of several Job Step or Potential Hazard concepts together representing applicability conditions (as discussed in Scenarios B2 in section 5.4.1.2).

In the scenario of assigning applicability values to multiple concepts, reasoning processes is performed in turns (one process will not start until previous one is finished), and the reasoning results are cumulative. Cumulative reasoning mean the safety rules, applicable or not applicable, identified in later reasoning process can join those in earlier reasoning to form the final set of safety rules. Examples of cumulative applicable safety rules are shown in test cases 1 and 2 of Table G.4; examples of cumulative not applicable ones are shown in test case 1 of Table G.4, and test cases in Table G.5 and Table G.6. However, there is an exception to this cumulative feature: when different reasoning processes have common concepts whose applicability values change (from applicable to not applicable, or the other way around) during reasoning, the results of reasoning about safety rules will not be cumulative. An example of non-cumulative applicable safety rules is shown in the test case of Table G.5 while an example of non-cumulative not applicable ones is shown in test case 2 of Table G.4. Situations of non-cumulative reasoning usually involve concept applicability contradiction and users have to resolve the contradiction before continuing the reasoning process, as discussed in section 4.2.3.2.

When applicable and not applicable safety rules are identified through assigning applicability values to concepts, users only need to focus on the applicable safety rules if they concentrate on the involved concepts only. The not applicable safety rules do not play a part until new concepts are considered and involve in later cumulative reasoning processes. That is, users need not to consider these not applicable safety rules until they start to assign applicability values to other concepts and identify additional
One limitation of the safety requirement reasoning engine was observed during the validation: if there are redundant safety requirements identified, i.e. safety requirements with exactly the same content, the reasoning engine currently does not remove redundancies. Although this limitation does not affect the correct operation of the reasoning engine, it may lessen the readability of identified safety requirements when the number of them becomes larger. This issue is expected to be addressed in future research, which I discuss in detail in section 6.4.5.

5.5 CONTEXT- VERSUS KEYWORD-BASED APPROACHES

In previous two sections, I respectively validate the Representation Model and Reasoning Engine of the developed Construction Safety Document Management Framework. To demonstrate the capability of the Framework in general, I conduct test cases to compare the context-based approach developed in this research with keyword-based approach. I use both approaches to search for safety rules related to given search scenarios from the acquired JHA documents. Lastly, I compare two approaches and discuss their merit and limitations based on the results gained in this test case.

To use keyword-based search, I focus on each 4-tuple formed with a safety rule and its corresponding JHA (i.e. Activity, Job Step and Potential Hazard concepts) concepts. A safety rule is deemed to be related to certain specified keywords only when these keywords appear in the 4-tuple of the safety rule, that is the safety rule or its corresponding JHA concepts contain these keywords. In addition, it is possible that the
identified safety rules were related to the keywords but are not relevant to the given scenario (An example was given later). Therefore, I also perform manual check in the keyword-based search process in this test case to ensure the identified rules apply to the keywords as well as the given scenario.

5.5.1 Search Safety Rules Relevant to Search Scenario “Pour Concrete”

5.5.1.1 Application of the context-based approach

The first test case is to search safety rules relevant to a search scenario “Pour Concrete”. When using the context-based search approach to search safety rules for the given activity scenario “Pour Concrete”, I first discover that there is not such a concept defined in the concept ontology for describing the context. However, I easily find three concepts which are related to the scenario under the concept “Concrete Activity”, which are “Pour Columns”, “Pour Walls” and “Pour Deck with Pump”. Therefore, by assigning these three concepts applicable value, the safety rules relevant to the scenario “Pour Concrete” are identified. In this context-based approach, these rules were flagged applicable, as discussed throughout this research, meaning they applied to the search scenario described by the three concepts.

To make it more straightforward to use this approach to search safety rules for the scenario, I purposefully define a new concept “Pour Concrete” in the concept ontology as a subclass of the concept “Concrete Activity”. In addition, I define a new association relationship, named “consistsOf”, and used this relationship to connect the new concept “Pour Concrete” to the search concepts “Pour Columns”, “Pour Walls” and “Pour Deck with Pump”. After this arrangement, I assign applicable value to the
concept “Pour Concrete”, and then the three associated concepts are automatically evaluated to be applicable and accordingly, 61 safety rules were correctly identified.

Moreover, another concept “Place Concrete” is also defined as a subclass of the concept “Concrete Activity” and I declare this new concept is equivalent to the concept “Pour Concrete”. Then, when this new concept is assigned applicable value, the same reasoning result as the previous one is obtained.

Table H.1 in Appendix H summarizes the result of using the context-based approach to search safety rules. It shows that there is no irrelevant safety rules identified using the approach in the test case. This meant safety rules which are identified applicable using the context-based approach are definitely relevant to the specified scenario.

5.5.1.2 Application of the keyword-based approach

To use keyword-based approach for the given scenario, the titles of the concept used in the context-based search are used as keywords in this keyword-based search. First, using the keywords “pour columns”, “pour walls” and “pour deck with pump” to search through the documents return the same related safety rules as using these concepts in the context-based search. Then, the keyword “pour concrete” is used in the search, and 52 relevant safety rules are identified, which is nine rules fewer than those identified through context-based search. That is, these nine rules that are relevant to the search scenario are missing in the keyword-based search. In addition, one irrelevant safety rule is identified, which is “Permanent metal stairways into which concrete is later poured, shall be temporarily filled with wood or
other solid material.” This rule was related to the keyword “pour concrete” but not relevant to the specified scenario “Pour Concrete”. Lastly, I use the keyword “place concrete” to perform the search and only one safety rule is found relevant. The result is shown in Table H.2 in Appendix H.

5.5.2 Search Safety Rules Relevant to Search Scenario “Steel Erection”

The second test case aims for search safety rules relevant to a search scenario “Steel Erection”. To use the context-based approach, I first identify two concepts of the concept ontology “Steel Erection Operation” and “Steel Erection”, which best describe the search scenario. Then, I separately assign applicable value to the two concepts and both return the same 18 relevant safety rules. In addition, the context-based approach also helps identify 31 safety rules which are not applicable to the search scenario, which is because there are disjoint relationships involved in the reasoning process. Table H.3 in Appendix H summarizes the result of using the context-based approach to search safety rules in this scenario.

To use the keyword-based approach, I choose two keywords “steel erect” and “steel erection” to perform the search. Both keywords help identify 18 relevant safety rules, which are the same as those identified using the context-based approach. However, no not applicable safety rules identified, as the context-based approach does, through the keyword-based approach. The result is shown in Table H.4.

5.5.3 Discussion on Comparison between Context- and Keyword-based Approaches

According to the test cases of comparing the two approaches, the context-bases approach developed
in this research have the following merits over the keyword-based one:

(1) Identify specific safety rule:

The context-based approach is capable of identifying safety rules which are actually applicable to specific scenarios. This merit can be illustrated from two aspects. First, using context-based approach only identifies semantically relevant (i.e. applicable) safety rules. However, although using keyword-based approach can identify relevant safety rules, it also may miss certain relevant rules and identify irrelevant ones, as the search using the keyword “pour concrete” shows in the first test case. Therefore, efforts of manually checking these irrelevant safety rules can be saved for the context-based approach but not for the keyword-based approach. Second, using context-based approach can help rule out safety rules which are not applicable to specific scenarios which are described by defined concepts with disjoint relationships. As discussed in section 5.4.2.4, identifying not applicable safety rules is beneficial when users consider adding more concepts into the reasoning process. Keyword-based approach, on the other hand, does not help reasoning in this regard since it cannot reason about the semantics of keywords which are used for search.

(2) Allow flexible search tasks:

The context-based approach is more flexible than keyword-based approach in terms of its capability of adding new concepts in and removing existing concepts from concept ontology. Flexible concept representation allows the approach to incorporate new concepts, enables reasoning about the new and existing concepts, and therefore benefits the reasoning about JHA documents. In addition, since contexts are described through concepts, flexible arrangement of concepts also shows the approach’s ability to describe and index more possible contextual information no matter how complicated the information is.
In addition to these two merits, the context-based approach indeed includes the keyword-based search in a sense: engineers can use keywords to search through the concept ontology to find out proper concepts to describe contexts. This function is implemented in JHA Advisor and enhances the management of concepts represented in the concept ontology.

While the context-based approach allows concepts to be flexibly represented and edited in an ontology and used to index JHA documents, this approach has to rely on ontology authoring and document modeling tools respectively for editing concepts and preparing documents, as discussed in section 5.2.1, if there is no tools which integrate both functions. Thereby, JHA Advisor is developed in the research to serve this integration purpose; I validate JHA Advisor by using a representative example to illustrate its functionality and also show its strength and weakness in the next section.

5.6 PREPARE A JHA DOCUMENT USING JHA ADVISOR

The purpose of this test case is to illustrate how easy or difficult JHA Advisor is to help define new concepts and prepare new documents using existing and newly defined concepts. I choose two JHA documents from the acquired documents for the demonstration in this section. The two documents are for excavation-related activities “Excavation Using Support Systems” and “Excavation Using Slopes”. Table 5.2 lists the two documents’ Activity, Job Step and Potential Hazard concept information.

In a synthetic scenario, I test in this case how to use JHA Advisor to create a document for the activity “Excavation Using Slopes” (target document), given that the background knowledge for this
activity and the other document for activity “Excavation Using Support Systems” (reference document) is available, and that the JHA concepts of the reference document are already in the concept ontology.

Table 5.2: Two JHA documents for JHA Advisor Validation

<table>
<thead>
<tr>
<th>Concept Type</th>
<th>JHA Concept Information</th>
<th>Reference Document</th>
<th>Target Document</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity</td>
<td>Excavation Using Support Systems</td>
<td>Excavation Using Slopes</td>
<td></td>
</tr>
<tr>
<td>Job Step</td>
<td>Excavation using shoring manufactured trench boxes or other support systems</td>
<td>Excavation using sloping and benching as protective measures</td>
<td></td>
</tr>
<tr>
<td>Potential Hazard</td>
<td>● Collapse of trench walls</td>
<td>● Collapse of excavation walls</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Failure of shoring/support system</td>
<td>● Vehicular traffic</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Soil caving in below the protective structure or shield</td>
<td>● Falling objects</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Employees being injured during positioning of shield or structure</td>
<td>● Limited access and egress</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Employees being injured by wall collapse outside of protective structure</td>
<td>● Hazardous atmosphere</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Vehicular traffic</td>
<td>● Electric Shock</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Falling objects</td>
<td>● Falling into excavations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Limited access and egress</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Hazardous atmosphere</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Electric Shock</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Falling into excavations</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

After reviewing the concept ontology and the reference document in *JHA Advisor*, I first find that no existing Activity and Job Step concepts in the ontology can be referred to in order to create the target document. Therefore, I need to define them in the concept ontology. I also find that the Potential Hazard concepts required for the target document is much the same as those in the reference document, except that a Potential Hazard concept “Collapse of excavation walls” should be added instead of using the existing concept “Collapse of trench walls”. Therefore, I have to define this new concept in the ontology but can reuse those same existing concepts. Lastly, I find that most of the safety rules in the reference document are suitable for being used in the target document, except that the Potential Hazard “Falling objects” should specify one safety rule that is not in the reference document.
5.6.1 Add New Concepts

Three new concepts should be defined in this validation: an Activity concept “Excavation Using Slope”, a Job Step concept “Excavation using sloping and benching as protective measures”, and a Potential Hazard concept “Collapse of excavation walls”. Screenshots of using JHA Advisor to add these new concepts are shown in Step 1 in Appendix I. Sub-steps include: click “Add Concept” button in JHA Advisor (Step 1a); type in concept name in the concept editor; specify the superclass into which the new concept is categorize; and, if applicable, select other concepts to which the new concept connects through association and logical relationships (Step 1b, c, d).

It is noteworthy that when using JHA Advisor to define new concepts which belong to the same JHA Activity section, the concepts are recommended to be defined according to the order of concept types from Potential Hazard to Job Step to Activity. This is because the latter can directly connect the former through association relationships once the former is already defined in the ontology.

5.6.2 Add a New JHA Activity Section

When the required concepts are defined in the ontology, they and other existing concepts can be used to describe the context for the applicability conditions of the target document. Screenshots of using JHA Advisor to add these new concepts are shown in Step 2 in Appendix I.

First click “Add JHA Activity” button in JHA Advisor (Step 2a); type in activity name and select
proper concepts for the applicability condition in a pop-up JHA Activity editor; and click “Add” button to keep adding new Job Step sections which are included by this Activity section. When JHA Job Step editor shows up, similar steps are taken to input job step name, concepts for applicability condition, and click “Add” button to continue adding Potential Hazard sections included by the Job Step section. For adding Potential Hazard sections, safety rules are finally added, following the input of potential hazard name and select concepts for applicability condition (Step 2b, c, d). Lastly, the newly added section can be seen in JHA Advisor (Step 2e).

Although I only show the procedure of adding a new Potential Hazard section for the concept “Collapse of excavation walls” in Step 2 in Appendix I, other sections that reuse the existing concepts of the reference document can be added into the new JHA document following the same steps.

5.6.3 Discussion

This test case illustrates JHA Advisor’s capability and flexibility of adding new concepts and composing JHA sections for a new JHA document. The new concepts together with other existing concepts of the reference document shown in Table 5.2 can be used to describe the applicability condition of those new JHA sections. In addition, the safety rules in the existing, reference document can be referred to as well: when I compose the safety rules for a new Potential Hazard section, I can scroll down the JHA document working area in JHA Advisor to find those safety rules in the existing Potential Hazard sections and reuse them with required revision.
While the screenshots in Appendix I are only for demonstrating the “Add Concept” and “Add Activity Section” of JHA Advisor, the user interface, functions and steps of modifying concepts and JHA sections are similar and therefore, the details of these are not given in the research.

Not only do the developed Framework and JHA Advisor aim for retrieving previous JHA documents using the concepts only within the documents, but they also aim for preparing new JHA documents using concepts from all collected documents. For example, when new equipments are adopted in the “Frame Columns” activity, Job Step and Potential Hazard concepts in the ontology which are related to the new equipments and different from those concepts associated with the concept “Frame Columns” need to be specified applicable, in addition to only specifying the Activity concept “Frame Columns” applicable. In other words, the concept ontology and the safety documents modeled in XML format together can be analogous to a database and JHA Advisor is to a database system: assigning concepts applicable or not applicable values acts as specifying queries to the system; identifying applicable safety requirements acts as finding out data matching the queries. This feature enables users to flexibly retrieve, add and remove applicable safety requirements using the developed Framework and assigning applicability values to concepts.

Integrating both ontology editing and document modeling functions in JHA Advisor facilitates the preparation of new JHA documents as well as management of concepts. First, users need not to switch between ontology editing and document modeling tools to deploy the developed Framework on JHA tasks. Second, users do not have to directly deal with XML and ontology syntax in order to organize JHA concepts and compose JHA documents.
*JHA Advisor* also has limitations. For example, if a specific existing Potential Hazard section is to be reused in a Potential Hazard section of a new JHA document, *JHA Advisor* currently cannot retrieve the safety rules from the existing sections and put them as default safety rules in the new section automatically. Users have to input these safety rules manually. This becomes cumbersome especially when lots of existing JHA information is reused in new JHA documents. In addition, *JHA Advisor* currently does not allow users to start organizing concepts and preparing documents from scratch, and users have to read in existing OWL and XML files for their following work. This issue detrimentally affects the convenience of using *JHA Advisor*. However, in spite of these limitations in *JHA Advisor*, they do not harm its ability of implementing the representation and reasoning functions of the developed Framework. Hence, *JHA Advisor* is proved to be able to realize the developed Construction Safety Document Management Framework for managing JHA information.

5.7 VALIDATION OF REASONING RATIONALE FEEDBACK

I discuss in section 4.4 the reason why an essential mechanism, feedback of reasoning rationale, should be taken into account in the developed Framework. In this research, this feedback mechanism is implemented in *JHA Advisor*. There are three ways by which users can retrieve the rationale of the concept and document reasoning processes.

5.7.1 Store Reasoning Rationale in A Log File

The first way is designed to show reasoning rationale for concept reasoning only. When starting *JHA*
Advisor, users are asked to specify a location in local hard drive to which a log file is to be saved. When reasoning about concepts is performed and finished, the log file is automatically updated with information of what reasoning processes have been done. For example, if a user specifies an Activity concept “Grind Concrete” to be applicable, a snippet of the log file will look like:

NEW FACT: Grind_Concrete FOUND _applicable_
BECAUSE (Grind_Concrete _applicable_) AND
(Grind_Concrete SUBCLASS_OF Concrete_Activity)

NEW FACT: Concrete_Activity FOUND _applicable_
BECAUSE (Concrete_Activity _applicable_) AND (Concrete_Activity SUBCLASS_OF Activity)

NEW FACT: Activity FOUND _applicable_

The above shows the rationale of how the superclasses of the concept “Grind Concrete” are concluded to be applicable. For example, because this concept is a subclass of a concept “Concrete Activity” (line 3), the concept “Concrete Activity” is evaluated to be applicable.

5.7.2 Show Reasoning Rationale in A Reasoning Window

A reasoning window is implemented in JHA Advisor, which can be activated by clicking “Show Reasoning Rationale Window” from the “Reasoning” menu. The window prints information of reasoning rationales for concept reasoning as well as document reasoning. Following the previous example, if I click a Job Step concept “Accessing Work Area” from the ontology in JHA Advisor, the following will be shown in the concept reasoning rationale area of the reasoning window:

BECAUSE (Grind_Concrete _applicable_) AND
(Grind_Concrete REFER_TO Accessing_Work_Area)

NEW FACT: Accessing_Work_Area FOUND _applicable_
This explains that the concept “Accessing Work Area” is *applicable* because it is referred to by an *applicable* concept “Grind Concrete”. In addition, if I click the title of a Job Step section “Accessing Work Area” in *JHA Advisor*, the following will be shown in the document reasoning rationale area of the reasoning window:

\[ \text{BECAUSE the Applicability Condition(s):} \]
\[ (\text{Grind} \_\text{Concrete}), \ AND \ (\text{Accessing} \_\text{Work} \_\text{Area}) \_\text{satisfied}, \]
\[ \text{[Job Step]} \ (\text{Accessing work area}) \ \text{FOUND} \ _\text{applicable}. \]

This explains that this Job Step section is *applicable* because its applicability condition, which is described by two *applicable* concepts “Grind Concrete” and “Accessing Work Area”, is *satisfied*.

5.7.3 Show Reasoning Rationale Using Tooltip¹

Rather than showing reasoning rationales in the reasoning window, the third way designed to give users feedback of the rationales is to use tooltips in *JHA Advisor*. When users click a certain concept or JHA section, a tooltip will appear and the rationale will be shown in the tooltip. Following the previous example, the same information shown in the reasoning windows will be shown to users in tooltips.

5.7.4 Discussion

Providing reasoning rationales to users aims to convey the proposed reasoning rules and principles to them in a language that can be more easily understood. The three proposed ways work properly to achieve

---

¹ According to Wiktionary, a tooltip is defined as “an element of a graphical user interface in the form of a box of text that appears when a cursor is made to hover over an item; normally used to explain the function of the item”.

152
this goal. While the second and third ways give quick rationale feedback for concepts or JHA sections, referring to the log file is still the best way to have a thorough understanding of the reasoning processes.

In addition, the mechanism of reasoning rationale feedback works only for applicable and not applicable concepts and JHA sections, and not for possibly applicable ones. For example, if users click a concept which is possibly applicable, no reasoning rationale will be returned to users. This is because only applicable and not applicable information has been involved in the reasoning processes, no matter whether the applicability values are automatically evaluated or assigned by users.

5.8 REASONING ABOUT JHA DOCUMENTS WITH APPLICABILITY EXCEPTIONS

In the developed Framework, specifying applicability exceptions for a safety requirement allows users to describe the particular circumstances under which the requirement does not apply as presented in section 4.3.1. The validations discussed in this chapter so far do not take into account applicability exceptions in those test cases because the acquired JHA documents have no applicability exceptions specified. Therefore, I conduct a synthetic case study in this section to validate the safety requirement reasoning engine’s ability in reasoning about applicability exceptions for JHA documents.

5.8.1 Descriptions of Applicability Exceptions

This case study is based on the JHA document for the activity “Frame Columns” shown in Figure 3.14. I adapt the document that is in XML format by adding synthetic applicability exceptions into it. First,
I assume that when concrete columns are precast, the activity “Frame Columns” does not apply anymore. To reflect this assumption, I add a new Activity concept “Precast Concrete Columns” under the concept “Concrete Activity” in the concept ontology and use this concept to describe the context for the applicability exception added to the JHA Activity section “Frame Columns”.

In addition, I assume that if the forms to be used for framing columns are moved to the designated areas manually, then the first job step “Fly forms to area to be installed” does not apply. To represent that in the document, I add a new Job Step concept “Manually Move forms to area to be installed” under the Job Step concept “Move Step”; then I use this new concept together with the “Precast Concrete Columns” to disjunctively describe the context for the applicability exception added to the Job Step section “Fly forms to area to be installed”.

After considering the assumptions, the new JHA document represented in XML format is:

```xml
<Activity>
   <ActTitle>Frame Columns</ActTitle>
   <Ctxt_App_Condition>
      <Ctxt>Frame_Columns</Ctxt>
   </Ctxt_App_Condition>
   <Ctxt_App_Exception>
      <Ctxt>Precast_Concrete_Columns</Ctxt>
   </Ctxt_App_Exception>
</Activity>

<JobStep>
   <StepTitle>Fly forms to area to be installed</StepTitle>
   <Ctxt_App_Condition>
      <Ctxt_AND>
         <Ctxt>A1_Frame_Columns</Ctxt>
         <Ctxt>Fly_Forms_To_Area_To_Be_Installed</Ctxt>
      </Ctxt_AND>
   </Ctxt_App_Condition>
   <Ctxt_App_Exception>
      <Ctxt_OR>
         <Ctxt>Precast_Concrete_Columns</Ctxt>
         <Ctxt>Manually_Move_Forms_To_Area_To_Be_Installed</Ctxt>
      </Ctxt_OR>
   </Ctxt_App_Exception>
</JobStep>
```
5.8.2 Reasoning about JHA Sections with Applicability Exceptions

I used several test cases to test whether the safety requirement reasoning engine is capable of correctly reasoning about the applicability exceptions in the case study. The test scenarios in the test cases included:

- Both the Activity concepts “Frame Columns” and “Precast Concrete Columns” are assigned applicability value *applicable*.
- The Activity concept “Frame Columns” and the Job Step concept “Manually move forms to area to be installed” are assigned applicability value *applicable*.
- The Activity concept “Frame Columns” is neglected through the concept neglect mechanism; both the Job Step concepts “Fly forms to area to be installed” and “Manually move forms to area to be installed” are assigned applicability value *applicable*.

In the first test case, when the concept “Frame Columns” was assigned *applicable* value, the whole Activity section including corresponding safety rules was evaluated to be *applicable* (applicable recall value is 1). Then, after the concept “Precast Concrete Columns” was assigned *applicable* value, the applicability exception for the JHA Activity section became satisfied. Thereby, the applicability value of those safety rules in the Activity section was negated and turned into *not applicable* (not-applicable recall
value is 1).

The second test case showed that the first Job Step section of the Activity section “Frame Columns” was evaluated to be not applicable due to the effect of the applicability exception, while other Job Step sections of the Activity section remained applicable since the concept “Frame Columns” was applicable.

In the third test case, I first neglected the concept “Frame Columns” to narrow down the reasoning scope to the Job Step sections only. Then, similar to the first test case, the applicability value of the Job Step section “Fly forms to area to be installed” changes from applicable to not applicable. This was because when Job Step concept “Manually move forms to area to be installed” gained its applicable value, it made the applicability exception satisfied and further made the Job Step section not applicable.

5.8.3 Discussion

These scenarios and the corresponding reasoning results are summarized in Table J.1 in Appendix J. The reasoning results are as expected and show that the safety requirement reasoning engine successfully identifies all applicable and not applicable safety rules (both applicable and not-applicable recall values are equal to 1) when applicability exceptions are involved in the JHA documents.

It should be noted that the applicability exceptions defined in construction safety documents are built upon their corresponding applicability conditions. In other words, no applicability exceptions can stand alone without applicability conditions. This notion is important because it represents the logic behind a
safety requirement: applicability exceptions aim to represent those situations which do not apply to ordinary situations represented by applicability conditions. From the reasoning perspective, this means that it is meaningless to only reason about applicability exceptions without reasoning about their corresponding applicability conditions. Take the first test case discussed in previous subsection for example. If users specify the concept “Precast Concrete Columns” to be applicable, the developed reasoning engine will not evaluate all the safety rules to be not applicable until the concept “Frame Columns” is made applicable.

According to these test cases, using applicability exceptions seems to be an alternative to defining disjoint relationship between concepts since both can achieve the purpose of identifying not applicable JHA sections. However, they cannot be lumped together due to their different semantics in reasoning. Disjoint relationship is used to indicate that when one JHA concept applies, the other must not apply; on the other hand, applicability exceptions do not have such kind of inference and when an applicability exception is satisfied, the whole JHA section having the exception becomes not applicable. Therefore, disjoint relationship is a concept-level element which is involved in the concept ontology reasoning engine while an applicability exception is a document-level or, specifically, safety requirement-level element which is involved in the safety requirement reasoning engine. For example, by specifying the concept “Precast Concrete Columns” to be applicable in the example of test case 1, the reasoning result only means that the whole JHA “Frame Columns” section will be exceptionally not applicable but does not mean that the Activity concept “Frame Columns” itself becomes not applicable.
5.9 APPLY THE DEVELOPED FRAMEWORK TO AN OSHA SAFETY DOCUMENT

In this research, I apply the developed Framework to the OSHA construction safety document 29 CFR 1926 (U.S. Department of Labor 2003) to further validate the applicability of the developed Framework. I use two Subparts of this safety document for validation: Subpart N Cranes, Derricks, Hoists, Elevators, and Conveyors and Subpart Q Concrete and Masonry Construction. In this section, I first discuss the representation of this document and the concepts extracted from this document, which is followed by a discussion on reasoning about the modeled documents and concepts. Lastly, I summarize this validation and discuss the difference between applying the developed Framework to JHA documents and to the OSHA construction safety documents.

5.9.1 Representation of The Document and Concepts

The OSHA safety document specifies the following hierarchy to structure the document: Part, Subpart, Standard, and Section. Safety requirements imposed by the OSHA document usually appear on the Section level, and each section can have multiple subsections. Part of the document is shown in Figure 3.9. To apply the textual document representation model, I use Part, Subpart, Standard and Section as the constituent elements for representing the OSHA document in XML format. I show the representation of the OSHA document of Figure 3.9 in XML format in Appendix K.

The OSHA document does not indicate a concept-grouping structure in the documents as JHA documents do (i.e. Activity, Job Step and Potential Hazard). Therefore, to use concept ontology
representation model representing concepts extracted from the OSHA construction safety document, any classification proper for organizing the concepts can be used and in this research, I adopt the CAR classification: Component, Action, and Resource discussed in section 2.5.1 as primary grouping concepts to structure the extracted OSHA concepts. For example, the primary grouping concept “Component” classifies the following concepts as its sub-concepts: “Column”, “Footing”, “Foundation”, “Framing”, “Panel”, “Pier”, “Slab”, and “Wall”.

Another aspect of using concept ontology representation model is to represent relationships between the OSHA concepts. Similar to concept representation, the OSHA document does not imply association relationships for connecting concepts, which is different from JHA documents that, for example, imply a “hasStep” relationship for Activity concepts to connect to Job Step concepts. Therefore, association relationships are flexibly defined for OSHA concepts in this validation according to the representation needs. For example, I define an association relationship “actOn” to connect Action concepts to Component concepts and also define “isComposedOf” to connect Component concepts to Material concepts, which is a sub-concept of the concept “Resource”. On the other hand, logical association relationships are also defined according to the representation needs. For example, two Equipment concepts “Elevator” and “Lift” can be defined as equivalent relationships because they both mean “a device that carries people up and down inside buildings”.

5.9.2 Reasoning about The Documents and Concepts

To test the applicability of the Reasoning Engine to the modeled OSHA document and concepts,
another prototype system Safety Advisor is developed. Safety Advisor is designed only to show concept ontology and modeled OSHA document and to implement the same Reasoning Engine as JHA Advisor does; hence, it does not have other functionality as JHA Advisor has, such as the concept and document editor, concept neglect mechanism and reasoning rationale feedback function.

I use Safety Advisor to perform several test cases, which are similar to those discussed in sections 5.3.2 and 5.4.2, to test the applicability of the Reasoning Engine works to identify applicable and not applicable safety requirements imposed by the OSHA document. The test results show that applicable and not applicable safety requirements can be correctly identified in the given testing scenarios. For example, by assigning applicable values to the concepts “Cast-In-Place”, “Concrete” and “Formwork”, the safety requirement 1926.703(a)(1), shown in Appendix K, becomes applicable.

It should be noted that the same representation and reasoning limitations of the Representation Model and Reasoning Engine discussed in the previous sections still applies to this validation since the developed Framework is directly deployed on the OSHA document in the validation.

5.9.3 Summary and Discussion

The validation result shows that the developed Framework can be applied to OSHA construction safety documents to identify applicable and not applicable safety requirements. While this validation is performed in a simpler scenario than the scenarios for the validations performed in the previous sections of this chapter, it still can show the flexibility of the developed Framework for representing and reasoning
about documents and concepts with different semantic features.

There are three major differences between applying the developed Framework to JHA documents and to the OSHA construction safety documents. First, representing applicability conditions for OSHA documents is more flexible than for JHA safety documents. As discussed in the previous sections, single-type and multi-type descriptions have to be considered for representing applicability conditions for JHA documents. For OSHA documents, representing applicability conditions only has to consider what context-describing concepts are in the documents and whether they are related semantically (i.e., conjunctively or disjunctively). For example, the applicability condition for the title of Subpart Q uses \(<\text{Concept\_OR/}>\) to represent the concepts “Concrete Construction” and “Masonry Construction” while that for the title of Standard 1926.736 uses \(<\text{Concept\_AND/}>\) to represent the concepts “Cast-In-Place” and “Concrete”, as shown in Appendix K.

Second, JHA documents imply a concept-grouping structure in the documents whereas OSHA documents do not, which makes it flexible to represent OSHA concepts. Third, different XML schemas need to be defined for JHA and OSHA documents since they have different constituent elements in their documents.

5.10 FEEDBACK FROM PROFESSIONALS

I presented the developed Construction Safety Document Management Framework and prototype system \textit{JHA Advisor} to safety professionals of the construction industry. I asked them questions about the
approach (including the prototype system), such as what the potential benefits can be brought by applying it, how well they are in understanding and using it, and what should be done to enhance it for practical application. Overall, they gave the research and the developed approach very positive feedback with a few comments on how to improve it in order to further strengthen its capabilities. I summarize their feedback in the following subsections.

5.10.1 Advantages of The Approach

The professionals pointed out two major advantages of the developed approach: separate representation of JHA concepts to benefit safety knowledge utilization and retrieval; and automated reasoning about JHA documents to facilitate identification of safety requirements. The details of the advantages are highlighted as follows.

5.10.1.1 Benefit safety knowledge utilization and retrieval

The professionals see two benefits from separately organizing JHA concepts using concept ontology representation model. First, it is much easier to understand what JHA concepts, i.e. activities, job steps and potential hazards, have previously been used and exist in their JHA document collections. Then when the professionals perform new JHAs, they can refer back to the construction safety knowledge base (i.e. the concept ontology) and use keywords to search the ontology for matching or similar concepts which are suitable for reuse or reference in the new JHA documents. Second, the professionals can easily find out other applicable concepts which are related to the search ones. These other concepts which are
potentially useful in the new JHA documents may not be considered in the first place. By using the concept ontology reasoning engine to reason about the relationships defined between concepts, those unattended concepts can less likely be ignored.

The professionals deem that both benefits can save their time for preparing new JHA documents in determining what concepts should be considered in new JHAs, such as what job steps are taken into account for a specific activity or what potential hazards are associated with a specific job step. If no appropriate concepts exist, they can follow the procedure of adding new concepts to define the concepts they need and they find that, as presented in the previous section, adding new concepts is a straightforward process in the developed approach.

5.10.1.2 Facilitate identification of safety requirements

Another advantage the professionals observe is that the developed approach can automatically identify applicable/not applicable safety requirements by assigning applicable/not applicable value to JHA concepts. The meaning of this advantage is threefold. First, the professionals consider the advantage can save time for preparing new JHA documents in determining applicable safety requirements. They can either directly reuse the identified safety requirements or revise them for their practical use. Also, by combining this and the previous advantage, a new JHA document can be easily generated. They believe that not only experienced safety professionals but also non-JHA experts can both benefit from the developed approach. For the former, the approach can provide JHA documents for their reference and support their decision making process; for the latter, the approach is easy to use and educationally is a
good starting point for them to involve in the JHA process.

In addition, the professionals think it an advantage to identify safety requirements automatically because when the contexts on site change, the approach allows them to quickly react to the changes by selecting different concepts which best describe the new contexts and generating new safety requirements for the contexts.

Lastly, the professionals find automated identification of safety requirements beneficial to their daily safety meeting which is held on site before workers start their work. They can use the approach to quickly identify necessary safety requirements applicable to the scheduled tasks and remind workers of these highlighted safety requirements and procedure.

5.10.2 Improvement of the Approach

In addition to the aforementioned benefits the developed approach can bring about, the professionals also gave a few suggestions on how the developed approach can be improved to strengthen its capabilities and make it more suitable for the needs of construction industry.

5.10.2.1 Integrate construction safety specifications into the developed approach

The professionals suggested that the developed approach can be further improved by integrating construction safety specifications, such as the OSHA safety regulation for construction safety (U.S.
Department of Labor 2003), with JHA knowledge to give engineers more information about safety requirements. Specifically, each safety requirement should have corresponding safety specifications to justify its necessity. They envision this upgrade can provide engineers more supportive safety information and benefit their safety planning and decision-making.

5.10.2.2 Add another grouping concept “Equipment” into the Representation Model

Three primary grouping concepts: “Activity”, “Job Step”, and “Potential Hazard” are used in the developed approach to organize JHA concepts. The professionals suggest adding another primary grouping concept “Equipment” in the concept ontology representation model and allowing Job Step concepts to either connect to Equipment concepts or connect to Potential Hazard concepts as has been done in this research. That is, adding the “Equipment” concept creates an alternative association route (Activity → Job Ste → Equipment → Potential Hazard) in addition to the original one presented in the research (Activity → Job Ste → Potential Hazard).

This new representation aims to represent the practice that a job step may involve different equipments to accomplish the same task and different equipments may associate different potential hazards with them. Therefore, by adding the new primary grouping concept, the professionals believe the developed approach can further reflect the reality and JHA concepts can be more finely represented.
5.10.2.3 Improve the usability of *JHA Advisor*

The professionals also suggest improving the usability of *JHA Advisor* by allowing users to retrieve the existing JHA information in JHA documents instead of inputting them manually, as already discussed in section 5.6.3.

5.10.3 Summary

The professionals’ positive feedbacks validate not only the motivation of this research but also the developed approach’s capabilities of reasoning about JHA concepts and JHA documents. Their feedback also proves the potential practicability of the approach in the construction industry. As for their suggestions of improving the approach, the first two of them will be taken into account in the future research, which is discussed in the next chapter, while the third one will have less priority than the others since it is only an implementation issue of the prototype system and does not harm the developed approach’s functionality.
CHAPTER 6: DISCUSSIONS AND CONCLUSIONS

6.1 INTRODUCTION

In this research, I have developed a Construction Safety Document Management Framework that enables automated identification of construction safety requirements for raising project participants’ awareness of construction safety issues. The developed Framework allows representing and reasoning about concept ontologies that structure construction safety concepts. The Framework also allows representing and reasoning about construction safety documents. By integrating the representation of and reasoning about concept ontologies and construction safety documents, construction safety requirements imposed by the documents can be identified and classified according to their applicability. The developed Framework was tested and validated for construction Job Hazard Analysis documents through different test cases.

In the following sections, I first describe the contributions and corresponding validations of this research. I then present this research’s practical implications and the identified future research opportunities.

6.2 SUMMARY OF CONTRIBUTIONS AND VALIDATIONS

The overall goal of this research is to enable automated reasoning about construction safety documents to improve the search for applicable safety requirements from the documents. To achieve this
objective, construction safety documents and contextual concepts in the documents that indicate safety requirements’ applicability conditions have to be represented in a computer readable and interpretable manner. The corresponding Representation Model of the developed Framework is developed for this purpose and hence is the first contribution of this research.

In addition, a reasoning mechanism that reasons about the represented context-describing concepts to evaluate their applicability needs to be developed. One part of the Reasoning Engine of the developed Framework, i.e. the concept ontology reasoning engine, addresses this need and constitutes the second contribution of this research.

In order to evaluate the represented construction safety documents and identify applicable and not applicable safety requirements from them, the represented documents also need to be reasoned about. The other part of the Reasoning Engine of the developed Framework, i.e. the safety requirement reasoning engine, satisfies this need and is the third contribution of this research.

The developed Framework was validated using JHA documents as the targeted construction safety documents. In the following subsections, I will discuss these contributions in more detail.

6.2.1 Representation Model for Representing Concepts and Construction Safety Documents

The Representation Model for representing concepts and construction safety documents is the first contribution of this research. The Representation Model is developed in response to the first two research
The principal goal of this research is to provide an approach that can benefit the search for safety requirements from construction safety documents to support construction safety planning. Safety requirements imposed by construction safety documents are good sources of safety knowledge and therefore, identifying applicable safety requirements is an essential solution to prevent accidents from occurrence. However, the large number of sources of safety requirements makes the identification tasks cumbersome and easy to overlook necessary safety requirements. Thus, one of the research goals is to develop a Representation Model that allows representing both construction safety documents and contextual concepts in a way that automatic reasoning about the documents and the concepts is enabled.

The Representation Model is the first part of the developed Framework and consists of two subparts: concept ontology representation model and textual document representation model. The two subparts provide systematic structures respectively for representing contextual concepts and representing construction safety documents in a computer readable and interpretable format, which enables the reasoning about the concepts as well as the documents.

The concept ontology representation model leverages OWL-based ontological modeling to formally structure contextual concepts. I elaborate the steps of using this developed representation model to create a concept ontology for concepts extracted from construction safety documents or for concepts describing
project contexts on sites. The major benefit of this representation model is that it is flexible so that it can represent concepts no matter what construction safety documents are focused on. In other words, the representation model can be used independent of documents and types of documents. The representation model’s flexibility results from two facts. First, users can create or use any classification with which they can represent the concepts in the most satisfied way, provided that the representation is reasonable in representing generalization-specialization relationships between concepts. Second, users can define any association and logical relationship between concepts based on the concepts’ semantics that users need to express in the ontology. Such flexibility is beneficial to engineers since different documents may comprise concepts with different semantic properties (e.g. Activity versus Job Step concepts) and forms (e.g. single versus combined concepts) and this result in different representation needs, such as diverse hierarchical structures and relationship definition.

Another advantage of the representation model is that it allows combining specific ontologies together into an integrated one. For example, if users feel that two separately created models, which were specifically for concepts from two different safety documents, should have certain relationships defined between their respective concepts, they can integrate them by representing the concepts of one ontology in new class hierarchies of the other ontology and then those necessary relationships can be established.

On the other hand, the textual document representation model leverages XML-based OHCO modeling technique to formally structure construction safety documents. Representing documents in XML format not only enables the documents to be electronically processed and interpreted but also allows document users to understand and author the documents straightforwardly. This representation
model also features its flexibility in representing different documents. Users can apply this representation model to different documents by analyzing them and defining suitable representation schemas for them, as illustrated in chapter 3. A defined schema specifies the important information of document representation, such as what the structure of a document is, how document constituent elements should be represented, and what descriptive markup tags need to be defined. Therefore, if users want to represent a new safety document, they need to define a new schema for the new document and such a change is independent of the representation of the concepts from the new documents and does not affect the later reasoning processes.

Additionally, a combined advantage of the two representation models is the convenience of defining applicability conditions and exceptions. The contexts for applicability conditions and exceptions in XML-based construction safety documents are described through concepts represented in a concept ontology (i.e. XML-based safety documents are indexed through concepts). The developed Representation Model allows users to directly search for suitable concepts from the concept ontology for the applicability conditions and exception in the represented documents. On the other hand, if users cannot find suitable ones in the ontology, they can add them into the ontology, which benefits reuse of these concepts in later document representation.

The developed concept ontology representation model and textual document representation model are validated for use in the JHA domain using the acquired JHA documents in multiple test cases. A validation shows that it is feasible to represent contextual concepts using the concept ontology representation model, no matter how diversified the concepts’ semantic properties are and what the
hierarchical depth of representation is required. Another validation also shows that flexibly representing construction safety documents in XML format is possible. The validation proves that by using conjunctive or disjunctive description of concepts, textual document representation model can be general or specific enough to represent applicability conditions of any complexity.

6.2.2 Reasoning Engine for Reasoning about Concept Ontologies

When developing the Framework, another research question which needs to be addressed is what reasoning mechanisms would be necessary to process the modeled concepts for identifying construction safety requirements. The goal is that when engineers decide certain concepts to be applicable to the project contexts, other related concepts can be automatically identified. Therefore, a relationship identification function needs to be considered that allows identifying all the relationships of a concept. In addition, an applicability propagation function also has to be developed in order to properly propagate one concept’s applicability value to others’.

The concept ontology reasoning engine of the Reasoning Engine is developed to incorporate the two functions by leveraging the semantically rich properties of a concept ontology. When any concept in the ontology is selected and assigned applicable or not applicable value, the reasoning engine first captures what association or logical relationships are involved and what other concepts are connected to through these relationships. Then, according to the reasoning principles defined in the research, the concept’s applicability value will be propagated to the connected concepts. Thereby, this reasoning engine can be deemed to be efficient in running reasoning processes because it only needs to reason about concepts
connected through the involved relationships and does not have to spend time evaluating other unconnected concepts.

Another major advantage of the concept ontology reasoning engine is that it is independent of the concept ontology representation model and can be applied to any concept ontology, no matter how users represent concepts in the ontology. This makes the reasoning engine flexible enough to support reasoning about concepts extracted from not only different safety documents but also documents in other application domains. In addition, the concept ontology reasoning engine features its mechanism of reasoning rationale feedback, which allows users to understand how reasoning results are generated and how certain concepts become applicable or not applicable.

The concept ontology reasoning engine is validated using the modeled JHA concepts extracted from the acquired JHA documents in three synthetic test cases. The validation shows that this reasoning engine is capable of correctly propagating applicability values between concepts and properly evaluating concepts’ applicability values in the given testing scenarios. The reasoning rationale feedback mechanism is also validated and the validation result showed this mechanism can correctly return reasoning rationale back to users.

6.2.3 Reasoning Engine for Reasoning about Construction Safety Documents

In addition to the concept ontology reasoning engine, the other reasoning engine, safety requirement reasoning engine, is developed in response to the fourth research question: “what reasoning mechanisms
would be necessary to process the modeled construction safety documents for identifying construction safety requirements?” Therefore, the goal of this reasoning engine is to support reasoning about construction safety documents modeled in XML format. Specifically, this reasoning engine is developed aiming to evaluate each safety requirement’s applicability by reasoning about its applicability condition and, if any, exception.

According to the reasoning rules developed in this research, reasoning about applicability conditions and exceptions requires first knowing the applicability of the concepts describing or indexing the conditions and exceptions. Hence, the safety requirement reasoning engine is designed to be automatically activated once the concept ontology reasoning process is finished and concept reasoning results are attained. This feature facilitates the application of the safety requirement reasoning engine, and users can just determine what concepts should be focused on during the reasoning processes.

Furthermore, the safety requirement reasoning engine features its concept neglect mechanism that can help users neglect concepts which they tend not to be reasoned about; ultimately, the concept neglect mechanism helps change the reasoning strategies for construction safety requirements or document sections which are fully or partially indexed through the ignored concepts. This feature benefits the overall safety requirement reasoning process because it makes reasoning about concepts more flexible and thereby allows producing more possible reasoning results.

In addition, the core of the safety requirement reasoning engine can be used to reason about any type of documents which are represented in XML format. However, if new XML schemas are defined for
different documents which are to be reasoned about, it becomes necessary to modify the prototype system which implements the safety requirement reasoning engine originally for a specific safety documents, such as the JHA documents in the validation.

The safety requirement reasoning engine is validated using the modeled JHA documents and JHA concepts in several test cases. The validation results show that applicable and, if any, not applicable recall values are equal to 1 in the test cases. That illustrated it is possible to use the safety requirement reasoning engine to correctly identify applicable and not applicable safety requirements from the modeled documents. The concept neglect mechanism is also validated and the validation result showed this mechanism functions as expected. Overall, the safety requirement reasoning engine is believed to be able to expand its application to different construction safety documents other than JHA documents.

6.3 PRACTICAL IMPLICATIONS

In this research, I develop a Framework that defines representation models for representing construction safety documents and their contextual concepts to support automated reasoning. The Framework also defines reasoning engines that allow automatically reasoning about the modeled concepts and documents to determine concepts’ applicability to given contexts and to identify applicable and not applicable construction safety requirements.

Construction knowledge management (KM) gradually gains the construction industry’s interests and the construction industry is interested in the potentials KM tools can provide. This fact inspires the
application of KM tools as their knowledge repository to construction projects. The ontological modeling and XML modeling techniques are two representatives of the KM tools; the developed Framework that utilizes the two representative techniques is one of the potentials. Thus, it is expected that practitioners will benefit from the automated reasoning about construction safety documents by utilizing the Framework to develop specific models of construction safety documents and key safety-related concepts. I expect that the industry will first start using the Framework for JHA documents. When tools for representing and reasoning about other types of safety documents or different construction documents become available, the Framework can be put into practice for these different areas or domains of uses.

With the developed a Framework, reasoning about construction safety documents can be automated, which is expected to affect a project in different phases. It is expected that identifying applicable safety requirements for given contexts can be faster than current practices. This enables engineers to reduce the time needed for project planning, such as preparing an estimate or a project schedule, where safety requirements have to be taken into account. It is also expected that the automated identification of applicable safety requirements can benefit design tasks since designers may be aware of some design and constructability issues that results from the identified applicable safety requirements.

The Framework will also allow automatically generating customized lists of safety requirements imposed by construction safety documents. Since the identified requirements are applicable to certain given contexts, the generated lists are deemed to be project- or specifically, task-specific. This feature can be beneficial for two reasons. For one, this can support current safety practices of highly relying on safety experts’ experience and knowledge and general safety checklists. For another, this allows raising
engineers’ awareness of requirements imposed by the construction safety documents, which usually need to be identified from a large number of document collections in a project. By providing engineers and inspectors with a compilation of project- or task-specific safety requirements, they are able to better focus their attention on only relevant safety requirements.

In sum, from the perspective of document generation, the Framework can help engineers quickly prepare new construction safety documents, such as JHA documents. From the perspective of specific knowledge retrieval, the Framework is capable of generating lists of safety requirements applicable to specific tasks or project contexts. With these dual effects, the Framework is believed to be able to benefit safety practices in the construction industry. By applying the developed Framework, it is also believed to allow a better construction information and knowledge management in different disciplines or domains in the construction industry.

6.4 FUTURE RESEARCH

To keep the scope of this research manageable, several important aspects could not be addressed in this dissertation. The following subsections describe these aspects and point out future research directions identified by this research.
6.4.1 Test Applicability to Other Construction Documents

The developed Framework is tested for JHA documents and the OSHA’s safety regulation for construction safety, 29 CFR 1926 (U.S. Department of Labor 2003). While this Framework is believed to be applied to different construction documents of safety or non-safety areas due to its modeling flexibility as discussed in section 6.2, it would be best to test the Framework by deploying it in other test cases involving other construction documents.

As for being applied to other construction safety documents, the Framework can be tested for applicability to proprietary construction safety manuals, i.e. companies’ own safety regulations. While this type of documents is usually developed on the basis of OSHA construction safety standard, they may also incorporate safety requirements for specific contexts that rarely occur but are learned from the companies’ past experiences.

In addition to these safety documents, the Framework can also be tested for applicability to construction documents of other areas or purposes, such as construction specifications, construction manuals, meeting minutes, and contracts. Testing the different, specific knowledge should be able to further expand the range of application of the developed Framework.

6.4.2 Test Applicability to Different Forms of Construction Information

Although the Framework is developed and tested for textual documents, I believe it can also be
applied to construction information in other forms rather than text, such as images, films or sounds. Specifically, the concept ontology representation model is flexible enough to model concepts for construction information in these forms. For example, engineers can determine proper concepts for describing the content of pictures taken on site and then use the representation model to model the concepts in an ontology. On the other hand, since XML is capable of comprising multimedia information, the textual document representation model is believed to be able to extend its use to model construction information in different forms and reuse the modeled concepts to index it.

Further study on this topic should focus on what the differences will be between using the Framework for text and for other forms of information and how the Framework should be modified to address, if any, these differences.

6.4.3 Automate Resolution of Concept Applicability Contradiction

The Framework can mainly handle combined concepts whose constituent concepts are respectively from different concept types. For example, the concept “Frame Columns” involves an Action constituent concept “Frame” and a Component constituent concept “Column”. When the concept ontology reasoning engine is deployed to concepts which aim to describe multiple same-type constituent concepts at once, it may lead to the constituent concepts that semantically contradict one another. For instance, given a scenario that a precast column is set up on a cast-in-place foundation, the concept “Precast Concrete Column” is contradictory to the concept “Cast-In-Place Foundation” due to the defined disjointedness between their constituent concepts “Precast” and “Cast-In-Place”. When the first concept has been
processed by the reasoning engine, concept applicability contradiction occurs once the second one is input into the reasoning process. When such contradictions take place, the developed Framework will ask users to manually determine whether to accept or reject the applicability of the contradicting concepts as I discussed in section 4.2.3.

To increase the reasoning efficiency and capability, further research on how to address and solve concept applicability contradictions in an automated manner in the developed Framework is necessary. A potential solution, for example, is to consider the propagation of applicabilities under conditions. That is, users can set applicability propagation rules in advance for different possible conditions in which concept applicability contradictions take place; when contradictions occur, the Framework can identify which conditions corresponding to the contradictions are satisfied and then determine which preset rules should be used to continue the reasoning process.

6.4.4 Integrate Construction Safety Specifications with JHA Knowledge

As the professionals from the construction industry point out, they can see benefits of integrating construction safety specifications, such as the aforementioned OSHA safety regulations, with JHA knowledge. The integration would allow providing supportive and justifiable information to users about why certain safety requirements are important and necessary. Therefore, studying on such integration in future research would be helpful to further advantage practitioners of the industry.

Such integration may include following key tasks: Use concept ontology representation model to
model contextual concepts extracted from construction safety specifications in an ontology where JHA concepts are modeled; use textual document representation model to model the specifications, including defining proper XML schema for the specifications; develop a mechanism which allows attaching safety regulations of the specifications to corresponding related safety requirements of JHA documents; and modify the prototype system in order to show both JHA knowledge and regulations of the specifications. The most important task in this future research to be studied is to understand the relationships between specification concepts and JHA concepts and how to reflect these relationships in the concept ontology representation model.

6.4.5 Eliminate Redundancies of Identified Safety Requirements

While applicable safety requirements can be successfully identified using the developed Framework, redundancies of the identified applicable safety requirements currently are not automatically removed from the identified safety requirement collection. In other words, it may be possible that several identical or very similar safety requirements which are applicable to the given contexts exist together in a list of applicable safety requirements.

Although this does not harm the Framework’s functionality, it indeed will lessen the usability or readability of the identified applicable safety requirements. Thus, it would be good to have a mechanism which can eliminate such kind of redundancies. Future research on how to achieve the elimination of applicable safety requirements’ redundancies without accidentally removing non-redundant information should be conducted.
6.4.6 Evaluate Relevance of Applicable Safety Requirements

The developed Framework is validated to be able to correctly identify applicable or not applicable safety requirements. However, when the number of identified applicable safety requirements becomes much larger, especially when users provide much input information, going through all of them may become a tedious work. Thus, it would be better if the Framework can provide a mechanism which is able to further evaluate the identified applicable safety requirements’ relevance to certain contexts and to rank these requirements according to their relevance.

The identified safety requirements of applicability value applicable mean that they have already been evaluated to be applicable, or completely relevant, to the given contexts. In other words, to further evaluate the relevance of the applicable requirements may require different evaluation contexts from the contexts used in the previous reasoning processes. For example, if the contexts for the reasoning processes are described through concept combinations, the constituent concepts of the concept combinations may be eligible for the relevance evaluation.

Several approaches for evaluating relevance between documents and queries have already been developed and widely used in the text retrieval domain, such as the vector space model (Salton et al. 1975) and classical probability model (Robertson and Sparck Jones 1976). These approaches are good starting points to study the feasibility of evaluating the applicable safety requirements’ relevance. With this evaluation mechanism, the developed Framework should be able to refine the automated identification of safety requirements and to provide more precise information to users, which can benefit their
decision-making processes.

6.4.7 Evaluate Possibly Applicable Safety Requirements

The developed Framework can identify *applicable* and *not applicable* safety requirements. Other many safety requirements, on the other hand, are left *possibly applicable* as no contextual information which relates to these requirements is available. These *possibly applicable* requirements will not get a chance to be re-evaluated until there is new contextual information being input into the model.

Rather than leaving these *possibly applicable* safety requirements in the model until new contextual information being available, it would be more beneficial to make them informative and then users can make most of them, e.g. knowing how possible a safety requirement can be applicable given current contexts and making some decisions accordingly. To achieve this goal, it is necessary to quantitatively describe the notion of “possibly applicable” in some way as the notion is now qualitative and abstract so that it is difficult to be further utilized by users. Therefore, study on this topic should be conducted in order to answer two main questions: How can the notion of “possibly applicable” be quantified and what measures are needed to best describe the level of possibilities?

To answer the questions, the text retrieval domain in the Computer Science is one of the promising areas worth researching in. Specifically, statistical language models, approaches developed in the text retrieval domain, have potential for providing solutions to the questions. A statistical language model (or language model for short) is defined as “a function that puts a probability measure over strings drawn
from some vocabulary” (Christopher D. Manning et al. 2008). Zhai (2009) claims that a language model is useful because it provides a principled way to quantify the uncertainties associated with the use of natural language. Thus, this is a good future research direction to look into in order to reinforce the developed Framework for evaluating possibly applicable safety requirements.

6.4.8 Integrate the Framework with Project Models

The developed Framework is tested and validated in an office setting in which contexts are selected manually. In addition, I discuss to use the developed concept ontology reasoning engine with an reality capture technology, radio frequency identification (RFID), which plays the role of identifying contextual information on site (Wang et al. 2009). To fully automate the identification of construction safety requirements, integrating the Framework with project models prepared through building information modeling technique is a promising way and worth being researched further.

Such integration can be illustrated from two different perspectives. First, project models can provide contextual information on sites as input into the Framework. Then the Framework knows what concepts should become applicable to correspond to the current contexts and further determine applicable safety requirements for the contexts. Therefore, project models in this sense act as the knowledge base of a project, such as information of processes and products, which shall benefit users’ faster determination of the target for which the Framework is used.

Second, while project models allow relationships defined between entities, such as processes and
products, the number and semantics of these kinds of relationships in a project modeling language are limited, as compared with relationships which can be defined through ontological modeling techniques. Therefore, the developed concept ontology reasoning engine can be used as an external reasoning tool helping perform more diverse semantic reasoning which cannot be done in the project models. With the external reasoning done, the concept reasoning results may be feed back to project models to extract information stored in them which is necessary for the deduced concepts.

6.4.9 Allow Evaluating Property Values of Concepts

In this research, the Framework is developed leveraging contexts’ semantically rich feature and thus, focuses on the semantics between concepts. In other words, I do not take into account the properties and property values which concepts may have textually in the safety documents. For example, a concept “Working at Height over 6 Feet” as a whole currently is viewed as contextual information but its property “Height” and corresponding property value “over 6 Feet” are not considered in the representation and reasoning.

Properties and their values can be regarded as another type of semantics, which belongs to concepts themselves and does not exist between concepts. Although ignoring this kind of semantics in the Framework does not harm the Framework’s representation and reasoning capability, it indeed, to some degree, prevent users from fully exploiting the semantics concepts may have. Therefore, taking properties and their values into consideration in the Framework would enable more profound reasoning about safety documents.
To allow evaluating property values in the Framework, the following questions have to be address:

What representation do I need to model properties? Can we build such representation upon the current representation model? Is the current reasoning engine capable of reasoning about property values? If not, what modifications of the reasoning engine are needed for the new reasoning, or what new reasoning engine has to be developed?

6.4.10 Formalize Reasoning Rules and Principles with Formal Logic

In chapter 4, I define the reasoning rules and principles for the developed Reasoning Engine. These definitions are given and discussed computationally; however, they are not represented in a formalized way, such as represented in formal logics. Although lack of such formalization of the rules and principles does not harm the Reasoning Engine’s reasoning capabilities, formalizing the rules and principles in formal logics, such as first order logic or description logic, may leverage the logics’ inference rules and then allow exploring the potentials of the defined rules and principles. For example, if one is certain that a concept “Precast Concrete” in the concept ontology is not applicable, current concept ontology reasoning principles will do nothing when this concept is specified to be not applicable. If this concept is viewed as a concept combination of two single concepts “Precast” and “Concrete” with a logical operator “∧” in between (i.e. “Precast ∧ Concrete”) and De Morgan’s laws\(^1\) of formal logic is deployed, I can infer that either the concept “Precast” or the concept “Concrete” or both are not applicable. This new information may be useful for further reasoning about concepts and safety requirements.

\(^1\) Suppose P and Q are two propositions. De Morgan’s law regulates that \(\neg(P \land Q)\) is equal to \(\neg(P) \lor \neg(Q)\) or that \(\neg(P \lor Q)\) is equal to \(\neg(P) \land \neg(Q)\).
In addition, another aspect of this future research direction is to evaluate the computational complexity, decidability and completeness of the concept ontology reasoning engine. The goal is to prove the concept ontology reasoning engine’s applicability in large-scale reasoning scenarios, as discussed in section 5.3.2.2. By representing the proposed reasoning principles of the concept ontology reasoning engine in formal logics, mathematical methodology can be used to evaluate the issues of computational complexity, decidability and completeness.

The topics which future research on this direction should look into include: study different kinds of formal logics and understand their application areas, strength and weaknesses; test whether the formal logics can properly work with the developed Reasoning Engine and determine the most appropriate one; and identify the needs and requirements for involving the formal logic in the developed Framework. I believe that the developed reasoning mechanisms can benefit from introducing formal logics and their inference rules into the Reasoning Engine.

6.4.11 Summary

The aforementioned research directions are those which seem most promising and interesting when this research is performed. Some of them describe different directions to which the developed Framework can be applied; the others describe the potential improvements the Framework can get in future research. Also, the whole Framework or its parts, i.e. the two representation models or reasoning engines, can be applied to any direction for any purpose as long as the developed approach can contribute to the direction. For example, the concept ontology reasoning engine offers a reasoning mechanism that can reason about
the semantics of construction contextual information in different forms, which no other mechanisms can
reason about. I am confident that this research presents a useful approach that has much potential for
improving and contributing to the construction industry.
REFERENCES


Fenves, S. J., Gaylord, E. H., and Goel, S. K. (1969). Decision Table Formulation of the 1969 AISC Specification, Department of Civil Engineering, University of Illinois at Urbana-Champaign, Urbana-Champaign, IL, USA.


APPENDIX A: ILLUSTRATION ON BASICS STEPS OF USING JHA ADVISOR

Step 1: JHA Advisor loads the concept ontology

a. Click “File” and then “Load JHA Concept (OWL) File” to open the “Open OWL File” Dialog

b. Select the concept ontology file to be loaded
c. *JHA Advisor* after reading in the concept ontology

Step 2: *JHA Advisor* loads the modeled JHA documents

a. Click “File” and then “Load JHA Document (XML) File” to open the “Open XML File” Dialog
b. Select the JHA document file to be loaded

c. *JHA Advisor* after reading in the modeled JHA documents

---

**Step 3: Specify applicability value for concepts**

a. Click to select concept(s) from the concept ontology in *JHA Advisor* (A1_Frame_Column is selected in the example); right click to open the “Applicability Assignment” menu for the
b. Choose one of the three applicability values for the selected concept(s) ("APPLICABLE" is chose in this example)

Step 4: Applicability propagation among the concepts as well as the safety requirements

The remaining concepts of the concept ontology are automatically reasoned about by JHA Advisor to determine their applicability. The applicability is then propagated to the safety
requirements (Applicable concepts and safety requirements are shown “Green” in *JHA Advisor* and possibly applicable ones are “Blue”)

Step 5: Output of reasoning results

a. Click “File” and then “Save Result As Excel File” to open the “Save As Excel File” Dialog
b. Specify the file name and where to save the file

c. *JHA Advisor* output the reasoning results as an MS Excel file that has different sheets for safety requirements with different applicability values.
### APPENDIX B: TEST CASES OF REPRESENTATION OF THE EXTRACTED CONCEPTS

Table B.1: Representations of concepts under primary grouping concept “Activity”

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Extracted Concept to be Modeled</th>
<th>Secondary Grouping Concept</th>
<th>Hierarchical Representation Category</th>
<th>Associated Job Step Concept</th>
<th>Disjoint Concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Frame Columns</td>
<td>Concrete Activity</td>
<td>A1</td>
<td>Fly Forms To Area To Be Installed, Take Forms Off Cart Or Blocking, Stand Forms Into Place, Set Pins</td>
<td>—</td>
</tr>
<tr>
<td>2</td>
<td>Excavation Using Support Systems</td>
<td>Excavation Activity</td>
<td>A1</td>
<td>Excavation Using Shoring Manufactured Trench Boxes Or Other Support Systems</td>
<td>Yes³</td>
</tr>
<tr>
<td>3</td>
<td>Working On Or Near Energized Circuit</td>
<td>Electrical Activity</td>
<td>A1</td>
<td>Working On Or Near Energized Circuit</td>
<td>—</td>
</tr>
<tr>
<td>4</td>
<td>Forklift Use</td>
<td>Equipment Activity</td>
<td>A1</td>
<td>Steps Of Forklift Use</td>
<td>—</td>
</tr>
<tr>
<td>5</td>
<td>Carcinogen Control</td>
<td>Existing Condition Activity</td>
<td>A1</td>
<td>Carcinogen Control Operation</td>
<td>—</td>
</tr>
<tr>
<td>6</td>
<td>Spray Painting</td>
<td>Finish Activity</td>
<td>A1</td>
<td>Spray Application Of Paint To Walls And Ceiling, Working With Elevated Heights</td>
<td>—</td>
</tr>
<tr>
<td>7</td>
<td>Disassembly Of Material And Equipment</td>
<td>General Activity</td>
<td>A1</td>
<td>Disassembly And Cutting Of Fixed Equipment, Disassembly Of Material And Equipment</td>
<td>—</td>
</tr>
<tr>
<td>8</td>
<td>Fibrous Insulation And Refractory Ceramic Fiber</td>
<td>HVAC Activity</td>
<td>A1</td>
<td>Fibrous Insulation And Refractory Ceramic Fiber Operation</td>
<td>—</td>
</tr>
<tr>
<td>9</td>
<td>Masonry Construction</td>
<td>Masonry Activity</td>
<td>A1</td>
<td>Masonry Construction Operation</td>
<td>—</td>
</tr>
<tr>
<td>10</td>
<td>Welding Operation</td>
<td>Mechanical Activity</td>
<td>A1</td>
<td>Welding</td>
<td>—</td>
</tr>
<tr>
<td>11</td>
<td>Install Process Piping</td>
<td>Plumbing Activity</td>
<td>A1</td>
<td>Layout Of Pipe, Moving Piping Into Position, Install Hangers, Install Pipe And Hangers, Solder Field Joints, Test Piping</td>
<td>—</td>
</tr>
<tr>
<td>12</td>
<td>Steel Erection Operation</td>
<td>Structural Activity</td>
<td>A1</td>
<td>Steel Erection</td>
<td>—</td>
</tr>
</tbody>
</table>

¹**Associated Concept** lists those Job Step-type concepts connected to the represented concept through association relationship “hasStep”.  
²**Disjoint Concept** lists the Activity-type concepts connected to the represented concept through disjoint relationships respectively.  
³The disjoint concept is “Excavation Using Sloping”.
<table>
<thead>
<tr>
<th>Test Case</th>
<th>Extracted Concept to be Modeled</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Hierarchical Representation Category</th>
<th>Associated Potential Hazard Concept¹</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Accessing Mobile Scaffold</td>
<td>Access Step</td>
<td>-</td>
<td>A1</td>
<td>Fall</td>
</tr>
<tr>
<td>2</td>
<td>Manually Adjust Tool To Desired Cutting Depth</td>
<td>Adjust Step</td>
<td>-</td>
<td>A1</td>
<td>Projectile</td>
</tr>
<tr>
<td>3</td>
<td>Floor Sealant Application</td>
<td>Apply Step</td>
<td>-</td>
<td>A1</td>
<td>Fatigue, Airborne Exposure To Epoxy, Ignition Source During Operation, Skin Exposure, Heat Exposure</td>
</tr>
<tr>
<td>4</td>
<td>Engage Gear To Start Material Stock Turning</td>
<td>Associate Step</td>
<td>-</td>
<td>A1</td>
<td>Projectile, Scalping, Amputation, Noise, Eye Injury</td>
</tr>
<tr>
<td>5</td>
<td>Check Dimension With Proper Machining Tools</td>
<td>Check Step</td>
<td>Check Dimension</td>
<td>C1</td>
<td>Exposure To Sharp Or Abrasive Materials</td>
</tr>
<tr>
<td>6</td>
<td>Clean Up Work Areas</td>
<td>Clean Step</td>
<td>-</td>
<td>A1</td>
<td>Eye Injury, Fall Due To Wet Surfaces, Slip Due To Wet Surfaces, Dropping Materials Or Equipment, Trip, Exposure To Dust, Exposure To Sharp Or Abrasive Materials, Manually Lifting, Burns</td>
</tr>
<tr>
<td>7</td>
<td>Clear Path To Destination Of Object</td>
<td>Clear Step</td>
<td>-</td>
<td>A1</td>
<td>Eye Injury, Back Injury, Fall, Trip, Slip</td>
</tr>
<tr>
<td>8</td>
<td>Fabrication Of Containment In Staging Area</td>
<td>Fabrication Step</td>
<td>Fabrication Of Containment</td>
<td>C1</td>
<td>Electric Shock, Hand And Power Tool Injuries</td>
</tr>
<tr>
<td>9</td>
<td>Cutting Off Tube</td>
<td>Cut Step</td>
<td>-</td>
<td>A1</td>
<td>Sharp Edges, Cutting Hands</td>
</tr>
<tr>
<td>10</td>
<td>Disassembly Of Piping And Other Overhead Items</td>
<td>Disassembly Step</td>
<td>-</td>
<td>A1</td>
<td>Injury Occurring During The Use Of Scissor Lift, Injury Occurring During The Use Of Ladder, Personnel Falling While Working At Elevated Heights, Shifting Of Unstable Items Causing Injuries To Personnel</td>
</tr>
<tr>
<td>11</td>
<td>Disconnect Pressure Supply Line From Test Assembly</td>
<td>Disconnect Step</td>
<td>-</td>
<td>A1</td>
<td>Breakage Of Existing Lines</td>
</tr>
</tbody>
</table>
Table B.2: Representations of concepts under primary grouping concept “Job Step” (cont.)

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Extracted Concept to be Modeled</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Hierarchical Representation Category</th>
<th>Associated Potential Hazard Concept¹</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>Drill And Secure Anchors And Hangers</td>
<td>Drill Step</td>
<td></td>
<td>A1</td>
<td>Inhalation Of Dust, Flying Debris, Falling Anchors And Piping Striking Personnel Around Lift, Falling Anchors And Piping Damaging Surrounding Equipment</td>
</tr>
<tr>
<td>15</td>
<td>Lead Exposure During Construction Activities</td>
<td>Exposure Step</td>
<td></td>
<td>A1</td>
<td>Inadvertent Exposure To Lead, Exposure To Lead During The Lead Assessment Period, Inadequate Methods Of Compliance, Employee Not Provided With Protective Clothing, Poor Housekeeping</td>
</tr>
<tr>
<td>16</td>
<td>Gather Materials Necessary For Job Task</td>
<td>Gather Step</td>
<td></td>
<td>A1</td>
<td>Trip, Compressed Gas Cylinder Rupture, Gas Cylinder Fall, Eye Injury, Noise, Dropping Materials Or Equipment, Projectile, Exposure To Sharp Or Abrasive Materials, Manually Lifting</td>
</tr>
<tr>
<td>17</td>
<td>Inspect Chains Straps And Hooks For Deficiencies</td>
<td>Inspect step</td>
<td></td>
<td>A1</td>
<td>Hand Injury, Eye Injury</td>
</tr>
<tr>
<td>18</td>
<td>Install Piping Overhead In Lateral Racks</td>
<td>Install Step</td>
<td></td>
<td>A1</td>
<td>Body Injury, Fall, Trip, Dropping Materials Or Equipment</td>
</tr>
<tr>
<td>19</td>
<td>Ignite Plasma Stream</td>
<td>Ignite Step</td>
<td></td>
<td>A1</td>
<td>Burns, Fire, Eye Injury</td>
</tr>
<tr>
<td>20</td>
<td>Loading Concrete Bucket</td>
<td>Load Step</td>
<td></td>
<td>A1</td>
<td>Contact With Bucket</td>
</tr>
<tr>
<td>21</td>
<td>Break Forms Loose</td>
<td>Loose Step</td>
<td></td>
<td>A1</td>
<td>Sprain, Strain, Unexpected Form Release</td>
</tr>
<tr>
<td>22</td>
<td>Lower Cable Hook To Attach Chains Straps And Hooks</td>
<td>Lower Step</td>
<td></td>
<td>A1</td>
<td>Eye Injury, Hand Injury, Head Injury, Foot Injury, Falling Objects</td>
</tr>
<tr>
<td>23</td>
<td>Movement Of Equipment And Materials</td>
<td>Move Step</td>
<td></td>
<td>A1</td>
<td>Dropped Loads During Use Of Forklift And Other Lifting Devices, Manually Lifting</td>
</tr>
<tr>
<td>Test Case</td>
<td>Extracted Concept to be Modeled</td>
<td>Secondary Grouping Concept</td>
<td>Tertiary Grouping Concept</td>
<td>Hierarchical Representation Category</td>
<td>Associated Potential Hazard Concept¹</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------------------------</td>
<td>----------------------------</td>
<td>---------------------------</td>
<td>-------------------------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>24</td>
<td>Acquire Job Task Materials And Hardware</td>
<td>Obtain Step</td>
<td>—</td>
<td>A1</td>
<td>Exposure To Sharp Or Abrasive Materials, Eye Injury, Manually Lifting, Trip</td>
</tr>
<tr>
<td>25</td>
<td>Bending Large Spools 3/4th Inch Or Less</td>
<td>Other Operation Step</td>
<td>—</td>
<td>A1</td>
<td>Hitting Building Component</td>
</tr>
<tr>
<td>26</td>
<td>Perform Task</td>
<td>Perform Step</td>
<td>—</td>
<td>A1</td>
<td>Burns, Noise, Eye Injury, Sparks, Exposure To Dust, Unguarded Physical Contact</td>
</tr>
<tr>
<td>27</td>
<td>Planning</td>
<td>Plan Step</td>
<td>—</td>
<td>A1</td>
<td>Miscommunication, Trip, Slip, Bracing</td>
</tr>
<tr>
<td>28</td>
<td>Position Material Stock On Table</td>
<td>Position Step</td>
<td>Position Material Stock</td>
<td>C1</td>
<td>Manually Lifting, Dropping Material, Exposure To Sharp Or Abrasive Materials</td>
</tr>
<tr>
<td>29</td>
<td>Pour Columns</td>
<td>Pour Step</td>
<td>—</td>
<td>A1</td>
<td>Fall, Trip, Slip, Contact With Bucket, Electrical, Eye Injury, Dermatitis</td>
</tr>
<tr>
<td>30</td>
<td>Prepare Area For Scissor Or Aerial Lift Travel</td>
<td>Prepare Step</td>
<td>—</td>
<td>A1</td>
<td>Fall, Trip, Slip, Personnel Falling From Lift</td>
</tr>
<tr>
<td>31</td>
<td>Pressurize Complete Torch System For Leak Test</td>
<td>Pressure Step</td>
<td>—</td>
<td>A1</td>
<td>Fire, Flying Debris, Compressed Gas Cylinder Rupture, Projectile</td>
</tr>
<tr>
<td>32</td>
<td>Raise Chuck Guard</td>
<td>Raise Step</td>
<td>—</td>
<td>A1</td>
<td>Exposure To Sharp Or Abrasive Materials</td>
</tr>
<tr>
<td>33</td>
<td>Remove Material Stock Using Overhead Crane</td>
<td>Remove Step</td>
<td>Remove Material Stock</td>
<td>C1</td>
<td>—²</td>
</tr>
<tr>
<td>34</td>
<td>Replace Object To Original Place Of Rest</td>
<td>Return Step</td>
<td>—</td>
<td>A1</td>
<td>Foot Injury, Head Injury, Hand Injury, Eye Injury</td>
</tr>
<tr>
<td>35</td>
<td>Detailling Of Pipe For Installation</td>
<td>Detailing Step</td>
<td>—</td>
<td>A1</td>
<td>Shutting Off Or Breaking Unprotected Valve Gauge Or EMO Buttons</td>
</tr>
<tr>
<td>36</td>
<td>Setting Forms</td>
<td>Set Step</td>
<td>—</td>
<td>A1</td>
<td>Fall, Sharp Edges, Struck By, Stuck Between</td>
</tr>
<tr>
<td>37</td>
<td>Storage Of Prefab Spools</td>
<td>Store Step</td>
<td>—</td>
<td>A1</td>
<td>Materials Falling From Racks</td>
</tr>
<tr>
<td>38</td>
<td>Test Piping</td>
<td>Test Step</td>
<td>—</td>
<td>A1</td>
<td>Air Introduced Into New System</td>
</tr>
</tbody>
</table>
Table B.2: Representations of concepts under primary grouping concept “Job Step” (cont.)

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Extracted Concept to be Modeled</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Hierarchical Representation Category</th>
<th>Associated Potential Hazard Concept&lt;sup&gt;1&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>Transporting Tanks</td>
<td>Transport Step</td>
<td></td>
<td>A1</td>
<td>Contact With Contaminants, Heat Exposure, Spread Of Contamination From Inside Tanks, Shifting Of Unstable Items Causing Injuries To Personnel, Impact To Charged Oxygen Line In Work Area</td>
</tr>
<tr>
<td>40</td>
<td>Turn Off Power At Circuit Breaker</td>
<td>Turn Off Step</td>
<td>Turn Off Power</td>
<td>C1</td>
<td>Electrocution</td>
</tr>
<tr>
<td>41</td>
<td>Turn On Power At Circuit Breaker</td>
<td>Turn On Step</td>
<td>Turn On Power</td>
<td>C1</td>
<td>Electrocution</td>
</tr>
<tr>
<td>42</td>
<td>Steps Of Using Mobile Scaffold</td>
<td>Use Step</td>
<td>Steps Of Scaffold Use</td>
<td>C1</td>
<td>Fall, Slip, Trip</td>
</tr>
<tr>
<td>43</td>
<td>Welding Of Tube</td>
<td>Weld Step</td>
<td>Welding</td>
<td>C1</td>
<td>Burns</td>
</tr>
</tbody>
</table>

<sup>1</sup>“Associated Concept” lists those Potential Hazard-type concepts connected to the represented concept through association relationship “hasHazard”.

<sup>2</sup>No Potential Hazard concepts are specified in the original JHA document.
Table B.3: Representations of concepts under primary grouping concept “Potential Hazard”

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Extracted Concept to be Modeled</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Hierarchical Representation Category</th>
<th>Equivalent Concept</th>
<th>Disjoint Concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Strain</td>
<td>Hazard From Bodily Reaction And Exertion</td>
<td>Bodily Reaction</td>
<td>A2</td>
<td>Sprain</td>
<td>—</td>
</tr>
<tr>
<td>2</td>
<td>Manually Lifting</td>
<td></td>
<td>Overexertion</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>3</td>
<td>Caught Between</td>
<td></td>
<td>Caught In Or Compressed By Equipment Or Objects</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>4</td>
<td>Collapse</td>
<td>Hazard From Contact With Objects And Equipment</td>
<td>Caught In Or Crushed In Collapsing Materials</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>5</td>
<td>Abrasion</td>
<td></td>
<td>Rubbed Or Abraded By Friction Or Pressure</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>6</td>
<td>Struck By</td>
<td></td>
<td>Struck By Object</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>7</td>
<td>Injury To Personnel Due To Hydraulic Jack Crushing</td>
<td></td>
<td>Other Contact With Objects And Equipment</td>
<td>A2</td>
<td>—</td>
<td>Yes¹</td>
</tr>
<tr>
<td>8</td>
<td>Impact To Charged Oxygen Line In Work Area</td>
<td></td>
<td>Contact With Electric Current</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>9</td>
<td>Burns</td>
<td></td>
<td>Contact With Temperature Extremes</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>10</td>
<td>Pressure Release</td>
<td>Hazard From Exposure To Harmful Substances Or Environments</td>
<td>Exposure To Air Pressure Changes</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>11</td>
<td>Exposure To Encapsulate Spray</td>
<td></td>
<td>Exposure To Caustic Noxious Or Allergenic Substances</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>12</td>
<td>Noise</td>
<td></td>
<td>Exposure To Noise</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>13</td>
<td>Exposure To Radiological Contaminants</td>
<td></td>
<td>Exposure To Radiation</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>14</td>
<td>Asphyxiation</td>
<td></td>
<td>Oxygen Deficiency</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>15</td>
<td>Debris</td>
<td></td>
<td>Other Exposure To Harmful Substances Or Environments</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>16</td>
<td>Fall Due To Poly Moving</td>
<td>Hazard From Falls</td>
<td>Fall On Same Level</td>
<td>A2</td>
<td>—</td>
<td>Yes²</td>
</tr>
<tr>
<td>17</td>
<td>Fall Into Excavitations</td>
<td></td>
<td>Fall To Lower Level</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>
Table B.3: Representations of concepts under primary grouping concept “Potential Hazard” (cont.)

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Extracted Concept to be Modeled</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Hierarchical Representation Category</th>
<th>Equivalent Concept</th>
<th>Disjoint Concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>Fall</td>
<td></td>
<td>Other Fall Hazard</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>19</td>
<td>Gas System Rupture</td>
<td>Hazard From Fires Or Explosions</td>
<td>Explosion</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>20</td>
<td>Fire Involving Beryllium Material</td>
<td></td>
<td>Fire</td>
<td>A2</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>21</td>
<td>Unforeseen Problems During A Lift</td>
<td></td>
<td>Action Event Or Exposure</td>
<td>B1</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>22</td>
<td>Damage To Equipment</td>
<td></td>
<td>Equipment Event Or Exposure</td>
<td>B1</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>23</td>
<td>Material Dislodgement</td>
<td></td>
<td>Material Event Or Exposure</td>
<td>B1</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>24</td>
<td>Untrained Personnel Working On Or Near Energized Circuits</td>
<td>Hazard From Other Events Or Exposures</td>
<td>Personnel Event Or Exposure</td>
<td>B1</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>25</td>
<td>Limited Access And Egress</td>
<td></td>
<td>Space Event Or Exposure</td>
<td>B1</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>26</td>
<td>Masonry Wall Falling Or Being Blown Over</td>
<td></td>
<td>Structure Event Or Exposure</td>
<td>B1</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>27</td>
<td>Air Introduced Into New System</td>
<td></td>
<td>System Event Or Exposure</td>
<td>B1</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>28</td>
<td>Damage To Vehicle</td>
<td></td>
<td>Vehicle Event Or Exposure</td>
<td>B1</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>29</td>
<td>Vehicle Accident</td>
<td>Hazard From Transportation Accidents</td>
<td>—</td>
<td>A1</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

1 “Injury Due To Use Of Lift”, “Injury Occurring During The Use Of Ladder”, and “Injury To Personnel During Refueling Or Recharging Forklifts”
2 “Fall Due To Wet Surface”
# APPENDIX C: TEST CASES OF REASONING ABOUT THE EXTRACTED CONCEPTS

Table C.1: Test results of reasoning about the concepts describing the synthetic contexts

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Concept Describing the Synthetic Context</th>
<th>Assigned Applicability Value</th>
<th>Grouping Concept</th>
<th>Associated Concept</th>
<th>Disjoint Concept</th>
<th>Equivalent Concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Excavation Using Support Systems</td>
<td><em>applicable</em></td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>—</td>
</tr>
<tr>
<td>2</td>
<td>HVAC Activity</td>
<td><em>not applicable</em></td>
<td>True</td>
<td>True</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Plumbing Activity</td>
<td><em>applicable</em></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Sprain</td>
<td><em>applicable</em></td>
<td>True</td>
<td>—</td>
<td>—</td>
<td>True</td>
</tr>
<tr>
<td></td>
<td>Slip Due to Poly Moving</td>
<td><em>applicable</em></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fall Due to Poly Moving</td>
<td><em>applicable</em></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Struck by</td>
<td><em>applicable</em></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 Proper identification of concepts means that the concepts which are related to the concept describing the synthetic context are correctly identified and their applicabilities are correctly evaluated to be *applicable* or *not applicable* according to the developed reasoning principles.
### Table D.1: Analysis of representing applicability conditions using logical AND/OR

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Applicability of Sub-evaluation Units</th>
<th>Applicability of Root Evaluation Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Act Unit(^1)</td>
<td>JS Unit(^1)</td>
</tr>
<tr>
<td>1</td>
<td>A(^2)</td>
<td>A</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>4</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>6</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>7</td>
<td>A</td>
<td>PA</td>
</tr>
<tr>
<td>8</td>
<td>A</td>
<td>PA</td>
</tr>
<tr>
<td>9</td>
<td>A</td>
<td>PA</td>
</tr>
<tr>
<td>10</td>
<td>NA</td>
<td>A</td>
</tr>
<tr>
<td>11</td>
<td>NA</td>
<td>A</td>
</tr>
<tr>
<td>12</td>
<td>NA</td>
<td>A</td>
</tr>
<tr>
<td>13</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>14</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>15</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>16</td>
<td>NA</td>
<td>PA</td>
</tr>
<tr>
<td>17</td>
<td>NA</td>
<td>PA</td>
</tr>
<tr>
<td>18</td>
<td>NA</td>
<td>PA</td>
</tr>
<tr>
<td>19</td>
<td>PA</td>
<td>A</td>
</tr>
<tr>
<td>20</td>
<td>PA</td>
<td>A</td>
</tr>
<tr>
<td>21</td>
<td>PA</td>
<td>A</td>
</tr>
<tr>
<td>22</td>
<td>PA</td>
<td>NA</td>
</tr>
<tr>
<td>23</td>
<td>PA</td>
<td>NA</td>
</tr>
<tr>
<td>24</td>
<td>PA</td>
<td>NA</td>
</tr>
<tr>
<td>25</td>
<td>PA</td>
<td>PA</td>
</tr>
<tr>
<td>26</td>
<td>PA</td>
<td>PA</td>
</tr>
<tr>
<td>27</td>
<td>PA</td>
<td>PA</td>
</tr>
</tbody>
</table>

\(^1\)Act=Activity; JS=Job Step; PH=Potential Hazard

\(^2\)A=applicable; NA=not applicable; PA=possibly applicable

\(^3\)DD= Disjunctive Description (using logical OR); CD=Conjunctive Description (using logical AND)
A representation example of a complete JHA document for “Purge Gas Operation” activity

```
<Activity>
  <ActTitle>Purge Gas Operation</ActTitle>
  <Concept_App_Condition>
    <Concept>Purge_Gas_Operation</Concept>
  </Concept_App_Condition>
</Activity>

<JobStep>
  <StepTitle>Set up flow meters; Install purge lines</StepTitle>
  <Concept_App_Condition>
    <Concept_AND>
      <Concept>Purge_Gas_Operation</Concept>
      <Concept>Set_Up_Flow_Meters</Concept>
      <Concept>Install_Purge_Lines</Concept>
    </Concept_AND>
  </Concept_App_Condition>
</JobStep>

<PotentialHazard>
  <HazardTitle>Dropping of tools, trips, falls</HazardTitle>
  <Concept_App_Condition>
    <Concept_AND>
      <Concept>Purge_Gas_Operation</Concept>
      <Concept>Set_Up_Flow_Meters</Concept>
      <Concept>Install_Purge_Lines</Concept>
      <Concept_OR>
        <Concept>Dropping_Of_Tools</Concept>
        <Concept>Trip</Concept>
        <Concept>Fall</Concept>
      </Concept_OR>
    </Concept_AND>
  </Concept_App_Condition>
  <RecommendedProcedure>
    <Rule>(1) 100% fall protection.</Rule>
    <Rule>(2) Do not lay tools on equipment or grating. Keep tools in toolbox or similar container.</Rule>
  </RecommendedProcedure>
</PotentialHazard>

<JobStep>
  <StepTitle>Introduction and removal of purge to POC</StepTitle>
  <Concept_App_Condition>
    <Concept_AND>
      <Concept>Purge_Gas_Operation</Concept>
      <Concept>Introduction_And_Removal_Of_Purge_To_POC</Concept>
    </Concept_AND>
  </Concept_App_Condition>
</JobStep>

<PotentialHazard>
```
<HazardTitle>Release of gases</HazardTitle>

<Concept_App_Condition>
  <Concept_AND>
    <Concept>Purge_Gas_Operation</Concept>
    <Concept>Introduction_And_Removal_Of_Purge_To_POC</Concept>
    <Concept>Release_Of_Gas</Concept>
  </Concept_AND>
</Concept_App_Condition>

<RecommendedProcedure>
  <Rule>(1) Communicate with facility tool owner representative. Inform them of operation.</Rule>
</RecommendedProcedure>

</PotentialHazard>
</JobStep>
</Activity>
APPENDIX F: MULTI-TYPE CONJUNCTIVE DESCRIPTION OF APPLICABILITY CONDITIONS FOR SAFETY RULES—CONSIDER CONCEPT NEGLECT

Table F.1: Analysis of representing applicability conditions w/ and w/o concept neglect

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Applicability of Sub-evaluation Units</th>
<th>Applicability of Root Evaluation Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Act Unit(^1)</td>
<td>JS Unit(^1)</td>
</tr>
<tr>
<td>1</td>
<td>A(^2)</td>
<td>A</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>4</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>6</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>7</td>
<td>A</td>
<td>PA</td>
</tr>
<tr>
<td>8</td>
<td>A</td>
<td>PA</td>
</tr>
<tr>
<td>9</td>
<td>A</td>
<td>PA</td>
</tr>
<tr>
<td>10</td>
<td>NA</td>
<td>A</td>
</tr>
<tr>
<td>11</td>
<td>NA</td>
<td>A</td>
</tr>
<tr>
<td>12</td>
<td>NA</td>
<td>A</td>
</tr>
<tr>
<td>13</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>14</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>15</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>16</td>
<td>NA</td>
<td>PA</td>
</tr>
<tr>
<td>17</td>
<td>NA</td>
<td>PA</td>
</tr>
<tr>
<td>18</td>
<td>NA</td>
<td>PA</td>
</tr>
<tr>
<td>19</td>
<td>PA</td>
<td>A</td>
</tr>
<tr>
<td>20</td>
<td>PA</td>
<td>A</td>
</tr>
<tr>
<td>21</td>
<td>PA</td>
<td>A</td>
</tr>
<tr>
<td>22</td>
<td>PA</td>
<td>NA</td>
</tr>
<tr>
<td>23</td>
<td>PA</td>
<td>NA</td>
</tr>
<tr>
<td>24</td>
<td>PA</td>
<td>NA</td>
</tr>
<tr>
<td>25</td>
<td>PA</td>
<td>PA</td>
</tr>
<tr>
<td>26</td>
<td>PA</td>
<td>PA</td>
</tr>
<tr>
<td>27</td>
<td>PA</td>
<td>PA</td>
</tr>
</tbody>
</table>

\(^1\)Act=Activity; JS=Job Step; PH=Potential Hazard
\(^2\)A=applicable; NA=not applicable; PA=possibly applicable
\(^3\)CN=Concept Neglect
**APPENDIX G: TEST CASES OF REASONING ABOUT THE MODELED JHA DOCUMENTS**

Table G.1: Test case results of assigning *applicable* value to single concept of the primary grouping concept “Activity”

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Secondary Grouping Concept</th>
<th>Applicable Recall Value Equal to 1</th>
<th>Not-applicable Recall Value Equal to 1&lt;sup&gt;1&lt;/sup&gt;</th>
<th>Equivalent Relationship Involved</th>
<th>Disjoint Relationship Involved</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Frame Columns</td>
<td>Concrete Activity</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>Excavation Using Support Systems</td>
<td>Excavation Activity</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Working On Or Near Energized Circuit</td>
<td>Electrical Activity</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>Forklift Use</td>
<td>Equipment Activity</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Carcinogen Control</td>
<td>Existing Condition Activity</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>6</td>
<td>Spray Painting</td>
<td>Finish Activity</td>
<td>True</td>
<td>True</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>7</td>
<td>Disassembly Of Material And Equipment</td>
<td>General Activity</td>
<td>True</td>
<td>True</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>Fibrous Insulation And Refractory Ceramic Fiber</td>
<td>HVAC Activity</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>9</td>
<td>Masonry Construction</td>
<td>Masonry Activity</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>Welding Operation</td>
<td>Mechanical Activity</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>11</td>
<td>Install Process Piping</td>
<td>Plumbing Activity</td>
<td>True</td>
<td>n/a</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>12</td>
<td>Steel Erection Operation</td>
<td>Structural Activity</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<sup>1</sup> Cells are noted as “n/a” when there are no *not applicable* safety rules identified for the test concept.
Table G.2: Test case results of assigning *applicable* value to single concept of the primary grouping concept “Job Step”

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Applicable Recall Value</th>
<th>Not-applicable Recall Value</th>
<th>Equivalent Relationship Involved</th>
<th>Disjoint Relationship Involved</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Access Mobile Scaffold</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Manually Adjust Tool To Desired Cutting Depth</td>
<td>Access Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Floor Sealant Application</td>
<td>Apply Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Engage Gear To Start Material Stock Turning</td>
<td>Associate Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Check Dimension With Proper Machining Tools</td>
<td>Check Step</td>
<td>Check Dimension</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>Clean Up Work Areas</td>
<td>Clean Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>Clear Path To Destination Of Object</td>
<td>Clear Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>Fabrication Of Containment In Staging Area</td>
<td>Fabrication Step</td>
<td>Fabrication Of Containment</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>9</td>
<td>Cutting Off Tube</td>
<td>Cut Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>Disassembly Of Piping And Other Overhead Items</td>
<td>Disassembly Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>Disconnect Pressure Supply Line From Test Assembly</td>
<td>Disconnect Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>12</td>
<td>Drill And Secure Anchors And Hangers</td>
<td>Drill Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>13</td>
<td>Ensure Torch Valves Are Off</td>
<td>Ensure Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>14</td>
<td>Excavation Using Sloping And Benching As Protective Measure</td>
<td>Excavation Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No</td>
</tr>
<tr>
<td>15</td>
<td>Lead Exposure During Construction Activities</td>
<td>Exposure Step</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No</td>
</tr>
</tbody>
</table>
Table G.2: Test case results of assigning *applicable* value to single concept of the primary grouping concept “Job Step” (cont.)

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Applicable Recall Value Equal to 1</th>
<th>Not-applicable Recall Value Equal to (1^1)</th>
<th>Equivalent Relationship Involved</th>
<th>Disjoint Relationship Involved</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>Gather Materials Necessary For Job Task</td>
<td>Gather Step</td>
<td>–</td>
<td>True</td>
<td>True</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>17</td>
<td>Inspect Chains Straps And Hooks For Deficiencies</td>
<td>Inspect step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>18</td>
<td>Install Piping Overhead In Lateral Racks</td>
<td>Install Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>19</td>
<td>Ignite Plasma Stream</td>
<td>Ignite Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>20</td>
<td>Loading Concrete Bucket</td>
<td>Load Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>21</td>
<td>Break Forms Loose</td>
<td>Loose Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>22</td>
<td>Lower Cable Hook To Attach Chains Straps And Hooks</td>
<td>Lower Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>23</td>
<td>Movement Of Equipment And Materials</td>
<td>Move Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>24</td>
<td>Acquire Job Task Materials And Hardware</td>
<td>Obtain Step</td>
<td>–</td>
<td>True</td>
<td>True</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>25</td>
<td>Bending Large Spools 3 4th Inch Or Less</td>
<td>Other Operation Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>26</td>
<td>Perform Task</td>
<td>Perform Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>27</td>
<td>Planning</td>
<td>Plan Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>28</td>
<td>Position Material Stock On Table</td>
<td>Position Step</td>
<td>Position Material Stock</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>29</td>
<td>Pour Columns</td>
<td>Pour Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>30</td>
<td>Prepare Area For Scissor Or Aerial Lift Travel</td>
<td>Prepare Step</td>
<td>–</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>31</td>
<td>Pressurize Complete Torch System For Leak Test</td>
<td>Pressure Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>32</td>
<td>Raise Chuck Guard</td>
<td>Raise Step</td>
<td>–</td>
<td>False</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table G.2: Test case results of assigning *applicable* value to single concept of the primary grouping concept “Job Step” (cont.)

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Applicable Value Equal to 1</th>
<th>Not-applicable Value Equal to 1&lt;sup&gt;1&lt;/sup&gt;</th>
<th>Equivalent Relationship Involved</th>
<th>Disjoint Relationship Involved</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>Remove Material Stock Using Overhead Crane</td>
<td>Remove Step</td>
<td>Remove Material Stock</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>34</td>
<td>Replace Object To Original Place Of Rest</td>
<td>Return Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>35</td>
<td>Detailing Of Pipe For Installation</td>
<td>Detailing Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>36</td>
<td>Setting Forms</td>
<td>Set Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>37</td>
<td>Storage OfPrefab Spools</td>
<td>Store Step</td>
<td>–</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>38</td>
<td>Test Piping</td>
<td>Test Step</td>
<td>–</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>39</td>
<td>Transporting Tanks</td>
<td>Transport Step</td>
<td>–</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>40</td>
<td>Turn Off Power At Circuit Breaker</td>
<td>Turn Off Step</td>
<td>Turn Off Power</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>41</td>
<td>Turn On Power At Circuit Breaker</td>
<td>Turn On Step</td>
<td>Turn On Power</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>42</td>
<td>Steps Of Using Mobile Scaffold</td>
<td>Use Step</td>
<td>Steps Of Scaffold Use</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>43</td>
<td>Welding Of Tube</td>
<td>Weld Step</td>
<td>Welding</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

<sup>1</sup> Cells of this column are noted as “n/a” when there are no *not applicable* safety rules identified for the test concept.
Table G.3: Test case results of assigning *applicable* value to single concept of the primary grouping concept “Potential Hazard”

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Applicable Recall Value Equal to 1</th>
<th>Not-applicable Recall Value Equal to 1</th>
<th>Equivalent Relationship Involved</th>
<th>Disjoint Relationship Involved</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Strain</td>
<td>Hazard From Bodily Reaction And Exertion</td>
<td>Bodily Reaction</td>
<td>True</td>
<td>n/a</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>Manually Lifting</td>
<td>Hazard From Contact With Objects And Equipment</td>
<td>Overexertion</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>Caught Between</td>
<td>Caught In Or Compressed By Equipment Or Objects</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Collapse</td>
<td>Caught In Or Crushed In Collapsing Materials</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Abrasion</td>
<td>Rubbed Or Abraded By Friction Or Pressure</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Struck By</td>
<td>Struck By Object</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Injury To Personnel Due To Hydraulic Jack Crushing</td>
<td>Other Contact With Objects And Equipment</td>
<td>True</td>
<td>True</td>
<td>No</td>
<td><strong>Yes</strong></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Impact To Charged Oxygen Line In Work Area</td>
<td>Contact With Electric Current</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Burns</td>
<td>Contact With Temperature Extremes</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Pressure Release</td>
<td>Exposure To Air Pressure Changes</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Exposure To Encapsulate Spray</td>
<td>Exposure To Caustic Noxious Or Allergenic Substances</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Noise</td>
<td>Exposure To Noise</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Exposure To Radiological Contaminants</td>
<td>Exposure To Radiation</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Asphyxiation</td>
<td>Oxygen Deficiency</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
Table G.3: Test case results of assigning *applicable* value to single concept of the primary grouping concept “Potential Hazard” (cont.)

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Secondary Grouping Concept</th>
<th>Tertiary Grouping Concept</th>
<th>Applicable Recall Value Equal to 1</th>
<th>Not-applicable Recall Value Equal to 1</th>
<th>Equivalent Relationship Involved</th>
<th>Disjoint Relationship Involved</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>Debris</td>
<td>Other Exposure To Harmful Substances Or Environments</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Fall Due To Poly Moving</td>
<td>Hazard From Falls</td>
<td>Fall On Same Level</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>17</td>
<td>Fall Into Excavations</td>
<td>Other Falls</td>
<td>True</td>
<td>n/a</td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Gas System Rupture</td>
<td>Hazard From Other Events Or Exposures</td>
<td>Explosion</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>20</td>
<td>Fire Involving Beryllium Material</td>
<td>Other Falls</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>Unforeseen Problems During A Lift</td>
<td>Action Event</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Damage To Equipment</td>
<td>Equipment Event</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Material Dislodgement</td>
<td>Material Event</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Untrained Personnel Working On Or Near Energized Circuits</td>
<td>Personnel Event</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Limited Access And Egress</td>
<td>Space Event</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>Masonry Wall Falling Or Being Blown Over</td>
<td>Structure Event</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>Air Introduced Into New System</td>
<td>System Event</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>Damage To Vehicle</td>
<td>Vehicle Event</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>Vehicle Accident</td>
<td>Hazard From Transportation Accidents</td>
<td>—</td>
<td>True</td>
<td>n/a</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

1 Cells of this column are noted as “n/a” when there are no *not applicable* safety rules identified for the search concept.
Table G.4: Test case results of assigning *applicable* value to multiple concepts

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Separate Reasoning</th>
<th>Combined Reasoning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Number of Applicable Safety Rules</td>
<td>Number of Not Applicable Safety Rules</td>
</tr>
<tr>
<td>1</td>
<td>Frame Columns</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Excavation Using Support Systems</td>
<td>29</td>
<td>23</td>
</tr>
</tbody>
</table>

Table G.5: Test case results of assigning *not applicable* value to multiple concepts

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Separate Reasoning</th>
<th>Combined Reasoning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Number of Applicable Safety Rules</td>
<td>Number of Not Applicable Safety Rules</td>
</tr>
<tr>
<td>1</td>
<td>Carcinogen Control</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>Spray Painting</td>
<td>0</td>
<td>14</td>
</tr>
</tbody>
</table>

Table G.6: Test case results of assigning both *applicable* and *not applicable* values to multiple concepts

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Concept</th>
<th>Assigned Value</th>
<th>Separate Reasoning</th>
<th>Combined Reasoning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Number of Applicable Safety Rules</td>
<td>Number of Not Applicable Safety Rules</td>
</tr>
<tr>
<td>1</td>
<td>Frame Columns</td>
<td>applicable</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Set Pins¹</td>
<td>not applicable</td>
<td>0</td>
<td>5</td>
</tr>
</tbody>
</table>

¹This is a Job Step concept of the Activity concept “Fame Columns”.
APPENDIX H: COMPARISON OF CONTEXT- AND KEYWORD-BASED APPROACHES

- **Search Scenario of “Pour Concrete”**

  Table H.1: Context-based search for scenario “Pour Concrete”

<table>
<thead>
<tr>
<th>Concept(s)</th>
<th>Number of Safety Rules Which Are Relevant</th>
<th>Irrelevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pour Columns</td>
<td>17</td>
<td>0</td>
</tr>
<tr>
<td>Pour Walls</td>
<td>19</td>
<td>0</td>
</tr>
<tr>
<td>Pour Deck with Pump</td>
<td>25</td>
<td>0</td>
</tr>
<tr>
<td>Pour Concrete</td>
<td>61</td>
<td>0</td>
</tr>
<tr>
<td>Place Concrete</td>
<td>61</td>
<td>0</td>
</tr>
</tbody>
</table>

  In this test case, relevant safety rules meant those safety rules identified through the developed context-based approach had *applicable* value.

  Table H.2: Keyword-based search for scenario “Pour Concrete”

<table>
<thead>
<tr>
<th>Keyword(s)</th>
<th>Number of Safety Rules Which Are Relevant</th>
<th>Irrelevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pour Columns</td>
<td>17</td>
<td>0</td>
</tr>
<tr>
<td>Pour Walls</td>
<td>19</td>
<td>0</td>
</tr>
<tr>
<td>Pour Deck with Pump</td>
<td>25</td>
<td>0</td>
</tr>
<tr>
<td>Pour Concrete</td>
<td>52</td>
<td>1</td>
</tr>
<tr>
<td>Place Concrete</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

  In the keyword-based approach, the relevance depends on whether the identified safety rules were relevant to the specified scenario “Pour Concrete” of the test case.

- **Search Scenario of “Steel Erection”**

  Table H.3: Context-based search for scenario “Steel Erection”

<table>
<thead>
<tr>
<th>Concept(s)</th>
<th>Number of Safety Rules Which Are Relevant</th>
<th>Number of Automatically Identified Not Applicable Safety Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Steel Erection Operation</td>
<td>18</td>
<td>31</td>
</tr>
<tr>
<td>Steel Erection</td>
<td>18</td>
<td>31</td>
</tr>
</tbody>
</table>

  Table H.4: Keyword-based search for scenario “Steel Erection”

<table>
<thead>
<tr>
<th>Concept(s)</th>
<th>Number of Safety Rules Which Are Relevant</th>
<th>Number of Automatically Identified Not Applicable Safety Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Steel Erect</td>
<td>18</td>
<td>0</td>
</tr>
<tr>
<td>Steel Erection</td>
<td>18</td>
<td>0</td>
</tr>
</tbody>
</table>
APPENDIX I: USE JHA ADVISOR TO PREPARE A JHA DOCUMENT

Step 1: Add new concepts

a. Click “Add Concept” button to start adding new concepts

b. Add a new Potential Hazard concept “Collapse of excavation walls”
c. Add a new Job Step concept “Excavation using sloping and benching as protective measures”

![Add New Concept for Excavation Using Sloping And Benching As Protective Measures](image1)

d. Add a new Activity concept “Excavation Using Slope”

![Add New Concept for Excavation Using Slope](image2)
Step 2: Add a new JHA Activity section

a. Click “Add JHA Activity” button to start adding a new JHA section

b. Add information for Potential Hazard section, including safety rules
c. Add information for Job Step section

d. Add information for Activity section
e. The new JHA Activity is shown in *JHA Advisor*
APPENDIX J: REASONING ABOUT APPLICABILITY EXCEPTIONS

Table J.1: Test cases of reasoning about applicability exceptions for JHA documents

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Concepts (Used in Applicability Condition or Exception)</th>
<th>Assigned Applicability Value</th>
<th>Applicable Recall Value Equal to 1&lt;sup&gt;1&lt;/sup&gt;</th>
<th>Not-applicable Recall Value Equal to 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Frame Columns (Condition)</td>
<td>applicable</td>
<td>n/a</td>
<td>True</td>
</tr>
<tr>
<td></td>
<td>Precast Concrete Columns (Exception)</td>
<td>applicable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Frame Columns (Condition)</td>
<td>applicable</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td></td>
<td>Manually move forms to area to be installed (Exception)</td>
<td>applicable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Frame Columns (Condition)</td>
<td>neglected</td>
<td>n/a</td>
<td>True</td>
</tr>
<tr>
<td></td>
<td>Fly forms to area to be installed (Condition)</td>
<td>applicable</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Manually move forms to area to be installed (Exception)</td>
<td>applicable</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<sup>1</sup> Cells of this column are noted as “n/a” when there are no *applicable* safety rules identified for the scenarios.
APPENDIX K: AN EXAMPLE OF OSHA DOCUMENT REPRESENTATION

<Part>
  <PartNumber>1926</PartNumber>
  <PartTitle>Safety and Health Regulations for Construction</PartTitle>
  <Subpart>
    <SubpartTitle>Subpart Q. Concrete and Masonry Construction</SubpartTitle>
    <Ctxt_App_Condition>
      <Concept_OR>
        <Concept>Concrete_Construction</Concept>
        <Concept>Masonry_Construction</Concept>
      </Concept_OR>
    </Ctxt_App_Condition>
    <Standard>
      <StandardNumber>1926.703</StandardNumber>
      <StandardTitle>Requirements for cast-in-place concrete</StandardTitle>
      <Ctxt_App_Condition>
        <Concept_AND>
          <Concept>Cast_In_Place</Concept>
          <Concept>Concrete</Concept>
        </Concept_AND>
      </Ctxt_App_Condition>
      <Section>
        <SectionTitle>(a) General requirements for formwork</SectionTitle>
        <Ctxt_App_Condition>
          <Concept_AND>
            <Concept>Cast_In_Place</Concept>
            <Concept>Concrete</Concept>
            <Concept>Formwork</Concept>
          </Concept_AND>
        </Ctxt_App_Condition>
        <SectionText>
          Formwork shall be designed, fabricated, erected, supported, braced and maintained so that it will be capable of supporting without failure all vertical and lateral loads that may reasonably be anticipated to be applied to the formwork. Formwork which is designed, fabricated, erected, supported, braced and maintained in conformance with the Appendix to this section will be deemed to meet the requirements of this paragraph.
        </SectionText>
      </Section>
      <Section>
        <SectionTitle>(a)(1)</SectionTitle>
        <SectionText>
          Formwork shall be designed, fabricated, erected, supported, braced and maintained so that it will be capable of supporting without failure all vertical and lateral loads that may reasonably be anticipated to be applied to the formwork. Formwork which is designed, fabricated, erected, supported, braced and maintained in conformance with the Appendix to this section will be deemed to meet the requirements of this paragraph.
        </SectionText>
      </Section>
    </Standard>
  </Subpart>
</Part>