A KNOWLEDGE MANAGEMENT TOOL FOR METHOD SELECTION AND FEASIBILITY REASONING

by

HRISHIKESH RAJARAM MORE

B.E. (Civil), Shivaji University, 1998

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES
Department Of Civil Engineering

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

November 2002

© Hrishikesh Rajaram More, 2002

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Civil Engineering

The University of British Columbia
Vancouver, Canada

Date November 8, 2002

DE-6 (2/88)

Abstract

Method selection for the various physical components that comprise a project is central to its successful and timely execution. The selection of appropriate construction method for a given project context is a daunting task given the plethora of available methods, resources, and change in technologies. Typically, preconstruction and prebid meetings serve as the venues for method selection decision-making, where experts from diverse backgrounds apply their knowledge and experience to determine a feasible construction process. Generally, these decision-making processes are not documented, and hence thought processes are not captured for reuse, are not readily transferred, and, as a consequence, mistakes can be repeated.

The emerging field of knowledge management in construction shows promise to manage knowledge and experience of construction personnel gained on past projects for future reuse. The knowledge and experience represented in knowledge management tools can be used for partially or fully automated method selection and feasibility reasoning.

In this thesis work, a knowledge management tool for method selection has been developed. After a thorough literature review and interviews with construction personnel, factors affecting method selection and feasibility were synthesized for activities related to formwork, reinforcement, and concrete placement. Using a product-modeling hierarchy, a method-modeling hierarchy, and an expert system inference engine, a feasibility reasoning schema was implemented. The thesis work is done in context of the knowledge domain of concrete high-rise construction. However, it is broadly applicable to other types of constructions as well. A conscious effort was made to provide comprehensive decision support for method selection based on technical considerations (i.e. will it work for the physical features present and the method feasibility considerations required), as opposed to optimization of construction method selection. A full-scale high-rise residential tower was used for proof of concept of the reasoning schema.

Table of Content

List of Figures

List of Tables

Acknowledgement

I would like to express my sincere gratitude towards the invaluable guidance, constructive criticism, and support given by my supervisor Dr. Alan Russell. I greatly appreciate his patience and dedication without which this work would not have been possible.

I would also like to thank Dr. Thomas Froese for the constructive criticism, review, and feedback he provided on the thesis work.

I am indebted to the cooperation and technical assistance provided by my friend and colleague Asad Udaipurwalla. The long hours of technical discussion spent with him were invariably helpful to shape the structure of the thesis.

I would like to thank William Wong for writing and debugging the code necessary for realization of the thesis concept.

I would like to appreciate the time, cooperation, guidance, and knowledge provided by the construction industry personnel interviewed.

Finally, I would like to extend my word of thanks to my friends and family for their encouragement and support.

Chapter 1. Introduction

1.1 Introduction

The important role of knowledge has been known for a long time. A prudent manager knows that the company's key assets are not the real estate, its market share, its stock prices, or its technological assets, but rather they are its people, their skills, and their knowledge. With the advent of information technology, knowledge and knowledge management have become industry buzzwords, because there is growing emphasis on embedding knowledge in organizational work processes for value added products and services. An increasing number of companies have realized the importance of these terms consciously or subconsciously (Davenport and Prusak, 1998).

The Construction industry, seen as a backward industry by some but still one of the bigger process industries, does not lag very far behind in the information technology revolution. Many construction companies are using information technology tools such as project intranets, project extranets, data warehousing, and so forth. These technological innovations have improved communication amongst the various players of a project. But now there is a growing realization by these companies of the need to "know what they know".

Construction companies in their day-to-day operations have to deal with information, data and knowledge. Construction knowledge lies with the company's personnel, who acquire it through their experience and information exchange. Typically, experts from various fields of construction come together in brain storming meetings to apply knowledge and exchange their experiences in order to come up with a feasible set of solutions to the problems at hand. The value of such knowledgeable individuals is realized when they leave the organization. Whenever such "knowledge walkouts" happen the organization suffers a big loss (Tiwana, 2000).

In an interesting example cited by Davenport and Prusak (1998), Russian officials wanted to build a new truck factory. They contacted International Harvester because it had built a plant in Russia twenty years earlier. It turned out that the company lacked the "necessary knowledge" because there was not a single soul still left in the organization that knew anything about the previous project. In such cases, a company pays a hefty price for having ignored the importance of knowledge. Similarly, in another example, a national level construction company was about to win an aquarium construction project contract, but the condition was to have a project manager with aquarium construction experience. The local division of the company did not have such an expert, and they had to relocate an individual from another division after a countrywide search. Such cases are very common in the construction industry, which highlight the importance of knowledge and knowledge management.

In this thesis the aim is to develop a knowledge management tool for the construction industry with particular reference to construction technologies. This tool will facilitate construction professionals to record their knowledge, which will enable organizational learning.

1.2 Terminology

Terms such as data, information, and knowledge are often used interchangeably in the construction literature. For purposes of this thesis it is essential to make them explicit, as follows.

Data

Davenport and Prusak (1998) defined data as, "Data is a set of discrete, objective facts about events". They further state that, "Data itself has little relevance or purpose". In the construction organization context, we can interpret data as "a discrete set of facts about events, processes, and objects, which by themselves don't make much sense about their purpose or relevance." For example, a shear wall can be described in terms of length, height, thickness, etc. These facts by themselves do not give the purpose or relevance of the shear wall. They simply describe the physical component called shear wall.

Information

Many researchers describe information as a message, which is either a document, or an audible or a visible communication (Davenport and Prusak, 1998). A more appropriate definition is given in KLICON (1999): "Information is data interpreted in a given context. Different information may be gleaned from a single data source if the context is different."

For example, the name of the contractor, name of the supplier, date of the receipt, weight of the product, and type of product constitute the discrete facts about an event. When all of this data is viewed from a transaction context, it gives the information about the event that happened, which was a purchase transaction between the contractor and a supplier regarding the given product.

Knowledge

There is general agreement that knowledge is broader, deeper, and richer than data and information (Davenport and Prusak, 1998): "Knowledge is a fluid mix of framed experience, values, contextual information, and expert insight that provides a framework for evaluating and incorporating new experiences and information."

The definition given in KLICON (1999) states: "Knowledge is a body of information, coupled with the understanding and reasoning about why it is correct."

In summary, knowledge can be defined as a body of information and experience with basic understanding and reasoning, which gives a framework for evaluating new information. For example, a piece of knowledge can be stated as: "The method flying truss formwork may be economically feasible only when the flying truss has a minimum of six reuses." This body of knowledge is a piece of experience with basic understanding and reasoning about the feasibility of *Flying Truss Formwork* method, which can be used in other cases to determine the feasibility of adopting this method.

Knowledge Management

After defining data, information, and knowledge it is essential to know "what knowledge management is all about." An elaborate definition of knowledge management cited by Rowley (1999) is: "Knowledge management is concerned with the exploitation and development of the knowledge assets of an organization with a view to furthering the organization's objectives. The knowledge to be managed includes both explicit, documented knowledge, and tacit, subjective knowledge."

Knowledge management essentially involves various processes, which are listed by Galagan (1997) as:

1. Generating new knowledge;
2. Accessing knowledge from external sources;
3. Representing knowledge in documents, databases, software, and so forth;
4. Embedding knowledge in processes, products, or services;
5. Transferring existing knowledge around an organization;
6. Using accessible knowledge in decision-making;
7. Facilitating knowledge growth through culture and incentives; and,
8. Measuring the value of knowledge assets and impact of knowledge management.

Thus, the task of knowledge management, which is often depicted as a "Knowledge Management Life Cycle", deals with the exploitation of knowledge assets by generating, accessing, representing, embedding, transferring, and reusing knowledge for achieving the organization's objectives.

1.3 Literature Review on Knowledge Management in Construction

The architecture-engineering-construction (AEC) industry is one of the largest process industries and a big player in a nation's economy. In Canada itself AEC accounts for 15% of the GDP (Industry Canada, 2002). It is distinct from any other process industry because of its unparalleled fragmentation, which occurs across phases of a project and between the specialists at a given phase (Rivard, 2002). Egbu and Botterill (2001) state that the very nature of the construction industry, which forms temporary multi-disciplinary teams, makes it rely heavily on experience for planning and decision-making, as well as the formation of project and organizational networks. This makes managing knowledge and human capital particularly relevant to the construction industry.

The majority of researchers agree that Knowledge Management (KM) will improve the competitiveness and organizational performance of construction organizations (Robinson et al., 2001; Egbu and Botterill, 2001; Al-Ghassani et al., 2002). The role of KM and organizational learning as a source of competitive advantage is also widely accepted in the AEC industry. A recent survey conducted by Robinson et al. (2001) in the UK, indicated that 80% of the construction organizations surveyed see benefit from KM to their organizations. Moreover, 30% of these construction organizations have a KM policy document and about 40% have a KM strategy.

The study conducted by Egbu and Botterill (2001) revealed that a successful KM program involves factors related to "hard issues" (e.g. technology and knowledge content) and "soft issues" (e.g. culture, people, leadership, motivation). When developing a KM strategy a number of researchers focus on "soft issues" and deal with development of a framework or organizational culture, which enable individuals of the company to share and create knowledge. Other researchers focus on "hard issues" of IT applications to deal with intelligent document management, data warehousing, web-based applications (project intranet, extranet), etc.

Kululanga and McCaffer (2001) argue that even though there is wide acceptance of human intellectual capital as a source of competitive advantage, when it comes to implementation of KM, a proper methodology is lacking. Moreover, they state that for the characterization of a successful KM strategy, construction organizations need to perform an assessment of their current knowledge management practices. In the framework developed by the authors, scaled statement indicators are used to quantify existing KM practices by benchmarking them against general business community KM practices.

In an effort to implement a KM strategy in construction industry organizations, Kamara, Anumba, and Carrillo (2001) advocate that knowledge needs to be managed at two different and interrelated levels:

- Knowledge Management within the project (i.e. across different stages of the project) from the perspective of a temporary organization and its supply chain; and,

- Knowledge Management within the organizations (e.g., construction firms, consultant firms, etc.) to enhance the firm's ability to respond to customer requirements and to transfer knowledge / learning across different projects.

When applying a knowledge management strategy in a construction organization, a number of hurdles have to be overcome. The survey conducted by researchers (Robinson et al. 2001) regarding perceptions and barriers of implementing a KM strategy for large construction organizations revealed that:

- The organizational culture is the most significant barrier. Culture in a construction organization is concerned with the values, beliefs, history, and traditions of the firm.

- Other key barriers include:

    - The lack of standard work processes.

    - Time Constraints- Since a construction project is faced with a fixed time scale; there is often insufficient time for recording and sharing knowledge before, during, and after a project.

    - Employee Resistance- This is closely associated with cultural factors.

A view expressed in the KLICON (1999) study conducted in the UK suggests that before an organization can establish a KM strategy, it must determine what knowledge to share, how to share it, and with whom to share it. The answer to "what knowledge" is important to share can be found in Kamara, Anumba, and Carrillo (2001) as:

- Knowledge of Organizational Processes and Procedures- This includes knowledge of construction processes, statutory regulations and standards, in house procedures and best practices.

- Technical / Domain Knowledge- This knowledge pertains to construction design, materials, specifications, and technology.

- "Know – who" Knowledge- This deals with knowledge of people with skills for a specific task, and knowledge of the abilities of suppliers and subcontractors.

The answers to "how to share knowledge" and "with whom to share it" are essentially the part of the organization's KM strategy that can be implemented with the help of KM tools.

While various computer tools to assist are available in the market, no clear distinction has been made between KM tools and information management tools. The KLICON (1999) study categorizes KM tools into the following categories:

- Knowledge Generation Tools- These tools aid and /or automate the tasks of obtaining, combining, and constructing knowledge (e.g. Internet and Data mining tools).

- Knowledge Representation Tools- Knowledge is context sensitive information. Tools can be used to store the meta-data after removing the context from knowledge. For example, the knowledge about the feasibility of a construction method can be stored as the general feasibility conditions, without any contextual information about the project.

- Knowledge Retrieval Tools- These tools are used for retrieving stored knowledge, summarizing documents and searching documents including emails, web sites, etc.

- Knowledge Sharing Tools- These tools are used to share knowledge through such mechanisms as a Project Intranet, web portals, and Lotus Notes.

In an approach to develop an information management system application called Constructability Lessons Learned Database (CLLD) for a construction contractor, Kartam and Al-Tabtabi (1995) used Lotus Notes. Use was made of the CSI master format as the primary source for listing information about lessons learned in the form of problem faced, solution attempted, and additional comments. The system assists in information management and information searching, but it seldom gives the user the ability to model his knowledge or experience in a readily usable format.

Elhag et al. (2000) developed The Knowledge System for the design phase of Liquid Natural Gas Tank (LNG) projects. The primary objective was to access, capture, and manage knowledge regarding the complex design process of LNG projects. The Knowledge System was implemented as a web-based system that captured information and knowledge across the life cycle of the project. Interestingly, the authors acknowledged the need for knowledge transfer as the information and knowledge evolved within different departments of the company.

The literature review of KM in construction is summarized as follows:

- Construction professionals have realized the importance of KM for their industry. There is greater awareness regarding managing intellectual capital and knowledge to achieve and maintain a competitive advantage.

- Performing audits of present KM practices and organizational processes is necessary to highlight existing KM problems. Analysis of KM objectives of organizations should be performed to identify priorities and to determine if a generic KM strategy can be formulated.

- Most KM work done in the construction industry to date is related to soft issues of the organization such as culture, people, and motivation. Various IT applications have been developed which are aimed at information management and document management. However, no broadly based "knowledge management tool" has been developed for construction processes.

1.4 Method Selection Process

Method selection is one of the important activities of the pre-bid and pre-construction planning processes. Effective method selection is central to the efficient and timely execution of a construction project. During the pre-bid and pre-construction planning phases various experts come together in brainstorming sessions. A survey conducted by Laufer et al. (1993) of leading construction companies in the Western United States shows that analyzing and evaluating various technological alternatives involves a significant amount of collective effort in both planning phases.

Project managers, general superintendents, subcontractors, and design engineers are among the dominant players of the project organization. While evaluating technological alternatives, these participants have to consider various factors including the overall configuration of the project (stand alone project or subproject of a larger complex), available time frame and milestones, structural characteristics and complexities of the project, work quantities and available resources, and costs associated with the alternatives for renting, leasing or purchasing major equipment and /or other temporary facilities for constructing the project.

Experts rely on their past experience and knowledge of technological developments while selecting a feasible set of construction methods for a given project context. There is no standardized process or standard code, which can be used as a guideline in methods selection. Therefore it becomes a highly individualized process. Typically during the method selection process a large amount of knowledge is applied and generated, various assumptions are made, and various method applicability requirements and constraints are discussed. Ironically, records of the process are seldom kept for future reference; basically the reference only exists in the form of the participant's experience.

The lack of explicit information regarding how method selection was performed along with assumptions, requirements, and constraints, hampers organizational learning. By keeping knowledge tacit, the construction industry forces itself to be an experience-based industry (Gil et al., 2001).

Thus a strong case exists to develop a knowledge management tool, which can be used to capture and model past experiences and knowledge regarding method selection in a reusable form. Exchange of such reusable knowledge repositories across the organization can help organizational learning, effective method selection, and facilitate standardization of work processes. Moreover, partially automating the processes of method selection and feasibility checking can relieve key personnel from repetitive work, and free them to identify and explore new innovations.

1.5 Thesis Objectives and Methodology

A thorough literature review, the results of which are presented in Chapter 2, revealed that very few researchers have tackled the construction method selection problem to date. In most cases, what has been done lacks a comprehensive representation of method knowledge, which is necessary for the evaluation and feasibility analysis of various applicable methods. However, as already noted, the newly emerging field of knowledge management in construction shows promise for managing a contractor's knowledge and experience for future reuse. It is desirable to have a tool that gives comprehensive decision support as well as provides a knowledge repository for storing knowledge and experience gained on past projects in reusable format. Currently, however, such a knowledge management tool does not exist.

1.5.1 Thesis Objectives

Specific research objectives are identified and explained below.

- Objective 1- Method selection knowledge and feasibility knowledge elicitation and categorization.

In general, knowledge is available from documented as well as undocumented sources. Documented knowledge is available in the academic literature, trade journals, product brochures, case studies, etc. This knowledge is highly fragmented, difficult to assemble, and often very general in nature. Undocumented or implicit knowledge is also available from construction personnel in the form of experience and rules-of-thumb.

The scope of knowledge acquisition for the thesis was limited to concrete high-rise construction methods, i.e., formwork methods, rebar placement methods, and concrete placement methods. The objective was to elicit available knowledge in the form of factors affecting method selection and feasibility. The emphasis was on technical feasibility factors, as opposed to the non-technical factors such as cost, organizational perspective, and local practices. Although a reasonably narrow application domain was selected, the approach used plus all of the accompanying constructs are broadly applicable to many construction domains.

- Objective 2- Represent knowledge related to method selection and feasibility reasoning in reusable format.

The objective was to represent available knowledge in the form of factors that affect method selection and feasibility in a reusable format. Previous work on a product modeling hierarchy (i.e. PCBS[1]) (Russell and Chevallier, 1998) (Udaipurwala, 1997) and

[1] PCBS i.e. Physical Component Breakdown Structure.

a method modeling hierarchy (i.e. M&RBS[2]) (Udaipurwala and Russell, 2002) as part of the research system called REPCON provided a foundation for the thesis work. These tools were used to model comprehensively the physical components and construction methods that characterize the superstructure system of a concrete high-rise construction project. Minor modifications to these representation schemas were made to allow a comprehensive representation of the knowledge needed for method modeling and feasibility checking. In this sense, the present work has helped to validate the broad applicability of the previous work on physical component and methods representation. What is important to note, however, is that the concepts described in this thesis can be implemented in any environment, which supports a product or physical component model of a project and a rich representation of construction methods.

- Objective 3- Method selection and feasibility reasoning framework.

The objective here was to develop a reasoning system framework using the product modeling (i.e. PCBS) and method modeling (i.e. M&RBS) hierarchies, and "user" defined rules that use physical component attributes and method parameters and condition arguments. This involved mapping the M&RBS hierarchy over the project PCBS hierarchy to allow rule based feasibility reasoning for Method Statement selection.

- Objective 4- Implementation and validation.

The framework developed under objective 3 was implemented to demonstrate workability, and data from an actual project was used to demonstrate that feasible methods could be identified. In reality, the system works by screening methods for infeasibility. No attempt is made to determine what the optimal solution would be for a given project context. This would require extensive consideration of cost, time, safety, quality, risk, and very complex reasoning.

1.5.2 Methodology

The methodology used to achieve the thesis research objectives is detailed below.

- A thorough review of the knowledge management literature pertaining to construction was made. The main emphasis was on knowledge management tools and practices in the construction industry. It was observed that despite the large volume of literature available about knowledge management practices in other process industries, the needs of the construction industry have received very little attention.

- A review was made of the construction method selection systems literature. Knowledge-based applications developed for construction method selection were

[2] M&RBS i.e. Method and Resource Breakdown Structure.

reviewed. The main emphasis of this literature review was on concrete construction method selection. We observed that the available expert systems for method selection use abstracted representations of project data and / or method data. In our opinion the feasibility reasoning they offer is of little help to experienced construction personnel. This observation helped to shape the basic framework for our knowledge management tool.

- A thorough review of different types of literature was made regarding concrete high-rise construction methods. Knowledge related to methods was available in the academic literature, trade journals, product brochures, and case studies in construction industry periodicals. We agree with Hanna et al. (1992) that knowledge available thorough case study and product reviews in trade journals such as *Concrete Construction*, *Concrete International*, *ENR*, and *Concrete Review* is an alternative to an induction technique of knowledge acquisition. Knowledge gleaned in this manner was especially useful for the background preparation for the semi-structured interviews that were conducted with construction industry personnel. Knowledge available in the form of experience and rules-of-thumb was elicited from these individuals along with explicit knowledge to synthesize factors affecting method selection and feasibility. The technical feasibility factors affecting selection of a method were categorized and tabulated. Construction personnel interviewed included a general contractor, two formwork contractors, a concrete contractor / supplier, two rebar contractors / fabricators, a rebar detailer, and a formwork designer. All of these individuals located in the lower mainland area of the Greater Vancouver District, Canada.

- An appropriate knowledge representation scheme was selected to represent the method selection and feasibility knowledge available in the form of tabulated factors. Similarly, an expert system shell was also selected. The key points of consideration for expert system selection were its ability to embed with the legacy system REPCON, relative ease in programming for data transfer between REPCON data structure and the expert system, and the cost of the system. Thus CLIPS 6.2, which was originally developed by NASA in the US, was selected as the expert system as it provides all the required functionality.

- The knowledge available in the form of technical feasibility factors was expressed in the form of production rules using CLIPS syntax. With the help of software codes[3] the PCBS and M&RBS hierarchies were expressed in the form of facts. PCBS and M&RBS Templates, in the CLIPS syntax, were finalized for validation and definition of these facts in CLIPS environment.

- Implementation of the reasoning schema developed and its validation for proof of concept was made by testing the system for a full-fledged high-rise construction project.

[3] The software codes (in C and C++) were developed by William Wong of Construction Management Lab.

1.6 Thesis Structure

As noted in section 1.5, the focus of this thesis is on the development of a knowledge management tool for method selection and feasibility reasoning in high-rise construction. The ability to model experience and knowledge about method selection in reusable form as well as partial automation of the method selection task are central ideas to the thesis. In support of this focus, the thesis is organized as follows:

- Chapter 2 examines past academic work regarding various method selection approaches. Emphasize is on the method selection and resource selection literature in the concrete construction domain.

- Chapter 3 provides an overview of the product modeling hierarchy and the process (method) modeling hierarchy used for the knowledge management tool along with desired modifications to further enhance the hierarchies for method selection.

- Chapter 4 presents an in-depth discussion regarding factors affecting high-rise construction methods selection. The scope is limited to formwork, rebar placement, and concrete placement methods.

- Chapter 5 illustrates the characterization of the feasibility factors knowledge regarding construction methods. This chapter also discusses the knowledge representation scheme used to represent these factors. Issues related to feasibility reasoning system and examples of feasibility rules are also described.

- Chapter 6 explains the feasibility reasoning schema and the steps involved in method statement reasoning.

- Chapter 7 deals with implementation of the feasibility reasoning schema and illustrates proof of concept.

- Chapter 8 concludes the thesis by listing contributions made, findings, and recommendations for future work.

A number of appendices support the foregoing chapters. Specifically, Appendix-A contains method selection and feasibility factors knowledge, Appendix-B contains examples of PCBS and M&RBS facts generated from the example high-rise project, Appendix-C contains examples of PCBS and M&RBS instances showing values associated with the facts, Appendix-D contains examples of Method Statement feasibility rules, Appendix-E contains Method Statement Feasibility Report files, and Appendix-F contains a report on the PCBS and M&RBS hierarchies.

Chapter 2. Method Selection Literature Review

2.1 Introduction

In this chapter we present a review of previous academic work by others related to method selection. We found it useful to divide the literature review into three sections: first, Computer assisted process planning related literature; second, Method selection expert systems related literature; and third, other method selection related literature such as constructability reasoning approach and simulation.

2.2 Computer Assisted Process Planning Related Literature

In a pioneering approach to knowledge-based project planning systems development, Hendrickson et al. (1987) designed the expert system, CONSTRUCTION PLANEX. The system follows essentially a "bottom up" approach while performing the construction-planning process. Method selection is performed in the system at two levels i.e., material selection and crew selection (i.e. technology selection). The first part of method selection is performed on element activities by selecting material packages, while the second part is performed on higher-level activities (i.e. project activities) in the form of technology selection, in which crew types and number of crews are selected. Interestingly, for technology selection the system uses heuristics about soil, site information, resource productivity information, and weather.

It is apparent from the foregoing that the authors do not give an explicit definition of a method. Material selection and crew selection are dealt with separately. Moreover, the authors reckon technology selection as crew type selection, in which they equate resource with technology.

Syal (1992) developed a Construction Method Selection (CMS) model for small to medium sized firms and a design-build environment. He advocates that a number of decision parameters need to be considered in the selection of one method over another. Unlike the authors of CONSTRUCTION PLANEX, Syal proposes that, "the formulation of project activities is dependent upon the selection of construction methods and the associated resources."

Syal (1992) defines construction method as the combination of construction option for the work item and the associated resources. In his model, method selection is a three-tiered process i.e., selection of construction option by defining Construction Process Elements (CPE), assigning crew types, and selecting resources (material, labor, equipment). When selecting a construction option and resources, the author uses firm related (i.e. internal) and project related (i.e. external) decision considerations. Interestingly, he treats defining CPE as part of method the selection process.

In an effort to build a knowledge-based construction scheduling system, Waugh (1989) based the reasoning on project-specific knowledge bases. The system describes a project in a knowledge base with three modules: assembly, crew, and method module. The description of a method is given by a hierarchically listed group of actions. The author used a project-specific description knowledge base for activity generation. Thus the system does method selection indirectly by selecting activities and methods not directly associated with resources or crews. Moreover, the requirements of resources such as technical feasibility, and sufficient space etc., are not explicitly dealt with.

In a similar effort for the development of an interactive planning tool called MDA planner, Jagbeck (1994) used product models and construction methods. The definition of construction method given by the author is very comprehensive and includes domain knowledge of construction and site management as well as rules to compute plans. The methods rules translate construction components into activities and resources. The logic of activity dependency is encoded with the activities. Therefore, an activity plan with a greater degree of detail can be obtained by using a more detailed method. However, the selection of an appropriate method is left to the user. The all-inclusive representation of "construction knowledge" in a method makes its description more complex. The author's approach is similar to Syal's where methods generate activities.

Many other knowledge-based construction project-planning systems have dealt with the generation of activities and schedules. OARPLAN (Winstanley, Chacon, and Levitt, 1993) deals with resources represented under standard classes of equipment, labor, and material. The resource hierarchy is explicitly linked with an action model such that an action (activity within a project) or set of actions is associated with a list of possible resources. Thus OARPLAN simply ignores the method concept although it considers resources. Similarly, GHOST (Navinchandra, Sriram, and Logcher, 1988), IKBS (Gray, 1986), SIPE (Kartam and Levitt, 1990), don't deal with methods and selection of methods for activity generation. Furthermore, Ganeshan et al. (1996) argue that the choice of construction method determines the crew requirements for construction activities and may affect the definition and sequencing of activities. But in their implementation of a rule-based planning system, they make the assumption that "the activity generation process is independent of construction method".

Unlike Ganeshan et al. (1996), Aalami (1998) advocates that, "a choice of a particular construction method determines the activities and their dependencies". (Similar views are expressed by Syal (1992) and Jagbeck (1994).) Aalami developed a construction method model template (CMMT), which formalizes planning knowledge in a computer interpretable format. CMMT allows the planner to model activities required for a particular method. Each activity in a CMMT is defined with fundamental construction entities: components, actions, resources, and sequencing constraints (i.e. <CARS> tuple). The system allows the user to pick a set of construction methods for a given project design, then generates activities, and sequences them automatically for visualization in a 4-D production model. The system leaves the appropriate method selection to the users' discretion and doesn't reason about technical feasibility requirements.

In a recent effort in computer assisted process planning by Rankin (2000; Froese and Rankin, 1998) developed a system called Computer Assisted Construction Planning (CACP). The system uses case-based reasoning (CBR) for method selection. The reasoning scenario considers a product (i.e. project component) in the target project and supports the user in selecting an associated process type (i.e. a method). The system performs a query based on the presented case, which includes the process type and the product type with variables and their values. The retrieved case is then adopted to suit the present case and the process type (with all the relational objects) is added to the target project.

Rankin argues that a method and its process share a "one-to-one relationship" and hence there is no reason to distinguish between them. In CACP the method is modeled in the form of process type, associated process subparts, controls, and other relational objects. The constraints such as feasibility requirements can be associated with Process objects and Product objects. The CBR system, however, compares only product attributes, and a method's (i.e. process type's) feasibility requirements are not accounted for.

As a summary of the Computer Assisted Process Planning Literature, observations relevant to the research detailed in this thesis are as follows:

- From the literature review, one can observe two distinct approaches regarding the generation of activities. Some researchers assume that activities are independent of the methods used in construction, while others believe that the activities exist because of the methods.

- There is no universally accepted definition of Method or Technology. Some authors defined method as a combination of resources while others defined method inclusive of activities, components, and resources.

- Most of the systems developed to date are specifically aimed at automated schedule generation with minimum user involvement, and they tend to ignore the important process of method selection. Albeit there are some systems that treat method selection as a "run-time" user choice. However, they don't do any reasoning about the method's technical feasibility requirements, and further, most use a very narrow definition of what constitutes a method (i.e. it is internal to a single activity and relates mainly to the selection of resources).

- The reasoning associated with a method is essentially about the generation of activities and their associations with resources. These systems do not reason about selection of the method itself.

2.3 Method Selection Expert Systems Related Literature

As noted previously, an objective of the thesis is to develop a Knowledge Management tool for method selection for the high-rise concrete construction domain. Therefore, the author reviewed domain specific method selection literature for formwork methods and concrete placement methods.

2.3.1 Formwork Method Selection

2.3.1.1 SLABFORM: Hanna and Sanvido (1991)

Hanna and Sanvido (1991) developed an interactive horizontal formwork selection expert system called SLABFORM for Horizontal formwork selection. The expert system has seven categories of horizontal formwork systems: conventional wood systems (stick forms), conventional metal systems (improved stick forms), flying truss systems, column-mounted shoring system, tunnel systems, joist-slab systems, and dome systems.

The expert system was implemented in the EXSYS Professional shell. Knowledge is modeled in the form of 'If-Then' rules. The rules are formed to reflect the factors that affect the selection of a particular formwork system. Various factors were identified such as type of slab, building shape, speed of construction, area practice, site characteristics, supporting organization, cost, hoisting equipment, and supporting yard facility.

During a consultation session the system asks the user questions with answer options listed. The system utilizes backward chaining to arrive at a conclusion. The result is displayed with a confidence factor out of 10 (the value of 10 denoting absolute suitability).

2.3.1.2 WALLFORM: Hanna and Sanvido (1990)

Hanna and Sanvido (1991) also developed a vertical formwork selection system called WALLFORM, which is very similar to SLABFORM. They categorized vertical formwork systems into five categories: Conventional formwork, Ganged forms, Slipforms, Jump forms, and Self-raising forms. However, the classification for vertical formwork systems considers only wall formwork systems. Further, they identified factors that affect the selection of formwork system such as vertical and lateral support, concrete finish, site characteristics, and hoisting equipment.

The system consultation works in a similar manner to that of SLABFORM. Again the results are displayed with a confidence factor.

For both expert systems, the rules used to determine feasibility are heuristic rules, which accept answers in abstract or quantitative terms, e.g. Building design is "uniform" or “irregular”. In practice, high-rise building floors are seldom exactly the same because sizes of structural components reduce at higher levels. The floor plate itself can also change in size. Therefore, one has to consider different sets of floors and go through the formwork selection procedure repeatedly.

2.3.1.3 Neuroform: Kamarthi et al. (1992)

This system is a neural network application for vertical formwork method selection. The neural network is trained extensively on the heuristic rule sets formed by Hanna (1991) for the WALLFORM system. In a typical training example, an input vector described the building characteristics and the output vector described the correct choices of formwork system or systems. During consultation the system asks the user various questions and gets information regarding the building characteristics and availability of resources. This information is then translated into an input vector, and the output vector consists of information of formwork systems in terms of ranking.

The goal of the system is not only to make an expert choice of formwork system, but also to understand and mimic the way an expert makes his decision. Heuristic rules are used to train the neural network. It is noted that neural networks lack control of the reasoning mechanism and hence cannot generate an explanation for the results given.

2.3.1.4 EXSOFS: Koo et al. (1992)

Koo et al. (1992) developed an expert system for horizontal formwork selection named EXSOFS (Expert System for Formwork Selection). Formwork systems were categorized into six different types: conventional, table form, flying truss, column mounted shoring system, progressive strength system, and tunnel forming system. Backward chaining was used for formwork selection.

During a consultation session the system asks the user a series of questions on building conditions, site conditions, and cost. Unlike the other systems described, this system asks the user more detailed numerical input regarding building size, typical floor area, number of stories, information regarding formwork resource, etc. Depending on the answers, system performs calculations to give an economical number of formwork sets, number of reuses, and suggestions to complete the project on time. The system even considers localized variations in beam, column, and wall sizes in order to suggest whether a conventional formwork or proprietary system (i.e. system formwork) is suitable. The system can also perform cost calculations for renting or purchasing given a formwork cost database.

2.3.2 Concrete Placement Method Selection

2.3.2.1 ESCAP: (Alkass, Aronian, and Moselhi, 1990; Alkass and Aronian, 1990)

The authors developed an integrated computer system called Expert System Advisor for Concrete Placing (ESCAP) for cost optimization in concrete transportation and concrete placement activities. The knowledge regarding concrete placement equipment selection was obtained from experts and stored in an expert system knowledge base module, which in turn was integrated with procedural algorithms for performing routine calculations needed for the selection process.

The system has four distinct modules: (1) Task identification; (2) Broad equipment selection; (3) Productivity; and (4) Final selection. In the Task identification module the system asks the user questions about site conditions, mixing procedures, accessibility, traffic laws, weather, etc. to determine job conditions in terms of GOOD, FAIR, and BAD. The Broad equipment selection module is subdivided into a Transportation submodule (which selects transportation equipment) and a Pour submodule (which helps the user to select crane and pump as placing equipment). In the Productivity module the ideal output rates of equipment are adjusted by taking into consideration the type of work, weather conditions, operator efficiency, etc. An external program performs calculations regarding type and number of equipment required. In the Final selection module, analysis is performed depending upon machine performance and economic factors for precise comparisons. This system provides extensive help in terms of optimizing the concrete placement process.

As a summary of the methods selection expert systems related literature, the following observations are relevant to this thesis work.

- Few attempts to develop formwork selection expert systems have been made. Systems developed to date use abstract terms (e.g., "uniform" or "irregular") or predefined quantitative terms to describe the project and seldom consider method / resource specific technical feasibility requirements such as minimum slab-bay width required for flying truss formwork, story height required, site space requirement for flying truss assembly, and vertical support requirements for column-mounted flytable system.

- Expert systems developed to date are mainly aimed at training inexperienced people and do not provide valuable project specific decision support to the expert user.

- The method selection approach adopted in these expert systems is aimed at optimization, which often includes a procedural flow of rule execution, making it difficult to update these systems to include new or enhanced technologies.

- The time available for concrete placement as well as the quantity to be placed influences the selection of concrete placement equipment. Often the quantity and the time frame vary according to the desired construction cycle on a high-rise project.

Such considerations demand a more detailed approach for concrete placement method selection.

- The method selection expert systems developed to date are domain specific stand-alone applications. Hence, they lack the necessary characteristics required by a Knowledge Management tool for managing the user's own knowledge / experience and sharing it across the organization in a readily usable format for a broad range of methods and applications.

2.4 Resource Selection Expert Systems Related Literature

A few attempts have been made to develop resource selection systems. Swahney and Mund (2002) took a unique approach for crane selection by using artificial neural networks and expert system technologies to produce a prototype called IntelliCranes. The system has two main modules; a neural network based crane type selection and a knowledge-based expert system for crane model selection. The neural network module selects the type of the crane depending upon user given input regarding the project such as type of use, duration of use, construction height, site spaciousness, etc. The model selection module gets input from the user regarding maximum radius expected, clearance between buildings and boom, load placement height, etc. to give the appropriate model of the crane.

SELECTCRANE is an expert system developed by Hanna (1994). The system asks the user information about the height of the building, maximum lifting capacity required, maximum lift radius, lifting frequency, site conditions, etc. The system recommends a type of crane suitable for the given project. The CRANE system developed by Chalabi and Christopher (1989) on the other hand helps in crane selection as well as its placement. According to the contextual information provided, the system determines the number of cranes necessary for the job. Using site information it also determines the optimum crane location.

Alkass and Harris (1988) developed an expert system for earthmoving equipment selection in road construction called ESEMPS. The system development is aimed at advising inexperienced personnel regarding earthmoving equipment selection. Knowledge is represented in the form of rules, which are linked by if-then logic in the logic tree to reflect the expert's reasoning mechanism. Equipment selection is performed in four stages: identify task and job conditions, select machine, output estimation and machine matching, and select machine by time and cost analysis. The answers given to the system during consultation can be factual (i.e. yes, no, and do not know) or probabilistic answers (i.e. –5 to +5 range).

2.5 Other Method Selection Related Literature

Fischer (1993; 1991) developed construction Knowledge Expert (COKE) for constructability reasoning by linking CAD with an expert system. The aim of the system was to make the contractor's construction knowledge readily available to the designer during the project design phase in order to make better-informed decisions. The author classified constructability knowledge into five categories:

1. *Application heuristics* are the knowledge items that relate overall project parameters (i.e. total floor area, number of floors, etc.)
2. *Layout knowledge* is knowledge related to the vertical and horizontal layout of structural elements (e.g. distance between columns)
3. *Dimensioning knowledge* such as the dimensions of structural elements (e.g. thickness of a slab)
4. *Detailing knowledge* shows the requirements of a given construction method as related to structural details (e.g. rebar arrangement)
5. *Exogenous knowledge* is knowledge that relates to exogenous constructability factors (e.g., weather conditions).

The system was implemented using CIFECAD and the KAPPA-PC expert system shell. The user creates an AutoCAD model of the project and stores the data about type, location, dimension, connections, and additional attributes for every project element. The file (ASCII file) is read by functions in KAPPA-PC to create a symbolic project model. For the purpose of reasoning, methods are represented by frames in the expert system shell. These frames include slots, which are the independent knowledge items with corresponding values. The system gives results in the form of constructability feedback e.g., a function in the system compares the bottom widths of all the beams with the available formwork sizes and alerts the designer by printing a message.

The approach demands extensive data input effort in the CAD model. Modeling knowledge in the form of a slot makes it easier for the user to change it without disturbing internal reasoning functions. The scope of the COKE is limited to feedback to designer. Therefore it only treats the first three categories of knowledge in the implementation. The system only treats knowledge about formwork methods. This system reflects many of the attributes we believe are important in a knowledge management tool.

Skibniewski and Chao (1992) used the analytical hierarchical process (AHP) to evaluate different technological alternatives for method selection. In their relatively informal approach they dealt with risk, return on investment, and benefits as well as intangible benefits such as competitive edge and quality performance. In AHP the decision maker can prioritize his objective by performing sensitivity analysis. This approach is more suitable for large organizations in decision-making regarding large investments such as the purchase of piling equipment, crane leasing or purchasing, etc. However, the evaluation of multiple methods simultaneously becomes a complex issue. Since it does not take into consideration the compatibility between methods as well as method-specific technical feasibility and resource requirements, it does not appear to be a useful tool for method selection at the project level.

Hastak (1998) also used the AHP approach for developing a decision support system. Unlike Skibniewski and Chao, he took a more formal approach and considered five criteria for evaluation: need-based, technological, economic, project specific, and safety or risk. In a group decision modeling system he evaluated each team member for their technical knowledge, experience, project knowledge, and knowledge about the firm. This input is provided to the model for pair-wise hierarchical evaluation of methods using the foregoing criteria plus additional sub-criteria such as labor, skill requirements, etc. This decision support system is useful for the evaluation of new technology where little experience is available. Interestingly, Hastak considered technical requirements and project-specific requirements as intangible evaluation criteria.

In terms of the development of a decision support system for method selection, Allouche (2001) developed such a system for trenchless construction methods. The system performs a two-stage method selection process: technical evaluation and preference evaluation. During the technical evaluation stage, various technical parameters (diameter, maximum drive length, etc.) and compatibility parameters (used for determining a method's suitability to anticipated ground conditions i.e., the project context) are considered. In the preference evaluation stage a risk index is computed for user specified preference attributes such as cost, environmental impact, etc. The system also calculates probabilistic estimates regarding how well the construction method satisfies the preference attributes.

Al-Hammad (1991) developed a knowledge based method selection system (CMSA) as a stand-alone expert system application for the cut-and-cover tunneling knowledge domain. The system uses four operators: suggest, design, predict, and analyze. The suggest operator selects a method, the design operator asks a series of questions to describe design element, the predict operator calculates production cost and assesses risks, and the analyze operator compares the time/cost to the target project time/cost. The system uses simple heuristic rules as well as computing procedures.

Simulation techniques are also useful for method selection where uncertainty is a prominent aspect of construction. AbouRizk and Mather (1998) developed a CAD-based simulation tool for earthmoving construction method selection. They advocate that the stochastic nature of construction processes as well as the dynamic interactions between resources and activities can be effectively handled by using simulation techniques. The system consists of a CAD structure and simulation entities, which are connected by data manager. CAD helps the user by comprehensively describing excavation site in terms of three-dimensional blocks with associated physical properties. The construction sequence and excavation method can be defined for each block. The results of simulation are used in a cost analysis to determine the construction methods with lowest overall project cost.

In an interesting effort towards selecting the optimum construction method strategy, Ugwu and Tah (1998) used a genetic algorithm (GA). He developed a hybrid GA system that is integrated with a project database to perform combinatorial resource optimization. He assumed that construction resources determine the construction method. The system is especially useful where a set of resources is used by a number of construction activities causing resource constraints.

As a summary of above approaches identified in the literature related to methods selection, relevant observations are as follows:

- None of these approaches described provide the characteristics needed for a broadly based knowledge management tool.

- AHP based decision support systems lack a comprehensive representation of methods.

- The simulation approach of method selection taken by a number of researchers is basically a process-oriented approach, and does not take into consideration the feasibility requirements for a method.

- The knowledge based method selection approach taken by a number of researchers is limited by the need to specify comprehensive method and product representations within an expert system shell. This limits their use as a "stand-alone" knowledge management tool.

Not withstanding the foregoing comments, where appropriate, use has been made of related aspects of the work cited (for example, the concept of screening for infeasibility see Al-Hammad (1991) and the knowledge bases compiled by others, especially those of Hanna and Sanvido (1991, 1990) and Fischer (1991)).

Chapter 3. PCBS and M&RBS Overview

3.1 Introduction

In this chapter we describe in detail the existing semantics of the product modeling and method (process) modeling hierarchies. The product modeling hierarchy is called Physical Component Breakdown Structure (PCBS), and the method (process) modeling hierarchy is called Method and Resource Breakdown Structure (M&RBS). The PCBS hierarchy was originally developed by Russell and Chevallier (1998), and the M&RBS hierarchy was developed by Udaipurwala (1997) and Sharma (1997). Use of these hierarchies (along with modifications identified as part of the work described herein) is central to the knowledge management tool described in this thesis.

3.2 Physical Component Breakdown Structure

The physical component breakdown structure (PCBS) consists of a semantically predefined hierarchy of project components. The standard vocabulary of project components includes *project, subproject, system, subsystem, element, subelement, subsubelement, content, material, location set, location, and sublocation. Locations* are mapped onto the components *project* through to *material.* The PCBS provides a robust and flexible hierarchical description of a construction project. The user can model the project in many ways and at varying levels of detail. The UML static structure diagram of PCBS hierarchy is included in Appendix-G. The predefined vocabulary of project components is elaborated upon in the following subsection.

3.2.1 Terminology

Project

The *project* component in the PCBS hierarchy provides an envelope that contains all physical entities and process locations related to the project. It allows the user to define attributes that apply to the overall project.

Subproject

For better understanding and control purposes a project can be divided into self-contained entities called *subprojects*. Each subproject may have its own location set containing locations of the subproject PCBS components.

System

A *system* can be defined as the collection of project components within a *subproject* or *project*. For example, on a high-rise project the superstructure can be modeled as a *system*. Similarly, the mechanical and electrical systems can be modeled as separate *systems* in a PCBS hierarchy.

Subsystem

A *subsystem* can be defined as the self-contained system within a *system*. For example, the vertical transportation system, sprinkler system, and HVAC system are self-contained systems, which can be modeled as *subsystems* under the mechanical *system*. Similarly, vertical elements and horizontal elements of the superstructure could be modeled as *subsystems* of the superstructure *system*.

Element

An *element* refers to a physical component of the project. It can also be defined as the collection of a type of physical components. For example, an individual column can be referred to as an *element* or all of the columns of a specific type (e.g., all round architectural finish column) or the collection of all columns independent of type can be referred to as an element. In general, one is not interested in examining individual components in the PCBS. The detailed information for individual instances of an element type may be found on the project's drawings.

Subelement

A *subelement* can be defined as a physical component or a group of physical components within a certain element category. For example, a group of round columns or rectangular columns could be the *subelement* listed under a category of *element* called "columns".

Subsubelement

A *subsubelement* can be defined as a physical component or a group of physical components that belongs to the *subelement* component. For example, a group of spandrel beams or beams belong to a particular slab-bay (which is a *subelement*).

Content

Content describes a collection of ingredients in a higher-level component. For example, the *element* column has *content* concrete that is a collection of aggregate, cement, water, admixtures, etc.

Material

Material refers to the specific material input of the *content*. For example, the aggregate of a concrete mix could be defined as the *material* of the *content* concrete.

Location set

A *location set* can be defined as the collection of physical or process locations of the project or a subproject. For example, the subproject high-rise tower can have its *location set* containing floor locations. Similarly, a bridge project can have its project *location set* containing pier and span locations.

Location

Location can be defined as the physical entity where the project component is present or a step in an administrative process. For example, main floor, first floor, second floor, third floor, etc. are the locations in the *location set* of a high-rise project or subproject, whereas prepare shop drawings, review shop drawings, fabricate, and ship are the steps in a procurement process.

Sublocation

Sublocation is defined as the part of physical entity called *location.* It allows a richer representation of the spatial dimension. For example, grid lines "A" to "C" or lobby space can form sublocations in a high-rise building. The concept of sublocations has yet to be fully exploited in the PCBS and other aspects of the representation.

Attributes

Attributes are defined as the quantitative or qualitative descriptors used to represent the physical properties of all PCBS component types (Chevallier, 1998). For example, formwork quantity and rebar quantity are the quantitative attributes, while surface finish and soil type are qualitative attributes. Qualitative attributes can be expressed either in linguistic or boolean terms.

3.2.2 Example Project

We modeled a residential high-rise tower project under construction using the PCBS hierarchy. The project consists of a 23 storey residential high-rise tower, three storey town house units, and a four level underground parking structure. The floor plate of the tower is approximately 7000 ft^2 and the total floor area to the project is 260,000 ft^2.

- The project was divided into two subprojects: the high-rise residential tower *subproject* and the town house *subproject.* These subprojects were self-contained entities because the schedule for town house construction was not dependent on the high-rise tower except for start and finish mile stones of the residential complex.

- The focus of this thesis is on methods associated with constructing the concrete superstructure of high-rise buildings. Hence, only the superstructure system of the building was modeled using the PCBS hierarchy. The superstructure *system* was further divided into the vertical components *subsystem* (Columns, walls, core, etc.) and horizontal components *subsystem*. Shown in figures 3-1 and 3-2 is a partially expanded PCBS hierarchy for the tower studied. Figure 3-3 depicts the attributes assigned to the column elements, and shows the mapping of locations onto columns for one of the attributes along with a specification of attribute values.

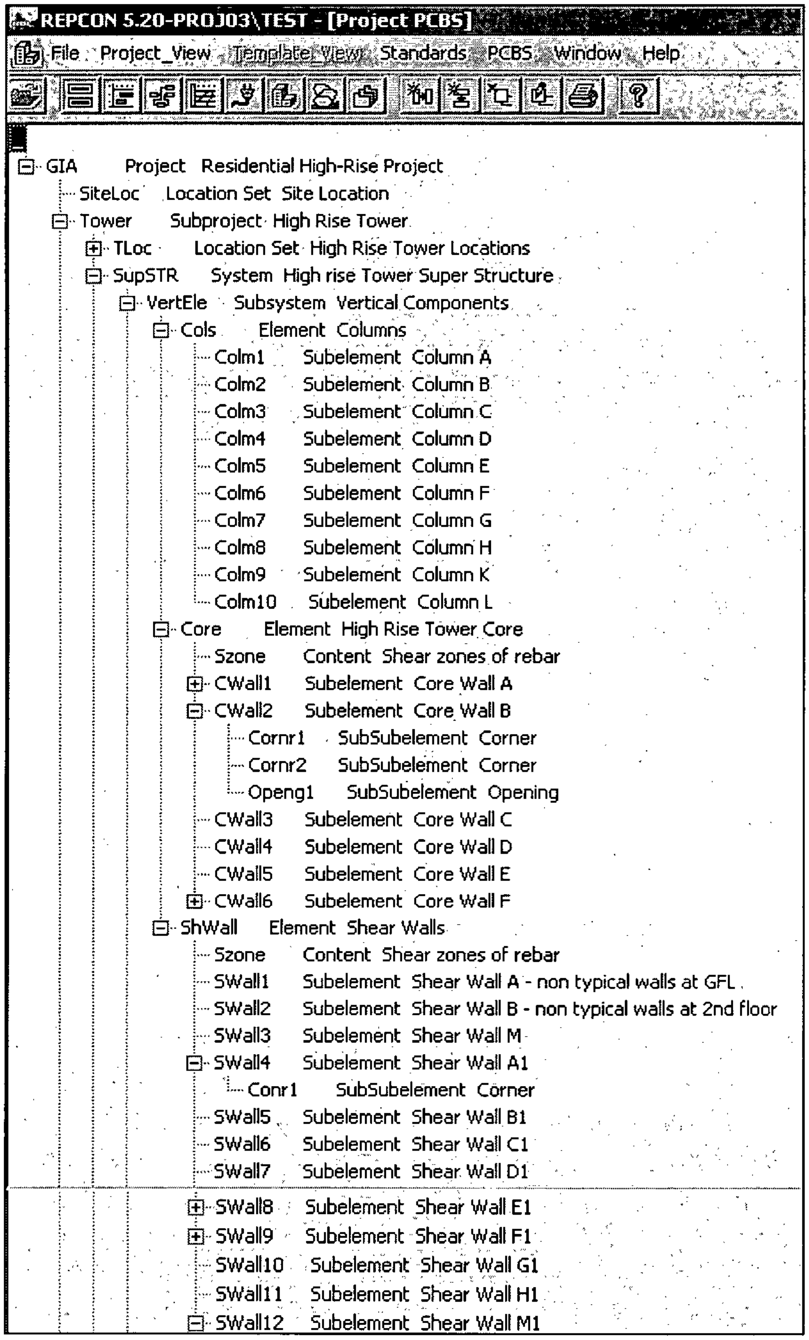


Figure 3.1. PCBS hierarchy (part 1) of the example project.

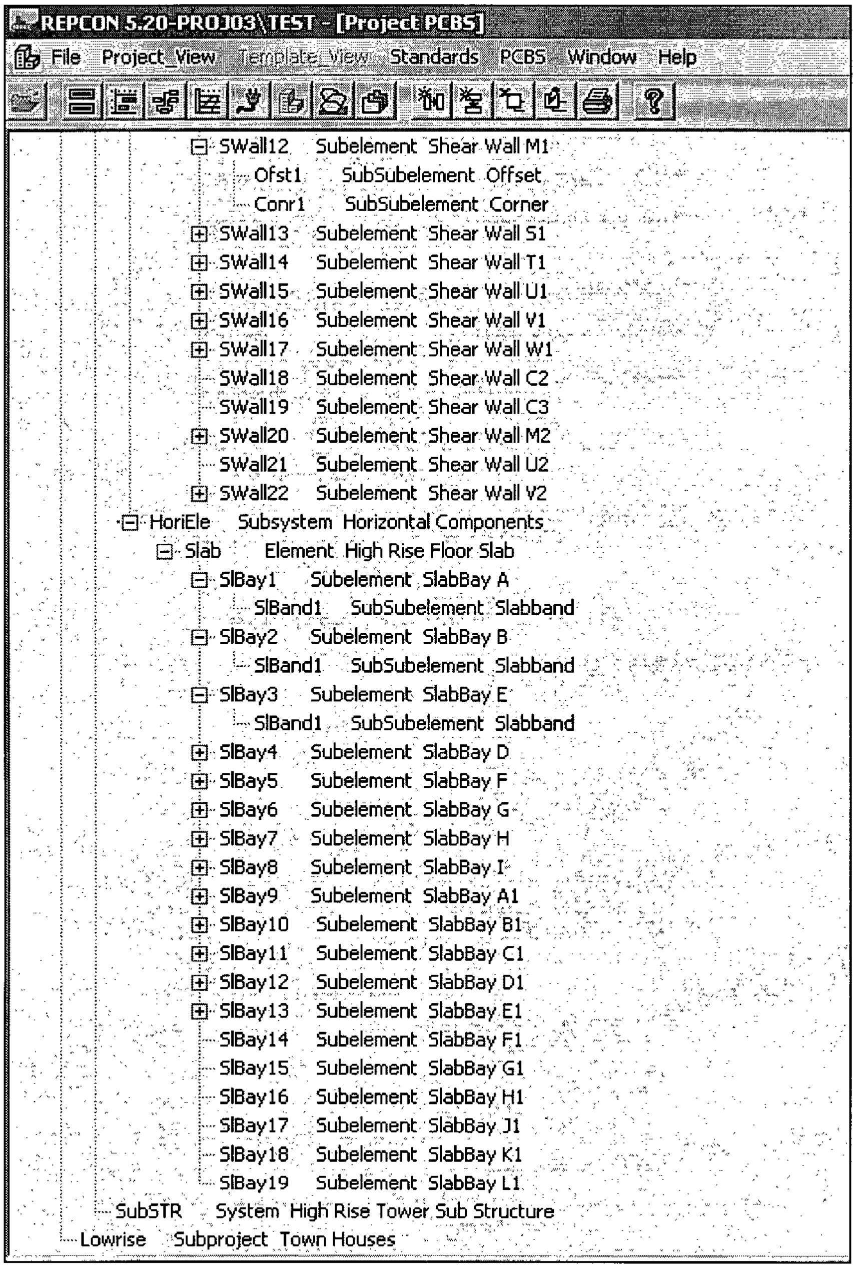


Figure 3.2. PCBS hierarchy (part 2) of the example project.

- The vertical PCBS components such as, columns and walls were further categorized according to the "type of" components present on all locations of the subproject. The columns are categorized according to their length (dimension in x direction). The walls are categorized according to their length (dimension in x direction) and subcomponents. These types of columns and walls were modeled as the column *subelement* and wall *subelement.* As shown in figure 3.1, Column A, Column B, etc. are the types of columns while shear wall A1, shear wall B1, etc. are the types of shear walls. Similarly core *element* and slab *element* were further categorized in terms of the *subelement* constitutes core walls and slab-bays, respectively. The description of slab in terms of constituent slab-bays was useful for modeling the possible orientation of the flying trusses or flytables.

- A point to note from figure 3.3 is that the existence of an *element* at a particular location is dictated by assigning values to the attributes against the desired *location*. These values can be assigned using various conditions such as less than, greater than, within range, etc. The aggregation of quantitative attribute values to a higher level is possible only in the case of inherited attributes, since the aggregation decision is made at a higher-level element (Chevallier, 1998).

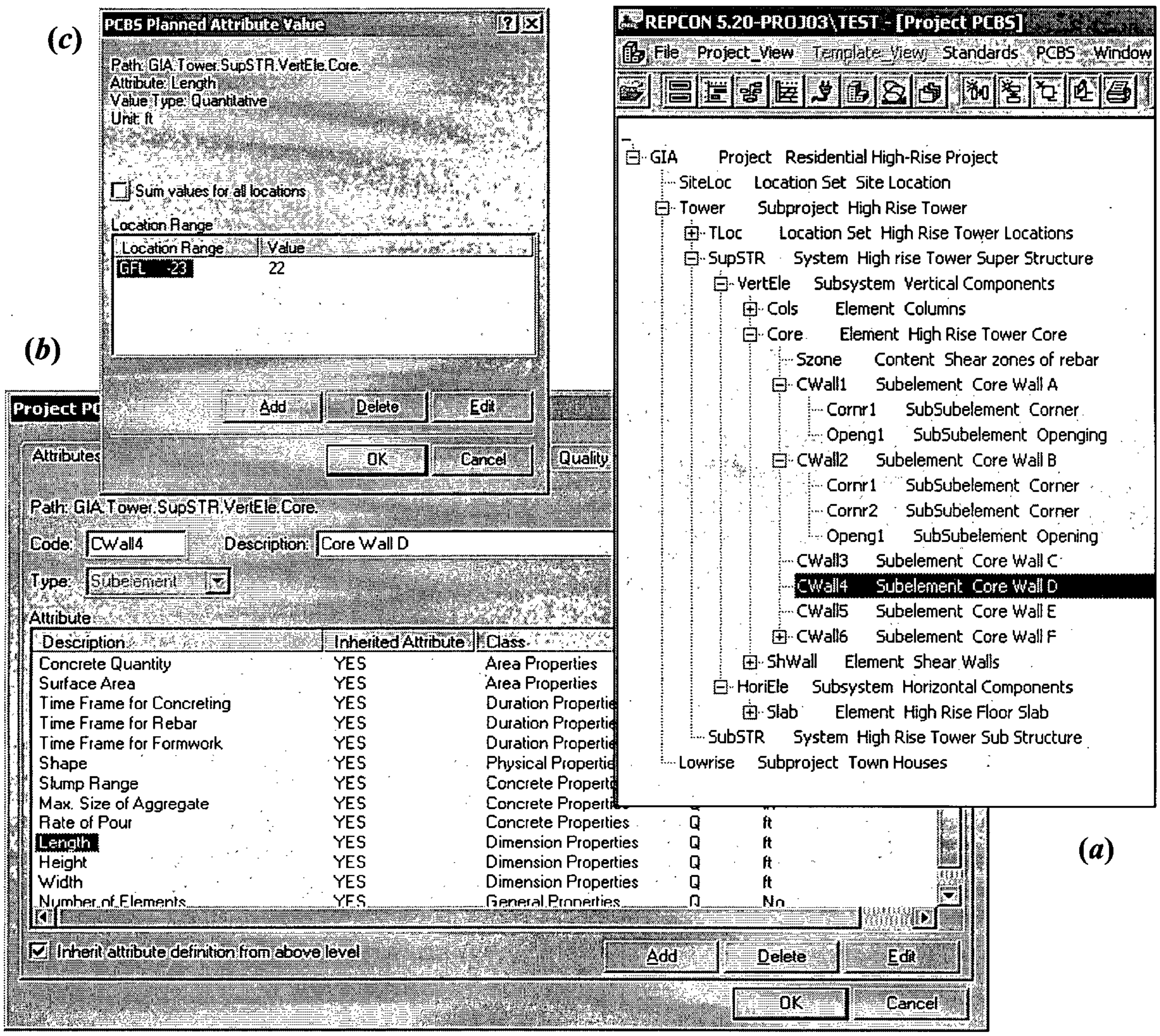


Figure 3.3. (*a*) PCBS component hierarchy with Subelement "Core Wall D"; (*b*) Subelement "Core Wall D" with attribute "Length"; (*c*) The value of attribute "Length" at location range "GFL – 23" showing existence of the component at that location range.

- Project site attributes relevant to decisions on construction methods are described as shown in figure 3.4 for the PCBS component *location* named "Site location". These attributes include length, width, area, site open space width, horizontal formwork storage area, vertical formwork storage area, rebar storage area, parking area, etc.

(*a*)

(*b*)

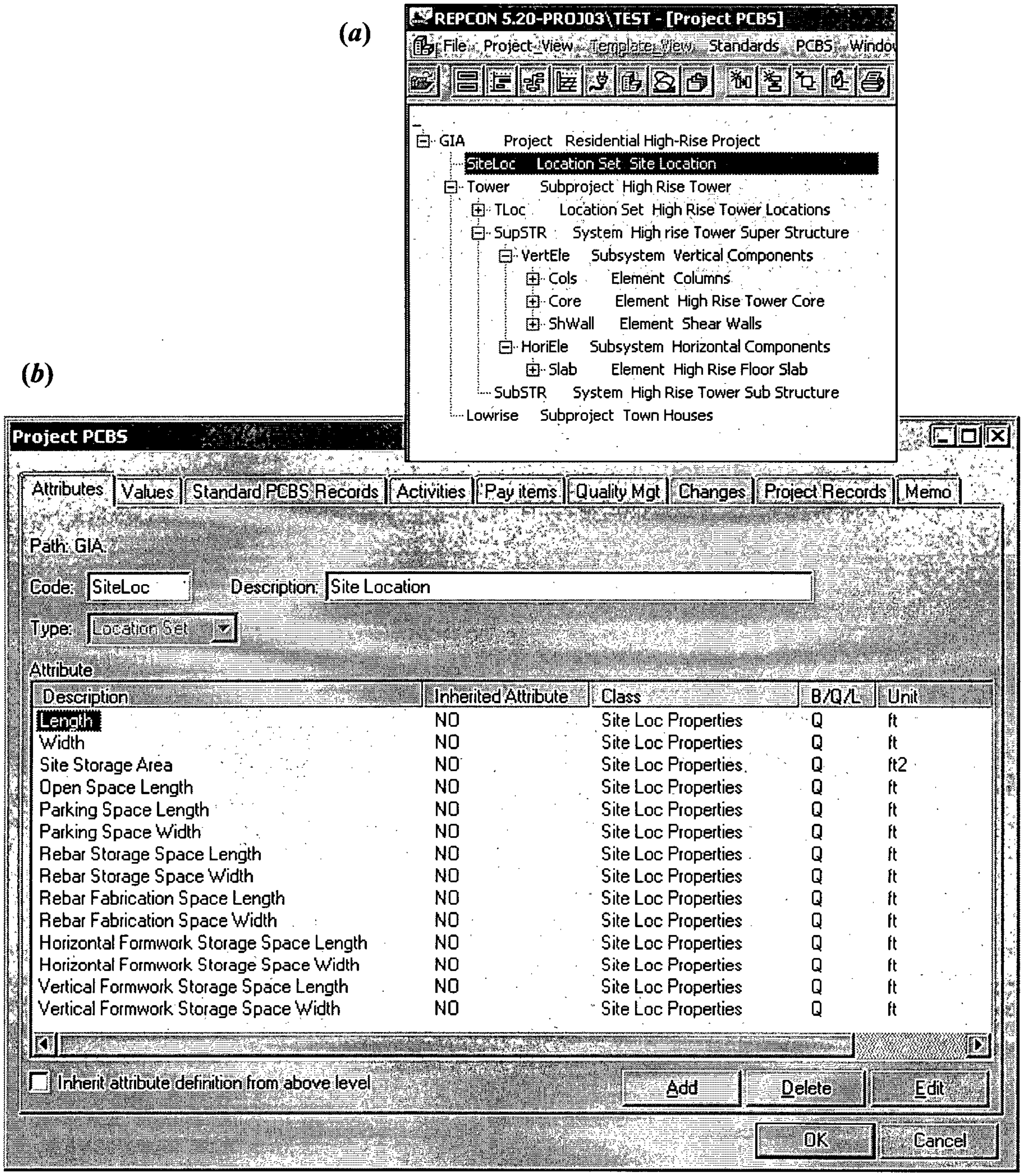


Figure 3.4. **(*a*)** PCBS component hierarchy with Location component "Site location"; **(*b*)** Location component "Site Location" with attributes including "Length".

3.2.3 Additional Features Desired for the PCBS Hierarchy

In order to support rule-based reasoning about a method's feasibility, additional features desired for the PCBS module include the following:

- The knowledge management tool is meant for rule-based feasibility reasoning for comprehensive decision support. The repetitive use of standard method templates is possible by using standard descriptors for defining the project PCBS. Therefore a desirable new feature would be to have classes of standard attributes that can be used when defining PCBS components. This feature was added as part of the current work.

- Standardization of the PCBS component code will assist in avoiding redundancy in the reasoning process by allowing the user to write generic feasibility rules that can be used on a variety of projects. For example, elements of type slab-bay could have a standard PCBS code such as SlBay1, SlBay2, etc. so that a generic rule can be written to check the feasibility of a method with respect to all the PCBS components coded "SlBay '*x*' ". This feature is further illustrated in section 5.8.1 on rule writing. Presently, the system does not offer any help in standardization of the PCBS component code; it is left to the user's discretion.

- Another important desired feature is selective inheritance. The present PCBS structure is a "quasi-object oriented and quasi-hierarchical" listing of project components (Udaipurwala, 1997). Attributes can be inherited from a higher level component to lower level components. However, one either inherits all or none of the attributes. Thus as an example, while describing the type of columns e.g., round columns or rectangular columns under column *element*, the inheritance of attributes from the *element* level to the *subelement* level creates redundant attributes such as the length and width attributes inherited for a round column.

- Arguably, the most desired modification is the addition of a seventh level in the hierarchy i.e., a sub-subelement level. For example, in assessing the feasibility of the column mounted flytable formwork method, it is essential to make sure that the slab-bays do not have a down-turn spandrel beam or one-way beams or two-way beams. Column-mounted jacks can lower the flytables only a few inches and cannot be used if beams are present in the slab (Heinz)(Holm). To reason about such feasibility knowledge, we need to model PCBS component beams or spandrel beams under the corresponding *subelement* slab-bay in the PCBS hierarchy. The additional level under *subelement* slab-bay i.e., the *subsubelement* would help in describing the PCBS hierarchy more comprehensively. Thus the *subsubelement* can be defined as a physical component or a group of physical components that belongs to the *subelement* component . This feature was added as the part of the current work.

An important question when modeling the physical aspects of a project is how much detailed representation is really required in order to perform rule-based reasoning. If too much detail is required, the likelihood of the approach being used in practice is reduced dramatically. In this thesis and for the proof of concept example, a relatively detailed

PCBS description was used. The most appropriate level of detail can only be determined after more knowledge is captured from industry personnel regarding the thought process used to assess a method's feasibility. Experience to date suggests that a method's feasibility depends on a number of determinants, which could be obtained by performing detailed checks on relevant physical attributes of the project's physical components.

3.2.4 PCBS Standard Side and Project Side

The project side and standard side are two important aspects of the PCBS. The standard PCBS side allows the user to form standard PCBS components and hierarchies, which can be used on future projects. The standard side is used to list PCBS components as well as to store knowledge and experience regarding past projects. The standard components listed can be copied over to create a project PCBS hierarchy. For example, a subproject hierarchy on the standard side can be copied to the project level or subproject level in the hierarchy. Standard side PCBS components do not have values assigned to their attributes because the values are project-specific.

3.3 Method and Resource Breakdown Structure

The Method and resource breakdown structure is also a semantically predefined hierarchy. Definitions of each of the components in the M&RBS hierarchy are as follows:

3.3.1 Terminology

Method Statement

A paper-based *method statement* is a formal description of how a physical component of a facility will be constructed. Such a statement is similar to a "work method" description in ISO 9001-2000. In the M&RBS hierarchy, we define *method statement* as a hierarchy containing basic-building blocks i.e., operations, methods and resources to be used in construction of a physical component, collection of physical components, or complete facility. Thus, a *method statement* can have different scopes i.e. a user can formulate a *method statement* about "constructing a typical floor of a high-rise" or about "constructing a specific component type (e.g. core)". The operations, methods, and resources are chosen accordingly. A *method statement* has implicit in it a specific context.

Operation

An *operation* can be defined as the contextual reference to the activity in the schedule i.e. an *operation* could be mapped one-to-one on an activity in the project's schedule. An operation is context sensitive i.e., the operation "Form superstructure column" would be different from "Form a bridge pier". Thus operation depends on the PCBS component, and the method being used must relate to that context. Operations are not listed in the standard M&RBS component library.

Operations can be described in a way that dictates the granularity of a *method statement*. For example, an *operation* can be defined as component specific i.e., "Form Columns" or it can be more broadly defined as "Form vertical elements", which includes walls, columns, and cores. A too broadly defined *operation* is not helpful for method feasibility testing. For example, it would be better to define three *operations* (Build columns, build walls, and build core) and have a method associated with each in order to test for feasibility of the overall *method statement*.

Method

A method can be defined as a standard approach, a novel approach, or a proprietary technology used for the construction of a project component type. Although various other researchers define construction methods by representing the processes and associated resources (e.g., (Rankin, 2000), (Jagbeck, 1994), (Syal, 1992)), we explicitly define methods along with their processes and resources.

A *method* is explicitly described by a set of parameters and conditions. The process aspect of *method* is described with the help of one or more fragnets. A *fragnet* is an ordered or non-ordered set of tasks associated with a *method*. For example a flying truss method is associated with a typical formwork activity sequence, which can be represented as a *fragnet* i.e., strip, rollout, fly-up, set the flying truss, and form deck. Resources used in support of a method can be listed explicitly under the *method*.

Resource

Resources are defined as the entities that get used or consumed during execution of a construction method. A *resource* in the M&RBS hierarchy deals with the physical inputs required to carryout tasks, methods, and operations (Russell et al., 1999). The resource is described with the help of attributes in the form of parameters and conditions, which in general are not context or application specific.

Parameters and Conditions

Parameters and conditions are descriptors used to define different physical and production characteristics, respectively, of methods and resources. Parameters are the descriptors used to describe physical properties of the methods or resources. Conditions are the descriptors used to model feasibility conditions and requirements of the methods and resources. For example, the truss height of the flying truss resource is specified using a *parameter*. Feasibility conditions for the method "Flying Truss Formwork", such as the "required storey height range" and "minimum reuse required" are modeled using *conditions*. Parameters and conditions can be expressed in terms of quantitative, linguistic, or boolean value descriptors. Quantitative values can be described using less than, greater than, within range, and equal to properties.

3.3.2 Example of a Method Statement

We have constructed an example *method statement* for construction of a typical floor in the high-rise tower studied, as shown in figure 3.5. Physical components treated are walls, columns, core, and slab. Observations of importance are as follows:

- The level of detail in the *method statement* (right hand side of figure 3.5) is dictated by the operations described under it. We have used *element* specific operations such as, form column, rebar core, and concrete slab. The descriptions of operations are dependent upon the desired level of detail in the construction schedule because the operations are contextual references to construction schedule activities.

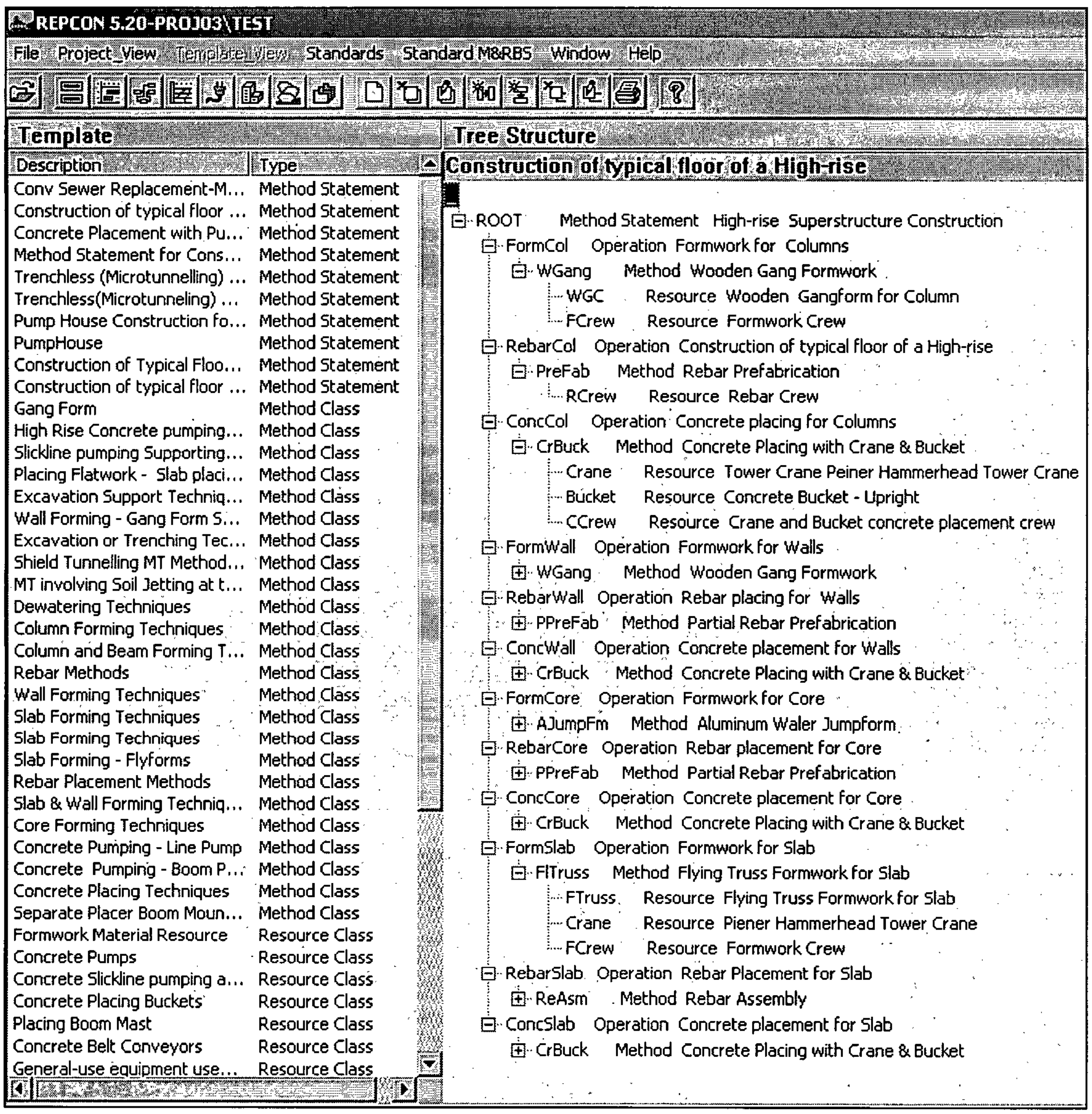


Figure 3.5. M&RBS standard library with component classes and an example Method Statement hierarchy with operations, methods, and resources.

- The methods selected for each operation are listed under the relevant operation. A method is copied under operations along with its attributes, multimedia records, memos, resources, and associated fragnets from the appropriate method class in the library. A fragnet is a set of standard tasks specific to the construction method. These tasks can be used as project schedule activities. However, we agree with the notion of a two level activity generation put forth by Ganeshan et al. (1996) that the first level of activities are the *operations* (which are independent of methods used) in the *method statement* and the second level activities are generated out of fragnets (which are dependent on methods used) associated with *method.* Generation of a hierarchical construction schedule at the proper level is outside the scope of this thesis.

Also shown in figures 3.6, the operations in a *method statement* hierarchy and the selection of a particular concrete placing method from the Concrete Placing Techniques Method Class.

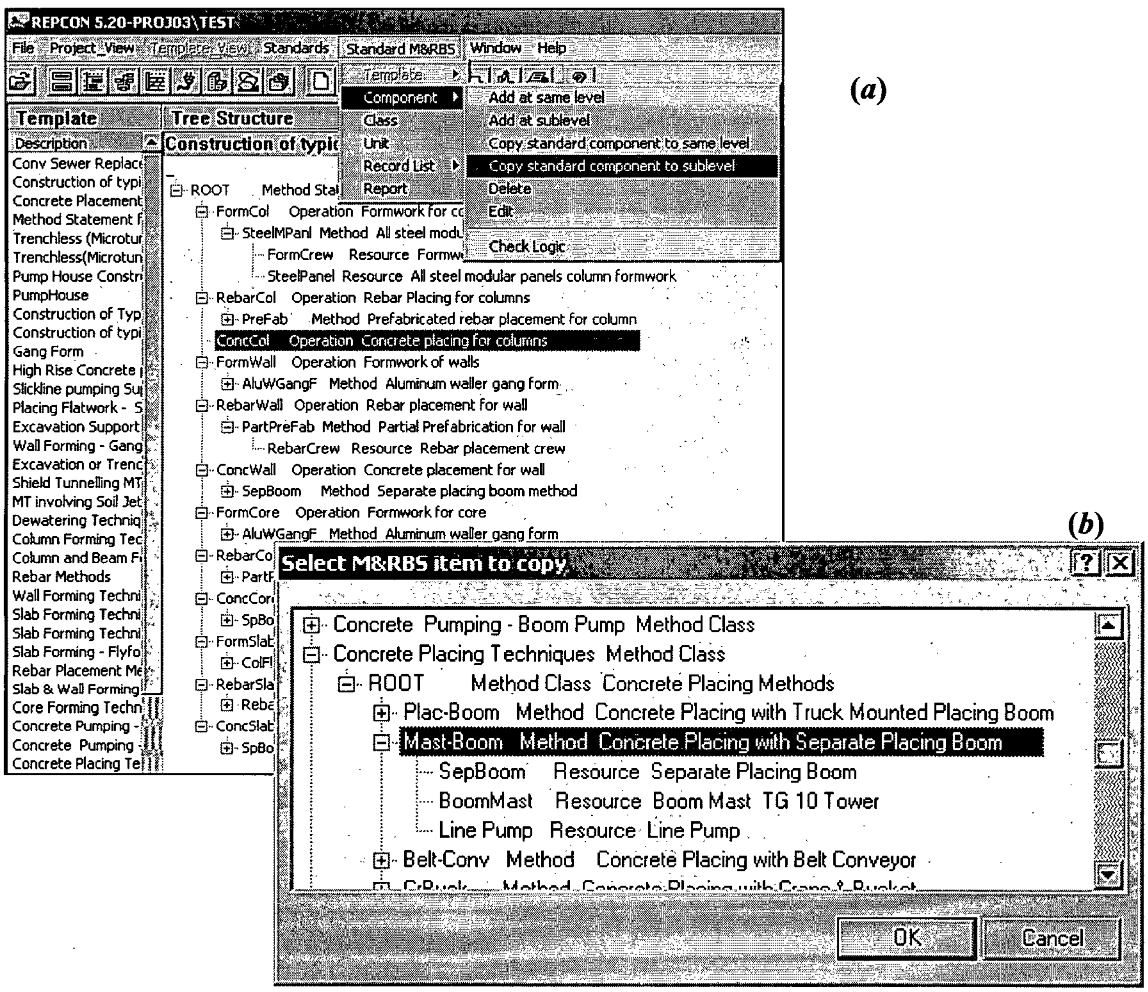


Figure 3.6. (*a*) Method Statement hierarchy with Operation "Concrete placing for columns"; (*b*) Copying standard Method "Separate Placing Boom" and its resources from Method Class "Concrete Placing Techniques".

- After the selection of methods, appropriate resources are selected from the library of resources. A method can have a number of resources listed under it. For example, the *method* Flying truss formwork (for slab forming) requires resources such as flying truss, crane, and formwork crew. Methods and resources are defined further by their parameters and feasibility conditions. Shown in figure 3.7 are some of the parameters and conditions for the method Flying truss formwork. An explicit listing of the feasibility factors knowledge related to concrete construction methods assembled as part of this thesis is given in the Appendix-A.

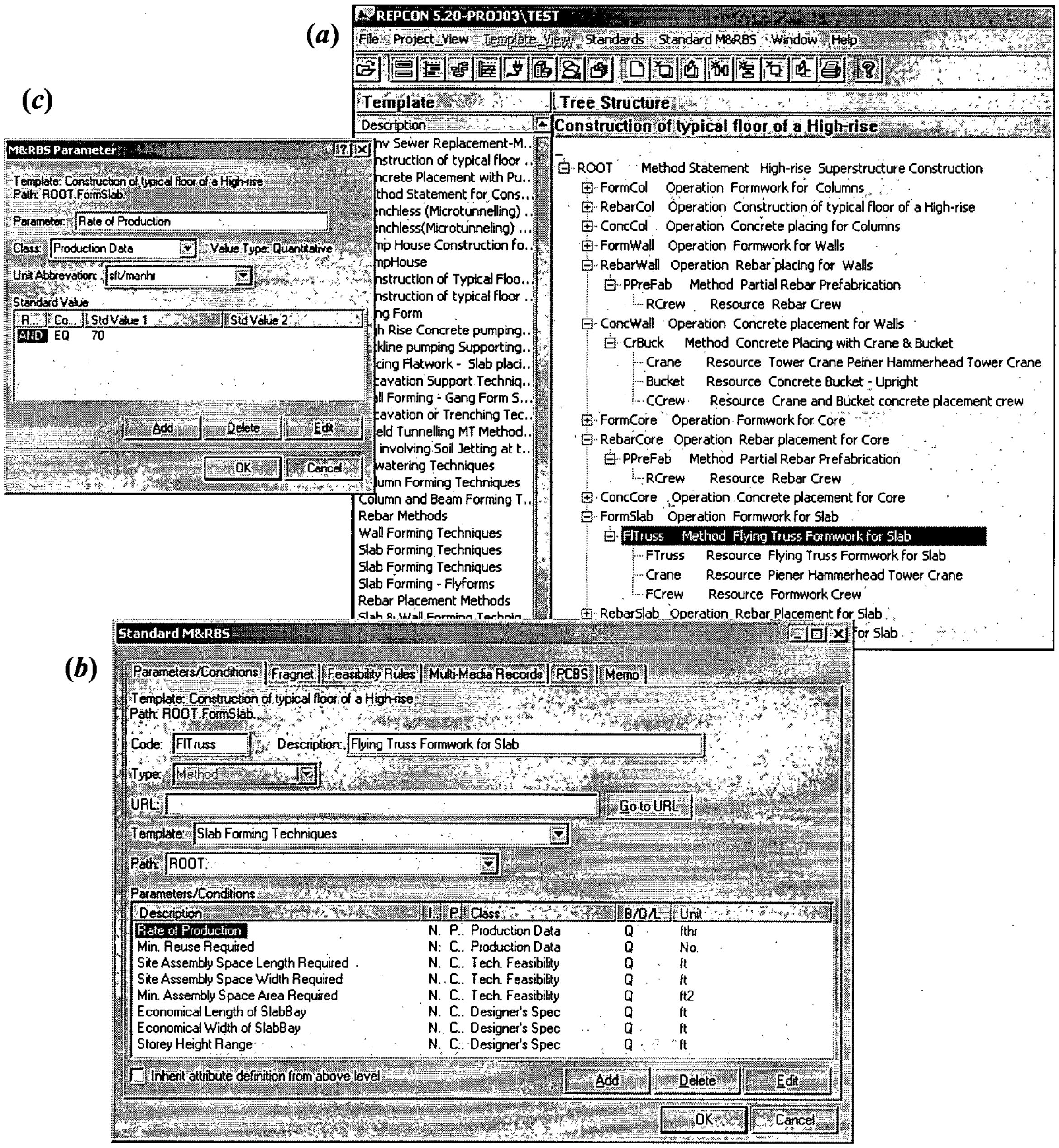


Figure 3.7. (***a***) Method Statement hierarchy with Method "Flying Truss Formwork"; (***b***) Method "Flying Truss Formwork" with its parameters and conditions; (***c***) The value of parameter "Rate of Production".

- As part of the thesis work, method and resource classes were developed for formwork, rebar placement work, and concrete placement work. Some of these are shown in the left hand side of figure 3.5.

3.3.3 Additional Features Desired for the M&RBS Hierarchy

During the course of this work, some desirable enhancements to the existing M&RBS framework were identified, as follows:

- Similar to the PCBS hierarchy, for uniformity in describing M&RBS components, standard classes of parameters and conditions would be helpful for the speedy description of M&RBS components and to assist in avoiding redundancy in the reasoning process due to a mismatch of attributes names. Presently, this feature is not implemented.

- Standardization of the M&RBS component code is desired in order to construct generic rule files that can be used on various projects for different combinations of standard methods and resources. For example, the Flying truss formwork *method* is used for slab forming *operation*. While listing a *method* under an *operation*, care should be taken to use a standard code (e.g. "FlyTruss") so that the generic rule (without any project specific customization) can reason about the method's feasibility by considering appropriate parameter or condition value. Presently, the system does not offer any help in standardization of M&RBS component codes; it is left to the user's discretion.

3.3.4 M&RBS Standard Side and Project Side

The M&RBS hierarchy exists on the project side in the form of a project *method statement* that is the collection of methods and resources to be used to construct the various parts of the project. Multiple *method statements* can exist on the project side. On the standard side, M&RBS modules called classes are used to organize knowledge pertaining to methods and resources. The left hand side of figure 3.5 shows *method statements*, method classes, and resource classes.

In conclusion, the PCBS and M&RBS hierarchies are robust and flexible enough to use as a part of knowledge management tool for method's selection. Over time, enhancements will be to the hierarchies to further assist in the feasibility reasoning process.

Chapter 4 Methods for Concrete High-Rise Construction

4.1 Introduction

In this chapter we review various available construction methods for concrete high-rise superstructure construction and the factors affecting their selection. Formwork, Concrete placement, and Rebar placement are the activities of interest. Extensive use of the literature combined with semi-structured interviews with construction personnel provided the source for the knowledge base assembled. Details of feasibility considerations are compiled in Appendix-A. How these conditions are represented electronically is shown in this chapter through a number of screen captures.

4.2 Formwork Methods

Formwork is one of the most important activity categories for concrete construction. Formwork is regarded as the single largest cost component in high-rise concrete construction. It accounts for 40-60 % of the cost of concrete frame construction (Hanna, 1998). The pace of concrete frame construction is often controlled by formwork related activities that are generally on the critical path of a high-rise project schedule. Thus, selection of appropriate formwork methods for the various physical components becomes an important decision problem because of their time and cost consequences (as well as quality considerations) for the overall project.

Various formwork systems are used on high-rise construction projects. They can be classified in two different ways; according to the resources they use and according to the elements they form. For this thesis we adopted an element-based classification of formwork systems e.g., column forming methods, wall forming methods, core forming methods, and slab forming methods. Furthermore, under the element-based classification we adopted a resource-based sub classification such as Wooden gang form, Aluminum gang form, etc. (see figure 4.1). The classification was made for the purpose of organizing feasibility factors knowledge collected from the literature and from semi-structured interviews with construction industry personnel. Interviews were held with a formwork subcontractor, a formwork designer, and a site superintendent in Vancouver (Lower Mainland area). The list of individuals interviewed is provided in Appendix-A. The surnames of the individuals consulted are cited in round brackets in the discussion that follows. They helped in understanding the thought process behind formwork method selection and in synthesizing factors affecting method selection and feasibility. The factors affecting the method selection are summarized in section 4.2.1.

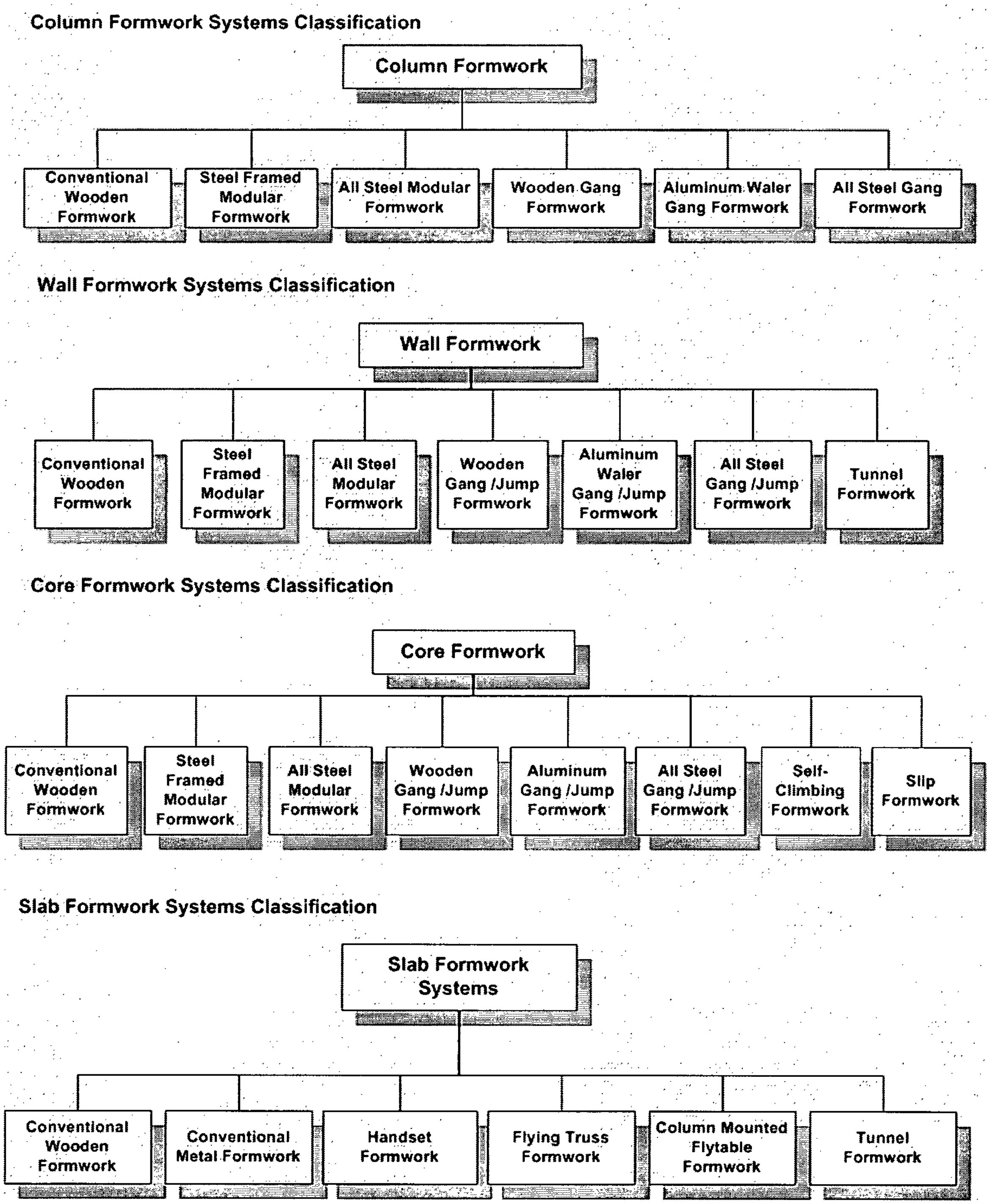


Figure 4.1. Classification of formwork systems.

4.2.1 Factors Affecting Formwork Method Selection

- Element specific characteristics

The selection of an appropriate formwork method depends on the structural characteristics of the physical element type being formed. Gang forming applications for column or wall elements need enough repetition or reuse of gang formwork panels to be justified. Repetition or reuse is essentially governed by size uniformity of the physical elements and their availability on the same floor or other floor of the high-rise project. Element specific characteristics of importance include length, width, height, surface area, storey height, etc. For example, according to Newell and Heinz, the Flying truss formwork system is economically feasible when at least 5 to 6 reuses are available. Similarly, for the method Column-mounted flytable to be feasible the vertical supporting sides of the slab-bay should be parallel and support should be uniformly available along the length of the slab-bay (Wallace, 1997) (Holm).

- Resource Availability & Characteristics

Resource availability plays an important role in formwork methods selection. The Gang form method needs a crane, which is a critical resource for lifting and transporting gang form panels. In some cases, the crane lifting capacity dictates the maximum size of the gang form panel to be used on the job site (Hurd, 1995). Also, the economical size of a flying truss table is determined according to the available lifting capacity of the crane on the site and the weight of the flying truss table assembly (Newell).

- Concrete Pour Characteristics

Placing of concrete faster in formwork can help achieve shorter construction cycle times. The rate of pour is an important characteristic for selection of a suitable vertical formwork system, which is expressed in unit ft/hr or m/hr. The pour pressure or the lateral pressure capacity of a vertical element formwork system depends on the rate of pour. The lateral pour pressure in turn dictates the required tie spacing for the formwork. For proper selection of the formwork system, such as a gang form system, one should know the lateral pressure it can take (Backe, 1986). It is common practice that formwork contractors ask specialist consultants to design formwork for "full head" or a specific rate of pour (Newell) depending upon the desired duration of the construction cycle.

- Site Properties

Access to the site and the site size are among the major influencing factors for formwork method selection. Proper site access is important for delivering preassembled gang form systems or jump form systems. Similarly, site formwork assembly space is required for preassembly of the flyforms for slab formwork (Young). On a constrained site these trussforms can be built in place i.e., on the floor itself, but a significant number of crane hours are needed for material transport (Young). On the other hand, truss forms can be preassembled and transported to the site. However, in such cases additional transportation cost is involved. Vertical formwork systems and horizontal formwork systems need sufficient formwork storage space on the jobsite.

- Time Allowance

The rate of production is an important factor for selection of formwork systems. Many proprietary formwork systems with special features are available in the market, which help in increasing the rate of formwork production by faster assembling, stripping, and recycling. On a high-rise construction project, a general contractor thinks of construction cycles in terms of number of days e.g., a 4-day construction cycle or 5-day construction cycle. To achieve shorter construction cycle times, faster recycling of formwork is of critical importance. Hence the available time frame for a formwork operation in the construction cycle proposed becomes a key factor in the selection of formwork system.

- Cost

Cost of a formwork system is arguably the most important decision making factor for formwork method selection. The contractor has various options of renting, leasing, or purchasing formwork systems. In making the decision as to which option to choose, the contractor has to consider various aspects such as the material cost, labor cost, formwork handling cost, repair and maintenance cost, and potential future uses.

- Other Issues

Various other issues related to formwork method selection are highlighted by researchers such as organizational policy decisions regarding renting and purchasing, and the organization's attitude towards cost, time, and quality aspects of construction (Syal, 1992). Weather conditions can also affect formwork selection because they can delay formwork stripping time, which in turn affects its reuse and reshoring plans (Hurd, 1995). In windy regions, safety requirements such as braces and fasteners for gang form panels should be taken into consideration when selecting a formwork method (Hurd, 1991). However, for the case study high-rise project, during discussions with construction personnel we could not observe any significant piece of knowledge related to these issues.

A summary of the various information categories involved in decision-making about formwork selection is presented in the figure 4.2. Based on the literature review and interviews conducted with construction industry experts, formwork method selection feasibility factors were categorized. Emphasis was placed on classifying tangible method selection and feasibility factors in a tabular format. These tables are formed according to element specific formwork methods as shown in table 4.1 and table 4.2.

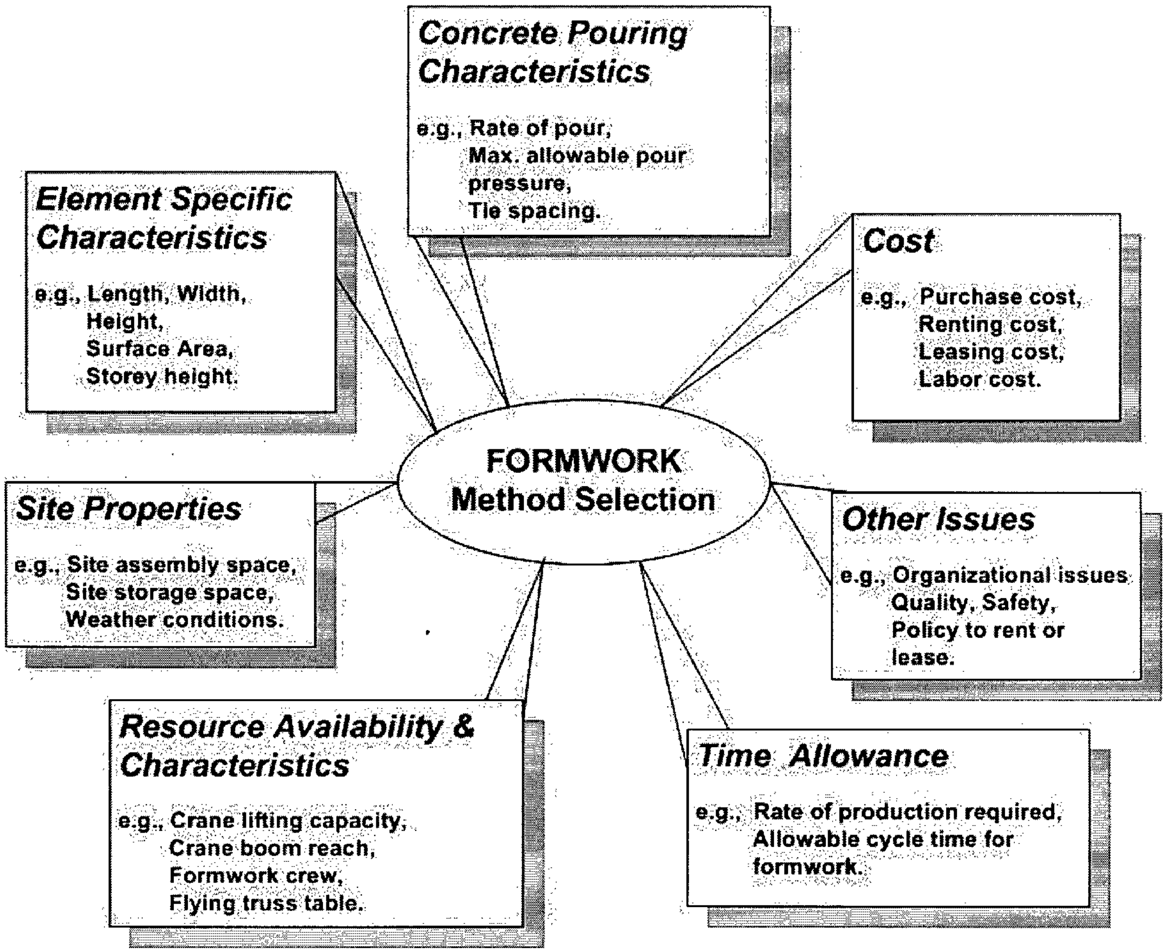


Figure 4.2. Factors affecting selection of formwork methods.

Implementation of some of the formwork knowledge collected within the M&RBS structure defined in Chapter 3 is shown in figures 4.3 through 4.6. Figure 4.3 shows the method class for column formwork, figure 4.4 shows the method class for wall formwork, figure 4.5 shows the method class for core formwork, and figure 4.6 shows the method class for slab formwork. Sample individual methods with parameters and conditions are also shown.

Table 4.1. Table of Slab formwork method selection and feasibility factors knowledge.

Slab Formwork

Method	Con. Wooden Formwork	Con. Metal Formwork	Hand Set Formwork	Flying Truss Formwork	Column mounted flytable	Tunnel Formwork
Factors						
Site Characteristics						
Site Storage/ Assembly						
Length (ft.)	-	-	-	Max. Length of flytable	Max. Length of flytable	-
Width (ft.)	-	-	-	Max. Width	Max. Width	-
Area (sq.ft.)	500 to 1000	>= 500	-	>=1200	-	>= 6000
Other (Open Space Required)	-	-	-	1.5 times Max. Length	1.5 times Max. Length	1.2 times Max. Length
Structural Characteristics						
Slab-bay						
Shape	-	-	-	Uniform	Uniform & rectangular	Uniform
Length (ft.)	-	-	-	22	16 to 30	<= 40
Width (ft)	-	-	-	15 to 30	30 to 40	8 to 18
Area (sq.ft.)	-	-	-	Constant	Constant	Constant
Thickness (in.)	-	-	<=22	-	-	5 to 7
Other	-	-	-	Truss Height <= Storey height - Spandrel beam depth	The spandrel beams should be <= 14 in. deep	Slab-bay should be supported by walls
Slab-bay components						
Beams, Slabbands, Spandrel beam, Droppanels	-	-	-	The beams and spandrel should be uniform.	No components	No components
Storey Height (ft.)	<= 14 ft	<= 40 ft	<= 19	7 to 20	-	7.5 to 12

Production Characteristics						
Rate of production (sq.ft. /manhr)	<= 15	25	<=125	<= 70	<= 71	50
Reuse Required	<= 5 to 6	-	-	>= 5 to 6	8 to 10	40
Resource Required						
Crew size (no. of crew members)	9	9	3 to 4	9	9	15 to 18
Other Equipments				Crane	Crane	Crane
Other	Used on 'non-typical' floors	Used on 'non-typical' floors especially high-rise floors.		Open space around building for maneuverability	Open space around building for maneuverability	Open space around building for maneuverability
				Crane lifting capacity	Crane lifting capacity	Crane lifting capacity

Table 4.2. Table of Wall / Core Wall / Column formwork method selection and feasibility factors knowledge.

Wall / Core Wall / Column Formwork

Method	Con. Wooden Formwork	Steel Framed Modular Formwork	All steel modular Formwork	Wooden Gang Form	Aluminum Waler Gang Form / Jump Form	All Steel Gang Form / Jump Form	Tunnel formwork	Self-Climbing Formwork	Slip Formwork
Factors									
Site Characteristics									
Site Storage/ Assembly									
Length (ft.)	-	-	-	Max. Length of panel	Max. Length of panel	Max. Length of panel	-	-	-
Area (sq.ft.)	>= 500	>= 500	>= 500	-	-	-	>=6000	-	-
Other (Open Space Required)	-	-	-	-	-	-	1.2 times Max. Length	-	-
Structural Characteristics									
Wall									
Shape	-	-	-	Uniform	Uniform	Uniform	Uniform	Uniform	Uniform
Length (ft.)	-	-	-	-	8 to 44	8 to 44	<= 40	-	-

Width / Thickness (in.)		-	-	-	-	-	-	-	-
Height (ft.)	<= 4 (columns <= 16 ft)	4 to 10	-	-	8 to 16 (Gang forms <=22 ft)	8 to 16	7.5 to 12	<= 15	200 to 400
Other	-	-	-	Jumpform need platform 5 ft wide	Jumpform need platform 5 ft wide	Jumpform need platform 5 ft wide	-	-	-
Wall components									
Offsets, Corners, Inserts, Pilasters	-	-	-	Size Uniformity	Size Uniformity	Size Uniformity	No components	Size uniformity	Size uniformity
Pour Characteristics									
Rate of Pour (ft. / hr)	<= 4; (columns 8 ft./hr)	<= 7	<= 7	<= 4; (columns 8 ft./hr)	6 to 9	6 to 9	-	-	-
Allowable pour pressure (psf)	600-750 (columns <= 1200 psf)	<= 1200	<=1200	600-750 (columns <= 1200 psf)	<= 1200	1200 to 1500	-	-	-
Tie Spacing (ft.)		1 to 2	2	3 to 4	6	6 to 8	-	-	-
Production Characteristics									
Rate of production (sft/manhr)	<= 19	19	<= 65	35	35	55 to 70	50	-	80 ft. / workweek (24hr)
Reuse Required	<= 3 to 4	-	-	30 to 40	40	40	40	>= 30	50-100
Resource Required									
Crew size (No. of Crew members)	7	7	6	7	7	7	15 to 18	10 to 12	-
Other Equipments				Crane	Crane	Crane	Crane	-	-
Other	-	-	-	Crane lifting capacity	Crane lifting capacity	Crane lifting capacity	-	Crane lifting for initiallation / removal	Concrete supply at rate of 20 c.y. / hr

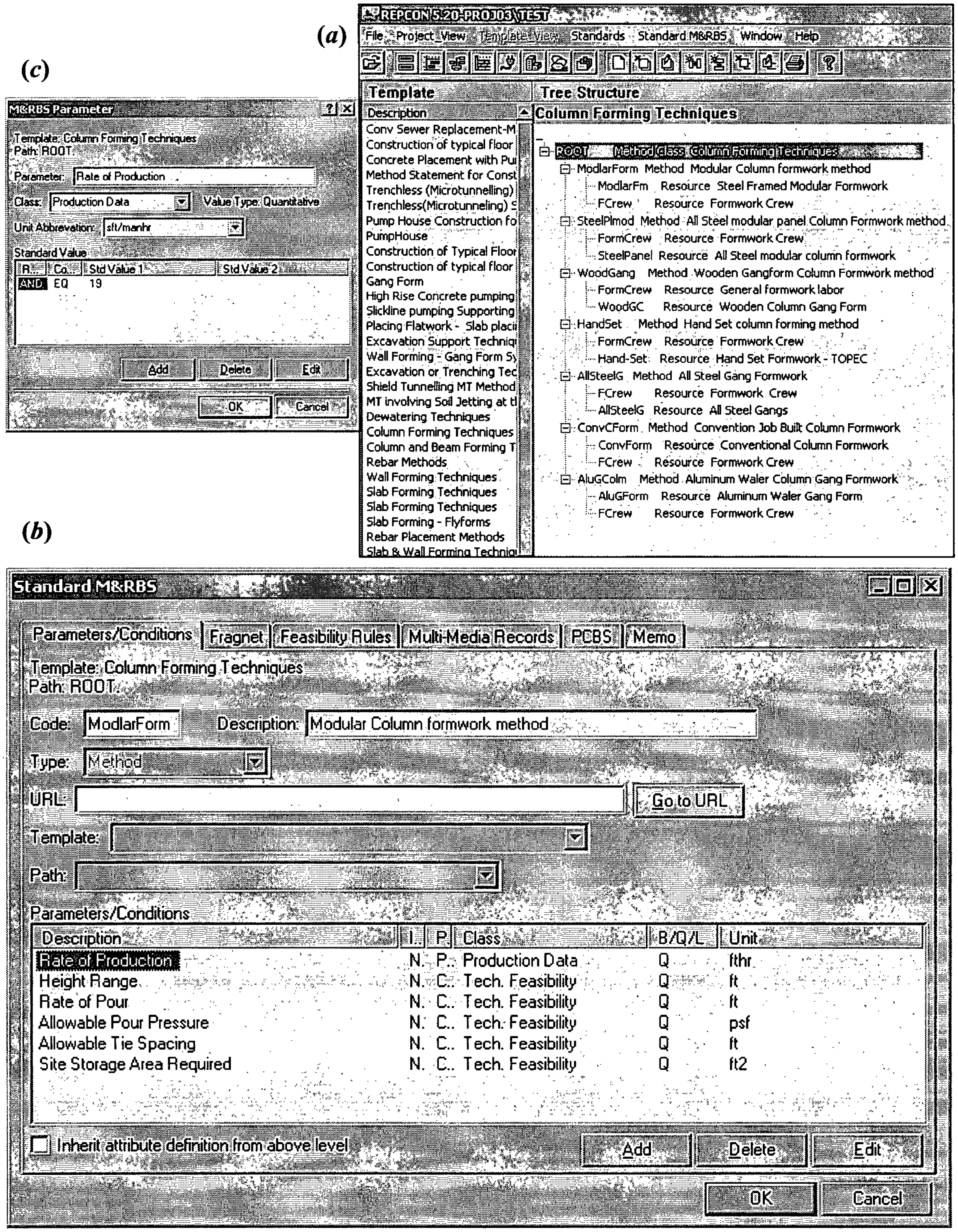


Figure 4.3. ***(a)*** Method Class "Column Forming Techniques"; ***(b)*** Method "Modular Column Formwork" with parameters and conditions; ***(c)*** Parameter "Rate of Production" with value.

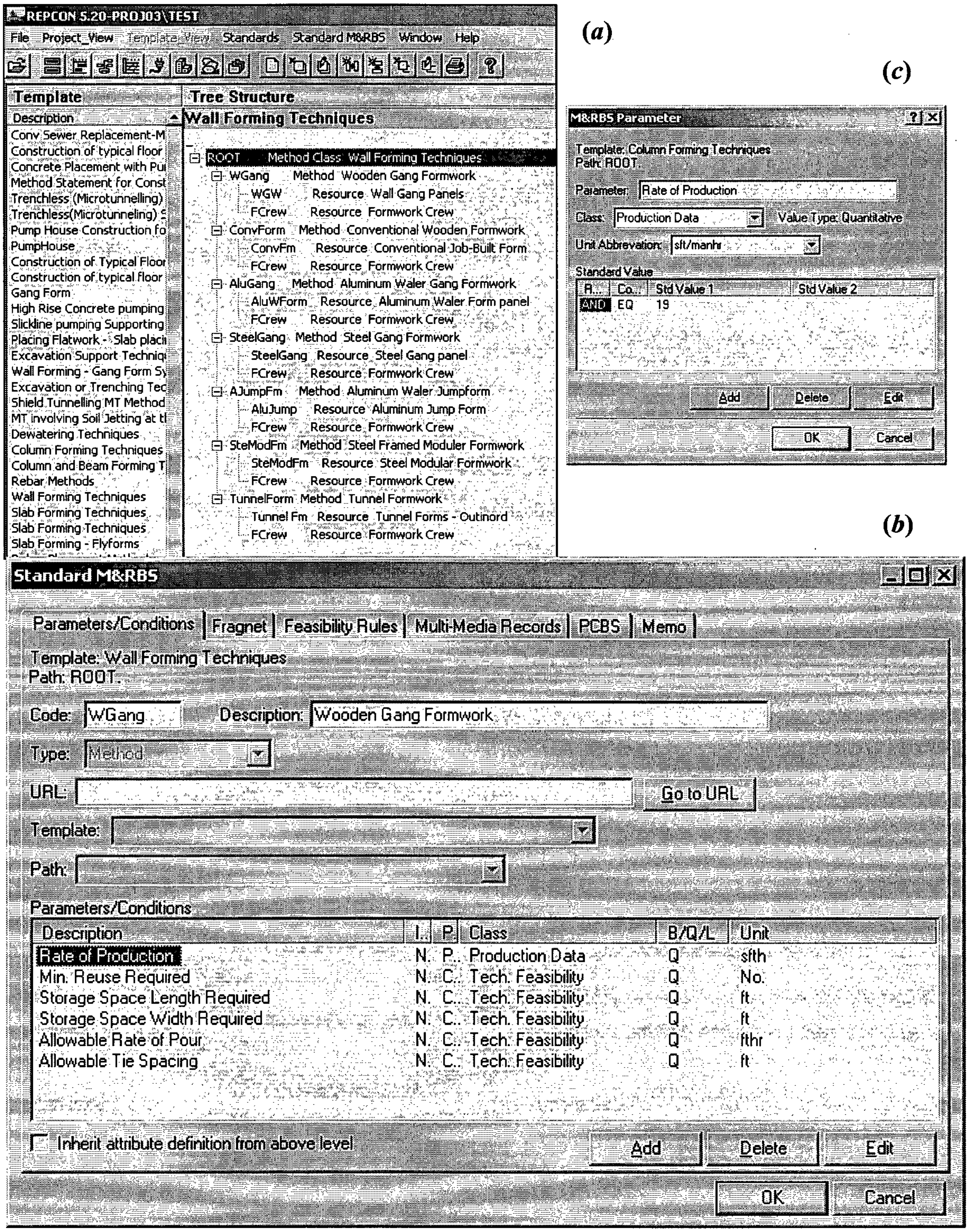


Figure 4.4. ***(a)*** Method Class "Wall Forming Techniques"; ***(b)*** Method "Wooden Gang Formwork" with parameters and conditions; ***(c)*** Parameter "Rate of Production" with value.

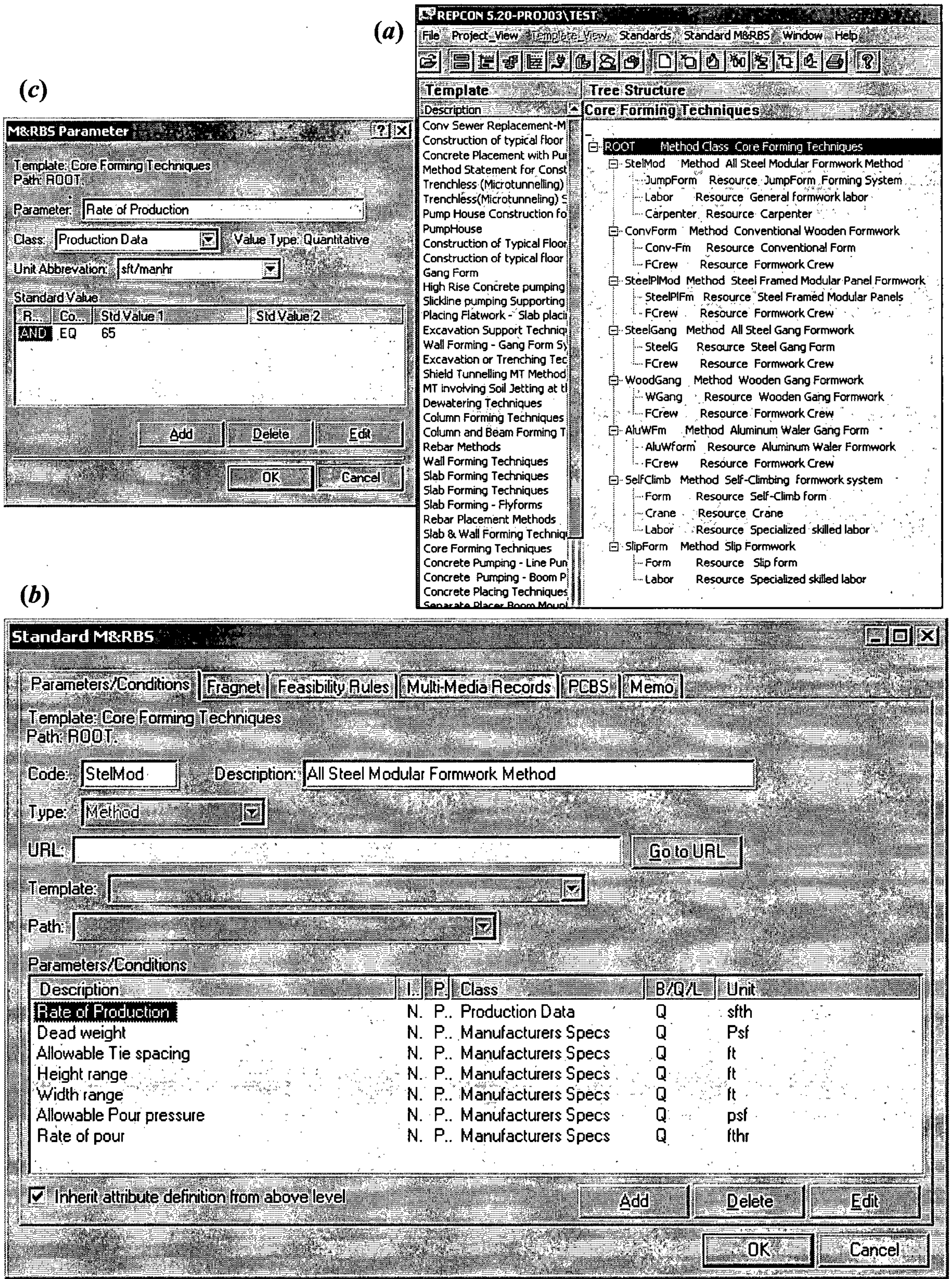


Figure 4.5. (*a*) Method Class "Core Forming Techniques"; (*b*) Method "All Steel Modular Formwork" with parameters and conditions; (*c*) Parameter "Rate of Production" with value.

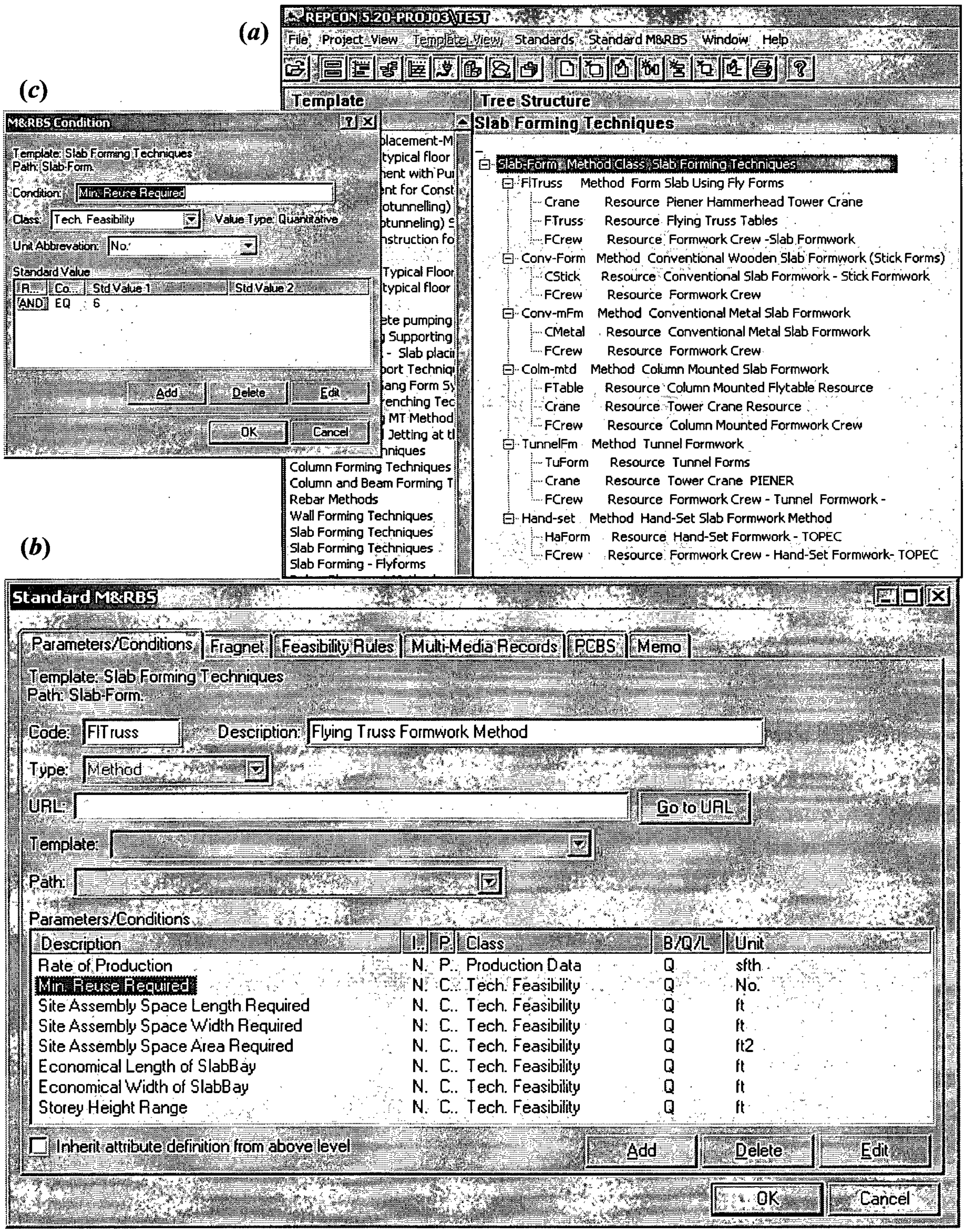


Figure 4.6. (*a*) Method Class "Slab Forming Techniques"; (*b*) Method "Flying Truss Formwork" with parameters and conditions; (*c*) Condition "Min. Reuse Required" with value.

4.3 Concrete Placement Methods

The selection of an appropriate concrete placement method needs careful analysis of various factors related to the project's context, the method's feasibility parameters, weather conditions, and cost of operations. Often in practice, concrete placement method selection is done in real time according to the quantity of concrete to be placed and the available time frame (Yaeger). For high-rise construction, the concrete placement method selection task becomes complicated due to the significant amount of vertical transportation involved.

Various concrete placement methods are available such as crane and bucket method, wheelbarrows or mechanical barrows, belt conveyors, slickline pumping, placing boom, and separate placing boom. These methods can be characterized according to the type of resources they use. For example, crane and bucket method can be characterized according to the type of the crane and the bucket as its resources, and the slickline-pumping method can be characterized according to the concrete pump it is using. The properties of these concrete placement methods are essentially governed by the properties resources they use.

Based on the literature survey and interviews conducted with a concrete placement subcontractor and a site superintendent, concrete placement methods were classified into five main types: Crane and bucket method, Belt conveyor method, Slickline pumping method, Placing boom method, and Separate placing boom method. Various other methods such as, Wheelbarrow, Motorized barrow, or Concrete hoist are rarely used on high-rise construction site, and hence were not treated.

4.3.1 Description of Concrete Placement Methods

- Crane and Bucket Method

This is the most commonly used concrete placement method. For high-rise construction, this method uses a tower crane as the vital resource along with one or more concrete buckets. The rate of concrete placement depends on the speed of the crane, concrete bucket capacity, number of buckets, and the travel distance. Generally, the rate of concrete placement varies from 25 yd^3 to 50 yd^3 per hour.

- Belt Conveyor Method

The belt conveyor method of concrete placement is generally used on below grade or low-rise concrete placement jobs. According to the type of the belt conveyor used, the method can have a vertical reach up to 87 ft, a horizontal reach up to 150 ft, and a vertically downward reach of 25 ft. The rate of concrete placement can vary from 50 to 230 yd^3 per hour. This method is very cost effective for mass concreting for foundation mats and grade slabs.

- Slickline Pumping Method

Slickline pumping is a commonly used concrete placement method. It uses a concrete pump and steel pipeline as its resources. The rate of concrete placement essentially depends upon the concrete pump rate. This method is useful for mass concreting. A steel or cast iron pipeline is used to pump concrete to its final destination. Various pipeline layouts are made according to the location and size of the concrete pour (Crepas, 1985).

- Placing Boom Method

This is one of the most commonly used methods of concrete placement on high-rise construction. It employs a truck mounted placer boom as its resource. Depending on its make, a placer boom can have a horizontal reach of 174 ft, a vertical reach up to188 ft, and a vertically downward reach of 137 ft. This method is very effective for concrete placement for vertical and horizontal elements, and massive foundation elements.

- Separate Placing Boom Method

The separate placing boom method includes a concrete placing boom separately mounted on a mast. The placing boom mast can be installed in several different ways. The most common way is to use a self-climbing boom mast in a blockout created in the slab. Concrete is supplied to the placing boom by a slickline and concrete pump assembly. This method essentially combines the advantages of both slickline pumping and placer boom method of concrete placement.

4.3.2 Factors Affecting Concrete Placement Method Selection

Various factors need to be considered in concrete placement selection. They are overviewed in figure 4.7 and discussed below.

- Concrete Properties

Various properties of the concrete mix need to be considered when selecting a concrete placement method. The maximum size of aggregate and the type of the aggregate (i.e. lightweight or conventional) influence the suitability of the concrete pumping method. Most of the available concrete pumps cannot handle an aggregate size of more than 2.5 in. (Putzmeister, 2001c). Moreover, special care should be taken when pumping lightweight concrete due to its higher slump loss while pumping. Concrete slump is the limiting factor for selection of a belt conveyor method, as the feasible range of concrete slump is 1 to 7 inches (CC, 1992). The concrete quantity and the required rate of concrete placement dictate the selection of the concrete placement method.

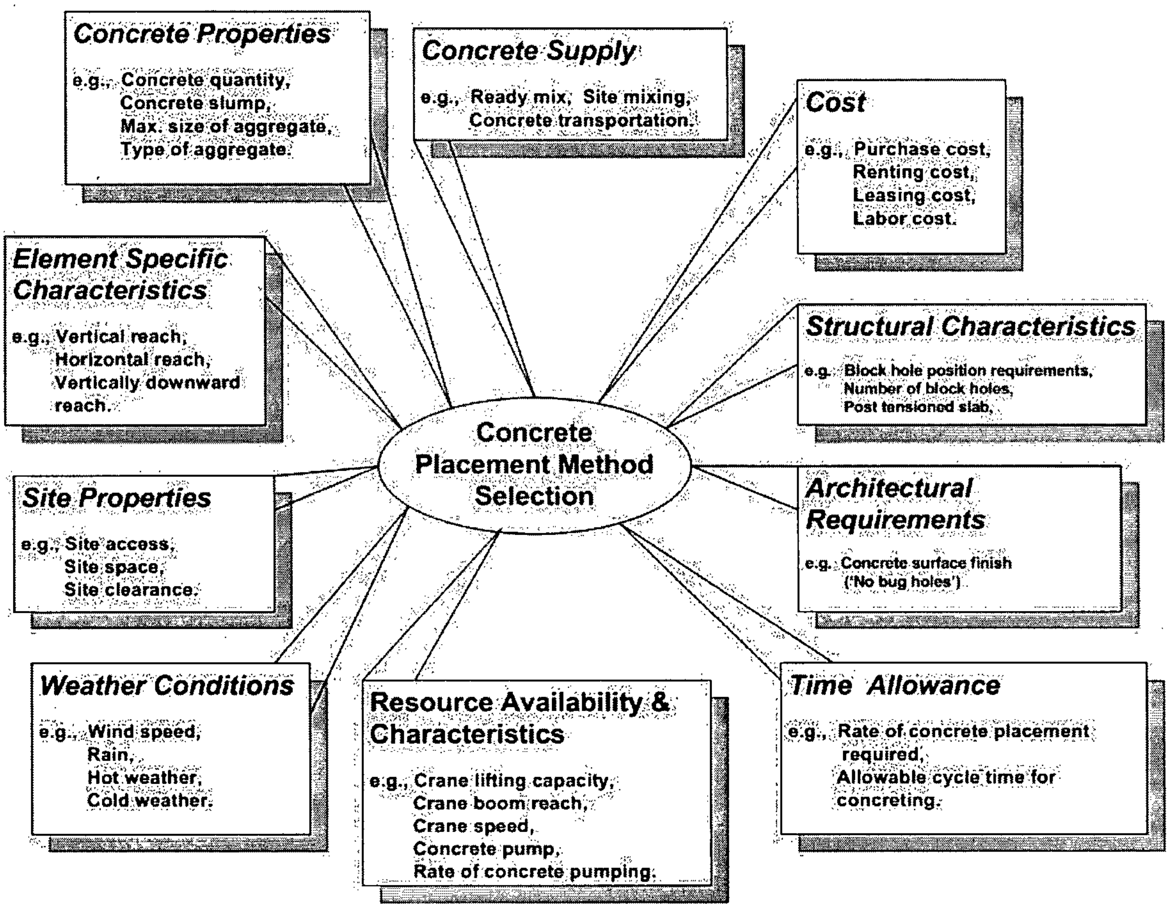


Figure 4.7. Factors affecting selection of concrete placement methods.

- Element Specific Characteristics

A concrete placement method's suitability is judged by element specific or physical component specific characteristic requirements such as maximum horizontal reach required, maximum vertical reach required, and maximum vertically downward reach required. These requirements are the limiting factors for selection of an appropriate concrete placement method (Gastaldo). For example, the Belt conveyor method can only place concrete up to an 87 ft vertical reach.

- Site Properties

Site properties are important for judging the suitability of a particular concrete placement method for a given project context. Concrete placement methods are characterized according to the resources they use. These resources, in turn, have their own feasibility requirements. For example, slickline pumping and separate placing boom use concrete line pumps. The footprints of these pumps require a certain parking space.

Further, placing boom pumps need a vertical unfolding height of 52 ft at site, which should be free from overhead electrical wires or any obstructions (Putzmeister, 2001a). Additional site requirements of ground conditions, safe distance from excavation ditches, site open space, equipment-cleaning space, etc. (Fisher, 1997) are also involved. Slickline pumping for high-rise construction needs a "base line" length to run the concrete pipeline on ground before raising the vertical pipe. The baseline length is required to generate the friction necessary to reduce backpressure on the pump and the site length required should be at least 150 ft (Crepas, 1985). Whenever the required site length is not available, a "basement loop" of at least 20 ft depth is required (Gastaldo).

- Concrete Supply

Concrete transportation methods play an important part in decision making about concrete placement method selection because the ability to achieve full performance of a concrete pump is limited by the supply of concrete. For example, for a concrete pump that can theoretically place concrete at a rate of 200 yd^3 per hr to perform at full capacity, it needs one 10-yd^3 capacity truck mixer every three minutes (Wallace, 1998). Therefore, availability of concrete transportation equipment such as trucks, truck mixers, and dumpers along with the necessary site access and space availability (Wallace, 1998), essentially limit the productivity of a concrete placement method.

- Weather Conditions

Hot weather concrete placement and cold weather concrete placement have their own guidelines to follow before and after concrete placement; however these seldom affect concrete method selection. On the other hand, weather conditions such as wind speed affect selection of a concrete placement method because it is unsafe to operate a placer boom pump when the wind speed is more that 77-km/ hr (ACPA, 2001). Similarly, the crane and bucket method cannot be used in windy weather conditions. Also, the belt conveyor method, unlike other methods, cannot be used for concrete placement in rainy conditions.

- Resource Availability & Characteristics

Resource availability and resource characteristics are important factors for concrete placement method selection. Similarly, crane-lifting capacity at the tip of its boom is important to judge the maximum capacity of a concrete bucket. Moreover, type of concrete pumps available and their rates of concrete pumping in turn dictate the rate of concrete placement by concrete pumping method.

- Cost

Various types of costs have to be taken into consideration during decision making for concrete placement method selection. The cost of purchasing, renting, or leasing equipment along with labor and operating costs need to be considered. Also, when comparing two concrete placement options such as crane and bucket method and separate placing boom, one has to take into account savings in overall project cost due to a shorter duration construction cycle (Harvell, 1991) (Gastaldo).

- Structural Characteristics

Structural characteristics of the facility also play an important part in the selection of a concrete placement method. The number and location of cranes on a jobsite depends on the geometric and structural characteristics of the project, which in turn determines feasibility of concrete placement by the crane and bucket method. When selecting the separate placing boom method one has to consider feasible locations of the blockout[1], which also depends upon structural characteristics of the building (Harvell, 1991). Similarly, the slickline-pumping method cannot be used when a floor slab is to be post tensioned as the pipe layout might disturb the post tensioning cable layout (Crepas, 1985).

- Architectural Requirements

Architectural requirements such as the surface finish quality required may also dictate concrete placement method selection. Often an architectural wall with heavy reinforcement congestion needs a bottom-up pumping method to achieve an architectural smooth finish with few if any "bug holes[2]" (Crepas, 1985). The bottom-up pumping method is a special type of slickline pumping method where the formwork is filled with concrete pumped from a shut-off valve located at the bottom of the formwork.

- Time Allowance

The available timeframe for concrete placement often dictates the concrete placement method. To achieve a shorter duration construction cycle a large quantity of concrete needs to be placed quickly, which demands the appropriate number of resources and productivity for the concrete placement method.

Based on the literature review and interviews with concrete sub contractors and suppliers, the feasibility parameters identified as shown in table 4.3. Implementation of some of the concrete placement knowledge collected within the M&RBS structure is shown in figure 4.8. The concrete placement method class containing various methods is shown along with the feasibility parameters and conditions.

---

[1] Blockout is the opening in the slab for the pedestal (or mast) of a separate placing boom.
[2] Bug holes are the voids formed during concrete placement.

Table 4.3. Table of Concrete placing method selection and feasibility factors knowledge.

Concrete Placement Methods

Method	Crane & Bucket Method	Belt Conveyor Method	Placer Boom Pumping Method	Slickline Pumping Method	Separate Placer Boom Method
Factors					
Site Characteristics					
Parking Space					
Length (ft.)	>= 30 + 8	>=45 + 30	>= 51+ 30	>= 25+ 30	>= 25+ 30
Width (ft.)	>= 15	>= 30	>=36	>= 30	>= 30
Area (sq.ft.)	>= 570	>= 2250	>= 2916 (comfortable is 5000 sq.ft.)	>=1650 (comfortable is 5000 sq.ft.)	>=1650 (comfortable is 5000 sq.ft.)
Other (Open Space Required)	No overhead obstructions due to electric lines		Unfolding height >= 52 ft	Open space + (bldg.width/2) >= 150 ft i.e. 'Baseline' length	Open space + (bldg.width/2) >= 150 ft i.e. 'Baseline' length
Structural Characteristics					
Concrete Properties					
Slump range (in.)	-	1 to 7	2 to 9	2 to 9	2 to 9
Slump loss (in.)	-	-	4	4	4
Max. Aggregate Size (in.)	-	-	<= 2.5	<= 2.5	<= 2.5
Equipment Reach					
Max. Vertical Reach (ft.)	-	<= 87	<= 188	<= 400- 600 ft	<= 400- 600 ft
Max. Horizontal Reach (ft.)	-	<= 150	<= 174	<= 1000-1200	<= 79-111
Max. Vertically Downward Reach (ft.)	-	<= 40	<= 137	-	-
Concrete Placement Rate					
Rate of Placement (c.y. /hr)	45-50	50 - 360	50 - 210	50 - 210	50 - 210
Production Characteristics					
Feasible Rate of placement (c.y. /hr)	<= 45	>= 50	85	65	50
Feasible concrete quantity required					
Vertical elements (c.y.)	-	-	50 to 65	-	50 to 65

Horizontal elements (c.y.)	-	-	104	-	104
Additional requirements					
Ground conditions	-	Firm leveled ground	Firm leveled ground / without excavation ditches	Firm leveled ground	Firm leveled ground
Crew (No. of Crew members)	8	8	8	8	8
Other		No obstruction due to overhead wires	Wind speed should be <= 70 km/hr	Thrust block and hydraulic diversion block need 45 ft x 30 ft space.	Thrust block and hydraulic diversion block need 45 ft x 30 ft space.

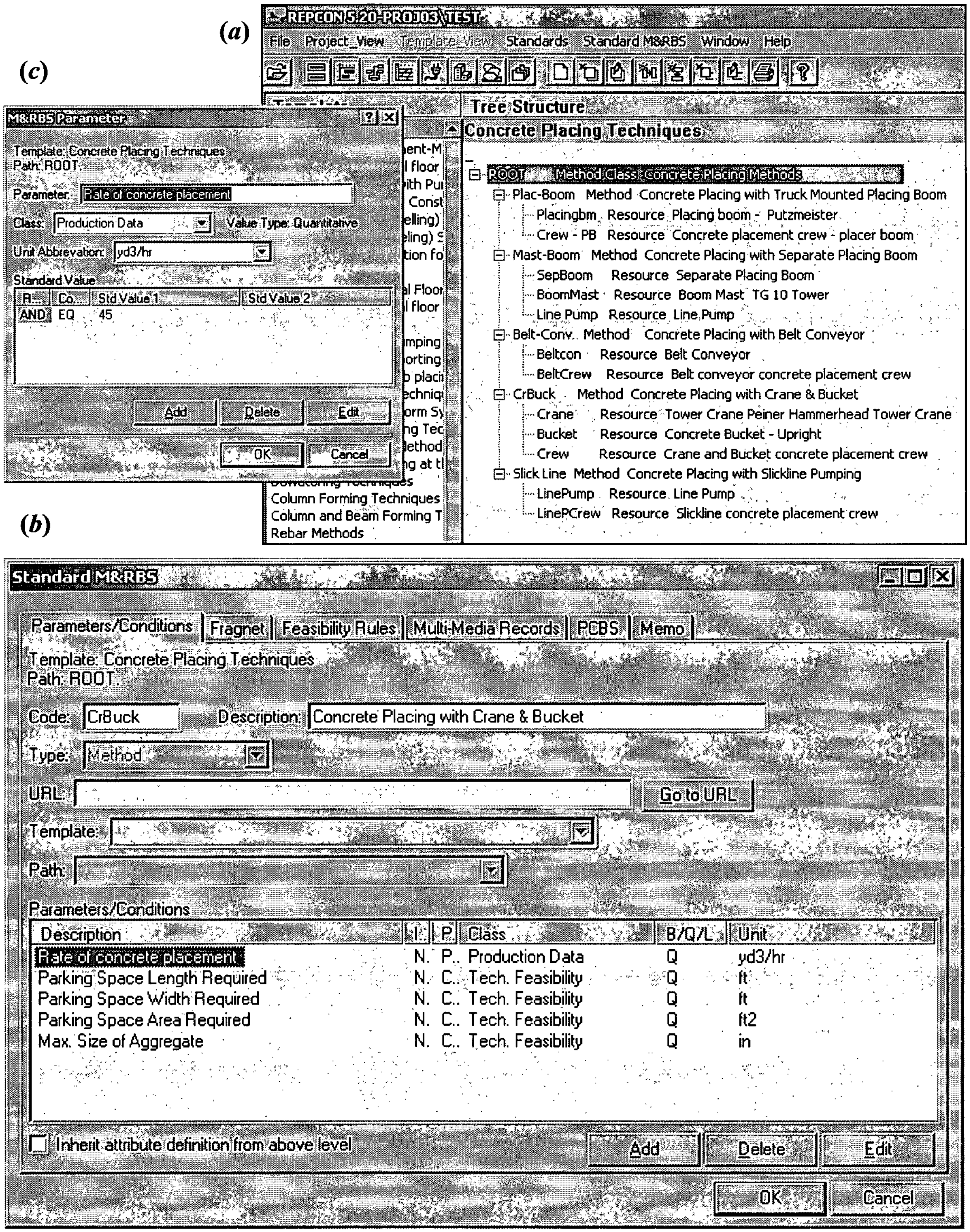


Figure 4.8. (*a*) Method Class "Concrete Placing Methods"; (*b*) Method "Concrete Placing with Crane & Bucket" with parameters and conditions; (*c*) Parameter "Rate of Concrete Placement" with value.

4.4 Rebar Placement Methods

For concrete construction, rebar placement is as important an activity as formwork and concrete placement. There are various views about the best way to carry out rebar fabrication and placement such as Reinforcement rationalization (Goodchild and Moss, 1999), Standardization of rebar (Theophilus, 1995), and Constructability considerations (Proverbs, Holt, and Olomolaiye, 1999). By and large, rebar placement methods can be classified into three categories such as Rebar assembly (i.e. onsite assembly from loose pre-cut and pre-bent rebar), Partial rebar prefabrication (i.e. partially prefabricated and partially assembled in place from loose pre-cut and pre-bent rebar), and Rebar prefabrication (i.e. totally prefabricated and placed on site). Proverb, Holt, and Olomolaiye (1999) classified rebar placement with one additional type of method i.e., "Bent and fabricated on site"[3]. Based on the literature review and interviews conducted with rebar contractors, suppliers, and detailers, we summarized various factors affecting the selection of rebar placement methods as described below. They are overviewed in figure 4.9.

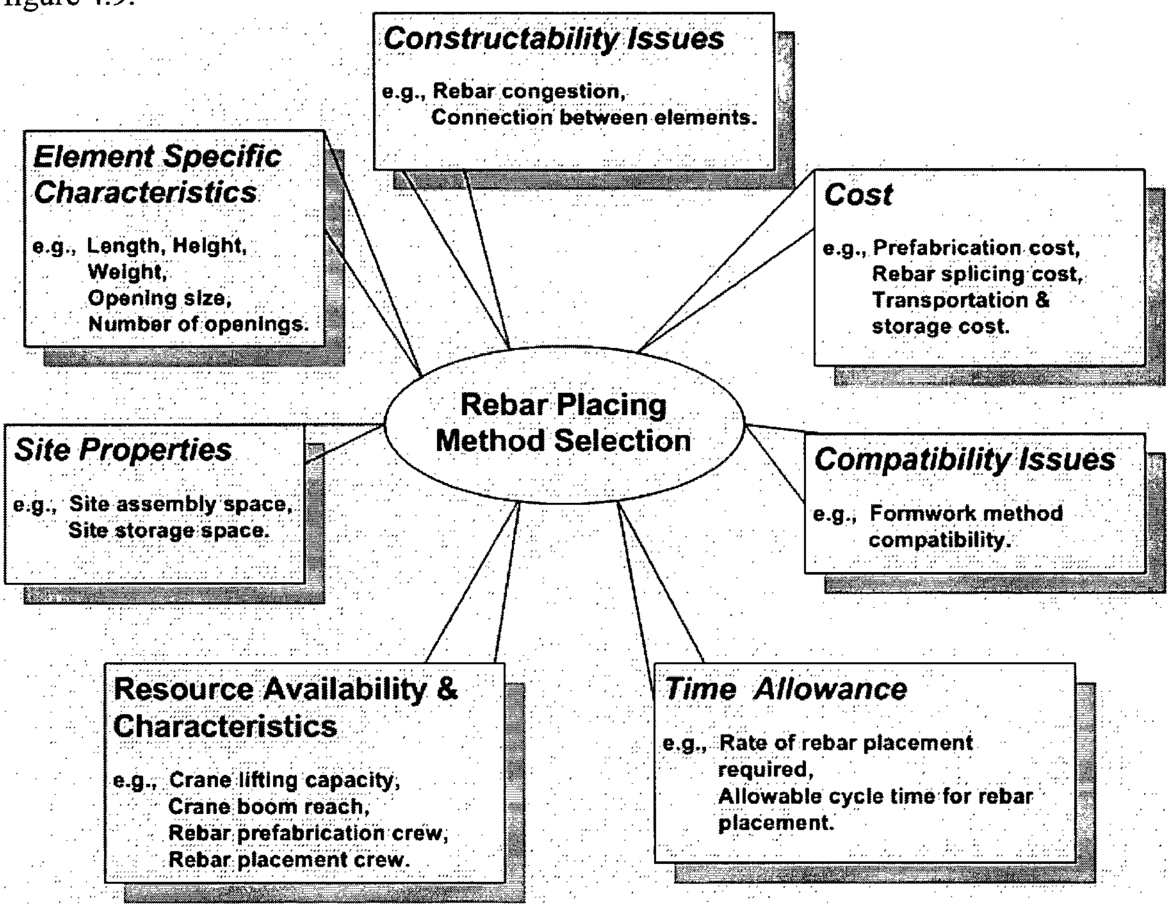


Figure 4.9. Factors affecting selection of rebar placement methods.

[3] During site visits and interviews with rebar sub contractors we observed that in high-rise construction the "onsite cutting and bending" method is rarely practiced. Therefore we assumed that the loose bars are precut and prebent in an offsite fabrication yard.

4.4.1 Description of Rebar Placement Methods

- Rebar Assembly

This is the most commonly used method of rebar placement on a construction site. Rebar cages for structural elements are assembled from loose pre-cut and pre-bent rebar. The elements such as slab, shear walls, and core walls are usually assembled using this method. The method requires sufficient site rebar storage area for pre-cut and pre-bent rebar. Assembling rebar for elements more than 8 ft high requires temporary scaffolding and safety harnesses, which results in lower overall productivity (Fradley).

- Partial Rebar Prefabrication

This is also a commonly used method for rebar placement on high-rise construction. Especially in earthquake prone zones such as Vancouver, the vertical shear reinforcement in core and shear walls needs to be staggered at alternate floors. These concentrated regions of rebar, generally two storeys high, are called as "zones" (Fradley) (Bitchel). To improve productivity and ease in assembly, these zones are essentially prefabricated and assembled in place along with loose bars (Fradley) (Bitchel). This method improves productivity of rebar placement for major shear elements such as cores and shear walls.

- Rebar Prefabrication

This is the total rebar prefabrication method. The rebar cage for the structural element is prefabricated. According to availability of site space, the prefabrication yard may be formed onsite or offsite. This method significantly improves rebar placement productivity, as the prefabricated cages only need to be lifted in place. By prefabricating rebar ahead of schedule the contractor can remove the rebar activity from the critical path of the project (Shaw). However, constructability issues need to be discussed before prefabrication. Generally, on high-rise construction, column rebar is totally prefabricated.

4.4.2 Factors Affecting Rebar Placement Method Selection

- Site Properties

Site storage area (length and width) and the site rebar assembly area (length and width) are important factors, which affect rebar placement method selection. The site should have sufficient storage area to store one truckload of rebar (Bitchel). For the case of on-site prefabrication of rebar cages for structural elements, the site space should be enough to store loose rebar, fabricate rebar cage, and to stack rebar cages i.e., the site should be at least 20 ft in width and 40 to 60 ft in length (Fradley). Moreover, prefabricated rebar mats for slab sections, walls sections, etc. need rebar storage area on site.

- Resource Availability and Characterization

Crane availability and its lifting capacity are the main factors, which determine the feasible rebar placement method on high-rise construction. Special arrangements such as lifting beams are required for transportation of prefabricated rebar cages (Shaw). For the cases of rebar prefabrication onsite or offsite, the rebar contractor needs two separate crews for rebar prefabrication and rebar placement.

- Element Specific Characteristics

Length, height, and width of the physical element are the decision-making parameters for rebar placement method selection. The length and height of the shear wall limits transportation options as well the handling ability of the crane. The weight of the rebar cage also affects selection of the rebar prefabrication method. Element specific characteristics such as "opening size" and "number of openings" also affect rebar prefabrication (Shaw). As described earlier, the regions of concentrated rebar i.e. "zones" need to be prefabricated, which also acts as an influencing factor for rebar placement method selection.

- Constructability Issues

Various constructability issues affect rebar placement method selection. Rationalization of flat slab reinforcement and constructability analysis can help rebar prefabrication. Various types of proprietary slab rebar mats are available, along with prefabricated punching shear reinforcement, which can help reduce 75 % of rebar fixing time and can save about 25 % in labor costs (Bennet and MacDonald, 1992). However, rebar congestion and the connection between elements such as beam, column, and slab make rebar prefabrication more expensive.

- Cost

Cost is the major consideration for rebar placement method selection. Rebar assembly is a cost effective method of rebar placement but requires more time and labor resources. Rebar prefabrication is a more costly method because of additional constructability considerations required by the detailer, as well as requirements of additional splices, and couplers. Moreover, an offsite prefabrication method involves significant additional transportation costs.

- Compatibility Issues

Compatibility issues arise between prefabricated rebar cages for slab mat reinforcement and punching shear reinforcement in terms of rebar spacing (BPG, 2001). Further, there can be rebar placement compatibility issues with the formwork method. For example, if the slipform method is to be used for wall or core forming, then the rebar placement method needs to take care of rebar lap staggering and use ninety-degree hooks instead of conventional hooks (Camellerie, 1978). Moreover, compatibility issues related to productivity also affect the selection of rebar placement method. The tunnel forming method needs higher productivity from rebar placement method, hence it is compatible with the rebar prefabrication method for wall rebar (deBruin and Fallowfield) (Wallace, 1985) (Quinton, 1991) (Prudhomme and Bradley, 1995) (basically, the faster the forming method used, the faster the rebar work has to be).

- Time Allowance

This is a very important determinant for the selection of the rebar placement method. Generally, in high-rise construction, to achieve a shorter construction cycle time, the contractor uses prefabricated column reinforcement, and the core and shear wall reinforcement are partially prefabricated in the form of zones. Depending upon the available timeframe for vertical and horizontal elements rebar placement, the constructability and cost considerations, the decision of partial or total prefabrication is made.

Method selection feasibility parameters gleaned from the literature and experts are summarized in table 4.4. Implementation of some of the rebar placing method knowledge is summarized in figure 4.10.

The summary of method selection feasibility factors, presented in tables 4.1 to 4.4 emphasize on technical feasibility aspects of construction methods, which are used to form production rules. In the next chapter we describe in detail the characterization scheme of these feasibility factors and the formation of production rules.

Table 4.4. Table of Rebar placement method selection and feasibility factors knowledge.

Reinforcement Placement Methods

Method	Column		Wall / Core			Slab	
	Rebar Assembly	Rebar Prefabrication	Rebar Assembly	Partial Rebar Prefabrication	Rebar Prefabrication	Rebar Assembly	Rebar Prefabrication
Factors							
Site Characteristics							
Site Storage & Assembly Space					-		
Length (ft.)	>= 60	(Max.Height + 4ft+ laps) *3	>= 60	>= 60	(Max.Length + 4ft+ laps) *3	>= 60	(Max.Length of slab section + 4ft+ laps) *3
Width (ft.)	>= 12	-	>= 12	>= 12	-	>= 12	(Max.width of slab section + 4ft+ laps)
Area (sq.ft.)	>= 720	>=1200	>= 720	>= 720	>=1200	>= 720	>=1200
Structural Characteristics							
Element							
Length (ft.)	-	-	-	-	<= 40	-	<= 40
Width (ft.)	-	-	-	-	-	-	<= 30
Height (ft.)	-	<= 30	-	-	-		
Weight (Tn.)	-	<= 2			<= 2		
Element components							
Openings							
Number openings	-	-	> 2	> 2	<= 2 (per 30 ft length)	-	-
Area / opening (sq.ft.)	-	-	> 12	> 12	<= 12	-	-

Production Characteristics							
Rate of production (Tn/manhr)	<= 0.07142	<= 0.125	<= 0.1	<= 0.1	<= 0.133, for core <= 0.1	<= 0.166	<= 0.32 (50 % less manhrs)
Resource Required							
Crew size (No. of crewmembers)	8	8	8	8	8	8	can be 2 placers
Other	Needs temporary scaffolding for rebar assembly reduces productivity.		Needs temporary scaffolding for rebar assembly reduces productivity.	The shear zones are prefabricated while rest of the rebar is assembled in place.	Compatible with Tunnel formwork method.	-	There should be atleast 20 identical slab sections.Compatible with Tunnel formwork method.

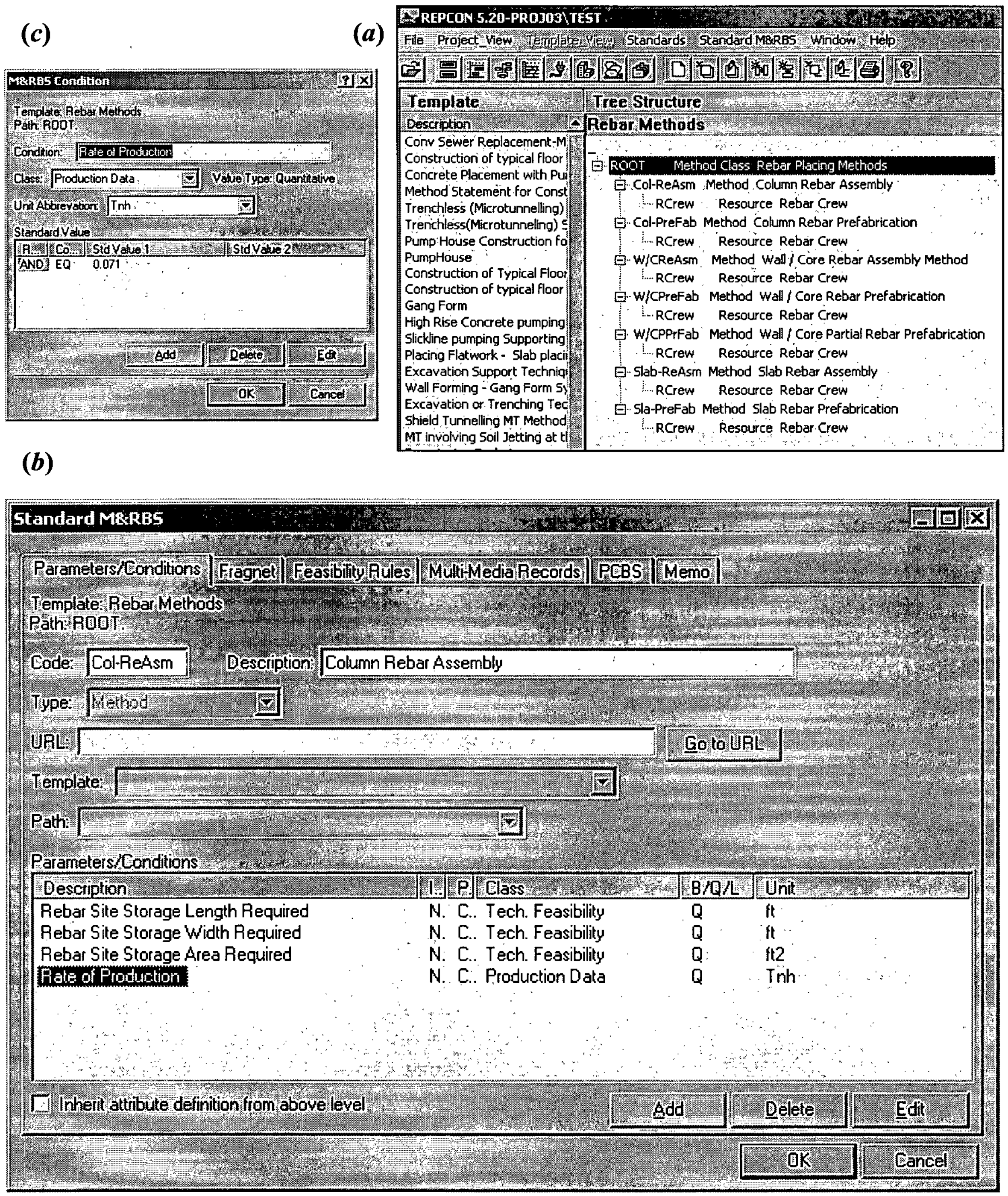


Figure 4.10. **(*a*)** Method Class "Rebar Placing Methods"; **(*b*)** Method "Column Rebar Assembly" with parameters and conditions; **(*c*)** Parameter "Rate of Production" with value.

Chapter 5. Rule Writing for Feasibility Checking

5.1 Introduction

This chapter describes encoding factors related to method selection and feasibility reasoning in a knowledge management tool. The knowledge representation scheme used to represent these factors is explained along with the available knowledge representation constructs in the CLIPS expert system. A number of issues related to the feasibility reasoning system are discussed and examples of feasibility reasoning rules are provided.

5.2 Method Selection and Feasibility Factors Characterization

We have categorized the factors affecting method selection and feasibility identified in Chapter 4, under three headings: Site characteristics, Structural characteristics, and Production characteristics. Our emphasis is on modeling declarative knowledge[1], which resides with construction personnel in the form of experience and rules-of-thumb regarding the technical feasibility of construction methods.

- Site characteristics

Site characteristics are defined as the properties of a jobsite location, which are relevant to the selection of a construction method and feasibility reasoning. As an example, the observation that sufficient site assembly space exists to allow Flying trusses to be assembled on-site helps ensure that at least one essential condition is met for this formwork method. In the case of concrete placement methods, jobsite space requirements are essential conditions for determining the feasibility of a particular method. For example, a Separate placing boom method for concrete placement is feasible when the jobsite has enough space for the baseline or the basement loop installation. Site characteristics of a jobsite location are expressed in terms of site storage space, site assembly space, parking space, and open space with their length, width, and area attributes.

- Structural characteristics

Structural characteristics are defined as the physical properties of the PCBS component and the M&RBS component, which are central for feasibility reasoning. Again, these properties constitute the necessary conditions of feasibility for the construction method of interest. For example, for application of a flying truss formwork system, a slab-bay width between 15 to 30 ft and slab-bay length of 22 ft is economical (Fischer, 1991). Thus the physical attributes of slab-bay i.e., Length and Width, become

[1] "Declarative knowledge is the surface level information that an expert can verbalize." In other words, declarative knowledge is the general heuristics available at a conscious level (McGraw and Harbinson-Briggs, 1989).

the factors for judging its economic feasibility of the Flying truss method. By listing these in the tables presented in Chapter 4, structural characteristics have been categorized into component characteristics, subcomponent characteristics, and pour characteristics for formwork methods. For concrete placement methods, structural characteristics are categorized as concrete properties, equipment reach, and concrete placement rate. For rebar placement methods, structural characteristics are categorized as component characteristics and subcomponent characteristics.

- Production characteristics

A method's production characteristics are arguably the most important factors affecting method selection. These factors include the method's productivity related aspects such as rate of production, minimum reuse required, crew type required, and minimum feasible quantity required. These factors act as necessary conditions for feasibility reasoning of a method.

As noted previously, tables 4.1 to 4.4 in Chapter 4 summarize the above-discussed feasibility factors in tabular format. The point to note here, however, is that not all the feasibility factors knowledge presented in Appendix-A can be listed in these tabular formats.

5.3 Knowledge Representation Scheme

A knowledge representation scheme is central to the development of a knowledge management tool, as it allows one to model method selection knowledge in a reusable format. There are two ways of representing knowledge: Procedural representation and Declarative representation (Adeli, 1988). In a procedural knowledge representation, knowledge is embedded in procedural code as pieces of information, which makes it difficult to update. In declarative knowledge representation, knowledge is stored in a knowledge base. In this thesis, we have used a declarative knowledge representation in the form of production rules about the feasibility of construction methods. The point to note here is that the declarative knowledge about the feasibility of a construction method comes from the relevant M&RBS component's parameters and conditions. We summarize various advantages we gain by using production rules as follows:

- Production rules are expressed in the form of "condition - action" pairs (Turban, 1998) that are easily understandable by the user.

- Production rules can be stored in separate rule repositories, which can be readily archived or updated.

- Each production rule is a piece of knowledge that can be developed and modified independently of other rules in the repository (Turban, 1998).

- Rule-based reasoning is appropriate for causal reasoning (Zizette, 1998) and allows the user to generate explanations by tracking the flow of inference.

- A production rule is a declarative rule which contains facts and "cause – effect" relationships (Zizette, 1998), where the facts are generally taken from the rule-based system's database (for our case the PCBS and M&RBS hierarchies). Thus by changing fact values one can update the rule based system.

Selecting an appropriate expert system for implementation of our knowledge representation scheme was an important task. Since our aim was to embed the inference engine of the expert system within the REPCON research system, the ability to integrate with this system and interoperability with C & C++ were important characteristics sought. A few commercially available expert systems with the foregoing characteristics were examined and tested for their ability to express domain knowledge and production rules. Based on this work, the CLIPS 6.2 expert system was selected.

5.4 CLIPS

The C Language Integrated Production System, i.e., CLIPS, is a rule-based and / or object based expert system developed by NASA's Johnson Space Center (CLIPS, 2002). It is a widely accepted expert system throughout government, industry, and academia, which is available as a "freeware" (which enhances its appeal because it makes the research system more readily accessible to other researchers). Because CLIPS has been written in C language, it can be easily embedded within a wide range of knowledge-based applications, and can be used in diverse computing environments.

Knowledge and information representation in CLIPS is made with the help of various constructs, which are described as follows:

Facts

Facts are the basic high-level forms of representing information, which can be asserted, retracted, modified and duplicated during run time. There are two types of facts i.e., ordered facts and non-ordered facts.

An "ordered" fact consists of a single symbol (relation name) followed by a sequence of zero or more fields (slots) separated by a space and delimited by an opening parenthesis on the left and a closing parenthesis on the right.

e.g., *(StoreyHeight 11 10 10 9 9 8 8)*

in which the first field *StoreyHeight* (relation name) is the "relation" applied to the remaining fields (slots) in the ordered fact.

A "non-ordered" fact provides the user with the ability to abstract the structure of a fact by assigning a name to each field (slot) in the fact. For example, a fact about the

PCBS component slab-bay can be expressed with the relation name "pcbs_component" and fields (slots) such as, name, description, and multi-slot attributes (in which attributes of slab-bay can be expressed as strings) as follows:

```
(pcbs_component
    (name "Slab-Bay")
    (description "The Superstructure Slab Bay")
    (attributes "Length" "Width" "Thickness")
)
```

The *Deffacts* construct is used to automatically assert a set of facts that are known before running a program. Any number of facts (either ordered or non-ordered) may be asserted into an initial fact-list by the *deffacts* construct using the "Reset" command in CLIPS. For example, the above stated "non-ordered" fact of *Slab-Bay* is asserted into the initial-fact list by using the *deffacts* construct as shown below:

```
(deffacts Pcbs-1
    (pcbs_component
    (name "Slab-Bay")
    (description "The Superstructure Slab Bay")
    (attributes "Length" "Width" "Thickness"))
)
```

Where *Pcbs-1* is the name of the *deffacts* construct.

The expression of PCBS data in terms of "facts" about PCBS components is used in feasibility reasoning about construction methods.

Templates

Before facts are created or defined, CLIPS needs to be informed about valid slots and corresponding valid value types for the relation name. The *deftemplate* construct is used to create a template, which can be used to access fields of the fact by name. For example, the template for defining / validating slab-bay "facts" can be listed with valid slots and their value types as follows.

```
(deftemplate pcbs_component
    (slot name (type STRING))
    (slot description (type STRING))
    (multislot attributes (type STRING))
)
```

This type of template can be used to define valid "facts" about the PCBS and M&RBS hierarchies for feasibility reasoning. The point to note here is that the "ordered" facts do not have a corresponding *deftemplate*. Whenever CLIPS encounters an ordered fact, it automatically creates an implied template (Giarratano and Riley, 1998).

Rules

These are the primary knowledge representation constructs composed of an antecedent (i.e. IF part or Left-hand-side of rule) and a consequent (i.e. THEN part or Right-hand-side of the rule). These constructs are useful for expressing method selection and feasibility knowledge. For example, a simple rule that checks for the existence of the PCBS component Slab-Bay can be written as

```
(defrule   check_for_Slabbay
      (pcbs_component (name  "Slab-Bay")
                                  (description  "The Superstructure Slab-Bay"))

=>
      (printout t  "The Superstructure Slab-Bay exists in Project PCBS" t)
 )
```

For this rule, the condition part checks if the fact *pcbs_component* with the corresponding name and description exists. If the result is true then the action part prints out the corresponding message.

Procedural Knowledge

Procedural knowledge representation constructs are similar to those of conventional programming languages such as PASCAL and C. These constructs include functions, generic functions, message-handlers, and modules. The procedural functions including "If – Then- Else", "While loops", and "Loop-for-count" help in the expression of the procedural part of feasibility reasoning knowledge such as checking for uniformity among PCBS components of a similar type.

Object Oriented Language

The Clips Object Oriented Language (COOL) paradigm includes abstraction, encapsulation, inheritance, polymorphism, and dynamic binding as the aspects of object oriented knowledge representation (CLIPS, 2002). COOL constructs such as, the *defclass* and *definstances* constructs are especially useful for expressing values associated with PCBS components, as explained in section 5.5.

5.5 Issues Related to Feasibility Reasoning System

Implementation of the rule based feasibility-reasoning system has used the above-described constructs for representation of the PCBS (i.e. the product model), M&RBS (i.e. the method model), and method selection knowledge. Issues related to this representation are discussed in the following sections of this chapter:

- Representation of PCBS (Section 5.6)

As described in Chapter 3, the project PCBS is a semantically predefined hierarchy of project elements. When representing this hierarchy using CLIPS syntax, we need to describe every project component along with its attributes and their values as well as the hierarchical relationship between these components.

- Representation of M&RBS (Section 5.7)

Similar to the PCBS, the M&RBS is also a semantically predefined hierarchy of method and resource components, described as a *method statement*. Thus to represent a project M&RBS, we need to express *method statement* with its constituent *operation*, *method*, and *resource* components along with their attributes and values using CLIPS syntax, as well as the hierarchical relationship between these components.

- Representation of Method Selection Knowledge (Section 5.8)

The method selection and feasibility factors knowledge available in tabular format can be modeled in terms of production rules. The syntax and heuristics of these production rules should be compatible with the representation schema of PCBS as well as M&RBS components or vice versa. Moreover, the rules should be modeled in such a way that they can be used on different projects.

5.6 CLIPS Template for Project PCBS

As stated previously, the Physical Component Breakdown Structure (PCBS) is a "quasi-hierarchical quasi-object oriented" (Udaipurwala, 1997) way of listing project components such as project, subproject, system, subsystem, elements and so on. These project components are hierarchically listed and can have a number of "parent node - child node" relationships.

The UML static structure diagram (UML, 2001) (Reed, 2000) depicting the association between project PCBS component, its attributes and their values, and corresponding locations is shown in figure 5.1. Every PCBS component is described as a class with name, path, code, description, type, and its corresponding physical attributes such as length, width, height, etc. Each attribute describing physical properties has its value described at a location or set of locations, thereby indicating presence of the element at that location or set of locations. This complex description of a physical component can be made with relative ease by using CLIPS's *deftemplate* construct and *defclass* construct.

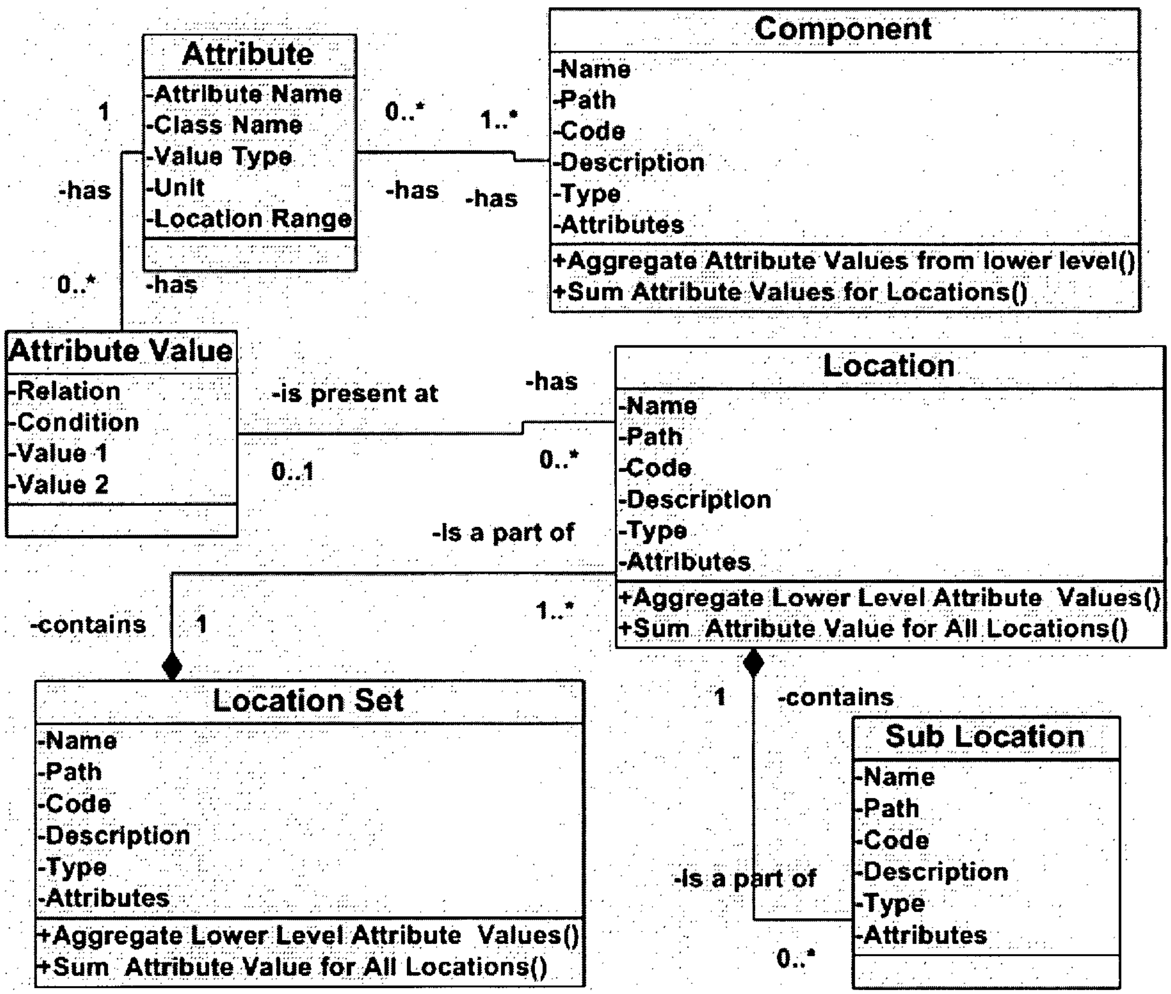


Figure 5.1. UML static structure diagram of the association between project PCBS component, its attributes and their values, and corresponding locations.

The *deftemplate* construct is useful for representing a structured and non-ordered fact such as a PCBS component. Each field in the fact is called as a *slot* or a *multislot* depending upon the type of value it stores. A component in the PCBS hierarchy is described by a name, code, path, description, component type, attributes, attribute type, and attribute values, which are expressed as *slots* or *multislots* in the template. The template structure appears as follows:

;;; Template for describing PCBS component ;;;

```
(deftemplate pcbs_component
 (slot name)
 (slot path)
 (slot code)
 (slot description)
 (slot component_type)
 (multislot attributes)
 (multislot attribute_type (default "Quantitative" "Boolean" "Linguistic"))
 (multislot attribute_values) )
```

Attribute values for a PCBS component are expressed with the help of the *defclass* construct. The class named *PCBS_DATA* is used to associate the attribute values with corresponding location ranges, while the *PCBS_VALUE* class is used to describe values of the attribute at each location. These classes are expressed as shown below.

```
;;; Class for giving location ranges to attributes values of PCBS component ;;;

(defclass PCBS_DATA (is-a USER)
 (role concrete) (pattern-match reactive)
 (slot unit (access read-write))
 (multislot location_list (access read-write))
 (multislot attribute_value_list (access read-write))
 )

;;; Class for giving values to attributes of PCBS component ;;;

(defclass PCBS_VALUE (is-a USER)
 (role concrete) (pattern-match reactive)
 (slot condition (access read-write))
 (slot value1 (access read-write))
 (slot value2 (access read-write))
 )
```

Here is an example fact for the PCBS component slab-bay.

The following *deffacts* construct is used to define a *pcbs_component* fact and associate attributes with corresponding instances of *PCBS_DATA* class (*attr1, attr2, etc.*). Component name, code, path, description, attribute type are also provided.

```
(deffacts pcbs_components1
 (pcbs_component
  (name "pcbs1")
  (code "SlBay")
  (path "GIA.Tower.SupSTR.HoriEle.Slab. SlBay")
  (description "Floor SlabBay1")
  (component_type "Subelement")
```

```
(attributes "Length" "Width" "Thickness" "Storey Height" "Shape"
            "SlabBay Supporting Sides are Parallel" "SlabBay Support is Uniform")
(attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative"
            "Linguistic" "Boolean" "Boolean")
(attribute_values [attr1] [attr2] [attr3] [attr4] [attr5] [attr6] [attr7])) )
```

The following construct *definstances DATA2* has instances of the class *PCBS_DATA* i.e., *"attr1* (*attr2*, *attr3*, etc.) *of PCBS_DATA"*, which are used to define the instances of class *PCBS_VALUE* (*aval1*, *aval2*, etc.) associated with a particular location (*"GFL"* in the present case). Units of the associated attribute values are also provided (e.g. *ft*).

```
(definstances DATA2
 (attr1 of PCBS_DATA (unit "ft") (location_list "GFL")
                                   (attribute_value_list [aval1]))
 (attr2 of PCBS_DATA (unit "ft") (location_list "GFL")
                                   (attribute_value_list [aval2]))
 (attr3 of PCBS_DATA (unit "ft") (location_list "GFL")
                                   (attribute_value_list [aval3]))
 (attr4 of PCBS_DATA (unit "ft") (location_list "GFL")
                                   (attribute_value_list [aval4]))
 (attr5 of PCBS_DATA             (location_list "GFL")
                                   (attribute_value_list [aval5]))
 (attr6 of PCBS_DATA             (location_list "GFL")
                                   (attribute_value_list [aval6]))
 (attr7 of PCBS_DATA             (location_list "GFL")
                                   (attribute_value_list [aval7]))
 )
```

The following construct *definstances DATA1* has instances of the class *PCBS_VALUE* i.e., *"aval1* (*aval2*, *aval3*, etc.) *of PCBS_VALUE"* used to define values of the attributes. The values are defined with associated condition (e.g. *EQ*).

```
(definstances DATA1
 (aval1 of PCBS_VALUE (condition "EQ") (value1 80.34))
 (aval2 of PCBS_VALUE (condition "EQ") (value1 32))
 (aval3 of PCBS_VALUE (condition "EQ") (value1 0.66))
 (aval4 of PCBS_VALUE (condition "EQ") (value1 11))
 (aval5 of PCBS_VALUE (condition "EQ") (value1 "Rectangular"))
 (aval6 of PCBS_VALUE (condition "EQ") (value1 "True"))
 (aval7 of PCBS_VALUE (condition "EQ") (value1 "False")))
```

In the foregoing example of PCBS fact, the instance named *"attr1"* in *deffacts* construct of *pcbs_component* has value instance named *"aval1"* referring to value which is 80.34 ft for location *"GFL"*.

5.7 CLIPS Template for M&RBS

Similar to a PCBS component, a M&RBS component can be expressed as a class with attributes such as name, path, code, description, type, attributes (i.e. parameters and / or conditions). Values are assigned to the attributes (i.e. parameter or condition) with corresponding conditions (i.e. EQ, LT, GT, NE, etc.). The static structure UML diagram is shown in figure 5.2.

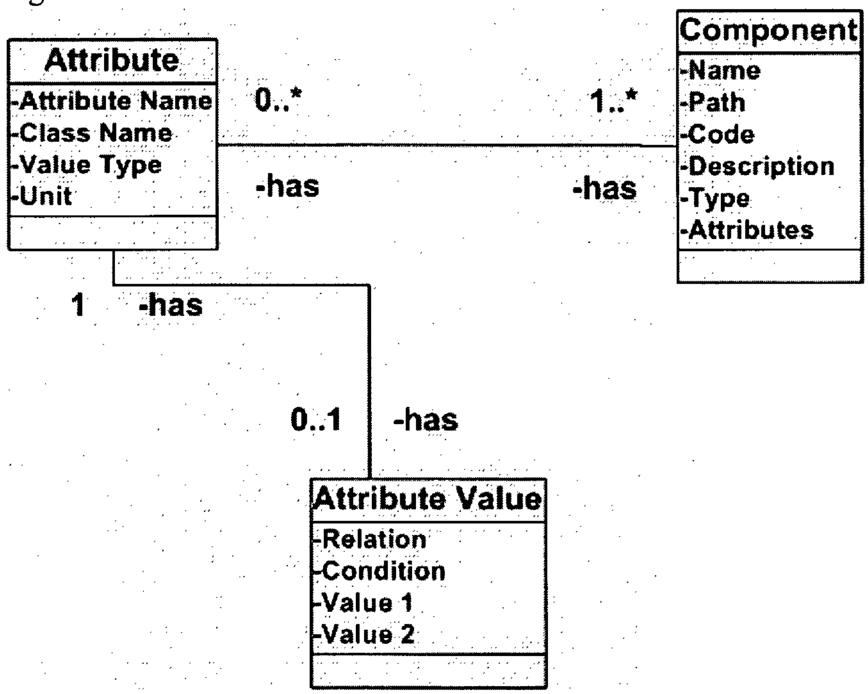


Figure 5.2. UML static structure diagram of the association between M&RBS component, its attributes, and their values.

Deftemplate and *defclass* constructs are used to describe a *method statement* and its constituent methods and resources. Since the concept of location does not apply to a M&RBS component, we need only one class object, i.e., *M&RBS_VALUE* class object for expressing attribute values with their conditions. A M&RBS template takes the following form:

;;; Template for describing M&RBS component ;;;

(deftemplate mrbs_component
(slot name)
(slot path)
(slot code)
(slot description)
(slot component_type)
(multislot attributes)

```
(multislot parameter_or_condition (default "Parameter" "Condition"))
(multislot attribute_type (default "Quantitative" "Boolean" "Linguistic"))
(multislot attribute_values)
)

;;; Class for giving values to attributes of M&RBS component ;;;

(defclass MRBS_VALUE (is-a USER)
 (role concrete) (pattern-match reactive)
 (slot unit (access read-write))
 (slot condition (access read-write))
 (slot value1 (access read-write))
 (slot value2 (access read-write))
 )
```

Here is an example of a M&RBS fact.

```
(deffacts mrbs_components80
 (mrbs_component
  (name "mrbs1")
  (code "WGang")
  (path "ROOT.FormCol.WGang ")
  (description "Column Formwork Method - Wooden Gangform")
  (component_type "Method")
  (attributes "Rate of Production" "Min. Reuse Required"
              "Storage Space Length Required" "Storage Space Width Required"
              "Allowable Rate of Pour" "Allowable Tie Spacing")
  (parameter_or_condition "Parameter" "Condition" "Condition"
                          "Condition" "Condition" "Condition")
  (attribute_type "Quantitative" "Quantitative" "Quantitative"
                  "Quantitative" "Quantitative" "Quantitative")
  (attribute_values [atval1] [atval2] [atval3] [atval4] [atval5] [atval6])
  )
 )
```

The construct *definstances DATA80* has the instances of class *M&RBS_VALUE* i.e., "*atval1* (*atval2*, *atval3*, etc.) *of MRBS_VALUE*", which are used to define values associated with them . The units and conditions associated with the values are also provided.

```
(definstances DATA80
 (atval1 of MRBS_VALUE (unit "ft2/mhr") (condition "EQ") (value1 35))
 (atval2 of MRBS_VALUE (unit "No.") (condition "EQ") (value1 30))
 (atval3 of MRBS_VALUE (unit "ft") (condition "EQ") (value1 50))
 (atval4 of MRBS_VALUE (unit "ft") (condition "EQ") (value1 30))
 (atval5 of MRBS_VALUE (unit "ft") (condition "EQ") (value1 8))
```

(atval6 of MRBS_VALUE (unit "ft") (condition "WR") (value1 2)(value2 3)))

In this example of a M&RBS fact about *method* Wooden Gang Form (for a column component), the value to the attribute i.e. parameter "Rate of Production" has an instance *"atval1 of MRBS_VALUE"* with value equal to 35 ft^2 / manhr. The M&RBS fact also provides code, path, description, attributes, parameter or condition, and type of attribute information about *method* Wooden Gang Form.

5.8 Expressing Hierarchical Relationships in CLIPS

We have expressed the hierarchical relationships in the PCBS and M&RBS using two basic CLIPS constructs: *deftemplate* and *defrule.* The *deftemplate* construct is used to define two types of component relationships i.e., parent component – child component relationship and ancestor component – descendant component relationship.

```
(deftemplate parent
        (slot parent_component)
        (slot child_component)
)

(deftemplate ancestor
            (slot ancestor_component)
            (slot descendant_component)
)
```

With the help of softcode[2] and predefined templates, the PCBS and M&RBS hierarchies are expressed in terms of facts. The example facts of PCBS component relationship are as shown below. First fact describes that the *parent_component "Slab"* has a *child_component* (i.e. child) *"SlBay"* (i.e. Slab-bay). Second fact depicts that the *parent_component "HorizontalEle"* (i.e. Horizontal element is a *subsystem* component) has a *child_component "Slab"*. Finally, the third fact shows that the *parent_component "SuperSTR"* (i.e. superstructure *system* component) has a *child_component "HorizontalEle"*.

```
(deffacts relationships_pcbs
          (parent (parent_component "Slab")(child_component "SlBay"))
          (parent (parent_component "HorizontalEle")(child_component "Slab"))
          (parent (parent_component "SuperSTR")
                                        (child_component "HorizontalEle"))
)
```

[2] The softcode is an external program written in C & C++ to provide REPCON data to the CLIPS environment in the form of "facts".

The following two rules are used for relationship interpretation.

```
;;; Rule to establish ancestor-descendant relationship;;;
;;; between parent component and child component ;;;

    (defrule ancestor1
            (parent (parent_component ?parent)
                                    (child_component ?child))
     =>
            (assert (ancestor (ancestor_component ?parent)
                                    (descendant_component  ?child)))
            )

;;; Rule to establish ancestor-descendant ;;;
;;; relationship between all other components ;;;

    (defrule ancestor2
            (parent (parent_component ?parent)
                                    (child_component ?child))
            (ancestor (ancestor_component ?comp)
                                    (descendant_component ?parent))
    =>
            (assert (ancestor (ancestor_component ?comp)
                                    (descendant_component  ?child)))
            )
```

The *ancestor1* rule says that "If there is any fact which has a parent component – child component relationship, then assert (i.e. make it a "fact") that the parent component is also the ancestor component of the child component." This rule establishes the ancestor-descendant relationship between project components to give "ancestor facts", which are used in the *ancestor2* rule along with "parent facts" to establish a complete hierarchical relationship. For example, slab-bay is a part of slab that in turn is a part of superstructure. Thus, the superstructure becomes the ancestor component of slab-bay as well as the slab. This type of relationship helps in causal analysis, as a slab-bay can also be present in the substructure system, which is different than the slab-bay being present in the superstructure system.

The foregoing constructs help provide a comprehensive description of the PCBS and M&RBS hierarchies with their components, component's attributes and values, and component inter-relationships. This description is available as universally accessible facts within the CLIPS environment, and can be interpreted with the help of production rules to determine the feasibility of a particular *method* or a set of methods i.e., a *method statement*.

5.9 Expressing Method Selection Knowledge in CLIPS

Method selection knowledge is present in the form of independent chunks of knowledge such as the feasibility parameters or conditions of methods. We have used the CLIPS *defrule* construct to form production rules about method selection. A rule in CLIPS is a "collection of conditions and the actions to be taken if the conditions are met". These rules are fired based on the existence of facts or instances. The inference engine fires rules by pattern-matching rule conditions with the existing fact-lists and/or instance-lists.

5.9.1 Examples of Rules and Their Modeling in CLIPS Syntax

- The rule for checking "uniformity of slab-bay" to test the applicability of a flying truss formwork method selection for high-rise construction can be stated as follows:

"If the slab-bay length and width is uniform for high-rise floors, and the available reuses are more than 6, then the Flying truss method is feasible for those floors."

The production rule performs the following checks: (1), the slab-bay belongs to the superstructure (see figure 5.3); (2), the operation in the *method statement* has Flying truss method (see figure 5.4); (3), the slab-bay that belongs to slab element is uniform in length and width for all of its locations; (4), the method has attribute "Min. Reuse Required". Finally the rule gives the result about the available reuses and feasibility of the Flying truss method.

Check 1:

```
(defrule slab_uniformity_reuses_check_flying_truss_slab

  (ancestor (ancestor_component ?ancestor1)(descendant_component ?descendant1))
  (pcbs_component (name ?ancestor1) (code ?code1))
  (pcbs_component (name ?descendant1) (code ?code2)(description ?desc2))
  (test (and(eq "SupSTR"(sub-string 1 6 ?code1))
                        (eq "SlBay"(sub-string 1 5 ?code2))))
```

The above stated condition checks that the *pcbs_component* with code ***"SlBay"*** (i.e. slab-bay) belongs to *pcbs_component* with code ***"SupSTR"*** (i.e. superstructure).

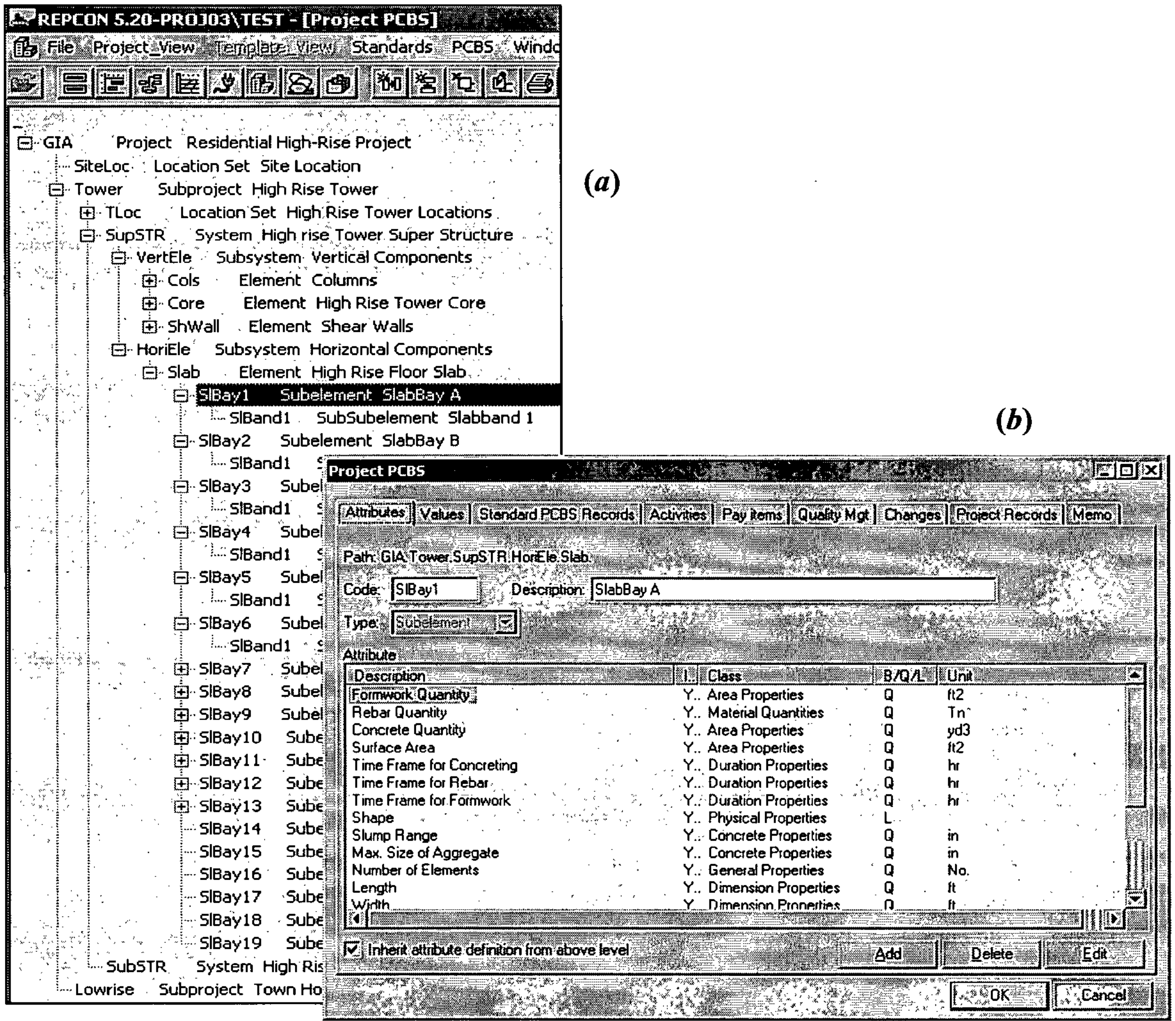


Figure 5.3. ***(a)*** PCBS hierarchy with slab-bay components belonging to superstructure; ***(b)*** Component "SlabBay A" with its attributes.

Check 2:

(ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
*(mrbs_component (name ?ancestor) (code ?**code3**))*
*(mrbs_component (name ?descendant) (code ?**code4**)(description ?desc1))*
*(test (and (eq "**FormSlab**"(sub-string 1 8 ?**code3**))*
*(eq "**FlTruss**"(sub-string 1 7 ?**code4**))))*

The condition checks that the *mrbs_component* with code *"FlTruss"* (i.e. method Flying Truss) belongs to *mrbs_component* with code ***"FormSlab"*** (i.e. operation Form Slab).

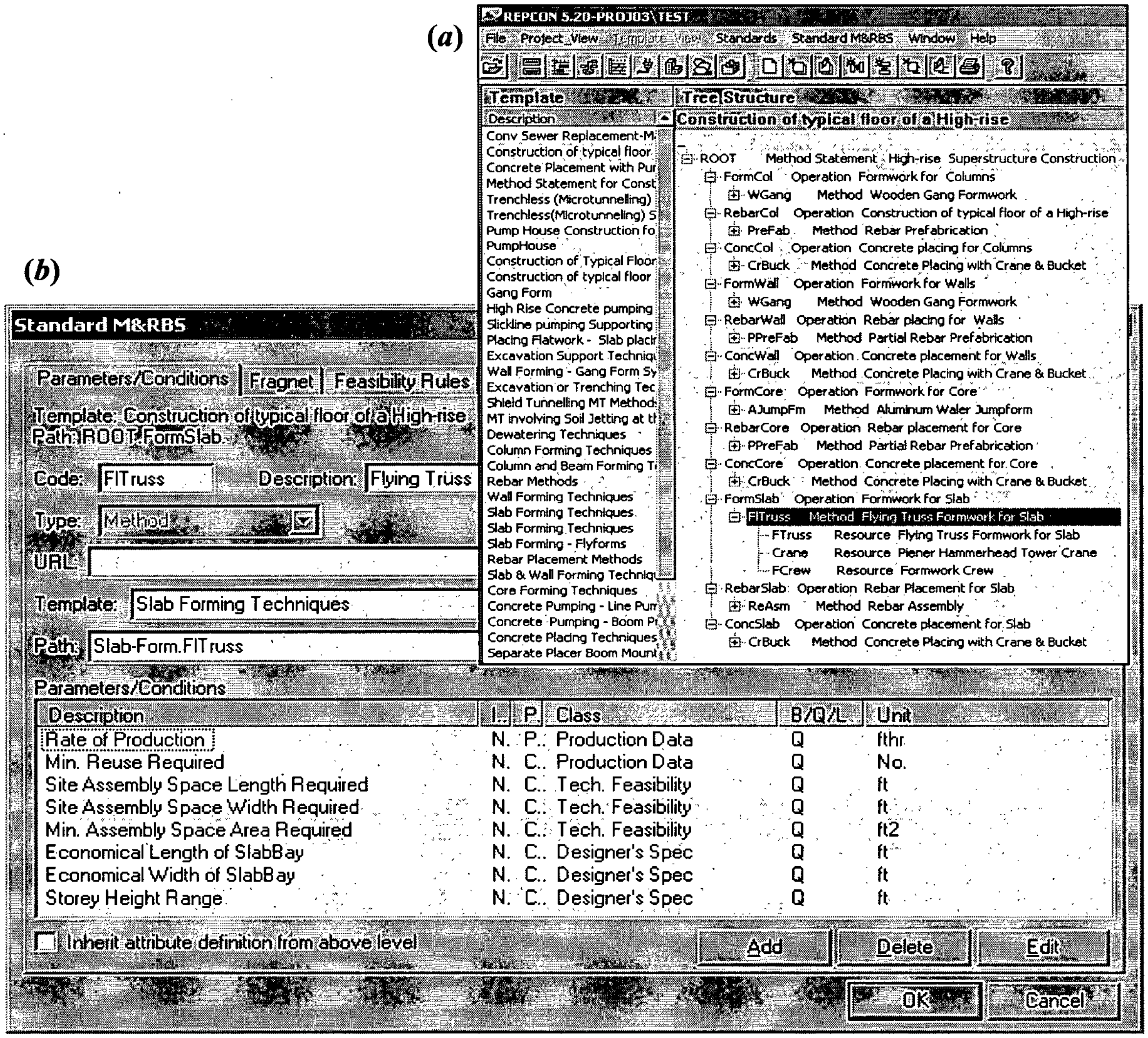


Figure 5.4. ***(a)*** M&RBS hierarchy with Method "Flying Truss Formwork" for Operation "Formwork for slab"; ***(b)*** Method "Flying Truss Formwork" with its parameters and conditions.

Check 3:

*(pcbs_component (name ?descendant1)(code ?**code2**)*
*(attributes **$?ahead "Length" $?atail**)*
*(attribute_values **$?avhead** ?**val1 $?avtail**))*
*(test (eq (length$ **$?ahead**)(length **$?avhead**)))*
*(test (eq (length$ **$?atail**)(length **$?avtail**)))*

```
(pcbs_component (name ?descendant1)(code ?code2)
                              (attributes $?ahead1 "Width" $?atail1)
                              (attribute_values $?avhead1 ?val2 $?avtail1))
    (test (eq (length$ $?ahead1)(length $?avhead1)))
    (test (eq (length$ $?atail1)(length $?avtail1)))

(pcbs_component (name ?descendant1)(code ?code2)
                              (attributes $?ahead2 "Shape" $?atail2)
                              (attribute_values $?avhead2 ?val3 $?avtail2))
    (test (eq (length$ $?ahead2)(length $?avhead2)))
    (test (eq (length$ $?atail2)(length $?avtail2)))
```

The condition (Check 3) tests that the *pcbs_component* with code *"**code2**"* (i.e. slab-bay) has attributes ***Length***, ***Width***, and ***Shape***. The ***val1***, ***val2***, and ***val3*** are the temporary variables storing the values of these attributes. In the above condition (Check 3) the wild cards[3] ***$?ahead***, ***$?atail***, ***$?avhead***, and ***$?avtail*** are used for attribute ***Length***. By testing equality of ***$?ahead*** and ***$?avhead*** as well as ***$?atail*** and ***$?avtail*** the condition ensures that the temporary variable ***val1*** refers to the appropriate attribute value.

Check 4:

```
(mrbs_component (name ?descendant)(code ?code4)
                              (attributes $?ahead3"Min. Reuse Required" $?atail3)
                              (attribute_values $?avhead3 ?val4 $?avtail3))
    (test (eq (length$ $?ahead3)(length $?avhead3)))
    (test (eq (length$ $?atail3)(length $?avtail3)))
```

The above portion of the condition checks that the *mrbs_component* with code *"**code4**"* has attribute (i.e. parameter or condition) named *"**Min. Reuse Required**"*. The *"**val4**"* is the pointer to the attribute value.

The action part of the rule is expressed as follows:

```
=>
(bind ?lmax (length$ (send ?val1 get-location_list)))
(bind ?a1 (send (nth$ 1 (send ?val1 get-attribute_value_list)) get-value1))
(bind ?a2 (send (nth$ 1 (send ?val2 get-attribute_value_list)) get-value1))
(bind ?a3 (send (nth$ 1 (send ?val3 get-attribute_value_list)) get-value1))
(bind ?a4 (send ?val4 get-value1))
(bind ?l1 ?lmax)
(bind ?reuse 0)
```

[3] The multifield wildcards denoted by a dollar sign followed by a question mark ($?), matches any value in zero or more fields in a pattern entity.

The above part of the action binds the values from the pointers to temporary variables, e.g., ***lmax*** stores the number of locations for the attribute ***Length***, ***a1*** stores the value of *Length* for first location in the location list, ***a2*** stores the value of *Width* for first location in the location list, ***a3*** stores the value of *Shape* for first location in the location list.

```
(while (> ?l1 0)
 (bind ?a5 (send (nth$ (- ?lmax(- ?l1 1))
                         (send ?val1 get-attribute_value_list)) get-value1))
 (bind ?a6 (send (nth$ (- ?lmax(- ?l1 1))
                         (send ?val2 get-attribute_value_list)) get-value1))
 (bind ?a7 (send (nth$ (- ?lmax(- ?l1 1))
                         (send ?val3 get-attribute_value_list)) get-value1))
 (if (and (eq  ?a1 ?a5)(eq  ?a2 ?a6)(eq  ?a3 ?a7) )
       then    (bind ?reuse (+ ?reuse 1)))
  (bind ?a1 ?a5)
  (bind ?a2 ?a6)
  (bind ?a3 ?a7)
 (bind ?l1 (- ?l1 1)))
```

The above stated portion of the action is the procedural function *While loop*. It checks the Length, Width, and Shape values for slab-bay for all of its locations and counts reuses.

```
(if (not(>= ?reuse ?a4))
  then (printout formworkFile "The Method \""?desc1 "\" for PCBS component
    \""?desc2 "\"" crlf " is infeasible due to insufficient reuses " ?reuse "." crlf crlf)

  else (printout formworkFile "The Method \""?desc1 "\" for PCBS component
    \""?desc2 "\"" crlf  " is feasible due to sufficient reuses " ?reuse "." crlf crlf))
 )
```

The final portion of action the procedural function *If-Then-Else,* checks to see if the available reuses are greater than or less than *"Min. Reuse Required"*. It then prints the result about feasibility of the method to a separate text file named *"formworkFile"*.

- The rule for checking feasibility of Separate placing boom method for core concrete placement can be stated as follows:

"The separate placing boom method is feasible when concrete volume for vertical elements is at least 40 to 50 m^3, i.e., 52 to 65 yd^3 and concrete volume per floor is 235 yd^3."

The production rule performs the following checks: (1) the core belongs to superstructure (see figure 5.5); (2) the superstructure has *subsystem* vertical elements (see figure 5.6); (3) the *operation* has *method* Separate placing boom; (4) the superstructure and vertical elements and the method have the required attributes. The point to note here is that even though this rule is used to check feasibility of Separate placing boom method for core *element*, according to the feasibility factor knowledge we had, the concrete quantity of vertical element *subsystem* (i.e. parent node of core *element*) is used, which is a sum of concrete quantities for all the vertical elements under it (i.e. column, wall, core, etc.). This is partly because the method does not have any specific feasible concrete quantity requirement for core elements as it has in case of vertical elements (in general) irrespective of their types.

Check 1:

```
(defrule feasibility_concrete_placement_separate_placing_boom_core

  (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
  (pcbs_component (name ?ancestor) (code ?code1))
  (pcbs_component (name ?descendant) (code ?code2))
  (test (and(eq "SupSTR"(sub-string 1 6 ?code1))
                                  (eq "Core"(sub-string 1 4 ?code2))))
```

This condition checks that the *pcbs_component* with code "***code2***" (i.e. core) belongs to *pcbs_component* with code "***code1***" (i.e. superstructure).

Check 2:

```
  (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant1))
  (pcbs_component (name ?descendant1) (code ?code3))
  (test (eq "VertEle" (sub-string 1 7 ?code3)))
```

This condition checks that the *pcbs_component* with code "***code3***" (i.e. vertical element system) belongs to *pcbs_component* with code "***code1***" (i.e. superstructure).

Check 3:

(ancestor(ancestor_component ?ancestor1) (descendant_component ?descendant2))
*(mrbs_component (name ?ancestor1) (code ?**code5**))*
*(mrbs_component (name ?descendant2) (code ?**code6**))*
*(test (and (eq **"ConcCol"**(sub-string 1 7 ?**code5**))*
*(eq **"SpBoom"**(sub-string 1 6 ?**code6**))))*

This condition checks that the *mrbs_component* with code *"**code6**"* (i.e. Separate placing boom method) belongs to *mrbs_component* with code *"**code5**"* (i.e. Operation concrete core).

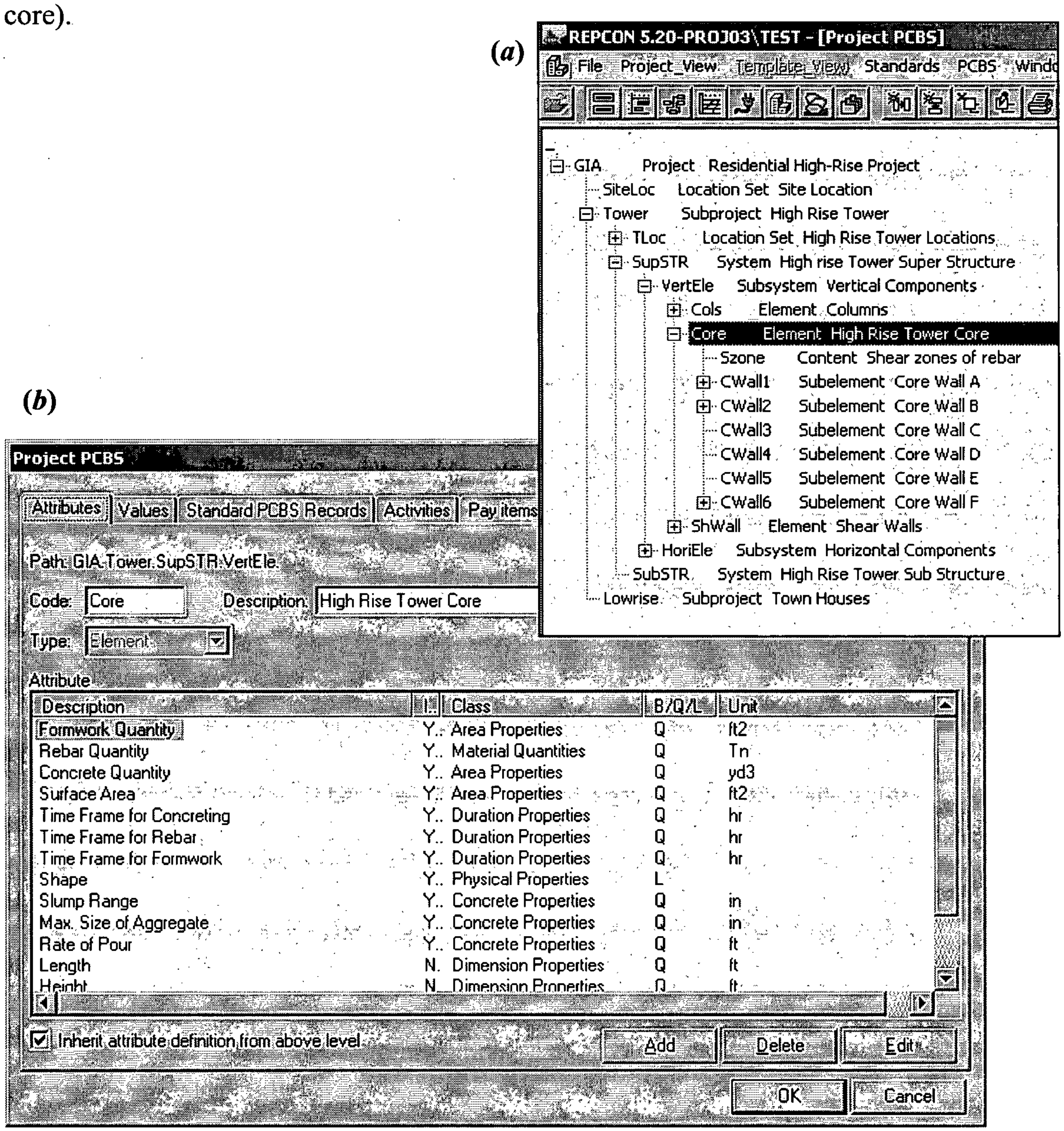


Figure 5.5. (*a*) PCBS hierarchy with component "Core" belonging to superstructure; **(*b*)** Component "Core" with its attributes.

(*a*)

(*b*)

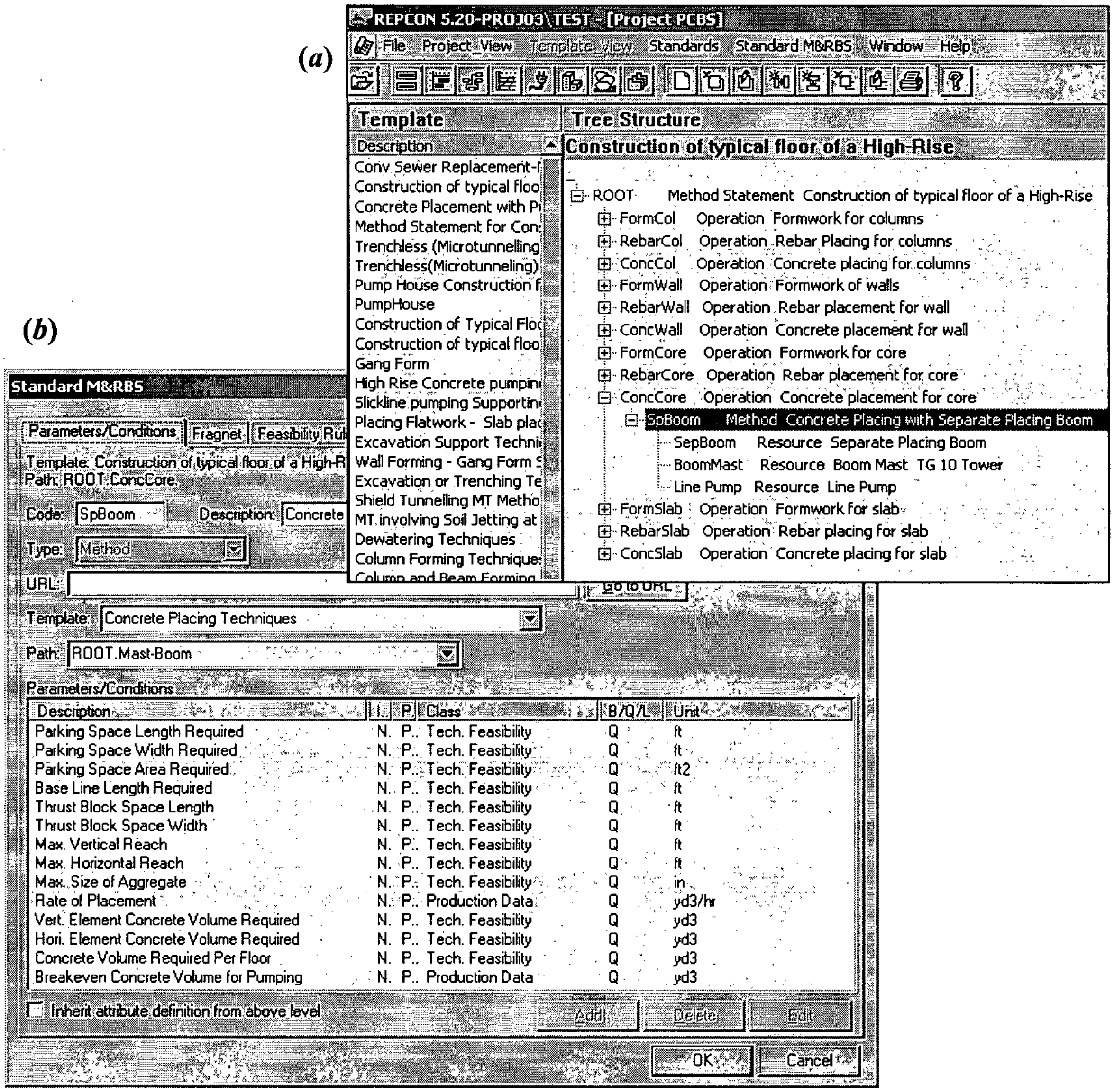


Figure 5.6. (***a***) M&RBS hierarchy with Method "Separate Placing Boom" for Operation " Concrete placement for core"; (***b***) Method "Separate Placing Boom" with its parameters and conditions.

Check 3:

*(pcbs_component (name ?ancestor)(code ?**code1**)*
*(attributes $?ahead1 **"Concrete Quantity"** $?atail1)*
*(attribute_values $?avhead1 ?**val1** $?avtail1))*
(test (eq (length$ $?ahead1)(length $?avhead1)))
(test (eq (length$ $?atail1)(length $?avtail1)))

```
(pcbs_component(name ?descendant1)(code ?code3)
                              (attributes $?ahead2 "Concrete Quantity" $?atail2)
                             (attribute_values $?avhead2 ?val2 $?avtail2))
(test (eq (length$ $?ahead2)(length $?avhead2)))
(test (eq (length$ $?atail2)(length $?avtail2)))

(mrbs_component(name ?descendant2)(code ?code6)(description ?desc2)
        (attributes $?ahead5 "Feasible Concrete Quantity For Verticals" $?atail5)
        (attribute_values $?avhead5 ?val5 $?avtail5))
(test (eq (length$ $?ahead5)(length $?avhead5)))
(test (eq (length$ $?atail5)(length $?avtail5)))

(mrbs_component (name ?descendant2)(code ?code6)(description ?desc2)
            (attributes $?ahead6 "Feasible Concrete Quantity Per Cycle" $?atail6)
            (attribute_values $?avhead6 ?val6 $?avtail6))
(test (eq (length$ $?ahead6)(length $?avhead6)))
(test (eq (length$ $?atail6)(length $?avtail6)))
```

The condition checks that the *mrbs_component* with code *"**code6**"* (i.e. separate placing boom method) has attributes (i.e. parameters or conditions) named ***"Feasible Concrete Quantity For Verticals"*** *and* ***"Feasible Concrete Quantity Per Cycle"***. The condition also checks that the *pcbs_component* with code *"**code1**"* (i.e. superstructure) and code *"**code3**"* (i.e. vertical element system) has an attribute named ***"Concrete Quantity"***. As explained previously the use of wildcards (*ahead1*, *atail1*, *avhead1*, *avtail1*, etc.) ensures that the appropriate attribute value is assigned to the attribute value pointer (*val1*, *val2*, *val3*, etc.).

The action part of the rule is a follows:

```
=>
(bind $?list1 (create$))
(bind $?list2 (create$))
(bind $?list3 (create$))
(bind $?list4 (create$))
(bind ?a5 (send ?val5 get-value1))
(bind ?a6 (send ?val6 get-value1))
(bind ?lmax (length$ (send ?val1 get-location_list)))
(bind ?l1 ?lmax)
```

This above part of the action binds the values from the pointers to temporary variables and creates empty lists (***list1***, ***list2***, etc.) to display results in the form of strings.

```
(while (> ?l1 0)
```

```
(bind ?a1 (send (nth$ (- ?lmax(- ?l1 1))
          (send ?val1 get-attribute_value_list)) get-value1))
(bind ?a2 (send (nth$ (- ?lmax(- ?l1 1))
          (send ?val2 get-attribute_value_list)) get-value1))
(if (< ?a2 ?a5)
 then (bind ?temp1 1)
      (bind $?list1 (create$ $?list1 (nth$ (- ?lmax(- ?l1 1))
                                      (send ?val1 get-location_list))))

 else (bind $?list2 (create$ $?list2 (nth$ (- ?lmax(- ?l1 1))
                                      (send ?val1 get-location_list)))))
(if (< ?a1 ?a6)
 then (bind ?temp2 1)
      (bind $?list3 (create$ $?list3 (nth$ (- ?lmax(- ?l1 1))
                                      (send ?val2 get-location_list))))

 else (bind $?list4 (create$ $?list4 (nth$ (- ?lmax(- ?l1 1))
                                      (send ?val2 get-location_list)))))
(bind ?l1 (- ?l1 1)))
```

In the forgoing, the procedural function *While loop* checks the attribute values to determine the feasibility of Separate placing boom method for core concrete placement.

```
(if (eq ?temp1 1)
  then (printout t "The Method \""?desc1 "\" is infeasible due to "
       " concrete quantity for Verticals is less than required "?a5" at location "
            $?list1 "." t)

  else (printout t "The Method \""?desc1 "\" is feasible considering "
            " concrete quantity for Verticals at location " $?list2 "." t ))

(if (eq ?temp2 1)
  then (printout t "The Method \""?desc1 "\" is infeasible due to "
  " concrete quantity for whole construction cycle is less than required "?a6" at
  location " $?list3 "." t)

  else (printout t "The Method \""?desc1 "\" is feasible considering "
   " concrete quantity for whole construction cycle at location " $?list4 "." t)))
```

Finally the rule prints out the list of feasible and infeasible locations, which can help the user to make the decision regarding selection of Separate placing boom considering the concrete quantity to be placed.

Other feasibility checking production rules can be formed using the feasibility factors knowledge, which is listed in Appendix-A.

Several closing observations are offered here. First, once the user becomes familiar with CLIPS' syntax, it is reasonably easy, if somewhat lengthy to formulate meaningful feasibility rules. Second, for the approach to be applicable in practice, an interface for expressing rules in more natural language needs to be developed. Third, having to express feasibility checks in the form of rules is of great assistance in making explicit the accumulated knowledge and experience of construction personnel. And finally, there is a need of modeling "judgment" in the formulation of rules. Actual feasibility reasoning is not always black and white. For example, a condition may not be fulfilled at every location instance; it may be sufficient that it is met at "most" instances.

Chapter 6. Reasoning Schema

6.1 Overview

Explained in this chapter is the schematic representation of rule based reasoning for method selection and feasibility analysis. The steps need to be performed in rule based reasoning are also elaborated.

6.2 Objectives of Reasoning Schema

The major obstacle to overcome in the reasoning schema was "mapping" the PCBS and M&RBS hierarchies. These hierarchies allow significant flexibility in their configuration. They can vary not only in terms of the desired scope of reasoning i.e., superstructure, substructure, or the whole project, but also according to the type of construction project such as repetitive or non-repetitive project. Thus, the reasoning schema should be general enough to handle various project scenarios. The association of production rules either with a method or a *method statement* is an important issue because of the flexible nature of the hierarchies.

6.3 Reasoning Schema

Reasoning approaches may be classified in two categories, namely, "Bottom-up" approach and "Top-down" approach. The bottom-up approach involves feasibility reasoning during individual method and resource selection. The rule files associated with individual methods or resources could be triggered while copying a method or resource over to the *method statement* hierarchy. However, this approach requires a large amount of contextual information to be embedded in the feasibility rules. For example, for selection of a particular method or resource, one has to check the operation and the *method statement* context in which it will be used. A resource can be used by a number of methods e.g., tower crane is used by Crane & Bucket concrete placement method as well as it is used by Flying Truss formwork Method. Feasibility checking for both scenarios simultaneously is difficult to achieve. Compatibility of construction methods is certainly an important issue; for example, to achieve faster construction cycle using tunnel formwork the wall rebar is generally prefabricated. Such method and resource compatibility checks are difficult to perform in the bottom-up approach.

We have therefore used a "Top-down" reasoning approach i.e. the *method statement* reasoning approach. A predefined *method statement* embodies the necessary contextual information by defining its scope and level of generality. Checking of the resource and method compatibility, in the current implementation, is left to the user. The *method statement* reasoning schema treats *method statement* as the basic unit for feasibility reasoning. As discussed in section 3.3.2, a *method statement* is comprised of operations, methods, and resources. Every *method statement* has one rule file associated with it, which contains all the production rules regarding feasibility of its constituent methods and resources for the context represented by the *method statement.* This is important to note, as both methods and resources can be applicable to a variety of contexts. It is the context that dictates what properties of the construction methods and resources are relevant to the project components at hand.

Discussed in the following sections are selected issues and the following steps involved in the *method statement* reasoning schema:

1. Formation of Project PCBS Hierarchy;
2. Exposing Project PCBS Facts, Instances, and Relationships to CLIPS;
3. Formation of Method Statement Hierarchy;
4. Formation of Method Statement Rule File;
5. Exposing M&RBS Facts, Instances, and Relationships to CLIPS;
6. Reasoning with CLIPS Inference Engine;
7. Result Analysis and Modifications to the Method Statement.

The schematic diagram of steps performed while reasoning is shown in figure 6.1.

6.3.1 Formation of Project PCBS Hierarchy

The formation of the PCBS hierarchy has been discussed in section 3.2.2.

6.3.2 Exposing Project PCBS Facts, Instances, and Relationships to CLIPS

Converting the Project PCBS hierarchy to CLIPS syntax is a crucial step for feasibility reasoning. The data from REPCON's PCBS data structure must be expressed in predefined PCBS template format so that the CLIPS inference engine can validate and define the data as facts and instances in its working memory. These facts and instances in the working memory are used in the evaluation of production rules regarding feasibility of the construction methods. The hierarchical relationships of parent-child and ancestor-descendant relationships also need to be expressed in a predefined relationship template format, as discussed in section 5.8.

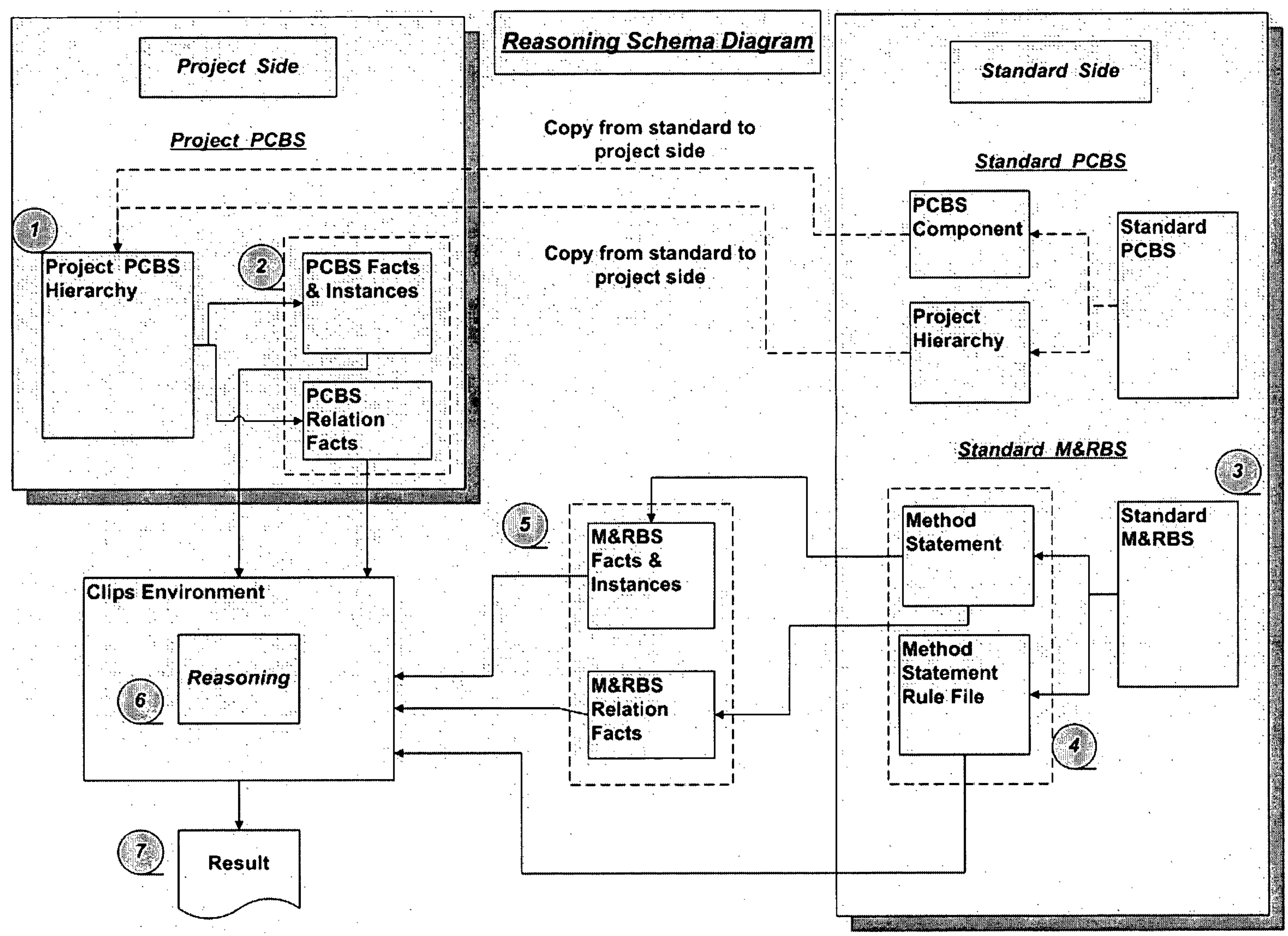


Figure 6.1. Reasoning Schema Diagram.

Subroutines coded in C and C++ have been used to expose the PCBS data structure to CLIPS as shown in figure 6.2. "Softcode 1" is used for exposing PCBS facts and instances, while "Softcode 2" is used for exposing PCBS component relationships. These softcodes produce two separate text files, which are then loaded in to the CLIPS environment.

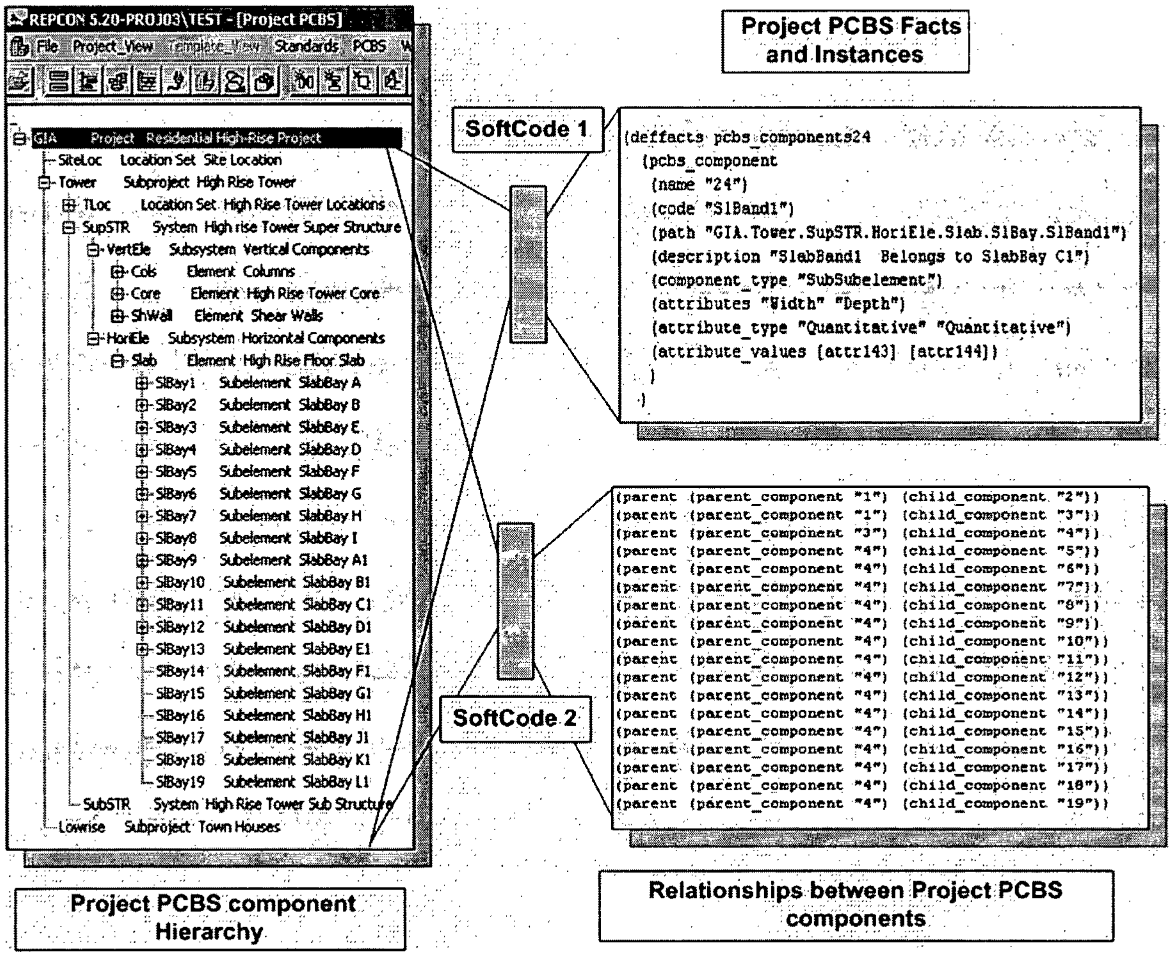


Figure 6.2. Exporting PCBS components in terms of facts, instances, and relationships to CLIPS.

6.3.3 Formation of Method Statement Hierarchy

The formation of M&RBS hierarchy is described in section 3.3.2.

6.3.4 Formation of Method Statement Rule File

The formation of the *method statement* rule file is performed in tandem with the formation of the *method statement.* As shown in the object model diagram of *method statement* (figure 6.3), only one rule file is associated with a *method statement.* This rule file contains production rules related to the constituent methods and resources for the application context of the *method statement.*

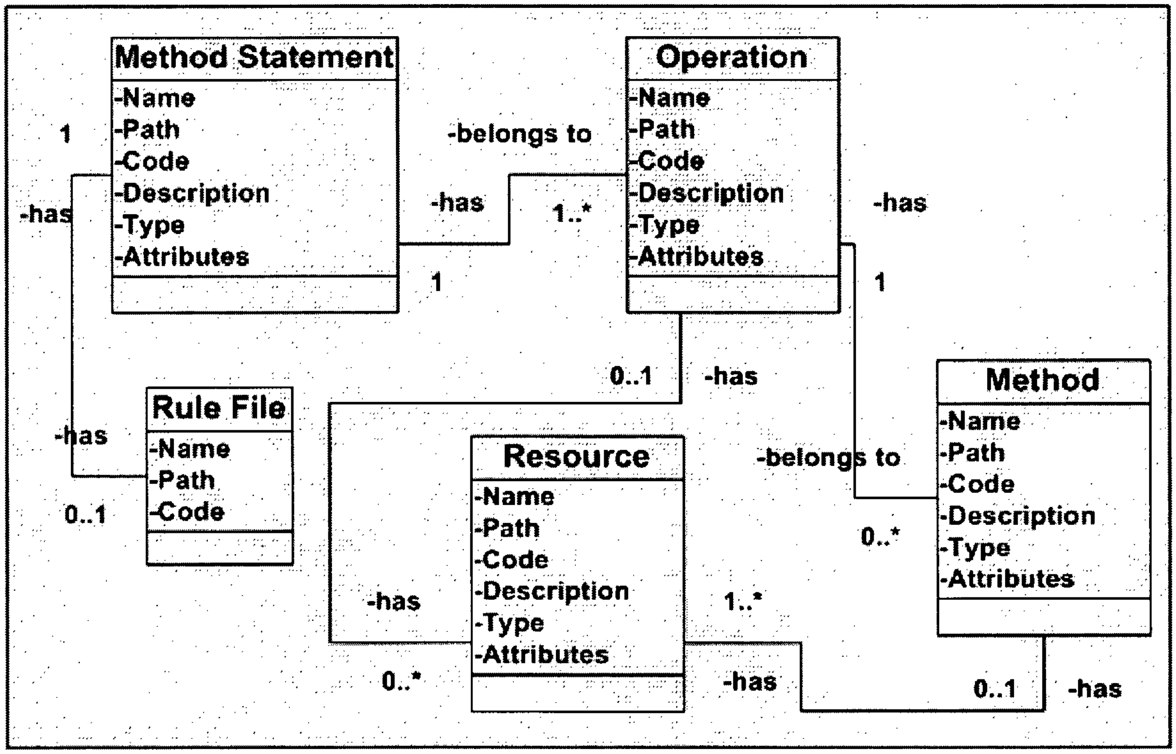


Figure 6.3. UML static structure diagram of Method Statement and its constituents.

On the other hand, the methods that are present in the standard M&RBS library have associated "Rule repositories" containing production rules. As shown in the object model diagram (figure 6.4), one method can have only one rule repository. For example, the method "Wooden Gang Formwork" is applicable to project component shear wall. The method's feasibility rules are component specific and they contain application context (e.g., whether the rule is applicable to superstructure walls or substructure walls). The rule repository of method Wooden Gang Formwork contains all these rules with their individual contexts and their associations (explained as follows).

- Rule Tagging:

Rule Tagging is used to identify each production rule. The CLIPS inference engine recognizes each production rule by its "Rule Name". For identification and selection of every production rule we have used a unique[1] rule name and its association. The association is a data element associated with each production rule indicating the PCBS components to which it is applicable. Every association has two PCBS component "codes". As shown in the object model diagram (figure 6.4), a rule has only one association, but one association of PCBS components can have a number of associated rules.

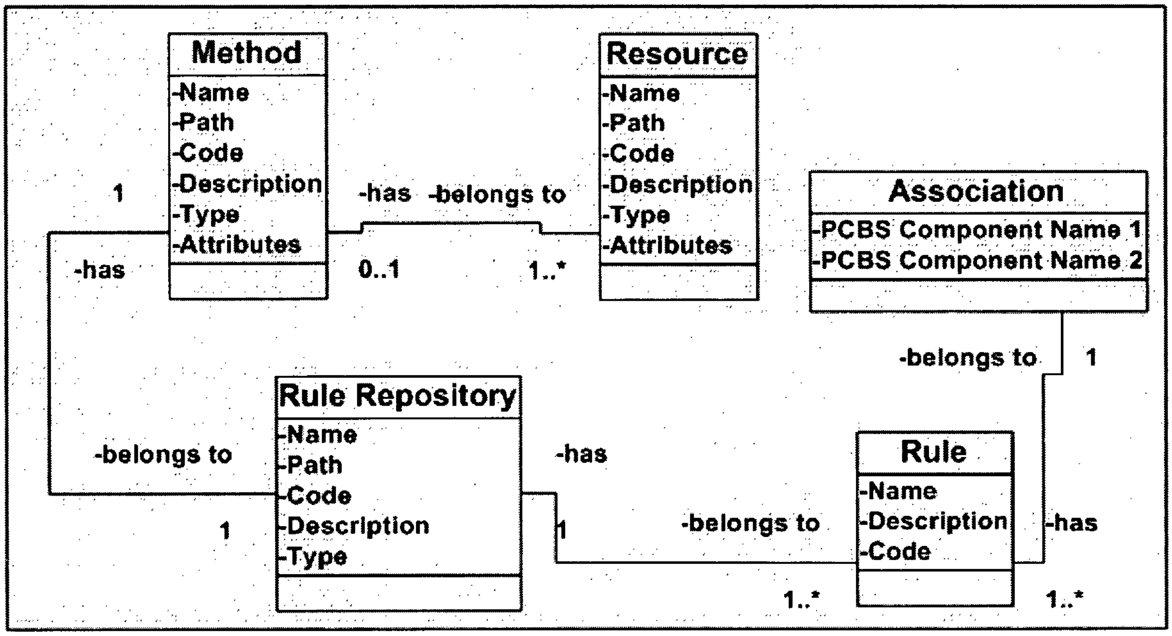


Figure 6.4. UML static structure diagram of Method, Rule repository, Rule, and Associations.

During the tree formation of a *method statement* the user selects methods from the standard library and copies them over to the *method statement*. Depending upon the application context of the method, the user can give the association with two physical components.

For example, when Wooden Gang Formwork is selected for the superstructure wall component, the association becomes "<*Superstructure*> + <*Wall*>".

Association	PCBS component code 1	PCBS component code 2
	Superstructure	*Wall*

These associations are used to retrieve only the relevant production rules from the rule repository to include in the *method statement* rule file as shown in figure 6.5. This operation is performed by an external subroutine code i.e., "Softcode3" written in C++.

[1] The unique rule name is generated by the system automatically while defining rules and their associations.

[3] Agenda, sometimes considered as the part of Inference engine itself (Giarratano and Riley, 1998).

Thus, Rule Tagging and Associations can be used to customize a *method statement* rule file according to the scope and level of the *method statement*.

6.3.5 Exposing M&RBS Facts, Instances, and Relationships to CLIPS

The *method statement* is copied onto the project side with all of its constituent operations, methods, and resources. As shown in figure 6.1, the *method statement* rule file is also copied along with the *method statement*. Similar to the Project PCBS, the *method statement* is a hierarchical listing of components, which need to be exposed in standard M&RBS template format for rule-based reasoning. In this step of the reasoning schema, the *method statement* gets exposed as M&RBS facts and instances with the help of "Softcode 4", and the hierarchical relationships between M&RBS components are exposed by "Softcode 5" (figure 6.6).

6.3.6 Reasoning

After loading the PCBS and M&RBS hierarchies in terms of facts and instances, the *Method statement* rule file and relationship rule file are also loaded into the CLIPS environment. The rule-based reasoning is then performed based on the facts, instances and rules. As shown in figure 6.7, the rule-based reasoning system has four main components the working memory, the agenda[3], the knowledge base, and the inference engine.

Working memory

The working memory is defined as a global database of facts used by rules (Giarratano and Riley, 1998). These facts include the PCBS and M&RBS facts, instances, and relationship facts. Since they contain instances of the classes, they are more appropriately called "pattern entities" (Giarratano and Riley, 1998). These entities are globally available within the CLIPS environment for pattern matching of production rules performed by the CLIPS inference engine. These pattern entities are created, modified, duplicated, retrieved, and removed from the working memory depending upon the execution of the production rules.

Knowledge Base

The knowledge base of the rule based reasoning system contains the domain knowledge available in the form of production rules. Two types of rule files i.e., *Method statement* rule file and Component Relationship rule file, are included in the CLIPS knowledge base by loading their ".clp" file form.

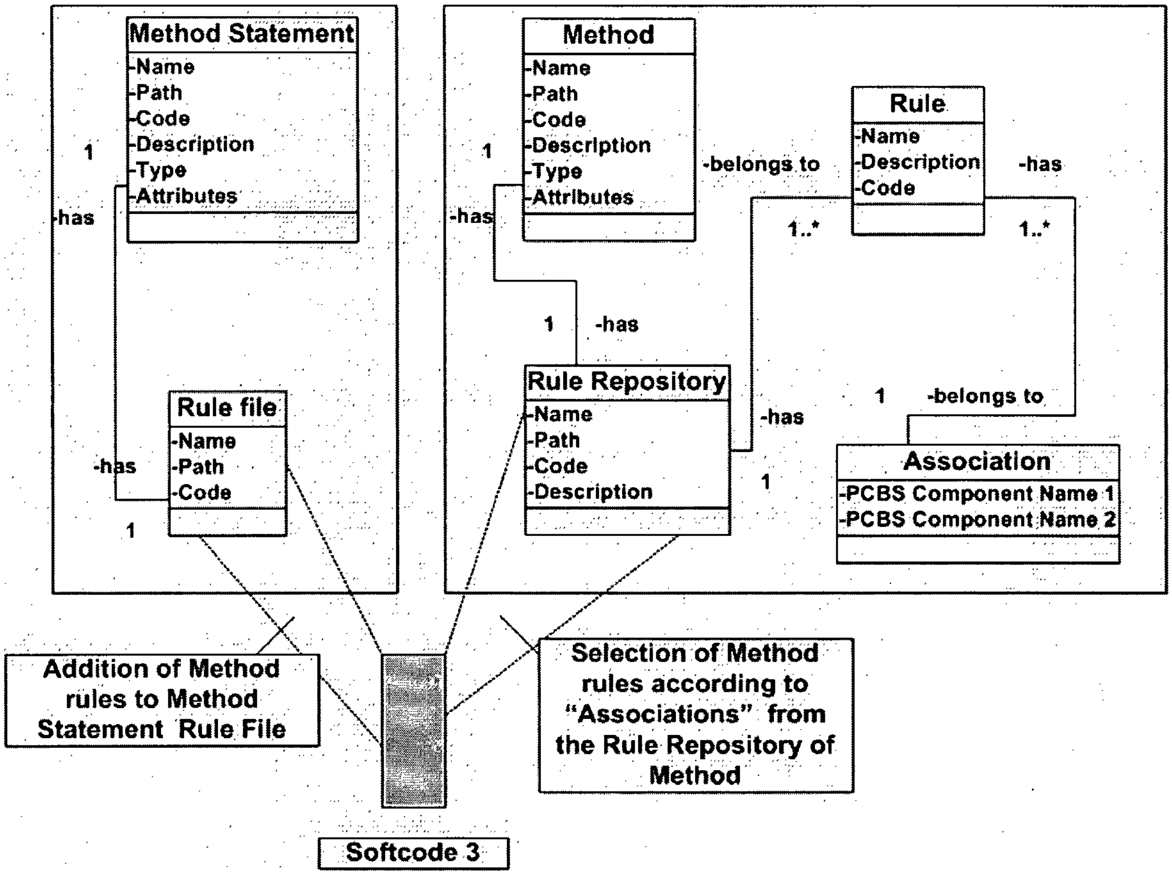


Figure 6.5. Schematic diagram of Method Statement Rule File formation.

The inference performed by the CLIPS inference engine is a forward chaining based pattern-matching mechanism, which uses the Rete algorithm (Giarratano and Riley, 1998). The algorithm matches available pattern entities in the working memory against the patterns in the rules to determine which rule conditions are satisfied. The rules for which all conditions are satisfied are said to be "activated" or instantiated. Whenever multiple rules get activated and become available to fire they are put on the "agenda". Pattern matching continues until all activated rules in the agenda are fired and no new facts are created.

Agenda

"The agenda is the list of all rules which have their conditions satisfied (and have not yet been executed)" (Giarratano and Riley, 1998). Whenever multiple rules are activated, the inference engine stores them in the temporary memory and orders them

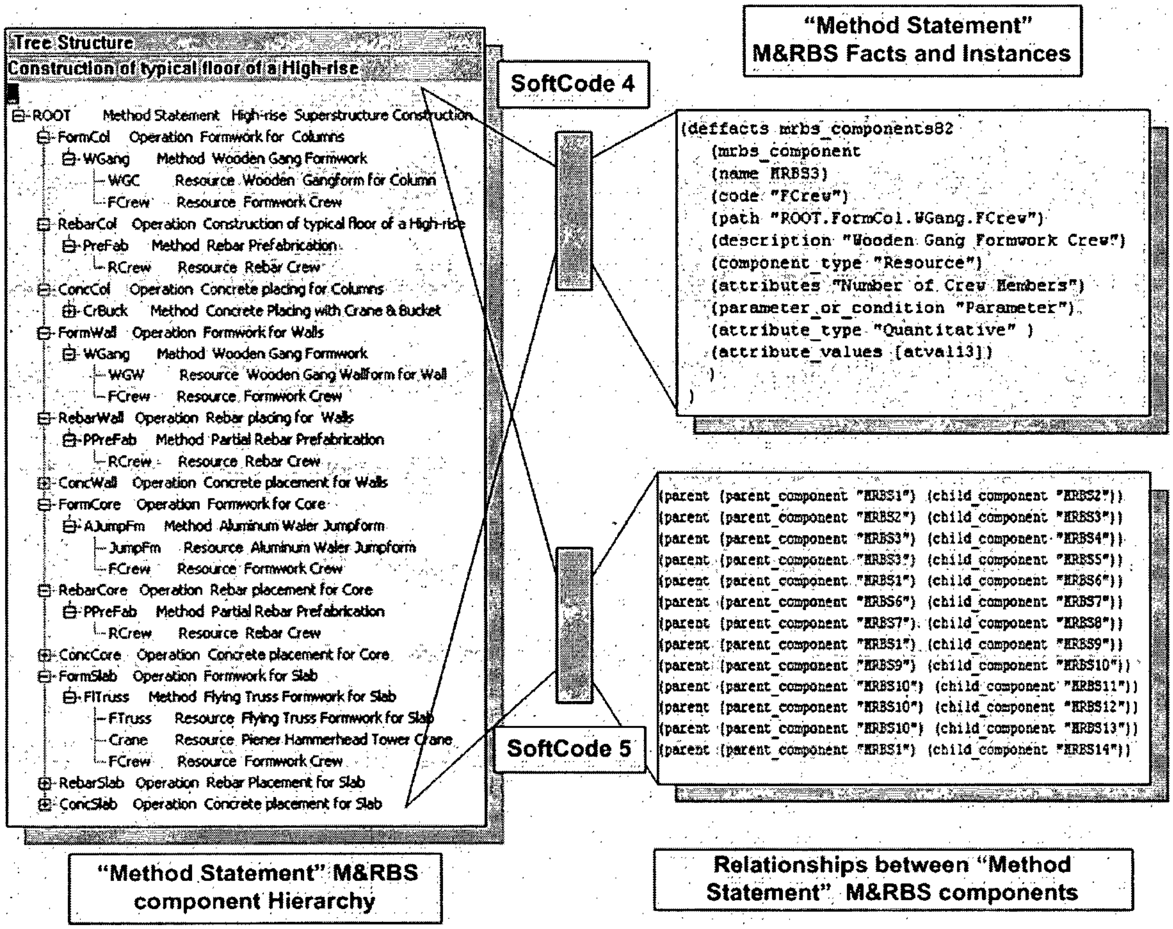


Figure 6.6. Exporting M&RBS facts, instances, and relationships to CLIPS.

according to priority for execution. The priority of the rule for execution, or in other words the placement of the rule on the agenda is determined by the salience[5] of the rule.

The placement of a rule on the agenda is based on the factors as follows:

1. Newly activated rules are placed above all rules of lower salience and below all rules of higher salience.

2. Among rules of equal salience, the current conflict resolution strategy is used to determine the placement among the other rules of equal salience.

3. If a rule is activated (along with several other rules) by the same assertion or retraction of a fact, and steps 1 & 2 are unable to

[5] Salience is the rule property that allows the user to assign a priority to a rule. Salience value should be an expression that evaluates to an integer in the range of –10000 to + 10000 (CLIPS, 2002).

specify an ordering, then the rule is arbitrarily (not randomly) ordered in relation to other rules with which it was activated.

- (Giarratano and Riley, 1998).

Seven conflict resolution strategies are available in CLIPS: depth, breadth, simplify, complexity, lex, mea, and random (CLIPS, 2002). We have used the default strategy i.e., the depth strategy in which the newly activated rules are placed above all rules of the same salience.

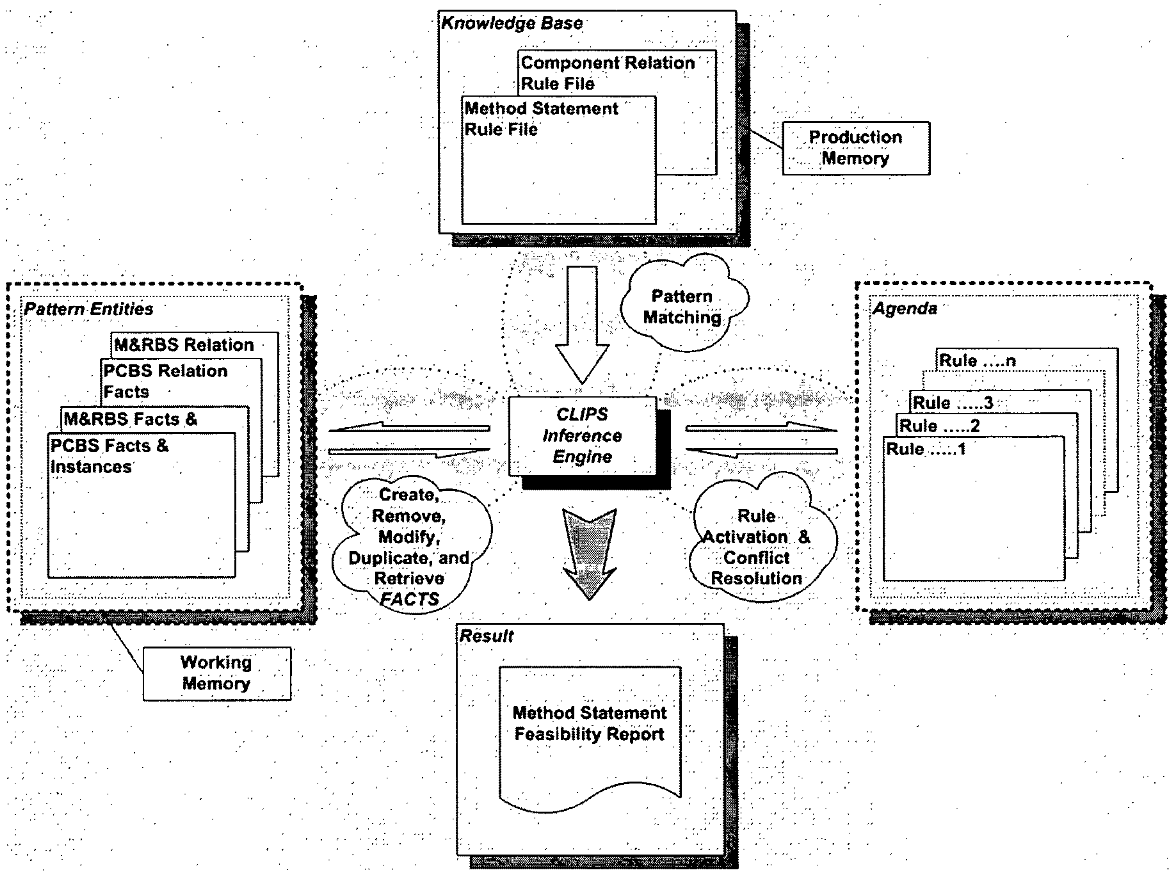


Figure 6.7. Reasoning with CLIPS inference engine.

6.3.7 Result Analysis

The inference engine executes the rules according to the agenda. Since every rule represents a different feasibility knowledge aspect, they produce specific feasibility results. For example, the rule for site-space availability will yield a result as to whether the site space available on the project is sufficient for the method or not. Depending upon the number of a certain type of fact (e.g. a PCBS component with code "Column"), the rule will get evaluated for each fact instances and provide component specific feasibility analysis. For example, if the super structure of the project facility has a number of slab-bays at each floor location, the feasibility rule will be evaluated for every component (all slab-bays) for each of its locations. The corresponding results are output as follows:

Result: *"The Method 'Wooden Gang Formwork' is not suitable for 'Shear Wall' because of lower production rate; the estimated resource usage is 15.17 crewhrs at location 2".*

Result: *"The Method 'Aluminum Waler Jumpform' is suitable for 'High Rise Tower Core' for the time allowance given considering rate of production at locations ("GFL" "3" "4" "5" "6" "7" "8" "9" "10")".*

Considerable effort is involved in encoding meaningful diagnostics / results for the user. For this thesis, we are interested in reporting the failed conditions and passed conditions for decision-making purposes. However, the user can write rules that produce various forms of output i.e., construction method related risk information, quality management issues, work method issues, etc. The output can be printed to separate text files.

Based on the results of the feasibility analysis, the user can choose to keep the *method statement*, modify it, or discard it.

Chapter 7. Implementation

7.1 Overview

Described in this chapter is proof of concept of the *method statement* reasoning schema discussed in previous chapters. A high-rise tower in downtown Vancouver is used as the case example. PCBS and M&RBS structures were created for the superstructure system of this project. Method statement reasoning was performed using production rules based on the feasibility factor knowledge listed in Appendix-A.

7.2 Project PCBS Description

The residential high-rise project, described in section 3.2.2, is used to demonstrate proof of concept (the actual construction strategy used was observed first hand). The high-rise project is divided into two subprojects i.e., a high-rise residential tower and low-rise townhouses. Only the high-rise tower subproject is used for *method statement* feasibility reasoning. We revisit the PCBS description of the subproject in this chapter in order to illustrate the description of the individual PCBS components such as columns, walls, core, and slab using a standard set of attributes.

7.2.1 Columns

The high-rise tower columns are described as the child node of the vertical elements *subsystem*. The *element* "Columns" is described as the collection of *subelements*, i.e., column types as shown in the figure 7.1. Each column type is described with attributes such as *Length*, *Width*, *Height*, *Number of Elements*, etc. Every attribute is assigned values corresponding to the locations on which they are present e.g., floor GFL, floor 2, etc. The attributes such as *Formwork Quantity*, *Rebar Quantity*, and *Concrete Quantity* are used to describe PCBS components at *element* level, as shown in figure 7.2. The quantitative values of these attributes are the summation of quantities for all the components under the *element* level. For example for *element* "Columns", at a particular location, the value of attribute *Formwork Quantity* is the cumulative formwork quantity of all column types and corresponding number of columns listed under them for that location.

In high-rise construction, contractors think in terms of a construction cycle for a typical floor. A shorter and more economical construction cycle is always desired. For a given construction cycle, the user can input the allowable timeframe for a specific operation at particular floor location. For example, the user can input *Timeframe for Formwork* for column formwork as 8 hours at locations "2", "3", etc. The timeframe and available formwork quantity at that location will be used for calculating the required rate of production which can be evaluated against the available rate of production from the column forming method (selected as a part of the *method statement*) considering crew

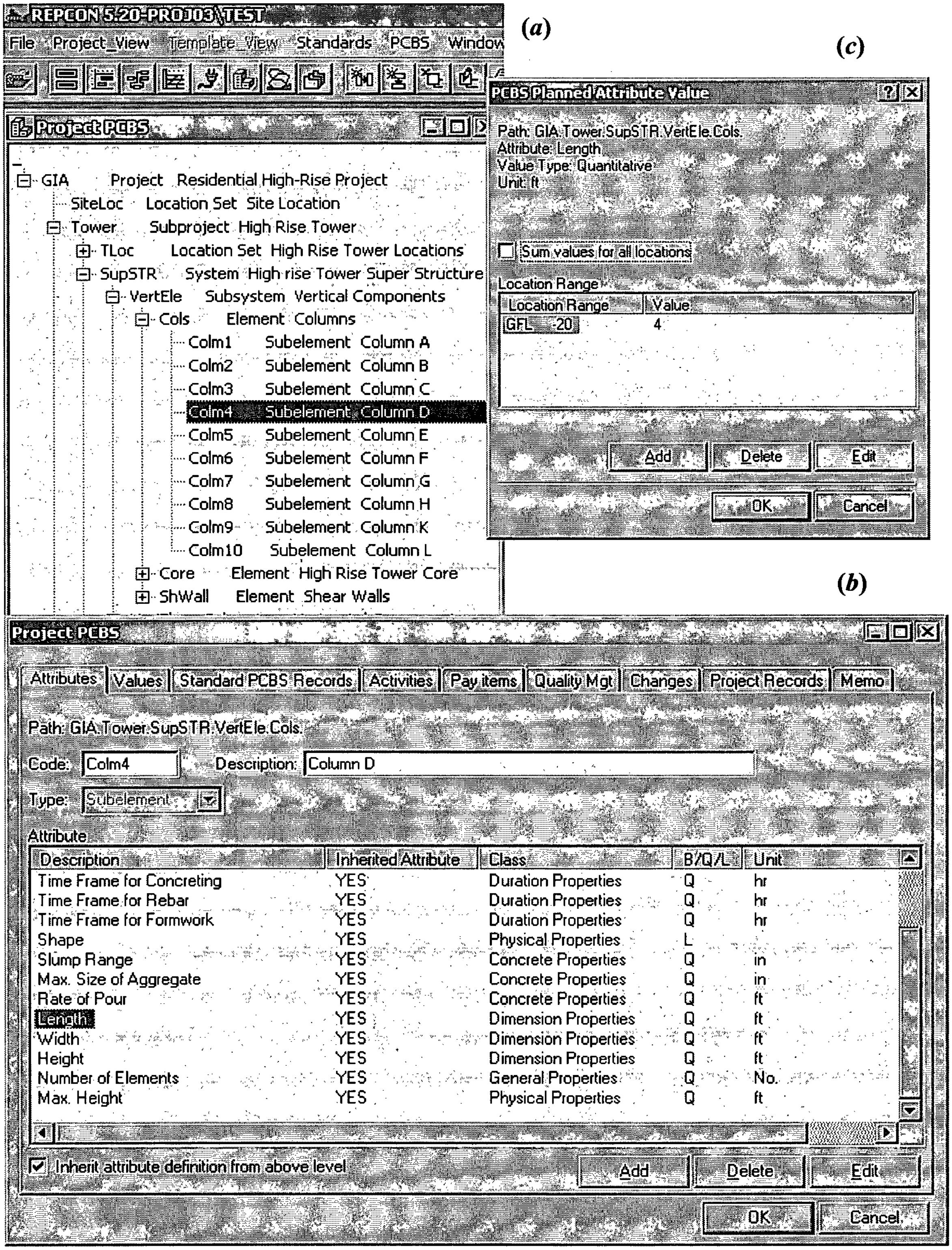


Figure 7.1. ***(a)*** Subelement "Column D" described as a column type; ***(b)*** Subelement "Column D" with attribute "Length"; ***(c)*** The value of attribute "Length" at the location range.

sizes (also selected as a part of the *method statement*). Similarly, Timeframe for Rebar and Timeframe for Concreting can also be specified.

For reasoning about concrete placement methods, we described the "Column" *element* with attributes showing maximum horizontal distance from the parking space used for concrete delivery and the maximum vertical distance for concrete placement from the ground level. The concrete properties such as *Slump Range* and *Max. Size of Aggregate* are also important attributes for feasibility reasoning for concrete placement methods. Every vertical component has assigned the attribute *Rate of Pour*[1].

7.2.2 Walls

Similar to columns, walls are also modeled as an *element*, which is a child node of vertical element *subsystem*. The *element* "Shear Walls" is a collection of shear-walls modeled as *subelements* at a lower level. The point to note is that the shear walls occurring at location GFL (i.e. ground floor) are modeled as a *subelement* "Shear Wall A" representing non-typical walls at that location. Similarly, *subelement* "Shear Wall B" represents non-typical walls occurring at location 2nd floor. Remaining shear walls present at multiple floor locations are categorized according to their lengths and associated subcomponents. The number of possible reuses for formwork is an important feasibility condition of the formwork method. The foregoing categorization of shear walls helps in the identification of the possible reuses for each type of shear wall.

Every *subelement* of "Shear wall" type is described with basic attributes such as *Length*, *Width*, and *Height*. The wall subcomponents such as corners, offsets, pilasters, and openings are further modeled as the *subsubelements* under the corresponding *subelement* wall. The attributes such as formwork quantity, rebar quantity, and concrete quantity are assigned to the *element* "Shear Wall", as shown in figure 7.3. Similar to column *element*, the shear wall *element* can also be described with timeframe attributes and concrete properties. The shear walls designed with shear zones[2] are modeled as the *element* "Shear Wall" with the *content* "Shear Zone".

[1] The *Rate of Pour* is an important property associated with formwork that is used for calculation of the maximum allowable pour pressure. Generally, the formwork contractor asks the designer to design formwork with a required rate of pour. The column gang formworks are generally designed for full head, i.e., 8 ft /hr rate of pour.

[2] In earthquake prone zones such as Vancouver, the vertical shear reinforcement in core and shear walls needs to be staggered at alternate floors. These concentrated regions of rebar, generally two storeys high, are called "zones" (Fradley) (Bitchel).

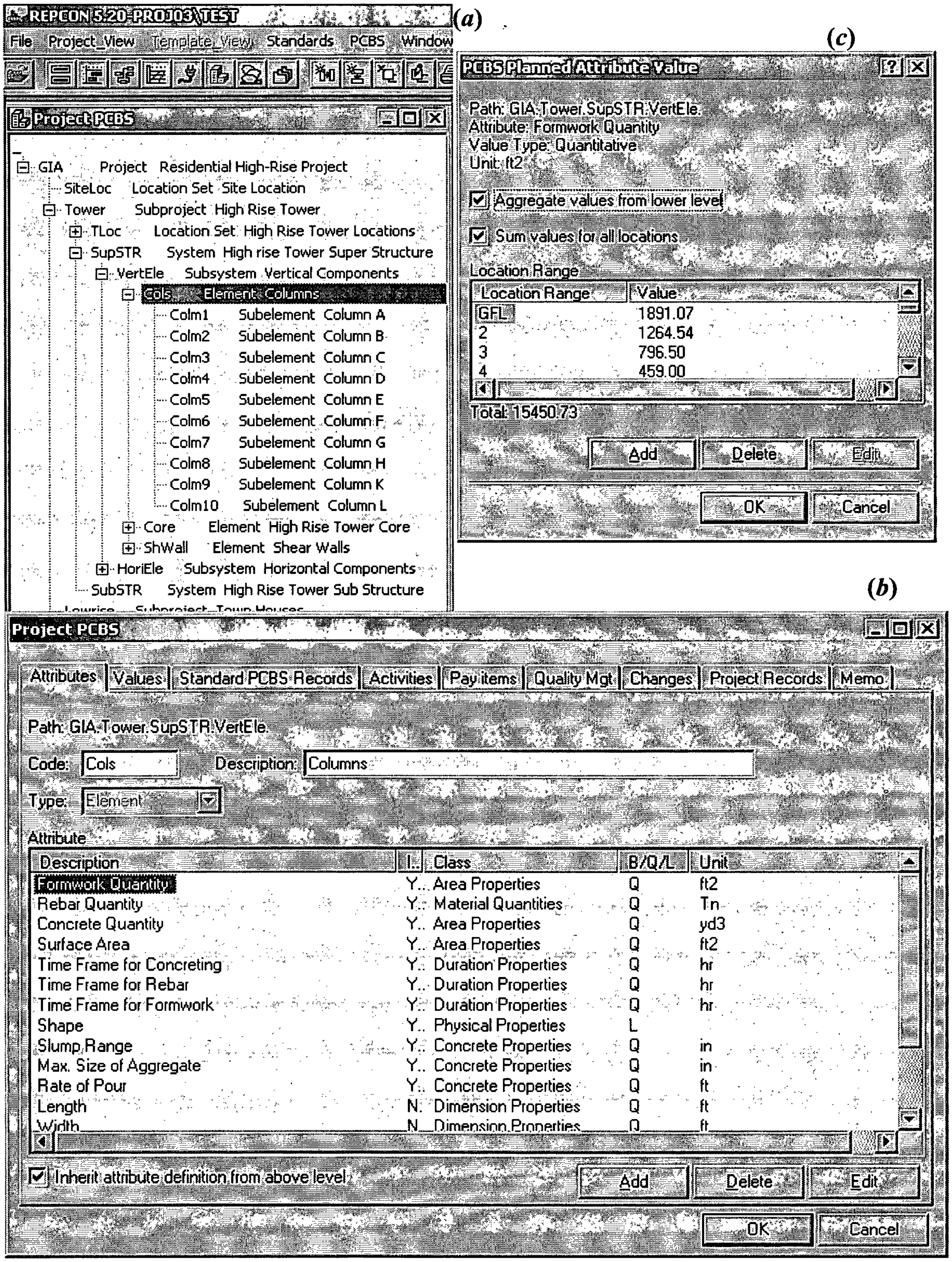


Figure 7.2. (***a***) Element "Column" described as the collection of column types; (***b***) Element "Column" with attribute "Formwork Quantity"; (***c***) The value of attribute "Formwork Quantity" at various locations.

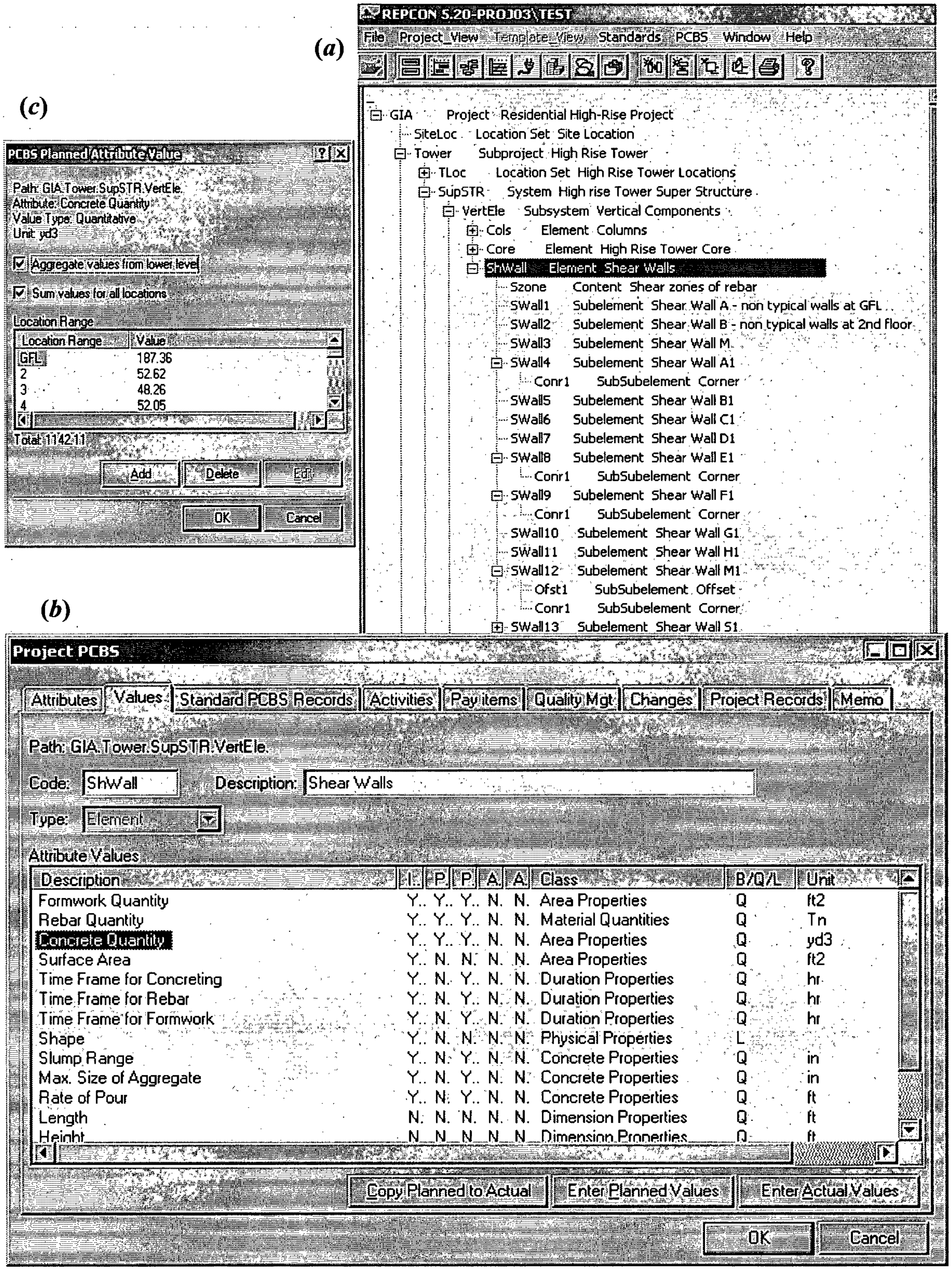


Figure 7.3. (***a***) Element "Shear Wall" described as the collection of shear wall types; (***b***) Element "Shear Wall" with attribute "Concrete Quantity"; (***c***) The value of attribute "Concrete Quantity" at various locations.

7.2.3 Core

The core of the high-rise tower is described under the "Vertical Elements" *subsystem* and listed as child node of the subsystem i.e. *element*. For the purpose of more detailed representation the core is further subdivided into its constituent walls, which are described as the *subelements*[3] (as shown in figure 7.4). The "Core Wall" types are categorized according to their physical parameters (length and height) and subcomponents (openings, corners, and offsets). Attributes such as *Length*, *Width*, and *Height* are assigned to every component of type "Core Wall". The openings of the core walls are modeled as the *subsubelement* under corresponding *subelement* core walls. The high-rise core walls are designed with shear zones and are modeled as the *element* "Core" with *content* "Shear Zone". The *element* "Core" is further described with attributes such as *Formwork Quantity*, *Rebar Quantity*, and *Concrete Quantity* along with timeframe attributes and concrete properties. It is to be noted that the core walls are modeled separately from the shear walls because it facilitates feasibility reasoning of formwork methods which are more commonly used for core forming such as slip forming and self-climbing formwork.

7.2.4 Slab

The *element* "Slab" is described as the child node of the *subsystem* "Horizontal Elements" with attributes such as *Formwork Quantity*, *Rebar Quantity*, and *Concrete Quantity*. The *element* slab is further subdivided into *subelements* called slab-bay according to the orientation of the vertical supports and possible orientation of the flytables as shown in the figure 7.5 and 7.6.

Each slab-bay is described with the help of the standard attributes *Length*, *Width*, *Thickness*, *Shape*, etc. For purposes of feasibility reasoning about various slab forming systems, the properties of the slab-bay are more appropriately expressed with the help of boolean attributes indicating whether or not the *SlabBay Supporting Sides are Parallel* and the *SlabBay Support is Uniform,* as shown in the figure 7.7. A slab-bay may contain beams, slabbands, and a spandrel beam, which can be modeled as the *subsubelements* for the *subelement* "Slab-bays". Similar to other *elements,* slabs are also described with formwork, rebar, and concrete quantities with corresponding timeframe attributes. Concrete properties are also listed.

[3] The structural element core can have various forms and layouts according to its constituents such as elevator shafts, lobby, staircase, toilets, and mechanical and electrical service rooms [Yeang 2000]. Therefore for the purpose of more detailed representation purpose we described core with its constituent walls as *subelements*.

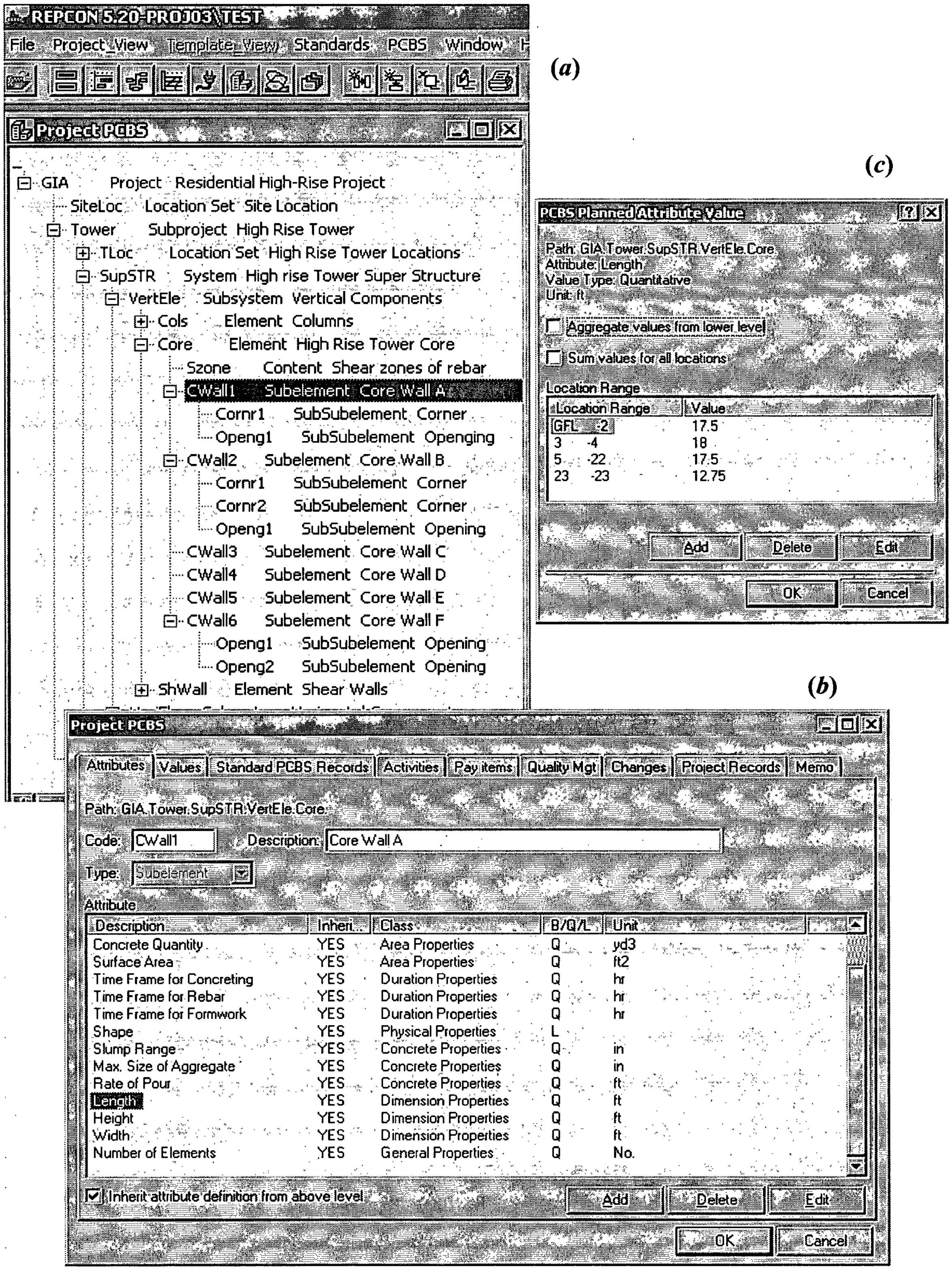


Figure 7.4. **(*a*)** Subelement "Core Wall A" described as a core wall type; **(*b*)** Subelement "Core Wall A" with attribute "Length"; **(*c*)** The value of attribute "Length" at various locations.

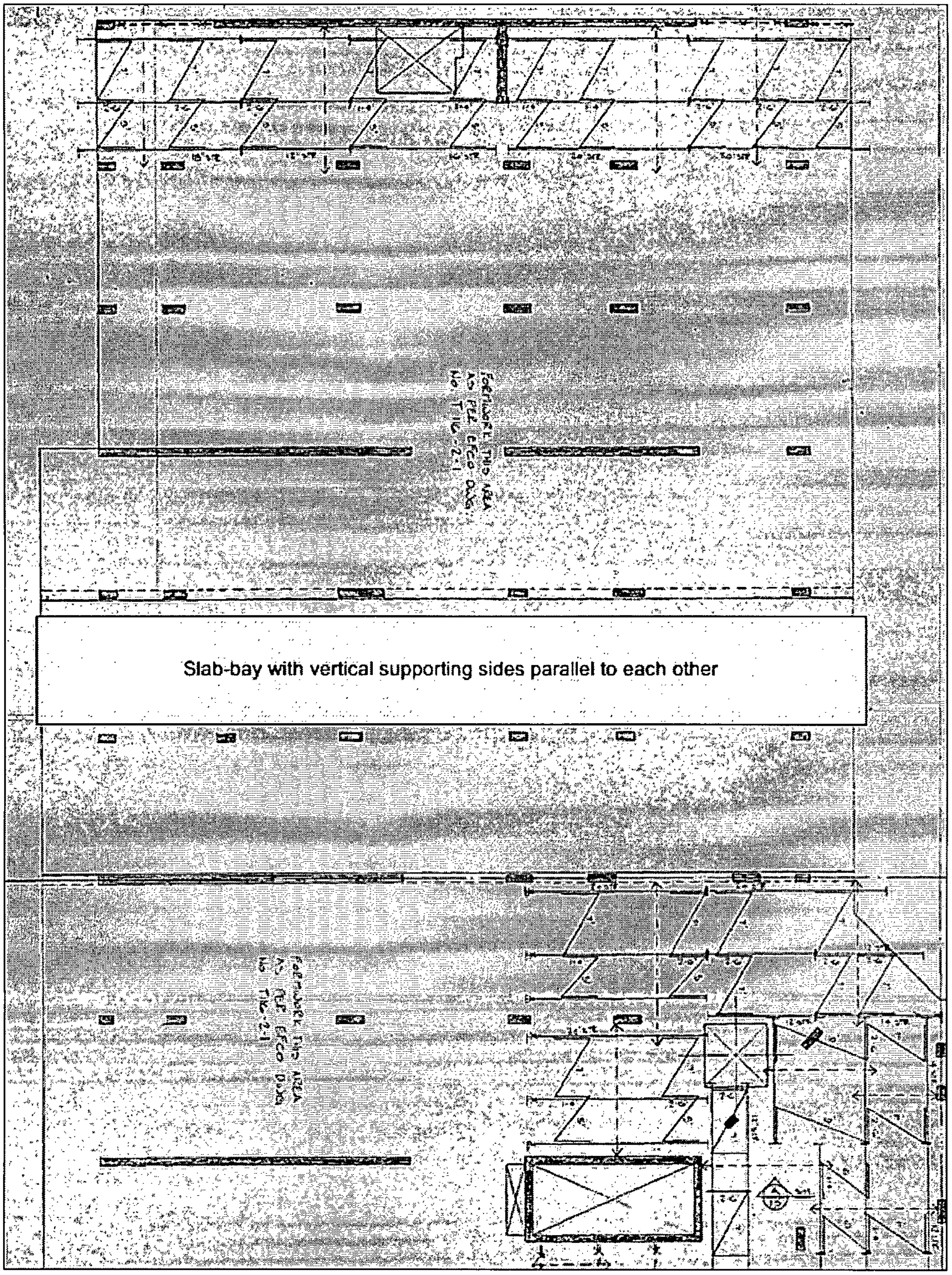


Figure 7.5. Plan showing slab-bays with vertical supporting sides parallel to each other.

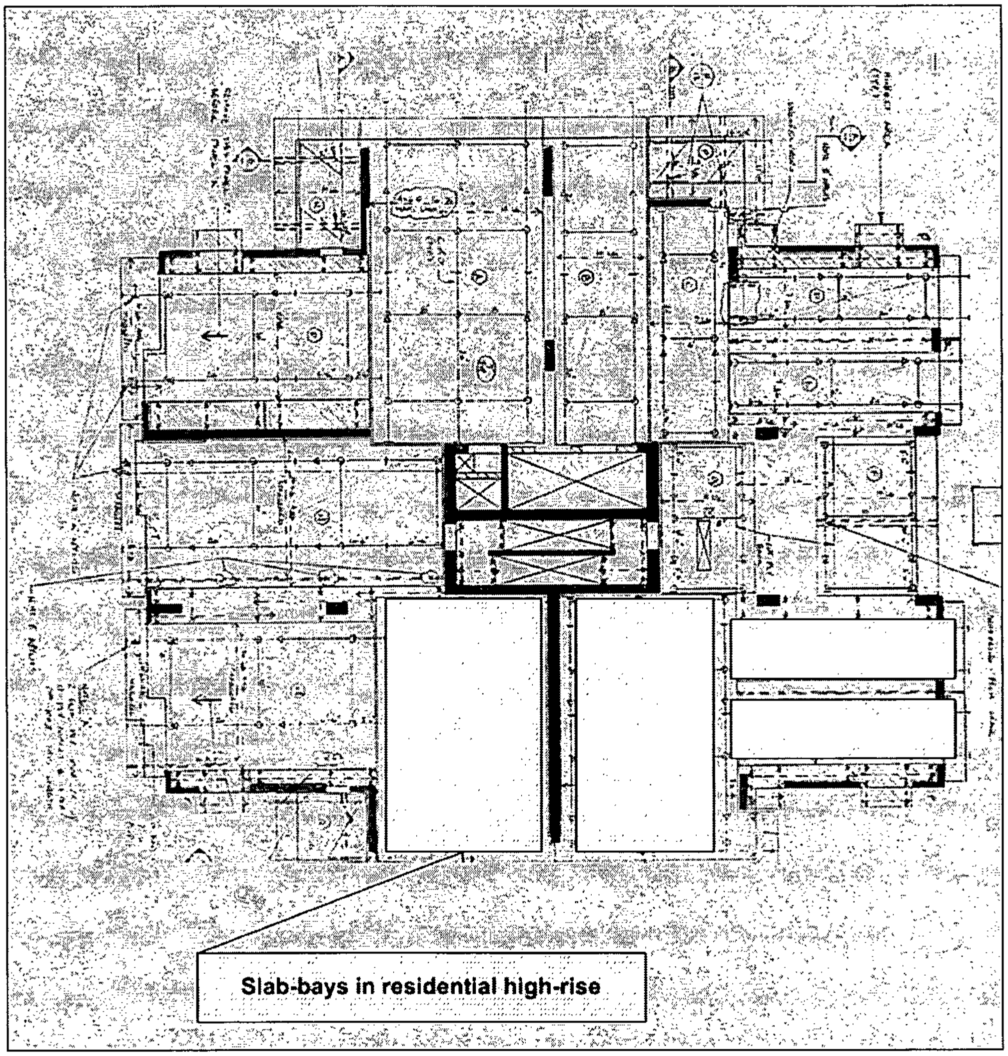


Figure 7.6. Plan showing slab-bays in case-study project.

7.2.5 Site Location and Tower Locations

The project site location is described as a *location* component for the high-rise tower. Available site storage area, rebar storage area, parking area available, and open space area are described by their length and width (see figure 7.8). These attributes are used in feasibility reasoning regarding the *method statement* defined for this type of project.

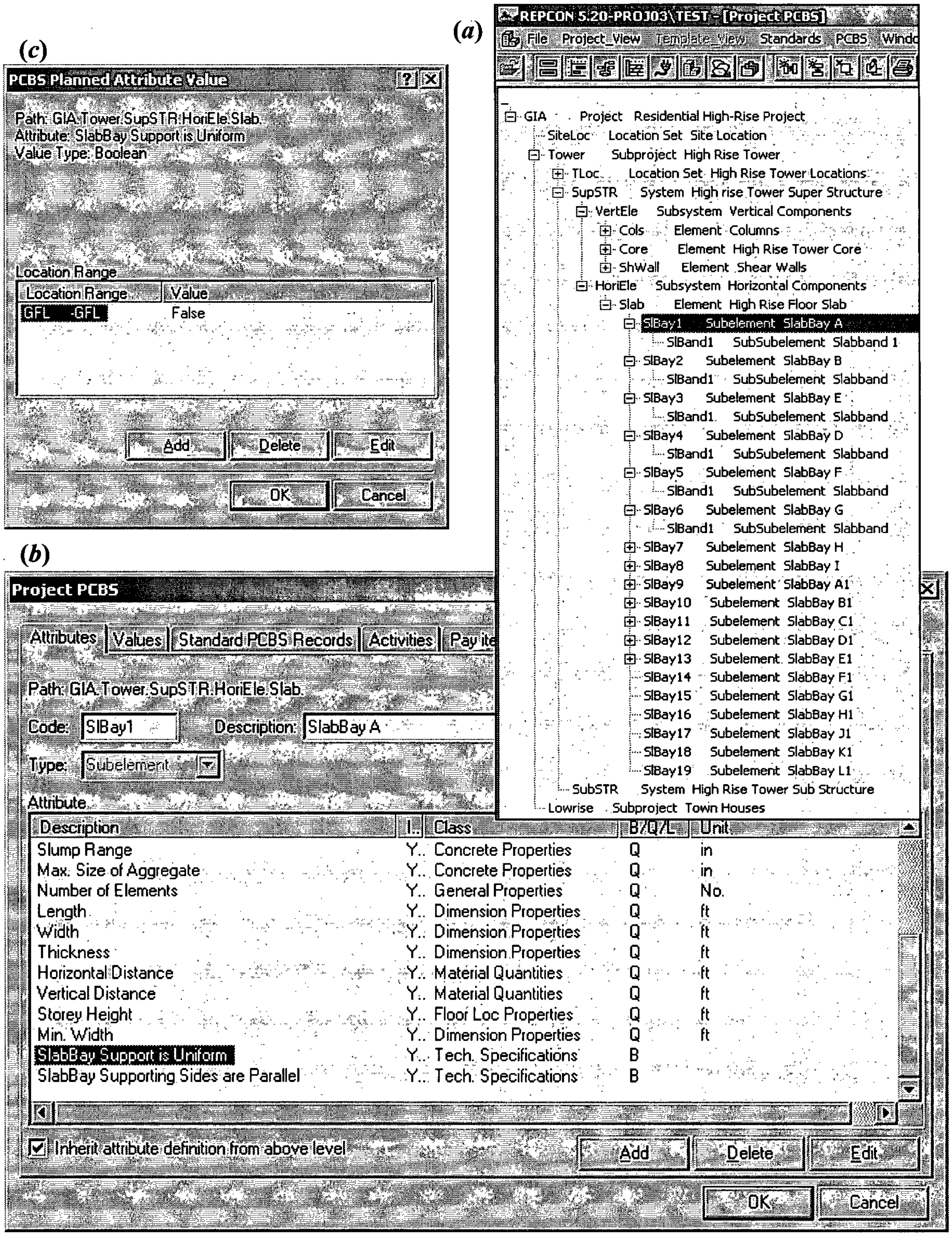


Figure 7.7. ***(a)*** Element "Slab" described as a collection of slab-bay subelements; ***(b)*** Subelement "SlabBay A" with attribute "SlabBay Support is Uniform"; ***(c)*** The value of attribute "SlabBay Support is Uniform" at location.

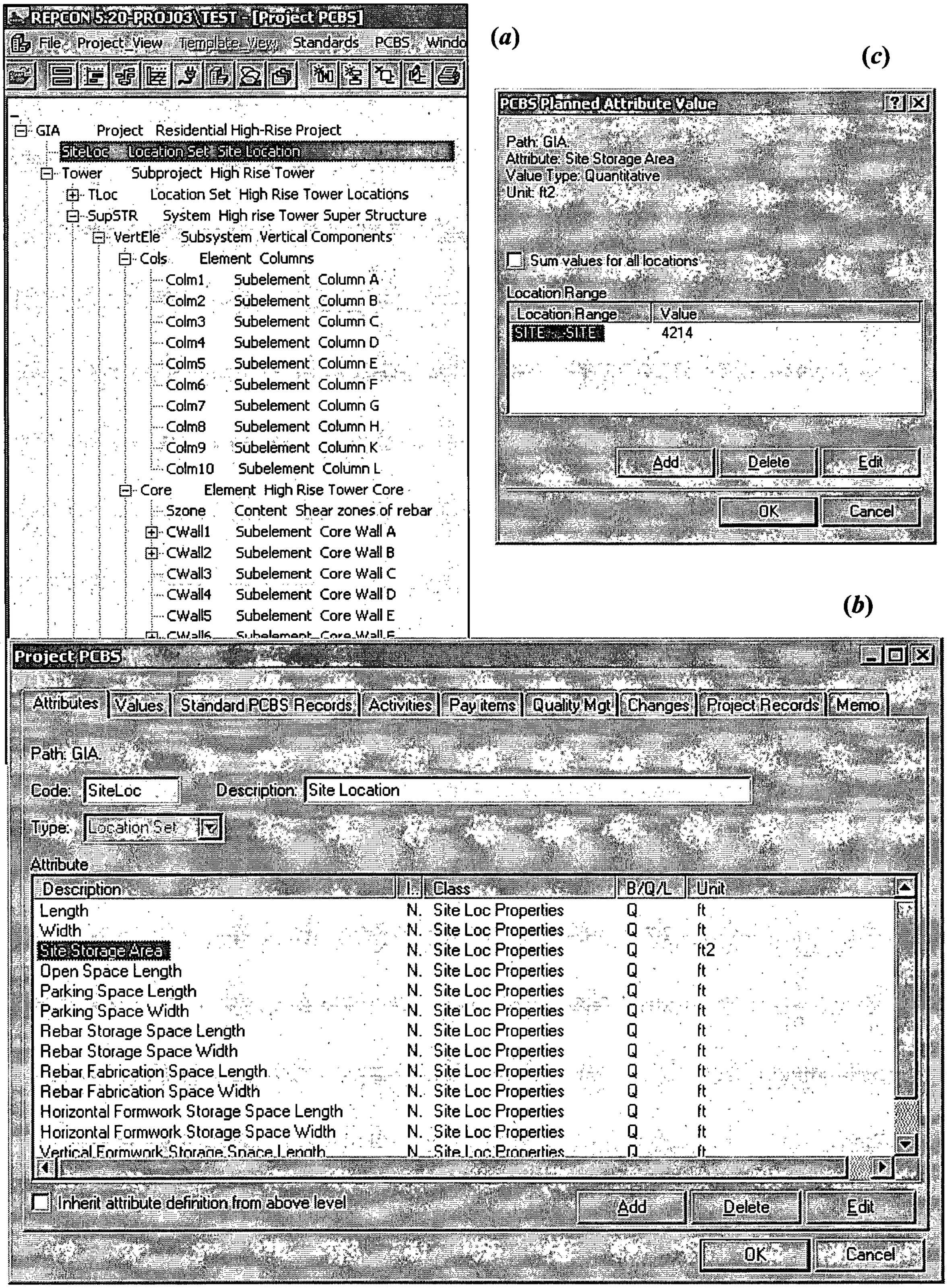


Figure 7.8. ***(a)*** Site location of the project described with component named "Site Location"; ***(b)*** Component "Site Location" with attribute "Site Storage Area"; ***(c)*** The value of attribute "Site Storage Area" at location.

7.3 Project M&RBS

The standard M&RBS *method statement* developed for high-rise superstructure construction is shown in figure 7.9.

Figure 7.9. Method Statement hierarchy with operations, methods, and resources.

Three basic *operations* were considered for each of the physical components (i.e., column, wall, core, and slab), these being formwork, rebar placement, and concrete placement. Each *operation* has its own methods and resources described under it along with their feasibility parameters and conditions as shown in figure 7.10. These parameters

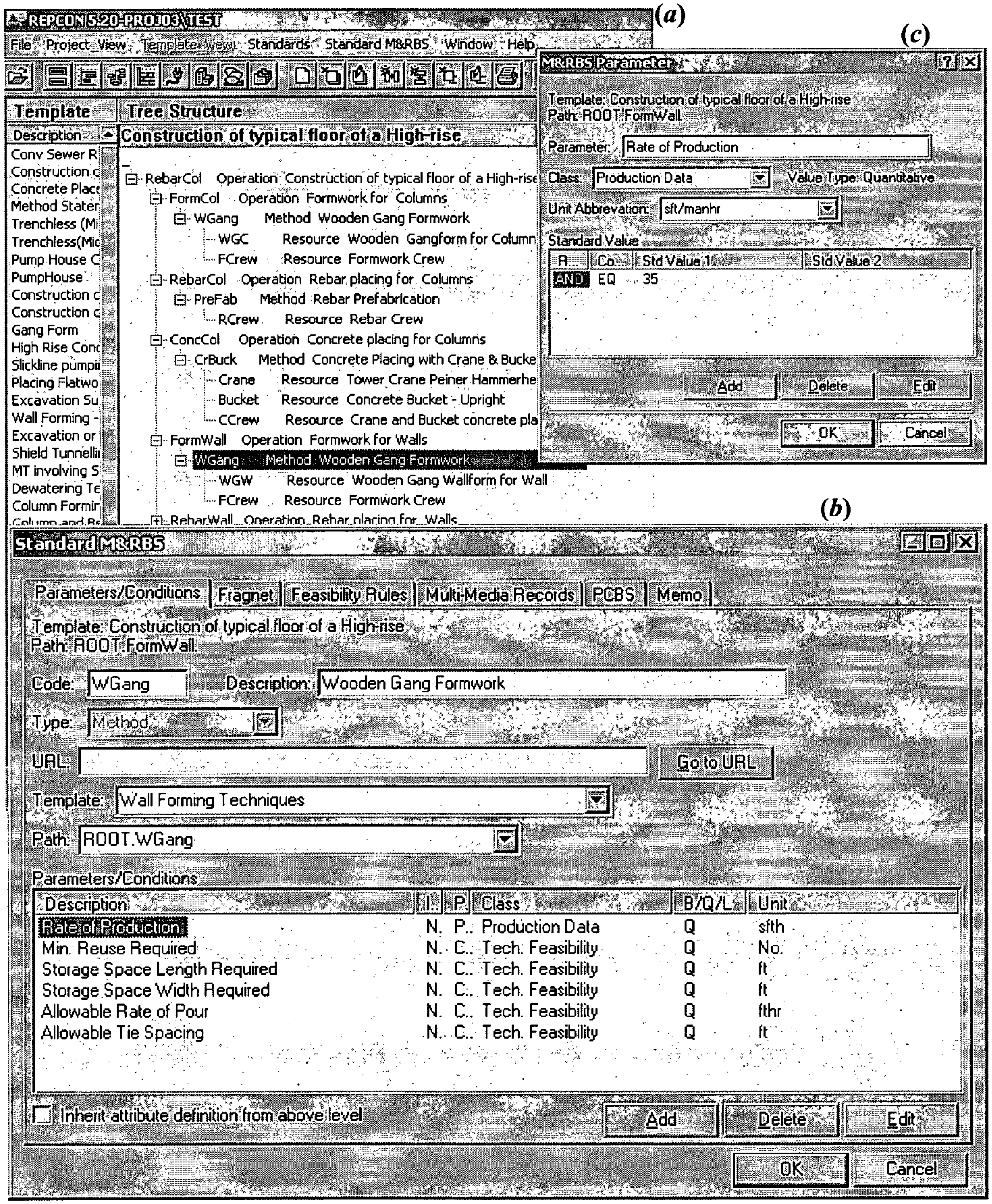


Figure 7.10. ***(a)*** Method Statement hierarchy with method "Wooden Gang Formwork" (highlighted); ***(b)*** Method "Wooden Gang Formwork" with parameter "Rate of Production"; ***(c)*** The value of parameter "Rate of Production".

and conditions are used in the formation of production rules, which are listed in the rule repositories associated with methods. The formation of the *method statement* rule file works in tandem with the formation of *method statement* as described in section 6.5. Initially, we manually formed the *method statement rule file*, which is listed in Appendix-B. Eventually, this process will be automated.

The *method statement* is then exposed in terms of "facts" in the CLIPS environment, and these facts are then interpreted by the relationship rules to the establish hierarchical relationships between the "facts" of the elements in CLIPS environment. The facts are listed in the Appendix-B.

7.4 Reasoning

The reasoning process starts with loading the PCBS template definition, M&RBS template definition, and relationship rules in the CLIPS environment, as shown in figure 7-11. The lists of PCBS and M&RBS relationship facts are also loaded and get defined in the CLIPS environment.

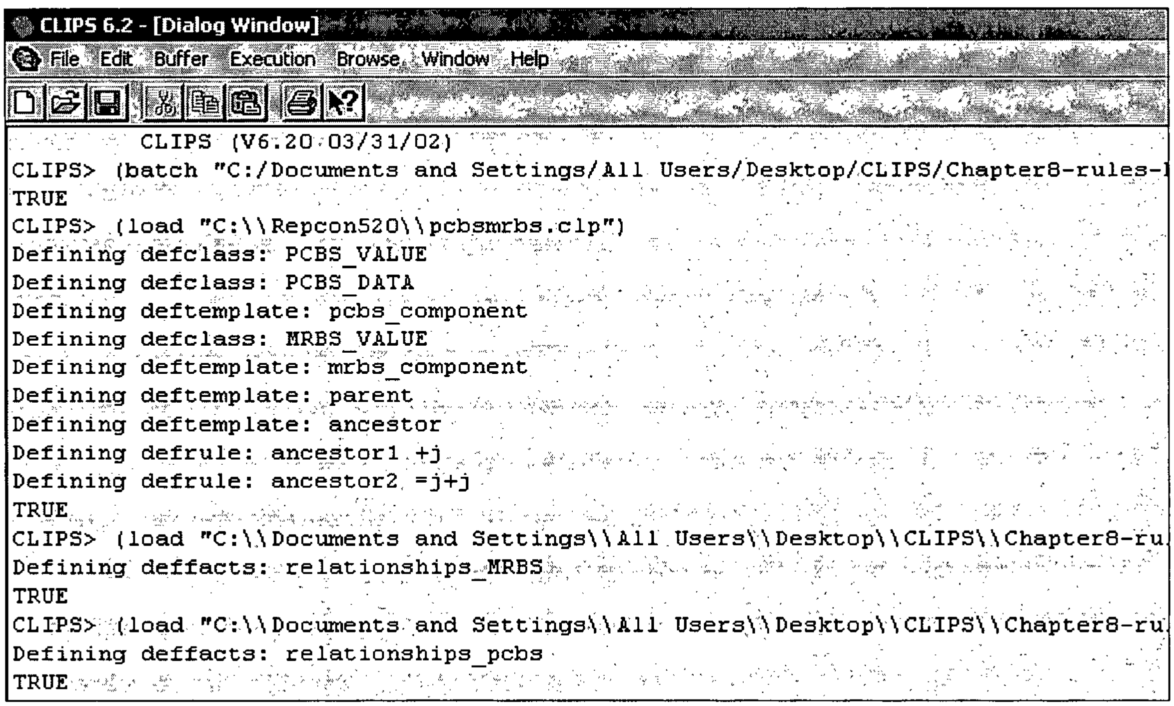


Figure 7.11. PCBS template, M&RBS template, and relationship rules and facts get defined in CLIPS environment.

The *method statement* rule file is also loaded, as shown in figure 7-12. The rules get defined in CLIPS environment. The lists of PCBS and M&RBS facts and instances are included in CLIPS environment by loading separate facts file and instances file, as shown in figure 7-13.

```
CLIPS 6.2 - [Dialog Window]
File Edit Buffer Execution Browse Window Help
Defining deffacts: relationships_pcbs
TRUE
CLIPS> (load "C:\\Documents and Settings\\All Users\\Desktop\\CLIPS\\Chapter8-rules-MS1\\Method_st
Defining defrule: file_open1 +j
Defining defrule: file_open2 +j
Defining defrule: file_open3 +j
Defining defrule: Shree1 +j
Defining defrule: Shree2 +j
Defining defrule: Shree3 +j
Defining defrule: site_space_check_wooden_gangform_column +j+j+j+j+j+j+j+j
Defining defrule: column_reuse_check_wooden_gangform_column +j+j+j+j+j+j+j+j+j
Defining defrule: column_rate_of_concrete_pour_wooden_gangform_column =j=j=j=j=j+j+j+j
Defining defrule: column_Tie_spacing_check_wooden_gangform_colmn =j=j=j=j=j=j+j+j
Defining defrule: column_Time_frame_for_formwork_wooden_gangform_column =j=j=j+j+j+j+j+j+j+j+j+j
Defining defrule: site_space_check_rebar_prefabrication_column_onsite +j+j+j+j+j+j+j+j+j+j+j+j
Defining defrule: site_space_check_storage_for_rebar_prefabrication_column +j+j+j+j+j+j+j+j
Defining defrule: time_frame_for_rebar_prefabrication_column =j=j+j+j+j+j+j+j+j+j+j
Defining defrule: site_space_check_crane_bucket_mathod_column =j=j+j+j+j+j+j
Defining defrule: rate_of_placement_crane_bucket_mathod_column =j+j+j+j+j+j+j+j
Defining defrule: max_size_of_aggregate_check_crane_bucket_mathod_column =j=j=j=j=j+j+j
Defining defrule: site_space_check_wooden_gangform_wall =j=j+j+j+j+j+j+j
Defining defrule: reuse_check_wooden_gangform_wall =j=j=j=j=j=j+j+j+j+j
Defining defrule: rate_of_concrete_pour_check_wooden_gangform_wall =j=j=j=j=j=j=j+j+j+j
Defining defrule: tie_spacing_check_wooden_gangform_wall =j=j=j=j=j=j=j+j+j
Defining defrule: time_frame_for_wooden_gangform_wall =j=j=j=j+j+j+j+j+j+j+j+j+j
Defining defrule: site_space_check_rebar_partprefab_wall =j=j+j+j+j+j+j+j+j+j+j
Defining defrule: site_space_check_rebar_onsite_fabrication_zones_wall =j+j+j+j+j+j+j+j+j+j
Defining defrule: time_frame_for_partial_rebar_prefabrication_of_wall =j=j+j+j+j+j+j+j+j+j+j+j
Defining defrule: site_space_check_crane_bucket_mathod_wall =j=j+j+j+j+j+j+j
Defining defrule: rate_of_placement_crane_bucket_mathod_wall =j=j+j+j+j+j+j+j
Defining defrule: max_size_of_aggregate_check_crane_bucket_mathod_wall =j=j=j=j=j+j+j
Defining defrule: site_space_check_aluminum_jump_form_corewall =j+j+j+j+j+j+j+j
Defining defrule: reuse_check_aluminum_jump_form_corewall =j=j+j+j+j+j+j+j+j+j+j
Defining defrule: rate_of_concrete_pour_check_aluminum_jump_form_corewall =j=j=j=j=j=j=j+j+j+j
Defining defrule: tie_spacing_check_aluminum_jump_form_corewall =j=j=j=j=j=j=j+j+j
Defining defrule: time_frame_for_formwork_aluminum_jump_form_corewall =j=j=j=j+j+j+j+j+j+j+j+j+j
Defining defrule: site_space_check_rebar_partprefab_corewall =j=j=j+j+j+j+j+j+j+j+j
Defining defrule: site_space_check_rebar_onsite_fabrication_zones_corewall =j=j=j+j+j+j+j+j+j
```

Figure 7.12. Method Statement rules get defined in CLIPS environment.

```
Defining defrule: site_space_check_crane_bucket_mathod_slab =j=j+j+j+j+j+j
Defining defrule: rate_of_placement_crane_bucket_mathod_slab =j=j=j=j=j+j+j+j+j
Defining defrule: max_size_of_aggregate_check_crane_bucket_mathod_slab =j=j=j=j=j+j+j
Defining defrule: file_close1 +j
Defining defrule: file_close2 +j
Defining defrule: file_close3 +j
TRUE
CLIPS> (reset)
(load-facts "C:\\Repcon520\\Proj03\\TESTda.fct")
TRUE
CLIPS> (load-instances "C:\\Repcon520\\Proj03\\TESTda.ist")
9582
CLIPS> (run)
```

Figure 7.13. PCBS and M&RBS facts (TESTda.fct) and instances (TESTda.ist) get loaded in CLIPS environment.

The reasoning process starts after the facts and rules are *Reset* and *Run* as shown in figure 7.13. During reasoning a number of rules are placed on the agenda and fired as shown in figure 7.14. The facts generated and the instances used during the "run" are also shown in the figure. For our example project for ease of checking, we added some additional rules to print the output for the *method statement* feasibility reasoning report to separate report files for formwork methods, rebar placement methods, and concrete placement methods. The output is included in Appendix-H.

7.4.1 Report Discussion

The report generated from the feasibility reasoning about the *method statement* High-*rise superstructure construction* indicates various failed and passed conditions related to feasibility of construction methods involved. For example, the report provides results describing the reasoning about reuses of the flying truss formwork system as follows

i.e., *"The Method 'Flying Truss Formwork' for PCBS component 'SlabBay E1' is feasible due to sufficient reuses 18".*

The report also indicates that by using the assigned formwork crew and the rate of production of the method (Flying Truss Formwork) the estimated duration of the slab formwork activity for the given formwork quantity at a particular location is more than the allowable time frame at that location.

i.e., *"The Method 'Flying Truss Formwork' is not suitable for 'High Rise Floor Slab' because the production rate does not meet the time constraint imposed; the estimated resource usage is 11.96 crewhrs at location 5".*

The user can either increase the crew size or can change the slab formwork method to a method with a higher production rate, such as tunnel formwork or column mounted flytable formwork. However, these methods also have their own feasibility conditions, which need to be satisfied before including them in the finalized *method statement.*

The production rules in the *method statement* rule file also indicate the availability of sufficient site storage space or assembly space for formwork, parking space for concrete placement equipment, and rebar storage and fabrication space.

i.e., *"The Method 'Rebar Prefabrication' does not have sufficient 'Onsite Fabrication Space Length' for 'Columns'."*

Similarly, the feasibility report indicates that the method of rebar placement is not suitable because it does not meet the imposed time constraint for the rebar placement operation at the location for the given rebar quantity, rebar crew, and rate of production of the rebar placement method.

i.e., *"The Method Partial Rebar Prefabrication Method is not suitable for High Rise Tower Core because of lower rate of production, the estimated resource usage is 8.82 crewhrs at location GFL".*

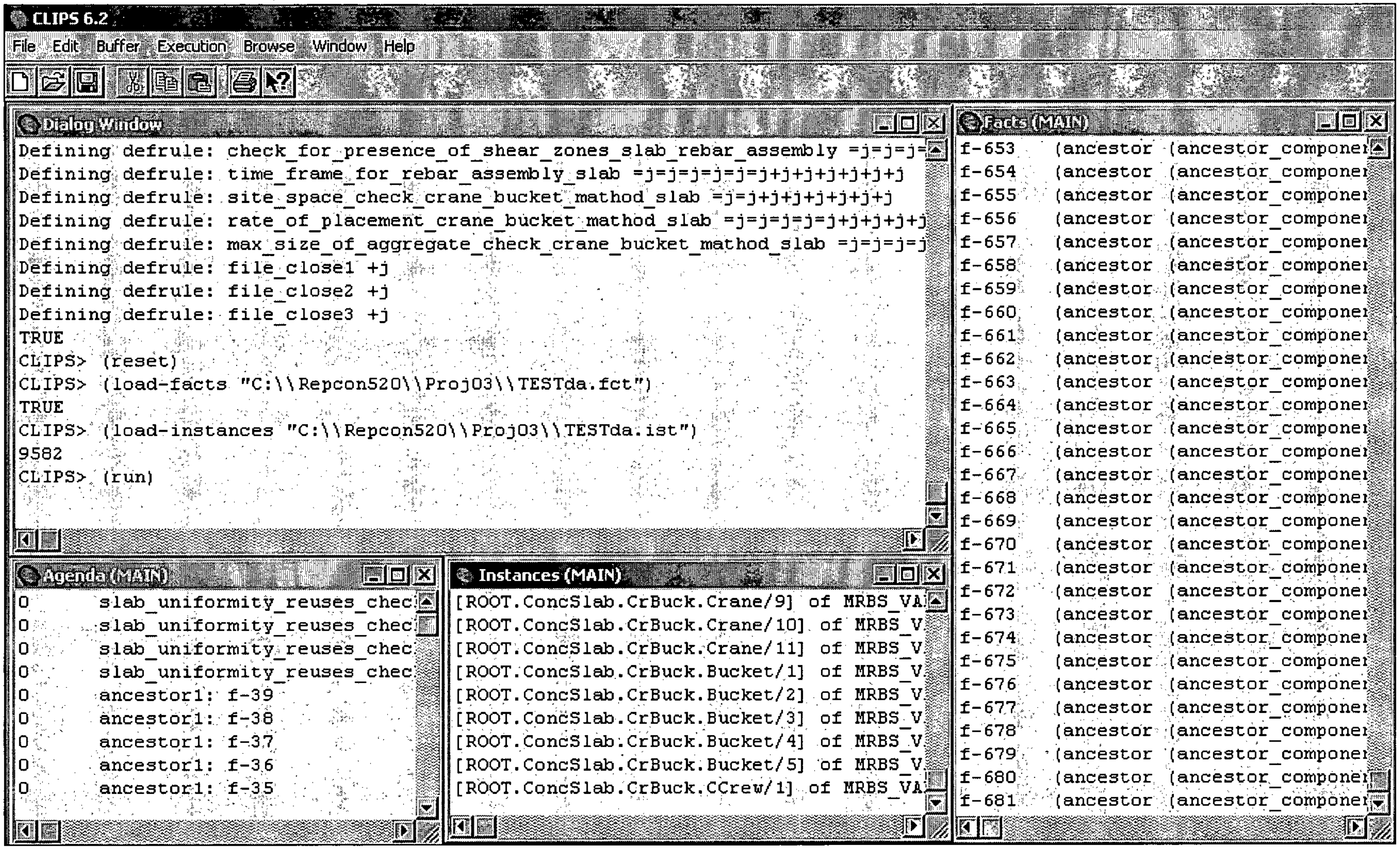


Figure 7.14. During “Run”, the facts, instances, and rules on agenda in the CLIPS environment.

Firing of the relevant production rules indicates that the concrete placement method Crane and Bucket is suitable for slab concrete placement. For the case study project, because of the relatively small size of the floor plate, the crane and bucket method was feasible. However, for a commercial high-rise building, which involves a larger concrete placement quantity, the user can modify the *method statement* by replacing the Crane and Bucket method with a Separate placing boom. Such a changed *method statement* was reasoned for the present case example. The feasibility reasoning report indicates that the method Separate placing boom for concrete placement of column, wall, core, and slab is infeasible because of an insufficient quantity of concrete. The concrete quantity for the whole construction cycle is assumed to be equal to the concrete quantity of one high-rise floor according to conservative assumption that construction cycle duration is of one work week i.e., 5-days.

i.e., *"The Method 'Slab Concrete placement – Separate Placing Boom' is infeasible due to concrete quantity for whole construction cycle is less than required 235 yd3 at location ("2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23") "*.

Even though, the diagnostic report indicates that the separate placing boom method is infeasible, the contractor may choose to keep the same *method statement* to achieve a faster construction cycle and reduce the over all project duration, thereby reducing overall cost of the project and thus making the method feasible (Harvell, 1991) (CC, 1988).

Similarly, by analyzing the feasibility reasoning report, the contractor may choose to use a Hand - set slab forming method for the ground floor location and the non-typical locations "2"and "3" floors, where the Flying truss formwork method is infeasible because of less repetition and presence of slab-bands.

The methods used in the *method statement* for the feasibility reasoning were actually observed on the case study project. However, as indicated in the feasibility reasoning reports, a few methods were found to be infeasible because of the following reasons:

- In the case of the Flying truss formwork method for slab formwork, the contractor had used a larger crew size of 12 crew members, unlike 7 crew members used in the *method statement* employed in feasibility reasoning. Moreover, the time frame we enforced for slab formwork is 8 hours, which is less than the 12 hour workday used by the contractor for the same task on the actual project.

- Similarly, larger crew sizes and time frames were used for rebar placement on the actual project than that of the *method statement* used for feasibility reasoning.

- According to the available knowledge the gang forming methods need at least 30 reuses on a project to be economically feasible. Therefore in the feasibility report file (included in Appendix-H) these methods, in the case study example, are regarded infeasible due to insufficient available reuses. In actual practice formwork, as subcontractors use their formworks on multiple projects makes the gang forming method economically feasible.

In summary, the feasibility reasoning report provides feedback that can be used in preconstruction and pre-bidding brainstorming sessions to determine a feasible *method statement*. The decision support available from the system depends upon the comprehensiveness of the PCBS description, M&RBS description, and quality of the production rules.

Much work remains to formulate a comprehensive set of feasibility screening rules, and reasoning diagnostics. What has been shown, however, is that construction method knowledge can be made explicit, captured in the form of rules, and applied to assist construction personnel assess feasibility of all or parts of a comprehensive *method statement*.

Chapter 8. Conclusion

8.1 Introduction

The primary objective of the thesis was to develop a knowledge management tool for method selection and feasibility reasoning. The emphasis of the work was on giving decision support to the user for method selection and encoding knowledge in a reusable format for use on future projects.

The methodology followed during the thesis work included a literature review on knowledge management and method selection practices, a review of method selection and feasibility factors, semi-structured interviews with construction personnel, characterization of technical feasibility knowledge, selection of an appropriate knowledge representation scheme, formation of feasibility production rules, selection of reasoning schema, and implementation of proof of concept.

8.2 Contributions

The thesis contributes to the state-of-the-art by:

- Examining high-rise construction methods in order to document technical feasibility factors knowledge for formwork, rebar, and concrete placement activities;

- Modeling the feasibility factors knowledge in a reusable format for automated feasibility reasoning during method selection; and,

- Giving the user decision support by using rule-based feasibility reasoning with the help of hierarchical descriptions of a project's physical description and a *method statement* comprised of methods and resources to be used for a predefined construction context. Such a decision support can be provided in any system that uses a hierarchical representation of the physical components of a project (i.e. a product model), and a rich representation of construction methods statement. The current work was developed within the context of the REPCON research system because it supported both representations and thus allowed the author to focus on knowledge capture and feasibility reasoning, without having to develop from scratch other supporting infrastructure. Areas of improvement described in the current system are noted in section 8.3.

The system allows the user to model his information, experience and knowledge pertaining to components to be constructed and relevant construction methods in the form of attributes or feasibility parameters and conditions. These factors are then modeled in the production rules for automated feasibility reasoning and decision support.

8.3 Findings

Important findings from the research are:

- Method selection and method's feasibility factor knowledge exists in various forms, including the academic and trade industry literature. Of special use are the construction case studies available in trade journals such as *Concrete International, Concrete, Concrete Construction,* and *Engineering News Record.* They are rich sources of method selection and feasibility knowledge. Further, seasoned industry personnel hold a wealth of practical knowledge, which can be collected using interview processes.

- By describing project Physical Component Breakdown Structure (PCBS) and Method and Resource Breakdown Structure (M&RBS) hierarchies with the help of standard attribute classes and standard coding schema, the user can repetitively use the feasibility rule file from project to project.

- In this proof of concept of knowledge management tool, production rules have been used. However, it is not always possible to express method selection and feasibility knowledge in declarative[1] form to determine feasibility of a construction method. In such cases production rules giving descriptive text message outputs can be used. These rules are especially useful to highlight theoretical knowledge or information associated with the methods, cost implications, quality management plans and work method related issues, etc.

- A few modifications such as the addition of seventh level (i.e. *subsubelement*) to the PCBS hierarchy and the formation of standard attribute classes have been made in order to assist in modeling and reasoning. It was observed that hierarchies are useful for comprehensive information, experience, and knowledge modeling. The required level of details while modeling, however, depends on the desired decision support from the system.

- Extensive knowledge related to High-Rise concrete construction was captured and elicited in the form of method selection and feasibility factors knowledge. A part of this knowledge is used for the proof of concept in the form of *method statement* rule file.

- The encoding of the method selection and feasibility factors knowledge in the production rule format needs a good working knowledge of CLIPS expert system. It was observed that procedural functions such as If-Then-Else and While loop are helpful in modeling the procedural part of the feasibility checks.

[1] "Declarative knowledge is the surface level information that expert can verbalize." In other words, declarative knowledge is the general heuristics available at a conscious level (McGraw and Harbinson-Briggs, 1989).

- *Method statement* reasoning schema was developed and tested on a full-scale concrete high-rise residential construction. Workability of the system was demonstrated by the comprehensive feasibility reasoning report obtained.

8.4 Recommendations for future work

The present research work did not consider the cost aspect of method selection. Given its importance in decision-making, adding a "cost" facet to the current technical feasibility reasoning would be desirable.

The scope of the present work included formation of feasibility reasoning rules, which requires a good understanding of CLIPS expert system language syntax. It is possible to form standard functions to perform routine procedures such as checking dimensional uniformity over a location range, which will significantly reduce the rule forming and checking work. These functions can be listed in a separate repository, which can be made globally available within the expert system environment. The user can simply pass on arguments to these functions to have a desired feasibility check done from within the rule. This feature will allow rule writing with nominal working knowledge of CLIPS syntax.

The domain of the present research was high-rise construction. The purpose was to explore formwork, rebar placement, concrete placement methods for highly repetitive construction cycles. The high-rise construction methods domain is a reasonably well-researched one, however, greater impact could be achieved by examining more complex projects such as bridges, tunnels, transit guideways, underground utilities, etc., where more variability in site conditions is encountered.

Presently we have implemented the exporting of the PCBS and M&RBS hierarchies to the CLIPS environment with the help of softcodes as explained in Chapter 5, which is sufficient for demonstrating proof of concept. The present redundancy of rule evaluation, such as the testing of site space requirements for the crane and bucket method for each of its uses, can be avoided by implementation of the rule-tagging feature explained in Chapter 5. Further, the fine-tuning of feasibility reasoning rules and creation of rule repositories is desirable. Last but not the least, an intuitive interface for rule writing as well as an interface for feasibility reasoning and report generation is essential for use of the reasoning schema in practice.

BIBLIOGRAPHY

Aalami, F. B. (1998). "Using Construction Method Models to Generate 4D Production Models." *Ph.D. Thesis*, CIFE, Stanford University.

AbouRizk, S., Mather, K. (1998). "A CAD-Based Simulation Tool for Earthmoving Construction Method Selection." *Computing in Civil Engineering*, ASCE, 39-52.

ACPA (2001). *Operator Safety Guide: Separate Placing Booms*, American Concrete Pumping Association, 1 October 2002 <www.concretepumpers.com/safetySGseparate.asp>.

Adeli, H. (1988). *Expert Systems in Construction and Structural Engineering*, Chapman & Hall, ISBN- 0412289105.

Al-Ghassani, A. M., Kamara, J. M., Anumba, C. J., and Carrillo, P. M. (2002). "A Tool for Developing Knowledge Management Strategies." *Electronic Journal of Information Technology in Construction*, Royal Institute of Technology, 7, 1 October 2002 <http://www.itcon.org/2002/5/paper.pdf>.

Al-Hammad, I. (1991). "A Knowledge-Based Framework for Construction Method Selection." *Ph.D. Thesis*, University of British Columbia.

Alkass, S., Aronian, A. (1990). "Computer Aided Equipment Selection for Concrete Placing." *Concrete International*, ACI, 12 (12), 38-45.

Alkass, S., Aronian, A., and Moselhi, O. (1990). "Computer-Aided Equipment Selection for Transporting and Placing Concrete." *Journal of Construction Engineering and Management*, ASCE, 119 (3), 445-465.

Alkass, S., Harris, F. (1988). "Expert System for Earthmoving Equipment selection in Road Construction " *Journal of Construction Engineering and Management*, ASCE, 114 (3), 426-440.

Allouche, E. (2001). "Decision-Support Model for Selection of a Trenchless Construction Method." *Ph.D. Thesis*, University of Alberta.

BAMTEC (No Date). "Concrete Slab Reinforcement System." *Product Brochure*, One Steel Reinforcing, 1 October 2002 <http://www.onesteel.com/reinforcing/products/BAMTEC_Concret_Slab_System.pdf>.

Bennett, D. F. H., MacDonald, L. A. M. (1992). *Economic Assembly of Reinforcement*, British Cement Association, ISBN- 0721014135.

BPG (2001). "Prefabricated Punching Shear Reinforcement for Reinforcing Concrete Flat Slabs." *Best Practice Guides for In-Situ Concrete Frame Buildings*, British Cement Association, ISBN- 0721015786.

Camellerie, J. F. (1978). "Vertical Slipforming as a Construction Tool." *Concrete Construction*, The Aberdeen Group, May Issue.

CC (1988). "Frequent Flyer Program for Concrete Booms." *Concrete Construction*, The Aberdeen Group, June Issue.

CC (1983). "Slabs Without Vertical Shoring." *Concrete Construction*, The Aberdeen Group, June Issue.

CC (1982). "Climbing Tower Crane and Extended-Life Superplasticizer Solve Difficult Placement Problems." *Concrete Construction*, The Aberdeen Group, September Issue.

Chalabi, A. F., Christopher Y. (1989). "Crane, an Expert system for Optimal Tower Selection and Placement." *Computing in Civil Engineering*, ASCE, 290-297.

Chevallier, N. J. (1998). "Automated Draft Plan and Schedule Generation Using Templates, Physical Breakdown Structures and Expert Systems." *M.A.Sc. Thesis*, University of British Columbia.

CLIPS (2002). *CLIPS Reference Manual Version 6.20*, GHG Internate Services, 1 June 2002 <http://www.ghg.net/clips/CLIPS.html>.

Crepas, R. A. (1985). *Pumping Concrete: Techniques and Applications*, Concrete Construction Publications.

Davenport, T. H., Prusak, L. (1998). *Working Knowledge: How Organizations Manage What They Know*, Harvard Business School Press, ISBN- 0875846556.

EFCO (2001). "Half the time." *Form Marks*, Economy Forms Ltd., Spring Issue.

EFCO (2000a). "E-Z Deck: Decking and Shoring System." *Product Brochure*, Economy Forms Ltd.

EFCO (2000b). "Hand-E-Form: Handset Forming System." *Product Brochure*, Economy Forms Ltd.

EFCO (2000c). "Four Pours, Fifteen Days." *Form Marks*, Economy Forms Ltd., Fall Issue.

EFCO (1999). "Casino Expansion." *Form Marks*, Economy Forms Ltd., Fall Issue.

EFCO (1994). "Plate Girder Form System." *Forming Systems Catalog*, Economy Forms Ltd.

Egbu, C., Botterill, K. (2001). "Knowledge Management and Intellectual Capital: Benefits for Project Based Industries." *CORBA 2001 Conference*, RICS Foundation, UK.

Elhag, T. M. S., Deason, P. M., Morris, P. W. G., and Patel, M. B. (2000). "Development of a Knowledge System for a Construction Contractor." *Project Management Institute Annual Conference*, PMI: New Zealand.

Fischer, M. A. (1993). "Automating Constructability Reasoning with a Geometrical and Topological Project Model." *Computing Systems in Engineering*, Pergamon Press, 4 (2-3), 179-192.

Fischer, M. A. (1991). " Using Construction Knowledge During Preliminary Design of Reinforced Concrete Structures." *Ph.D. Thesis*, CIFE, Stanford University.

Fisher, T. S. (1997). "Spotting Boom Pump Safety." *Concrete Construction*, The Aberdeen Group, November Issue.

Froese, T., Rankin, J. (1998). "Construction Methods in Total Project Systems" *Proceedings of the International Computing Congress*, Boston, ASCE, 383-394.

Fulton, W. R. (1989). "Self-Lifting Forms." *Concrete Construction*, The Aberdeen Group, November Issue.

Galagan, P. A. (1997). "Smart Companies." *Training & Development*, Madison Wisc., 51 (12), 20-24.

Ganeshan, R., Stumpf, A., Chin, S., Liu, L., and Harrison, B. (1996). "Integrating Object-Oriented CAD and Rule-Based Technologies for Construction Planning." *USACERL Technical Report 96/82*, US Army Corps of Engineers, Construction Engineering Research Laboratories.1 October 2002 <www.cecer.army.mil/TechReports/ stu_ruby/ stu_ruby.post.pdf>.

Giarratano, J., Riley, G. (1998). *Expert Systems: Principles and Programming*, PWS Publishing Company, 3, ISBN- 0534950531.

Gil, N., Tommelein, I. D., Kirkendall, R. L., and Ballard, G. (2001). "Leveraging Speciality-Contractor Knowledge in Design-Build Organizations." *Engineering, Construction and Architectural Management*, Blackwell Science Ltd., 8 (5/6), 355- 367.

GoodChild, C., Moss, R. (1999). "Reinforcing Cardington." *Concrete*, Eng. Concrete Society, 33(1), 11-14.

Gray, C. (1986). "Intelligent Construction Time and Cost Analysis." *Construction Management and Economics*, 4 (2), 135-150.

Hanna, A. S. (1998). *Concrete Formwork Systems*, Marcel Dekker Inc., ISBN-0824700724.

Hanna, A. S. (1994). "SELECTCRANE: An Expert System for Optimum Crane Selection." *Proceedings of the 1st Congress on Computing in Civil Engineering,* ASCE, 1, 958-963.

Hanna, A. S., Sanvido, V. E. (1991). "Interactive Horizontal Formwork Selection System." *Concrete International*, ACI, 13(8), 50-56.

Hanna, A. S., Sanvido, V. E. (1990). "Interactive Vertical Formwork Selection System." *Concrete International*, ACI, 12 (4), 26-32.

Hanna, A. S., Willenbrock, J. H., and Sanvido, V. E. (1992). "Knowledge Acquisition and Development for Formwork Selection System." *Journal of Construction Engineering and Management*, ASCE, 118 (1), 179-198.

Harvell, D. B. (1991). "Consider All Costs When Choosing a Concrete Placing Method." *Concrete Construction*, The Aberdeen Group, August Issue.

Hastak, M. (1998). "Advanced Automation or Conventional Construction Process ?" *Automation in Constriction*, Elsevier Science, 7(4), 299-314.

Hendrickson, C., Zozaya-Gorostiza, C., Rehak, D. (1989). *Knowledge Based Process Planning for Construction and Manufacturing*, Academic Press, Boston, Massachusetts, U.S.A., ISBN 0127819002.

Hendrickson, C., Zozaya-Gorostiza, C., Rehak, D., Baracco-Miller, E., and Lim, P. (1987). "Expert System For Construction Planning." *Journal of Computing in Civil Engineering*, ASCE, 1(4), 253-269.

Hurd, M. K. (1995). *Formwork for Concrete*, ACI, SP-4, 6th Edition.

Hurd, M. K. (1991). "Bracing for Wall Formwork." *Concrete Construction*, The Aberdeen Group, July Issue.

Industry Canada (2002). "Sector Competitive Frameworks Series: Growth Prospects for the Industry." *Industry Canada*, 1 October 2002 <http://strategis.ic.gc.ca/SSG/1/ce01318e.html>.

Jagbeck, A., (1994). "MDA Planner: Interactive Planning Tool Using Product Models and Construction Methods." *Journal of Computing in Civil Engineering*, ASCE, 8(4), 536-554.

Kamara, J. M., Anumba, C. J., and Carrillo, P. M. (2001). "Knowledge Management in a Multi-Project Environment in Construction." *Proceedings of the First International Structural Engineering and Construction Conference: Creative Systems in Structural and Construction Engineering*, Rotterdam: Balkema, 321-326.

Kamarthi, S. V., Sanvido, V. E., and Kumara S. R. T. (1992). "NEUROFORM – Neural Network System for Vertical Formwork Selection." *Journal of Computing in Civil Engineering*, ASCE, 6 (2), 178-199.

Kartam, N., Al-Tabtabai, H. (1995). "Interactive Knowledge-Intensive System for Constructability Improvement." *Proceedings of the Fourth Construction Congress*, ASCE, 322-330.

KLICON (1999). The Role of Information Technology in Knowledge Management within the Construction Industry, *Project Report*, CRMP, University of Manchester Institute of Science and Technology, UK.

Koel, L. (1997). *Concrete Formwork,* American Technical Publishers, ISBN-0826907067.

Koo, T. K., Tiong, R. L. K., Wong, T. F., and Tay, M. C. (1992). "Design and Development of an Intelligent System for Formwork Selection." *Structures and Buildings*, Proceedings of the Institute of Civil Engineers, 95 (1), 73-91.

Kululanga, G. K., McCaffer, R. (2001). "Measuring Knowledge Management for Construction Organizations." *Engineering, Construction and Architectural Management*, Blackwell Science Ltd., 8 (5/6), 346-354.

Laufer, A., Shapira, A., Cohenca-Zall, D., and Hall, G. (1993). "Prebid and Preconstruction Planning Process." *Journal of Construction Engineering and Management*, ASCE, 119 (3), 426-444.

Lewis, A. (1998). "In the Pipeline." *Concrete*, Eng. Concrete Society, 32(8), 12-14.

McGraw, K., Harbinson-Briggs, K. (1989). *Knowledge Acquisition: Principles and Guidelines*, Prentice Hall Publication, ISBN- 0135164362.

Morey, D., Maybury, M., Thuraisingham, B. (2000). *Knowledge Management: Classic and Contemporary Works*, The MIT Press, ISBN- 0-262-133384-9.

Navinchandra, D., Sriram, D., Logcher, R. D. (1988). "GHOST: Project Network Generator." *Journal of Computing in Civil Engineering*, ASCE, 2 (3), 239-253.

Patent (2001). "Mod-U-Form: Field Forming Manual." *Product Brochure*, Patent Construction Systems.

Patent (2000). "Interform: Aluminum Flying Form System." *Product Brochure*, Patent Construction Systems.

Patent (1999). "Studform: The Interform Joist and Waler Wall Forming System." *Product Brochure*, Patent Construction Systems.

Peurifoy, R. L., Oberlender, G. D. (1995). *Formwork for Concrete Structures*, McGraw-Hill, 3, ISBN-0070498385.

Proverbs, D. G., Holt, G. D., and Olomolaiye, P. O. (1999). "Construction Resource / Method Factors Influencing Productivity for High-Rise Concrete Construction." *Construction Management and Economics*, E & FN SPON, 17 (5), 577-587.

Putzmeister (2001a). "Telebelt Putzmeister" *Product Brochures*, Putzmeister America, 1 October 2002 <http://www.putzmeister.com/products/telebelt/>.

Putzmeister (2001b). "Truck-Mounted Concrete Boom Pumps" *Product Brochures*, Putzmeister America, 1 October 2002 <http://www.putzmeister.com/products/truckmount/>.

Putzmeister (2001c). "Trailer-Mounted Pumps" *Product Brochures*, Putzmeister America, 1 October 2002 <http://www.putzmeister.com/products/trailermount/index.cfm>.

Putzmeister (2001d). "Separate Placing Booms" *Product Brochures*, Putzmeister America, 1 October 2002 <http://www.putzmeister.com/products/placingboom/>.

Quinton, M. E. (1991). "Room Tunnel Forming: A Day in the Life..." *Concrete Construction*, The Aberdeen Group, November Issue.

Rankin, J. H. (2000). "Computer-Assisted Construction Planning." *Ph.D. Thesis*, University of British Columbia.

Reed, P. R. (2000). *Developing Applications with Visual Basic and UML*, Addison Wesley, ISBN-0-201-61579-7.

Rezgui, Y., Cooper, G., and Brandon, P. (1998). "Information Management in a Collaborative Multiactor Environment: The COMMIT Approach." *Journal of Computing in Civil Engineering*, ASCE, 12 (3), 136-144.

Rivard, H. (2002). "Information Technology in AEC Industry – An Overview." *Canadian Civil Engineer*, CSCE, 19 (2), 10-11.

Robinson, H. S., Carrillo, P. M., Anumba, C. J., and Al-Ghassani, A. M. (2001). "Perceptions and Barriers in Implementing Knowledge Management Strategies in Large Construction Organizations." *CORBA 2001 Conference*, RICS Foundation, UK.

Rowley, J. (1999). "What is Knowledge Management ?" *Library Management*, Emerald, 20 (8), 416-420.

Russell, A. D., Chevallier, N. J. (1998). "Representing Project's physical view in support of project management functions." *Canadian Journal of Civil Engineering*, NRC, 25 (4), 705-717.

Russell, A. D., Sharma, A., and Udaipurwala, A. (1997). "Construction Method Selection and Cycle Design." *Proceedings of the Fifth Construction Congress: Managing Engineering Construction in Expanding Global Markets*, ASCE, 486-493.

Russell, A. D., Udaipurwala, A., Alldritt, M., El-Guindy, K. (1999). "Computer System for the Selection of Trenchless and Conventional Methods for Underground Utilities." *Tunneling and Underground Space Technology*, Elsevier Science Ltd., 14 (2), 1-12.

Sawhney, A., Mund, A. (2002). "Adaptive Probabilistic Neural Network-based Crane Type Selection System." *Journal of Construction Engineering and Management*, ASCE, 128 (3), 265-273.

Scheuer, J., Budenski, J. (1993). "Tunnel Forms Speed Construction of Multistorey Residence Hall." *Concrete Construction*, The Aberdeen Group, April Issue.

Sharma, A. (1997). "Computer Assisted Methods Selection for High-Rise Construction." *M.A.Sc. Thesis*, University of British Columbia.

Skibniewski, M. J., Chao, L. (1992). "Evaluation of Advanced Construction Technology with AHP Method." *Journal of Construction Engineering and Management*, ASCE, 118 (3), 265-273.

Slagle, M. (1997). "Choosing a Concrete Conveyor." *Concrete Construction*, The Aberdeen Group, May Issue.

Syal, M.G. (1992). "Construction Process Knowledge Model to Assist Method Selection in Project Planning." *Ph.D. Thesis*, The Pennsylvania State University.

Syal, M.G., Parfitt, M. K., and Willenbrock, J. H. (1993). "Design Information Hierarchy for Construction Method Selection Process." *Computing in Civil and Building Engineering: Proceedings of the Fifth International Conference (V- ICCCBE)*, Anaheim, California, 1, 742-749.

Theophilus, J. (1995). "Rationalised Reinforcement Design." *Concrete*, Eng. Concrete Society, 29(2), 17-18.

Tiwana, A. (2000). *The Knowledge Management Toolkit: Practical Techniques for Building a Knowledge Management System,* Prentice Hall PTR, ISBN- 0130128538.

TOPEC (2000). "Topec Soffit Formwork." *Product Brochure: Instructions for Erection and Reuse*, Thyssen Hunnebeck Schalung Company, 1 October 2002 <http://www.huennebeck-schalung.de/english/leistungen/fs_index.php3?ziel =fs_schalungen_e.html>.

TOPEC (2001). "The 2 Component Slab Formwork System." *Product Brochure*, Safway Formwork Systems LLC, 1 October 2002 <http://www.safway.com/index.html>.

Turban, E. (1990). *Decision Support and Expert Systems: Management Support Systems*, Macmillian Publication, 2, ISBN- 0024216631.

Turban, E., Aronson, J. E. (1998). *Decision Support Systems and Intelligent Systems*, Prentice-Hall, 5, ISBN- 0-13-740937-0.

Udaipurwala, A. (1997). "Hierarchical Representation of Information for Construction Methods Selection." *M.A.Sc. Thesis*, University of British Columbia.

Udaipurwala, A., Russell, A. (2002). "Computer-Assisted Construction Methods Knowledge Management and Selection." *Canadian Journal of Civil Engineering*, NRC, 29 (3), 499-516.

Ugwu, O. O., Tah, J. H. M. (1998). "Towards Optimising Construction–Method Selection Strategies Using Genetic Algorithms." *Engineering Applications of Artificial Intelligence*, Pergamon Press, 11 (4), 567-577.

UML (2001). "OMG Unified Modeling Language Specification." *UML 1.4 Specification*, Object Management Group, 1 October 2002 <http://www.omg.org/cgi-bin/doc?formal/01-09-67.pdf>.

Wallace, M. (1985). "Cast Walls and Slab at the Same Time with Tunnel Forms." *Concrete Construction*, The Aberdeen Group, February Issue.

Wallace, M. (1997). "Raising the Rio." *Concrete Construction*, The Aberdeen Group, May Issue.

Wallace, M. A. (1998). "Finding the Best Spot for a Boom Pump." *Concrete Construction*, The Aberdeen Group, January Issue.

Waugh, L. M., (1989). "Knowledge-based Construction Scheduling." *Proceedings of Computing in Civil Engineering*, ASCE, 84-91.

WCB (no date). "Introduction to Personal Fall Protection Equipment: Safety Belt, Harnesses, Lanyards and lifelines." *Health and Safety Topics*, Workers Compensation Board of BC, 12 October 2002 < http://www.worksafebc.com/publications/health_and_safety_information/by_topic/assets/pdf/fall_protection.pdf>.

Winstanley, G., Chacon, M. A., and Levitt, R. E. (1993). "Model-Based Planning: Scaled-Up Construction Application." *Journal of Computing in Civil Engineering*, ASCE, 7(2), 199-217.

Yeang, K. (2000). *Service Cores*, Wiley-Academy, ISBN- 047197904 X.

Zizette, B. (1998). "A Purely Object-Oriented Approach for Rule-Based Paradigms." *Expert Systems With Applications*, Elsevier Science Ltd., 14 (4), 483- 492.

APPENDICES

APPENDIX – A

Method Selection and Feasibility Factors Knowledge

Appendix A Method Selection and Feasibility Factors Knowledge

A1. List of Construction Personnel Interviewed / Referred

(Bichel) Bichel, Pat; General Manager, Plains Reinforcing Ltd., Surrey, BC. *(Rebar placement)*

(DeBruin) De Bruin, Henk; Manager, Outinord Universal Inc., Miami, FL, USA. *(Tunnel formwork)* www.outinord.com

(Fallowfield) Fallowfield, Rob; P.Eng., Outinord Universal Inc., Miami, FL, USA. *(Tunnel formwork)* www.outinord.com

(Fradley) Fradley, Wayne; Operations Manager, Raymond Rebar Inc., Surrey, BC. *(Rebar placement)* www.rrebar.com

(Gastaldo) Gastaldo, Paolo; Estimator/Manager, Gastaldo Concrete Ltd., Delta, BC. *(Concrete placement)* www.gastaldoconcrete.com

(Heinz) Heinz, Dale; District Manager, EFCO Corp., Calgary, Alberta. *(Formwork)* www.efco-usa.com

(Holm) Holm, Garret; Formwork Designer, EFCO Corp., Kent, WA, USA. *(Formwork)* www.efco-usa.com

(Kennedy) Kennedy, Rod; Manager, Grand Sierra Constructions Ltd., Surrey, BC. *(Method selection)*

(McFEE) McFEE, Ron; Manager, Preconstruction services, Stuart Olson Construction, Richmond, BC. *(Method selection)*

(Newell) Newell, Ted; Formwork Designer, Ted Newell Engineering Ltd., Vancouver, BC. *(Formwork)*

(Shaw) Shaw, Al; Manager, Prebar Inc., Surrey, BC. *(Rebar placement)*

(Stefanich) Stefanich, Joe; Detailer / Coordinator, Harris Rebar, Delta, BC. *(Rebar placement)* www.harrisrebar.com

(Yaeger) Yaeger, Mark; Superintendent, Stuart Olson Constructions Ltd., Richmond, BC. *(Method selection)* www.stuartolson.com

(Young) Young, Norm; Manager / Estimator, Willow Bay Constructions Ltd., Surrey, BC. *(Formwork)* www.willowbayforming.com

A2. Feasibility Factors Knowledge Regarding Construction Methods

I. Formwork Systems

a) Slab formwork systems

1. Conventional wooden formwork system
2. Conventional metal formwork system
3. Flying Truss formwork system
4. Column-mounted flytable formwork system
5. Tunnel formwork system
6. Handset formwork system

1. Conventional wooden formwork system

Site Characteristics

1. Site storage space area should be within 500 to 1000 ft^2. Built floor area can also be used as a storage area (Young).

Structural characteristics

1. The system is feasible on "non-typical" locations e.g., parking structure, non-typical floor locations, etc.
2. The shape of the slab-bay can be varying i.e. not constant for high-rise floors.
3. The area of the slab-bay can be varying i.e. not constant for high-rise floors.
4. The system is feasible when the slab-bay has beams, spandrel beams, and slab-bands (Newell).
5. The system is feasible when the sizes of beams, spandrel beams, slab-bands vary from location to location (Newell).
6. The system is feasible when storey height is less than 14 ft (Hanna, 1991).
7. The system is feasible when less than 5-6 reuses are required (Hanna, 1998).

Production characteristics

1. The rate of production achievable is up to 15 ft^2 / manhr (Young).
2. The formwork crew has normally 9 crew members (Young).

2. Conventional metal formwork system

Site characteristics

1. Site storage space area should be at least 500 ft^2; usually already built areas are used for storing shoring towers frames (Young).

Structural characteristics

1. The system is feasible on "non-typical" locations parking structure, non-typical floor locations, etc.
2. The shape of the slab-bay can vary i.e. not constant for high-rise floors.
3. The area of the slab-bay can vary i.e. not constant for high-rise floors.
4. The system is feasible when the slab-bay contains beams, spandrel beams, and slab-bands.
5. The system is feasible when the size of beams, spandrel beams, slab-bands vary from location to location.
6. The system can be used for storey heights up to 40 ft (EFCO, 2000a).

Production characteristics

1. The rate of production is 25 ft^2 / manhr (EFCO, 2001).
2. The formwork crew has 9 crew members (Young).

3. Flying Truss formwork system

Site Characteristics

1. Site Assembly space lengths should be at least equal to the maximum length of the flytable truss.
2. Site Assembly space widths should be at least equal to the maximum width of the flytable truss.
3. Site Assembly space area should be at least 1200 ft^2; there should be at least 2 to 4 flying trusses assembly or dismantling space available on site (Young). The trusses can be assembled in-place at typical floor locations, but this method uses critical crane hours for material transportation.

Structural characteristics

1. The location should be a "typical location" e.g., High-rise floor.
2. The shape of the slab-bay[1] should be constant.
3. The area of the slab-bay should be constant for high-rise floors (Hanna, 1991) (Fischer, 1991).
4. The slab-bay should not contain slab-bands, beams, spandrel beams, or drop panels.
5. If the slab-bay has drop panels, maximum width and maximum length of the drop panels should be taken into consideration for optimum flytable size determination (Newell).

[1] "Slab-bay" is an arbitrary concept, which can be defined as the slab portion that is supported by walls or columns (Newell) taking into consideration the possible orientation of flying trusses or column-mounted flytables.

6. If the slab-bay has beams and spandrel beams, then their size should not vary more than 10 %, for high-rise floors (Hanna, 1991).
7. If the slab-bay has spandrel beam, then the truss height of the flytable should be less than or equal to the storey height minus the spandrel beam depth (Heinz). Truss height of flytable <= (story height to spandrel beam depth).
8. The system is feasible when the story height is within 7ft to 20 ft (Patent, 2000).
9. For the system to be economical the slab-bay width should be between 15 ft to 30ft. (Fischer, 1991), (Hanna, 1998).
10. The economical length of slab-bay is 22 ft (Fischer, 1991) (Hanna, 1998).

Production characteristics

1. The rate of production is up to 70 ft^2 / manhr (Young).
2. The minimum reuses available should be at least 5-6 (Heinz) (Newell).
3. The flytable flying crew has 10 crew members (Young).
4. The open space must be at least equal to 1.5 times the maximum length of the flytable (Hanna, 1991).
5. The size of the flying truss table is usually dictated by the lifting capacity of the crane (Newell).

4. Column-mounted flytable formwork system

Site characteristics

1. Site Assembly space lengths should be at least equal to the maximum length of the flytable.
2. Site Assembly space widths should be at least equal to the maximum width of the flytable.
3. Flytables are either assembled on site or the pre-assembled modules are bolted on site (Hanna, 1998) (CC, 1983).

Structural characteristics

1. The location should be a "typical location" e.g., High-rise floors.
2. The shape of the slab-bay should be the same for all high-rise floors (Heinz) (Holm).
3. The shape of the slab-bay should be "rectangular". The columns or supporting walls need to be in a straight line (Holm) i.e., the supporting sides of flytable should be parallel to each other (Wallace, 1997)[2].
4. The area of the slab-bay should be constant for all high-rise floors. Slab-bay width should remain constant for at least 6 to 8 floors (Hanna, 1998).

[2] A case study indicated in the article *Raising the Rio*, Wallace (1997) described that the wedge-shaped column mounted flytable posed difficulties due to non-parallel supporting sides. "The workers had to move the column-mounted rollers back and forth while setting the forms; otherwise the forms would slip off the rollers." The contractor eventually switched to the flying truss method.

5. The system is feasible when it has at least 8 to10 reuses (Hanna, 1998).
6. The slab-bay should not contain slab-bands, beams, spandrel beams, drop panels i.e. the slab-bay should be a "flat plate" (Holm) (Hanna, 1998).
7. If the slab-bay has a spandrel beam it should not be more than14 inches deep for economical use of the system (Hanna, 1998).
8. The optimum slab-bay width is between 16 ft to 20ft (Hanna, 1998).
9. The optimum length of slab-bay is between 30 to 40ft (Hanna, 1998).

Production characteristics

1. The rate of production of the system is equal to 45 ft^2 / manhr (Hanna, 1991).
2. The rate of production can be up to 70 ft^2 / manhr (EFCO, 1999).
3. The formwork crew has 9 crew members (Hanna, 1998).
4. The open space must be at least equal to 1.5 times the maximum length of the flytable (Hanna, 1991).

5. Tunnel formwork system (Outinord Universal Inc) (DeBruin; Fallowfield)

Site characteristics

1. Site Assembly space area should be at least 6000 ft^2, as the tunnel forms arrive in modular sections and need to be bolted together.

Structural characteristics

1. The location should be a "typical" location e.g., High-rise floors.
2. The shape of the slab-bay should be constant for each reuse of the tunnel, with some minor width adjustment allowed via hinge panels.
3. The slab-bay should be supported by walls.
4. The wall should not have offsets, pilasters, or corners.
5. The height of wall should be constant for all floor locations.
6. The height of wall should be within 7.5 ft to 12 ft
7. The area of the slab-bay should be constant for each tunnel reuse.
8. A slab thickness within 5 to 7 in. is the most economical.
9. For use of tunnel form the slab-bay should not contain slab-bands, beams, spandrel beams, drop panels i.e. the slab-bay should be "flat plate". Slab beams within the constant slab depth are acceptable.
10. For use of a tunnel form the most economical slab-bay width is between 8 ft to 18ft. the maximum possible width is up to 32 ft.
11. For use of tunnel form the length of slab-bay should be less than 40 ft, although 80 ft length is achieved by placing tunnels end to end.
12. The quality of surface finish required is "smooth finish", ready for skim coat only.
13. The open space around the building should be at least 1.2 times the maximum length of the tunnel form for easy maneuverability around the building when flying tunnels.

14. The average weight of the tunnel formwork is 20.5 lbs / ft^2.
15. There should be at least 100,000 ft2 floor area for tunnel formwork to be economically feasible.
16. There should be at least 40 concrete pours (daily cycles) for tunnel formwork to be economically feasible.

Production characteristics

1. The rate of production of the method is approximately equal to 50 ft^2 / manhr.
2. The formwork crew has 15 to 18 crew members.
3. The formwork system is compatible with "prefabrication of reinforcement" (welded wire fabric) for walls and slab deck to maintain a daily construction cycle.
4. The system requires normal strength concrete and heat curing (in colder climates) to achieve overnight stripping strength of 33% of the designed strength, to allow stripping 12 hr cured concrete.
5. The construction cycle should be 24 hours (1 day) for effective use of tunnel formwork.

6. Hand set slab formwork system (Topec, 2001) (Topec, 2000)

Site characteristics

1. No feasibility knowledge was observed regarding site characteristics.

Structural characteristics

1. The location should be a "non-typical" location e.g., parking structure.
2. The method is feasible when the shape of the slab-bay is varying i.e. not constant
3. The area of the slab-bay is varying i.e. not constant
4. Feasible if the slab-bay contains beams, spandrel beams, slab-bands, column capitals etc.
5. Feasible if the size of beams, spandrel beams, slab-bands, column capitals varying from location to location.
6. The quality of surface finish required is "smooth finish" (Topec, 2001).
7. The slab-bay has camber this system is especially feasible (Topec, 2001).
8. The system is feasible when storey height is up to 19 ft (Topec, 2001).
9. The slab thickness is up to 22 inches (Topec, 2000).

Production characteristics

1. The rate of production required is 35 to 50 ft^2 / manhr (Topec, 2000).
2. The formwork crew has 3 to 4 crew members (Topec, 2000).

b) Wall / Core wall / Column formwork systems

1. Conventional wooden formwork system
2. Steel framed modular formwork system
3. All steel modular formwork system
4. Wooden gang form system
5. Aluminum waler gang form system / Jump form system
6. All steel gang form system / Jump form system
7. Tunnel formwork system
8. Self climbing system
9. Slip form system

1. Conventional wooden formwork system (Young) (Newell)

Site characteristics

1. Site storage space area should be at least 500 ft^2 (Young).

Structural characteristics

1. The system is feasible when the size of the offsets, inserts, corners, pilasters is varying i.e. not constant.
2. The system is economical when the available reuse for elements is up to 3 to 4 reuses (Hanna, 1998).
3. The wall thickness can be within 6 to 16 inches. (Koel, 1997).
4. The height of the wall is generally limited to 4 ft (Koel, 1997).
5. Using proprietary column clamps, column forms can be constructed up to 16 ft in height (Peurifoy, 1995).
6. The rate of pour for wall formwork is up to 4 ft / hr (Newell).
7. The allowable pour pressure is 600 to 750 psf. (Newell).
8. The column formwork is usually designed for full liquid head i.e., rate of pour is 8 ft / hr (Newell).
9. The column formwork is usually designed for pour pressure 1200 psf (Newell).

Production characteristics

1. The rate of production is less than 19 ft^2 / manhr (Young).
2. The "formwork crew" is of size 7 crew members (Young).

2. Steel framed modular formwork system (Young) (Newell)

Site characteristics

1. Site storage space area should be at least 500 ft^2 (Young).

Structural characteristics

1. If the wall has offsets, inserts, corners, pilasters then this system is still feasible.
2. The size of the offsets, inserts, corners, pilasters is varying i.e., not constant for all high-rise locations.
3. The formwork can be used on walls ranging from 4 to 10 ft in height (Hurd, 1995).
4. The rate of pour is generally 7ft / hr (Newell).
5. The allowable pour pressure is 1200 psf. (Newell).
6. Typically tie spacing is 2 ft horizontally and 1 ft vertically (Patent, 2001).

Production characteristics

1. The rate of production is equal to 19 ft^2 / manhr approx. (Young).
2. The formwork crew has 7 crew members (Young).

3. All Steel Modular formwork

Site characteristics

1. Site storage space area should be at least 500 ft^2 (Young).

Structural characteristics

1. If wall has offsets, inserts, corners, pilasters then this system is still feasible.
2. The size of the offsets, inserts, corners, pilasters is varying i.e., not constant.
3. The system is generally not used for cores because of very close tie spacing; it is preferred for foundation work, walls, and columns (Holm).
4. The quality of surface finish is "smooth finish" (EFCO, 2000b).
5. For columns with width 10 to 30 inches no ties are required. All metal modules need three bolts for 8 ft height column corner (EFCO, 2000b).
6. The rate of pour is less than 7ft / hr (Newell) (Holm).
7. The allowable pour pressure is less than 1200 psf (Newell) (Holm).
8. The allowable tie spacing is generally 2 ft horizontal and 2 ft vertical (Holm).

Production characteristics

1. The rate of production can be up to 65 ft^2 / manhr (EFCO, 2000c).
2. The formwork crew has 6 crew members (EFCO, 2000c).

4. Wooden gang form

Site characteristics

1. Site storage space area is required to assemble or store the largest gang form panel. These panels can be assembled on ground where it is easier to work .The gang size can be up to 30 x 50 ft (Hurd, 1995).

Structural characteristics

1. The method is not suitable for walls with pilasters (Hanna, 1998). The offset and corners variations from floor to floor can be can be adjusted by considering their maximum sizes and the use of fillers (Newell).
2. The available reuse of formwork should be at least 30 to 40 (Backe, 1986).
3. The rate of pour is usually equal to 4ft / hr (Newell).
4. The allowable pour pressure is 600 to 750 psf (Newell).
5. The allowable tie spacing is between 3 ft to 4 ft (Newell).
6. Wooden column gang forms are generally designed for full liquid head (8 ft) i.e., 1200 psf (Newell).
7. The rate of pour for column formwork is usually 8 ft/ hr i.e., full liquid head (Newell).
8. The tie spacing for a column form is 2 ft 6 in (Newell).

Production characteristics

1. The rate of production required is about 35 ft^2/ manhr (Young).
2. The formwork crew has 7 crew members (Young).

5. Aluminum waler gang form / jump form system

Site characteristics

1. Site storage space area is required to assemble or store largest gang form panel.

Structural characteristics

1. The method is not suitable for walls with pilasters (Hanna, 1998). The offset and corners variations from floor to floor can be adjusted by considering their maximum sizes and the use of fillers (Newell).
2. The available reuse of formwork should be 30 to 40 (Backe, 1986).
3. Length of wall formwork can be up to 40ft (Patent ,1999).
4. Height of the wall formwork can be up to 22ft (Patent, 1999).
5. Jump forms can be 8 to 16 ft high and they can be 8 to 44 ft wide (Peurifoy, 95).

6. Jump form system needs a 5 ft wide operating platform (Peurifoy, 1995).
7. The rate of pour is usually between 6 to 9 ft/hr (Newell).
8. The allowable pour pressure is 1200 psf (Newell), but maximum designed pour pressure can be up to 2250 psf (Patent, 1999).
9. The allowable tie spacing for wall form is 6 ft (Newell).
10. The tie spacing for column form is usually 5 ft (Newell).

Production characteristics

1. The rate of production required is about 35 ft^2 / manhr (Young).
2. The formwork crew has 7 crew members (Young).

6. Steel gang form / Jump form

Site characteristics

1. Site storage space area is required to assemble or store largest gang form panel.

Structural characteristics

1. The method is not suitable for walls with pilasters (Hanna, 1998). The offset and corners variations from floor to floor can be can be adjusted by considering their maximum sizes and the use of fillers (Newell).
2. The available reuse of formwork should be 30 to 40 (Backe, 1986).
3. Jump forms can be 8 to 16 ft high and they can be 8 to 44 ft wide (Peurifoy, 1995).
4. The jump form system needs a 5 ft wide operating platform (Peurifoy, 1995).
5. The rate of pour is usually between 6 to 9 ft/hr (Newell).
6. The allowable pour pressure is within the range 1200 psf to 1500 psf (EFCO, 1994).
7. The allowable tie spacing is 6 ft to 8 ft (EFCO, 1994).
8. The tie spacing for column forms is usually 5 ft (Newell).

Production characteristics

1. The rate of production required is within 55 ft^2 / manhr to 70 ft^2 / manhr (Form Marks).
2. The formwork crew has 7 crew members (Young).

7. Tunnel form system

Please refer to Method (5).

8. Self-climbing formwork

Site characteristics

1. No feasibility knowledge was observed regarding site characteristics.

Structural characteristics

1. The method is economically feasible if the building has at least 15 floors (Peurifoy, 1998).
2. There should be at least 30 reuses (Hanna, 1990).
3. Maximum floor-to-floor lift is up to15 ft (Fulton, 1989).
4. The structure needs to be brought up several floors before using a self-climbing formwork system (Fulton, 1989).
5. No feasibility knowledge was observed regarding pour characteristics.

Production characteristics

1. The formwork crew has 10 to12 crew members (Hanna, 1998).
2. The location and capacity of cranes must be considered because it affects installation and removal of the self-climbing system (Fulton, 1989).

9. Slip form system (Camellerie, 1978)

Site characteristics

1. No feasibility knowledge was observed regarding site characteristics.

Structural characteristics

1. If the core wall has offsets, inserts, corbels then these members are placed later.
2. The available repetition of an element should provide 50 to 100 reuses i.e., the core should be 200 to 400 ft high (Hanna, 1990).
3. The ideal slipform should require at least 20 cubic yards of concrete per foot of height or per hour.
4. The quality of surface finish obtained is without horizontal construction joints and without tie holes.
5. The slump of concrete required is 4 inches plus or minus 1 inch.

Production characteristics

1. The average rate of production is 8 to 12 inches per hour.
2. The production rate is dependent upon initial setting time of concrete, which in turn is dictated by the amount, type and grind of cement, concrete temperature, and admixtures.

II. Rebar Placement methods

1. Column Rebar Assembly
2. Column Rebar Prefabrication
3. Wall / Core Rebar Assembly
4. Wall / Core Partial Rebar Prefabrication
5. Wall / Core Rebar Prefabrication
6. Slab Rebar Assembly
7. Slab Rebar Prefabrication

1. Column Rebar Assembly (Fradley) (Stefanich) (Shaw) (Bichel)

Site characteristics

1. Rebar is delivered to site in the order of assembly, which is dictated by the construction schedule. The rebar storage space should be enough to unload a delivery truck i.e., it should be at least 12 x 60 ft (Bichel).

Structural characteristics

1. The column rebar is assembled in place when there are architectural details e.g., changing shape (Bichel).
2. The column rebar is assembled in place when there are multiple embedded metal plates for structural members. The prefabrication becomes time consuming due to the details of nails and studs of the metal plates to be embedded (Bichel).
3. Columns of greater heights are preferably prefabricated because the in-place rebar assembly needs scaffolds; moreover beyond 10 ft height one needs safety belts (WCB, n.d.). These factors contribute to a lower rate of production.

Production characteristics

1. The rate of production is approximately 0.071 ton / manhr (14 manhr/ ton) (Shaw).
2. The rebar crew has 8 crew members (Stefanich).

2. Column Rebar Prefabrication (Fradley) (Stefanich) (Shaw) (Bichel)

Site characteristics

1. In case of just in time delivery of prefabricated column rebar, site storage space is not needed (Bichel). Site rebar storage space length should be at least equal to the

sum of maximum height of column with laps and 4 ft space i.e., (Max.height of column + 4ft + laps).

2. If prefabrication is "Onsite", then site rebar assembly space length should be at least three times the sum of maximum height of column with laps and 4 ft space i.e., (Max.height of column + 4ft + laps) x 3. The linear layout of assembly space generally has rebar storage area, prefabrication area with a jig for rebar assembly, and stacking area for prefabricated elements (Fradley), (Bichel).

3. If prefabrication is "Onsite", then site rebar assembly space area should be at least 1200 ft^2 for a typical high-rise construction project (Bichel) i.e., 20 ft x 40 to 60 ft (Fradley).

Structural characteristics

1. The height of large and heavy prefabricated column can be up to 30 ft (Stefanich) (Shaw). Special lifting devices and guy wires are required for rebar cage installation.
2. The weight large and heavy prefabricated column cage can be up to 2 tons (Shaw). Limited by the crane lifting capacity, at the tip of the boom, available on site (Bichel).

Production characteristics

1. The rate of production is 0.125 ton / manhr (8 manhr / ton) (Shaw).
2. The rebar crew has 8 crew members (Stefanich).

3. Wall Rebar Assembly (Fradley) (Stefanich) (Shaw)

Site characteristics

1. Site storage space for rebar should be at least 12 x 60 ft (Bichel).

Structural characteristics

1. Walls with a number of openings and larger openings such as doors are assembled in place (Shaw).

Production characteristics

1. The rate of production is 0.1 ton / manhr (10 manhr/ ton) (Shaw).
2. The rebar crew has 8 crew members (Stefanich).

4. Wall / Core wall Rebar Partial Prefabrication

Site characteristics

1. Enough site rebar storage space should be available to store zones[3], which are generally 24 ft in length (Bichel).

Structural characteristics

1. The wall has shear zones that are prefabricated. The wall portion in between the zones is assembled in place (Bichel).

Production characteristics

1. The rate of production is 0.1 ton / manhr (10 manhr/ ton) (Shaw).
2. The rebar crew has 8 crew members (Stefanich).

5. Wall / Core wall Rebar Prefabrication

Site characteristics

1. If the prefabrication is "Onsite", then site rebar assembly space length should be at least three times the sum of maximum length of wall with laps and 4 ft space i.e., (Max.length of wall + 4ft + laps) x 3.

2. If the prefabrication is "Onsite", then site rebar assembly space area should be at least 1200 ft^2 i.e., 20 ft x 40 to 60 ft (Fradley).

Structural characteristics

1. The wall should not have more than 2 openings in a 30 ft length (Shaw). More openings makes it difficult to prefabricate.
2. The opening area should not be more than 1 m^2 i.e., 10.7639 ft^2 (Shaw). Openings need additional steel around them.

Production characteristics

1. The rate of production is 0.11 ton / manhr (9 manhr / ton) (Bichel).
2. The rate of production for a core wall is 0.1 ton / manhr (10 manhr / ton) (Shaw).
3. If the wall has shear "zones", total prefabrication will need more lap length and more rebar tonnage.

---

[3] According to CSA Standard A23.3-94, in a seismic zone a structural frame of greater ductility is required. As an elasto-plastic system, such a frame is designed to accommodate the formation of plastc hinges. This seismic design creates regions of concentrated reinforcement in the shear elements (Fradley). These regions, called as "zones", are generally prefabricated.

6. Slab Reinforcement Assembly

Site characteristics

1. The rebar storage space should be enough to unload a delivery truck i.e., it should be at least 12 x 60 ft (Bichel).

Structural characteristics

1. If the vertical i.e., shear elements have "zones" this method is preferred.

Production characteristics

1. The rate of production is 0.166 ton / manhr (6 manhr / ton) (Stefanich).
2. The rebar crew has 8 crew members (Stefanich).

7. Slab Rebar Prefabrication

Site characteristics

1. If prefabrication is "Onsite", then site rebar assembly space length should be at least three times the sum of maximum length of slab section with laps and 4 ft space i.e., (Max. length of slab section + 4ft + laps) x 3.

2. If prefabrication is "Onsite", then site rebar assembly space width should be at least maximum width of slab section with laps and 4 ft space i.e., (Max.width of slab section + 4ft + laps).

3. If prefabrication is "Onsite", then site rebar assembly space area should be at least 1200 ft^2 i.e., 20 ft x 40 to 60 ft (Fradley).

Structural characteristics

1. The bottom rebar of the slab is seldom prefabricated; on the other hand top rebar is prefabricated depending upon the areas of typical top mats (Bichel).
2. There should be at least 20 identical sections of slab for this method to be feasible (Bennett, 1992).
3. There are various proprietary punching shear reinforcements available for flat slab rebar placement (BPG, 2001). Moreover, slab rebar can be prefabricated in the form of rebar mats, which makes rebar placement easier and faster (BAMTEC, n.d.).

Production characteristics

1. Proprietary prefabricated slab rebar mats can be placed at 4.5 ton / manhr (BAMTEC, n.d.).
2. The rebar placement crew can be 2 crew members (BAMTEC, n.d.).
3. Prefabricated mats and proprietary punching shear reinforcement can save up to 50 % in man-hours (BRE).
4. The method is compatible with tunnel forming method for a faster construction cycle (deBruin).

III. Concrete placement techniques

1. Crane and bucket method
2. Belt conveyor method
3. Placing boom pumping
4. Slickline pumping
5. Separate placing boom pumping

1. Crane and bucket method (Gastaldo)

Site characteristics

1. Site Concrete equipment parking space length should be at least 30 ft + 8 ft i.e., space for ready-mix truck and bucket loading.
2. Site Concrete equipment parking space width should be at least 15 ft.
3. Site Concrete equipment parking space area should be at least 570 ft^2. The site space should be sufficient for the concrete truck mixer parking and concrete bucket loading. One ready-mix truck needs at least 15 x 30 feet space (Wallace, 1998).

Structural characteristics

1. The Concrete equipment parking space should not have any obstruction due to overhead electrical wires.
2. The method can handle low slump concrete.
3. Concrete of maximum aggregate size up to 4 inches can be placed with this method (Slagle, 1997).
4. The method becomes feasible when small quantities of different strength concrete need to be placed almost simultaneously (CC, 1982).

Production characteristics

1. The rate of concrete placement is 45 to 50 yd^3 / hr.
2. The concrete placement crew has 8 crew members.

2. Belt Conveyor method

Site characteristics

1. Site concrete equipment parking space length should be at least 45 ft + 30 ft i.e., space enough for a truck mounted belt conveyor and ready-mix trucks.
2. Site concrete equipment parking space width should be at east 30 ft i.e., maximum outrigger spread of the truck mounted belt conveyors (Putzmeister, 2001a).
3. Site concrete equipment parking space area should be at least 2250 ft^2 i.e.,
4. The Truck mounted belt conveyors have outriggers that need to be set on firm and leveled ground. Rough, sandy, or sloped terrain as well as tight quarters at the site can rule out economical use of portable conveyors (Sagle, 1997).

Structural characteristics

1. The maximum vertical reach required for concrete placement can be up to 87 ft.
2. The maximum horizontal reach required for concrete placement can be up to 150 ft.
3. The maximum vertical downward reach required for concrete placement can be up to 40 ft.
4. The maximum size of the concrete aggregate can be is 4 inches (Putzmeister, 2001a).
5. The range of required slump is within 1 to 7 inches (CC, 1992).
6. The best slump range is between 2 and 4 inches (CC, 1992).
   Continuous concrete placing ability with higher rate of concrete placement makes the conveyor method cost effective (Slagle, 1997).

Production characteristics

1. The concrete placement crew has 8 crew members (Gastaldo).
2. The rate of concrete placement should be within 50 to 360 yd^3 / hr.

3. Placing boom concrete placement (Gastaldo)

Site characteristics

1. Site concrete equipment parking space length should be at least 51ft + 30ft.
2. Site concrete equipment parking space width should be at least 36 ft i.e., maximum outrigger spread of the truck mounted placing booms (Putzmeister, 2001b).
3. Site concrete equipment parking space area should be at least 2916 ft^2.
4. Job site with area 50 x 100 i.e., 5000 ft^2 is a comfortable jobsite (Wallace, 1998).
5. The unfolding height required is up to 52 ft (Putzmeister, 2001b).

6. The concrete equipment parking space should be stable, flat, and clear of rubble (Wallace, 1998). The vehicle parking spot should be away from any excavations, power lines, and other obstructions (Fisher, 1997).

Structural characteristics

1. The concrete equipment parking space should not have any obstruction due to overhead electrical wires i.e. the boom should have at least 17 ft clearance at anytime (Fisher, 1997).
2. The maximum vertical reach required for concrete placement can be up to 188 ft.
3. The maximum horizontal reach required for concrete placement can be up to 174 ft.
4. The maximum vertical downward reach required for concrete placement can be up to 137 ft.
5. The maximum size of the concrete aggregate should be 2.5 inches (Putzmeister, 2001b).
6. The range of required slump is within 2 to 9 inches (Gastaldo).

Production characteristics

1. The concrete placement crew is of size 8 crew members (Gastaldo).
2. The rate of concrete placement should be within 50 to 210 yd^3 / hr.
3. The method is feasible when concrete volume to be placed for horizontal elements is equal to 80 m^3 i.e., 104 yd^3.
4. The economical rate of concrete placement is 65 m^3 / hr i.e., 85 yd^3 / hr.
5. The method is feasible when concrete volume to be placed for vertical elements is at least 40- 50 m^3 i.e., 52 to 65 yd^3.
6. The "slump loss" in pumping can be up to 4 inches (Gastaldo), (Crepas, 1985).
7. The method should not be used when wind speed is more than 70 km / hr (Gastaldo).

4. Slickline pumping method (Gastaldo) (Crepas, 1985)

Site characteristics

1. Site concrete equipment parking space length should be at least 25 ft + 30 ft
2. Site concrete equipment parking space width should be at least 30 ft. There should be room for two ready-mix trucks at the pump hopper (Crepas, 1985).
3. Site concrete equipment parking space area should be at least 1650 ft^2.
4. Job site with area 50 x 100 i.e., 5000 ft^2 is a comfortable jobsite (Wallace, 1998).
5. The slickline pumping for high rises needs a 150 feet long "base line" to run on ground before vertical concrete pipeline (Crepas, 1985). Therefore the open space around building width + (Building width / 2) should be at least 150 ft.

6. Site space for thrust block and hydraulic diversion block should be at least 45ft x 30ft (Gastaldo).

Structural characteristics

1. The maximum vertical reach for concrete placement can be up to 400 to 600 ft (Putzmeister).
2. The maximum horizontal reach for concrete placement can be up to1000 to 1200 ft (Putzmeister).
3. The maximum size of the aggregate can be up to 2.5 inches (Putzmeister, 2001c).
4. The range of required slump should be within 2 to 9 inches (Gastaldo).

Production characteristics

1. The concrete placement crew has 8 crew members (Gastaldo).
2. The rate of concrete placement should be within 50 to 210 yd^3 / hr.
3. The breakeven point for concrete pumping is assumed to be 50 m^3 i.e., 65 yd^3 (Lewis, 1999).
4. The slickline pumping method is used for "bottom-up" pumping for vertical elements.
5. The "slump loss" in pumping can be up to 4 inches (Gastaldo) (Crepas, 1985).

5. Separate Concrete placement boom

Site characteristics

1. Site concrete equipment parking space length should be at least 25 ft + 30 ft.
2. Site concrete equipment parking space width should be at least 30 ft. There should be room for two ready-mix trucks at the pump hopper (Crepas, 1985).
3. Site concrete equipment parking space area should be at least 1650 ft^2.
4. Job site with area 50 x 100 i.e., 5000 ft^2 is a comfortable jobsite (Wallace, 1998).
5. The slickline pumping for high rises need 150 feet long "base line" to run on ground before vertical concrete pipeline (Crepas, 1985). Therefore the open space around building width + (Building width / 2) should be at least 150 ft.
6. Site space for thrust block and hydraulic diversion block should be at least 45 ft x 30 ft (Gastaldo).

Structural characteristics

1. The maximum vertical reach required for concrete placement can be 400 to 600 ft (Putzmeister, 2001c).
2. The maximum horizontal reach required for concrete placement (boom) should be within 79 to 111 ft (Putzmeister, 2001d).
3. The pedestal for placing boom requires "block hole" for separate boom mast of size 3 ft x 3 ft (Harvell, 1991).

4. The block hole location should be such that it covers all concrete placement area within the boom's horizontal reach.
5. The block hole location should be such that it covers concrete placement area for all the floors (Harvell, 1991).
6. The maximum size of the concrete aggregate can be up to 2.5 inches (Putzmeister, 2001d).
7. The range of required slump should be within 2 to 9 inches (Gastaldo).

Production characteristics

1. The concrete placement crew is of size 8 crew members (Gastaldo).
2. The rate of concrete placement should be within 50 to 210 yd^3 / hr.
3. The method is feasible when concrete volume for horizontal elements is at least 80 m^3 i.e., 104 yd^3.
4. The economical rate of concrete placement is 65 m^3 / hr i.e., 85 yd^3 / hr.
5. The method is feasible when concrete volume for vertical elements is at least 40-50 m^3 i.e., 52 to 65 yd^3.
6. The "slump loss" in pumping can be up to 4 inches (Gastaldo), (Crepas, 1985).
7. Concrete placement should be at least 3 times per week with minimum size of concrete placement being 60 to 100 m^3 (Gastaldo) i.e., 78 to 130 yd^3. The concrete quantity should be at least 235 yd^3 per floor assuming construction of one week with at least three concrete placements.
8. The method is suitable when the slab is post tensioned (Crepas, 1985).
9. For safety reasons the placer booms should not be operated if the wind speed exceeds 77 km / hr (ACPA, 2001).

APPENDIX – B

Examples of PCBS and M&RBS Facts Exported to the CLIPS Environment

(Excerpted from file "TESTda.fct")

Appendix B Examples of PCBS and M&RBS Facts Exported to the CLIPS Environment

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;; PCBS relationship facts ;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(parent (parent_component "1") (child_component "2"))
(parent (parent_component "1") (child_component "3"))
(parent (parent_component "3") (child_component "4"))
(parent (parent_component "4") (child_component "5"))
(parent (parent_component "4") (child_component "6"))
(parent (parent_component "4") (child_component "7"))
(parent (parent_component "4") (child_component "8"))
(parent (parent_component "4") (child_component "9"))
(parent (parent_component "4") (child_component "10"))
(parent (parent_component "4") (child_component "11"))
(parent (parent_component "4") (child_component "12"))
(parent (parent_component "4") (child_component "13"))
(parent (parent_component "4") (child_component "14"))
(parent (parent_component "4") (child_component "15"))
(parent (parent_component "4") (child_component "16"))
(parent (parent_component "4") (child_component "17"))
(parent (parent_component "4") (child_component "18"))
(parent (parent_component "4") (child_component "19"))
(parent (parent_component "4") (child_component "20"))
(parent (parent_component "4") (child_component "21"))
(parent (parent_component "4") (child_component "22"))
(parent (parent_component "4") (child_component "23"))
(parent (parent_component "4") (child_component "24"))
(parent (parent_component "4") (child_component "25"))
(parent (parent_component "4") (child_component "26"))
(parent (parent_component "4") (child_component "27"))
(parent (parent_component "4") (child_component "28"))
(parent (parent_component "4") (child_component "29"))
(parent (parent_component "4") (child_component "30"))
(parent (parent_component "4") (child_component "31"))
(parent (parent_component "4") (child_component "32"))
(parent (parent_component "4") (child_component "33"))
(parent (parent_component "4") (child_component "34"))
(parent (parent_component "3") (child_component "35"))
(parent (parent_component "35") (child_component "36"))
(parent (parent_component "36") (child_component "37"))
(parent (parent_component "37") (child_component "38"))
```

```
(parent (parent_component "37") (child_component "39"))
(parent (parent_component "37") (child_component "40"))
(parent (parent_component "37") (child_component "41"))
(parent (parent_component "37") (child_component "42"))
(parent (parent_component "37") (child_component "43"))
(parent (parent_component "37") (child_component "44"))
(parent (parent_component "37") (child_component "45"))
(parent (parent_component "37") (child_component "46"))
(parent (parent_component "37") (child_component "47"))
(parent (parent_component "36") (child_component "48"))
(parent (parent_component "48") (child_component "49"))
(parent (parent_component "48") (child_component "50"))
(parent (parent_component "50") (child_component "51"))
(parent (parent_component "50") (child_component "52"))
(parent (parent_component "48") (child_component "53"))
(parent (parent_component "53") (child_component "54"))
(parent (parent_component "53") (child_component "55"))
(parent (parent_component "53") (child_component "56"))
(parent (parent_component "48") (child_component "57"))
(parent (parent_component "48") (child_component "58"))
(parent (parent_component "48") (child_component "59"))
(parent (parent_component "48") (child_component "60"))
(parent (parent_component "60") (child_component "61"))
(parent (parent_component "60") (child_component "62"))
(parent (parent_component "36") (child_component "63"))
(parent (parent_component "63") (child_component "64"))
(parent (parent_component "63") (child_component "65"))
(parent (parent_component "63") (child_component "66"))
(parent (parent_component "63") (child_component "67"))
(parent (parent_component "63") (child_component "68"))
(parent (parent_component "68") (child_component "69"))
(parent (parent_component "63") (child_component "70"))
(parent (parent_component "63") (child_component "71"))
(parent (parent_component "63") (child_component "72"))
(parent (parent_component "63") (child_component "73"))
(parent (parent_component "73") (child_component "74"))
(parent (parent_component "63") (child_component "75"))
(parent (parent_component "75") (child_component "76"))
(parent (parent_component "63") (child_component "77"))
(parent (parent_component "63") (child_component "78"))
(parent (parent_component "63") (child_component "79"))
(parent (parent_component "79") (child_component "80"))
(parent (parent_component "79") (child_component "81"))
(parent (parent_component "63") (child_component "82"))
(parent (parent_component "82") (child_component "83"))
(parent (parent_component "63") (child_component "84"))
```

(parent (parent_component "84") (child_component "85"))
(parent (parent_component "63") (child_component "86"))
(parent (parent_component "86") (child_component "87"))
(parent (parent_component "86") (child_component "88"))
(parent (parent_component "63") (child_component "89"))
(parent (parent_component "89") (child_component "90"))
(parent (parent_component "89") (child_component "91"))
(parent (parent_component "63") (child_component "92"))
(parent (parent_component "92") (child_component "93"))
(parent (parent_component "63") (child_component "94"))
(parent (parent_component "63") (child_component "95"))
(parent (parent_component "63") (child_component "96"))
(parent (parent_component "96") (child_component "97"))
(parent (parent_component "96") (child_component "98"))
(parent (parent_component "63") (child_component "99"))
(parent (parent_component "63") (child_component "100"))
(parent (parent_component "100") (child_component "101"))
(parent (parent_component "35") (child_component "102"))
(parent (parent_component "102") (child_component "103"))
(parent (parent_component "103") (child_component "104"))
(parent (parent_component "104") (child_component "105"))
(parent (parent_component "103") (child_component "106"))
(parent (parent_component "106") (child_component "107"))
(parent (parent_component "103") (child_component "108"))
(parent (parent_component "108") (child_component "109"))
(parent (parent_component "103") (child_component "110"))
(parent (parent_component "110") (child_component "111"))
(parent (parent_component "103") (child_component "112"))
(parent (parent_component "112") (child_component "113"))
(parent (parent_component "103") (child_component "114"))
(parent (parent_component "114") (child_component "115"))
(parent (parent_component "103") (child_component "116"))
(parent (parent_component "116") (child_component "117"))
(parent (parent_component "103") (child_component "118"))
(parent (parent_component "118") (child_component "119"))
(parent (parent_component "103") (child_component "120"))
(parent (parent_component "120") (child_component "121"))
(parent (parent_component "103") (child_component "122"))
(parent (parent_component "122") (child_component "123"))
(parent (parent_component "122") (child_component "124"))
(parent (parent_component "103") (child_component "125"))
(parent (parent_component "125") (child_component "126"))
(parent (parent_component "103") (child_component "127"))
(parent (parent_component "127") (child_component "128"))
(parent (parent_component "103") (child_component "129"))
(parent (parent_component "129") (child_component "130"))

(parent (parent_component "103") (child_component "131"))
(parent (parent_component "103") (child_component "132"))
(parent (parent_component "103") (child_component "133"))
(parent (parent_component "103") (child_component "134"))
(parent (parent_component "103") (child_component "135"))
(parent (parent_component "103") (child_component "136"))
(parent (parent_component "3") (child_component "137"))
(parent (parent_component "1") (child_component "138"))

;;
;;;;;; PCBS facts ;;;;;;;;;;;;;
;;

(pcbs_component (name "1") (path "GIA") (code "GIA") (description " Residential High-Rise Project") (component_type "Project") (attributes) (attribute_type) (attribute_values))

(pcbs_component (name "2") (path "GIA.SiteLoc") (code "SiteLoc") (description "Site Location") (component_type "Location Set") (attributes "Length" "Width" "Site Storage Area" "Open Space Length" "Parking Space Length" "Parking Space Width" "Rebar Storage Space Length" "Rebar Storage Space Width" "Rebar Fabrication Space Length" "Rebar Fabrication Space Width" "Horizontal Formwork Storage Space Length" "Horizontal Formwork Storage Space Width" "Vertical Formwork Storage Space Length" "Vertical Formwork Storage Space Width") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values [GIA.SiteLoc/1] [GIA.SiteLoc/2] [GIA.SiteLoc/3] [GIA.SiteLoc/4] [GIA.SiteLoc/5] [GIA.SiteLoc/6] [GIA.SiteLoc/7] [GIA.SiteLoc/8] [GIA.SiteLoc/9] [GIA.SiteLoc/10] [GIA.SiteLoc/11] [GIA.SiteLoc/12] [GIA.SiteLoc/13] [GIA.SiteLoc/14]))

(pcbs_component (name "3") (path "GIA.Tower") (code "Tower") (description "High Rise Tower") (component_type "Subproject") (attributes) (attribute_type) (attribute_values))

(pcbs_component (name "4") (path "GIA.Tower.TLoc") (code "TLoc") (description "High Rise Tower Locations") (component_type "Location Set") (attributes) (attribute_type) (attribute_values))

(pcbs_component (name "35") (path "GIA.Tower.SupSTR") (code "SupSTR") (description "High rise Tower Super Structure") (component_type "System") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic") (attribute_values [GIA.Tower.SupSTR/1] [GIA.Tower.SupSTR/2] [GIA.Tower.SupSTR/3] [GIA.Tower.SupSTR/4]

[GIA.Tower.SupSTR/5] [GIA.Tower.SupSTR/6] [GIA.Tower.SupSTR/7] [GIA.Tower.SupSTR/8]))

(pcbs_component (name "36") (path "GIA.Tower.SupSTR.VertEle") (code "VertEle") (description "Vertical Components") (component_type "Subsystem") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Rate of Pour") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative") (attribute_values [GIA.Tower.SupSTR.VertEle/1] [GIA.Tower.SupSTR.VertEle/2] [GIA.Tower.SupSTR.VertEle/3] [GIA.Tower.SupSTR.VertEle/4] [GIA.Tower.SupSTR.VertEle/5] [GIA.Tower.SupSTR.VertEle/6] [GIA.Tower.SupSTR.VertEle/7] [GIA.Tower.SupSTR.VertEle/8] [GIA.Tower.SupSTR.VertEle/9] [GIA.Tower.SupSTR.VertEle/10] [GIA.Tower.SupSTR.VertEle/11]))

;;
;;;;;;;;; PCBS columns ;;;;;;;;;;
;;

(pcbs_component (name "37") (path "GIA.Tower.SupSTR.VertEle.Cols") (code "Cols") (description "Columns") (component_type "Element") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Width" "Height" "Number of Elements" "Max. Height")(attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values [GIA.Tower.SupSTR.VertEle.Cols/1] [GIA.Tower.SupSTR.VertEle.Cols/2] [GIA.Tower.SupSTR.VertEle.Cols/3] [GIA.Tower.SupSTR.VertEle.Cols/4] [GIA.Tower.SupSTR.VertEle.Cols/5] [GIA.Tower.SupSTR.VertEle.Cols/6] [GIA.Tower.SupSTR.VertEle.Cols/7] [GIA.Tower.SupSTR.VertEle.Cols/8] [GIA.Tower.SupSTR.VertEle.Cols/9] [GIA.Tower.SupSTR.VertEle.Cols/10] [GIA.Tower.SupSTR.VertEle.Cols/11] [GIA.Tower.SupSTR.VertEle.Cols/12] [GIA.Tower.SupSTR.VertEle.Cols/13] [GIA.Tower.SupSTR.VertEle.Cols/14] [GIA.Tower.SupSTR.VertEle.Cols/15] [GIA.Tower.SupSTR.VertEle.Cols/16]))

(pcbs_component (name "38") (path "GIA.Tower.SupSTR.VertEle.Cols.Colm1") (code "Colm1") (description "Column A") (component_type "Subelement") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Width" "Height" "Number of Elements" "Max. Height") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic"

"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/1]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/2]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/3]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/4]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm1/16]))

(pcbs_component (name "39") (path "GIA.Tower.SupSTR.VertEle.Cols.Colm2") (code "Colm2") (description "Column B") (component_type "Subelement") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Width" "Height" "Number of Elements" "Max. Height") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/1]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/2]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/3]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/4]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm2/16]))

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;  PCBS core   ;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(pcbs_component (name "48") (path "GIA.Tower.SupSTR.VertEle.Core") (code "Core")
(description "High Rise Tower Core") (component_type "Element") (attributes
"Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time
Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape"
"Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Height" "Width"
"Number of Elements") (attribute_type "Quantitative" "Quantitative" "Quantitative"
"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative"
"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative"
"Quantitative") (attribute_values [GIA.Tower.SupSTR.VertEle.Core/1]
[GIA.Tower.SupSTR.VertEle.Core/2] [GIA.Tower.SupSTR.VertEle.Core/3]
[GIA.Tower.SupSTR.VertEle.Core/4] [GIA.Tower.SupSTR.VertEle.Core/5]
[GIA.Tower.SupSTR.VertEle.Core/6] [GIA.Tower.SupSTR.VertEle.Core/7]
[GIA.Tower.SupSTR.VertEle.Core/8] [GIA.Tower.SupSTR.VertEle.Core/9]
[GIA.Tower.SupSTR.VertEle.Core/10] [GIA.Tower.SupSTR.VertEle.Core/11]
[GIA.Tower.SupSTR.VertEle.Core/12] [GIA.Tower.SupSTR.VertEle.Core/13]
[GIA.Tower.SupSTR.VertEle.Core/14] [GIA.Tower.SupSTR.VertEle.Core/15]))

(pcbs_component (name "49") (path "GIA.Tower.SupSTR.VertEle.Core.Szone") (code
"Szone") (description "Shear zones of rebar") (component_type "Content") (attributes)
(attribute_type) (attribute_values))

(pcbs_component (name "50") (path "GIA.Tower.SupSTR.VertEle.Core.CWall1") (code
"CWall1") (description "Core Wall A") (component_type "Subelement") (attributes
"Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time
Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape"
"Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Height" "Width"
"Number of Elements") (attribute_type "Quantitative" "Quantitative" "Quantitative"
"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative"
"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative"
"Quantitative") (attribute_values [GIA.Tower.SupSTR.VertEle.Core.CWall1/1]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/2]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/3]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/4]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/5]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/6]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/7]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/8]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/9]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/10]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/11]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/12]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/13]
```

[GIA.Tower.SupSTR.VertEle.Core.CWall1/14]
[GIA.Tower.SupSTR.VertEle.Core.CWall1/15]))

(pcbs_component (name "51") (path "GIA.Tower.SupSTR.VertEle.Core.CWall1.Cornr1") (code "Cornr1") (description "Corner") (component_type "SubSubelement") (attributes "Length" "Width") (attribute_type "Quantitative" "Quantitative") (attribute_values [GIA.Tower.SupSTR.VertEle.Core.CWall1.Cornr1/1]
[GIA.Tower.SupSTR.VertEle.Core.CWall1.Cornr1/2]))

(pcbs_component (name "52") (path "GIA.Tower.SupSTR.VertEle.Core.CWall1.Opengl") (code "Opengl") (description "Openging") (component_type "SubSubelement") (attributes "Length" "Height") (attribute_type "Quantitative" "Quantitative") (attribute_values [GIA.Tower.SupSTR.VertEle.Core.CWall1.Opengl/1]
[GIA.Tower.SupSTR.VertEle.Core.CWall1.Opengl/2]))

(pcbs_component (name "53") (path "GIA.Tower.SupSTR.VertEle.Core.CWall2") (code "CWall2") (description "Core Wall B") (component_type "Subelement") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Height" "Width" "Number of Elements") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values [GIA.Tower.SupSTR.VertEle.Core.CWall2/1]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/2]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/3]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/4]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/5]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/6]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/7]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/8]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/9]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/10]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/11]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/12]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/13]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/14]
[GIA.Tower.SupSTR.VertEle.Core.CWall2/15]))

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;  PCBS walls  ;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(pcbs_component (name "63") (path "GIA.Tower.SupSTR.VertEle.ShWall") (code
"ShWall") (description "Shear Walls") (component_type "Element") (attributes
"Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time
Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape"
"Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Height" "Width"
"Number of Elements") (attribute_type "Quantitative" "Quantitative" "Quantitative"
"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative"
"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative"
"Quantitative") (attribute_values [GIA.Tower.SupSTR.VertEle.ShWall/1]
[GIA.Tower.SupSTR.VertEle.ShWall/2] [GIA.Tower.SupSTR.VertEle.ShWall/3]
[GIA.Tower.SupSTR.VertEle.ShWall/4] [GIA.Tower.SupSTR.VertEle.ShWall/5]
[GIA.Tower.SupSTR.VertEle.ShWall/6] [GIA.Tower.SupSTR.VertEle.ShWall/7]
[GIA.Tower.SupSTR.VertEle.ShWall/8] [GIA.Tower.SupSTR.VertEle.ShWall/9]
[GIA.Tower.SupSTR.VertEle.ShWall/10] [GIA.Tower.SupSTR.VertEle.ShWall/11]
[GIA.Tower.SupSTR.VertEle.ShWall/12] [GIA.Tower.SupSTR.VertEle.ShWall/13]
[GIA.Tower.SupSTR.VertEle.ShWall/14] [GIA.Tower.SupSTR.VertEle.ShWall/15]))

(pcbs_component (name "64") (path "GIA.Tower.SupSTR.VertEle.ShWall.Szone")
(code "Szone") (description "Shear zones of rebar") (component_type  "Content")
(attributes) (attribute_type) (attribute_values))

(pcbs_component (name "65") (path "GIA.Tower.SupSTR.VertEle.ShWall.SWall1")
(code "SWall1") (description "Shear Wall A - non typical walls at GFL")
(component_type "Subelement") (attributes "Formwork Quantity" "Rebar Quantity"
"Concrete Quantity" "Surface Area" "Time Frame for Concreting"  "Time Frame for
Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate"
"Rate of Pour" "Length" "Height" "Width"    "Number of Elements") (attribute_type
"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative"
"Quantitative" "Linguistic"    "Quantitative" "Quantitative" "Quantitative" "Quantitative"
"Quantitative" "Quantitative" "Quantitative") (attribute_values
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/1]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/2]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/3]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/4]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/5]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/6]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/7]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/8]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/9]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/10]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/11]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/12]
```

[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/13]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/14]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall1/15]))

(pcbs_component (name "66") (path "GIA.Tower.SupSTR.VertEle.ShWall.SWall2") (code "SWall2") (description "Shear Wall B - non typical walls at 2nd floor") (component_type "Subelement") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Height" "Width" "Number of Elements") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/1]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/2]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/3]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/4]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/5]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/6]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/7]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/8]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/9]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/10]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/11]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/12]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall2/13]

[GIA.Tower.SupSTR.VertE(pcbs_component (name "67") (path "GIA.Tower.SupSTR.VertEle.ShWall.SWall3") (code "SWall3") (description "Shear Wall M") (component_type "Subelement") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Rate of Pour" "Length" "Height" "Width" "Number of Elements") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/1]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/2]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/3]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/4]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/5]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/6]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/7]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/8]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/9]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/10]

[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/11]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/12]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/13]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/14]
[GIA.Tower.SupSTR.VertEle.ShWall.SWall3/15]))

(pcbs_component (name "102") (path "GIA.Tower.SupSTR.HoriEle") (code "HoriEle") (description "Horizontal Components") (component_type "Subsystem") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Number of Elements") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative") (attribute_values [GIA.Tower.SupSTR.HoriEle/1] [GIA.Tower.SupSTR.HoriEle/2] [GIA.Tower.SupSTR.HoriEle/3] [GIA.Tower.SupSTR.HoriEle/4] [GIA.Tower.SupSTR.HoriEle/5] [GIA.Tower.SupSTR.HoriEle/6] [GIA.Tower.SupSTR.HoriEle/7] [GIA.Tower.SupSTR.HoriEle/8] [GIA.Tower.SupSTR.HoriEle/9] [GIA.Tower.SupSTR.HoriEle/10] [GIA.Tower.SupSTR.HoriEle/11]))

;;;
;;;;;;;;;; PCBS slab ;;;;;;;;;;
;;;

(pcbs_component (name "103") (path "GIA.Tower.SupSTR.HoriEle.Slab") (code "Slab") (description "High Rise Floor Slab") (component_type "Element") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Number of Elements" "Length" "Width" "Thickness" "Horizontal Distance" "Vertical Distance" "Storey Height" "Min. Width" "SlabBay Support is Uniform" "SlabBay Supporting Sides are Parallel") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Boolean" "Boolean") (attribute_values [GIA.Tower.SupSTR.HoriEle.Slab/1] [GIA.Tower.SupSTR.HoriEle.Slab/2] [GIA.Tower.SupSTR.HoriEle.Slab/3] [GIA.Tower.SupSTR.HoriEle.Slab/4] [GIA.Tower.SupSTR.HoriEle.Slab/5] [GIA.Tower.SupSTR.HoriEle.Slab/6] [GIA.Tower.SupSTR.HoriEle.Slab/7] [GIA.Tower.SupSTR.HoriEle.Slab/8] [GIA.Tower.SupSTR.HoriEle.Slab/9] [GIA.Tower.SupSTR.HoriEle.Slab/10] [GIA.Tower.SupSTR.HoriEle.Slab/11] [GIA.Tower.SupSTR.HoriEle.Slab/12] [GIA.Tower.SupSTR.HoriEle.Slab/13] [GIA.Tower.SupSTR.HoriEle.Slab/14] [GIA.Tower.SupSTR.HoriEle.Slab/15] [GIA.Tower.SupSTR.HoriEle.Slab/16] [GIA.Tower.SupSTR.HoriEle.Slab/17] [GIA.Tower.SupSTR.HoriEle.Slab/18] [GIA.Tower.SupSTR.HoriEle.Slab/19] [GIA.Tower.SupSTR.HoriEle.Slab/20]))

(pcbs_component (name "104") (path "GIA.Tower.SupSTR.HoriEle.Slab.SlBay1") (code "SlBay1") (description "SlabBay A") (component_type "Subelement") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Number of Elements" "Length" "Width" "Thickness" "Horizontal Distance" "Vertical Distance" "Storey Height" "Min. Width" "SlabBay Support is Uniform" "SlabBay Supporting Sides are Parallel") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Boolean" "Boolean") (attribute_values [GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/1]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/2]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/3]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/4]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/5]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/6]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/7]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/8]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/9]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/10]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/11]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/12]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/13]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/14]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/15]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/16]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/17]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/18]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/19]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1/20]))

(pcbs_component (name "105") (path "GIA.Tower.SupSTR.HoriEle.Slab.SlBay1.SlBand1") (code "SlBand1") (description "Slabband 1") (component_type "SubSubelement") (attributes "Depth" "Width") (attribute_type "Quantitative" "Quantitative") (attribute_values
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1.SlBand1/1]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay1.SlBand1/2]))

(pcbs_component (name "106") (path "GIA.Tower.SupSTR.HoriEle.Slab.SlBay2") (code "SlBay2") (description "SlabBay B") (component_type "Subelement") (attributes "Formwork Quantity" "Rebar Quantity" "Concrete Quantity" "Surface Area" "Time Frame for Concreting" "Time Frame for Rebar" "Time Frame for Formwork" "Shape" "Slump Range" "Max. Size of Aggregate" "Number of Elements" "Length" "Width" "Thickness" "Horizontal Distance" "Vertical Distance" "Storey Height" "Min. Width" "SlabBay Support is Uniform" "SlabBay Supporting Sides are Parallel") (attribute_type

"Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Linguistic" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Boolean" "Boolean") (attribute_values [GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/1] [GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/2]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/3]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/4]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/5]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/6]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/7]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/8]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/9]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/10]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/11]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/12]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/13]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/14]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/15]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/16]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/17]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/18]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/19]
[GIA.Tower.SupSTR.HoriEle.Slab.SlBay2/20]))

(pcbs_component (name "107") (path "GIA.Tower.SupSTR.HoriEle.Slab.SlBay2.SlBand1") (code "SlBand1") (description "Slabband") (component_type "SubSubelement") (attributes "Width" "Depth") (attribute_type "Quantitative" "Quantitative") (attribute_values [GIA.Tower.SupSTR.HoriEle.Slab.SlBay2.SlBand1/1] [GIA.Tower.SupSTR.HoriEle.Slab.SlBay2.SlBand1/2]))

;;
;;;;;;;;;; M&RBS Realtionship facts ;;;;;;;;;;;
;;

(parent (parent_component "MRBS1") (child_component "MRBS2"))
(parent (parent_component "MRBS2") (child_component "MRBS3"))
(parent (parent_component "MRBS3") (child_component "MRBS4"))
(parent (parent_component "MRBS3") (child_component "MRBS5"))
(parent (parent_component "MRBS1") (child_component "MRBS6"))
(parent (parent_component "MRBS6") (child_component "MRBS7"))
(parent (parent_component "MRBS7") (child_component "MRBS8"))
(parent (parent_component "MRBS1") (child_component "MRBS9"))
(parent (parent_component "MRBS9") (child_component "MRBS10"))
(parent (parent_component "MRBS10") (child_component "MRBS11"))
(parent (parent_component "MRBS10") (child_component "MRBS12"))

```
(parent (parent_component "MRBS10") (child_component "MRBS13"))
(parent (parent_component "MRBS1") (child_component "MRBS14"))
(parent (parent_component "MRBS14") (child_component "MRBS15"))
(parent (parent_component "MRBS15") (child_component "MRBS16"))
(parent (parent_component "MRBS15") (child_component "MRBS17"))
(parent (parent_component "MRBS1") (child_component "MRBS18"))
(parent (parent_component "MRBS18") (child_component "MRBS19"))
(parent (parent_component "MRBS19") (child_component "MRBS20"))
(parent (parent_component "MRBS1") (child_component "MRBS21"))
(parent (parent_component "MRBS21") (child_component "MRBS22"))
(parent (parent_component "MRBS22") (child_component "MRBS23"))
(parent (parent_component "MRBS22") (child_component "MRBS24"))
(parent (parent_component "MRBS22") (child_component "MRBS25"))
(parent (parent_component "MRBS1") (child_component "MRBS26"))
(parent (parent_component "MRBS26") (child_component "MRBS27"))
(parent (parent_component "MRBS27") (child_component "MRBS28"))
(parent (parent_component "MRBS27") (child_component "MRBS29"))
(parent (parent_component "MRBS1") (child_component "MRBS30"))
(parent (parent_component "MRBS30") (child_component "MRBS31"))
(parent (parent_component "MRBS31") (child_component "MRBS32"))
(parent (parent_component "MRBS1") (child_component "MRBS33"))
(parent (parent_component "MRBS33") (child_component "MRBS34"))
(parent (parent_component "MRBS34") (child_component "MRBS35"))
(parent (parent_component "MRBS34") (child_component "MRBS36"))
(parent (parent_component "MRBS34") (child_component "MRBS37"))
(parent (parent_component "MRBS1") (child_component "MRBS38"))
(parent (parent_component "MRBS38") (child_component "MRBS39"))
(parent (parent_component "MRBS39") (child_component "MRBS40"))
(parent (parent_component "MRBS39") (child_component "MRBS41"))
(parent (parent_component "MRBS39") (child_component "MRBS42"))
(parent (parent_component "MRBS1") (child_component "MRBS43"))
(parent (parent_component "MRBS43") (child_component "MRBS44"))
(parent (parent_component "MRBS44") (child_component "MRBS45"))
(parent (parent_component "MRBS1") (child_component "MRBS46"))
(parent (parent_component "MRBS46") (child_component "MRBS47"))
(parent (parent_component "MRBS47") (child_component "MRBS48"))
(parent (parent_component "MRBS47") (child_component "MRBS49"))
(parent (parent_component "MRBS47") (child_component "MRBS50"))
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;; M&RBS Method Statement facts ;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(mrbs_component (name "MRBS1") (path "ROOT") (code "ROOT") (description " High-
rise  Superstructure Construction") (component_type "Method Statement") (attributes)
(parameter_or_condition) (attribute_type) (attribute_values))

(mrbs_component (name "MRBS2") (path "ROOT.FormCol") (code "FormCol")
(description "Formwork for  Columns") (component_type "Operation") (attributes)
(parameter_or_condition) (attribute_type) (attribute_values))

(mrbs_component (name "MRBS3") (path "ROOT.FormCol.WGang") (code "WGang")
(description "Wooden Gang Formwork") (component_type "Method") (attributes "Rate
of Production" "Min. Reuse Required" "Storage Space Length Required" "Storage Space
Width Required" "Allowable Rate of Pour" "Allowable Tie Spacing")
(parameter_or_condition "Parameter" "Condition" "Condition" "Condition" "Condition"
"Condition") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative"
"Quantitative" "Quantitative") (attribute_values [ROOT.FormCol.WGang/1]
[ROOT.FormCol.WGang/2] [ROOT.FormCol.WGang/3] [ROOT.FormCol.WGang/4]
[ROOT.FormCol.WGang/5] [ROOT.FormCol.WGang/6]))

(mrbs_component (name "MRBS4") (path "ROOT.FormCol.WGang.WGC") (code
"WGC") (description "Wooden  Gangform for Column") (component_type "Resource")
(attributes "Rate of Production" "Min. Reuse Required" "Storage Space Length
Required" "Storage Space Width Required" "Allowable Rate of Pour" "Allowable Tie
Spacing") (parameter_or_condition "Parameter" "Condition" "Condition" "Condition"
"Condition" "Condition") (attribute_type "Quantitative" "Quantitative" "Quantitative"
"Quantitative" "Quantitative" "Quantitative") (attribute_values
[ROOT.FormCol.WGang.WGC/1] [ROOT.FormCol.WGang.WGC/2]
[ROOT.FormCol.WGang.WGC/3] [ROOT.FormCol.WGang.WGC/4]
[ROOT.FormCol.WGang.WGC/5] [ROOT.FormCol.WGang.WGC/6]))

(mrbs_component (name "MRBS5") (path "ROOT.FormCol.WGang.FCrew") (code
"FCrew") (description "Formwork Crew") (component_type "Resource") (attributes
"Number of Crew Members") (parameter_or_condition "Parameter") (attribute_type
"Quantitative") (attribute_values [ROOT.FormCol.WGang.FCrew/1]))

(mrbs_component (name "MRBS6") (path "ROOT.RebarCol") (code "RebarCol")
(description "Construction of typical floor of a High-rise") (component_type
"Operation") (attributes) (parameter_or_condition) (attribute_type) (attribute_values))

(mrbs_component (name "MRBS7") (path "ROOT.RebarCol.PreFab") (code "PreFab")
(description "Rebar Prefabrication") (component_type "Method") (attributes "Rate of
Production" "Rebar Site Storage Length Required" "Rebar Fabrication Site Length
Required" "Rebar Fabrication Site Width Required" "Rebar Fabrication Site Area
```

Required") (parameter_or_condition "Parameter" "Condition" "Condition" "Condition" "Condition") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values [ROOT.RebarCol.PreFab/1] [ROOT.RebarCol.PreFab/2] [ROOT.RebarCol.PreFab/3] [ROOT.RebarCol.PreFab/4] [ROOT.RebarCol.PreFab/5]))

(mrbs_component (name "MRBS8") (path "ROOT.RebarCol.PreFab.RCrew") (code "RCrew") (description "Rebar Crew") (component_type "Resource") (attributes "Number of Crew Members") (parameter_or_condition "Parameter") (attribute_type "Quantitative") (attribute_values [ROOT.RebarCol.PreFab.RCrew/1]))

(mrbs_component (name "MRBS9") (path "ROOT.ConcCol") (code "ConcCol") (description "Concrete placing for Columns") (component_type "Operation") (attributes) (parameter_or_condition) (attribute_type) (attribute_values))

(mrbs_component (name "MRBS10") (path "ROOT.ConcCol.CrBuck") (code "CrBuck") (description "Concrete Placing with Crane & Bucket") (component_type "Method") (attributes "Rate of Concrete Placement" "Parking Space Length Required" "Parking Space Width Required" "Parking Space Area Required" "Max. Size of Aggregate") (parameter_or_condition "Parameter" "Condition" "Condition" "Condition" "Condition") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values [ROOT.ConcCol.CrBuck/1] [ROOT.ConcCol.CrBuck/2] [ROOT.ConcCol.CrBuck/3] [ROOT.ConcCol.CrBuck/4] [ROOT.ConcCol.CrBuck/5]))

(mrbs_component (name "MRBS11") (path "ROOT.ConcCol.CrBuck.Crane") (code "Crane") (description "Tower Crane Peiner Hammerhead Tower Crane") (component_type "Resource") (attributes "Rate of Concrete Placement" "Max. Hook height" "Horizontal hook speed" "Vertical Speed" "Boom length" "Max.Weight" "Max.Weight at boom tip" "Parking Space Length Required" "Parking Space Width Required" "Parking Space Area Required" "Max. Size of Aggregate") (parameter_or_condition "Parameter" "Parameter" "Parameter" "Parameter" "Parameter" "Parameter" "Parameter" "Condition" "Condition" "Condition" "Condition") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values [ROOT.ConcCol.CrBuck.Crane/1] [ROOT.ConcCol.CrBuck.Crane/2] [ROOT.ConcCol.CrBuck.Crane/3] [ROOT.ConcCol.CrBuck.Crane/4] [ROOT.ConcCol.CrBuck.Crane/5] [ROOT.ConcCol.CrBuck.Crane/6] [ROOT.ConcCol.CrBuck.Crane/7] [ROOT.ConcCol.CrBuck.Crane/8] [ROOT.ConcCol.CrBuck.Crane/9] [ROOT.ConcCol.CrBuck.Crane/10] [ROOT.ConcCol.CrBuck.Crane/11]))

(mrbs_component (name "MRBS12") (path "ROOT.ConcCol.CrBuck.Bucket") (code "Bucket") (description "Concrete Bucket - Upright") (component_type "Resource") (attributes "Concrete Capacity" "Loading Height" "Outside diameter" "Inside diameter" "Weight") (parameter_or_condition "Parameter" "Parameter" "Parameter" "Parameter"

"Parameter") (attribute_type "Quantitative" "Quantitative" "Quantitative" "Quantitative" "Quantitative") (attribute_values [ROOT.ConcCol.CrBuck.Bucket/1] [ROOT.ConcCol.CrBuck.Bucket/2] [ROOT.ConcCol.CrBuck.Bucket/3] [ROOT.ConcCol.CrBuck.Bucket/4] [ROOT.ConcCol.CrBuck.Bucket/5]))

(mrbs_component (name "MRBS13") (path "ROOT.ConcCol.CrBuck.CCrew") (code "CCrew") (description "Crane and Bucket concrete placement crew") (component_type "Resource") (attributes "Number of crew members") (parameter_or_condition "Parameter") (attribute_type "Quantitative") (attribute_values [ROOT.ConcCol.CrBuck.CCrew/1]))

APPENDIX – C

Examples of PCBS and M&RBS Instances Exported to the CLIPS Environment

(Excerpted from file "TESTda.ist")

Appendix C Examples of PCBS and M&RBS Instances Exported to the CLIPS Environment

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; PCBS "Site Location" instances ;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

([GIA.SiteLoc/1] of PCBS_DATA
  (unit "ft")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/1/SITE]))

([GIA.SiteLoc/1/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 199.25)
  (value2 nil))

([GIA.SiteLoc/2] of PCBS_DATA
  (unit "ft")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/2/SITE]))

([GIA.SiteLoc/2/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 120.75)
  (value2 nil))

([GIA.SiteLoc/3] of PCBS_DATA
  (unit "ft2")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/3/SITE]))

([GIA.SiteLoc/3/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 4214.0)
  (value2 nil))

([GIA.SiteLoc/4] of PCBS_DATA
  (unit "ft")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/4/SITE]))

([GIA.SiteLoc/4/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 117.5)
```

```
 (value2 nil))

([GIA.SiteLoc/5] of PCBS_DATA
 (unit "ft")
 (location_list "SITE")
 (attribute_value_list [GIA.SiteLoc/5/SITE]))

([GIA.SiteLoc/5/SITE] of PCBS_VALUE
 (condition "EQ")
 (value1 96.0)
 (value2 nil))

([GIA.SiteLoc/6] of PCBS_DATA
 (unit "ft")
 (location_list "SITE")
 (attribute_value_list [GIA.SiteLoc/6/SITE]))

([GIA.SiteLoc/6/SITE] of PCBS_VALUE
 (condition "EQ")
 (value1 37.0)
 (value2 nil))

([GIA.SiteLoc/7] of PCBS_DATA
 (unit "ft")
 (location_list "SITE")
 (attribute_value_list [GIA.SiteLoc/7/SITE]))

([GIA.SiteLoc/7/SITE] of PCBS_VALUE
 (condition "EQ")
 (value1 64.0)
 (value2 nil))

([GIA.SiteLoc/8] of PCBS_DATA
 (unit "ft")
 (location_list "SITE")
 (attribute_value_list [GIA.SiteLoc/8/SITE]))

([GIA.SiteLoc/8/SITE] of PCBS_VALUE
 (condition "EQ")
 (value1 10.0)
 (value2 nil))

([GIA.SiteLoc/9] of PCBS_DATA
 (unit "ft")
 (location_list "SITE")
 (attribute_value_list [GIA.SiteLoc/9/SITE]))
```

```
([GIA.SiteLoc/9/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 0.0)
  (value2 nil))

([GIA.SiteLoc/10] of PCBS_DATA
  (unit "ft")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/10/SITE]))

([GIA.SiteLoc/10/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 0.0)
  (value2 nil))

([GIA.SiteLoc/11] of PCBS_DATA
  (unit "ft")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/11/SITE]))

([GIA.SiteLoc/11/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 51.0)
  (value2 nil))

([GIA.SiteLoc/12] of PCBS_DATA
  (unit "ft")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/12/SITE]))

([GIA.SiteLoc/12/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 32.0)
  (value2 nil))

([GIA.SiteLoc/13] of PCBS_DATA
  (unit "ft")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/13/SITE]))

([GIA.SiteLoc/13/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 51.0)
  (value2 nil))
```

```
([GIA.SiteLoc/14] of PCBS_DATA
  (unit "ft")
  (location_list "SITE")
  (attribute_value_list [GIA.SiteLoc/14/SITE]))

([GIA.SiteLoc/14/SITE] of PCBS_VALUE
  (condition "EQ")
  (value1 32.0)
  (value2 nil))

([GIA.Tower.TLoc.2/1] of PCBS_DATA
  (unit "mm2")
  (location_list)
  (attribute_value_list))

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; PCBS subelement "Column 4"  instances ;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1] of PCBS_DATA
  (unit "ft2")
  (location_list "GFL" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18"
"19" "20")
  (attribute_value_list [GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/GFL]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/16]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/17]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/18]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/19]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/20]))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/GFL] of PCBS_VALUE
  (condition "EQ")
  (value1 121.0)
  (value2 nil))
```

```
([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/5] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/6] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/7] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/8] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/9] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/10] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/11] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/12] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/13] of PCBS_VALUE
  (condition "EQ")
  (value1 86.94)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/14] of PCBS_VALUE
```

(condition "EQ")
(value1 86.94)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/15] of PCBS_VALUE
(condition "EQ")
(value1 86.94)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/16] of PCBS_VALUE
(condition "EQ")
(value1 86.94)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/17] of PCBS_VALUE
(condition "EQ")
(value1 86.94)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/18] of PCBS_VALUE
(condition "EQ")
(value1 86.94)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/19] of PCBS_VALUE
(condition "EQ")
(value1 86.94)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/1/20] of PCBS_VALUE
(condition "EQ")
(value1 86.94)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2] of PCBS_DATA
(unit "Tn")
(location_list "GFL" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20")
(attribute_value_list [GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/GFL]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/11]

[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/16]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/17]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/18]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/19]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/20]))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/GFL] of PCBS_VALUE
 (condition "EQ")
 (value1 0.41)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/5] of PCBS_VALUE
 (condition "EQ")
 (value1 0.19)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/6] of PCBS_VALUE
 (condition "EQ")
 (value1 0.19)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/7] of PCBS_VALUE
 (condition "EQ")
 (value1 0.19)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/8] of PCBS_VALUE
 (condition "EQ")
 (value1 0.19)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/9] of PCBS_VALUE
 (condition "EQ")
 (value1 0.19)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/10] of PCBS_VALUE
 (condition "EQ")
 (value1 0.19)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/11] of PCBS_VALUE

```
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/12] of PCBS_VALUE
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/13] of PCBS_VALUE
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/14] of PCBS_VALUE
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/15] of PCBS_VALUE
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/16] of PCBS_VALUE
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/17] of PCBS_VALUE
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/18] of PCBS_VALUE
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/19] of PCBS_VALUE
  (condition "EQ")
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/2/20] of PCBS_VALUE
  (condition "EQ")
```

```
  (value1 0.19)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3] of PCBS_DATA
  (unit "yd3")
  (location_list "GFL" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18"
"19" "20")
  (attribute_value_list [GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/GFL]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/16]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/17]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/18]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/19]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/20]))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/GFL] of PCBS_VALUE
  (condition "EQ")
  (value1 2.44)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/5] of PCBS_VALUE
  (condition "EQ")
  (value1 1.11)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/6] of PCBS_VALUE
  (condition "EQ")
  (value1 1.11)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/7] of PCBS_VALUE
  (condition "EQ")
  (value1 1.11)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/8] of PCBS_VALUE
```

```
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/9] of PCBS_VALUE
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/10] of PCBS_VALUE
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/11] of PCBS_VALUE
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/12] of PCBS_VALUE
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/13] of PCBS_VALUE
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/14] of PCBS_VALUE
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/15] of PCBS_VALUE
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/16] of PCBS_VALUE
 (condition "EQ")
 (value1 1.11)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/17] of PCBS_VALUE
 (condition "EQ")
```

```
(value1 1.11)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/18] of PCBS_VALUE
(condition "EQ")
(value1 1.11)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/19] of PCBS_VALUE
(condition "EQ")
(value1 1.11)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/3/20] of PCBS_VALUE
(condition "EQ")
(value1 1.11)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/4] of PCBS_DATA
(unit "ft2")
(location_list)
(attribute_value_list))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/5] of PCBS_DATA
(unit "hr")
(location_list)
(attribute_value_list))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/6] of PCBS_DATA
(unit "hr")
(location_list)
(attribute_value_list))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/7] of PCBS_DATA
(unit "hr")
(location_list)
(attribute_value_list))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8] of PCBS_DATA
(unit "")
(location_list "GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16"
"17" "18" "19" "20")
(attribute_value_list [GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/GFL]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/2]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/3]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/4]
```

[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/16]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/17]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/18]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/19]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/20]))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/GFL] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/2] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/3] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/4] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/5] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/6] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

```
([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/7] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/8] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/9] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/10] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/11] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/12] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/13] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/14] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/15] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))
```

```
([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/16] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/17] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/18] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/19] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/8/20] of PCBS_VALUE
  (condition "EQ")
  (value1 "Rectangular")
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/9] of PCBS_DATA
  (unit "in")
  (location_list)
  (attribute_value_list))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/10] of PCBS_DATA
  (unit "in")
  (location_list)
  (attribute_value_list))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/11] of PCBS_DATA
  (unit "ft")
  (location_list)
  (attribute_value_list))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12] of PCBS_DATA
  (unit "ft")
  (location_list "GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16"
"17" "18" "19" "20")
  (attribute_value_list [GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/GFL]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/2]
```

```
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/3]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/4]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/16]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/17]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/18]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/19]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/20]))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/GFL] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/2] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/3] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/4] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/5] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/6] of PCBS_VALUE
  (condition "EQ")
```

```
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/7] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/8] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/9] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/10] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/11] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/12] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/13] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/14] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
 (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/15] of PCBS_VALUE
 (condition "EQ")
 (value1 4.0)
```

```
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/16] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/17] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/18] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/19] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/12/20] of PCBS_VALUE
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13] of PCBS_DATA
  (unit "ft")
  (location_list "GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16"
"17" "18" "19" "20")
  (attribute_value_list [GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/GFL]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/2]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/3]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/4]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/15]
```

```
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/16]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/17]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/18]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/19]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/20]))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/GFL] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/2] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/3] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/4] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/5] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/6] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/7] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/8] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))
```

```
([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/9] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/10] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/11] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/12] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/13] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/14] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/15] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/16] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/17] of PCBS_VALUE
  (condition "WR")
  (value1 0.83)
  (value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/18] of PCBS_VALUE
```

(condition "WR")
(value1 0.83)
(value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/19] of PCBS_VALUE
(condition "WR")
(value1 0.83)
(value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/13/20] of PCBS_VALUE
(condition "WR")
(value1 0.83)
(value2 1.5))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14] of PCBS_DATA
(unit "ft")
(location_list "GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20")
(attribute_value_list [GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/GFL]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/2]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/3]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/4]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/16]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/17]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/18]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/19]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/20]))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/GFL] of PCBS_VALUE
(condition "EQ")
(value1 11.0)
(value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/2] of PCBS_VALUE
(condition "EQ")

```
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/3] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/4] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/5] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/6] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/7] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/8] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/9] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/10] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/11] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
```

```
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/12] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/13] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/14] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/15] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/16] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/17] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/18] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/19] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/14/20] of PCBS_VALUE
  (condition "EQ")
  (value1 9.0)
  (value2 nil))
```

```
([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15] of PCBS_DATA
  (unit "No.")
  (location_list "GFL" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18"
"19" "20")
  (attribute_value_list [GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/GFL]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/5]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/6]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/7]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/8]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/9]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/10]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/11]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/12]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/13]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/14]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/15]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/16]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/17]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/18]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/19]
[GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/20]))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/GFL] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/5] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/6] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/7] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/8] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
```

```
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/9] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/10] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/11] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/12] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/13] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/14] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/15] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/16] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/17] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))
```

```
([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/18] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/19] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/15/20] of PCBS_VALUE
  (condition "EQ")
  (value1 1.0)
  (value2 nil))

([GIA.Tower.SupSTR.VertEle.Cols.Colm4/16] of PCBS_DATA
  (unit "ft")
  (location_list)
  (attribute_value_list))


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; M&RBS Method Statement instances ;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

([ROOT.FormCol.WGang/1] of MRBS_VALUE
  (unit "sfth")
  (condition "EQ")
  (value1 35.0)
  (value2 nil))

([ROOT.FormCol.WGang/2] of MRBS_VALUE
  (unit "No.")
  (condition "EQ")
  (value1 30.0)
  (value2 nil))

([ROOT.FormCol.WGang/3] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 50.0)
  (value2 nil))

([ROOT.FormCol.WGang/4] of MRBS_VALUE
  (unit "ft")
```

```
  (condition "EQ")
  (value1 30.0)
  (value2 nil))

([ROOT.FormCol.WGang/5] of MRBS_VALUE
  (unit "fthr")
  (condition "EQ")
  (value1 8.0)
  (value2 nil))

([ROOT.FormCol.WGang/6] of MRBS_VALUE
  (unit "ft")
  (condition "WR")
  (value1 2.0)
  (value2 3.0))

([ROOT.FormCol.WGang.WGC/1] of MRBS_VALUE
  (unit "sfth")
  (condition "EQ")
  (value1 35.0)
  (value2 nil))

([ROOT.FormCol.WGang.WGC/2] of MRBS_VALUE
  (unit "No.")
  (condition "EQ")
  (value1 30.0)
  (value2 nil))

([ROOT.FormCol.WGang.WGC/3] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 50.0)
  (value2 nil))

([ROOT.FormCol.WGang.WGC/4] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 30.0)
  (value2 nil))

([ROOT.FormCol.WGang.WGC/5] of MRBS_VALUE
  (unit "fthr")
  (condition "EQ")
  (value1 8.0)
  (value2 nil))
```

```
([ROOT.FormCol.WGang.WGC/6] of MRBS_VALUE
  (unit "ft")
  (condition "WR")
  (value1 2.0)
  (value2 3.0))

([ROOT.FormCol.WGang.FCrew/1] of MRBS_VALUE
  (unit "No.")
  (condition "EQ")
  (value1 7.0)
  (value2 nil))

([ROOT.RebarCol.PreFab/1] of MRBS_VALUE
  (unit "Tnh")
  (condition "EQ")
  (value1 0.125)
  (value2 nil))

([ROOT.RebarCol.PreFab/2] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 60.0)
  (value2 nil))

([ROOT.RebarCol.PreFab/3] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 60.0)
  (value2 nil))

([ROOT.RebarCol.PreFab/4] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 20.0)
  (value2 nil))

([ROOT.RebarCol.PreFab/5] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 1200.0)
  (value2 nil))

([ROOT.RebarCol.PreFab.RCrew/1] of MRBS_VALUE
  (unit "No.")
  (condition "EQ")
  (value1 8.0)
```

```
  (value2 nil))

([ROOT.ConcCol.CrBuck/1] of MRBS_VALUE
  (unit "yd3/hr")
  (condition "EQ")
  (value1 45.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck/2] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 38.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck/3] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 15.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck/4] of MRBS_VALUE
  (unit "ft2")
  (condition "EQ")
  (value1 570.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck/5] of MRBS_VALUE
  (unit "in")
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/1] of MRBS_VALUE
  (unit "yd3/hr")
  (condition "EQ")
  (value1 45.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/2] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 246.75)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/3] of MRBS_VALUE
  (unit nil)
```

```
  (condition "EQ")
  (value1 290.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/4] of MRBS_VALUE
  (unit nil)
  (condition "EQ")
  (value1 90.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/5] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 229.5)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/6] of MRBS_VALUE
  (unit "lb")
  (condition "EQ")
  (value1 17600.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/7] of MRBS_VALUE
  (unit "lb")
  (condition "EQ")
  (value1 6800.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/8] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 38.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/9] of MRBS_VALUE
  (unit "ft")
  (condition "EQ")
  (value1 15.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Crane/10] of MRBS_VALUE
  (unit "ft2")
  (condition "EQ")
  (value1 570.0)
  (value2 nil))
```

```
([ROOT.ConcCol.CrBuck.Crane/11] of MRBS_VALUE
  (unit "in")
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Bucket/1] of MRBS_VALUE
  (unit "yd3")
  (condition "EQ")
  (value1 4.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Bucket/2] of MRBS_VALUE
  (unit "in")
  (condition "EQ")
  (value1 80.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Bucket/3] of MRBS_VALUE
  (unit "in")
  (condition "EQ")
  (value1 72.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Bucket/4] of MRBS_VALUE
  (unit "in")
  (condition "EQ")
  (value1 68.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.Bucket/5] of MRBS_VALUE
  (unit "lb")
  (condition "EQ")
  (value1 570.0)
  (value2 nil))

([ROOT.ConcCol.CrBuck.CCrew/1] of MRBS_VALUE
  (unit "No.")
  (condition "EQ")
  (value1 8.0)
  (value2 nil))
```

APPENDIX – D

Examples of Method Statement Rules Exported to the CLIPS Environment

(Excerpted from Method Statement Rule File)

Appendix D Examples of Method Statement Rules Exported to the CLIPS Environment

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;; Method Statement for High-rise Superstructure Construction ;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule file_open1
 (declare (salience 5000))

 =>
       (open "C:\\Documents and Settings\\All Users\\Desktop\\CLIPS\\Chapter8-rules-MS1\\results1.txt" formworkFile "w"))

(defrule file_open2
 (declare (salience 5000))

 =>
       (open "C:\\Documents and Settings\\All Users\\Desktop\\CLIPS\\Chapter8-rules-MS1\\results2.txt" rebarFile "w"))

(defrule file_open3
 (declare (salience 5000))

 =>
       (open "C:\\Documents and Settings\\All Users\\Desktop\\CLIPS\\Chapter8-rules-MS1\\results3.txt" concreteFile "w"))

(defrule Shree1
 (declare (salience 500))

 =>
       (printout formworkFile " This is the Report generated for Method Statement - High-rise Superstructure Construction" crlf
              crlf  " Formwork Methods for PCBS components"  crlf crlf))
```

```
(defrule Shree2
 (declare (salience 500))

 =>
        (printout rebarFile " This is the Report generated for Method Statement - High-rise Superstructure Construction" crlf
                crlf " Rebar Placement Methods for PCBS components"  crlf crlf))

(defrule Shree3
 (declare (salience 500))

 =>
        (printout concreteFile " This is the Report generated for Method Statement - High-rise Superstructure Construction" crlf
          crlf " Concrete Placement Methods for PCBS components"  crlf crlf))


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;; Formwork for Columns ;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;; This rule checks for site space for Wooden gang formwork
;;; by taking relevant attributes from method and PCBS component "Column"

(defrule site_space_check_wooden_gangform_column

   (pcbs_component (code "SiteLoc")
                (attributes $?ahead "Vertical Formwork Storage Space Length" $?atail)
                (attribute_values $?avhead ?val1 $?avtail))
   (test (eq (length$ $?ahead)(length $?avhead)))
   (test (eq (length$ $?atail)(length $?avtail)))
```

```
 (mrbs_component (name ?ancestor1)(code ?code1)
                (attributes $?ahead1 "Storage Space Length Required"  $?atail1)
                (attribute_values $?avhead1 ?val2 $?avtail1))
 (test (eq (length$ $?ahead1)(length $?avhead1)))
 (test (eq (length$ $?atail1)(length $?avtail1)))

 (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
 (mrbs_component (name ?ancestor1) (code ?code1)(description ?desc1))
 (mrbs_component (name ?descendant1) (code ?code2))
 (test (and (eq "WGang" (sub-string 1 5 ?code1)) (eq "WGC" (sub-string 1 3 ?code2))))

;;; Above condition checks if the Wooden Gang Form Method has Column Form resource

 (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
 (pcbs_component (name ?ancestor) (code ?code3))
 (pcbs_component (name ?descendant) (code ?code4)(description ?desc2) )
 (test (and (eq "SupSTR" (sub-string 1 6 ?code3))(eq "Cols" (sub-string 1 4 ?code4))))

;;; Above condition checks if the Column exists in superstructure

=>

 (bind ?a1 (send (nth$ 1 (send ?val1 get-attribute_value_list)) get-value1))
 (bind ?a2 (send ?val2 get-value1))
 (if (not (>= ?a1 ?a2))
    then (printout formworkFile  "The Method \""?desc1 "\" for PCBS component \"Column\""  crlf
              " has insufficient site storage space at \"Site Location\"." crlf crlf )
    else (printout formworkFile  "The Method \""?desc1  "\" for PCBS component \"Column\"" crlf
              " has sufficient site storage space at \"Site Location\"."  crlf crlf ))
)
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;; This rule checks for uniformity of column element and possible reuses for column gang form
;; the horizontal recycling of formwork is not consider in this rule

(defrule column_reuse_check_wooden_gangform_column

    (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
    (mrbs_component (name ?ancestor1) (code ?code1)(description ?desc1))
    (mrbs_component (name ?descendant1) (code ?code2))
    (test (and (eq "WGang"(sub-string 1 5 ?code1)) (eq "WGC" (sub-string 1 3 ?code2))))

    (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
    (pcbs_component (name ?ancestor) (code ?code3))
    (pcbs_component (name ?descendant) (code ?code4)(description ?desc2))
    (test (and (eq "SupSTR" (sub-string 1 6 ?code3))(eq "Colm" (sub-string 1 4 ?code4))))

;;; Above condition checks if the Columns exists in superstructure

    (pcbs_component (name ?descendant)(code ?code4)
                    (attributes $?ahead "Length" $?atail)
                    (attribute_values $?avhead ?val1 $?avtail))
    (test (eq (length$ $?ahead)(length $?avhead)))
    (test (eq (length$ $?atail)(length $?avtail)))

    (pcbs_component (name ?descendant)(code ?code4)
                    (attributes $?ahead1 "Height" $?atail1)
                    (attribute_values $?avhead1 ?val2 $?avtail1))
    (test (eq (length$ $?ahead1)(length $?avhead1)))
    (test (eq (length$ $?atail1)(length $?avtail1)))
```

```
  (mrbs_component (name ?ancestor1)(code ?code1)
                (attributes $?ahead2 "Min. Reuse Required" $?atail2)
                (attribute_values $?avhead2 ?val3 $?avtail2))
  (test (eq (length$ $?ahead2)(length $?avhead2)))
  (test (eq (length$ $?atail2)(length $?avtail2)))

=>

  (bind ?lmax (length$ (send ?val1 get-location_list)))
  (bind ?a1 (send (nth$ 1 (send ?val1 get-attribute_value_list)) get-value1)) ;;;Length
  (bind ?a2 (send (nth$ 1 (send ?val2 get-attribute_value_list)) get-value1)) ;;;Height
  (bind ?a3 (send ?val3 get-value1)) ;;; Min. Reuse Required
  (bind ?reuse 0)
  (bind ?l1 ?lmax)
  (while (> ?l1 0)
      (bind ?a4 (send (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-attribute_value_list)) get-value1))
       (bind ?a5 (send (nth$ (- ?lmax(- ?l1 1)) (send ?val2 get-attribute_value_list)) get-value1))
      (if (and (eq  ?a1 ?a4)(eq  ?a2 ?a5))
            then (bind ?reuse (+ ?reuse 1)))
      (bind ?a1 ?a4)
       (bind ?a2 ?a5)
      (bind ?l1 (- ?l1 1))
      )
   (if (not(>= ?reuse ?a3))
     then (printout formworkFile "The Method \""?desc1 "\" for PCBS component \""?desc2 "\"" crlf
                 " is infeasible due to insufficient number of reuses: " ?reuse"."  crlf crlf )
     else (printout formworkFile "The Method  \""?desc1 "\" for PCBS component \""?desc2 "\""crlf
                 " is economically feasible due to sufficient number of reuses: " ?reuse"."  crlf crlf ))
 )
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;; This rule checks for rate of pour for column element

(defrule column_rate_of_concrete_pour_wooden_gangform_column

   (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
   (mrbs_component (name ?ancestor1) (code ?code1)(description ?desc1))
   (mrbs_component (name ?descendant1) (code ?code2))
   (test (and (eq "WGang" (sub-string 1 5 ?code1)) (eq "WGC" (sub-string 1 3 ?code2))))

   (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
   (pcbs_component (name ?ancestor) (code ?code3))
   (pcbs_component (name ?descendant) (code ?code4)(description ?desc2))
   (test (and (eq "SupSTR" (sub-string 1 6 ?code3))(eq "Cols" (sub-string 1 4 ?code4))))

  ;;; Above condition checks if the Columns exists in superstructure

   (pcbs_component (name ?descendant)(code ?code4)
                  (attributes $?ahead "Rate of Pour" $?atail)
                  (attribute_values $?avhead ?val1 $?avtail))
   (test (eq (length$ $?ahead)(length $?avhead)))
   (test (eq (length$ $?atail)(length $?avtail)))

   (mrbs_component (name ?ancestor1)(code ?code1)
                  (attributes $?ahead1 "Allowable Rate of Pour" $?atail1)
                  (attribute_values $?avhead1 ?val2 $?avtail1))
   (test (eq (length$ $?ahead1)(length $?avhead1)))
   (test (eq (length$ $?atail1)(length $?avtail1)))

 =>
```

```
(bind $?list1 (create$))    ;;;;;Creating empty field ;;;;;;;;
(bind $?list2 (create$))    ;;;;;Creating empty field ;;;;;;;;
(bind ?lmax (length$ (send ?val1 get-location_list)))
(bind ?a2 (send ?val2 get-value1))  ;;;; Allowable rate of pour ;;;;
(bind ?l1 ?lmax)
(bind ?temp1 0)
(bind ?temp2 0)
(while (> ?l1 0)
      (bind ?a1 (send (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-attribute_value_list)) get-value1))
      (if (not(>= ?a2 ?a1))
            then (bind ?temp1 1)
                  (bind $?list2 (create$ $?list2 (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list))))
            else (bind ?temp2 1)
                  (bind $?list1 (create$ $?list1 (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list)))))
    (bind ?l1 (- ?l1 1)))
 (if (eq ?temp1 1)
        then
      (printout formworkFile "The Method \""?desc1 "\" is not suitable for \"" ?desc2"\""   crlf
          " because of higher rate of pour required at locations "
          crlf tab $?list2 "."  crlf crlf))
(if (eq ?temp2 1)
        then
      (printout formworkFile "The Method \""?desc1 "\" is suitable for \"" ?desc2"\""  crlf
          " because the rate of pour required is within the required range at locations "
          crlf tab $?list1 "." crlf crlf))
)
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;; This rule checks for tie-spacing for Column element

(defrule column_Tie_spacing_check_wooden_gangform_colmn

    (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
    (mrbs_component (name ?ancestor1) (code ?code1)(description ?desc1))
    (mrbs_component (name ?descendant1) (code ?code2))
    (test (and (eq "WGang" (sub-string 1 5 ?code1)) (eq "WGC" (sub-string 1 3 ?code2))))

    (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
    (pcbs_component (name ?ancestor) (code ?code3))
    (pcbs_component (name ?descendant) (code ?code4)(description ?desc2))
    (test (and (eq "SupSTR" (sub-string 1 6 ?code3))(eq "Cols"(sub-string 1 4 ?code4))))

    ;;; Above condition checks if the Columns exists in superstructure

    (pcbs_component (name ?descendant)(code ?code4)(description ?desc)
                (attributes $?ahead "Tie Spacing" $?atail)
                (attribute_values $?avhead ?val1 $?avtail))
    (test (eq (length$ $?ahead)(length $?avhead)))
    (test (eq (length$ $?atail)(length $?avtail)))

    (mrbs_component (name ?ancestor1)(code ?code1)
                (attributes $?ahead1 "Allowable Tie Spacing" $?atail1)
                (attribute_values $?avhead1 ?val2 $?avtail1))
    (test (eq (length$ $?ahead1)(length $?avhead1)))
    (test (eq (length$ $?atail1)(length $?avtail1)))

 =>
```

```
(bind $?list1 (create$))    ;;;;;Creating empty field ;;;;;;;;
(bind $?list2 (create$))    ;;;;;Creating empty field ;;;;;;;;
(bind ?lmax (length$ (send ?val1 get-location_list)))
(bind ?l1 ?lmax)
(bind ?temp1 0)
(bind ?temp2 0)
(bind ?a21 (send ?val2 get-value1))
(bind ?a22 (send ?val2 get-value2))
(while (> ?l1 0)
      (bind ?a1 (send (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-attribute_value_list)) get-value1))
     (if (not (>= ?a22 ?a1))
               then (bind ?temp1 1)
                     (bind $?list2 (create$ $?list2 (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list))))
               else (bind ?temp2 1)
                    (bind $?list1 (create$ $?list1 (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list)))))
   (bind ?l1 (- ?l1 1)))
  (if (eq ?temp1 1)
     then (printout formworkFile "The Method \""?desc1 "\" is not suitable for \""?desc2 "\"" crlf
                  " the tie spacing exceeds the specified tie spacing "
                  " at locations "  crlf  tab $?list2 "."  crlf crlf))
 (if (eq ?temp2 1)
      then (printout formworkFile "The Method \""?desc1 "\" is suitable for \""?desc2 "\"" crlf
                  " the tie spacing required is within the specified range "
                  " at locations "  crlf  tab $?list1 "."  crlf crlf))
)
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;; This rule checks for timeframe for formwork of column and feasibility of column formwork method

(defrule column_Time_frame_for_formwork_wooden_gangform_column

   (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
   (mrbs_component (name ?ancestor1) (code ?code1)(description ?desc1))
   (mrbs_component (name ?descendant1) (code ?code2))
   (test (and (eq "WGang" (sub-string 1 5 ?code1)) (eq "WGC" (sub-string 1 3 ?code2))))

   (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant2))
   (mrbs_component (name ?descendant2) (code ?code3))
   (test (eq "FCrew" (sub-string 1 5 ?code3)))

   (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
   (pcbs_component (name ?ancestor) (code ?code4))
   (pcbs_component (name ?descendant) (code ?code5)(description ?desc2))
   (test (and (eq "SupSTR" (sub-string 1 6 ?code4))(eq "Cols" (sub-string 1 4 ?code5))))

   ;;; Above condition checks if the Columns exists in superstructure

   (pcbs_component (name ?descendant)(code ?code5)(description ?desc)
                (attributes $?ahead "Time Frame for Formwork" $?atail)
                (attribute_values $?avhead ?val1 $?avtail))
   (test (eq (length$ $?ahead)(length $?avhead)))
   (test (eq (length$ $?atail)(length $?avtail)))

   (pcbs_component (name ?descendant)(code ?code5)(description ?desc)
                (attributes $?ahead1 "Formwork Quantity" $?atail1)
                (attribute_values $?avhead1 ?val2 $?avtail1))
```

```
  (test (eq (length$ $?ahead1)(length $?avhead1)))
  (test (eq (length$ $?atail1)(length $?avtail1)))

  (mrbs_component (name ?ancestor1)(code ?code1)
                 (attributes $?ahead2 "Rate of Production"  $?atail2)
                 (attribute_values $?avhead2 ?val3 $?avtail2))
  (test (eq (length$ $?ahead2)(length $?avhead2)))
  (test (eq (length$ $?atail2)(length $?avtail2)))

  (mrbs_component (name ?descendant2)(code ?code3)
                 (attributes $?ahead3 "Number of Crew Members" $?atail3)
                 (attribute_values $?avhead3 ?val4 $?avtail3))
  (test (eq (length$ $?ahead3)(length $?avhead3)))
  (test (eq (length$ $?atail3)(length $?avtail3)))

=>

  (bind ?lmax (length$ (send ?val1 get-location_list)))
  (bind ?l1 ?lmax)
  (bind $?list1 (create$))
  (bind ?a3 (send ?val3 get-value1))
  (bind ?a4 (send ?val4 get-value1))
  (bind ?temp 0)
  (while (> ?l1 0)
     (bind ?a1 (send (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-attribute_value_list)) get-value1))
      (bind ?a2 (send (nth$ (- ?lmax(- ?l1 1)) (send ?val2 get-attribute_value_list)) get-value1))
    (if (not (>= ?a1 (/ ?a2 (* ?a4 ?a3))))
          then  (printout formworkFile "The Method \""?desc1 "\" is not suitable for \""?desc2 "\"" crlf
                 " because of lower rate of production;"
                 (format nil " the estimated resource usage is %7.2f crewhrs" (/ ?a2 (* ?a4 ?a3)))
                 " at location"
```

```
                    (nth$ (- ?lmax(- ?l1 1))(send ?val1 get-location_list))"."  crlf crlf )
            else  (bind ?temp 1)
                  (bind $?list1 (create$ $?list1 (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list)))))
    (bind ?l1 (- ?l1 1)))
  (if (eq ?temp  1)
      then (printout formworkFile "The Method \""?desc1 "\" is suitable for \""?desc2 "\"" crlf
                " in the given time frame by considering rate of production"
                " at locations "  crlf tab $?list1 "."  crlf crlf))
  )


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;; Rebar Placing for columns ;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;; This rule checks for site space for Rebar prefabrication method
;;; by taking relevant attributes from method and PCBS component

(defrule site_space_check_rebar_prefabrication_column_onsite

  (pcbs_component (code "SiteLoc")
                (attributes $?ahead "Rebar Fabrication Space Length" $?atail)
                (attribute_values $?avhead ?val1 $?avtail))
  (test (eq (length$ $?ahead)(length $?avhead)))
  (test (eq (length$ $?atail)(length $?avtail)))

  (pcbs_component (code "SiteLoc")
                (attributes $?ahead1 "Rebar Fabrication Space Width" $?atail1)
                (attribute_values $?avhead1 ?val2 $?avtail1))
  (test (eq (length$ $?ahead1)(length $?avhead1)))
  (test (eq (length$ $?atail1)(length $?avtail1)))
```

```
(ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
(pcbs_component (name ?ancestor) (code ?code1))
(pcbs_component (name ?descendant) (code ?code2)(description ?desc2))
(test (and (eq "SupSTR" (sub-string 1 6 ?code1))(eq "Cols" (sub-string 1 4 ?code2))))

;;; Above condition checks if the Columns exists in superstructure

(pcbs_component (name ?descendant)(code ?code2)
                (attributes $?ahead2 "Max. Height" $?atail2)
                (attribute_values $?avhead2 ?val3 $?avtail2))
(test (eq (length$ $?ahead2)(length $?avhead2)))
(test (eq (length$ $?atail2)(length $?avtail2)))

(ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
(mrbs_component (name ?ancestor1) (code ?code3))
(mrbs_component (name ?descendant1) (code ?code4)(description ?desc1))
(test (and (eq "RebarCol" (sub-string 1 8 ?code3)) (eq "PreFab" (sub-string 1 6 ?code4))))

;;; Above condition checks if operation has the method PreFabrication

(mrbs_component (name ?descendant1)(code ?code4)
                (attributes $?ahead3 "Rebar Fabrication Site Length Required" $?atail3)
                (attribute_values $?avhead3 ?val4 $?avtail3))
(test (eq (length$ $?ahead3)(length $?avhead3)))
(test (eq (length$ $?atail3)(length $?avtail3)))

(mrbs_component (name ?descendant1)(code ?code4)
                (attributes $?ahead4 "Rebar Fabrication Site Width Required" $?atail4)
                (attribute_values $?avhead4 ?val5 $?avtail4))
(test (eq (length$ $?ahead4)(length $?avhead4)))
```

```
 (test (eq (length$ $?atail4)(length $?avtail4)))

 (mrbs_component (name ?descendant1)(code ?code4)
                (attributes $?ahead5 "Rebar Fabrication Site Area Required" $?atail5)
                (attribute_values $?avhead5 ?val6 $?avtail5))
 (test (eq (length$ $?ahead5)(length $?avhead5)))
 (test (eq (length$ $?atail5)(length $?avtail5)))

=>

 (bind ?lmax (length$ (send ?val3 get-location_list)))
 (bind ?a3 (send (nth$ 1 (send ?val3 get-attribute_value_list)) get-value1));;;height;;;
 (bind ?a7 4) ;;;Assumed Lap Length;;;
 (bind ?a1 (send (nth$ 1 (send ?val1 get-attribute_value_list)) get-value1)) ;;;site rebar fabrication space length;;;
 (bind ?a2 (send (nth$ 1 (send ?val2 get-attribute_value_list)) get-value1)) ;;;site rebar fabrication space width;;;
 (bind ?a4 (send ?val4 get-value1)) ;;; Site Length required ;;;
 (bind ?a5 (send ?val5 get-value1)) ;;; Site Width required ;;;
 (bind ?a6 (send ?val6 get-value1)) ;;; Site Area required ;;;
 (bind ?l1 ?lmax)
 (bind ?max ?a3)
 (while (> ?l1 0)
     (bind ?a4 (send (nth$ (- ?lmax(- ?l1 1)) (send ?val3 get-attribute_value_list)) get-value1));;;
     (if (< ?max ?a4)
          then (bind ?max ?a4))
   (bind ?l1 (- ?l1 1))
  )
 (if (not (>= ?a1 (* (+ ?max ?a7 4)3))) ;;; (Max. Height of column + lap length + 4 ft space)x 3 ;;;
     then (printout rebarFile "The Method \""?desc1 "\" does not have sufficient"  crlf
               " \"Onsite Fabrication Space Length\" for \""?desc2 "\"."  crlf crlf )
     else (printout rebarFile "The Method \""?desc1 "\" has sufficient"  crlf
               " \"Onsite Fabrication Space Length\" for \""?desc2 "\"."  crlf crlf ))
```

```
  (if (not (>= (* ?a1 ?a2) ?a6))
      then (printout rebarFile "The Method \""?desc1 "\" does not have sufficient"  crlf
                    " \"Onsite Fabrication Space Area\" for \""?desc2 "\"."  crlf crlf )
      else (printout rebarFile "The Method \""?desc1 "\" has sufficient"  crlf
                    " \"Onsite Fabrication Space Area for \""?desc2 "\"."  crlf crlf ))
  )


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule site_space_check_storage_for_rebar_prefabrication_column

  (pcbs_component (code "SiteLoc")(attributes $?ahead "Rebar Storage Space Length" $?atail)
                (attribute_values $?avhead ?val1 $?avtail))
  (test (eq (length$ $?ahead)(length $?avhead)))
  (test (eq (length$ $?atail)(length $?avtail)))

  (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
  (pcbs_component (name ?ancestor) (code ?code1))
  (pcbs_component (name ?descendant) (code ?code2)(description ?desc2))
  (test (and (eq "SupSTR" (sub-string 1 6 ?code1))(eq "Cols" (sub-string 1 4 ?code2))))

  (pcbs_component (name ?descendant)(code ?code2)
                (attributes $?ahead1  "Max. Height" $?atail1)
                (attribute_values $?avhead1 ?val3 $?avtail1))
  (test (eq (length$ $?ahead1)(length $?avhead1)))
  (test (eq (length$ $?atail1)(length $?avtail1)))

  (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
  (mrbs_component (name ?ancestor1) (code ?code3))
  (mrbs_component (name ?descendant1) (code ?code4)(description ?desc1))
```

```
  (test (and (eq "RebarCol" (sub-string 1 8 ?code3)) (eq "PreFab" (sub-string 1 6 ?code4))))

  (mrbs_component (name ?descendant1)(code ?code4)
              (attributes $?ahead2 "Rebar Site Storage Length Required" $?atail2)
              (attribute_values $?avhead2 ?val4 $?avtail2))
  (test (eq (length$ $?ahead2)(length $?avhead2)))
  (test (eq (length$ $?atail2)(length $?avtail2)))

=>

  (bind ?lmax (length$ (send ?val3 get-location_list)))
  (bind ?a3 (send (nth$ 1 (send ?val3 get-attribute_value_list)) get-value1));;;height;;;
  (bind ?a7 4) ;;;Assumed Lap Length;;;
  (bind ?a1 (send (nth$ 1 (send ?val1 get-attribute_value_list)) get-value1)) ;;;site rebar storage space length;;;
  (bind ?a4 (send ?val4 get-value1)) ;;; Site storage space Length required ;;;
  (bind ?l1 ?lmax)
  (bind ?max ?a3)
  (while (> ?l1 0)
     (bind ?a8 (send (nth$ (- ?lmax(- ?l1 1)) (send ?val3 get-attribute_value_list)) get-value1));;;
    (if (< ?max ?a8)
     then (bind ?max ?a8))
   (bind ?l1 (- ?l1 1))
  )
     (bind ?a9 (max ?a4 (+ ?max ?a7 4)));;;Max.of (Max. Height of column + lap length + 4 ft space)& required ;;;
  (if (not (>= ?a1 ?a9))
     then (printout rebarFile "The Method \""?desc1 "\" does not have sufficient"  crlf
              " site \"Rebar Storage Space Length\" for \""?desc2 "\"."  crlf crlf )
     else (printout rebarFile "The Method \""?desc1 "\" has sufficient"  crlf
              " site \"Rebar Storage Space Length\" for \""?desc2 "\"."  crlf crlf ))
 )
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defrule time_frame_for_rebar_prefabrication_column

   (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
   (mrbs_component (name ?ancestor1) (code ?code3))
   (mrbs_component (name ?descendant1) (code ?code4)(description ?desc1))
   (test (and (eq "RebarCol" (sub-string 1 8 ?code3)) (eq "PreFab" (sub-string 1 6 ?code4))))

  ;;; Above condition checks if operation has the method PreFabrication

   (ancestor (ancestor_component ?ancestor) (descendant_component ?descendant))
   (pcbs_component (name ?ancestor) (code ?code1))
   (pcbs_component (name ?descendant) (code ?code2)(description ?desc2))
   (test (and (eq "SupSTR" (sub-string 1 6 ?code1))(eq "Cols" (sub-string 1 4 ?code2))))

  ;;; Above condition checks if the Columns exists in superstructure

   (ancestor (ancestor_component ?descendant1) (descendant_component ?descendant2))
   (mrbs_component (name ?descendant2) (code ?code5))
   (test (eq "RCrew" (sub-string 1 5 ?code5)))

  ;;; Above condition checks if the method has rebar crew

   (pcbs_component (name ?descendant)(code ?code2)
                 (attributes $?ahead "Time Frame for Rebar" $?atail)
                 (attribute_values $?avhead ?val1 $?avtail))
   (test (eq (length$ $?ahead)(length $?avhead)))
   (test (eq (length$ $?atail)(length $?avtail)))

   (pcbs_component (name ?descendant)(code ?code2)
```

```
              (attributes $?ahead1 "Rebar Quantity" $?atail1)
              (attribute_values $?avhead1 ?val2 $?avtail1))
(test (eq (length$ $?ahead1)(length $?avhead1)))
(test (eq (length$ $?atail1)(length $?avtail1)))

(mrbs_component (name ?descendant1)(code ?code4)
              (attributes $?ahead2 "Rate of Production" $?atail2)
              (attribute_values $?avhead2 ?val3 $?avtail2))
(test (eq (length$ $?ahead2)(length $?avhead2)))
(test (eq (length$ $?atail2)(length $?avtail2)))

(mrbs_component (name ?descendant2)(code ?code5)
              (attributes $?ahead3 "Number of Crew Members" $?atail3)
              (attribute_values $?avhead3 ?val4 $?avtail3))
(test (eq (length$ $?ahead3)(length $?avhead3)))
(test (eq (length$ $?atail3)(length $?avtail3)))

=>

(bind $?list1 (create$))    ;;;;;Creating empty field ;;;;;;;;
(bind ?lmax (length$ (send ?val1 get-location_list)))
(bind ?a3 (send ?val3 get-value1)) ;;; Rate of production
(bind ?a4 (send ?val4 get-value1)) ;;; Crew members
(bind ?l1 ?lmax)
(bind ?temp 0)
(while (> ?l1 0)
     (bind ?a1 (send (nth$ (- ?lmax(- ?l1 1))(send ?val1 get-attribute_value_list)) get-value1)) ;;; Time Frame for Rebar
     (bind ?a2 (send (nth$ (- ?lmax(- ?l1 1))(send ?val2 get-attribute_value_list)) get-value1)) ;;; Rebar Quantity
  (if (not (>= ?a1 (/ ?a2 (* ?a4 ?a3))))
   then (printout rebarFile "The Method \""?desc1 "\" is not suitable for \""?desc2 "\""  crlf
              " because of lower rate of production;"
```

```
                (format nil " the estimated resource usage is %7.2f crewhrs " (/ ?a2 (* ?a4 ?a3)))
                " at location"
                (nth$ (- ?lmax(- ?l1 1))(send ?val1 get-location_list))"."  crlf crlf )
      else (bind ?temp 1)
            (bind $?list1 (create$ $?list1 (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list)))))
   (bind ?l1 (- ?l1 1)))
  (if (eq ?temp  1)
     then (printout rebarFile "The Method \""?desc1 "\" is suitable for \""?desc2 "\""  crlf
                " in the given time frame by considering rate of production "
                " at locations "  crlf  tab $?list1 "." crlf crlf ))
 )


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; Concrete Placement for Columns ;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;; This rule checks for site space for Crane and bucket method
;;; by taking relevant attributes from method and PCBS component

(defrule site_space_check_crane_bucket_mathod_column

  (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))
  (mrbs_component (name ?ancestor1) (code ?code1)(description ?desc2))
  (mrbs_component (name ?descendant1) (code ?code2)(description ?desc1))
  (test (and (eq "ConcCol"(sub-string 1 7 ?code1)) (eq "CrBuck"(sub-string 1 6 ?code2))))

 ;;; Above condition checks if operation has the method Crane & Bucket

  (pcbs_component (code "SiteLoc")
              (attributes $?ahead "Parking Space Length" $?atail)
```

```
		(attribute_values $?avhead ?val1 $?avtail))
	(test (eq (length$ $?ahead)(length $?avhead)))
	(test (eq (length$ $?atail)(length $?avtail)))

	(pcbs_component (code "SiteLoc")
		(attributes $?ahead1 "Parking Space Width" $?atail1)
		(attribute_values $?avhead1 ?val2 $?avtail1))
	(test (eq (length$ $?ahead1)(length $?avhead1)))
	(test (eq (length$ $?atail1)(length $?avtail1)))

	(mrbs_component (name ?descendant1)(code ?code2)
		(attributes $?ahead2 "Parking Space Length Required" $?atail2)
		(attribute_values $?avhead2 ?val3 $?avtail2))
	(test (eq (length$ $?ahead2)(length $?avhead2)))
	(test (eq (length$ $?atail2)(length $?avtail2)))

	(mrbs_component (name ?descendant1)(code ?code2)
		(attributes $?ahead3 "Parking Space Width Required" $?atail3)
		(attribute_values $?avhead3 ?val4 $?avtail3))
	(test (eq (length$ $?ahead3)(length $?avhead3)))
	(test (eq (length$ $?atail3)(length $?avtail3)))

	(mrbs_component (name ?descendant1)(code ?code2)
		(attributes $?ahead4 "Parking Space Area Required" $?atail4)
		(attribute_values $?avhead4 ?val5 $?avtail4))
	(test (eq (length$ $?ahead4)(length $?avhead4)))
	(test (eq (length$ $?atail4)(length $?avtail4)))

=>

	(bind ?a1 (send (nth$ 1 (send ?val1 get-attribute_value_list)) get-value1)) ;;;; Parking Space Length ;;;;
```

```
(bind ?a2 (send (nth$ 1 (send ?val2 get-attribute_value_list)) get-value1)) ;;;; Parking Space Width ;;;
(bind ?a3 (send ?val3 get-value1)) ;;; Parking Space Length required ;;;
(bind ?a4 (send ?val4 get-value1)) ;;; Parking Space Width required ;;;
(bind ?a5 (send ?val5 get-value1)) ;;; Parking Space Area Required ;;;
(bind ?temp 0)
 (if (not (>= ?a1 ?a3))
   then (bind ?temp 1)
        (printout concretFile "The Method  \""?desc1 "\" does not have sufficient " crlf " \"Parking Space Length\", "
            ?a1" is less than " ?a3 " at \"Site location\". "  crlf crlf ))
(if (not (>= ?a2 ?a4))
   then (bind ?temp 1)
        (printout concreteFile "The Method  \""?desc1 "\" does not have sufficient  " crlf " \"Parking Space Width\", "
            ?a2" is less than " ?a4 " at \"Site location\". "  crlf crlf ))
(if (not (>= (* ?a1 ?a3) ?a5))
   then (bind ?temp 1)
        (printout concreteFile "The Method  \""?desc1 "\" does not have sufficient " crlf " \"Parking Space Area\", "
            (* ?a1 ?a3)" is less than " ?a5 " at \"Site location\". "  crlf crlf ))
(if (eq ?temp 0)
   then (printout concreteFile "The Method  \""?desc1 "\" for operation "crlf "\"" ?desc2"\""
                " has sufficient site space." crlf crlf ))
)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;; This rule checks for rate of concrete placement for crane and bucket method
;;; by taking relevant attributes from method and PCBS component

(defrule rate_of_placement_crane_bucket_mathod_column

   (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant2))
```

```
(pcbs_component (name ?ancestor1) (code ?code4))
(pcbs_component (name ?descendant2) (code ?code5)(description ?desc2))
(test (and (eq "SupSTR" (sub-string 1 6 ?code4))(eq "Cols"(sub-string 1 4 ?code5))))

;;; Above condition checks if the Slab exists in superstructure

(ancestor (ancestor_component ?ancestor2) (descendant_component ?descendant1))
(mrbs_component (name ?ancestor2) (code ?code1))
(mrbs_component (name ?descendant1) (code ?code2)(description ?desc1))
(test (and (eq "ConcCol"(sub-string 1 7 ?code1)) (eq "CrBuck"(sub-string 1 6 ?code2))))

;;; Above condition checks if operation has the method Crane & Bucket

(pcbs_component (name ?descendant2)(code ?code5)
        (attributes $?ahead "Concrete Quantity" $?atail )
        (attribute_values $?avhead ?val1 $?avtail))
(test (eq (length$ $?ahead)(length $?avhead)))
(test (eq (length$ $?atail)(length $?avtail)))
(pcbs_component (name ?descendant2)(code ?code5)
        (attributes $?ahead2 "Time Frame for Concreting" $?atail2)
        (attribute_values $?avhead2 ?val2 $?avtail2))
(test (eq (length$ $?ahead2)(length $?avhead2)))
(test (eq (length$ $?atail2)(length $?avtail2)))

(mrbs_component (name ?descendant1)(code ?code2)
        (attributes $?ahead3 "Rate of Concrete Placement" $?atail3)
        (attribute_values $?avhead3 ?val3 $?avtail3))
(test (eq (length$ $?ahead3)(length $?avhead3)))
(test (eq (length$ $?atail3)(length $?avtail3)))

=>
```

```
(bind $?list1 (create$))    ;;;;;Creating empty field ;;;;;;;;
(bind ?lmax (length$ (send ?val1 get-location_list)))
(bind ?a3 (send ?val3 get-value1)) ;;; Rate of Concrete Placement ;;;
(bind ?l1 ?lmax)
(bind ?temp 0)
(while (> ?l1 0)
    (bind ?a1 (send (nth$ (- ?lmax(- ?l1 1))(send ?val1 get-attribute_value_list)) get-value1)) ;;; Concrete Quantity ;;;
    (bind ?a2 (send (nth$ (- ?lmax(- ?l1 1))(send ?val2 get-attribute_value_list)) get-value1))
    (bind ?a4 (/ ?a1 ?a2))
 (if (not (>= ?a3 ?a4))
    then (printout concreteFile "The Method  \""?desc1 "\" is infeasible because of lower " crlf
                " concrete placement rate for \""?desc2 "\" is " ?a4 " at location "
                (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list ))"." crlf crlf )
    else (bind ?temp 1)
        (bind $?list1 (create$ $?list1 (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list)))))
 (bind ?l1 (- ?l1 1)))
(if (eq ?temp 1)
   then (printout concreteFile "The Method  \""?desc1 "\" is feasible considering rate of " crlf
               " concrete placement for  \""?desc2 "\" at locations " crlf tab $?list1 "." crlf crlf ))
)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;; This rule checks for rate of concrete placement for crane and bucket method
;;; by taking relevant attributes from method and PCBS component

(defrule max_size_of_aggregate_check_crane_bucket_mathod_column

  (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant2))
  (pcbs_component (name ?ancestor1) (code ?code4))
```

```
(pcbs_component (name ?descendant2) (code ?code5)(description ?desc2))
(test (and (eq "SupSTR"(sub-string 1 6 ?code4))(eq "Cols"(sub-string 1 4 ?code5))))

;;; Above condition checks if the Slab exists in superstructure

(ancestor (ancestor_component ?ancestor2) (descendant_component ?descendant1))
(mrbs_component (name ?ancestor2) (code ?code1))
(mrbs_component (name ?descendant1) (code ?code2)(description ?desc1))
(test (and (eq "ConcCol" (sub-string 1 7 ?code1)) (eq "CrBuck"(sub-string 1 6 ?code2))))

;;; Above condition checks if operation has the method Crane & Bucket

(pcbs_component (name ?descendant2) (code ?code5)
        (attributes $?ahead "Max. Size of Aggregate" $?atail )
        (attribute_values $?avhead ?val1 $?avtail))
(test (eq (length$ $?ahead)(length $?avhead)))
(test (eq (length$ $?atail)(length $?avtail)))

(mrbs_component (name ?descendant1)(code ?code2)
        (attributes $?ahead1 "Max. Size of Aggregate" $?atail1)
        (attribute_values $?avhead1 ?val2 $?avtail1))
(test (eq (length$ $?ahead1)(length $?avhead1)))
(test (eq (length$ $?atail1)(length $?avtail1)))

=>

(bind $?list1 (create$))
(bind ?lmax (length$ (send ?val1 get-location_list)))
(bind ?a2 (send ?val2 get-value1))
(bind ?l1 ?lmax)
(bind ?temp 0)
```

```
(while (> ?l1 0)
     (bind ?a1 (send (nth$ (- ?lmax(- ?l1 1))(send ?val1 get-attribute_value_list)) get-value1))
  (if (not (<= ?a1 ?a2))
     then (printout concreteFile "The Method \""?desc1 "\" is infeasible because \""?desc2"\"" crlf
                    " requires maximum size of aggregate "?a1 " inches at location "
                    (nth$ (- ?lmax(- ?l1 1))(send ?val1 get-location_list ))"." crlf crlf)
     else (bind ?temp 1)
          (bind $?list1 (create$ $?list1 (nth$ (- ?lmax(- ?l1 1)) (send ?val1 get-location_list)))))
  (bind ?l1 (- ?l1 1)))
(if (eq ?temp 1)
   then (printout concreteFile "The Method \""?desc1 "\" is feasible considering " crlf
       "  maximum size of aggregate for \"" ?desc2 "\" at locations "
       crlf tab $?list1"." crlf crlf ))
)
```

APPENDIX – E

Method Statement Feasibility Report Files

Appendix E Method Statement Feasibility Report Files

;;
;;;;;; Report for Formwork Methods ;;;;;;;;;;
;;

This is the Report generated for Method Statement - High-rise Superstructure Construction

Formwork Methods for PCBS components

The Method "Flying Truss Formwork for Slab" has sufficient assembly space for operation "Formwork for Slab"
at "Site location".

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay A"
is infeasible due to insufficient reuses 1.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay B"
is infeasible due to insufficient reuses 1.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay E"
is infeasible due to insufficient reuses 1.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay D"
is infeasible due to insufficient reuses 1.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay F"
is infeasible due to insufficient reuses 1.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay G"
is infeasible due to insufficient reuses 1.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay H"
is infeasible due to insufficient reuses 1.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay I"
is infeasible due to insufficient reuses 1.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay A1"
is feasible due to sufficient reuses 18.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay B1"
is feasible due to sufficient reuses 18.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay C1" is feasible due to sufficient reuses 18.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay D1" is infeasible due to insufficient reuses 2.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay E1" is feasible due to sufficient reuses 18.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay F1" is feasible due to sufficient reuses 19.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay G1" is feasible due to sufficient reuses 18.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay H1" is feasible due to sufficient reuses 19.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay J1" is feasible due to sufficient reuses 19.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay K1" is feasible due to sufficient reuses 18.

The Method "Flying Truss Formwork for Slab" for PCBS component "SlabBay L1" is feasible due to sufficient reuses 18.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 13.52 crewhrs at location 3.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 4.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 5.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 6.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 7.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 8.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 9.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 10.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 11.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 12.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 13.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 14.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 15.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 16.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 17.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 18.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab"

because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 19.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 20.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 11.96 crewhrs at location 21.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 9.22 crewhrs at location 22.

The Method "Flying Truss Formwork for Slab" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 9.21 crewhrs at location 23.

The Method "Flying Truss Formwork for Slab" is suitable for "High Rise Floor Slab" in the given time frame by considering rate of production at locations ("GFL" "2").

The Method "Wooden Gang Formwork" for PCBS component "Column A" is infeasible due to insufficient number of reuses: 2.

The Method "Wooden Gang Formwork" for PCBS component "Column B" is infeasible due to insufficient number of reuses: 19.

The Method "Wooden Gang Formwork" for PCBS component "Column C" is infeasible due to insufficient number of reuses: 21.

The Method "Wooden Gang Formwork" for PCBS component "Column D" is infeasible due to insufficient number of reuses: 19.

The Method "Wooden Gang Formwork" for PCBS component "Column E" is infeasible due to insufficient number of reuses: 1.

The Method "Wooden Gang Formwork" for PCBS component "Column F" is infeasible due to insufficient number of reuses: 1.

The Method "Wooden Gang Formwork" for PCBS component "Column G" is infeasible due to insufficient number of reuses: 1.

The Method "Wooden Gang Formwork" for PCBS component "Column H" is infeasible due to insufficient number of reuses: 18.

The Method "Wooden Gang Formwork" for PCBS component "Column K"
is infeasible due to insufficient number of reuses: 18.

The Method "Wooden Gang Formwork" for PCBS component "Column L"
is infeasible due to insufficient number of reuses: 1.

The Method "Wooden Gang Formwork" for PCBS component "Column"
has sufficient site storage space at "Site Location".

The Method "Wooden Gang Formwork" is suitable for "Columns"
because the rate of pour required is within the required range at locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Wooden Gang Formwork" is suitable for "Columns"
in the given time frame by considering rate of production at locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Aluminum Waler Jumpform" for PCBS component "Core Wall A"
is infeasible due to insufficient reuses 18.

The Method "Aluminum Waler Jumpform" for PCBS component "Core Wall B"
is infeasible due to insufficient reuses 18.

The Method "Aluminum Waler Jumpform" for PCBS component "Core Wall C"
is infeasible due to insufficient reuses 18.

The Method "Aluminum Waler Jumpform" for PCBS component "Core Wall D"
is infeasible due to insufficient reuses 20.

The Method "Aluminum Waler Jumpform" for PCBS component "Core Wall E"
is infeasible due to insufficient reuses 19.

The Method "Aluminum Waler Jumpform" for PCBS component "Core Wall F"
is infeasible due to insufficient reuses 18.

The Method "Aluminum Waler Jumpform" for PCBS component "High Rise Tower Core"
has sufficient site storage space at "Site location".

The Method "Aluminum Waler Jumpform" is suitable considering rate of
pour for "High Rise Tower Core" at locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Aluminum Waler Jumpform" is not suitable for "High Rise Tower Core" because of lower rate of production; the estimated resource usage is 10.68 crewhrs at location 2.

The Method "Aluminum Waler Jumpform" is suitable for "High Rise Tower Core" in the given time frame by considering rate of production at locations ("GFL" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall A - non typical walls at GFL"
is infeasible due to insufficient number of reuses 1.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall B - non typical walls at 2nd floor"
is infeasible due to insufficient number of reuses 1.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall M"
is infeasible due to insufficient number of reuses 17.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall A1"
is infeasible due to insufficient number of reuses 17.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall B1"
is infeasible due to insufficient number of reuses 17.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall C1"
is infeasible due to insufficient number of reuses 18.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall D1"
is infeasible due to insufficient number of reuses 18.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall E1"
is infeasible due to insufficient number of reuses 18.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall F1"
is infeasible due to insufficient number of reuses 18.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall G1"
is infeasible due to insufficient number of reuses 1.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall H1"
is infeasible due to insufficient number of reuses 17.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall M1" is infeasible due to insufficient number of reuses 16.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall S1" is infeasible due to insufficient number of reuses 16.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall T1" is infeasible due to insufficient number of reuses 17.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall U1" is infeasible due to insufficient number of reuses 16.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall V1" is infeasible due to insufficient number of reuses 17.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall W1" is infeasible due to insufficient number of reuses 1.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall C2" is infeasible due to insufficient number of reuses 2.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall C3" is infeasible due to insufficient number of reuses 1.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall M2" is infeasible due to insufficient number of reuses 2.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall U2" is infeasible due to insufficient number of reuses 1.

The Method "Wooden Gang Formwork" for PCBS component "Shear Wall V2" is infeasible due to insufficient number of reuses 1.

The Method "Wooden Gang Formwork" for PCBS component "Shear Walls" has sufficient site storage space at "Site location".

The Method "Wooden Gang Formwork" is feasible considering rate of pour required for "Shear Walls" concrete placement at locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 45.24 crewhrs at location GFL.

The Method "Wooden Gang Formwork" is not suitable for

"Shear Walls" because of lower rate of production; the estimated resource usage is 13.99 crewhrs at location 2.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 12.97 crewhrs at location 3.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.56 crewhrs at location 4.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 5.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 6.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 7.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 8.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 9.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 10.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 11.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 12.

The Method "Wooden Gang Formwork" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 13.

The Method "Wooden Gang Formwork" is not suitable for
"Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 14.

The Method "Wooden Gang Formwork" is not suitable for
"Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 15.

The Method "Wooden Gang Formwork" is not suitable for
"Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 16.

The Method "Wooden Gang Formwork" is not suitable for
"Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 17.

The Method "Wooden Gang Formwork" is not suitable for
"Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 18.

The Method "Wooden Gang Formwork" is not suitable for
"Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 19.

The Method "Wooden Gang Formwork" is not suitable for
"Shear Walls" because of lower rate of production; the estimated resource usage is 15.15 crewhrs at location 20.

The Method "Wooden Gang Formwork" is not suitable for
"Shear Walls" because of lower rate of production; the estimated resource usage is 14.50 crewhrs at location 21.

The Method "Wooden Gang Formwork" is suitable for
"Shear Walls" in the given time frame by considering rate of production at locations ("22" "23").

;;
;;;;;;; Report for Rebar Placement Methods ;;;;;;;;;;;
;;

This is the Report generated for Method Statement - High-rise Superstructure Construction

Rebar Placement Methods for PCBS components

The Method "Rebar Assembly" does not have sufficient "Storage Space Width" 10.0 for component "High Rise Floor Slab" is less than 12.0 at "Site location".

The Method "Rebar Assembly" does not have sufficient "Storage Space Area" 640.0 for component "High Rise Floor Slab" is less than 720.0 at "Site location".

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 36.33 crewhrs at location GFL.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.97 crewhrs at location 3.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 4.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 5.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 6.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 7.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab"

because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 8.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 9.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 10.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 11.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 12.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 13.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 14.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 15.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 16.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 17.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 18.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 19.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 20.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 16.07 crewhrs at location 21.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 12.40 crewhrs at location 22.

The Method "Rebar Assembly" is not suitable for "High Rise Floor Slab" because of lower rate of production; the estimated resource usage is 12.39 crewhrs at location 23.

The Method "Rebar Assembly" is suitable for "High Rise Floor Slab" in the given time frame by considering rate of production at locations ("2").

The Method "Rebar Prefabrication" does not have sufficient "Onsite Fabrication Space Length" for "Columns".

The Method "Rebar Prefabrication" does not have sufficient "Onsite Fabrication Space Area" for "Columns".

The Method "Rebar Prefabrication" has sufficient site "Rebar Storage Space Length" for "Columns".

The Method "Rebar Prefabrication" is suitable for "Columns" in the given time frame by considering rate of production at locations ("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Partial Rebar Prefabrication" for "High Rise Tower Core" does not have sufficient "Onsite Prefabrication Space".

The Method "Partial Rebar Prefabrication" is not suitable for "High Rise Tower Core" because of lower rate of production; the estimated resource usage is 8.82 crewhrs at location GFL.

The Method "Partial Rebar Prefabrication" is not suitable for "High Rise Tower Core" because of lower rate of production; the estimated resource usage is 9.59 crewhrs at location 2.

The Method "Partial Rebar Prefabrication" is not suitable for "High Rise Tower Core" because of lower rate of production; the estimated resource usage is 8.38 crewhrs at location 22.

The Method "Partial Rebar Prefabrication" is suitable for "High Rise Tower Core" in the given time frame by considering rate of production at locations ("3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "23").

The Method "Rebar Assembly" is feasible because of presence of shear zones in "High Rise Tower Core".

The Method "Partial Rebar Prefabrication" for "Shear Walls" does not have sufficient "Onsite Prefabrication Space".

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 11.06 crewhrs at location 2.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.32 crewhrs at location 3.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 11.15 crewhrs at location 4.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 5.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 6.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 7.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 8.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 9.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 10.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 11.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 12.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 13.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 14.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 15.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 16.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 17.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 18.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 19.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls" because of lower rate of production; the estimated resource usage is 10.07 crewhrs at location 20.

The Method "Partial Rebar Prefabrication" is not suitable for "Shear Walls"

because of lower rate of production; the estimated resource usage is 9.22 crewhrs at location 21.

The Method "Partial Rebar Prefabrication" is suitable for "Shear Walls" in the given time frame by considering rate of production at locations ("GFL" "22" "23").

The Method "Rebar Assembly" is feasible because of presence of shear zones in "Shear Walls".

;;
;;;;;; Report for Concrete Placement Methods ;;;;;;;;;;
;;;

This is the Report generated for Method Statement - High-rise Superstructure Construction

Concrete Placement Methods for PCBS components

The Method "Concrete Placing with Crane & Bucket" for operation "Concrete placement for Slab"
has sufficient site space.

The Method "Concrete Placing with Crane & Bucket" for operation "Concrete placement for Core"
has sufficient site space.

The Method "Concrete Placing with Crane & Bucket" for operation "Concrete placement for Walls"
has sufficient site space.

The Method "Concrete Placing with Crane & Bucket" for operation "Concrete placing for Columns" has sufficient site space.

The Method "Concrete Placing with Crane & Bucket" is feasible considering rate of concrete placement for "High Rise Floor Slab" at its locations ("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Concrete Placing with Crane & Bucket" is feasible considering maximum aggregate size for "High Rise Floor Slab" concrete placement at locations ("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Concrete Placing with Crane & Bucket" is feasible considering rate of concrete placement for "Columns" at locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Concrete Placing with Crane & Bucket" is feasible considering maximum size of aggregate for "Columns" at locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Concrete Placing with Crane & Bucket" is feasible considering rate of concrete placement for "High Rise Tower Core" at its locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Concrete Placing with Crane & Bucket" is feasible considering maximum aggregate size for "High Rise Tower Core" concrete placement at locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Concrete Placing with Crane & Bucket" is feasible considering rate of concrete placement for "Shear Walls" at its locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

The Method "Concrete Placing with Crane & Bucket" is feasible considering maximum aggregate sizes for "Shear Walls" at locations
("GFL" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12" "13" "14" "15" "16" "17" "18" "19" "20" "21" "22" "23").

APPENDIX – F

PCBS and M&RBS Hierarchy Reports

Test Report

File Used: PROJ03\TEST

Select: Selectively

Report Date: 28OCT02
Report Time: 11:20:15pm
Revision Number: 0
Progress Date:

PHYSICAL COMPONENT BREAKDOWN STRUCTURE PCBS PATH	DESCRIPTION	Type
GIA.Tower.SupSTR.VertEle.Cols.Colm4	Column D	Subelement

ATTRIBUTE

INHERITED	DESCRIPTION / COND VALUE 1	VALUE 2	CLASS	TYPE	UNIT
Y	Formwork Quantity		Area Properties	Quantitative	ft2
Y	Rebar Quantity		Material Quantities	Quantitative	Tn
Y	Concrete Quantity		Area Properties	Quantitative	yd3
Y	Surface Area		Area Properties	Quantitative	ft2
Y	Time Frame for Concreting		Duration Properties	Quantitative	hr
Y	Time Frame for Rebar		Duration Properties	Quantitative	hr
Y	Time Frame for Formwork		Duration Properties	Quantitative	hr
Y	Shape		Physical Properties	Linguistic	
Y	Slump Range		Concrete Properties	Quantitative	in
Y	Max. Size of Aggregate		Concrete Properties	Quantitative	in
Y	Rate of Pour		Concrete Properties	Quantitative	ft
Y	Length		Dimension Properties	Quantitative	ft
Y	Width		Dimension Properties	Quantitative	ft
Y	Height		Dimension Properties	Quantitative	ft
Y	Number of Elements		General Properties	Quantitative	No.
Y	Max. Height		Physical Properties	Quantitative	ft

PLANNED ATTRIBUTE VALUE

INHERITED / LOC	DESCRIPTION / RANGE	COND VALUE 1	VALUE 2	CLASS	TYPE	UNIT	AGGREGATED
Y	Formwork Quantity			Area Properties	Quantitative	ft2	N
GFL	-GFL	AND EQ	121				
5	-20	AND EQ	86.94				
Y	Rebar Quantity			Material Quantities	Quantitative	Tn	N
GFL	-GFL	AND EQ	0.41				
5	-20	AND EQ	0.19				
Y	Concrete Quantity			Area Properties	Quantitative	yd3	N
GFL	-GFL	AND EQ	2.44				
5	-20	AND EQ	1.11				
Y	Surface Area			Area Properties	Quantitative	ft2	N
Y	Time Frame for Concreting			Duration Properties	Quantitative	hr	N
Y	Time Frame for Rebar			Duration Properties	Quantitative	hr	N
Y	Time Frame for Formwork			Duration Properties	Quantitative	hr	N
Y	Shape			Physical Properties	Linguistic		N
GFL	-20	AND EQ	Rectangular				
Y	Slump Range			Concrete Properties	Quantitative	in	N
Y	Max. Size of Aggregate			Concrete Properties	Quantitative	in	N
Y	Rate of Pour			Concrete Properties	Quantitative	ft	N
Y	Length			Dimension Properties	Quantitative	ft	N
GFL	-20	AND EQ	4				
Y	Width			Dimension Properties	Quantitative	ft	N

```
|   GFL     -20      AND  WR  0.83              1.5                                              |
|   Y     Height                               Dimension Properties Quantitative ft         N    |
|   GFL     -GFL     AND  EQ  11                                                                 |
|   2       -4       AND  EQ  9                                                                  |
|   5       -20      AND  EQ  9                                                                  |
|   Y     Number of Elements                   General Properties   Quantitative No.        N    |
|   GFL     -GFL     AND  EQ  1                                                                  |
|   5       -20      AND  EQ  1                                                                  |
|   Y     Max. Height                          Physical Properties  Quantitative ft         N    |
-------------------------------------------------------------------------------------------------

-------------------------------------------------------------------------------------------------
PHYSICAL COMPONENT BREAKDOWN STRUCTURE                                              |Type       |
PCBS PATH                               |DESCRIPTION                                |           |
GIA.Tower.SupSTR.VertEle.Core.CWall1    |         Core Wall A                       |Subelement |
-------------------------------------------------------------------------------------------------
   |ATTRIBUTE                                                                       |
   |INHERITED DESCRIPTION                      CLASS                TYPE      UNIT  |
   |            COND VALUE 1           VALUE 2                                      |
   |   Y     Formwork Quantity                 Area Properties      Quantitative ft2 |
   |   Y     Rebar Quantity                    Material Quantities  Quantitative Tn  |
   |   Y     Concrete Quantity                 Area Properties      Quantitative yd3 |
   |   Y     Surface Area                      Area Properties      Quantitative ft2 |
   |   Y     Time Frame for Concreting         Duration Properties  Quantitative hr  |
   |   Y     Time Frame for Rebar              Duration Properties  Quantitative hr  |
   |   Y     Time Frame for Formwork           Duration Properties  Quantitative hr  |
   |   Y     Shape                             Physical Properties  Linguistic       |
   |   Y     Slump Range                       Concrete Properties  Quantitative in  |
   |   Y     Max. Size of Aggregate            Concrete Properties  Quantitative in  |
   |   Y     Rate of Pour                      Concrete Properties  Quantitative ft  |
   |   Y     Length                            Dimension Properties Quantitative ft  |
   |   Y     Height                            Dimension Properties Quantitative ft  |
   |   Y     Width                             Dimension Properties Quantitative ft  |
   |   Y     Number of Elements                General Properties   Quantitative No. |
-------------------------------------------------------------------------------------------------
   |PLANNED ATTRIBUTE VALUE                                                                      |
   |INHERITED DESCRIPTION                      CLASS                TYPE      UNIT    AGGREGATED|
   |   LOC RANGE    COND VALUE 1         VALUE 2                                                 |
   |   Y     Formwork Quantity                 Area Properties      Quantitative ft2        N    |
   |   GFL     -GFL     AND  EQ  467.5                                                           |
   |   2       -2       AND  EQ  510                                                             |
   |   3       -4       AND  EQ  382.5                                                           |
   |   5       -21      AND  EQ  373.5                                                           |
   |   22      -22      AND  EQ  415                                                             |
   |   23      -23      AND  EQ  267.50                                                          |
   |   Y     Rebar Quantity                    Material Quantities  Quantitative Tn         N    |
   |   GFL     -GFL     AND  EQ  1.60                                                            |
   |   2       -2       AND  EQ  1.75                                                            |
   |   3       -4       AND  EQ  1.32                                                            |
   |   5       -21      AND  EQ  1.29                                                            |
   |   22      -22      AND  EQ  1.43                                                            |
   |   23      -23      AND  EQ  0.5                                                             |
   |   Y     Concrete Quantity                 Area Properties      Quantitative yd3        N    |
   |   GFL     -GFL     AND  EQ  9.52                                                            |
   |   2       -2       AND  EQ  10.39                                                           |
   |   3       -4       AND  EQ  7.83                                                            |
   |   5       -21      AND  EQ  7.62                                                            |
   |   22      -22      AND  EQ  8.47                                                            |
   |   23      -23      AND  EQ  2.95                                                            |
   |   Y     Surface Area                      Area Properties      Quantitative ft2        N    |
   |   Y     Time Frame for Concreting         Duration Properties  Quantitative hr         N    |
```

```
|  Y     Time Frame for Rebar                  Duration Properties  Quantitative hr           N   |
|  Y     Time Frame for Formwork               Duration Properties  Quantitative hr           N   |
|  Y     Shape                                 Physical Properties  Linguistic                N   |
|  GFL     -23       AND  EQ  Rectangular                                                         |
|  Y     Slump Range                           Concrete Properties  Quantitative in           N   |
|  Y     Max. Size of Aggregate                Concrete Properties  Quantitative in           N   |
|  Y     Rate of Pour                          Concrete Properties  Quantitative ft           N   |
|  Y     Length                                Dimension Properties Quantitative ft           N   |
|  GFL     -2        AND  EQ  17.5                                                                |
|  3       -4        AND  EQ  18                                                                  |
|  5       -22       AND  EQ  17.5                                                                |
|  23      -23       AND  EQ  12.75                                                               |
|  Y     Height                                Dimension Properties Quantitative ft           N   |
|  GFL     -GFL      AND  EQ  11                                                                  |
|  2       -2        AND  EQ  12                                                                  |
|  3       -21       AND  EQ  9                                                                   |
|  22      -23       AND  EQ  10                                                                  |
|  Y     Width                                 Dimension Properties Quantitative ft           N   |
|  GFL     -22       AND  EQ  1.25                                                                |
|  23      -23       AND  EQ  0.625                                                               |
|  Y     Number of Elements                    General Properties   Quantitative No.          N   |
|  GFL     -23       AND  EQ  1                                                                   |
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
PHYSICAL COMPONENT BREAKDOWN STRUCTURE                                                  |Type        |
PCBS PATH                                   |DESCRIPTION                                |            |
GIA.Tower.SupSTR.VertEle.Core.CWall1.Cornr1 |           Corner                          |SubSubelement|
-------------------------------------------------------------------------------------------------------
   |ATTRIBUTE                                                                           |
   |INHERITED DESCRIPTION                      CLASS                TYPE         UNIT   |
   |             COND VALUE 1            VALUE 2                                        |
   |  N     Length                             Dimension Properties Quantitative ft     |
   |  N     Width                              Dimension Properties Quantitative ft     |
-------------------------------------------------------------------------------------------------------
   |PLANNED ATTRIBUTE VALUE                                                                  |
   |INHERITED DESCRIPTION                      CLASS                TYPE         UNIT   AGGREGATED|
   |  LOC RANGE     COND VALUE 1            VALUE 2                                          |
   |  N     Length                             Dimension Properties Quantitative ft        N    |
   |  GFL     -2        AND  EQ  1.5                                                         |
   |  3       -22       AND  EQ  1                                                           |
   |  N     Width                              Dimension Properties Quantitative ft        N    |
   |  GFL     -22       AND  EQ  1                                                           |
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
PHYSICAL COMPONENT BREAKDOWN STRUCTURE                                                  |Type        |
PCBS PATH                                   |DESCRIPTION                                |            |
GIA.Tower.SupSTR.VertEle.Core.CWall1.Opengl |           Openging                        |SubSubelement|
-------------------------------------------------------------------------------------------------------
   |ATTRIBUTE                                                                           |
   |INHERITED DESCRIPTION                      CLASS                TYPE         UNIT   |
   |             COND VALUE 1            VALUE 2                                        |
   |  N     Length                             Dimension Properties Quantitative ft     |
   |  N     Height                             Dimension Properties Quantitative ft     |
-------------------------------------------------------------------------------------------------------
   |PLANNED ATTRIBUTE VALUE                                                                  |
   |INHERITED DESCRIPTION                      CLASS                TYPE         UNIT   AGGREGATED|
   |  LOC RANGE     COND VALUE 1            VALUE 2                                          |
   |  N     Length                             Dimension Properties Quantitative ft        N    |
   |  3       -22       AND  EQ  3                                                           |
   |  23      -23       AND  EQ  3.25                                                        |
```

```
|  N    Height                              Dimension Properties Quantitative ft              N     |
|  3      -22      AND  EQ  8                                                                       |
|  23     -23      AND  EQ  10                                                                      |
----------------------------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------
PHYSICAL COMPONENT BREAKDOWN STRUCTURE                                                  |Type       |
PCBS PATH                               |DESCRIPTION                                    |           |
GIA.Tower.SupSTR.VertEle.ShWall.SWall7  |        Shear Wall D1                          |Subelement |
----------------------------------------------------------------------------------------------------
   |ATTRIBUTE                                                                          |
   |INHERITED DESCRIPTION                           CLASS               TYPE        UNIT |
   |           COND VALUE 1            VALUE 2                                         |
   |   Y    Formwork Quantity                       Area Properties     Quantitative ft2 |
   |   Y    Rebar Quantity                          Material Quantities Quantitative Tn  |
   |   Y    Concrete Quantity                       Area Properties     Quantitative yd3 |
   |   Y    Surface Area                            Area Properties     Quantitative ft2 |
   |   Y    Time Frame for Concreting               Duration Properties Quantitative hr  |
   |   Y    Time Frame for Rebar                    Duration Properties Quantitative hr  |
   |   Y    Time Frame for Formwork                 Duration Properties Quantitative hr  |
   |   Y    Shape                                   Physical Properties Linguistic       |
   |   Y    Slump Range                             Concrete Properties Quantitative in  |
   |   Y    Max. Size of Aggregate                  Concrete Properties Quantitative in  |
   |   Y    Rate of Pour                            Concrete Properties Quantitative ft  |
   |   Y    Length                                  Dimension Properties Quantitative ft |
   |   Y    Height                                  Dimension Properties Quantitative ft |
   |   Y    Width                                   Dimension Properties Quantitative ft |
   |   Y    Number of Elements                      General Properties  Quantitative No. |
----------------------------------------------------------------------------------------------------
   |PLANNED ATTRIBUTE VALUE                                                                     |
   |INHERITED DESCRIPTION                           CLASS               TYPE        UNIT    AGGREGATED|
   |   LOC RANGE   COND VALUE 1            VALUE 2                                                   |
   |   Y    Formwork Quantity                       Area Properties     Quantitative ft2        N    |
   |   3      -4       AND  EQ  175.50                                                               |
   |   5      -20      AND  EQ  354.60                                                               |
   |   21     -21      AND  EQ  351.36                                                               |
   |   Y    Rebar Quantity                          Material Quantities Quantitative Tn         N    |
   |   3      -4       AND  EQ  0.38                                                                 |
   |   5      -20      AND  EQ  0.77                                                                 |
   |   21     -21      AND  EQ  0.67                                                                 |
   |   Y    Concrete Quantity                       Area Properties     Quantitative yd3        N    |
   |   3      -4       AND  EQ  2.25                                                                 |
   |   5      -20      AND  EQ  4.55                                                                 |
   |   21     -21      AND  EQ  4.00                                                                 |
   |   Y    Surface Area                            Area Properties     Quantitative ft2        N    |
   |   Y    Time Frame for Concreting               Duration Properties Quantitative hr         N    |
   |   Y    Time Frame for Rebar                    Duration Properties Quantitative hr         N    |
   |   Y    Time Frame for Formwork                 Duration Properties Quantitative hr         N    |
   |   Y    Shape                                   Physical Properties Linguistic              N    |
   |   3      -21      AND  EQ  Rectangular                                                          |
   |   Y    Slump Range                             Concrete Properties Quantitative in         N    |
   |   Y    Max. Size of Aggregate                  Concrete Properties Quantitative in         N    |
   |   Y    Rate of Pour                            Concrete Properties Quantitative ft         N    |
   |   Y    Length                                  Dimension Properties Quantitative ft        N    |
   |   3      -4       AND  EQ  9                                                                    |
   |   5      -21      AND  EQ  9.1                                                                  |
   |   Y    Height                                  Dimension Properties Quantitative ft        N    |
   |   3      -21      AND  EQ  9                                                                    |
   |   Y    Width                                   Dimension Properties Quantitative ft        N    |
   |   3      -4       AND  EQ  0.75                                                                 |
   |   5      -20      AND  EQ  0.66                                                                 |
```

```
|  21    -21      AND  EQ  0.66                                                                    |
|  Y     Number of Elements                   General Properties   Quantitative No.            N   |
|  3     -21      AND  EQ  1                                                                       |
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
PHYSICAL COMPONENT BREAKDOWN STRUCTURE                                                  |Type      |
PCBS PATH                                |DESCRIPTION                                   |          |
GIA.Tower.SupSTR.HoriEle.Slab.SlBay9     |          SlabBay A1                          |Subelement|
----------------------------------------------------------------------------------------------------
  |ATTRIBUTE                                                                            |
  |INHERITED DESCRIPTION                      CLASS                TYPE         UNIT    |
  |             COND VALUE 1           VALUE 2                                          |
  |   Y     Formwork Quantity                 Area Properties      Quantitative ft2     |
  |   Y     Rebar Quantity                    Material Quantities  Quantitative Tn      |
  |   Y     Concrete Quantity                 Area Properties      Quantitative yd3     |
  |   Y     Surface Area                      Area Properties      Quantitative ft2     |
  |   Y     Time Frame for Concreting         Duration Properties  Quantitative hr      |
  |   Y     Time Frame for Rebar              Duration Properties  Quantitative hr      |
  |   Y     Time Frame for Formwork           Duration Properties  Quantitative hr      |
  |   Y     Shape                             Physical Properties  Linguistic           |
  |   Y     Slump Range                       Concrete Properties  Quantitative in      |
  |   Y     Max. Size of Aggregate            Concrete Properties  Quantitative in      |
  |   Y     Number of Elements                General Properties   Quantitative No.     |
  |   Y     Length                            Dimension Properties Quantitative ft      |
  |   Y     Width                             Dimension Properties Quantitative ft      |
  |   Y     Thickness                         Dimension Properties Quantitative ft      |
  |   Y     Horizontal Distance               Material Quantities  Quantitative ft      |
  |   Y     Vertical Distance                 Material Quantities  Quantitative ft      |
  |   Y     Storey Height                     Floor Loc Properties Quantitative ft      |
  |   Y     Min. Width                        Dimension Properties Quantitative ft      |
  |   Y     SlabBay Support is Uniform        Tech. Specifications Boolean              |
  |   Y     SlabBay Supporting Sides are Parallel  Tech. Specifications Boolean         |
----------------------------------------------------------------------------------------------------
  |PLANNED ATTRIBUTE VALUE                                                                 |
  |INHERITED DESCRIPTION                      CLASS                TYPE         UNIT   AGGREGATED|
  |   LOC RANGE    COND VALUE 1          VALUE 2                                           |
  |   Y     Formwork Quantity                 Area Properties      Quantitative ft2        N   |
  |   2      -2       AND  EQ  640                                                         |
  |   3      -3       AND  EQ  656                                                         |
  |   4      -21      AND  EQ  533                                                         |
  |   22     -22      AND  EQ  295.63                                                      |
  |   23     -23      AND  EQ  301                                                         |
  |   Y     Rebar Quantity                    Material Quantities  Quantitative Tn         N   |
  |   2      -2       AND  EQ  2.64                                                        |
  |   3      -3       AND  EQ  2.39                                                        |
  |   4      -21      AND  EQ  1.94                                                        |
  |   22     -22      AND  EQ  1.08                                                        |
  |   23     -23      AND  EQ  1.10                                                        |
  |   Y     Concrete Quantity                 Area Properties      Quantitative yd3        N   |
  |   2      -2       AND  EQ  15.64                                                       |
  |   3      -3       AND  EQ  14.17                                                       |
  |   4      -21      AND  EQ  11.51                                                       |
  |   22     -22      AND  EQ  6.39                                                        |
  |   23     -23      AND  EQ  6.50                                                        |
  |   Y     Surface Area                      Area Properties      Quantitative ft2        N   |
  |   Y     Time Frame for Concreting         Duration Properties  Quantitative hr         N   |
  |   Y     Time Frame for Rebar              Duration Properties  Quantitative hr         N   |
  |   Y     Time Frame for Formwork           Duration Properties  Quantitative hr         N   |
  |   Y     Shape                             Physical Properties  Linguistic              N   |
  |   3      -23      AND  EQ  Rectangular                                                 |
```

```
|   Y     Slump Range                             Concrete Properties  Quantitative in                N    |
|   Y     Max. Size of Aggregate                  Concrete Properties  Quantitative in                N    |
|   Y     Number of Elements                      General Properties   Quantitative No.               N    |
|   3       -23       AND  EQ  1                                                                           |
|   Y     Length                                  Dimension Properties Quantitative ft                N    |
|   3       -3        AND  EQ  32                                                                          |
|   4       -21       AND  EQ  26                                                                          |
|   22      -23       AND  EQ  21.5                                                                        |
|   Y     Width                                   Dimension Properties Quantitative ft                N    |
|   3       -3        AND  EQ  20.5                                                                        |
|   4       -21       AND  EQ  20.5                                                                        |
|   22      -22       AND  EQ  13.75                                                                       |
|   23      -23       AND  EQ  14                                                                          |
|   Y     Thickness                               Dimension Properties Quantitative ft                N    |
|   3       -23       AND  EQ  0.5833                                                                      |
|   Y     Horizontal Distance                     Material Quantities  Quantitative ft                N    |
|   Y     Vertical Distance                       Material Quantities  Quantitative ft                N    |
|   Y     Storey Height                           Floor Loc Properties Quantitative ft                N    |
|   3       -21       AND  EQ  9                                                                           |
|   2       -2        AND  EQ  12                                                                          |
|   22      -23       AND  EQ  10                                                                          |
|   Y     Min. Width                              Dimension Properties Quantitative ft                N    |
|   2       -2        AND  EQ  14                                                                          |
|   3       -3        AND  EQ  16                                                                          |
|   4       -21       AND  EQ  6.25                                                                        |
|   22      -22       AND  EQ  13.75                                                                       |
|   23      -23       AND  EQ  14                                                                          |
|   Y     SlabBay Support is Uniform              Tech. Specifications Boolean                        N    |
|   3       -23       AND  EQ  False                                                                       |
|   Y     SlabBay Supporting Sides are Parallel   Tech. Specifications Boolean                        N    |
|   3       -23       AND  EQ  False                                                                       |
-----------------------------------------------------------------------------------------------------------

-----------------------------------------------------------------------------------------------------------
PHYSICAL COMPONENT BREAKDOWN STRUCTURE                                                         |Type       |
PCBS PATH                                    |DESCRIPTION                                      |           |
GIA.Tower.SupSTR.HoriEle.Slab.SlBay9.SlBand1|           Slabband                             |SubSubelement|
-----------------------------------------------------------------------------------------------------------
    |ATTRIBUTE                                                                                 |
    |INHERITED DESCRIPTION                         CLASS                TYPE         UNIT       |
    |             COND VALUE 1            VALUE 2                                               |
    |   N     Width                                Dimension Properties Quantitative ft         |
    |   N     Depth                                Dimension Properties Quantitative ft         |
    -------------------------------------------------------------------------------------------------------
    |PLANNED ATTRIBUTE VALUE                                                                           |
    |INHERITED DESCRIPTION                         CLASS                TYPE         UNIT     AGGREGATED|
    |   LOC RANGE     COND VALUE 1            VALUE 2                                                  |
    |   N     Width                                Dimension Properties Quantitative ft           N    |
    |   2       -2        AND  WR  4                7.5                                                |
    |   N     Depth                                Dimension Properties Quantitative ft           N    |
    |   2       -2        AND  WR  1.5              2.5                                                |
-----------------------------------------------------------------------------------------------------------
```

Type, Para/ Con

Report Date: 28OCT02
Report Time: 11:22:30pm

Select: Selectively

Template: Construction of typical floor of a High-rise

PHYSICAL COMPONENT BREAKDOWN STRUCTURE M&RBS PATH	DESCRIPTION	Type	INHERITED	PARAMETER/CONDITION DESCRIPTION	COND	VALUE 1	VALUE 2	P/C	CLASS	TYPE	UNIT
ROOT	High-rise Superstructure Construction	Method Statement									
ROOT.FormCol	Formwork for Columns	Operation									
ROOT.FormCol.WGang	Wooden Gang Formwork	Method	N	Rate of Production	AND EQ	35		P	Production Data	Quantitative	sfth
			N	Min. Reuse Required	AND EQ	30		C	Tech. Feasibility	Quantitative	No.
			N	Storage Space Length Required	AND EQ	50		C	Tech. Feasibility	Quantitative	ft
			N	Storage Space Width Required	AND EQ	30		C	Tech. Feasibility	Quantitative	ft
			N	Allowable Rate of Pour	AND EQ	8		C	Tech. Feasibility	Quantitative	fthr
			N	Allowable Tie Spacing	AND WR	2	3	C	Tech. Feasibility	Quantitative	ft
ROOT.FormCol.WGang.WGC	Wooden Gangform for Column	Resource	Y	Rate of Production	AND EQ	35		P	Production Data	Quantitative	sfth
			Y	Min. Reuse Required	AND EQ	30		C	Tech. Feasibility	Quantitative	No.
			Y	Storage Space Length Required	AND EQ	50		C	Tech. Feasibility	Quantitative	ft
			Y	Storage Space Width Required	AND EQ	30		C	Tech. Feasibility	Quantitative	ft
			Y	Allowable Rate of Pour	AND EQ	8		C	Tech. Feasibility	Quantitative	fthr
			Y	Allowable Tie Spacing	AND WR	2	3	C	Tech. Feasibility	Quantitative	ft
ROOT.FormCol.WGang.FCrew	Formwork Crew	Resource	N	Number of Crew Members	AND EQ	7		P	Labor	Quantitative	No.
ROOT.RebarCol	Construction of typical floor of a High-rise	Operation									
ROOT.RebarCol.PreFab	Rebar Prefabrication	Method	N	Rate of Production	AND EQ	0.125		P	Production Data	Quantitative	Tnh
			N	Rebar Site Storage Length Required	AND EQ	60		C	Tech. Feasibility	Quantitative	ft
			N	Rebar Fabrication Site Length Required	AND EQ	60		C	Tech. Feasibility	Quantitative	ft
			N	Rebar Fabrication Site Width Required	AND EQ	20		C	Tech. Feasibility	Quantitative	ft

PHYSICAL COMPONENT BREAKDOWN STRUCTURE M&RBS PATH	DESCRIPTION	Type	PARAMETER/CONDITION INHERITED	DESCRIPTION COND VALUE 1 VALUE 2	P/C	CLASS	TYPE	UNIT
			N	Rebar Fabrication Site Area Required AND EQ 1200	C	Tech. Feasibility	Quantitative	ft2
ROOT.RebarCol.PreFab.RCrew	Rebar Crew	Resource	N	Number of Crew Members AND EQ 8	P	Labor	Quantitative	No.
ROOT.ConcCol	Concrete placing for Columns	Operation						
ROOT.ConcCol.CrBuck	Concrete Placing with Crane & Bucket	Method	N	Rate of Concrete Placement AND EQ 45	P	Production Data	Quantitative	yd3/hr
			N	Parking Space Length Required AND EQ 38	C	Tech. Feasibility	Quantitative	ft
			N	Parking Space Width Required AND EQ 15	C	Tech. Feasibility	Quantitative	ft
			N	Parking Space Area Required AND EQ 570	C	Tech. Feasibility	Quantitative	ft2
			N	Max. Size of Aggregate AND EQ 4	C	Tech. Feasibility	Quantitative	in
ROOT.ConcCol.CrBuck.Crane	Tower Crane Peiner Hammerhead Tower Crane	Resource	Y	Rate of Concrete Placement AND EQ 45	P	Production Data	Quantitative	yd3/hr
			N	Max. Hook height AND EQ 246.75	P	Manufacturers Specs	Quantitative	ft
			N	Horizontal hook speed AND EQ 290	P	Manufacturers Specs	Quantitative	
			N	Vertical Speed AND EQ 90	P	Manufacturers Specs	Quantitative	
			N	Boom length AND EQ 229.5	P	Manufacturers Specs	Quantitative	ft
			N	Max.Weight AND EQ 17600	P	Manufacturers Specs	Quantitative	lb
			N	Max.Weight at boom tip AND EQ 6800	P	Manufacturers Specs	Quantitative	lb
			Y	Parking Space Length Required AND EQ 38	C	Tech. Feasibility	Quantitative	ft
			Y	Parking Space Width Required AND EQ 15	C	Tech. Feasibility	Quantitative	ft
			Y	Parking Space Area Required AND EQ 570	C	Tech. Feasibility	Quantitative	ft2
			Y	Max. Size of Aggregate AND EQ 4	C	Tech. Feasibility	Quantitative	in
ROOT.ConcCol.CrBuck.Bucket	Concrete Bucket - Upright	Resource	N	Concrete Capacity AND EQ 4	P	Manufacturers Specs	Quantitative	yd3
			N	Loading Height AND EQ 80	P	Manufacturers Specs	Quantitative	in
			N	Outside diameter AND EQ 72	P	Manufacturers Specs	Quantitative	in
			N	Inside diameter AND EQ 68	P	Manufacturers Specs	Quantitative	in
			N	Weight AND EQ 570	P	Manufacturers Specs	Quantitative	lb
ROOT.ConcCol.CrBuck.CCrew	Crane and Bucket concrete placement crew	Resource	N	Number of crew members AND EQ 8	P		Quantitative	No.

PHYSICAL COMPONENT BREAKDOWN STRUCTURE M&RBS PATH	DESCRIPTION	Type	PARAMETER/CONDITION INHERITED	DESCRIPTION	COND	VALUE 1	VALUE 2	P/C CLASS	TYPE	UNIT
ROOT.FormWall	Formwork for Walls	Operation								
ROOT.FormWall.WGang	Wooden Gang Formwork	Method	N	Rate of Production				P Production Data	Quantitative	sfth
				AND	EQ	35				
			N	Min. Reuse Required				C Tech. Feasibility	Quantitative	No.
				AND	EQ	30				
			N	Storage Space Length Required				C Tech. Feasibility	Quantitative	ft
				AND	EQ	50				
			N	Storage Space Width Required				C Tech. Feasibility	Quantitative	ft
				AND	EQ	30				
			N	Allowable Rate of Pour				C Tech. Feasibility	Quantitative	fthr
				AND	EQ	8				
			N	Allowable Tie Spacing				C Tech. Feasibility	Quantitative	ft
				AND	WR	2	3			
ROOT.FormWall.WGang.WGW	Wooden Gang Wallform for Wall	Resource	Y	Rate of Production				P Production Data	Quantitative	sfth
				AND	EQ	35				
			Y	Min. Reuse Required				C Tech. Feasibility	Quantitative	No.
				AND	EQ	30				
			Y	Storage Space Length Required				C Tech. Feasibility	Quantitative	ft
				AND	EQ	50				
			Y	Storage Space Width Required				C Tech. Feasibility	Quantitative	ft
				AND	EQ	30				
			Y	Allowable Rate of Pour				C Tech. Feasibility	Quantitative	fthr
				AND	EQ	8				
			Y	Allowable Tie Spacing				C Tech. Feasibility	Quantitative	ft
				AND	WR	2	3			
ROOT.FormWall.WGang.FCrew	Formwork Crew	Resource	N	Number of Crew Members				P Labor	Quantitative	No.
				AND	EQ	7				
ROOT.RebarWall	Rebar placing for Walls	Operation								
ROOT.RebarWall.PPreFab	Partial Rebar Prefabrication	Method	N	Rate of Production				P Production Data	Quantitative	Tnh
				AND	EQ	0.1				
			N	Rebar Site Storage Length required				C Tech. Feasibility	Quantitative	ft
				AND	EQ	60				
			N	Rebar Site Storage Width Required				C Tech. Feasibility	Quantitative	ft
				AND	EQ	12				
			N	Rebar Site Storage Area Required				C Tech. Feasibility	Quantitative	ft2
				AND	EQ	720				
			N	Rebar Fabrication Site Length Required				C Tech. Feasibility	Quantitative	ft
				AND	EQ	72				
			N	Rebar Fabrication Site Width Required				C Tech. Feasibility	Quantitative	ft
				AND	EQ	20				
			N	Rebar Fabrication Site Area Required				C Tech. Feasibility	Quantitative	ft2
				AND	EQ	1440				
ROOT.RebarWall.PPreFab.RCrew	Rebar Crew	Resource	N	Number of Crew Members				P Labor	Quantitative	No.
				AND	EQ	8				
ROOT.ConcWall	Concrete placement for Walls	Operation								
ROOT.ConcWall.CrBuck	Concrete Placing with Crane & Bucket	Method	N	Rate of Concrete Placement				P Production Data	Quantitative	yd3/hr
				AND	EQ	45				
			N	Parking Space Length Required				C Tech. Feasibility	Quantitative	ft

PHYSICAL COMPONENT BREAKDOWN STRUCTURE M&RBS PATH	DESCRIPTION	Type	PARAMETER/CONDITION INHERITED	DESCRIPTION	COND VALUE 1	VALUE 2	P/C	CLASS	TYPE	UNIT
					AND EQ 38					
			N	Parking Space Width Required	AND EQ 15		C	Tech. Feasibility	Quantitative	ft
			N	Parking Space Area Required	AND EQ 570		C	Tech. Feasibility	Quantitative	ft2
			N	Max. Size of Aggregate	AND EQ 4		C	Tech. Feasibility	Quantitative	in
ROOT.ConcWall.CrBuck.Crane	Tower Crane Peiner Hammerhead Tower Crane	Resource	Y	Rate of Concrete Placement	AND EQ 45		P	Production Data	Quantitative	yd3/hr
			N	Max. Hook height	AND EQ 246.75		P	Manufacturers Specs	Quantitative	ft
			N	Horizontal hook speed	AND EQ 290		P	Manufacturers Specs	Quantitative	
			N	Vertical Speed	AND EQ 90		P	Manufacturers Specs	Quantitative	
			N	Boom length	AND EQ 229.5		P	Manufacturers Specs	Quantitative	ft
			N	Max.Weight	AND EQ 17600		P	Manufacturers Specs	Quantitative	lb
			N	Max.Weight at boom tip	AND EQ 6800		P	Manufacturers Specs	Quantitative	lb
			Y	Parking Space Length Required	AND EQ 38		C	Tech. Feasibility	Quantitative	ft
			Y	Parking Space Width Required	AND EQ 15		C	Tech. Feasibility	Quantitative	ft
			Y	Parking Space Area Required	AND EQ 570		C	Tech. Feasibility	Quantitative	ft2
			Y	Max. Size of Aggregate	AND EQ 4		C	Tech. Feasibility	Quantitative	in
ROOT.ConcWall.CrBuck.Bucket	Concrete Bucket - Upright	Resource	N	Concrete Capacity	AND EQ 4		P	Manufacturers Specs	Quantitative	yd3
			N	Loading Height	AND EQ 80		P	Manufacturers Specs	Quantitative	in
			N	Outside diameter	AND EQ 72		P	Manufacturers Specs	Quantitative	in
			N	Inside diameter	AND EQ 68		P	Manufacturers Specs	Quantitative	in
			N	Weight	AND EQ 570		P	Manufacturers Specs	Quantitative	lb
ROOT.ConcWall.CrBuck.CCrew	Crane and Bucket concrete placement crew	Resource	N	Number of crew members	AND EQ 8		P		Quantitative	No.
ROOT.FormCore	Formwork for Core	Operation								
ROOT.FormCore.AJumpFm	Aluminum Waler Jumpform	Method	N	Rate of Production	AND EQ 35		P	Production Data	Quantitative	sfth
			N	Min. Reuse Required	AND EQ 30		C	Tech. Feasibility	Quantitative	No.
			N	Storage Space Length Required	AND EQ 50		C	Tech. Feasibility	Quantitative	ft
			N	Storage Space Width Required	AND EQ 30		C	Tech. Feasibility	Quantitative	ft
			N	Allowable Rate of Pour			C	Tech. Feasibility	Quantitative	fthr

PHYSICAL COMPONENT BREAKDOWN STRUCTURE

M&RBS PATH	DESCRIPTION	Type	INHERITED	PARAMETER/CONDITION DESCRIPTION / COND VALUE 1	VALUE 2	P/C	CLASS	TYPE	UNIT
				AND EQ 9					
			N	Allowable Tie Spacing AND WR 6	8	C	Tech. Feasibility	Quantitative	ft
			N	Length Range AND WR 8	44	C	Tech. Feasibility	Quantitative	ft
			N	Width Range AND WR 8	16	C	Tech. Feasibility	Quantitative	ft
ROOT.FormCore.AJumpFm.JumpFm	Aluminum Waler Jumpform	Resource	Y	Rate of Production AND EQ 35		P	Production Data	Quantitative	sfth
			Y	Min. Reuse Required AND EQ 30		C	Tech. Feasibility	Quantitative	No.
			Y	Storage Space Length Required AND EQ 50		C	Tech. Feasibility	Quantitative	ft
			Y	Storage Space Width Required AND EQ 30		C	Tech. Feasibility	Quantitative	ft
			Y	Allowable Rate of Pour AND EQ 9		C	Tech. Feasibility	Quantitative	fthr
			Y	Allowable Tie Spacing AND WR 6	8	C	Tech. Feasibility	Quantitative	ft
			Y	Length Range AND WR 8	44	C	Tech. Feasibility	Quantitative	ft
			Y	Width Range AND WR 8	16	C	Tech. Feasibility	Quantitative	ft
ROOT.FormCore.AJumpFm.FCrew	Formwork Crew	Resource	N	Number of Crew Members AND EQ 7		P	Labor	Quantitative	No.
ROOT.RebarCore	Rebar placement for Core	Operation							
ROOT.RebarCore.PPreFab	Partial Rebar Prefabrication	Method	N	Rate of Production AND EQ 0.1		P	Production Data	Quantitative	Tnh
			N	Rebar Site Storage Length required AND EQ 60		C	Tech. Feasibility	Quantitative	ft
			N	Rebar Site Storage Width Required AND EQ 12		C	Tech. Feasibility	Quantitative	ft
			N	Rebar Site Storage Area Required AND EQ 720		C	Tech. Feasibility	Quantitative	ft2
			N	Rebar Fabrication Site Length Required AND EQ 72		C	Tech. Feasibility	Quantitative	ft
			N	Rebar Fabrication Site Width Required AND EQ 20		C	Tech. Feasibility	Quantitative	ft
			N	Rebar Fabrication Site Area Required AND EQ 1440		C	Tech. Feasibility	Quantitative	ft2
ROOT.RebarCore.PPreFab.RCrew	Rebar Crew	Resource	N	Number of Crew Members AND EQ 8		P	Labor	Quantitative	No.
ROOT.ConcCore	Concrete placement for Core	Operation							
ROOT.ConcCore.CrBuck	Concrete Placing with Crane & Bucket	Method	N	Rate of Concrete Placement AND EQ 45		P	Production Data	Quantitative	yd3/hr
			N	Parking Space Length Required AND EQ 38		C	Tech. Feasibility	Quantitative	ft
			N	Parking Space Width Required AND EQ 15		C	Tech. Feasibility	Quantitative	ft

PHYSICAL COMPONENT BREAKDOWN STRUCTURE

M&RBS PATH	DESCRIPTION	Type	INHERITED	PARAMETER/CONDITION DESCRIPTION COND VALUE 1	VALUE 2	P/C	CLASS	TYPE	UNIT
			N	Parking Space Area Required AND EQ 570		C	Tech. Feasibility	Quantitative	ft2
			N	Max. Size of Aggregate AND EQ 4		C	Tech. Feasibility	Quantitative	in
ROOT.ConcCore.CrBuck.Crane	Tower Crane Peiner Hammerhead Tower Crane	Resource	Y	Rate of Concrete Placement AND EQ 45		P	Production Data	Quantitative	yd3/hr
			N	Max. Hook height AND EQ 246.75		P	Manufacturers Specs	Quantitative	ft
			N	Horizontal hook speed AND EQ 290		P	Manufacturers Specs	Quantitative	
			N	Vertical Speed AND EQ 90		P	Manufacturers Specs	Quantitative	
			N	Boom length AND EQ 229.5		P	Manufacturers Specs	Quantitative	ft
			N	Max.Weight AND EQ 17600		P	Manufacturers Specs	Quantitative	lb
			N	Max.Weight at boom tip AND EQ 6800		P	Manufacturers Specs	Quantitative	lb
			Y	Parking Space Length Required AND EQ 38		C	Tech. Feasibility	Quantitative	ft
			Y	Parking Space Width Required AND EQ 15		C	Tech. Feasibility	Quantitative	ft
			Y	Parking Space Area Required AND EQ 570		C	Tech. Feasibility	Quantitative	ft2
			Y	Max. Size of Aggregate AND EQ 4		C	Tech. Feasibility	Quantitative	in
ROOT.ConcCore.CrBuck.Bucket	Concrete Bucket - Upright	Resource	N	Concrete Capacity AND EQ 4		P	Manufacturers Specs	Quantitative	yd3
			N	Loading Height AND EQ 80		P	Manufacturers Specs	Quantitative	in
			N	Outside diameter AND EQ 72		P	Manufacturers Specs	Quantitative	in
			N	Inside diameter AND EQ 68		P	Manufacturers Specs	Quantitative	in
			N	Weight AND EQ 570		P	Manufacturers Specs	Quantitative	lb
ROOT.ConcCore.CrBuck.CCrew	Crane and Bucket concrete placement crew	Resource	N	Number of crew members AND EQ 8		P		Quantitative	No.
ROOT.FormSlab	Formwork for Slab	Operation							
ROOT.FormSlab.FlTruss	Flying Truss Formwork for Slab	Method	N	Rate of Production AND EQ 70		P	Production Data	Quantitative	fthr
			N	Min. Reuse Required AND EQ 6		C	Production Data	Quantitative	No.
			N	Site Assembly Space Length Required AND EQ 40		C	Tech. Feasibility	Quantitative	ft
			N	Site Assembly Space Width Required AND EQ 30		C	Tech. Feasibility	Quantitative	ft
			N	Min. Assembly Space Area Required AND EQ 1200		C	Tech. Feasibility	Quantitative	ft2
			N	Economical Length of SlabBay AND EQ 22		C	Designer's Spec	Quantitative	ft

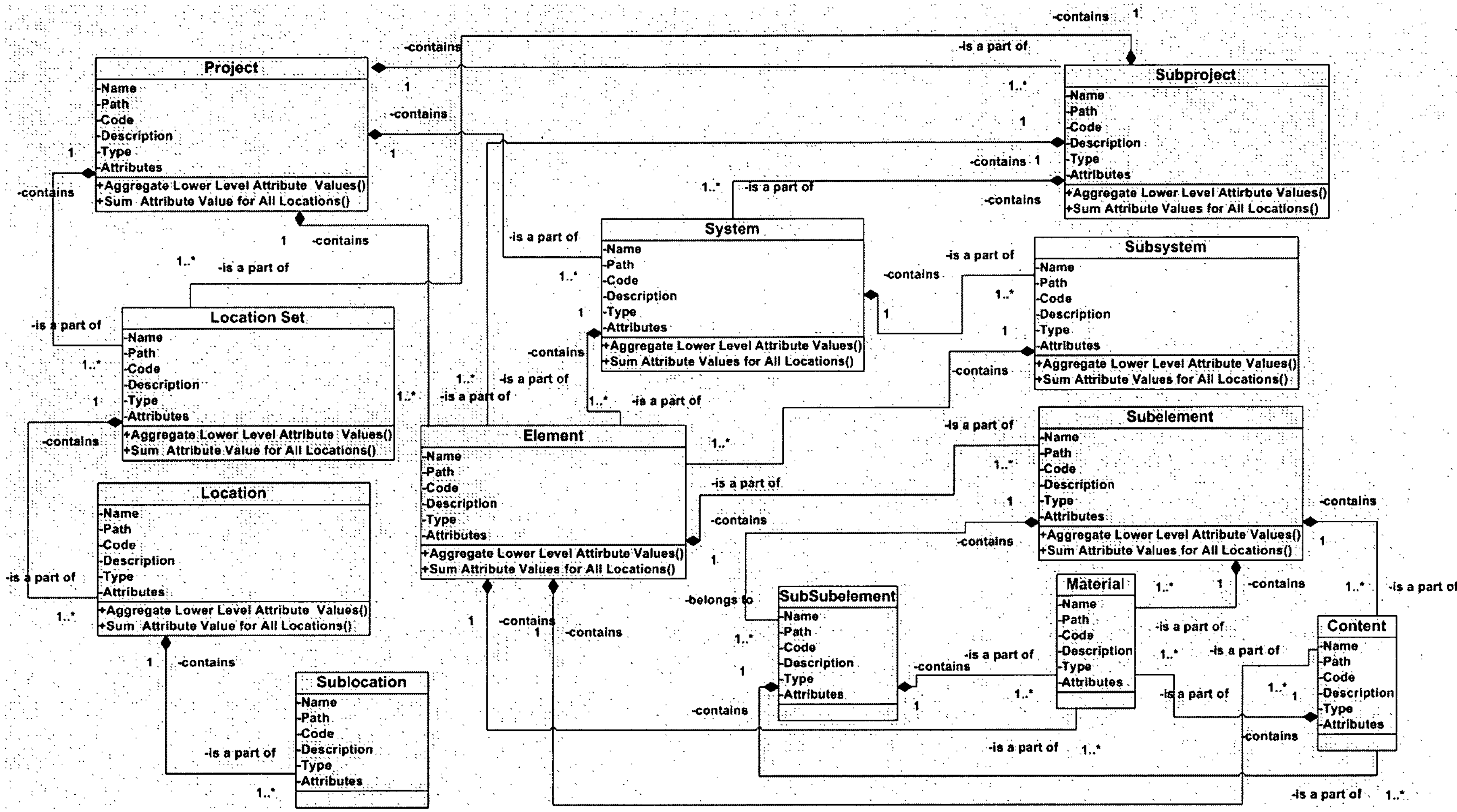
PHYSICAL COMPONENT BREAKDOWN STRUCTURE M&RBS PATH	DESCRIPTION	Type	PARAMETER/CONDITION INHERITED	DESCRIPTION COND VALUE 1	VALUE 2	P/C	CLASS	TYPE	UNIT
			N	Economical Width of SlabBay AND WR 15	30	C	Designer's Spec	Quantitative	ft
			N	Storey Height Range AND WR 7	20	C	Designer's Spec	Quantitative	ft
ROOT.FormSlab.FlTruss.FTruss	Flying Truss Formwork for Slab	Resource	Y	Rate of Production AND EQ 70		P	Production Data	Quantitative	fthr
			Y	Min. Reuse Required AND EQ 6		C	Production Data	Quantitative	No.
			Y	Site Assembly Space Length Required AND EQ 40		C	Tech. Feasibility	Quantitative	ft
			Y	Site Assembly Space Width Required AND EQ 30		C	Tech. Feasibility	Quantitative	ft
			Y	Min. Assembly Space Area Required AND EQ 1200		C	Tech. Feasibility	Quantitative	ft2
			Y	Economical Length of SlabBay AND EQ 22		C	Designer's Spec	Quantitative	ft
			Y	Economical Width of SlabBay AND WR 15	30	C	Designer's Spec	Quantitative	ft
			Y	Storey Height Range AND WR 7	20	C	Designer's Spec	Quantitative	ft
ROOT.FormSlab.FlTruss.Crane	Piener Hammerhead Tower Crane	Resource	N	Max. Hook Height AND EQ 245.75		P	Manufacturers Specs	Quantitative	ft
			N	Horizontal Hook Speed AND EQ 290		P	Manufacturers Specs	Quantitative	
			N	Vertical Speed AND EQ 90		P	Manufacturers Specs	Quantitative	
			N	Boom Length AND EQ 229.5		P	Manufacturers Specs	Quantitative	ft
			N	Max. Weight AND EQ 17600		P	Manufacturers Specs	Quantitative	lb
			N	Max. Weight at Boom Tip AND EQ 6800		P	Manufacturers Specs	Quantitative	lb
ROOT.FormSlab.FlTruss.FCrew	Formwork Crew	Resource	N	Number of Crew Members AND EQ 7		P	Labor	Quantitative	No.
ROOT.RebarSlab	Rebar Placement for Slab	Operation							
ROOT.RebarSlab.ReAsm	Rebar Assembly	Method	N	Rate of Production AND EQ 0.166		P	Production Data	Quantitative	Tnh
			N	Rebar Site Storage Length Required AND EQ 60		C	Tech. Feasibility	Quantitative	ft
			N	Rebar Site Storage Width Required AND EQ 12		C	Tech. Feasibility	Quantitative	ft
			N	Rebar Site Storage Area Required AND EQ 720		C	Tech. Feasibility	Quantitative	ft2
ROOT.RebarSlab.ReAsm.RCrew	Rebar Crew	Resource	N	Number of Crew Members AND EQ 8		P	Labor	Quantitative	No.
ROOT.ConcSlab	Concrete placement for Slab	Operation							
ROOT.ConcSlab.CrBuck	Concrete Placing with Crane & Bucket	Method	N	Rate of Concrete Placement AND EQ 45		P	Production Data	Quantitative	yd3/hr

PHYSICAL COMPONENT BREAKDOWN STRUCTURE

M&RBS PATH	DESCRIPTION	Type	INHERITED	PARAMETER/CONDITION DESCRIPTION (COND VALUE 1 / VALUE 2)	P/C	CLASS	TYPE	UNIT
			N	Parking Space Length Required AND EQ 38	C	Tech. Feasibility	Quantitative	ft
			N	Parking Space Width Required AND EQ 15	C	Tech. Feasibility	Quantitative	ft
			N	Parking Space Area Required AND EQ 570	C	Tech. Feasibility	Quantitative	ft2
			N	Max. Size of Aggregate AND EQ 4	C	Tech. Feasibility	Quantitative	in
ROOT.ConcSlab.CrBuck.Crane	Tower Crane Peiner Hammerhead Tower Crane	Resource	Y	Rate of Concrete Placement AND EQ 45	P	Production Data	Quantitative	yd3/hr
			N	Max. Hook height AND EQ 246.75	P	Manufacturers Specs	Quantitative	ft
			N	Horizontal hook speed AND EQ 290	P	Manufacturers Specs	Quantitative	
			N	Vertical Speed AND EQ 90	P	Manufacturers Specs	Quantitative	
			N	Boom length AND EQ 229.5	P	Manufacturers Specs	Quantitative	ft
			N	Max.Weight AND EQ 17600	P	Manufacturers Specs	Quantitative	lb
			N	Max.Weight at boom tip AND EQ 6800	P	Manufacturers Specs	Quantitative	lb
			Y	Parking Space Length Required AND EQ 38	C	Tech. Feasibility	Quantitative	ft
			Y	Parking Space Width Required AND EQ 15	C	Tech. Feasibility	Quantitative	ft
			Y	Parking Space Area Required AND EQ 570	C	Tech. Feasibility	Quantitative	ft2
			Y	Max. Size of Aggregate AND EQ 4	C	Tech. Feasibility	Quantitative	in
ROOT.ConcSlab.CrBuck.Bucket	Concrete Bucket - Upright	Resource	N	Concrete Capacity AND EQ 4	P	Manufacturers Specs	Quantitative	yd3
			N	Loading Height AND EQ 80	P	Manufacturers Specs	Quantitative	in
			N	Outside diameter AND EQ 72	P	Manufacturers Specs	Quantitative	in
			N	Inside diameter AND EQ 68	P	Manufacturers Specs	Quantitative	in
			N	Weight AND EQ 570	P	Manufacturers Specs	Quantitative	lb
ROOT.ConcSlab.CrBuck.CCrew	Crane and Bucket concrete placement crew	Resource	N	Number of crew members AND EQ 8	P		Quantitative	No.

APPENDIX – G

UML Static Structure Diagram for PCBS Hierarchy



UML Static Structure Diagram for the PCBS Hierarchy