

**AUTOMATED EXTRACTION AND QUERYING OF  
CONSTRUCTION-SPECIFIC DESIGN FEATURES FROM A  
BUILDING INFORMATION MODEL**

by

MADHAV PRASAD NEPAL

B.Eng., Tribhuvan University, Nepal, 1996

M.Eng., Asian Institute of Technology, Thailand, 2001

M.Sc., National University of Singapore, Singapore, 2004

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE STUDIES

(Civil Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

October 2011

© Madhav Prasad Nepal, 2011

## Abstract

In recent years, several research and industry efforts have focused on developing building information models (BIMs) to support various aspects of the architectural, engineering, construction and facility management (AEC/FM) industry. BIMs provide semantically-rich information models that explicitly represent both 3D geometric and non-geometric information. While BIMs have many useful applications to the construction industry, there are enormous challenges in getting construction-specific information out of BIMs, limiting the usability of these models. This research addresses this problem by developing a novel approach to extract construction features from a given BIM and support the processing of user-driven queries on a BIM.

In this dissertation, we formalized:

- An *ontology of design features* that explicitly represents design conditions that are relevant to construction practitioners and supports the generation of a construction-specific feature-based model;
- A *query specification vocabulary* which characterizes spatial and non-spatial queries, and developed query templates to guide non-expert BIM users to specify queries; and
- An *integrated approach* that combines model-based reasoning and query-based approach to automatically extract design features to create a project-specific feature-based model (FBM) and provide support for answering queries on the FBM.

The construction knowledge formalized in this research was gathered from a variety of sources, which included a detailed literature review, several case studies, extensive observations of design and construction meetings, and lengthy discussions with different construction practitioners. We used three different tests to validate the research contributions. We conducted semi-structured, informal interviews with four construction experts for the four building projects studied to validate the content, representativeness and the generality of the concepts formalized in this research. We conducted retrospective analysis for different features to evaluate the soundness of our research in comparison with the state-of-the-art

tools. Finally, we performed descriptive and interpretive analysis to demonstrate that our approach is capable of providing richer, insightful and useful construction information.

This research can help to make a BIM more accessible for construction users. The developed solutions can support decision making in a variety of construction management functions, such as cost estimating, construction planning, execution and coordination, purchasing, constructability analysis, methods selection, and productivity analysis.

## Preface

This Ph.D. research is a part of a collaboration of the ARTIFACT (Advanced Research, Techniques, and Informatics for Future Advantages in Construction Technology) Project, which is a joint effort between civil engineers and computer scientists at the University of British Columbia (UBC) and the University of Victoria.

The author of this thesis is solely responsible for writing this manuscript, the literature review, research design, case studies, the development of ontology and query specifications, validation and evaluation studies, and the analysis and interpretation of the results. Some of the interviews and meetings with experts were jointly conducted with the help of Dr. Sheryl Staub-French and Dr. Rachel Pottinger.

The prototype implementations related to this research were implemented by two Master's students, Jiemin Zhang and April Webster in the Computer Science Department, at UBC. Yun Lou, an undergraduate student in the Computer Science Department contributed in developing the user interfaces.

Two peer-reviewed conference papers related to this thesis have been published. The coauthors of these papers provided the guidance and contributed to write the implementation and user interface parts.

- i. Nepal, M. P., Staub-French, S., Zhang, J., Lawrence, M. and Pottinger, R. (2008). Deriving construction features from an IFC model. *Proceedings of the CSCE 2008 Annual Congress*, Quebec City, Canada, 11 pages.
- ii. Nepal, M. P., Zhang, J., Webster, A., Staub-French, S., Pottinger, R., and Lawrence, M. (2009). Querying IFC-based building information models to support construction management functions. *Proceedings of the 2009 Construction Research Congress, Building a Sustainable Future*, Seattle, WA, pp. 506-515.

Additionally, two journal papers related to this research have also been published and are listed below. The first paper describes the ontology formalized to characterize component similarity and the generic reasoning process used to evaluate the degree of similarity of building components in a given product model. The second paper, written by our Artifact Project team, examines the usability of standard schemas focusing on XML data models and

describes the use of semantic modeling systems to automate the knowledge extraction process using ifcXML schemas.

- a. Staub-French, S., and Nepal, M. P. (2007). Reasoning about component similarity in building product models from the construction perspective. *Automation in Construction*, 17(1), 11-21.
- b. Zhang, J., Webster, A., Lawrence, M., Nepal, M., Pottinger, R., Staub-French, S., and Tory, M. (2011). Improving the usability of standard schemas. *Information Systems*, 36, 209-221.

Versions of Chapters 2, 3, and 4 of this thesis have been either submitted, or are in the process of submission for possible publication.

# Table of Contents

<b>Abstract.....</b>	<b>ii</b>
<b>Preface.....</b>	<b>iv</b>
<b>Table of Contents .....</b>	<b>vi</b>
<b>List of Tables .....</b>	<b>x</b>
<b>List of Figures.....</b>	<b>xii</b>
<b>List of Abbreviations .....</b>	<b>xvi</b>
<b>Acknowledgements .....</b>	<b>xvii</b>
<b>Dedication .....</b>	<b>xix</b>
<b>Chapter 1: Introduction .....</b>	<b>1</b>
1.1    Proposed Solution .....	3
1.2    Research Questions .....	6
1.3    Expected Contributions .....	9
1.4    Research Process and Methodology.....	9
1.5    Research Scope .....	18
1.6    Reader’s Guide.....	19
<b>Chapter 2: Towards Automated Extraction and Querying of Construction-Specific Design Features from a Building Information Model (BIM) .....</b>	<b>21</b>
2.1    Introduction .....	21
2.2    Desirable Requirements of the Proposed Framework .....	24
2.3    Motivating Case .....	25
2.4    Related Work.....	29
2.4.1  Design Relevant Construction Knowledge and Information .....	29
2.4.2  Building Product Models.....	33
2.4.3  Features and Feature-Based Modeling .....	34
2.4.4  Related Research on Product Model Reasoning.....	34
2.5    Research Framework.....	36
2.5.1  Knowledge Structures.....	39
2.5.1.1  Feature Ontology .....	40

2.5.1.2	Query Specifications .....	44
2.5.2	Reasoning Modules .....	46
2.5.2.1	Extracting Features to Create a Feature-Based Model.....	46
2.5.2.2	Querying Features .....	48
2.6	Evaluation Studies and Discussion.....	53
2.7	Conclusions .....	59

### **Chapter 3: A Construction-Specific Ontology of Design Features and Feature-Based**

<b>Model .....</b>	<b>61</b>	
3.1	Introduction .....	61
3.2	Practical Motivation .....	64
3.3	Literature Review .....	66
3.3.1	Related Research on Features and Feature-Based Modeling.....	66
3.3.2	Previous Research on Ontological Modeling .....	68
3.3.3	Previous Research on Building Product Modeling.....	69
3.3.4	Previous Research on Design Relevant Construction Knowledge .....	70
3.4	An Ontology of Design Features.....	71
3.4.1	Acquisition and Elicitation of Knowledge for Developing the Feature Ontology .....	72
3.4.2	Developing the Feature Ontology.....	72
3.4.3	Representation of Features in the Ontology .....	74
3.4.3.1	Classifying Features.....	74
3.4.3.2	Attributes of Feature Classes .....	79
3.5	Feature Extraction and the Feature-Based Model .....	86
3.5.1	Mapping the BIM Model to the Ontology: Automating Feature Extraction .....	86
3.5.2	Feature-Based Model Interface and its Navigability .....	94
3.6	Evaluating the Feature Ontology.....	96
3.7	Conclusions .....	102

### **Chapter 4: Query Specifications and Templates for Specifying Queries..... 103**

4.1	Introduction .....	103
4.2	Motivating Case Examples.....	105
4.3	Related Research Background.....	110
4.3.1	Building Information and its Representation.....	110
4.3.2	Reasoning about Building Product Models .....	112
4.3.3	State-of-the-Art BIM Analysis Tools .....	114

4.4	System Architecture for Querying Construction-Specific Design Information .....	115
4.5	Specification of Construction-Relevant Queries .....	116
4.5.1	Classifying Queries.....	117
4.5.2	Specifications of Basic Queries .....	118
4.5.2.1	Common Attributes for Basic Queries.....	119
4.5.2.2	Attributes Specific to Opening and Penetration Queries .....	120
4.5.2.3	Attributes Specific to Component Intersection Queries.....	120
4.5.3	Extended Queries.....	121
4.5.4	Specifications of Extended Queries.....	125
4.5.4.1	Common attributes of Extended Queries .....	125
4.5.4.2	Attributes Specific to Location Queries.....	126
4.5.4.3	Attributes Specific to Spacing Queries .....	127
4.5.4.4	Attributes Specific to Alignment Queries .....	129
4.5.4.5	Attributes Specific to Design Uniformity Queries.....	131
4.6	Re-Usable and Computer Interpretable Query Templates .....	132
4.6.1	Query Templates for Formulating Basic Queries .....	132
4.6.1.1	Example # 1 Query on a Component Feature .....	132
4.6.1.2	Example # 2 Query on Opening (or Penetration) Feature.....	135
4.6.1.3	Example # 3 Query on Component Intersection.....	138
4.6.2	Query Templates for Formulating Extended Queries.....	141
4.6.2.1	Example # 1 Query on Feature Location .....	142
4.6.2.2	Example # 2 Query on Feature Spacing.....	145
4.6.2.3	Example # 3 Query on Feature Alignment .....	148
4.7	Summary and Conclusions.....	154
<b>Chapter 5: Conclusions .....</b>		<b>157</b>
5.1	Research Contributions .....	157
5.2	Validation Studies .....	161
5.2.1	Interviews with Construction Experts .....	165
5.2.1.1	Expert Opinion on the Feature “Wall” .....	167
5.2.1.2	Expert Opinion on the Feature “Column”.....	171
5.2.1.3	Expert Opinion on the Feature “Component Intersection” .....	175
5.2.1.4	Expert Opinion on the Features “Opening” and “Penetration”.....	179
5.2.2	Retrospective Analysis .....	185
5.2.2.1	Summary of the Retrospective Analysis.....	195

5.2.3	Descriptive and Interpretative Analysis.....	198
5.2.3.1	Richer Representation of Component Types .....	198
5.2.3.2	Richer Support for Identifying Component Intersections .....	208
5.2.3.3	Richer Support for Identifying Component Penetrations and Openings and Other Insightful Information .....	210
5.2.3.4	Richer Support for Identifying the Spacing of Features .....	213
5.2.3.5	Richer Support to Query for Feature Alignment.....	215
5.2.3.6	Richer Support for Identifying the Design Uniformity of Features .....	217
5.2.4	Conclusions from Validation Studies .....	219
5.3	Practical Implications .....	220
5.4	Limitations of the Current Research and Future Research Directions .....	221
<b>References .....</b>		<b>224</b>
<b>Appendix A: Numbered References Cited in Tables 2.3 and 5.8 to 5.13.....</b>		<b>235</b>

## List of Tables

Table 2.1	Distribution of column sizes on different floors in the Chem-Bio building .....	26
Table 2.2	Different criteria for defining wall types .....	28
Table 2.3	Summary of design conditions identified in the literature that impact construction .....	31
Table 2.4	Summary of the level of support results for identifying different design conditions .....	55
Table 3.1	Generic attributes to the feature class “Component” .....	82
Table 3.2	Generic attributes of the feature class “Wall” .....	83
Table 3.3	Generic attributes for the feature class “Column” .....	84
Table 3.4	Generic attributes for the feature class “Intersection” .....	84
Table 3.5	Generic attributes for the feature class “Component Intersection” .....	85
Table 3.6	Generic attributes for the feature class “Opening” .....	85
Table 3.7	Generic attributes for the feature class “Penetration” .....	85
Table 3.8	Expert opinion on design conditions related to the feature “Wall” .....	98
Table 5.1	Expert opinion on design conditions related to the feature “Wall” .....	168
Table 5.2	Expert opinion on design conditions related to the feature “Column” .....	173
Table 5.3	Expert opinion on design conditions related to the feature “Component Intersection” .....	177
Table 5.4	Expert opinion on different types of “Wall-to-Wall Intersections” .....	178
Table 5.5	Expert opinion on design conditions related to the feature “Opening” .....	181
Table 5.6	Expert opinion on design conditions related to the feature “Penetration” .....	182
Table 5.7	Expert opinion on design conditions related to the “Horizontal and Vertical Locations of Openings and Penetrations” .....	184
Table 5.8	Level of support results for “Component” features in general .....	187
Table 5.9	Level of support results for the feature “Wall” .....	188
Table 5.10	Level of support results for the feature “Column” .....	190
Table 5.11	Level of support results for the feature “Component Intersection” .....	191
Table 5.12	Level of support results for the feature “Opening” .....	193
Table 5.13	Level of support results for feature “Penetration” .....	194

Table 5.14	Summary of the level of support results for identifying different design conditions .....	196
Table 5.15	Different criteria for defining wall types by cost estimators .....	199

## List of Figures

Figure 1.1	A general framework for extracting and querying a BIM.....	4
Figure 1.2	CIFE ‘Horseshoe’ research process (Adapted from: Fischer 2006).....	10
Figure 2.1	Annotated drawings of the size and location of openings on walls (left) and penetrations on a slab (right) by a site superintendent on a local project that we studied .....	22
Figure 2.2	The 3-D model of the Chem-Bio building .....	25
Figure 2.3	Different wall types defined for the Chem-Bio project.....	27
Figure 2.4	Color marking of different design conditions by an estimator on a pdf drawing.....	29
Figure 2.5	A generic research framework for extracting and querying a BIM.....	36
Figure 2.6	The feature hierarchy represented in <i>Protégé Frame Editor</i> .....	43
Figure 2.7	A taxonomy of queries .....	45
Figure 2.8	The FBM and browsing of features of a sample model .....	48
Figure 2.9	Query interface for specifying a query .....	49
Figure 2.10	Step 1 — Feature selection.....	50
Figure 2.11	Step 2 — Property filtration .....	50
Figure 2.12	Step 3 — Grouping.....	51
Figure 2.13	Step 4 — Quantification.....	52
Figure 2.14	A sample query output.....	53
Figure 2.15	Summary of the results for the level (or degree) of support for different features .....	56
Figure 3.1	Wall configuration on the fifth floor of the Chem-Bio building, UBC.....	65
Figure 3.2	The feature hierarchy.....	76
Figure 3.3	Slab openings in the fifth floor of the Chem-Bio Building.....	77
Figure 3.4	Examples of different types of penetrations on a typical lab floor of the Chem-Bio building.....	77
Figure 3.5	Specification diagram of features and their relationships using UML (Booch et al. 1999).....	79
Figure 3.6	Class diagram with feature attributes .....	81

Figure 3.7	An <i>IDEFO</i> model for extracting features from a BIM, to create a feature-based model .....	87
Figure 3.8	A wall with corresponding ifc and ifcXML representations .....	88
Figure 3.9	Attributes and relationships with reference paths showing their linkage to a wall object in ifcXML .....	89
Figure 3.10	Example of a wall-to-wall intersection and the details provided by the ‘intersection query’ predicate.....	93
Figure 3.11	Browsing construction-specific features in the FBM .....	95
Figure 3.12	Explicit representation and browsing of component intersections in the FBM .....	96
Figure 3.13	Comparison of level of support results: our approach and state-of-the-art tools .....	101
Figure 4.1	Example from the <i>CIRS</i> project showing a design conflict in which a duct cannot fit in the allocated wall opening .....	106
Figure 4.2	Annotated drawings created by the superintendent on the Engineering Design Centre (EDC) project identifying the size and location of openings on walls (left) and penetrations on a slab (right) .....	106
Figure 4.3	Variation in Column size, shape, and location in a floor, and from floor to floor, in the Chem-Bio building project, UBC .....	108
Figure 4.4	Colour marking of different design conditions by an estimator on a 2D drawing.....	109
Figure 4.5	An <i>IDEFO</i> model architecture for automatically querying features .....	116
Figure 4.6	A taxonomy of queries .....	118
Figure 4.7	Basic queries and associated attributes formalized in this research .....	119
Figure 4.8	Groupings of uniform walls based on: (a) wall types and (b) wall height .....	124
Figure 4.9	Extended query types and associated attributes .....	125
Figure 4.10	Illustration of the different attributes required to locate penetrations on walls, based on a lab in the Chem-Bio Building project, UBC .....	127
Figure 4.11	Illustration of some attributes for column spacing .....	129
Figure 4.12	Illustrations of the “feature alignment reference” attribute for opening and column features .....	130

Figure 4.13	Step 1 – Feature selection.....	133
Figure 4.14	Step 2 - Property filtration.....	134
Figure 4.15	Step 3 – Grouping.....	134
Figure 4.16	Step 4: Quantification.....	135
Figure 4.17	Step 2(a) – Filtration - Specifying properties for the feature “Opening” .....	136
Figure 4.18	Step 2 (b) – Filtration - Specifying properties for the host feature “Wall” .....	137
Figure 4.19	Step 3 – Grouping - Specifying grouping criteria for “Opening” .....	137
Figure 4.20	Step 4 – Quantification - Specifying the parameter(s) and aggregate function(s) to be used in quantifying feature properties .....	138
Figure 4.21	Step 1 – Feature selection process to specify a component intersection query.....	139
Figure 4.22	Step 2 – The process of property filtration in specifying a component intersection query.....	140
Figure 4.23	Example of a wall-to-wall intersection and the details provided by the “intersection query” .....	141
Figure 4.24	Step 1 – Feature selection for specifying the location of penetration .....	142
Figure 4.25	Step 2 – Property filtration .....	143
Figure 4.26	Step 3 – Parameters specification process for specifying the location of penetration.....	143
Figure 4.27	Example of duct penetration and its location in relation to the wall boundaries (sides) indicated.....	145
Figure 4.28	Step 3 – Parameters specification process for specifying the column spacing query .....	146
Figure 4.29	Spacing of proximate, on-grid columns in the first floor plan .....	147
Figure 4.30	Parameters specification process for specifying the horizontal alignment of columns.....	149
Figure 4.31	A portion of the column layout plan in a typical floor of the Chem-bio building project, UBC.....	150
Figure 4.32	On-grid and off-grid columns on part of Level 1 .....	152
Figure 5.1	3D models of the case study projects used in this research.....	164

Figure 5.2	Summary of the results for the level (or degree) of support for different construction-specific features .....	197
Figure 5.3	Style of designating and defining wall types in traditional work practice .....	200
Figure 5.4	An example of applying the wall definition from Figure 5.3 to the actual design .....	201
Figure 5.5	Methods of defining wall types in <i>Revit Architecture</i> .....	202
Figure 5.6	Colour-marking the appropriate type of component – walls, columns, slabs on grade, suspended slabs – on Level 1 .....	204
Figure 5.7	Colour-marking of different floor finishes on Level 1 .....	204
Figure 5.8	Colour-marking of different ceiling types and finishes in the reflected ceiling plan of Level 1 .....	205
Figure 5.9	Colour-marking of curtain walls, sunscreen system, and vents, in elevation view .....	205
Figure 5.10	Grouping of walls using different criteria .....	207
Figure 5.11	Example of a wall-to-wall intersection and the details provided by the ‘intersection query’ predicate.....	209
Figure 5.12	Site superintendent marking by hand, the size of duct penetrations, on a portion of the slab on the EDC project.....	211
Figure 5.13	Site superintendent manually locating the openings on walls on the EDC project.....	211
Figure 5.14	Site superintendent marking the openings on structural drawings on the EDC project.....	212
Figure 5.15	Example of duct penetration, and its location in relation to the wall boundaries (sides) in one of the masonry walls on the Chem-Bio building, UBC .....	213
Figure 5.16	Spacing of proximate, on-grid columns on Level 1 .....	215
Figure 5.17	The designation of on-grid and off-grid columns on part of a floor plan.....	217
Figure 5.18	Variation in column size/shape, location in a floor, and from floor to floor in the first three floors of the Chem-Bio building, UBC.....	218

## List of Abbreviations

2D	Two Dimensional
3D	Three Dimensional
4D	Four Dimensional
AEC	Architectural, Engineering and Construction
BIM	Building Information Model (or Modeling)
CAD	Computer Aided Design
Chem-Bio	Chemical and Biological Engineering
CIRS	Centre for Interactive Research and Sustainability
CM	Construction Management
CSI	Construction Specifications Institute
EDC	Engineering Design Centre
FBM	Feature Based Model (or Modeling)
FM	Facility Management
GIS	Geographic Information Systems
GML	Geography Markup Language
IAI	International Alliance for Interoperability
IDEF0	Integrated Definition for Function Modeling
IFC	Industry Foundation Classes
ISO	International Organization for Standardization
MBR	Model Based Reasoning
MEP	Mechanical, Electrical and Plumbing
OCCS	Overall Construction Classification System
SMC	Solibri Model Checker
SQL	Structure Query Language
STEP	Standard Exchange for Product Model Data
UBC	University of British Columbia
UBC-O	University of British Columbia – Okanagan
XML	Extensible Markup Language

## Acknowledgements

This Ph.D. research is a long journey. I could not have come to this stage without the help, guidance, and support of many individuals. First and foremost, I am very grateful to my supervisor, Dr. Sheryl Staub-French for believing in me. She helped me in every possible way she could. She provided unabated support – financial, moral, and technical. She provided comfort when I had difficulties. Thank you so much for your help, inspiration, dedication and guidance. It's been a great pleasure working with you.

I thank Dr. Rachel Pottinger for honing my research skills through her insightful, critical comments and feedback and helping me to think logically and learn how to do a research. Without her help and inspiration this research would not have materialized. I am very thankful to Prof. Alan Russell for his suggestions and advice on some of the most critical aspects of this research. You are truly phenomenal in guiding me and showing the roadmap when I had difficulties and was stuck on issues. Your comments and feedback have been very instrumental. I am grateful to Dr. Terje Haukaas for helping me to learn how to answer questions and keep focused. My appreciation also goes to Prof. Siegfried Stiemer for helping me out and lifting my motivation. I also thank Prof. Thomas Froese for teaching what research is and encouraging academic participation.

I have been fortunate to have the support and help of computer scientists and experts. Dr. Melanie Tory has been a very resourceful person. Special thanks go to Jiemin Zhang and April Webster for materializing the conceptualization through implementations. Michael Lawrence has been of great help, especially in digging out the BIM technology and setting things up for the Artifact project. Yun Lou did a great job in improving the user interface design. Jamila Salari was helpful in some aspect of the implementation.

I would like to thank all my colleagues in the project and construction management program at UBC. Special thanks go to Jehan Zebb for his help, encouragement and untiring support. I have benefited greatly from him. I would like to thank Saad Abo-Moslim, Chao-Ying Chiu, Ming-En Li, Hasan Burak Cavaka, Ngoc Tran, and Cheryl Nelms for many useful discussions, and for their help and moral support. It has always been refreshing to chat on issues with Amir Tangestani, Behzad Pilehchian, Fengyang Chen, Leo Gu, Sadaf Nezafatkah, Helia Amir, and Mohammadali Ghamsari Esfahani. Thank you all.

I have been very lucky to meet and get help from and talk over ideas with many graduate students over the years. They include: Vikram Singh Parmar, Mani Golparvar Fard, Ganesh Subramanian, Zonghai Han, Kehui Zhang, Sanjaya de Zoysa, Asad Udaipurwala, Abdoreza Tabesh, Paul Adu Tawiah, Reem Abdul-Hafidh, and Seyed Mahdi Behestian.

I am truly grateful to Norman Petersen for his help. I've greatly benefited from his expertise over the past two and half years. Also, my special thanks go to other construction experts: Umar Sharif, Daniel Bjornson, Norman Young, Arthur Atkinson, Gord Sookaveiff for sharing their construction knowledge and providing valuable resources, not to mention their time. I would also like to thank Julie Sedger for her help in editing this thesis.

Many Nepali friends and families at UBC have helped and supported me in many different ways. My thanks go to Bishnu Pandey, Indira Pandey, Ratna K. Shrestha, Sanjaya Regmi, Laxmi Subedi, Umesh Phuyal, Kusum Poudel, Aditya Sharma, Tejendra Chapagain, Sarita Chapagain, Satish Jah, Conrad Rijal, Sushma Rijal, and Madhab Bajgain.

My parents — Kabi Raj Nepal and Shiva Kumari Nepal — sacrificed a lot for the sake of my study and future. I can't express in words how much devastated I was when my father — Kabi Raj Nepal — suddenly passed away just a year ago. "Time and tides wait for none!" I miss him so much and he will forever be remembered. All my brothers, sisters, brother in laws, sister in laws, nephews, and nieces have always been very supportive, caring and encouraging throughout my study. Thank you all.

Finally, my wife Suchitra, alias Sonu, has been extraordinarily patient, supportive, and caring. Thank you for teaching me to realize the joyful aspects of life and learn to enjoy today because tomorrow never comes the way you think it would. My children, the by-product of my Ph.D., deserved more time and care from me. Yet, they understood I had to go to the campus to study even during weekends and holidays. It's wonderful seeing them growing along with my Ph.D. They have been truly inspirational for the success of my Ph.D.

## **Dedication**

In loving memory of my father, Kabi Raj Nepal  
You will always be in my heart and soul.

To my mother, Shiva Kumari Nepal  
For your unconditional love, care and patience.

To my wife Suchitra (Sonu)  
For your love, patience and support.

To my son, Mihir and daughter, Akina  
For your understanding and cooperation,

and

To my brothers, sisters and other relatives.

## Chapter 1: Introduction

Building Information Modeling or Model (BIM) is a promising technology in the architectural, engineering and construction (AEC) industry. It provides a digital representation of a building that contains intelligent objects with precise geometry, topology and relevant data needed to support design, construction, fabrication, procurement and facility management activities. BIM involves more than simply representing buildings in three dimensions (3D) or using 3D modeling software. It also represents a novel process and a new way of working. BIM has changed the way buildings are designed, analyzed, constructed and managed (Hardin 2009). When properly implemented, BIM facilitates more integrated design and construction processes, which results in improved communication, coordination and collaboration, not only among architects, engineers, and contractors, but also with owners, facility managers, and sustainable design professionals. This results in a faster and more cost-effective project delivery process, and higher quality buildings that perform at reduced costs (Hardin 2009; Eastman et al. 2008). The ongoing development of BIM technology facilitated by the *Industry Foundation Classes (IFC)* has enabled the sharing, exchange and reuse of such information across multiple disciplines and software applications (IAI 2010).

BIM offers major advantages to the construction industry. BIM has proven to be useful to a variety of construction management functions, including construction planning and scheduling (e.g., Kristiina et al. 2009; Russell et al. 2009), clash detection and construction coordination (e.g., Korman et al. 2003; Khanzode et al. 2008), cost estimating (e.g., Staub-French et al. 2003), constructability analysis (e.g., Wissam et al. 2009), offsite fabrication (Eastman et al. 2008) and purchasing (Hardin 2009).

While BIM has proven to be useful to the construction industry, there are still enormous challenges in order to fully leverage BIM for construction. Specifically, BIMs do not explicitly represent all relevant construction information in a form required or desired by construction practitioners (Borrmann and Rank 2009; buildingSMART alliance & OGC Inc. 2008). This can be attributed to the fact that: (a) designers cannot foresee every intended use of the developed models during the 3D modeling process (*Graphisoft* 2004); (b) production

and construction requirements are often not taken into account due to the separation of design and construction (Laitinen 1998); (c) BIMs are mostly designer-centric and construction models differ from the design model (*Graphisoft* 2004); or (d) construction knowledge has not yet been formalized to the extent necessary to explicitly consider construction input in the information models (Fischer 2006). BIM can inundate construction users with information (Hardin 2009). Current BIM tools do not provide sufficient reasoning mechanisms to derive production relevant information at the level of detail necessary for practitioners (Haymaker et al. 2004; Katranuschkov et al. 2003; Borrmann and Rank 2009). Construction practitioners cannot directly use *IFC* models to support their work because these are data models designed primarily for specification and exchange of data.

While it is not reasonable to expect all relevant information needed by all dependent applications or management functions to be explicitly represented in a building product model (or BIM), query facilities (especially spatial queries) and reasoning mechanisms can support the deduction of spatial relationships between components and provide different views and abstractions of design (Wong and Sriram 1999; Haymaker et al. 2004; Katranuschkov et al. 2003). However, with existing tools and applications, construction practitioners cannot formulate user-driven and customizable queries to fulfill varying construction requirements and preferences involved in using design information. For instance, BIM users today do not have sufficient computer support to easily and expediently formulate queries, such as identifying the locations of duct and pipe penetrations on walls on each floor of a building. Reasoning about more complex spatial relationships in a design, such as a change in size, location, or alignment of columns from floor to floor is an even more daunting task. These kinds of design conditions can have a significant impact on the efficiency and quality of construction.

Lacking adequate computer support, construction practitioners today spend a significant amount of time manually interpreting and analyzing BIMs, or more typically 2D drawings, to identify these design conditions. The process is inefficient, time-consuming, and prone to error. For large and more complex projects, visual inspection and manual analysis of BIMs to identify this information becomes prohibitive. Emerging BIM applications are addressing parts of this problem. *Solibri Model Checker*<sup>®</sup> leverages BIM models for checking building designs for compliance with a given set of design requirements.

*NavisWorks*<sup>®</sup> identifies interferences and component clashes. These tools provide sophisticated mechanisms for browsing and navigating product data that is explicitly defined in the design model, but offer limited support for practitioners needing to identify construction-specific design features from a BIM in a way that suits their needs and requirements.

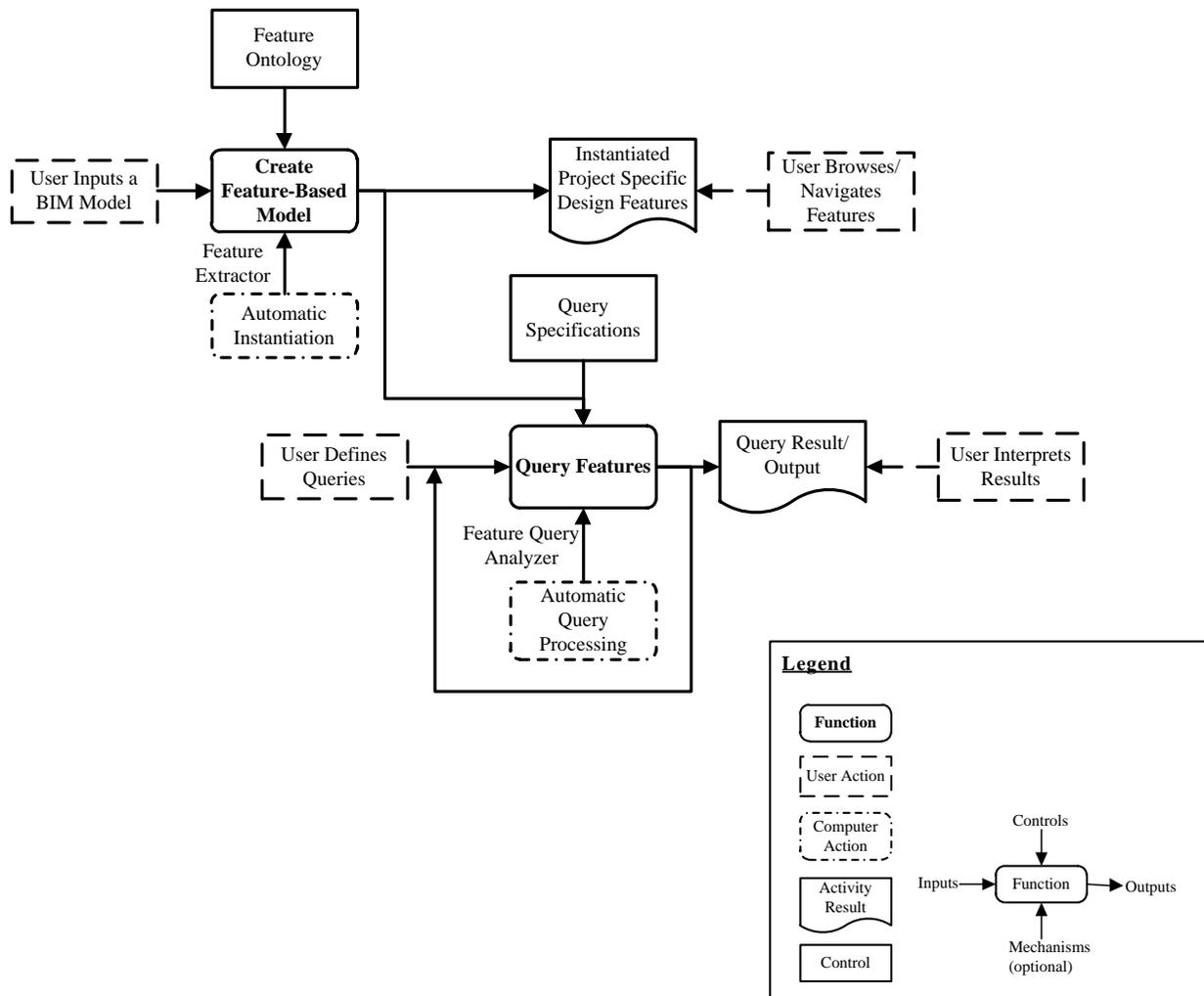
A variety of research efforts have developed novel approaches to address this challenge. For example, Haymaker et al. (2004) developed a novel method using primitive features that enables engineers to construct integrated and task-specific geometric views based on a BIM. However, the approaches to date have been limited in many aspects: require extensive user input (Nguyen and Oloufa 2002), represent and analyze a narrow set of design features and conditions (Chen et al. 2005), lack user customizability and flexibility (Haymaker et al. 2004), focus only on *IFC* and/or product data (Katranuschkov et al. 2003), lack practical applications to construction (Borrmann and Rank 2009), and provide query support exclusively for explicitly defined *IFC* properties and relationships, which are limited to containment relationships defined in the product model only (Adachi 2003).

This research aims to address some of these limitations by developing an approach to support automated extraction and querying of construction relevant design information from a BIM. It is believed that the results of this research can support decision making processes for a variety of construction management functions, including cost estimating, construction planning, execution and coordination, purchasing, constructability analysis, methods selection, and productivity analysis.

## **1.1 Proposed Solution**

The proposed solution uses a novel approach to extract construction features from a given BIM and support the processing of user-driven queries. Figure 1.1 graphically illustrates the approach and architecture of the proposed system. In the first step, the input *IFC*-based BIM model is transformed into a project-specific feature-based model which explicitly represents the features integral to a particular construction domain. For this step, we formalized a *feature ontology* to generically represent construction-specific design conditions. Concepts in the feature ontology are mapped to the concepts in the underlying BIM or *IFC* model to automatically instantiate the feature ontology to create a feature-based

model for a particular project. In the second step, users configure queries that operate on the project-specific feature-based model. For this step, we developed *query specifications* to formalize the language and structure of the user-driven queries in relation to a BIM. The query specifications define a query vocabulary and attributes to specify different types of spatial and non-spatial queries.



**Figure 1.1 A general framework for extracting and querying a BIM**

Our approach is novel in that it combines model-based reasoning, feature-based modeling, and query processing that enhance the flexibility of the system. Model-based reasoning allows for the interpretation of a BIM to abstract the geometric, topological and

other data about objects in a BIM model. Feature-based modeling enables the customization of a BIM to a specific practitioner or domain. A query-based approach allows for the flexibility to define queries and rapidly generate the desired view. It is more flexible than attempting to define all construction features *a priori* and allows construction practitioners to specify queries according to their particular needs.

The proposed solution should provide generic, formal and flexible representation of domain concepts relevant for practitioners. It should provide construction practitioners, who are non-expert BIM users, with easy-to-use and customizable query support to specify user queries that are both informative and meaningful. These requirements are explained briefly below.

- **Generic:** The requirement for the generality of the research tool is well documented (Haymaker et al. 2004; Staub-French et al. 2003; Li 2009). The approach developed in this research should, ideally, be general enough to extract and query different types of components and features, and be useful for a broad range of construction management functions.
- **Formal:** The framework needs to have a structured, explicit, and computer-interpretable representation of concepts and computer implemented methods in order to leverage the richness of digital information available in BIMs. The formal representation should include all the attributes necessary to support the automatic instantiation and querying of construction-specific design features from a BIM.
- **Flexible:** In order to accommodate the varying requirements of construction practitioners, information models require flexible representation and manipulation of information (Haymaker et al. 2004; van Leeuwen and Wagter 1997; Katranuschkov et al. 2003). In the context of this research, the representational structure and reasoning methods should be flexible enough to accommodate construction practitioners' varied preferences or requirements for defining features and specifying queries.
- **Easy to Use:** End-users are those who interact with the developed system or tool. No matter how effective the developed tool, if it is difficult to use and the end-users cannot communicate or interact easily with it, the product will be of limited use (Carrico et al. 1989). In the context of this research, users should be able to configure

queries according to their preferences, without having to undertake programming or query writing.

## 1.2 Research Questions

The overall goal of this research is to formalize construction-specific information about building components in a formal, flexible and computer interpretable way in order to enable the automated extraction of features and to support the formulation of user-driven and customizable queries from a BIM model. To fulfill this broad research goal, the research specifically seeks to respond to the following research questions:

### **Research Question 1: *How can construction-specific design conditions be formalized to support extraction and querying of a building information model (BIM)?***

Different types of design conditions impact construction. They include the horizontal and vertical layout of elements; spacing between elements; component materials; location, size, shape, type, similarity, repetition, alignment and uniformity of components, among others (Thomas and Zavrski 1999; Fischer 1991; Hanna and Sanvido 1990). Construction practitioners scrutinize the aforementioned design conditions (and others) in every building model as they are critical to a variety of construction management functions, including constructability assessment (Boeke 1990), productivity analysis (Thomas and Zavrski 1999), method selection (Fisher and Tatum 1997), and cost estimating (Staub-French et al. 2003). In an effort to generically and formally represent different design conditions of concern to construction practitioners, this research develops an ontology of design conditions, which we call a *feature ontology*, a term created by Staub-French et al. (2003).

Ontologies, which are explicit specifications of a conceptualization, are highly useful for representing domain specific knowledge (Gruber 1995), such as knowledge about construction-specific design conditions. The feature ontology generically defines design conditions using the manufacturing concept of *features* (Shah and Rogers 1988) and is formally represented to act as a semantic layer between the end users and the BIM, which enables automated feature extraction from a BIM and provides support for answering user queries. The concepts in the feature ontology are represented generically as an object hierarchy which categorizes design conditions into different feature classes (types) or

subclasses (subtypes), defines feature properties, and the relationships among features. It provides a computer interpretable vocabulary of design conditions, independent of a particular project. It also acts as a knowledge repository representing domain-specific knowledge or the user's view of a design. The feature ontology extends the representation of building product models (or BIMs) by explicitly representing components (e.g., walls and columns) and intersections (e.g., penetrations and openings) as features. It also provides a richer representation of feature properties, such as wall curvature and the dimensions of wall penetrations.

We created a prototype application, the *Feature Extractor*, which automatically extracts the features in a given BIM based on the feature ontology to create a project-specific Feature-Based Model (FBM). The resulting FBM enables practitioners to browse and navigate the features in a given design, providing an increased understanding of the features present that may impact construction. The FBM also forms the knowledge base for answering users' queries about a given design. The feature ontology and FBM are described in more detail in Chapter 3.

**Research Question 2: *How can query specifications be formalized to support spatial and non-spatial queries on BIMs?***

Construction practitioners look for different design conditions in a given design to account for them properly in support of various construction management tasks, such as cost estimating, methods selection, constructability analysis, and construction planning and execution. Practitioners have varied preferences for what design conditions are important and how to characterize them. There is limited understanding about the language and structure of construction-specific design queries and the knowledge associated with them. Moreover, there is a lack of computer support to enable users to easily and flexibly define queries from a BIM. This research attempts to bridge this gap by providing formal query specifications and templates.

In regards to the second research question, this research identifies different types of queries relevant to practitioners when analyzing a BIM. We developed a taxonomy of queries, and we have defined attributes which characterize these queries. The attributes represent query specific domain knowledge, which can be customized and specified by the

user during the query formulation process. Thus, they allow users to express queries according to their rationale and preferences.

We developed query specifications to support queries on components, openings, penetrations, and component intersection features, and the location, spacing, alignment, and design uniformity of these features. We formalized the query vocabulary and attributes of different types of queries, independent of a BIM. These attributes leverage the feature ontology and capture both generic, as well as the query-specific knowledge, to provide users with the ability to define queries on design features. We have developed reusable and customizable query templates (or forms) that the user can populate as needed to easily and quickly specify queries without programming effort and query language expertise. We developed a prototype application, *Feature Query Analyzer*, which executes the user-defined query specifications to automatically process queries for a given BIM. Chapter 4 provides a detailed description of query specifications and templates.

**Research Question 3:** *How can an integrated approach leverage the representations formalized in (1) and (2) to enable automated processing of user-driven and customizable queries on BIMs?*

The third research question formalizes an integrated approach that combines the results from questions 1 and 2 above for answering queries on BIMs. Our approach integrates the generic feature ontology and project-specific FBM with the query specifications. It applies the model-based reasoning for feature extraction to create a feature-based model that explicitly represents construction-specific design conditions and that supports user-driven queries. This approach provides sufficient ‘flexibility’ to meet the varied requirements and preferences of construction practitioners, and ‘comprehensiveness’ to support the needs of different construction management functions (e.g., cost estimating, site management, method selection, etc.). The query-based approach implemented in a prototype application, the *Feature Query Analyzer*, provides support to answer user queries on the FBM created by the complementary application, the *Feature Extractor*. The *Feature Query Analyzer* application processes queries based on the query vocabulary we developed in accordance with the query specifications specified by the user. Chapter 2 describes the development of the framework

and the approaches employed to automatically extract features and answer user queries over BIMs.

### 1.3 Expected Contributions

The expected contributions of this Ph.D. research can be categorized as primary and secondary contributions. The primary contributions of this research are summarized below. Chapter 5 provides more detailed discussion of these contributions.

Primary Contribution 1 – the Feature Ontology: We formalized a feature ontology that explicitly represents design conditions that are relevant to construction practitioners and supports the generation of a construction-specific feature-based model (FBM).

Primary Contribution 2 – the Query Specifications: We formalized a query specification vocabulary that characterizes spatial and non-spatial queries, and developed query templates to guide non-expert BIM users to specify queries.

Primary Contribution 3 – the Integrated Approach: We developed an integrated approach that combines model-based reasoning and query processing to automatically extract design features to create a project-specific feature-based model (FBM) and provide support for answering queries on the FBM.

The secondary contributions of this research study are as follows:

Secondary Contribution 1: We have identified and systematically categorized the design conditions that impact different construction domains and functions, based on case studies, interviews with construction experts and a thorough review of the literature.

Secondary Contribution 2: We have identified the degree of relevance of different design conditions and explored when and how these conditions impact construction.

### 1.4 Research Process and Methodology

Designing and developing a research platform is a “blueprint,” a *logical plan*, more than a work plan, *for getting from here to there*, i.e., from setting up the research question to answering the research question through the *logical model of proof* (Yin 2009). To develop this research, we followed a process that is iterative and driven by the observation of the

practical problem, a thorough literature review and case studies. Specifically, we used the ‘Horseshoe’ research process shown in Figure 1.2 that was developed at the Centre for Integrated Facility Engineering at Stanford University (Fischer 2006). Fischer (2006) argues that “*this method supports researchers in building on the experiential knowledge and anecdotal evidence that can be gathered on construction sites in the context of existing theory and expert knowledge to carry out practically-relevant and scientifically-sound research.*” An important aspect to highlight in this research is the use of triangulation, which uses multiple techniques to formalize the construction knowledge, implement prototypes, and ultimately, validate the developed research. The use of multiple methods in a research study requires little explanation, as different research methods overlap in many ways because they are not marked by sharp boundaries (Yin 2009). In the following section, we describe the major steps of the ‘Horseshoe’ research process and its application to this Ph.D. research.

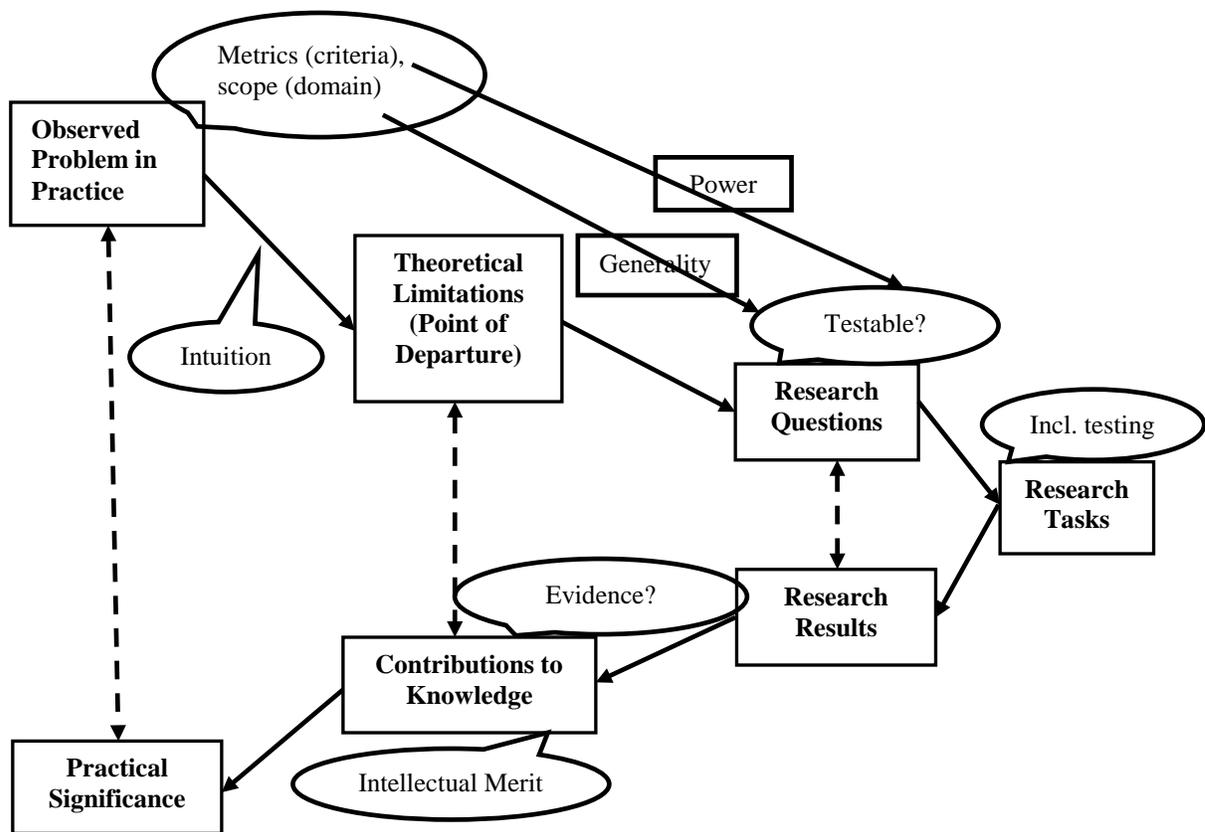


Figure 1.2 CIFE ‘Horseshoe’ research process (Adapted from: Fischer 2006)

### **a) Observed Problem in Practice**

The practical motivation for this research emerged from our observation of current work practice related to the use of BIM for construction. We found that many of the design conditions that matter to construction practitioners were not explicitly represented in current BIMs. The visual inspection and manual analysis of BIM to identify this kind of design information can be unreasonably difficult, especially for large and complex projects. Current approaches to automatically extract construction-specific design information are inadequate to fulfill the unique requirements of construction and other downstream processes.

### **b) Intuition**

An intuition relates to how the observed problem could be addressed generically, or independent to a particular construction project (Fischer 2006). Our intuition was that an ontology of design conditions could formally, generically and flexibly be developed using the manufacturing concept of “features” (Cunningham and Dixon 1988; Shah 1991), and that formal mappings could be created using XQuery (W3C 2010) to map each concept defined in the ontology to a corresponding design concept in the standard schema, such as ifcXML representation of BIM data, in order to automatically extract features. Our second intuition was that a query vocabulary combined with reusable, customizable and computer interpretable query templates might yield an effective solution that would help non-expert BIM users to easily and quickly formulate spatial and non-spatial queries on BIM in a way that extracts specific, relevant and useful construction information out of a BIM.

### **c) Points of Departure**

We reviewed a broad spectrum of literature to further enhance our understanding of the problem, identify the existing work that is useful in addressing the problem, and identify the techniques that could be used to address the problem. This research builds on, combines, and extends previous research on design relevant construction knowledge, ontological modeling, features and feature-based modeling, building product models (or BIMs), and product model reasoning.

A significant body of knowledge was identified from the existing literature on different design conditions that impact different aspects of construction or construction

management functions. We reviewed literature dealing with different design conditions that influence the applicability of a particular construction method, the constructability of a design, the productivity of resources, and the construction costs. We also examined the relevant literature to understand the type and nature of building information and its usefulness. We reviewed the related literature in ontological development to formalize construction knowledge. This broad spectrum of literature helped us to identify the research gap and provided a basis to formalize the feature ontology and query specifications.

For features and feature-based modeling, we reviewed relevant literature in the mechanical/manufacturing domains. We focused on relevant literature in AEC industry, particularly in construction, that employ feature and feature-based modeling to represent relevant construction information. In terms of building product models, we focused on related research on representational approaches, which has developed generic and specific schemas for representing building product information. We focused particularly on *IFC* to understand how it represents building information. Specifically, we investigated how construction-specific information is represented, either explicitly or implicitly, and how to create mappings from the ontology to ifcXML in order to extract features and answer queries.

In the area of product model reasoning, we reviewed the related literature on model-based reasoning and query-based approach. We studied a number of research efforts and state-of-the-art BIM analysis tools which include formalized ways for identifying design conditions that are important for construction from a BIM.

#### **d) Research Questions**

We developed research questions based on the identified problem and the research gap through thoughtful and extensive literature review. Section 1.2 describes these research questions in detail so we will simply restate them here:

- 1) *How can construction-specific design conditions be formalized to support extraction and querying of a building information model (BIM)?*
- 2) *How can query specifications be formalized to support spatial and non-spatial queries on BIMs?*

3) *How can an integrated approach leverage the representations formalized in (1) and (2) to enable automated processing of user-driven and customizable queries on BIMs?*

**e) Research Tasks**

Developing this research involved several tasks. The distinct tasks carried out include: (i) knowledge acquisition, (ii) development of the research framework and related components, (iii) prototype implementation, and (iv) evaluation/validation studies. These tasks are summarized below:

(i) **Knowledge Acquisition:**

We acquired knowledge from different sources to formalize the feature ontology and query specifications. The major undertakings include: literature review, case studies, observation of design and construction meetings, and discussions and interactions with practitioners.

First and foremost, we conducted an extensive literature review to identify the theoretical limitations. The literature reviewed spans a wide range of topics and issues, as discussed above. We reviewed literature on design constructability, value engineering, cost estimating, methods selection, construction planning and scheduling. We identified design conditions that are important to construction, reasons for their importance, and how and under what conditions they impact construction. We thoroughly reviewed the related research about the formalization of construction knowledge and the mechanisms used to reason about it from building product models. We also investigated state-of-the-art BIM authoring and analysis tools to better understand which construction information is represented and how it is represented, and to analyze the strengths and limitations of these tools.

We performed a detailed case study of the Chem-Bio building project at the University of British Columbia (UBC) and documented the existence of different design features and their characterizations by design and construction practitioners. We studied different types of documents, which included 2D drawings, 3D models, construction specifications, cost estimates, and construction schedules. We also studied related case studies of the Chem-Bio building (Tabesh and Staub-French 2006) and a number of other

construction projects at UBC in order to understand and compile a list of the design features that impact construction.

We spent over six months attending weekly design and construction meetings at the *Centre for Interactive Research and Sustainability (CIRS)* project at UBC, Point Grey campus. The *CIRS* building will be a state-of-the-art showcase of green construction and will be considered one of the greenest buildings in the world (UBC 2011). The meetings involved the participation of designers, suppliers, cost consultants, general contractors, other specialty MEP trades, and owner representatives. These meetings covered design parameters, building systems and subsystems, component typing and sizing, constructability analysis, value engineering, cost and schedule analysis, and design coordination issues. We also gained extensive access to various design and construction documents, such as meeting minutes, cost estimates, project schedules, design documents, consultants' reports, etc. This provided further information about which design conditions are important to practitioners and how they ultimately impact construction.

We participated in a series of meetings with three cost estimators in order to understand the relationship between design and construction and their processes of cost estimating and the specific items they look for in a design. We had many discussions with a construction expert with 30 years of construction experience so that we could understand design related issues and their impact on various construction trades and construction management functions, including constructability, site layout, execution and coordination, cost, and productivity. We were also able to observe the actual work practice of construction practitioners for analyzing a design. This, along with other undertakings, helped us to develop motivating cases and the research requirements.

(ii) **Development of the research framework and related components:**

As described in section 1.1, we developed a general research framework for extracting information and querying a BIM. We developed this framework using an *IDEFO* (*Integrated Definition for Function Modeling*) technique (Colquhoun et al. 1993). The specific components of this framework include the development of: (a) the feature ontology, and (b) query specifications and templates. Essentially, we developed models, which are the abstract representations of some real world entity. Modeling allows the characteristics of the

real-life situations or class of phenomena to be reproduced or represented in models (Wilson 1984). Basically, the feature ontology and query specifications that explicitly represent or describe domain concepts related to design conditions important from the construction perspective represent symbolic domain models (or formalizations). These domain models are used to access and manipulate BIM data and analyze building product models (or BIMs). In generic modeling terms, building product models can be described as “iconic,” i.e., scaled replicas or virtual representations of real world systems, i.e., buildings (Wilson 1984).

It should be understood that the models developed are qualitative and interpretative due to the fact that they are based on the explicit understanding of the entities and the relationships between them through intuition, interviews, case studies, and observational studies. From a social research perspective, this is qualitative research because we have not conducted experiments or surveys or run simulations. We have not undertaken quantitative analysis of data, such as statistical analyses for hypothesis testing. Our research is largely descriptive because we systematically attempt to develop a coherent and comprehensive view of a phenomenon and explain specific issues and solutions (Fellows and Liu 2003).

While Chapter 2 presents an overview of our integrated framework, Chapter 3 and 4 respectively provide the detailed description of the feature ontology and query specifications.

**(iii) Prototype Implementation:**

We formally represented the feature ontology in *Protégé Frame Editor*, an open-source, ontology development platform. The *Protégé-Frame Editor* provides a suite of tools for building domain models and knowledge-based applications with ontologies (Protégé 2008). We developed two prototype systems: (a) the *Feature Extractor* to automate the extraction of features in order to create an FBM, and (b) the *Feature Query Analyzer* to automatically process queries according to the taxonomy of queries and query specification definitions. It should be noted that the implementation for feature extraction and answering queries was done by two computer science researchers in our Artifact project team (Zhang 2008; Webster 2010). The input data to the system comes from a 3D BIM model modeled in an IFC-compliant 3D BIM application (in our case, a *Revit*). We use the XML representation of BIM data for feature extraction and answering queries. We used the standard XML query language, XQuery (W3C 2010), to extract and query features. Chapter 2 provides a general

overview of our implementation. Chapter 3 provides more detailed description of the feature extraction process.

(iv) **Validation:**

We used three different tests to validate our research. We used a number of metrics (or criteria), mostly subjective in nature, to gather evidence for the power and generality of the research, including the *content*, *representativeness*, *generalizability*, *soundness*, and *intellectual merits*. The following sections provide a brief discussion of the tests performed. Chapter 5 provides a detailed description of the validation studies conducted.

- ***Detailed Interviews:*** We conducted interviews with four construction experts to validate the ‘*content*’ and ‘*representativeness*’ of the concepts formalized in the feature ontology and query specifications. The interviews provide the wisdom of authorities gained from the experts’ experience and work (Neuman 2006). Personal interviews with domain experts are the most valuable and effective way to acquire domain vocabulary or knowledge and receive immediate data and feedback to verify the implemented system (Carrico et al. 1989). The evidence gathered from interviews also shows the ‘*generality*’ of the knowledge formalized in this research as four different experts representing different construction domains and viewpoints provided their inputs on the design conditions that matter to them, reasons behind why they matter, and the conditions under which they would matter.
- ***Retrospective Analysis:*** We developed retrospective analysis based on interviews, case studies of four different projects, and a thorough literature review to demonstrate the ‘*soundness*’ of our approach in relation to the state-of-the-art tools. We designed retrospective analysis of design conditions related to a “component” in general, “wall” and “column” components in particular, and “component intersection,” “opening,” and “penetration” features. This test also demonstrates the ‘*generality*’ of our approach as our research enables practitioners to extract or query different design conditions for different components, intersections, openings and penetrations features.
- ***Descriptive and Interpretive Analysis:*** Finally, we performed descriptive and interpretive analysis of the developed research for generating useful, actionable

and insightful information in order to demonstrate the ‘*intellectual merits.*’ The technique employed is largely what social scientists refer to as “second-order interpretation.” In the context of qualitative research, such as in this research, the “second-order interpretation” is qualitative interpretations from the point of view of the researcher who conducted a study (Neuman 2006). As such, we provided empirical interpretation and evidence to demonstrate the significance and validity of the developed research.

#### **f) Research Results**

The results of this research include: (i) the feature ontology that generically represents design features relevant to construction, (b) the feature-based model that represents project specific features and the FBM browser, (c) the specification of the queries and query templates and the implementation of queries, (d) the integrated framework and its implementation to automatically extract features and answer queries to identify the required information from a BIM, (e) insights from the validation tests, particularly from interviews with domain experts and descriptive and interpretative analysis.

#### **g) Contribution to Knowledge**

The validated results of the research extend previous work and address gaps in research for identifying construction-specific information from a BIM. The main contributions to knowledge resulting from this research are: (a) the formalization of an ontology of design conditions; (b) the formalization of a query specification vocabulary, and the development of query templates to specify spatial and non-spatial queries on features; (c) the formalization of an integrated approach to automatically instantiate features of a feature ontology for a particular BIM model, and to answer user-driven and customizable queries. Section 5.1 of Chapter 5 provides a detailed discussion of each contribution.

#### **h) Practical Significance**

This research can be considered as “applied” research as it is directed at end users and provides practical solutions to concrete problems and/or addresses the specific needs of practitioners (Neuman 2006). This research has practical implications to construction. Our

approach not only automatically extracts and answers queries on construction-specific design information but also provides rich, actionable and insightful information about features of a design. It thus contributes towards improving efficiency, consistency and effectiveness in decision making process in construction by eliminating the tedious manual approach of identifying design features. Although this research is primarily driven from the construction perspective and targeted to construction practitioners, the approach employed could also be useful to designers to support design analysis and facility managers to carry out facility management operations. We anticipate that this work could have impact in the following ways:

- (1) Quickly identify cost incurring features of a design to support cost estimating.
- (2) Improve the consistency and accuracy of information extracted from a BIM.
- (3) Identify constructability issues prior to construction and provide constructability feedback to designers and owners.
- (4) Improve construction efficiency and productivity by improving the speed and ease with which construction information can be obtained.
- (5) Support decision-making tasks related to purchasing and methods selection.
- (6) Provide information in a form that helps practitioners to manage the construction process and coordinate trades.
- (7) Help to make informed decisions and avoid costly mistakes in the layout and installation of components.
- (8) Make BIMs more accessible to construction.

## **1.5 Research Scope**

The main focus of this research is on construction projects with a particular emphasis on the building components, *walls* and *columns*. The topic was chosen mostly because of the fact that these components are less standardized than other components, and possess considerable variability in terms of their design. They normally have a wide range of design features affecting construction costs, means and methods of construction, etc. As such, they pose challenges for construction practitioners. In addition, the construction of these components is highly labour intensive. A large body of research, albeit fragmented and disjointed, is also available on the design conditions for these components. With our

research, therefore, we attempt to leverage the existing knowledge in order to provide practical benefits for practitioners.

The scope of the research, however, is limited in many ways. This research intends to focus on the aforementioned components and component-related features: openings, intersections, and penetrations. Currently, this research does not focus on low-level design conditions, such as connections and joints. In its current form, it does not cover certain design conditions, such as tolerances, clearances, and modularity of design. This research has the potential to provide useful information to aid in decision making for different construction management functions in general, but in particular, it provides more direct support to cost estimating, constructability analysis, construction planning and site execution.

## **1.6 Reader's Guide**

Readers are reminded that this thesis follows the manuscript-based thesis guidelines. Efforts have been made to minimize repetition of content from chapter to chapter and to present the entire content in a unified and coherent form. The remainder of this thesis is organized as follows:

In Chapter 2, we provide a comprehensive overview of our research. We provide the theoretical and practical motivations for the need for this research. We describe the characteristics of our approach and methods implemented for automated feature extraction and for answering queries that are meaningful to construction practitioners.

Chapter 3 describes the generic representation of design conditions in a feature ontology, and the reasoning mechanisms developed to instantiate project-specific features from the feature ontology for a given BIM.

Chapter 4 describes the formalization of query specifications and corresponding templates designed to assist practitioners, who are not BIM or computer experts, to easily and expediently specify customizable and user-driven queries.

In Chapter 5, we present our conclusions, focusing mainly on the contributions resulting from this research and the validation studies conducted. We highlight practical implications of this research, describe some limitations of the research and suggest recommendations for future research.

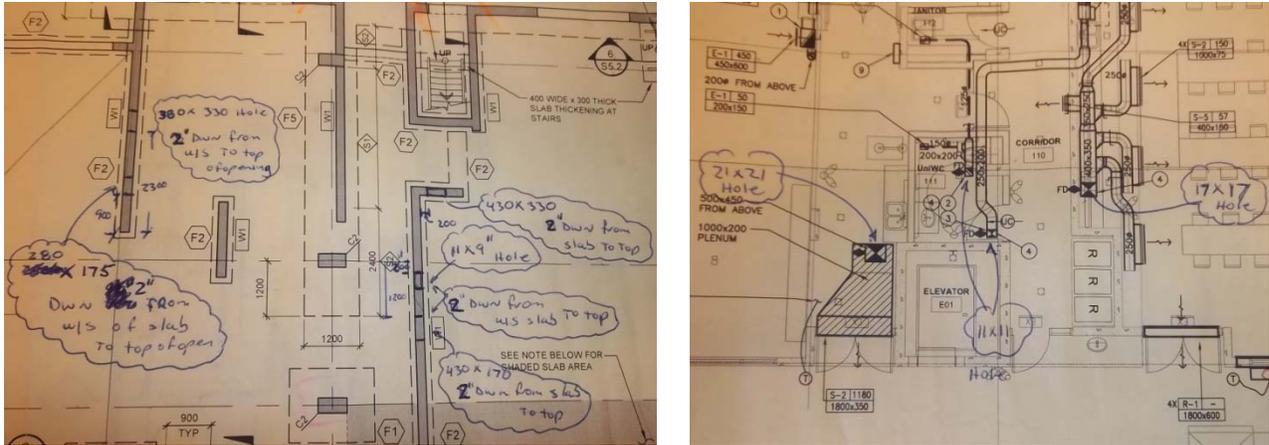
## Chapter 2: Towards Automated Extraction and Querying of Construction-Specific Design Features from a Building Information Model (BIM)

### 2.1 Introduction

In recent years, there has been a growing interest from the design and construction community to adopt Building Information Models (BIMs). BIMs are intelligent information models that explicitly represent geometric (e.g., dimensions), topological (e.g., connections), and semantic (e.g., material properties) information related to the life cycle of a facility. The continued development of BIM technology facilitated by *Industry Foundation Classes (IFC)*, has enabled the sharing, exchange and reuse of such information across multiple disciplines and software applications. While the richness of design information offered by BIMs is evident, there are still tremendous challenges in getting construction-specific information out of BIMs, limiting the usability of these models for construction and other downstream processes.

Consider the following example scenario from a project we studied:

*Forming trades on the job site need to know in advance, the size, location and type of **openings** and **penetrations** on concrete walls, slabs and beams, for forming and shoring. Drywall and masonry trades also require similar information on the layout and construction of walls. Mechanical, electrical and plumbing (MEP) trades require this information in order to layout, position, and route building service components and to ensure that design, construction, and operational requirements are met. All of these trades not only need to collaborate with each other, but more importantly, must be aware of any changes in structural and architectural designs in order to accurately account for those changes in the execution of their work. Today, construction practitioners working for these trades must manually analyze and interpret design drawings and related documents to identify these kinds of **construction-specific design conditions**. Then they typically mark-up or annotate these conditions in the drawings, as shown in Figure 2.1; they must repeat this same process each time a design changes.*



**Figure 2.1** Annotated drawings of the size and location of openings on walls (left) and penetrations on a slab (right) by a site superintendent on a local project that we studied

Penetrations and openings are just a few examples of the design conditions that are important to construction practitioners. Others noted in the literature and confirmed in our own case studies include: the horizontal and vertical layout of elements, spacing between elements, tolerances, alignment, modularity, repetition, similarity, uniformity, and standardization (Fischer and Tatum 1997; Hanna and Sanvido 1990; ASCE 1991; BCA 2001). These kinds of design conditions must be present in a BIM or derived from a BIM to provide optimal use for construction.

Current BIM tools provide some support for identifying these kinds of design conditions. For example, Autodesk® Navisworks® Manage (hereafter referred to as *Navisworks*) is often used on BIM projects to support building system coordination to help identify design conditions like the ones shown in Figure 2.1. *Navisworks* provides sophisticated functionality for detecting conflicts between systems and managing the resolution of these conflicts over time, which is very important for design coordination. However, *Navisworks* is not able to differentiate between a conflict and a penetration, and they do not provide any information on the location and size of the opening or penetration, which is what the practitioner really needs. *Solibri Model Checker (SMC)* is another sophisticated BIM analysis tool that provides some support for analyzing a BIM from different perspectives, including construction. *SMC* can be used to visualize and analyze a BIM for its integrity, quality, and for compliance with a given set of design requirements, which may include code checking, information takeoff, and conflict detection

(SMC Inc. 2010). However, similar to *Navisworks*, *SMC* is unable to provide the necessary support for practitioners for the broad range of construction-specific design conditions described above. To summarize, current BIM models and analysis tools do not represent most relevant construction-specific information explicitly and in a form required by construction practitioners (Borrmann and Rank 2009; buildingSMART alliance & OGC Inc. 2008; Haymaker et al. 2004).

Recently, researchers have tried to address these challenges by providing additional computer-based support for various construction management functions. Some researchers have focused on deriving new models or extending the designer-focused BIM models to represent other perspectives (e.g., Haymaker et al. 2004). Others have developed mechanisms to transform or enrich design-focused product models to derive construction-useful information (Akbas 2004; Akinci et al. 2002; Navon et al. 2000). Some researchers have used a feature-based modeling approach to extend BIMs for other domains (e.g., Staub-French et al. 2003). And finally, others argue for providing query facilities (especially spatial queries) in order to rapidly derive the required information and views of a design that satisfy the needs of different BIM users (Wong and Sriram 1999; Borrmann and Rank 2009). These approaches provide solutions that meet the needs for their particular purpose. However, each one of these approaches will not satisfy the requirements for the problem we are trying to solve. Specifically, they do not provide sufficient ‘flexibility’ to meet the varied requirements and preferences of construction practitioners, and they do not provide a ‘comprehensive’ solution that supports the needs of different construction management functions (e.g., cost estimating, site management, method selection, etc.).

This research aims to address these limitations by providing a novel approach that combines *feature extraction* with *query processing* to leverage BIMs for a broad range of construction management functions. This approach supports the automatic extraction and querying of construction-specific features from a given BIM according to the preferences of a particular practitioner or domain. Specifically, the approach has three distinct components: (1) an ontology of design features (i.e., a *feature ontology*) that explicitly and generically represents design conditions relevant to construction practitioners, (2) an instantiation of the feature ontology that results in a project-specific *feature-based model* (FBM), and (3) the formalization of *query specifications* and user-customizable *templates* that provide the language and structure to specify queries on a BIM.

In this chapter, we describe our vision and approach for the automated extraction and querying of construction-specific design features from a BIM. Specifically, we discuss the motivation and requirements for the proposed framework, describe relevant literature and points of departure, and finally describe the proposed framework and its components in detail.

## 2.2 Desirable Requirements of the Proposed Framework

Based on an extensive review of the literature, and numerous discussions with practitioners and research of case studies, we have identified four desirable requirements of the proposed framework. The motivating case study in Section 2.3 highlights many of these requirements as well.

- **Generic:** The requirement for the generality of this research tool is well documented (Haymaker et al. 2004; Staub-French et al. 2003; Li 2009). This approach should be general enough to extract and query different types of components and features, and should be useful for multiple construction management functions.
- **Formal:** The framework needs to have a structured, explicit, and computer-interpretable representation of concepts and computer-implemented methods in order to leverage the richness of digital information available in BIMs. The formal representation should include all the attributes necessary to support the automatic instantiation and querying from a BIM.
- **Flexible:** In order to accommodate the varying requirements of construction practitioners, information models require flexible representation of information (Haymaker et al. 2003; van Leeuwen and Wagter 1997; Katranuschkov et al. 2003). This implies that our representation of ontology and query specifications should be adequately flexible to accommodate construction practitioners' varied preferences or requirements for specifying queries.
- **Easy to Use:** In order to be accepted by construction practitioners, the developed system must be easy to use. Users should be able to configure queries according to their preferences without having to undertake programming.

### 2.3 Motivating Case

In this section, we describe a case study of the Chemical and Biological Engineering (Chem-Bio) building at the University of British Columbia (UBC). The Chem-Bio building provides a variety of teaching and research spaces for the study of biological, environmental and process engineering. This 123,000 square foot building includes two distinct structures: a six storey structure and two low-rise high-head laboratories. Here, we focus on the six-storey structure, concentrating mostly on major building components, including walls, columns, beams and slabs. Figure 2.2 shows two different views of a 3D model of the Chem-Bio building.

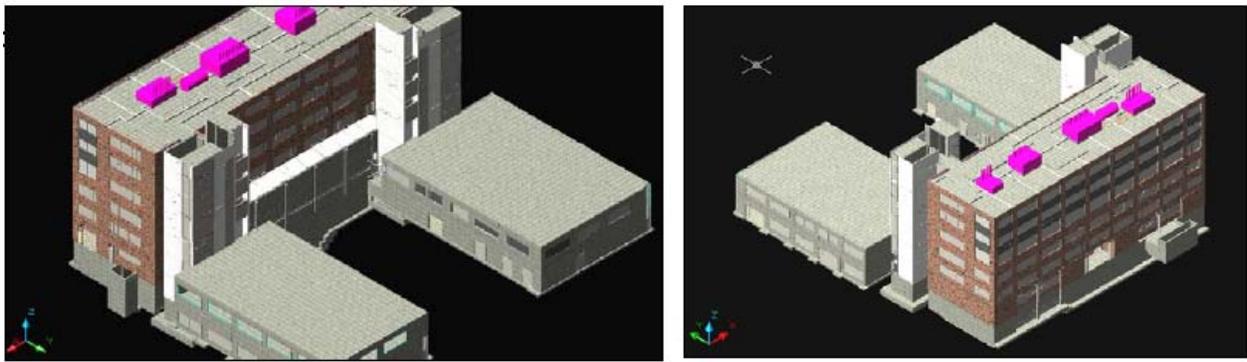
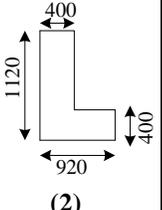
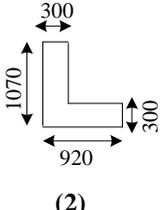


Figure 2.2 The 3-D model of the Chem-Bio building

The “columns” in the Chem-Bio building showed significant variation, as illustrated in Table 2.1. In particular, there is considerable *variation* in column *size* and *shape* and *location* both within a floor and from floor to floor. Columns located at the same *grid intersections* also have varying size and/or shape from floor to floor. There are changes in column *location* from floor to floor. The *orientation* of columns also changes on some floors of the building.

Due to variation in the size, shape, location, and orientation of columns in a floor or from floor to floor, the formwork contractor for this building would be motivated to: (a) find the *unaligned* columns in a floor and from floor to floor (i.e., check for horizontal and vertical alignment of columns); (b) locate *off-grid* columns, if any; (c) identify the maximum and minimum *spacing* of columns or bay sizes; and (d) identify the *uniformity* of column *location* and *size* from floor to floor.

**Table 2.1 Distribution of column sizes on different floors in the Chem-Bio building**

Column Type	Column Sizes in Floor						
	Basement	1	2	3	4	5	6
1	450x1200 (4)	400x1200 (4)	400x1200 (4)	400x1200 (4)	N/A	N/A	N/A
2	Do (2)	Do (2)	400x900 (2)	400x860 (2)	N/A	N/A	N/A
3	Do (2)	750 Φ (2)	400x1200 (2)	400x1200 (2)	N/A	N/A	N/A
4	350x2000 (6)	500 Φ (12)	500 Φ (12)	300x1500 (6)	300x1110 (6)	300x1110 (6)	300x1110 (6)
5	 (2)	Same as in the basement floor (2)	 (2)	Same as in the 2 <sup>nd</sup> floor (2)	Same as in the 2 <sup>nd</sup> floor (2)	Same as in the 2 <sup>nd</sup> floor (2)	Same as in the 2 <sup>nd</sup> floor (2)
6	300x1200 (4)	300x1200 (4)	300x1110 (4)	300x1110 (4)	300x1110 (4)	300x1110 (4)	300x1110 (4)
7	550x550 (2)	550 Φ (2)	300x1110 (2)	300x1110 (2)	300x1110 (2)	300x1110 (2)	300x1110 (2)
8	750x750 (2)	750 Φ (2)	300x1110 (2)	300x1110 (2)	300x1110 (2)	300x1110 (2)	300x1110 (2)
9	500x900 (4)	N/A	N/A	N/A	N/A	N/A	N/A
10	600x900 (1)	N/A	N/A	N/A	N/A	N/A	N/A
11	300x1100 (2)	300x1100 (2)	300x1100 (2)	300x1110 (2)	300x1110 (2)	300x1110 (2)	300x1110 (2)
12	N/A	N/A	N/A	N/A	450x900 (8)	450x900 (2)	450x900 (2)
13	N/A	N/A	N/A	N/A	450x750 (4)	450x750 (4)	450x750 (4)
14	N/A	N/A	N/A	N/A	450x750 (4)	600 Φ (4)	600 Φ (4)
15	500x500 (1)	500x500 (1)	500x500 (1)	N/A	N/A	N/A	N/A
16	N/A	500x500 (5)	500x500 (5)	N/A	N/A	N/A	N/A
<b>No. of columns</b>	<b>(32)</b>	<b>(38)</b>	<b>(38)</b>	<b>(26)</b>	<b>(34)</b>	<b>(34)</b>	<b>(34)</b>
<b>No. of sizes used</b>	<b>9</b>	<b>8</b>	<b>6</b>	<b>5</b>	<b>4</b>	<b>5</b>	<b>5</b>

The “walls” of the Chem-Bio building, as with many other building projects, are of varying *types* and are distributed in each floor. Figure 2.3 shows different wall type assemblies as defined by the architect for the main floor. Construction practitioners involved in this project, particularly the cost estimators, emphasized that they manually *re-categorize* and *re-organize* wall *types* according to their own preferred structure that correlates with their cost estimating database. For example, they filter and group all interior walls by type and height and then calculate the total linear foot and area for each wall type since different wall types may require different construction methods, productivity rates, or equipment for different ranges of wall heights. Table 2.2 lists some criteria used by construction practitioners for defining wall types.

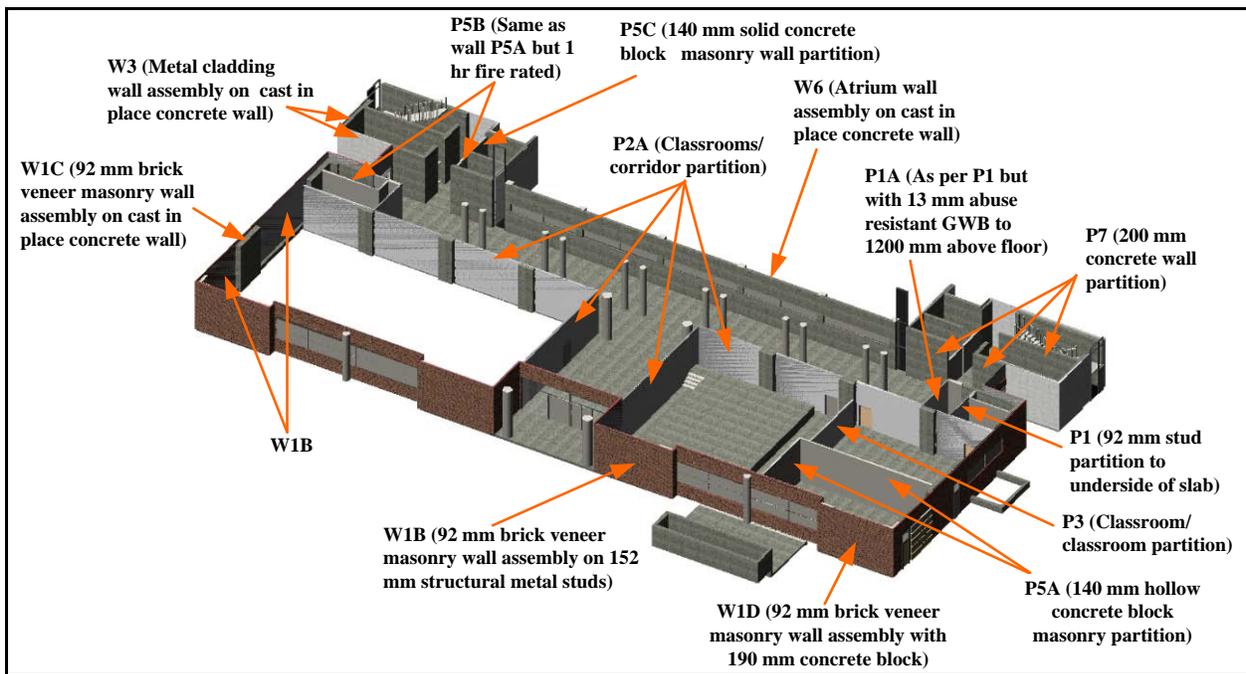


Figure 2.3 Different wall types defined for the Chem-Bio project

**Table 2.2 Different criteria for defining wall types**

<b>Criteria</b>	<b>Example/s (with typing <i>Italicized</i>)</b>
Generic wall name	<i>Masonry wall, Drywall, Concrete wall</i>
Constituent materials	<i>Steel stud drywall, Brick veneer wall</i>
Material properties	<i>Hollow concrete block masonry wall, Wall concrete-35Mpa</i>
Location as to interior or exterior of a building	<i>Interior steel stud walls, Exterior wall 8' to 16 high</i>
Shape (plan view)	<i>Straight wall, Curved wall</i>
Shape (elevation view)	<i>Vertical wall, Battered wall</i>
Change in height	<i>Clipped wall, Non-clipped wall</i>
Dimensions (height/length/thickness)	<i>190 mm Concrete block wall</i>
Wall height relative to slab and ceiling	<i>Full height wall, Ceiling height wall</i>
Location on the floor	<i>Basement wall-300 mm, Foundation wall-concrete block</i>
Location on the floor space	<i>Classroom wall, Corridor wall, Theatre wall-300 mm</i>
Generic wall properties	<i>Fire rated wall, Acoustically rated wall, Load Bearing wall</i>
Type of construction	<i>Precast wall panel, CIP concrete wall</i>
Wall function/usage	<i>Shaft wall, Core wall, Fire wall</i>

The drywall and mechanical contractors on the Chem-Bio project also needed to identify “components” with “openings” or “penetrations” (e.g., fire-rated dry walls with penetrations) as well as the “openings” and “penetrations” on specific types of components (e.g., all openings on masonry walls). The drywall contractor needed to identify the *size* and *extent* of openings and penetrations on walls and to determine if the *location* of those penetrations occurred below ceiling height. For the mechanical, electrical and plumbing (MEP) sub trades, the *location* and *size* of openings was required to determine if all ducts/conduits/pipes could fit into the provided opening and meet the required horizontal and vertical clearance from the adjacent wall(s) and slab above.

The masonry contractor was also concerned about the “intersection” of walls with other structural components, such as a slab, because many intersections imposed additional requirements, including fire rating, bond beam, and lateral supports, at the top of every wall. The formwork contractor, was especially concerned about the *type* and *nature* of “component intersections” (e.g., whether any partition walls were connected with round columns) because their existence could impact the constructability of a design.

The practitioners working for the general contractor and subcontractors in the Chem-Bio project manually analyzed and interpreted architectural, structural and/or mechanical drawings

and other design documents to identify the required and most relevant design conditions for constructability analysis, cost estimating, MEP coordination, and methods selection. The estimator for the general contractor spent a significant amount of time colour marking appropriate conditions using pdf drawings of the building plans in order to categorize components by material and other component properties (Figure 2.4 illustrates this process); when the design changed, the estimator had to manually re-calculate these conditions..

In summary, this case study illustrates that practitioners require automated computer support to quickly and easily identify a variety of design conditions with varied parameters to support different construction management functions.

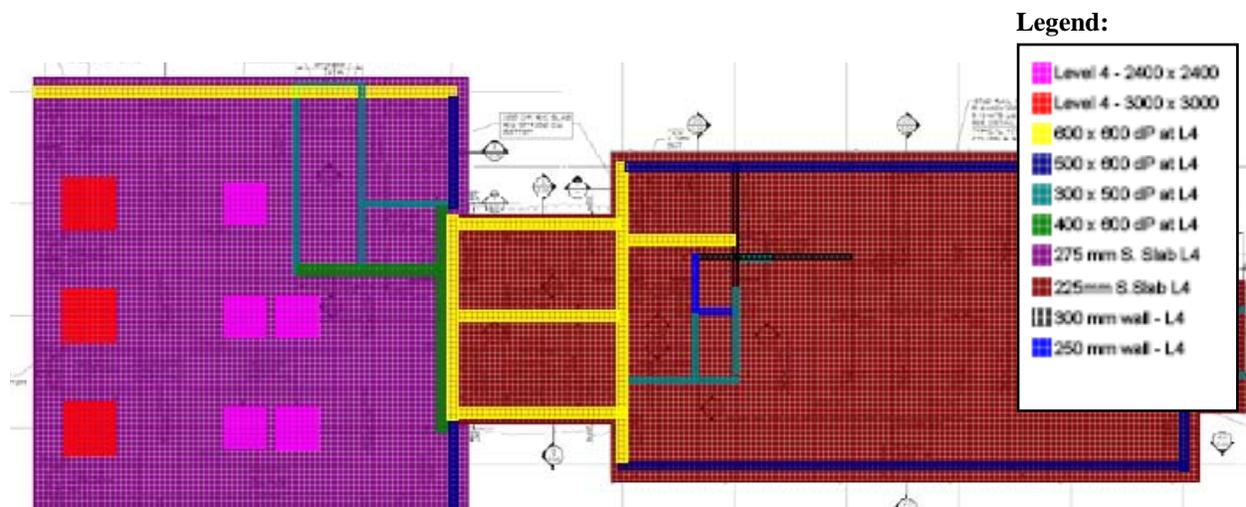


Figure 2.4 Color marking of different design conditions by an estimator on a pdf drawing

## 2.4 Related Work

This research builds on, combines, and extends previous research on design relevant construction knowledge and information, building product models, feature-based modeling, and product model reasoning.

### 2.4.1 Design Relevant Construction Knowledge and Information

Design relevant construction knowledge describes important features of a design of a building from a construction perspective, when these are important, and how they impact construction. It relates to spatial elements of a building (e.g., site, floor/storey), different building

systems, elements/components, and a number of design details. Researchers have used various terminologies, such as design factors, design features, design characteristics, design decisions, etc., to describe such knowledge; what we refer to as *design conditions*.

We conducted an extensive review of the literature that identifies these different design conditions because they:

- Influence the applicability and suitability of a particular construction method, such as in the selection of formwork systems in concrete structures (Fischer and Tatum 1997; Hanna et al. 1992; Thomas and Zavrski 1999) and on the constructability of a design (Boeke 1990; Glavinich 1995; Burkhart et al. 1987; Skibniewski et al. 1997; Ugwu et al. 2004);
- Impact labour productivity (Smith and Hanna 1993; Sanders and Thomas 1991; Thomas and Zavrski 1999);
- Relate to cost estimating via their impact on resource/method selection, construction activity requirements, and productivity (Hanna and Sanvido 1990; Thomas and Sackrakan 1994; Staub-French et al. 2003); and

We have summarized and compiled a partial list of some important design conditions identified in the literature which impact construction (see Table 2.3; the numbered references are listed in Appendix A), and structured them into different categories for the purpose of clarity and comprehensibility. The intent is to exemplify the varied nature and characteristics of design conditions. These design conditions relate to different building components and their characteristics, intersections of components and their characteristics, the existence and the characteristics of openings and penetrations on components, among others. They also occur at different levels of detail. For example *spacing* is important for beams, columns, etc. but it can also be relevant at a more detailed level, such as the spacing of rebar within a component.

**Table 2.3 Summary of design conditions identified in the literature that impact construction**

<b>Design Entities</b>	<b>Design Conditions</b>	<b>References</b>
<b>Component Characteristics in General</b>	Component dimensions (e.g., height, depth, width, thickness, length)	[1,2,3,4]
	Maximum/minimum dimensions and spacing of components	[5]
	Component location in a floor (e.g., below grade, main floor, top floor)	[6,7]
	Modular layout of components	[8]
	Adjacency of building components	[9,10]
	Repetition of component dimensions/sizes and distances in a floor and from floor to floor	[1,2,3,8,11]
	Shape of components (e.g., round column)	[2,3]
	Existence of blockouts, bulkheads, pilasters, drop heads	[2,3]
	Irregularity in the shape of components	[2,3,4]
	Changes in dimensions, shape, size/cross section of structural components	[1,2,3]
	Type of component material/s	[12,13,14]
	Variation in the size and location of components (e.g., columns, structural walls)	[1,2,3,15]
	Component spacing	[1,16]
	No. of components attached or connected to a component	[17]
Material characteristics of a component (e.g., concrete strength)	[1]	
<b>Component Parts/Details</b>	Component parts	[12,14]
	Characteristics of component parts (e.g., clear spacing of rebars, size of masonry units)	[5, 6,14, 17,18]
	Component finish type	[4,12]
	Type of connections	[13]
<b>Component - Slab</b>	Variation in the dimensions of the slab-bay	[1,7]
	Type of slab (e.g., flat slab, ribbed slab, waffle slab)	[2,14]
	Change in elevation of the bottom of a slab	[2]
<b>Component - Beam</b>	Irregularity in the shape of beam (e.g., tapered end beams)	[2,3]
	Variation in the size, shape and location of beams	[19]
	Spacing of beams	[13]
	Existence of drop beams	[2]
<b>Component - Column</b>	Existence of column head/drop panels	[2,8]
	Uniformity in the layout and spacing of columns	[20]
	Variation in the size and location of columns from floor to floor	[15]
	Horizontal and vertical alignment of columns	[5,21]
<b>Component - Wall</b>	Wall type	[4,6,16,22]
	Presence of sloped walls	[12]
	Straight walls	[6,12]
	Presence of curved walls	[1]
	Exterior or interior walls	[6]
	Ceiling or full height walls	[16]
	Length of wall; minimum and maximum wall length	[1, 4,6,12]
	Wall height	[1,4,16,22]
	Wall curvature	[22,23]
	Existence of wall corbels, ledges, and pilasters	[12,23]
	Variation in the size, height and shape of walls	[12]

<b>Design Entities</b>	<b>Design Conditions</b>	<b>References</b>
<b>Component- Intersection</b>	Intersection or connectivity of building components (e.g., a beam connected to a column)	[3,5, 9,12,17,24]
	Wall turns (corners)/no. of wall turns	[6,22,23]
	Wall to wall intersections	[4]
	Intersection of masonry wall with structural steel elements	[6,12]
	Non-perpendicular wall turns (orientation of wall turns)	[6,12, 22]
	No. of intersecting components	[6,17]
	Property of intersecting components (e.g., material type)	[12]
	Type of intersecting components	[12]
	Relative dimension of intersecting components	[2,5,13,25]
	Complexity of intersection (e.g., complex slab-beam intersection)	[3]
	Horizontal location of intersection	[24]
Reinforcement ratio of the attached or connected components	[17]	
<b>Penetration</b>	Existence and extent of component penetrations	[12,16]
	Vertical location of wall penetrations	[11,16]
	Uniformity in the location of wall penetrations	[11]
<b>Opening</b>	No. of wall openings (e.g., window and door openings)	[12]
	Existence of openings	[6]
	Size and location of component openings	[1]
	Uniformity in the size and location of openings	[1,4,12]
	Uniform spacing of openings	[12,19]
<b>Floor/Storey</b>	Storey/floor height	[1,15]
	Change in floor height	[1,2,15]
	Floor area/floor plate	[1,6]
	Repetition of floor to floor height	[8]
	Repetition of horizontal grids (between supports)	[8]
	Change of floor layout from floor to floor	[1,3,8]
	Vertical repetition of floor layout	[3]

This research has tried to incorporate much of the broad range of design conditions identified in this table, though we do not claim to support all of these. It is also worth noting that many of these design conditions apply to many components. Although we focus more on architectural and structural elements, we believe that our approach, in many situations, would work equally well for building service components. Consider the installation, operation and maintenance of building systems and their components, particularly MEP systems (Korman et al. 2003; Tabesh and Staub-French 2006), which relate to the positioning, layout and the spatial relationships between MEP components. Based on our analysis, we found that there exist many commonalities among design conditions between building services components and architectural/structural components. For example, the location, spacing, uniformity of design, size/dimensions, type of component, type of connection, and material characteristics of

components are important to both MEP and architectural/structural components. As such, the framework presented in this research can be equally extended or applied to building services components as well.

#### **2.4.2 Building Product Models**

Building product models are essentially the same models as BIMs (Howard and Björk 2008). They develop generic and specific schemas for representing building product information which represent different types of information: generic entities or objects (e.g., space, slab, column, wall), composition/decomposition (or object) hierarchy, attributes (height, length, geometric representation), and relationships (e.g., part-of, connected-to) pertaining to all aspects and stages of the construction of a building. The research concerning BIMs started to gain momentum around 1985, with the *ISO STEP* standardization project (Howard and Björk 2008). Early efforts to develop generic building models include the *RATAS model* (Bjork 1989), global *AEC reference model* (Gielingh 1988) and the *building systems model* (Turner 1990). Later research implemented some of the general concepts of these models to a specific application area {*COMBINE* (Augenbroe 1994); *CIMSTEEL* (Watson 1995)}.

*Industry Foundation Classes (IFC)* is the primary product model exchange standard for the AEC/FM industry and has been built on extensive previous product modeling and standardization efforts. As an open BIM, *IFC* provides detailed BIM specifications, representing semantically-rich information pertaining to the life cycle of an AEC/FM industry, using a high level, object-oriented data model, based on class definitions. It can be used to exchange and share BIM data between AEC/FM computer applications by importing or exporting data to *IFC* file(s) without the necessity to support numerous native formats (*IAI* 2010).

No single product model schema, however, can satisfy all practitioners' tasks, since different practitioners conceptualize AEC projects in different ways (Turk 2001). In addition, it is impossible to foresee all possible usages of a model and requirements of downstream disciplines in the building design models (*Graphisoft* 2004). In this research, we leverage and employ the relevant pre-defined schemas and entities (objects, properties, and relationships) defined in the *IFC* model and BIM authoring tools in order to extract other relevant information useful to construction, which is not explicitly defined in the *IFC* model or BIM authoring tools.

### **2.4.3 Features and Feature-Based Modeling**

The concepts of ‘feature’ and ‘feature-based modeling’ (FBM) first surfaced in the mechanical and manufacturing industries to describe the shape and geometry of parts that constitute an assembly and are used in reasoning about the topology and geometry of designed artefacts during various design and manufacturing activities (Cunningham and Dixon 1988; Shah 1991). There are essentially two approaches in the FBM technique: design with features, and feature extraction (or recognition). The first approach uses primitive and user-defined features to generate higher level features to develop a design. The second approach recognizes and/or extracts features from the already designed geometric models (normally the solid model) and is analogous to our approach, which extracts features from BIMs.

Several researchers in the AEC domain use features to represent design information, such as to model architectural design (van Leeuwen and Wagter 1997), structural design (Zamanian et al. 1991) and to develop a core product model (Fenves 2002). Others have used features for specific construction applications (Haymaker et al. 2004; Boukamp 2006). Features, in the context of our research, denote a set of information entities related to building components, with a semantic meaning which are integral to a particular view or domain of construction. We build on and extend the feature-based representation of design in the manufacturing and mechanical domains to the building construction domain, where “feature” represents components, the intersection of components, and the attributes that characterize them. We then use ‘feature extraction’ to identify these features in a given BIM based on the geometric, symbolic and topological information in the BIM to create a feature-based model for construction.

### **2.4.4 Related Research on Product Model Reasoning**

Considerable research efforts have been devoted to reasoning about product models using model-based reasoning (MBR). MBR has the ability to abstract graphical, geometrical, topological, and behavioural characteristics from the components in the product model for the reasoning processes (Kunz et al. 1996). Approaches using MBR add representation schemas and use task-specific reasoning structures to construct specific views from building product models. Some related studies in this area include: Fischer (1991); Dym et al. (1988); Akinci et al. (2002); Han et al. (2000); and Akbas (2004). Generally, all of these model-based reasoning systems use task-specific purpose and are limited to generate views intended only for the knowledge engineer

or programmer (Haymaker et al. 2004). They fall short of providing flexible and easy-to-use query support in the generation or derivation of a particular view containing construction-relevant information.

BIMs (e.g., *IFC* data model) formalize pre-defined schemas that could be used for reasoning about different concepts, explicitly defined in the conceptual models. These models, however cannot foresee all possible usages of a model and requirements of downstream disciplines. Sufficient reasoning mechanisms are needed to derive a particular view from these models. As for *IFC*, it is not intended to support a particular discipline or the application requirements of such a view. The *IFC* data model focuses on classes needed to share, and exchange information, rather than processing it in a particular proprietary software application (IAI 2000). Some studies add reasoning dimensions to *IFC*, such as the work by Katranuschkov et al. (2003), which provides a mechanism for browsing and navigating *IFC* data.

Query languages and query-based approaches can be used to retrieve information and generate task-specific views out of a product model. They act on predefined model schemas or they support the definition of schemas to query a product model database. Mechanical engineering research (Lou et al. 2003) investigates generic CAD query languages which enable engineers to query a model for geometric features. It does not, however, provide simple, generic and explicit support to derive construction relevant features from BIMs. Other efforts have attempted to provide query support on BIMs, such as in the Partial Model Query Language (Adachi 2003) and Product Model Query Language of the *EuroStep Model Server*. These retrieve explicitly defined *IFC* properties and relationships limited to simple containment relationships. Recent research explores the development of a 3D spatial query language to enable the spatial analysis of objects in a BIM (Borrmann et al. 2006), but it provides limited support to extract the kind of information needed by practitioners, as described in the motivating case. An example of this involves identifying specific type of feature based on its properties and its topological relationships with other features.

We build on and extend previous research efforts dealing with product model reasoning. We use the combination of MBR, feature-based modeling and a query-based approach to extract and query features from the design-focused BIM.

## 2.5 Research Framework

Figure 2.5 presents our research framework in the form of an *IDEF0* model (Colquhoun et al. 1993) for extracting and querying construction features from a BIM. *IDEF0* (*Integrated Definition for Function Modeling*) is a modeling technique designed to model the decisions, actions, and activities or functions of an organization or system. The model shows the identifiable functions, information that controls that function, input information to execute the function, and the distinct output from the execution of that function. It also symbolically presents the end user's action(s) or role(s) and computer action(s) or reasoning method(s) required.

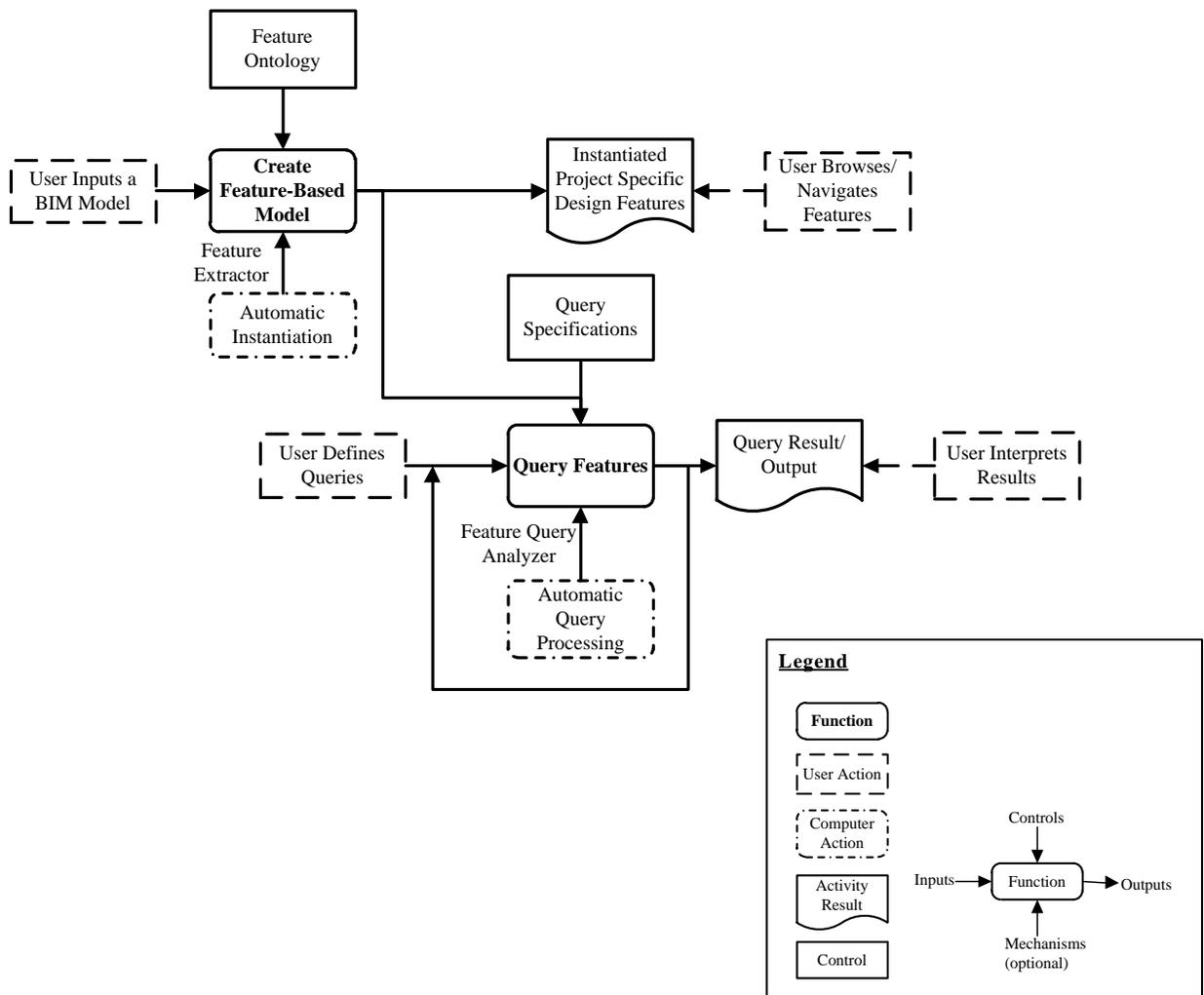


Figure 2.5 A generic research framework for extracting and querying a BIM

The two main functions of the framework are to (1) *create a feature-based model* and (2) *query features*. We use the example scenario highlighted in Section 2.3 to illustrate these functions and to describe how our system helps users to identify the information required. Consider a scenario in which a dry-wall contractor wants to explore a BIM to find walls with penetrations, essentially requesting the query: “*Identify all fire-rated dry-walls with penetrations*”.

1. Create Feature-Based Model: Our system helps guide the user to load the pre-defined feature ontology and input the *IFC*-based BIM model of a building. The feature ontology (described in more detail in Section 2.5.1.1) generically defines and represents different design conditions relevant to practitioners and in terms of features and their attributes (feature properties or relationships between features). Specific to the above example, the feature ontology describes both “wall” and “penetration” as distinct features. The ontology describes different characteristics of “wall” and “penetration” features as properties. For instance, the properties “*Is-FireRated*” and “*Material*” characterize the feature “wall”; the relational property “*has penetration*” associates the feature “wall” with the “penetration” feature. The system uses the application *Feature Extractor* to analyze the input *IFC*-based 3D model of the project and identifies or extracts feature instances and property values defined in the feature ontology (Section 2.5.2.1 describes the feature extraction process in more detail). We term this process as creating the feature-based model, and the output of this process is the project-specific feature-based model (FBM). It is the domain-specific secondary representation of the design in terms of features that is of principal concern to the domain experts. The user is able to browse and navigate the FBM. For example, users can click each instance of the feature “wall” and navigate the wall properties of interest, or examine the property “*has penetration*,” to locate wall instances with penetration(s).

2. Query Features: In many situations, practitioners ask specific questions and expect answers that are filtered, organized, documented or quantified in a way that is useful for their particular purpose. The function “Query Features” enables the user to specify a query in terms of simple, readable, and interpretable language. Query specifications (explained in detail in Section 2.5.1.2) describe the necessary vocabulary or language for specifying different types of queries, which

are formally encoded into sets of computer interpretable templates. The user chooses the appropriate query templates, follows pre-defined query steps and, for each step, defines query-relevant attributes to define a query (to be described later in Section 2.5.2.2). The system takes the query input from the user along with its relevant attributes and executes the application *Feature Query Analyzer*. The *Feature Query Analyzer* searches the FBM and, depending on the type of query, analyzes other data extracted from underlying BIMs to answer the query. The query results are displayed in a variety of formats, including tabular, text, and graphical displays.

We worked with computer scientists to implement this framework. The prototype applications were created by Zhang (2008) and Webster (2010). The user interface and most of the backend were written in Java. A small subset of functions, as well as the *IFC* converter, was written in C++. These components are called by the Java program. We used the standard XML query language, XQuery (W3C 2010), to query features. The Java programs translate the user-created queries into XQuery expressions, which are then analyzed and manipulated to answer queries.

The framework is unique in that it combines model-based reasoning, feature-based modeling, and query processing in reasoning about a BIM. Model-based reasoning (MBR) uses a 3D model of a building as the basis for reasoning and allows the user to abstract relevant geometric, topological, and other product data (e.g., material properties) and attributes in order to facilitate knowledge extraction. This is the reasoning method used in the feature extraction process. The query-based approach uses query languages, or predicates, to construct schemas using domain models. This means that the user defines the task-specific view(s) in terms of dedicated queries. A query-based approach adds flexibility because the user does not need to define all construction features *a priori*, but can use queries on features to extract the information needed.

This approach also demonstrates how domain models could be used as a gateway to extract and query pertinent information from a BIM. Domain models, in the context of our research, represent pre-defined model schemas, defined in the feature ontology and query specifications and templates. We created the ontology and query specifications to encode the domain users' view on design in a formal and machine readable way, and also to have the application automatically create mappings between domain concepts and the BIM model. The granular representation of concepts is the key to representing construction-specific domain

knowledge. The types of features, feature attributes, and query attributes represent atomic concepts to allow domain experts to compose and combine them according to their preferences at the required level of specificity. For instance, the framework not only enables the user to find any type of column based on its attributes, but also facilitates reasoning about higher-order knowledge, such as column alignment, spacing of columns, and uniformity of columns' location. The atomicity and flexibility of knowledge represented in the feature ontology and query specifications assist construction practitioners to define queries. For instance, the system does not merely identify intersections between features (e.g., wall to column) but allows options for defining conditions, to find intersections between specific types of features. An example of this is the intersection between round column and dry-wall. Our framework provides practitioners with an easy-to-use and intuitive approach for manipulating the BIM and configuring queries without necessitating any programming or knowledge of query languages, such as *XQuery*.

We differentiate the presented framework into two key aspects: representational structures and reasoning modules. A representational structure formalizes knowledge and represents it in a computer interpretable way. In the context of this research, the formalization of the feature ontology and query specifications serve this purpose. This knowledge provides a controlled vocabulary and the domain knowledge necessary for searching a BIM database, extracting relevant information, and processing queries. Reasoning modules, represented as “functions” in the *IDEF0* model, leverage formalized knowledge in the feature ontology and query specifications to carry out reasoning functions. They automatically extract design features and interactively query design features, respectively.

### **2.5.1 Knowledge Structures**

Knowledge structures provide the systematic schematization or organization of knowledge of a domain to represent information about the domain and make inferences about it (Galambos et al. 1986). In this research, a feature ontology and query specifications provide knowledge structures, or schemas, for extraction and querying of design features from a BIM. They control the functions for creating the feature-based model and for querying features.

### 2.5.1.1 Feature Ontology

Ontologies provide a means of defining or representing knowledge about a domain of interest, and include a set of concepts, their definitions, relationships and semantics (Genesereth and Nilsson 1987). They enable the reuse, sharing, exchange, and analysis of domain knowledge, and provide a sound framework for the structure of information, among users or via computer systems. They define the vocabulary with which queries and assertions are exchanged among agents (Gruber 1995).

Several construction organizations and research efforts have focused on developing the common vocabulary or ontologies of building and construction information. A framework for the organization and classification of building construction information (*ISO 2001*) and various existing classification systems, such as *Uniformat™* and *Master Format™* (CSI 2008) and the *Overall Construction Classification System™* (OCCS 2008) provide standardized concepts and terminologies for the AEC/FM industry. They do not, however, explicitly represent design conditions in a format to automatically extract them and enable the formulations of customizable and user-driven queries. Several researchers have worked on developing a taxonomy or vocabulary of building and construction information (Woestenenk et al. 2000; El-Diraby et al. 2005). Others have focused on developing application-level ontologies for a specific purpose, such as the constructability reasoning for steel frame structures (Ugwu et al. 2005), or to support construction cost estimating (Staub-French et al. 2003).

Our research leverages design conditions identified in the existing research for developing an ontology of design conditions (termed the feature ontology) and query specifications. This is an application-level ontology which builds on and extends the feature ontology developed by Staub-French et al. (2003) to support the automatic extraction and querying of a variety of design conditions that could be relevant for construction practitioners using BIMs. The research challenge to develop a feature ontology includes the following: (a) individual construction practitioners use different vocabulary for describing design conditions; (b) many different design conditions impact construction; and (c) design conditions are related to various levels of detail, such as the building, system, or component details (e.g., connection, joints) of a building. Moreover, ontology must be at the appropriate level of generality to be useful (Uschold et al. 1998).

The feature ontology explicitly defines a vocabulary or concepts to describe a broader set of design conditions relevant to construction practitioners. It generically and flexibly defines concepts in order to provide a common language for practitioners from different construction management functions to characterize a design and to account for the varying needs and preferences of practitioners. The feature ontology also acts as a communication and specification medium between different users including users and system developers, for understanding and specifying the requirements for end-user applications; users and computational systems; and different computational systems.

This research considers building components as the core level of analysis in developing the feature ontology because it allows for conceptualizing design conditions at varying levels of detail. Based on the motivating case and a thorough review of related research, we classify features into two types: *component* and *intersection*. The component feature refers to common building elements which constitute parts of the actual building being constructed. It is a generic concept and is further categorized into subclasses representing more specific concepts such as wall, column, slab, beam, ceiling, roof, etc. The feature type *intersection* is defined to mean the physical/geometric interaction between components or elements of a building which results in the formation of intersections between components. We further categorize intersection features into three types: *opening*, *penetration* and *component*. These features characterize the type and the nature of components, or elements, involved in the intersection relationship.

The feature *opening* refers to door opening, window opening, and other types of openings that may exist on component features. We use the feature *penetration* to describe an intersection relationship involving building service elements and components, such as duct penetrations in a wall or slab. We use the feature *component intersection* to describe intersections of building components that may involve different conditions, such as a component attached to another component (attachment), a component connected to, or connected from, another component (connectivity), a component overlapping another component, a component meeting/touching another component, a component crossing another component, a component supporting or supported by another component, or those components related within a defined tolerance value. Component intersections can occur between the same types of components (e.g., wall to wall intersections) or between different component types (e.g., wall to column intersections).

We formally represented the feature ontology in *Protégé Frame Editor*, an open-source, ontology development platform. *Protégé-Frame* provides a suite of tools for building domain models and knowledge-based applications with ontologies (*Protégé* 2008). The frame-based ontology consists of a set of classes organized in a subsumption hierarchy to represent a domain's salient concepts, a set of slots associated to classes to describe their properties and relationships, a set of facets that describe properties of slots, and axioms to specify additional constraints. As shown in Figure 2.6, different feature types are represented in a hierarchical structure. We define, for each feature type (or class/subclass), different properties (relationships are also treated as a type of property in this research) and their data type. Several relational properties (e.g., *has opening*, *has penetration*, and *intersects*), establish links between different feature classes (for instance, the property *has opening* links component feature with opening feature). New properties can also be defined for each subclass.

One of the major characteristics of the feature ontology is its explicit and flexible representation of design conditions as discrete properties and/or relationships of features. Such representation not only provides the flexibility to query a BIM, but also enables users to flexibly define features or component types (e.g., wall types) during query run time, without altering the original structure of the ontology. Such a flexible representation and the requirement for such capability, as illustrated in the motivating case, allows the evolution and adaptation of information models to accommodate the specific requirements of the end users (van Leeuwen and Wagter 1997).

The screenshot displays the Protégé 3.4 interface for editing a feature ontology. The main window is titled "Feature Ontology\_August 2011 Protégé 3.4". The interface is divided into several panes:

- CLASS BROWSER:** Shows the class hierarchy for the project "Feature Ontology\_August 2011". The hierarchy is:
  - :THING
  - :SYSTEM-CLASS
  - Feature
    - Component
      - Beam
      - Column
      - Slab
      - Wall
    - Intersection
      - Component Intersection
      - Opening
      - Penetration

- CLASS EDITOR:** Shows the editor for the class "Wall". The role is set to "Concrete".
- Slot Definition Window:** A pop-up window titled "is curved at class Wall (instance of :STANDARD-SL...)". It defines the slot "is curved" with the following properties:
- Name: is curved
- Value Type: Boolean
- Documentation: "When the wall axis or base line of the wall is curve, the wall is said to be curved wall. If the base is an arc or the whole circumference of a circle, or a portion or the entire"
- Cardinality: at most 1
- Domain: Wall
- Template Slots Table:** A table listing various slots for the "Wall" class. The table has columns for Name, Cardinality, and Type.
 

Name	Cardinality	Type
acoustic rated	single	Boolean
acoustic rating	single	String
area	single	Float
ceiling height wall	single	Boolean
contained in the story	single	String
curvature	single	Float
fire rated	single	Boolean
fire rating	single	String
forms intersection	multiple	Instance of Component Intersection
full height wall	single	Boolean
has opening	multiple	Instance of Opening
has penetration	multiple	Instance of Penetration
height	multiple	Float
intersects with component	multiple	Instance of Component
is clipped	single	Boolean
is curved	single	Boolean
is exterior	single	Boolean
is interior	single	Boolean
is load bearing	single	Boolean
is sloped	single	Boolean
is straight	single	Boolean
is vertical	single	Boolean
length	single	Float
material	multiple	Symbol
thickness	multiple	Float
volume	single	Float
wall type	required single	String

Figure 2.6 The feature hierarchy represented in *Protégé Frame Editor*

### 2.5.1.2 Query Specifications

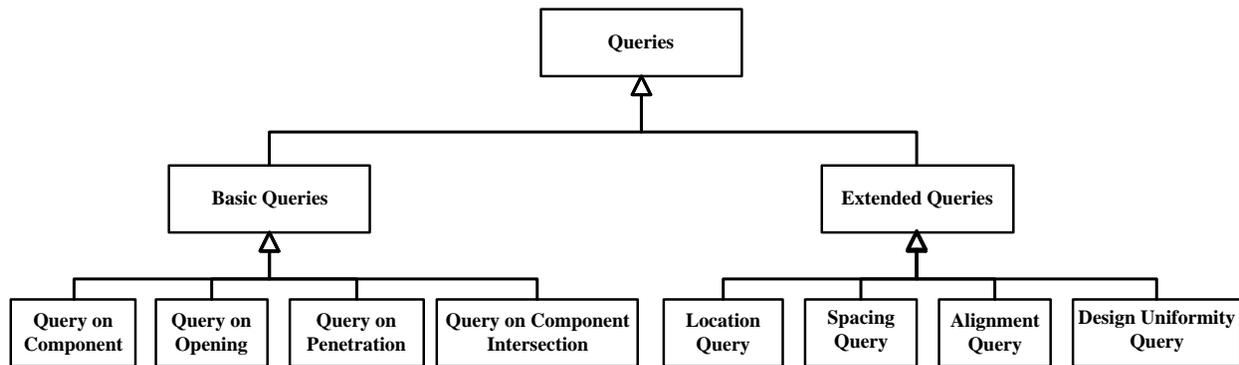
Query specifications provide controlled and structured vocabularies to specify queries. They represent the underlying domain knowledge for the formulation and processing of different types of queries. The knowledge needs to be represented and structured to enable users to flexibly define queries. Query specifications should ideally allow practitioners to easily and intuitively formulate queries without the knowledge of the *IFC* model, underlying data models of a BIM, or query languages. The research challenge with respect to formalizing query specifications is that construction practitioners require different types of queries, have different ways of expressing queries, and different levels of knowledge specifications are needed for describing queries. Our goal with this research is to provide a formal and structured way to specify queries on features formalized in the feature ontology and represented in the FBM. This is achieved by defining query vocabularies and their attributes, and by developing computer interpretable query specification templates.

We classify queries into two major categories: *basic* and *extended* queries (see Figure 2.7). Basic queries relate to the manipulation of feature properties in the feature ontology and their instantiated values in the FBM. They extend the functionality of the FBM to enable users to query component, opening, penetration and component intersection features. Extended queries expand on the basic queries and act on concepts that are not explicitly represented in the feature ontology and resulting FBM. Such concepts include spatial and other higher-level concepts, which currently include the concepts of spacing, location, alignment and design uniformity.

**Location queries** are used to identify the location of features relative to some frame of reference, such as the location of columns with respect to related grid lines. Being able to quickly identify the location of features is critical for construction planning, constructability assessment, and facility management (Allen and Iano 2002). For instance, knowing the location of all of the penetrations will result in more accurate cost estimates of the work (Bisharat 2004).

**Spacing queries** are used to identify the distance between (or spacing of) proximate features of the same type or different types. Spacing between similar features, such as column spacing, opening spacing, and so forth, can impact construction planning, constructability assessment and formwork method selection (Fischer and Tatum 1997). Spacing between dissimilar features (e.g., spacing between walls and adjacent ducts) can be important during the

construction and operation of the facility (Korman et al. 2003; Tabesh and Staub-French 2006). A spacing query might be asked as: “*Identify the maximum and minimum clear spacing between proximate columns.*”



**Figure 2.7** A taxonomy of queries

**Alignment queries** are used to identify the orientation and/or placement of the instances of a feature with respect to some criteria. The purpose is to find the unaligned features, if any, that may be present in a given design. The proper alignment of features, such as the column alignment in a floor, or from floor to floor, is crucial for the constructability of a design and installation of the façade and curtain walls (Fischer and Tatum 1997; Allen and Iano 2002). A potential alignment query is: “*Identify columns that do not align horizontally (i.e., in a particular floor) or vertically (i.e., floor to floor).*”

**Design uniformity** queries are used to identify or gain insights about the consistency (or variation) of design features on a particular building floor or from floor to floor of a building. The identification could be done in terms of feature properties {e.g., the size and/or dimension (width, thickness, diameter, length, depth, height)} or other extended properties (e.g., the location and spacing of features, the variation in the openings’ size and location on walls, changes in columns’ size and location from floor to floor).

For each type of basic and extended query, we formalize the attributes and in some cases, the attribute values, to capture the nuance for how practitioners think about the design. These attributes enable practitioners to state their varied preferences and customize the unique

construction requirements in the query specification process. These attributes provide a generic language and vocabulary to describe and specify different types of queries on features.

## 2.5.2 Reasoning Modules

As mentioned earlier, reasoning modules, represented as “functions” in the *IDEFO* model carry out reasoning functions to automatically extract and query features. We briefly describe the characteristics of these modules below.

### 2.5.2.1 Extracting Features to Create a Feature-Based Model

The feature ontology acquires its project-specific context when it is applied to a particular design model or BIM. The result is a project-specific Feature-based Model (FBM), which is the secondary representation or view of a design populated with project specific feature instances and properties. We developed a prototype system, the *Feature Extractor* to automate the extraction of features in order to create an FBM. The input data to the system comes from an *IFC*-compliant 3D BIM model, in our case, a *Revit* Model, and is in ifcXML format, an XML representation of *IFC* data. This format was chosen among four candidate standards (*Microsoft Access*, *ifcXML*, *gbXML* and *DWF-content XML*), all supported by *Revit*. As such, building design data must be converted into an ifcXML file. In our case, we exported the *Revit 3D* model to an *IFC* file using *Revit*'s export mechanism. The *IFC* file was then converted to an ifcXML file, using a converter based on *IFC2x2/ifcXML2x2* specifications. While ifcXML is the best choice, due to its higher expressivity, given the standards considered in this study, there are still several user challenges when working with ifcXML data.

When the user loads an *IFC* file of the BIM model, our system automatically converts the input model into (an) ifcXML file(s). The *Feature Extractor* establishes mappings between features in the feature ontology with the corresponding objects in the *IFC*-based BIM model, and analyzes the *IFC* attributes, property sets, the geometric representation of objects, and topological relationships between objects in the input model. It automatically creates the XML version of the FBM and subsequently translates to the FBM view. The result is the creation of a project-specific view of a design in terms of instantiated features that are present in a given design and that is useful to practitioners. Since BIM models are normally designer-focused, the

creation of an FBM allows construction practitioners to transform the designer-focused BIM model into a construction-focused, project-specific model.

The FBM is organized and presented in a view such that practitioners can browse and navigate project-specific features, trace the relationships between features, and gain insights about a design. The FBM interface should ideally include all basic browsing and navigation capabilities needed for features, including 3D visualization. Currently, a number of applications provide support for users to browse BIM models. *IFC* browsers, for example, help users to browse *IFC* building elements and their attributes. *Solibri Model Checker* provides a rich view of a design by adding additional properties, such as location information, quantities, and some relationships. This research shares similar attributes with these browsing applications, but extends them in order to create a project-specific view of a BIM for design features relevant to construction practitioners.

The FBM presents a hierarchical view of the instantiated features which are organized first by level and then by the hierarchy specified in the feature ontology. Figure 2.8 shows the view of the FBM for a sample model. The user can navigate feature instances and view detailed information about their properties, as defined in the feature ontology. The explicit representation of features (e.g., penetrations) and feature properties (e.g., different wall shapes, such as curved wall, clipped wall, etc.) serve to enrich a BIM as such information is not always explicit in a given 3D model, and the resultant *IFC* export.

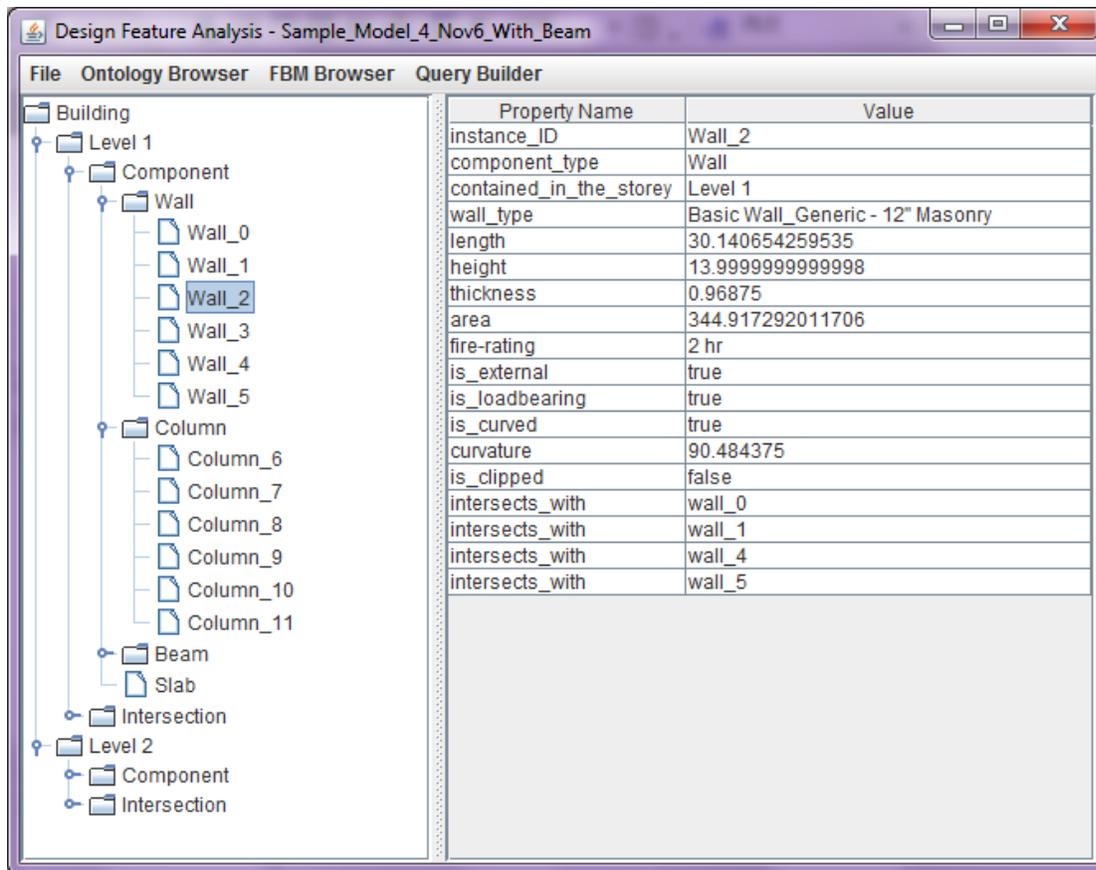


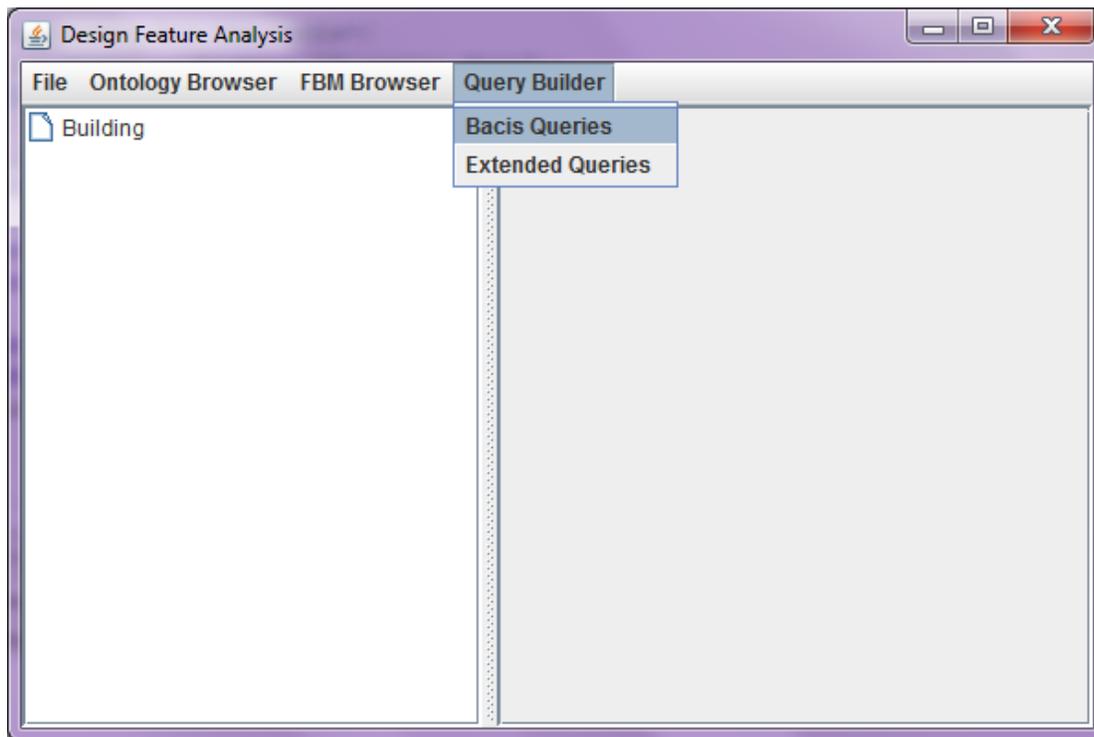
Figure 2.8 The FBM and browsing of features of a sample model

### 2.5.2.2 Querying Features

Query modules allow the user to run queries on an input BIM model. We developed another prototype system, the *Feature Query Analyzer* to automatically process queries according to the taxonomy of queries and query specification definitions. We developed query specification templates in order to facilitate the querying process and fulfill practitioners' specific requirements for describing and formulating queries in a consistent, unambiguous, and computer interpretative way. The form-based templates are based on the taxonomy of queries and query specifications described in the previous section. They explicitly organize and represent domain knowledge about different spatial and non-spatial queries. They are structured in such a way which allows practitioners to customize their varying needs, rational or preferences for important design conditions. The templates correspond to the features and feature properties/relationships in the feature ontology, and the query taxonomy formalized in the query specifications. They provide highly structured and guided assistance to specify queries, using

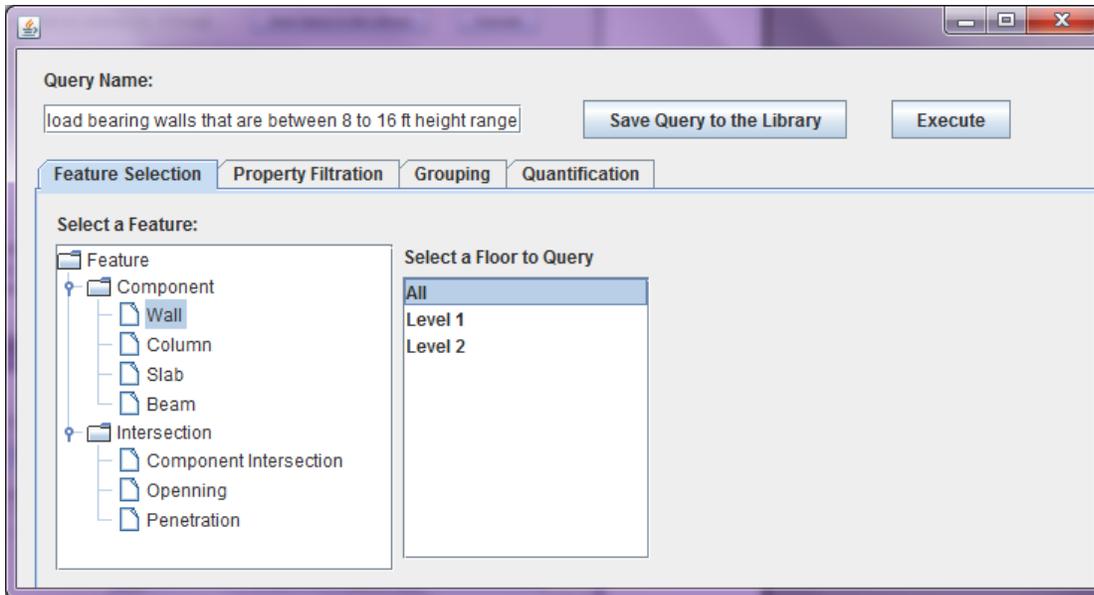
predefined fields and step-by-step methods that are explicitly presented. The user can select a query from the query library or formulate a new query, based on the predefined concepts and parameters in the query specifications.

To perform the analysis, the user first loads a new model or runs the existing model in the system by using the “File Menu” in the user interface. The FBM model is automatically created by the system, as explained in the previous section, and is found under the “FBM Browser” menu. To query for a feature, the user selects the ‘The Query Builder’ menu and then selects either “Basic” query or “Extended” query (Figure 2.9). We present the following query as an example to illustrate common query steps and the query templates involved for specifying basic queries: “Identify all load-bearing walls that are between 8 to 16 ft. height range.”



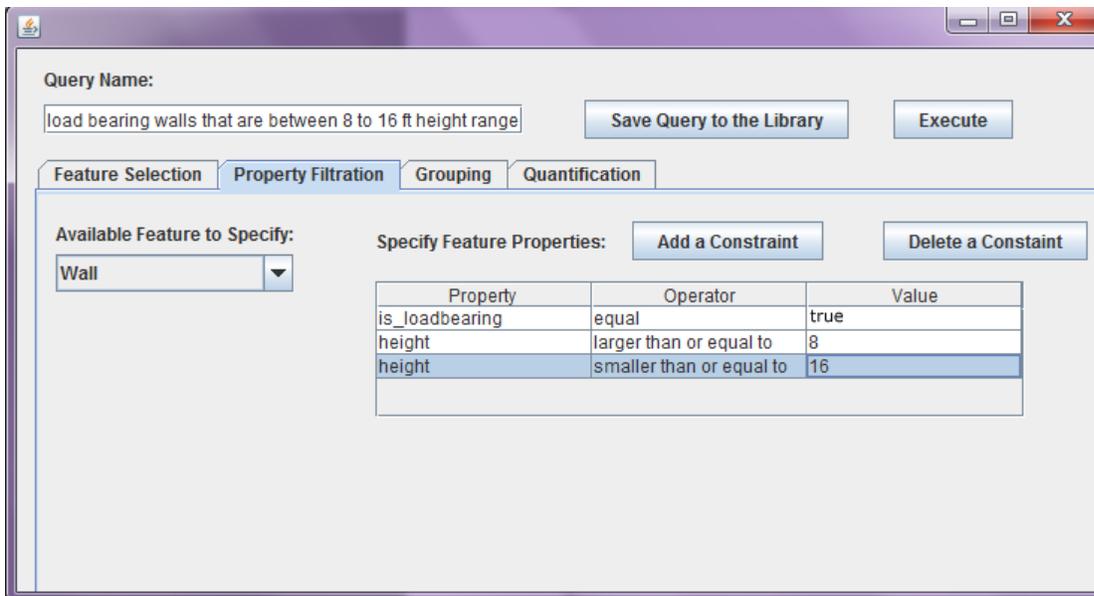
**Figure 2.9** Query interface for specifying a query

Figures 2.10 through 2.13 illustrate different query steps – *feature selection*, *property filtration*, *grouping*, and *quantification* – to specify basic queries. The actual query specification process starts first with the user selecting a feature, such as ‘wall’ to query for. As presented in Figure 2.10, the query interface shows the list of available features in the hierarchy, as defined in the feature ontology. In this step, the user also must select a floor(s) to run a query.



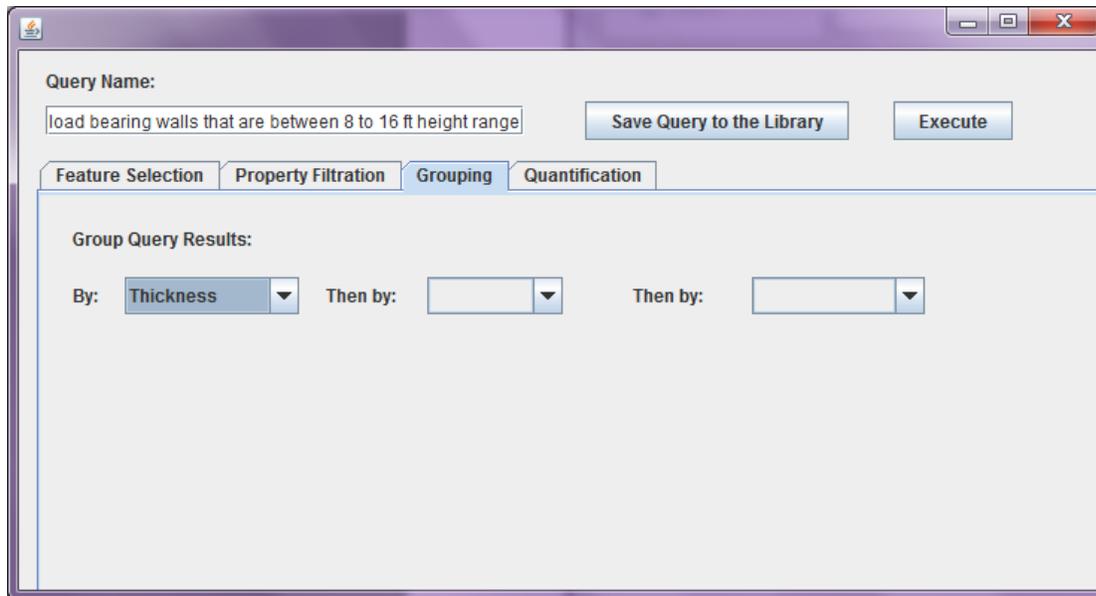
**Figure 2.10 Step 1 — Feature selection**

The next step in the query specification process is ‘Property Filtration’ which allows the user to define properties or constraints for the feature selected in Step 1. For the example above, the user selects the properties: “is load bearing” and “height” and selects (an) appropriate operator(s) and a corresponding value, as illustrated in Figure 2.11. The properties available to specify for a feature correspond to the feature attributes defined in the feature ontology.



**Figure 2.11 Step 2 — Property filtration**

After specifying property filtration, the user has the option to specify the grouping “Property” in order to group query results. This might involve grouping all wall instances that meet the constraints defined in the property filtration step above (Figure 2.12). Multiple grouping criteria can be defined as necessary.



**Figure 2.12 Step 3 — Grouping**

The final step in the query specification process is ‘Quantification.’ In quantification the user has the option to select a numeric parameter and an aggregate function (e.g., “length” and “sum” respectively) to calculate the total length of walls (Figure 2.13). The user can also select other aggregation methods, such as “count instances” and “percentage of total instances.” One final thing to note is that not all queries require all four steps to be specified in the query process.

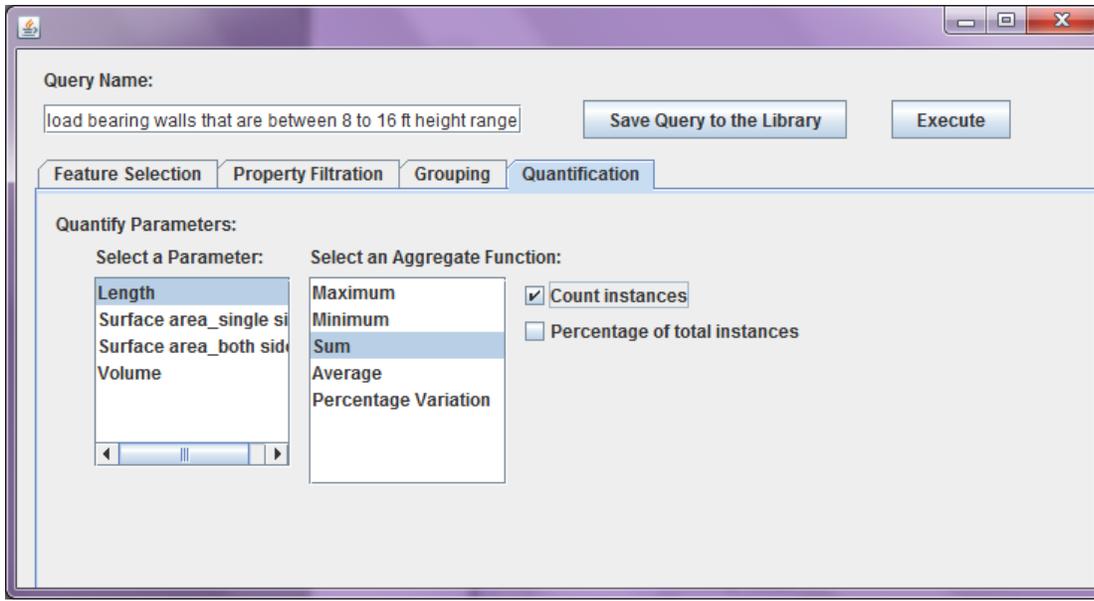


Figure 2.13 Step 4 — Quantification

Figure 2.14 shows the sample output after specifying the four query steps for the above-mentioned query example and executing the query for a simple BIM model. The query templates thus provide simple, user-friendly, expressive, and intuitive interfaces for seasoned construction practitioners to interactively formulate queries - spatial and non-spatial - and they allow practitioners to customize queries according to their individual needs or preferences. Practitioners do not need to understand the representation and data structure of objects in the underlying data models or *IFC*, nor require knowledge about the syntax or grammar of the query languages, such as XQuery or SQL, in order to formulate queries.

	B	C
Level: 1		
thickness	No. of instances	Sum of length
0.96875	2	60.140654259535
1.625	2	54.9999999999996

Continue with another query

**Figure 2.14** A sample query output

## 2.6 Evaluation Studies and Discussion

We performed a combination of tests to validate the content, representativeness, generality, soundness and intellectual merits of our approach. A more detailed description of these validation studies is provided in Chapter 5 of this thesis.

We conducted informal interviews with four construction experts, in relation to four building construction projects, to establish the content and face validity of our research. The construction experts interviewed were: (i) the project manager affiliated with one of the world's leading planning, engineering, and program and construction management organizations; (ii) the formwork manager working for the formwork contractor who specializes in the construction and erection of concrete formwork, serving the commercial, high-rise and high-end residential homes industries; (iii) the site superintendent for a medium-sized construction company, specializing in multi-residential, commercial, institutional and mixed-use projects; and (iv) the chief estimator employed by a general contractor who provides construction management services to clients. The interviews with these experts were supplemented with observations made on four different building projects, all with varying levels of complexity.

These interviews demonstrate that the concepts formalized in this research are relevant and representative of reality. The interview results also show the generality of the knowledge formalized in this research due to the fact that four different experts, representing different

construction domains or viewpoints, provided their (i) input on which design conditions are important and to what extent they are relevant (or matter) to construction; and (ii) rationale on why, and how (or under which conditions) they would matter in general, and specific to a referenced project in question.

We also evaluated the level of support provided by our approach, in relation to the state-of-the-art tools. We conducted retrospective analysis of several design conditions that construction practitioners are concerned with. Table 2.4 shows the summary of the relevant design conditions treated and level of support differentiated into three categories: “full,” “partial,” and “none.” Figure 2.15 shows the summarized results graphically by percentage of relevant design conditions supported. While the state-of-the-art tools provide comparable support to identify design conditions related to components in general, our approach provides increased support for identifying design conditions related to the features “wall,” “column,” “component intersection,” “opening,” and “penetration.” The results indicate that the state-of-the-art tools lack substantial support for identifying construction-relevant design conditions; the only features that they can identify more than 50% of are “openings.” In contrast, our approach finds roughly 80% of “opening” and 75% of “penetration” related design conditions. While we still fail to identify around 35% of construction-specific design conditions related to “wall,” “column,” and “component intersection” features, this is still considerably better than the current state-of-the-art tools.

**Table 2.4 Summary of the level of support results for identifying different design conditions**

Feature	Relevant Number of Design Conditions Treated	State-of-the-Art Tools						Our Approach					
		Full Support		Partial Support		No Support		Full Support		Partial Support		No Support	
		Count	Percent (%)	Count	Percent (%)	Count	Percent (%)	Count	Percent (%)	Count	Percent (%)	Count	Percent (%)
Components in general	22	4	18	6	27	12	55	4	18	8	36	10	45
Wall	29	8	28	4	14	17	59	15	52	4	14	10	34
Column	20	2	10	0	0	18	90	8	40	5	25	7	35
Component intersection	22	2	9	4	18	16	73	13	59	1	5	8	36
Opening	15	5	33	4	27	6	40	9	60	3	20	3	20
Penetration	12	2	17	0	0	10	83	8	67	1	8	3	25

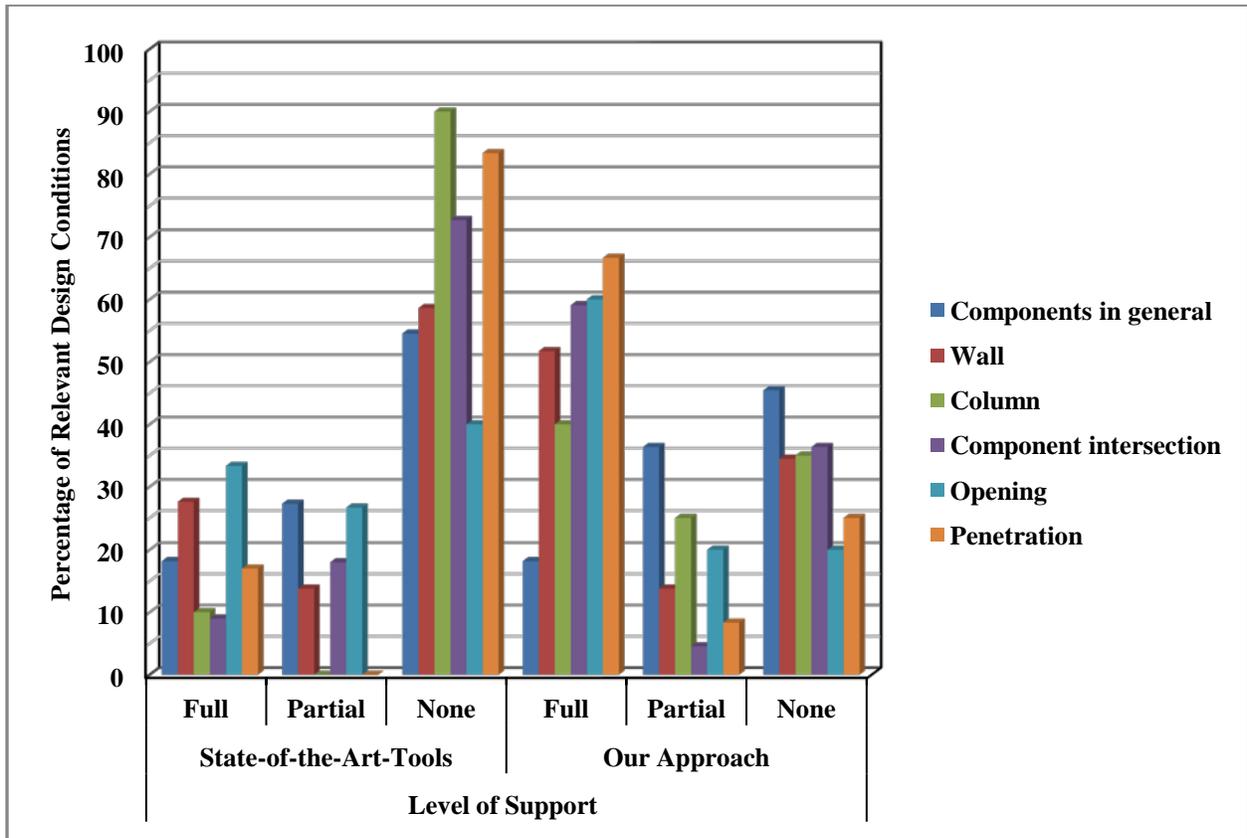


Figure 2.15 Summary of the results for the level (or degree) of support for different features

Our approach demonstrates multiple intellectual merits. It provides a richer and more flexible representation of building models that can be queried in order to retrieve insightful and useful component-specific construction information. State-of-the-art BIM tools provide limited support in this regard and do so in a less flexible way. Our approach allows construction practitioners to query for a specific type of component by specifying the relevant properties of a component. Construction practitioners, such as cost estimators, can filter from the range of pre-defined wall properties in order to define wall types. They can choose a property with which to group filtered components to organize them into useful categories, and specify aggregate functions to automatically derive the required aggregated information. Our approach can eliminate the usual manual tasks carried out by practitioners, such as cost estimators, who colour mark appropriate conditions on pdf drawings of building plans in tools, and such as on screen takeoff, to categorize components by material and other component properties. Practitioners using our system can easily specify component properties and other query attributes to

automatically apply the appropriate conditions. The process is efficient, intelligent, and low maintenance.

Our approach also provides richer support for identifying component intersections, which are not always explicit in BIM models. The ‘intersection query’ in our research not only identifies intersections between components, but also provides additional, more detailed and insightful information about the intersecting region: its location, dimensions (X, Y, Z dimensions), size, area and volume. Such detailed information is not explicitly defined in either *IFC* or *Revit* because they do not treat “component intersection” as distinct objects. While BIM analysis software, such as *SMC* and *NavisWorks* provides excellent support to check for interferences and clashes of components, the clash detection mechanism in these tools is meant to detect errors or interferences to help ensure the quality and integrity of the given model. They do not provide sufficient support to find intersections that are not conflicts between components, but are genuine intersections of interest to practitioners. The ‘intersection query,’ as formalized in our research, enables the user to find the specific type of intersection (e.g., intersections between concrete block wall and slab, intersections between dry wall and round columns), based on specific properties, or characteristics (such as material and shape properties), of intersecting components.

We also provide improved automated support to help practitioners easily identify component penetrations and openings, and other insightful information, such as their size, type, and location. Our approach allows the user the flexibility to define the host component where the penetrations (or openings) have to be found, to specify the type of penetrating elements (i.e., duct, pipe, conduit, etc.) and to specify relevant properties of interest, such as the dimension (size, area, perimeter, etc.) of penetrations or openings in the query formulation process. The practitioners whom we interviewed confirmed the critical importance of this type of information. Furthermore, the user can specify the properties of the host component, so as to locate the penetrations that exist on a specific type of host component. For instance, a drywall contractor can find penetrations on fire rated walls only and a roofing contractor can quickly identify all roof penetrations and their pertinent details. The clash detection mechanisms in *Navisworks* and *SMC* can be used to find penetrations on building components, but they do not provide support either to specify the host component, nor the properties of the host component. Unfortunately, they do not differentiate between a conflict, an intersection, or a penetration.

Our approach also provides the generic representation of domain concepts that are commonly needed to specify and query for other types of spatial concepts such as “location,” “spacing,” “alignment,” and “design uniformity” of features. It has the ability to identify the spacing between proximate features, such as columns, as well as other relevant information, such as maximum spacing, minimum spacing, average spacing, and percentage variation. Depending on the type of feature queried, the spacing query can be customized to query for spacing of a feature in a particular direction, be that horizontal (or along the specific coordinate axis, X- or Y-axis), or vertical. The key concepts formalized in this research are also relevant and sufficiently generic for querying the spacing of many other components or features, such as the spacing of openings (or penetrations), walls, beams, etc.

The ‘alignment query’ in our developed approach allows the user to query for unaligned features, and provides related information that is useful to practitioners. For example, not only does the ‘alignment query’ find horizontally unaligned columns in each floor of a building with respect to on- or off-grid criteria of columns location, but it also provides other useful information, such as the distance from the related grids. If a column is on-grid, then the corresponding the x- and y-gridlines are reported in the query results. If a column is not on-grid, it is described as off-grid, and the x- and y-gridlines that are closest are reported, as well as its distance to the grid line(s). The identification of unaligned columns location, in relation to the grid lines, can thus provide insightful information about the horizontal alignment of columns.

Our approach also provides more sophisticated support to assess “design uniformity” of features, in a floor or across floors. The intent is to identify the cluster of similar components or features, and recognize non-uniform features. Various techniques that have been used in our approach to support reasoning about design uniformity include (1) grouping of similar features, based on user defined criteria (i.e., feature properties); and (2) identifying features that have a change in location and other properties from floor to floor (e.g., change in columns location and size/shape from floor to floor). The first type of uniformity creates different categories of components which are grouped based on similar property (or properties) or property values, and calculates some measure of variation, such as count, percent count, etc. We provide additional support and mechanisms for evaluating component similarity by combining the nominal and quantitative attributes and by using a simple matching approach (Staub-French and Nepal 2007). The second category of uniformity assessment is more complex because it deals with the spatial

location of features and other attributes. It is a complex, time consuming, and error prone process used to manually identify the variation of different design features, for instance, to analyze the variation in column locations and size from floor to floor, particularly for complex multi-storey buildings, because the construction practitioner needs to overlay different floor plans and manually track the location, and other attributes. Our approach can reveal otherwise normally hidden, designer-focused and implicitly represented information in a BIM, to allow for explicit and construction-focused views.

Our descriptive and interpretative analysis demonstrates that our approach is capable of providing richer, insightful and actionable information, relevant for construction practitioners. The approach is flexible and expressive to answer a broad range of construction queries. State-of-the-art tools provide limited support to flexibly specify queries on components, component intersections, openings, penetrations, and reasoning about the location, spacing, alignment, and design uniformity of design features, in a way that is representative of the needs and requirements of construction practitioners. Our approach is also reasonably easy to use as it allows construction practitioners to formulate feature queries without any programming effort, know-how of underlying BIMs, standard schemas, or query language.

## **2.7 Conclusions**

This chapter describes our framework for automatically extracting and querying design conditions from a BIM. We motivated the functionalities and requirements of this automated system to be useful in terms of the practical needs of construction professionals. We provided analyses of the related literature in several areas. First, a significant body of knowledge was identified from the existing literature on the different design conditions that impact different aspects of construction or construction management functions. Second, we provided an overview of the related research on representational approaches, which has developed generic and specific schemas for representing building product information. Third, we described a number of research efforts and state-of-the-art computer applications which include formalized ways to reason about building product models.

The specific contribution of this chapter includes the development of an integrated approach to automatically extract features from a BIM and answer user queries from BIMs. The novelty of our approach is demonstrated in the use of model-based reasoning, feature modeling,

and a query-based approach, which provides more flexibility when extracting and querying construction features from a BIM. Our approach is generic in scope in order to support queries for different design conditions. It is flexible in that it allows BIM users to define queries that meet the varying needs, preferences and views of practitioners in terms of suitable design conditions, and when and how they are relevant.

The practical benefits ensuing from this research, when fully implemented, include considerable time and effort savings in understanding and extracting construction-relevant design information from a BIM, particularly for large projects. Moreover, it allows construction practitioners to quickly and readily access relevant information from BIMs, with no programming effort. This research provides useful information to aid in the decision making for different construction management functions in general, and in particular, to cost estimating, purchasing, constructability analysis, construction planning, and site execution. The various design conditions and queries formalized in this research provide a methodological basis for a logical extension towards automated extraction and querying of BIMs, for significantly increased productivity and efficiency in construction.

## Chapter 3: A Construction-Specific Ontology of Design Features and Feature-Based Model

### 3.1 Introduction

In recent years, several research and industry efforts have focused on developing building information models and leveraging those models to support various aspects of the architectural, engineering, construction and facility management (AEC/FM) industry. The emergence of building information models (BIMs) has created new challenges as well as new hope for construction practitioners. While the richness of design information offered by BIM is evident, there are still tremendous challenges in getting construction-specific information out of BIM, limiting the usability of these models for construction and other downstream processes. While BIM explicitly represents building components, component properties, and relationships between components, they do not explicitly represent many of the design conditions that are important for construction. As a result, construction professionals spend a significant amount of time and effort browsing, analyzing and interpreting design information to identify the relevant design conditions. This process is ad-hoc, inefficient, and inconsistent.

While there are many factors that affect construction (e.g., site, environmental, organizational, contractual, behavioural etc.), design-specific factors, or conditions, are particularly important because they have the greatest influence on construction costs (Paulson 1976) and design constructability (CII 1986). Many researchers have recognized the various design conditions that affect construction. These include, among others, the horizontal and vertical layout of elements, spacing between elements, dimensions, tolerances, alignment, modularity, repetition, similarity, uniformity, and standardization (Fischer and Tatum 1997; Hanna and Sanvido 1990; ASCE 1991; BCA 2001). Many design conditions occur frequently from project to project and are critical to different construction domains. For example, modularity, similarity, layout, and standard sizing are important design conditions in the construction of walls, ductwork, piping, and columns in building construction, and girders and trusses in bridge construction. Many of these design conditions are important for a variety of construction management functions and processes, including constructability assessment (Boeke 1990), productivity analysis (Thomas and Zavrski 1999), method selection (Fisher and Tatum

1997), cost estimating (Staub-French et al. 2003), and construction operations (Haymaker et al. 2004). Several design conditions also affect start-up, testing, operation, and maintenance of a facility (Tabesh and Staub-French 2006; Korman et al. 2003).

Current BIM tools provide some support for extracting construction-specific information, such as dimensional and component property information extracted from BIM-generating applications (e.g., *Revit*) and BIM-analysis applications (e.g., *Innovaya*). *Solibri Model Checker*© in particular provides sophisticated tools for extracting information present in an *IFC* model using design rules. However, these tools do not provide sufficient flexibility, ease of use and the necessary mechanisms to identify many of the design conditions that are important for construction practitioners. Moreover, they do not provide sufficient support to derive production relevant information at the level of detail necessary for practitioners (Haymaker et al. 2004; Katranuschkov et al. 2003; Borrmann et al. 2006).

The research challenge is that practitioners have differing preferences, viewpoints, and rationale for expressing the requirements for a particular design condition. Researchers have tried to encode design-relevant construction knowledge to provide more computer-based support for construction. Their approaches to date, however, have been limited: focusing on a narrow set of design conditions (Chen et al. 2005), requiring extensive user input (Nguyen and Oloufa 2002), and lacking user customizability (Haymaker et al. 2004).

This research seeks to address these challenges by formalizing an ontology of design features (i.e., a feature ontology), and its instantiation in the form of a feature-based model (FBM). The feature ontology formalizes the generic representation of construction-specific design conditions in a computer interpretable way. In essence, the feature ontology provides the blueprint for the additions and changes needed to transform the *IFC*-based BIM into a construction-specific FBM. The FBM enhances the BIM representation by explicitly representing a broader range of component properties (e.g., shape and curvature), and a richer representation of intersections (e.g., openings and penetrations). The FBM is an essential component in our overall approach to extract construction features from a given BIM and support the processing of user-driven queries on a BIM.

There are three desirable characteristics of a generic feature ontology:

- i. *Generality*: The ontology should be general enough to represent various design features relevant for different management functions. It should include the necessary attributes of features to sufficiently reason about them. In the context of our research, the generality characteristics also indicate that the ontology should not merely represent design conditions that are only applicable to one-of-a-kind or uncommon types of buildings, but rather be sufficiently representative to accommodate common project situations.
- ii. *Formal*: The ontology needs to have a structured, explicit, and computer-interpretable representation of features and feature attributes, in order to leverage the richness of digital information available in the BIMs and support the automatic instantiation of the ontology.
- iii. *Flexible*: The representation of concepts in the ontology should be flexible enough to accommodate the varied preferences or requirements of construction practitioners for defining features. The flexible representation of the ontology also ensures the wide applicability of potential applications that leverage the ontology, such as our interest in specifying user-driven and customizable queries. Flexibility of information models is one of the key desirable characteristics of any conceptual or information model (Haymaker et al. 2004; van Leeuwen and Wagter 1997; Katranuschkov et al. 2003) because it allows the extensibility and reusability of the models to accommodate changing user requirements, insights, and preferences.

In this chapter, we first provide a brief overview of a case study that motivates the need for a feature ontology and, that illustrates the subtle characteristics of design conditions that practitioners consider. Next, we provide background on the related literature. In Section 3.4, we provide a detailed description of our ontology and its representation, including the general design principles and knowledge acquisition process followed in its development. Section 3.5 explains the process of extracting features in order to create the feature-based model, and briefly discusses the FBM interface. Following that, we provide a brief overview of the tests performed to demonstrate the validity of the developed ontology and the resulting FBM.

### 3.2 Practical Motivation

In this section, we describe some scenarios from the recently constructed Chemical & Biological Engineering (Chem-Bio) building at the University of British Columbia (UBC). Specifically, we highlight design conditions that are explicit and implicit in the design of that building, and which impact different aspects of construction (e.g., cost estimating, productivity analysis, and method selection) for constructing walls.

A variety of design conditions impact wall construction. Figure 3.1 shows the configuration of walls on the fifth floor of the Chem-Bio building. Note the variety and spatial distribution of different wall types. Walls are characterized using various attributes, such as the *geometry* (e.g., curved wall, clipped wall), *function* (e.g., shear wall, partition wall), *material* (e.g., brick masonry wall), *spatial location* (e.g., office partition, atrium wall assembly), *exposure to the environment* (exterior wall, interior wall), and other *symbolic properties* (fire-rated wall, acoustic wall, etc.) to distinguish their types. Practitioners also distinguish walls based on their *dimensional* or *geometric parameters*, such as height (e.g., full height wall), the *specific value* of a parameter (e.g., wall height between 8 ft and 16 ft), and *constituent elements* or *parts* (e.g., stud walls) and their *properties* (e.g., metal stud wall, 15 MPa concrete wall). For example, wall types P5A and P5B are both interior hollow concrete masonry wall partitions, but the former is not fire-rated, while the latter has a 1-hour fire-rating. Construction practitioners derive the meaningful type and description of walls based on the definition and specification of walls defined by the architect or designer in a given design, such as the one shown in Figure 3.1.

The design conditions mentioned above impact wall construction in different ways. The spatial distribution of different wall types impacts productivity because crews have to continually switch materials as they progress linearly through the floor. For example, if a design has different wall heights and types, this can lead to decreased productivity due to the different construction methods required (e.g., different equipment is needed for full-height walls, for example, the use of scaffolding for wall heights between 9 and 13 feet) and activities required (e.g., cutting and framing are required for drywall panels). Different productivity rates are assigned to different wall thicknesses (e.g., 8, 12 and 15 inch walls), ranges of wall heights (e.g., wall heights between 8 and 16 feet), and types of wall shapes (e.g., clipped or curved walls). For example, consider the case of a brick veneer wall: RS Means Inc. (2004) states that if the wall is battered, 30% labour cost should be added to the cost; if it is a curved wall, an extra 30% is

added; and if the wall has corbels, an extra 75% labour cost is added. Understanding these design conditions, therefore, is essential when assessing productivity, method selection, constructability, and ultimately, the cost of the construction.

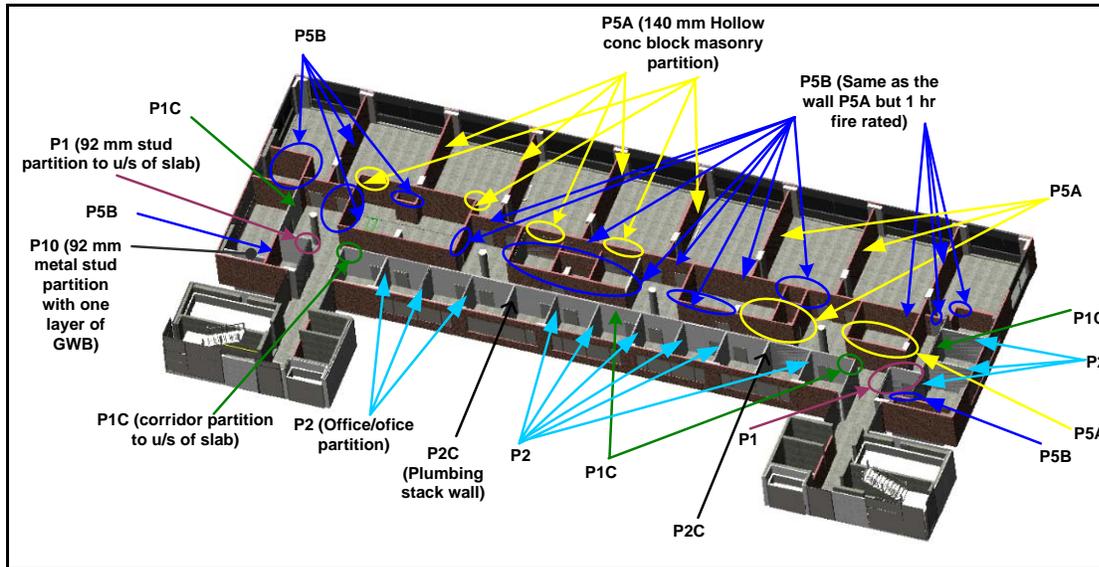


Figure 3.1 Wall configuration on the fifth floor of the Chem-Bio building, UBC

The interactions between walls (e.g., wall turns/corners, wall to wall intersections), and their spatial relationship with other components, such as the existence of wall openings and penetrations creates a number of construction considerations. For instance, pipe penetrations through fire-rated walls and slabs must be fire stopped; all service penetrations through P3 acoustic wall types and through block walls between labs must be packed and caulked. Using tunnel forms for concrete wall construction requires uniformity in the size and location of openings (Fischer and Tatum 1997). Different types of component intersections create different implications for construction. For instance, the intersection of two walls (i.e., wall turns) is important because this may require additional construction work for framing, layout, detailing, forming etc. Wall to column intersections are relevant because they may require additional set up, framing, and allocation for movement joints. These kinds of design conditions also impact the construction of other components, such as columns, slabs, beams, etc.

Construction practitioners look for the aforementioned design conditions (and others) in every building project when constructing walls and other building components because these are critical for assessing construction methods, productivity, constructability, costs, etc. They

analyze a design to derive a particular construction view relevant to them. For instance, they (a) identify walls that are between 8 and 16 feet high; (b) identify interior stud walls that have duct penetrations; (c) group walls with similar thicknesses or heights; (d) distinguish different types of walls, such as dry walls, concrete walls, masonry walls, fire-rated walls; and (e) calculate the total length and surface area for each type of wall.

Construction practitioners need a vocabulary to describe the design conditions that affect construction. A conceptual model is needed for representing these concepts generically in the computer. Formal mappings are required to map concepts in the conceptual model to the concepts in the standard schemas that describe a design in a BIM to enable the automatic extraction of knowledge required by practitioners. In addition, a viewer application is needed to facilitate the browsing and navigation of the extracted concepts.

This research develops an ontology of design conditions (or feature ontology) using the manufacturing concept of *features* (Cunningham and Dixon 1988; Shah 1991). We design and model the ontology using *Protégé Frame Editor* (Protégé 2008), an ontology development platform. We create mappings between the feature ontology and the *IFC*-based BIM and extract features in order to create a feature-based model (FBM). FBM enriches a given BIM by populating it with construction-specific design features and allowing the user to browse and navigate the extracted features present in the given design. The feature ontology and the resulting FBM provide a foundation to specify queries such as the ones highlighted in (a) to (e) above to help practitioners to easily and quickly identify the required information from a BIM. Chapter 4 describes query specifications and templates developed to specify such queries.

### **3.3 Literature Review**

To formalize an ontology of features, our research combines and extends the previous research in feature-based modeling, ontological modeling, product modeling and design relevant construction knowledge.

#### **3.3.1 Related Research on Features and Feature-Based Modeling**

The FBM developed in this research relies on the process of modeling product information using the manufacturing concept of features. Features in the mechanical or manufacturing domain describe characteristics, normally the geometry, of the part or assembly,

and carry a special significance or meaning in design, engineering, planning, and manufacturing of a product, such as manufacturability analysis and process planning (Cunningham and Dixon 1988; Shah 1991). They are associated with such attributes as intrinsic geometric parameters—length, width, and depth—as well as position, orientation, geometric tolerances, material properties, and references to other features (Mäntylä et al. 1996). They are relevant in terms of reasoning about the part or assembly, and in the process of manufacturing the part. While features in manufacturing can refer to geometric as well as non-geometric, or abstract, concepts, the most commonly known type of feature in the manufacturing process is *form feature* and is used for machining information of the mechanical parts.

Research in the architectural, engineering and construction (AEC) domain has shown the benefits of representing building design information using features (Zamanian 1991; van Leeuwen 1999). Specific to construction, researchers have used features to represent design and construction-specific information that is vital for the downstream applications executed during the construction, maintenance and operation of the facility. Some related studies describe features to represent lower level geometric aspects of a design, such as 3D surfaces, lines, and points (Boukamp 2006; Haymaker et al. 2004). Others use features to represent attributes of a design important to construction tasks, such as cost estimating (Staub-French et. al. 2003). Unlike manufacturing, features in the AEC domain can be present at any level of detail and design stage (van Leeuwen and Wagter 1997). As such, they can be used to assign semantics to design at any spatial element level, e.g., site, building, storey, and space as well as to a system, component, part, or connection. Features can also refer to geometry, topology or any other characteristics of building design.

There are essentially two approaches in feature-based modeling: design-with-features, and feature extraction or recognition. Design-with-features are specialized modeling entities which satisfy the specific needs of various design disciplines. The primary attribute of each design with feature, however, is still its three dimensional geometry (Zamanian et al. 1991). This approach uses primitives and user-defined features to generate higher level features to develop a design. It also captures the intent of a designer and is normally undertaken by defining constraints, which play a major role in feature modeling (Noort et al. 2000). The second approach in FBM recognizes or extracts features from the already designed artifact. It derives data from geometric models (normally the solid model) and builds up the feature model from an

analysis and interpretation of this data (van Leeuwen 1999). Feature extraction may be automated by means of computer algorithms, or it may be necessary for the user to perform the task (Shah and Mäntylä 1995). In this research, we define features, and automatically extract them from the already designed building product model, or BIM. The extracted information can later be manipulated to interactively query features of concern to construction practitioners.

### **3.3.2 Previous Research on Ontological Modeling**

Ontologies provide a means of representing knowledge about some domain of interest, and include a set of concepts (e.g., entities, attributes, and processes), their definitions, relationships and semantics (Genesereth and Nilsson 1987). They are explicit specifications of a conceptualization (Gruber 1995). Depending on the application domain and context, the purpose of an ontology can be one or all of the following: to reuse, share, exchange, and analyze content or domain knowledge, and to share a common understanding of the structure of information among users or computer systems.

There have been noteworthy industry-led efforts to standardize various aspects of building and construction information and to develop the vocabulary of building and construction terms. The development of classification systems, such as *Unifomat*<sup>TM</sup>, *Master Format*<sup>TM</sup>, and the *Overall Construction Classification System* (known as *OmniClass*<sup>TM</sup> or OCCS) comprise some of these major efforts. From a product modeling point of view, *ISO 12006-2:2001* (ISO 2001) provides a framework for the organization and classification of building construction information. The *Industry Foundation Classes (IFC)*, which builds on that framework, provides standardized concepts and terminologies and defines model schemas for representing information related to various aspects of the AEC/FM industry (IAI 2010). All of these efforts could be considered domain level ontologies, for the building and construction industry.

Many researchers have also focused on developing a taxonomy, or vocabulary, of building and construction information. *The LexiCon* (Woestenenk et al. 2000) consists of a structured set of terms and definitions, containing lexical, construction-related information about building products, materials and services for categorizing entities. El-Diraby et al. (2005) developed a domain taxonomy of construction concepts to help organizations in the area of knowledge management. Others have targeted more specific construction domains and

applications. For instance, Ugwu et al. (2005), provide an analysis and description of an ontology of constructability factors for steel frame structures. Staub-French et al. (2003) developed a vocabulary of the different types of features that affect construction costs and that represent the estimators' rationale for relating features of building product models with construction activities. These researchers, however, have not proposed mechanisms to extract and query construction-specific design information, relevant to various construction practitioners from a BIM. We build on the feature ontology proposed by Staub-French et al. (2003) and extend this formalism to support automatic feature extraction and querying. Unlike their focus on cost estimating, the ontology developed as part of this research, is generic and can aid in the decision making for a variety of construction management (CM) functions.

### **3.3.3 Previous Research on Building Product Modeling**

Building product modeling, or BIM, provides a semantically rich representation of components found in a building and describes the relationships between them. The information is stored as information objects, according to standard data structures. Some of the early works on building product models include, the *General AEC Reference Model* (Gielingh 1988), *RATAS* model (Bjork 1989), *COMBINE* (Augenbroe 1994), and *CIMSteel* (Watson 1995). These research efforts developed methods for product model-based information exchange for buildings. All of these efforts have led to the development of a product model- based information exchange for buildings within *STEP* (STandard Exchange for Product model data) and now the *IFC*, an undertaking by *IAI* (*International Alliance for Interoperability*), a global alliance of researchers, software developers, vendors, and other organizations.

The *IFC* is an *ISO*-endorsed, high level, object-oriented data model, based on class definitions for the AEC/FM industry (*IAI* 2010). It has developed model schema that is able to support a semantically rich representation of information pertaining to the life cycle of a building. *IFC* technology promises to offer seamless data exchange between different AEC/FM computer applications and to provide a platform for collaboration and the exchange of information through the definition and standardization of data models. *IFC* defines three fundamental entity types—objects, relationships, and properties, and these form the first level of specialization within the *IFC* class hierarchy (Liebich 2009).

Standard schemas, such as *IFC*, provide a standardized structure to construct and interpret a BIM. However, no single schema satisfies all construction practitioners' tasks, as these have different conceptualizations on when and how particular information about a building is relevant or impacts construction. Building product models, however, cannot foresee all possible usages of a model and requirements of downstream disciplines (*Graphisoft* 2004). Moreover, they do not provide sufficient reasoning mechanisms to guide construction practitioners to derive the required information from a design model.

This research relies on a BIM model available as a part of design process, and transforms it into a construction-specific FBM.

### **3.3.4 Previous Research on Design Relevant Construction Knowledge**

Many researchers have identified design conditions that are important or that impact different aspects of construction. They have investigated what and how design conditions influence the applicability and suitability of a particular construction method, especially in the selection of formwork systems and on the constructability of a design. Some related research in this area includes the work of Boeke (1990); Glavinich (1995); Burkhart et al. (1987); Thomas and Zavrski (1999); and O'Connor et al. (1987). Many researchers (e.g., Skibniewski et al. 1997; Ugwu et al. 2004) have also attempted to formalize construction knowledge in the form of specific design rules to assess the constructability of designs. Others (e.g., Hanna et al. 1992; Fischer and Tatum 1997) have built the knowledge base for the selection and application of certain formwork systems or methods for concrete structures. Previous researchers have recognized the impact of different design conditions on construction labour productivity (Smith and Hanna 1993; Thomas and Zavrski 1999). Many have identified the different design conditions that affect building construction costs (Hanna and Sanvido 1990; Thomas and Sackrakan 1994; Staub-French et al. 2003).

As discussed in Section 2.4.1 of Chapter 2, the nature and characteristics of identified design conditions in the previous research are varied and related to different levels of detail. Previous researchers, however, do not provide a formal way for representing design conditions. The identified concepts are generally unstructured and therefore fall short of supporting automatic extraction and querying of design-relevant construction-specific information from a BIM. We formally represent many of the design conditions identified by previous researchers to

systematically extract and query them from a BIM. While we focus on certain types of components, such as walls and columns, design conditions and their specifics, our approach can be generically applied to other components and design conditions, at varying levels of detail. The next section provides a detailed description of the ontology of design features we developed.

### **3.4 An Ontology of Design Features**

The ontology of design features developed in this research formalizes a common vocabulary, or language, to characterize design conditions relevant to construction practitioners, such as cost estimators, construction planners, and site coordinators. The ontology enables the systematization and explication for what is often implicit knowledge in a design, using a structured set of terms (concepts), that is general, computer interpretable, and easily understood by practitioners. This formal and explicit representation of construction-specific design conditions enables the extraction of relevant information from a BIM in a way that is consistent and unambiguous, and fulfills the varying needs and requirements of practitioners. The hierarchical structure of the ontology, its entities, their properties, and relationships are particularly useful in the generation of the FBM and the processing of user-driven queries.

The focus of the ontology is on the generic representation of knowledge to support the automatic extraction of construction useful information. We focus on the component-centric view and representation of a design in the development of the ontology. Consequently, the ontology considers a building in terms of major, basic building construction components, such as slabs, columns, beams, walls, etc., and their interactions and relationships with other design entities. As such, many detailed design features, as well as system level features, are excluded from the scope of the ontology. Although we have strived for comprehensiveness and generality in the ontology, we do not claim that it is sufficiently comprehensive and detailed enough to equally and completely support all construction management functions. The current set of terms may need to be extended to include additional detail or specific requirements for a particular CM function or application, and to integrate with existing applications, such as cost estimating.

### **3.4.1 Acquisition and Elicitation of Knowledge for Developing the Feature Ontology**

The elicitation of relevant concepts and acquisition of knowledge for the ontology is the collective process that comes from various knowledge sources, including the experience, intuition, and accumulated knowledge of the ontology developers. To develop the ontology, we gathered relevant terms and concepts of the feature ontology, through an extensive literature review on design constructability, value engineering, cost estimating, methods selection, construction planning and scheduling, etc. We also performed a detailed case study of the Chem-Bio building project at the University of British Columbia (UBC) and documented the existence of different design features, and their description and characterization by designers and construction practitioners. We also studied a number of other projects at UBC (The UBC Life-Sciences Building, the Centre for Interactive Research and Sustainability (*CIRS*) projects, and the Michael Smith Building) and analyzed a variety of design and construction documents (e.g., drawings, 3D models, construction specifications, cost estimates, and construction schedules) to study and understand the design features that impact construction.

We also extensively involved over a period of six months in the observation and documentation of weekly design and construction meetings on the *CIRS* project. Designers, suppliers, cost consultants, general contractors, and other specialty MEP trades, as well as owner representatives participated in these meetings. The meetings covered design parameters, building systems and subsystems, component typing and sizing, constructability analysis, value engineering, cost and schedule analysis, and design coordination issues. We also acquired access to various design and construction documents, such as meeting minutes, cost estimates, project schedules, design documents, consultants' reports, etc. This gave us a firsthand opportunity to gather, understand and familiarize ourselves as to what design conditions are most important to practitioners, and how they impact construction. We also interviewed construction practitioners, and had a series of interactions with them to better understand how they describe and characterize different design conditions that impact construction, and to substantiate concepts gathered from various knowledge sources.

### **3.4.2 Developing the Feature Ontology**

Ontology development is a difficult task due to the lack of a gold standard on what constitutes a common vocabulary or shared definition. It is important, in principle, that the

ontology provides a common and shared understanding of a domain, irrespective of the ultimate purpose of the developed ontology. This can be partly established through a general, shared, and consistent usage of terms. Also, specific to this research, flexible representation of the terms or concepts in the ontology is critically important for practitioners to enable the formulation of user-driven and customizable queries, on the premise that flexible representation of concepts would reasonably incorporate the varying perspectives and requirements of construction practitioners in the reasoning process.

The terms that describe the concepts in the feature ontology need to be chosen in order to match the natural usage of words by design and construction practitioners. This is often a difficult task. For a term to be used in an ontology, it should ideally have one precisely defined meaning. The inherent challenge in developing a common language to describe design conditions, is that researchers and practitioners use different terminologies, oftentimes in an *ad-hoc* and ambiguous manner. They use various terms to describe seemingly the same concept, or use similar terms (concepts) to refer to different things. Practitioners also have varied ways or preferences for describing when and how design conditions impact construction. In practice (mainly, face to face interactions), the particular meaning of potentially ambiguous words used in a particular context is correctly interpreted using common sense. This means that individuals deliberately use terms that have limited meaning or wide meaning for the given situation, but with the context and supplementary information, they come to the common agreement and meaning (Uschold et al. 1998).

Since there can be ambiguity in the usage of terms, a clear definition of terms is needed to clearly indicate the meaning, usage and scope of the term in any given context. The purpose of the definitions in an ontology, however, is very different from that of dictionary definitions, which describe meanings of words. Ontology definitions, have more of a normative role, and define how a limited set of terms are to be used in relation to each other. As such, each definition in an ontology requires careful understanding in relationship to the other definitions in the ontology (Uschold et al. 1998). The use of natural language and description is always critical in defining the terms as precisely and specifically as possible. Such precision is normally gained through the knowledge elicitation process, and by exercising careful judgment and consideration.

Once important domain concepts and terms are identified, the concepts are organized and represented into classes, and their relevant properties or relationships are defined. Classes that

describe more general concepts become the higher level classes, and more specific concepts become the subclass of the superclass (or superclasses). The arrangement of superclass-subclass hierarchy is often referred to as taxonomic hierarchy. For each class, the relevant properties are defined. Relationships are also types of properties, but they link different classes or class instances. The properties and relationships could be the remaining concepts or terms identified in the earlier stage. As acknowledged by Noy and McGuinness (2001), there is no one correct way or methodology for developing ontologies—there are always viable alternatives and the process is very iterative, and the best solution almost always depends on the potential use of the ontology proposed by the ontology developer.

We followed the above-discussed principles as much as possible in the development of the feature ontology. The following section describes the formalization of the feature ontology and the extensions made to the previous research.

### **3.4.3 Representation of Features in the Ontology**

We use the manufacturing concept of “features” to define different entities of the ontology. We combine and extend previous research on feature-based modeling from the mechanical domain to an AEC domain to explicitly represent design information that is important from the construction perspective. A feature-based representation is a representation of design information from a specific viewpoint. The set of features for a viewpoint is the set of semantics in that viewpoint (Rosen et al. 1991). In the context of our research, the set of features apply the set of semantics to the geometry, topology and other characteristics or attributes of the components of a building, and related entities or elements that exist on components. These features relate to the philosophy about the design by construction practitioners in different application domains.

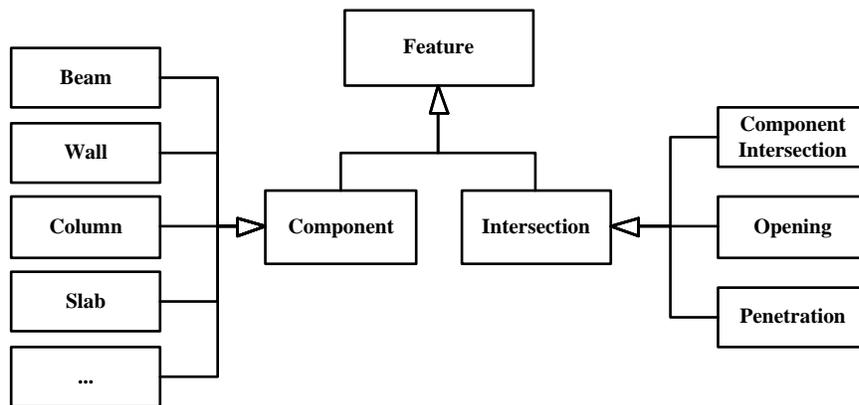
#### **3.4.3.1 Classifying Features**

Feature types or classes are the formal representation of the domain knowledge (van Leeuwen 1999). Although there could be many ways to classify the number of possible features and feature types (or classes), the classification of features into categories is useful for a number of reasons. First, it could lead to the use of common terminologies; facilitate the development of a library of feature types and associated attributes; facilitate the handling of inherent

complexities; enable the abstraction of realities at various levels of abstraction and detail (Shah and Mäntylä 1995). Many researchers have proposed different ways of classifying features (Shah and Mäntylä 1995; Cunningham and Dixon 1988; van Leeuwen 1999; Staub-French et al. 2003). Our research builds on the “primitive” and “intersection” features proposed by Cunningham and Dixon (1988) but extends it to be useful in the context of building design and construction.

We classify features into two broad categories: *component* and *intersection* features. Our research shares the similar definition for component and intersection features as that of Staub-French et al. (2003). We, however, formalize openings and penetrations as distinct feature entities. The feature “*component*” refers to common building elements and is further categorized into feature subclasses representing more specific concepts, such as walls, columns, slabs, and beams, etc. The feature “*component*” has similar connotations to what *IFC* defines as *elements*, but it has more of a construction meaning or usage than in the design thinking of a building. It resembles what Lawson and Roberts (1991) see as the “component mode” of thinking of a design in their formalization of different organizations of knowledge, or modes of thinking, about a building design.

The feature type, *intersection*, is defined as the physical/geometric interaction between components that results in the formation of different types of intersections between components. We further classify intersection features into three types, as *component intersection*, *opening*, and *penetration*, features. These subtypes characterize the type and nature of components involved in intersection relationships. Figure 3.2 summarizes the classification of feature types formalized in this research.



**Figure 3.2** The feature hierarchy

We define the feature “*component intersection*” to mean the physical/geometric interaction between components that may involve a variety of conditions, such as a component attached to another component (attachment), a component connected to or connected from another component (connectivity), a component overlapping another component, a component meeting/touching another component, a component crossing another component, a component supporting or supported by another component, or component in relation to another component within a defined tolerance value. One component can intersect with any other components. Component intersections can occur between same type, such as intersections between walls (wall to wall intersection) or between different types (e.g., wall to column intersection) of components.

The feature “*opening*” refers to door openings, window openings, and other types of openings on instances of building components, such as walls, slabs, etc. Openings could be through or partial, void (or empty), or filled with elements (e.g., doors or windows). Figure 3.3 shows slab openings in the Chem-Bio building.

We use the feature “*penetration*” to describe design conditions that involve building service elements entering or passing through building components. Examples include duct, pipe or cable penetrating a wall or slab. Figure 3.4 shows different types of wall penetrations on a portion of a typical lab floor in the Chem-Bio building.

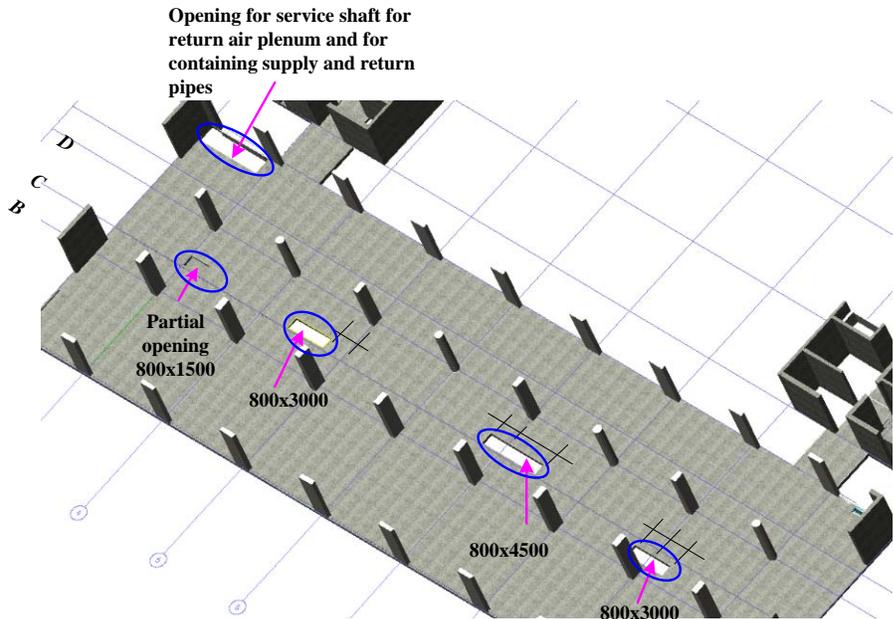


Figure 3.3 Slab openings in the fifth floor of the Chem-Bio Building

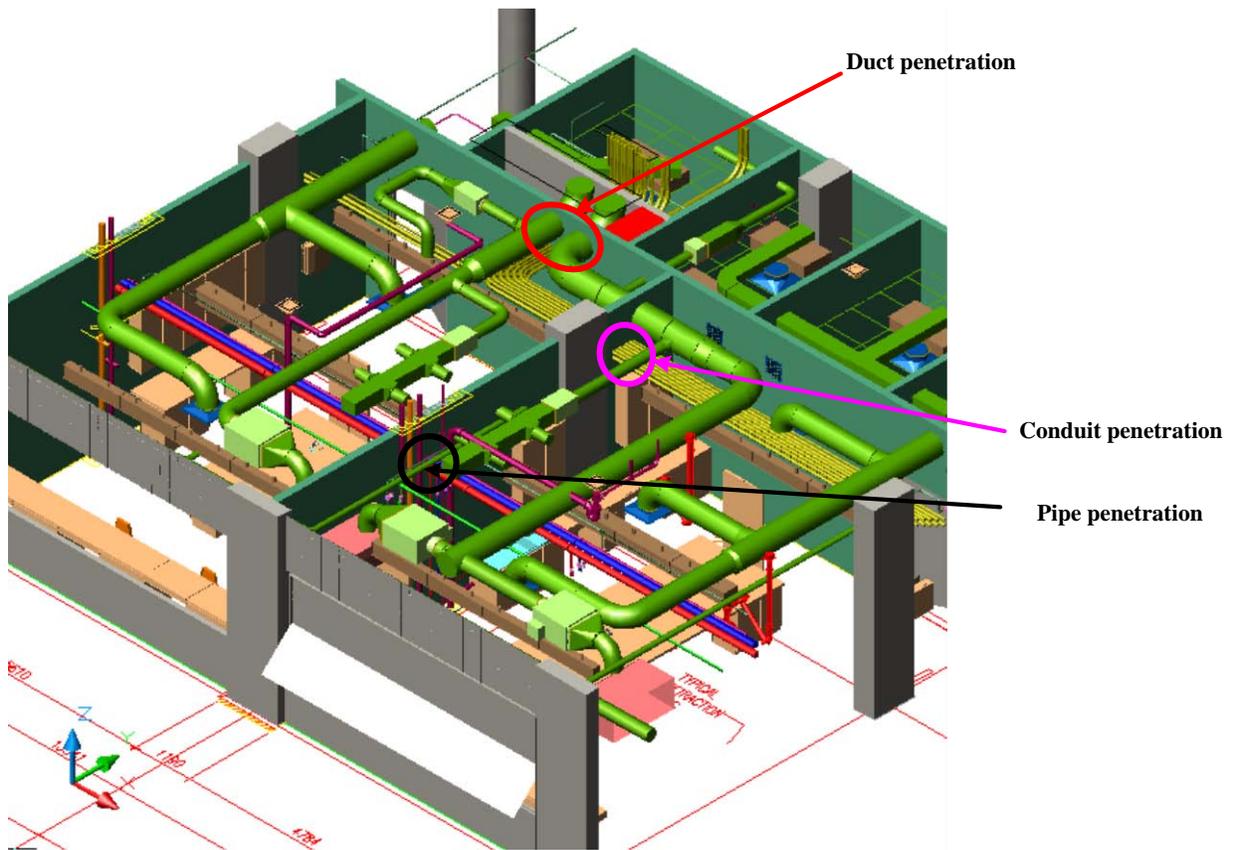


Figure 3.4 Examples of different types of penetrations on a typical lab floor of the Chem-Bio building

The division of intersection features into different types arises due to the semantics of the components involved in the intersection relationship. For instance, duct and pipe penetrations convey different contextual meaning to construction practitioners than do door and window openings. Also, the type and nature of the host component, on which openings or penetrations exist, can convey different point of view to different practitioners. For instance, a practitioner would treat window and door openings on walls differently than slab penetrations. Such meanings arise from the type and characteristics of objects involved in the relationships, the nature and the context of the relationships.

The representation of the ‘intersection feature’ in the current BIM applications also varies. For instance, door and window openings are generally explicitly defined in the model, but openings on slabs, although explicitly shown in the model, normally have implicit representation in current BIM authoring tools. The penetration of architectural or structural components by elements, such as duct or pipe, is not explicit in the BIM model because different professionals are involved in the modeling of these different building systems. *IFC* defines many spatial relationships (mainly topological) that may occur between objects or elements of a building and provides a foundation for topological reasoning about the design model. A connectivity relationship (*IfcRelConnects*) is a generalized objectified relationship that connects elements under various criteria. Various connectivity relationships, or subtypes of *IfcRelConnects*, are defined in the *IFC* model. For instance, the *IfcRelConnectsElements* is a 1 to 1 relationship that provides the generalization of the connectivity between elements. *IFC* defines attributes the *RelatingElement* and *RelatedElement* that provide reference to connecting elements. However, *IFC*-based BIM applications do not provide a mechanism to filter for specific types of intersections that are meaningful to construction practitioners. Moreover, some component connections are not explicitly defined in the *IFC*-based product models. We formally and explicitly represent intersection features and their subtypes (i.e., component intersection, opening and penetration) and attributes that are relevant to construction, and instantiate them for a particular building design.

### 3.4.3.2 Attributes of Feature Classes

Feature attributes characterize the different types of features. They consist of relational attributes and feature-specific properties. Relational attributes establish relationships between features. Feature-specific properties, on the other hand, are distinct attributes that are generally applied or assigned to a specific feature. They have little or no relationship to other features. Conceptually, various relationships between features can be generalized as being association and specialization relationships (see Figure 3.5). The specialization relationships represented as class sub-class relationships result in the taxonomy of features organized hierarchically and related by Is-a type relationships. Associative relationships are the most common types of relationships, linking one feature to other features. They also indicate, where applicable, the navigability and multiplicity of relationships. While the feature *component intersection* and its subtypes are distinct classes onto themselves, they can also be thought of as association relationships between components. For instance, *wall to column intersection* is a feature that represents relationships between instances of *wall* and *column* features; the feature *wall to wall intersection* is an association between wall objects or instances. In this research, we use a two-pronged approach: we represent component intersection features as a distinct feature type on its own, as the intersection feature has its own significance and physical existence. We also use intersection relationships to associatively link components that may be involved in such relationships, because it assists users to trace relationships between building components during the browsing and navigation of the feature-based model.

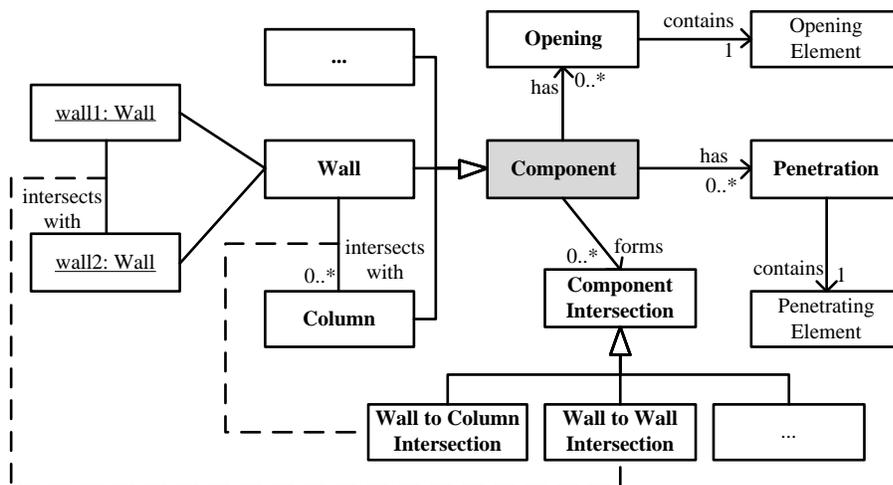


Figure 3.5 Specification diagram of features and their relationships using UML (Booch et al. 1999)

The conceptual perspective discussed thus far is extended to an implementation perspective in which we specifically consider the frame-based representation of features and feature attributes. Using the frame-based knowledge representation, we represent relational attributes of features by attaching them directly to a feature class, as if they are the intrinsic feature properties. Figure 3.6 shows the class diagram of an ontology with attributes relevant to each feature class. Similar to many object-oriented design methodologies, feature subclasses inherit the attributes defined in the superclass level. Moreover, new attributes can be defined or overridden at the subclass level. For instance, the feature *wall* inherits common attributes defined for the *component* feature, but a wall component feature includes many other attributes, not generally applicable to other components, such as columns. Each attribute has a unique name, data type, and if applicable, reference and default values. Tables 3.1 through 3.7 list different feature attributes, a brief explanation of each attribute, and the value type and cardinality of each attribute.

We recognize that *IFC* has defined some of the features and attributes in the ontology (designated with asterisks in Figure 3.6). *IFC* also indicates that exact definition and calculation rules for quantitative parameters may depend on the method of measurement or application used. We have tried to provide a more precise definition and calculation rules for quantitative parameters that may depend on the method of measurement or application used. We use the attributes as defined in Tables 3.1 through 3.7 to fulfill our requirements for the FBM and subsequently provide a support for the users to flexibly and easily specify queries that meet construction requirements.

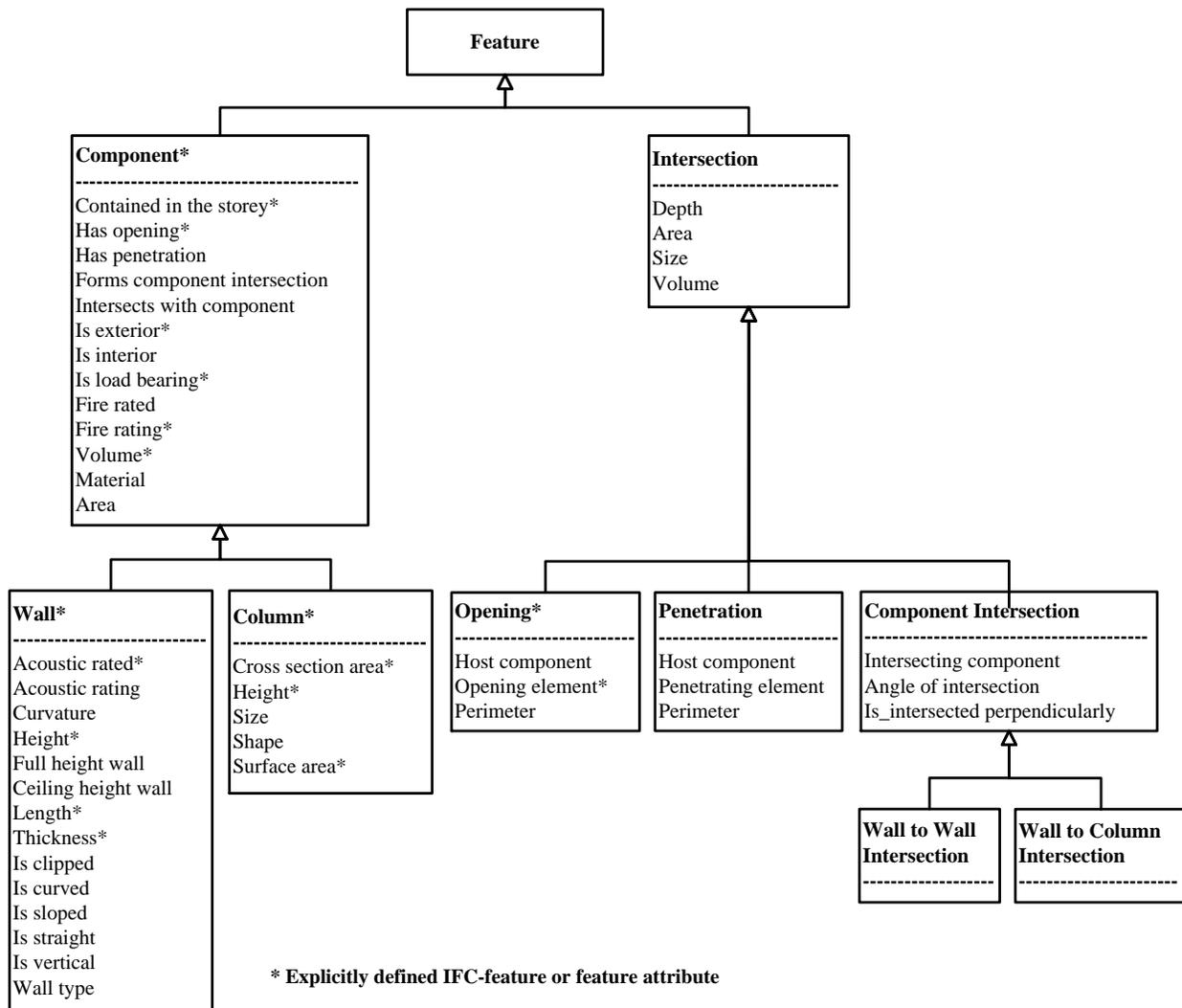


Figure 3.6 Class diagram with feature attributes

The **value type** of an attribute determines the kind of values that the attribute may hold. A **Boolean** value type can hold a logical Boolean value, a value that is either true or false. An attribute of value type **class**, takes one of the feature classes, or any of their subclasses. An attribute of type **float** has floating point values. An attribute of type **instance** has instances of the designated feature class as values. An attribute of type **string** has text strings as values. An attribute of type **symbol** allows the user to select from a list of strings. The **cardinality** of an attribute indicates the number of values it can have. An attribute with **single** cardinality, allows the attribute only one value, or no value, whereas **multiple** cardinality allows the attribute to have multiple values.

**Table 3.1 Generic attributes to the feature class “Component”**

<b>Attribute</b>	<b>Explanation</b>	<b>Value Type</b>	<b>Cardinality</b>
Contained in the storey	Denotes a floor or storey to which a component is contained or belongs to.	String	Single
Has opening	A relational property that points to an instance of opening existing on the given component.	Instance	Multiple
Has penetration	A relational property that points to an instance of penetration on the given component. For instance, a wall may have duct penetrations, pipe penetrations, conduit penetrations, etc.	Instance	Multiple
Forms intersection	A relational property referring to the instances of feature “component intersection.”	Instance	Multiple
Intersects with component	A relational property indicating to an instance of a component (similar or different types) with which the given component intersects.	Instance	Multiple
Is exterior	Indicates whether a component is an exterior element and faces the outside of the building.	Boolean	Single
Is interior	Indicates whether a component is an interior element and faces the inside of the building.	Boolean	Single
Is load bearing	Indicates whether a component is intended to carry loads.	Boolean	Single
Fire rated	Represents whether or not a component is fire rated.	Boolean	Single
Fire rating	Represents the fire resistance rating (FRR) of a component.	String	Single
Volume	Refers to the volumetric space that a three dimensional component occupies or contains.	Float	Single
Material	Represents major constituent material(s) or part(s) constituting the given component.	Symbol	Multiple
Area	Represents the area of a component as viewed by an elevation view (for wall, column) or as viewed by a plan view (for slab). Further specialization will depend on the type of component and area measures desired (gross, net, side/s, cross section).	Float	Single

**Table 3.2 Generic attributes of the feature class “Wall”**

<b>Attribute</b>	<b>Explanation</b>	<b>Value Type</b>	<b>Cardinality</b>
Acoustic rated	Represents whether or not a wall is acoustically rated.	Boolean	Single
Acoustic rating	Represents the Sound Transmission Class (STC) or acoustic rating of a wall.	String	Single
Curvature	Represents the degree of curvature, which is measured by the wall radius. The smaller the radius, the more curved the wall will be.	Float	Single
Height	Refers to the height of a wall measured in a vertical plane.	Float	Multiple
Full height wall	Indicates a type of wall expanding from the floor to the slab above.	Boolean	Single
Ceiling height wall	Indicates a type of wall expanding from the floor to the ceiling above.	Boolean	Single
Is clipped	Indicates a shape parameter of an instance of a wall with varying or different wall heights.	Boolean	Single
Is curved	Indicates a shape parameter of an instance of a wall with the wall axis or the base line as curved.	Boolean	Single
Is load bearing	Indicates whether a wall is intended to carry loads.	Boolean	Single
Is sloped	Indicates a shape parameter of an instance of a wall with inclined surface in a vertical plane. A sloped wall is non-vertical.	Boolean	Single
Is straight	Indicates a shape parameter of an instance of a wall with a straight axis or base line.	Boolean	Single
Is vertical	Indicates a shape parameter of an instance of a wall with all sections perpendicular to its base.	Boolean	Single
Length	Refers to the longitudinal dimension (or extrusion length) of a wall.	Float	Single
Thickness	Refers to the dimension along the direction right angle to the wall axis.	Float	Single
Wall type	Refers to a particular type of wall designation or typing used or specified in a BIM, normally by an architect, for design specification and/or communicating the design intent.	String	Single

**Table 3.3 Generic attributes for the feature class “Column”**

Attribute	Explanation	Value Type	Cardinality
Cross section area	The cross section area of a column in the plan view.	Float	Single
Height	The height of a column; <i>IFC</i> uses nominal length to denote the column height.	Float	Single
Size	Refers to the size of the column in the plan view. For rectangular columns, size is measured by X-Dimension and Y- (e.g., 400 x 600). Column sizes 400 x 600 and 600 x 400 are considered as same size. For round column, size is measured by its diameter (e.g., 600 dia).	Float	Double
Shape	Indicates the geometric shape of the column, such as rectangular, square, round, L-shaped etc.	String	Single
Surface area	The surface area of a column in its elevation views (excludes top and bottom area)	Float	Single

**Table 3.4 Generic attributes for the feature class “Intersection”**

Attribute	Explanation	Value Type	Cardinality
Depth	Refers to the intrusion depth along the direction perpendicular to the intersecting surface and indicates the linear amount by which one component is inside another component or vice versa at intersection. If a component touches another component, the depth of intersection is nil. For opening and penetration features, it indicates the depth of intrusion of an opening and penetration onto the host component, respectively.	Float	Single
Size	Refers to the size of the intersection measured as the combination of two linear dimensions on the surface of the intersection plane. The exact definition depends on the type of intersection.	Float	Double
Area	Represents the common area of intersection of intersecting objects by converting size measures into area measures.	Float	Single
Volume	Refers to the volumetric region formed by intersecting components at the intersection and is calculated as the product of the area and depth of the intersection.	Float	Single

**Table 3.5 Generic attributes for the feature class “Component Intersection”**

<b>Attribute</b>	<b>Explanation</b>	<b>Value Type</b>	<b>Cardinality</b>
Intersecting component	Refers to an instance of related/relating intersecting component forming a component intersection.	Instance	Multiple
Angle of intersection	Refers to an angle (degrees) formed by two intersecting components. The exact calculation may depend on the type of intersecting components.	Float	Single
Is intersected perpendicularly	Indication whether a component intersection is formed by components intersecting at 90 degrees.	Boolean	Single

**Table 3.6 Generic attributes for the feature class “Opening”**

<b>Attribute</b>	<b>Explanation</b>	<b>Value Type</b>	<b>Cardinality</b>
Host component	Indicates the component (e.g., wall, slab) where an opening exists.	Class	Single
Opening element	Indicates a physical component or element creating an opening or carrying a designated function. Different types of openings may result, depending on the type of opening element and host component, where an opening exists.	Symbol	Single
Perimeter	Represents the perimeter of an opening on the outside surface of the host component.	Float	Single

**Table 3.7 Generic attributes for the feature class “Penetration”**

<b>Attribute</b>	<b>Explanation</b>	<b>Value Type</b>	<b>Cardinality</b>
Host component	Indicates the component (e.g., wall, slab) where a penetration exists.	Class	Single
Penetrating element	Building services element(s) that forms a penetration on the host component. Different types of penetrations may result depending on the type of penetrating element and host component where a penetration exists.	Symbol	Single
Perimeter	The perimeter of the penetration on the plan or elevation view. Represents the perimeter of an opening on the outside surface of the host component.	Float	Single

*IFC* explicitly defines reusable *Property Sets* for different component types to represent the properties that are important to designers. The *Property Sets* consist of the predefined property definitions. The *IFC Property Sets* are to be seen as prototypes, but not as complete

property sets, and will need a more systematic account and specifications of attributes or properties to allow computer-based information management for building and civil engineering work (Ekholm 2002). We reuse the relevant predefined *IFC* properties for components as much as possible, and define new properties that are important from a construction perspective. Moreover, we provide a formal specification and representation of attributes of components, component intersections, opening and penetration features, in order to automatically extract them for a given BIM, and to enable query for the specific type or instance of these features according to the preferences of construction practitioners.

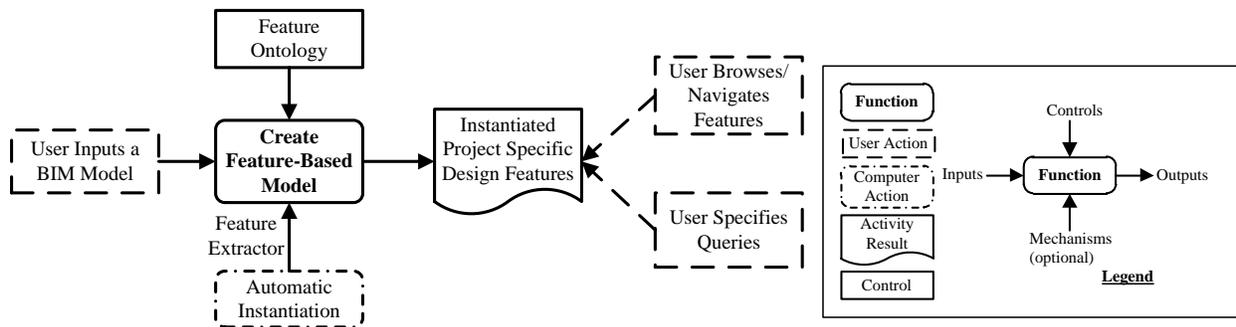
### **3.5 Feature Extraction and the Feature-Based Model**

In the previous section, we generically defined features that are important to support various CM functions. The feature-based model (FBM) instantiates these project independent features to a particular project, and allows construction practitioners to browse and navigate them. A number of applications currently provide a support for users to browse BIM models. For example, *IFC* browsers help users to browse *IFC* building elements and their attributes. *Solibri Model Checker* provides a richer view of design, by adding additional properties, such as location information, quantities, and some relations. This research builds on existing technologies to extend them to create project specific views of a BIM for features and their characteristics that are relevant to construction practitioners. We automatically extract the instances of component, component intersection, opening and penetration features, and the attributes of these features. This means that all predefined features are extracted or populated with project specific feature instances and their corresponding attribute values. The FBM can thus be considered as the secondary representation, or view, of a design in the eyes of the construction practitioner.

#### **3.5.1 Mapping the BIM Model to the Ontology: Automating Feature Extraction**

This section describes the feature extraction process and discusses some of the challenges involved when extracting features. We also detail some specific examples to illustrate the process of extracting features that encompasses a variety of the design conditions highlighted in the motivating cases. Figure 3.7 shows a process diagram in the form of an *IDEF0 model* (Colquhoun et al. 1993) for extracting features from an input BIM model. The identifiable

function in the model is to create the FBM based on the domain concepts formalized in the feature ontology. The model also symbolically presents the end users' actions or roles, as well as the computer action or mechanism, for executing the function. The ontology (explained in Section 3.4.3) specifies which features and attributes are of interest to the user.

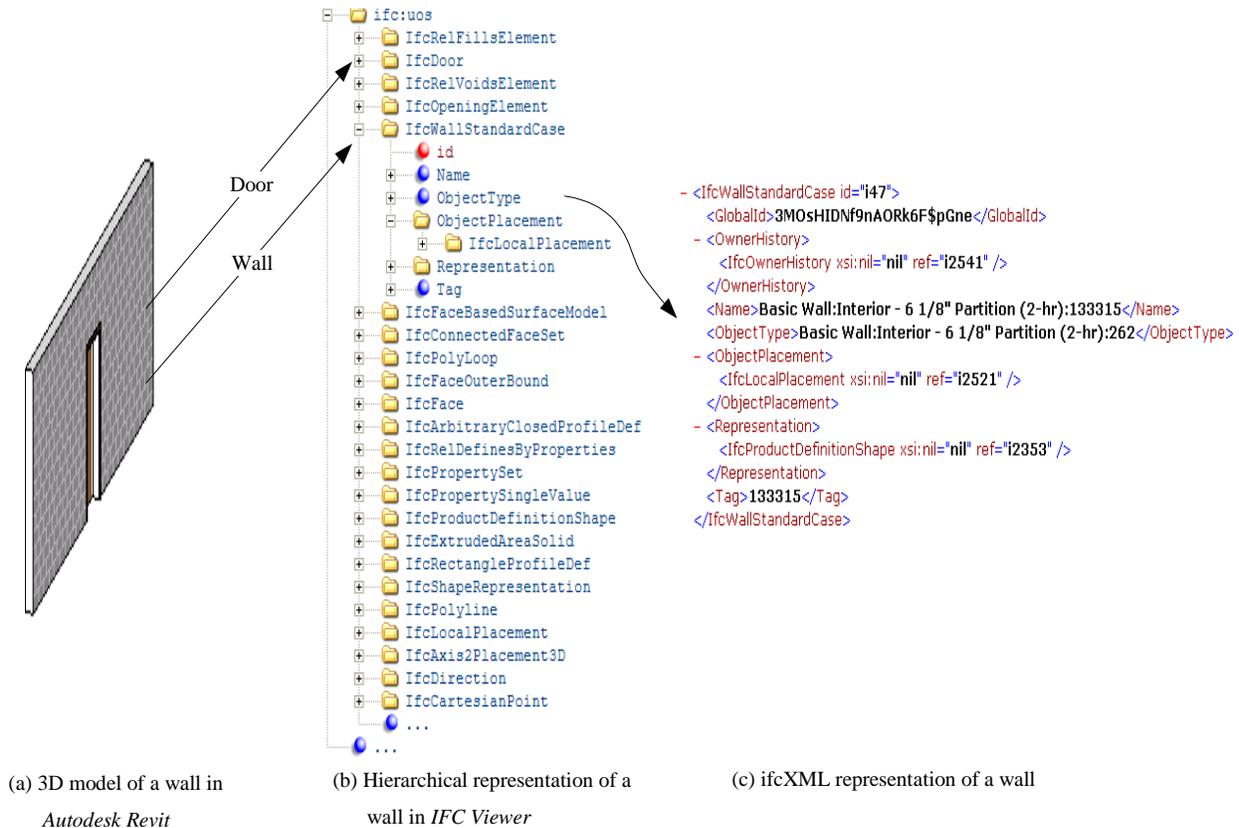


**Figure 3.7** An IDEF0 model for extracting features from a BIM, to create a feature-based model

We implemented this model in a prototype application, *Feature Extractor*, which abstracts and analyzes the relevant geometric, topological, and other attributes and characteristics of objects in the input *IFC*-based BIM to identify the feature instances and attributes defined in the feature ontology. This process transforms the designer-focused *IFC*-based model into a construction-focused, project-specific FBM. It is the domain-specific secondary representation of the design in terms of features that are relevant to the domain experts. Users can interactively browse, navigate and, subsequently, query the project-specific FBM. Note that the prototype application was created by Zhang (2008) and Webster (2010).

We use an XML representation of BIM data for feature extraction, as it is the most common data storage format. As such, building design data must be converted into an XML file. We use building design data stored in *Autodesk® Revit®* (referred to as *Revit*), a state-of-the-art BIM application for this purpose. BIM data from *Revit* can be converted to different files, including a handful of XML-based options: DWF-content XML, gbXML, and ifcXML. The required BIM data was extracted from *Revit* in two different ways. First, we made use of ifcXML data as much as possible. It offers the most comprehensive coverage of the relevant features represented in the feature ontology than other XML formats (Zhang et al. 2011). Additional supplemental data, not available in ifcXML, was extracted from the *Revit API*, as described in

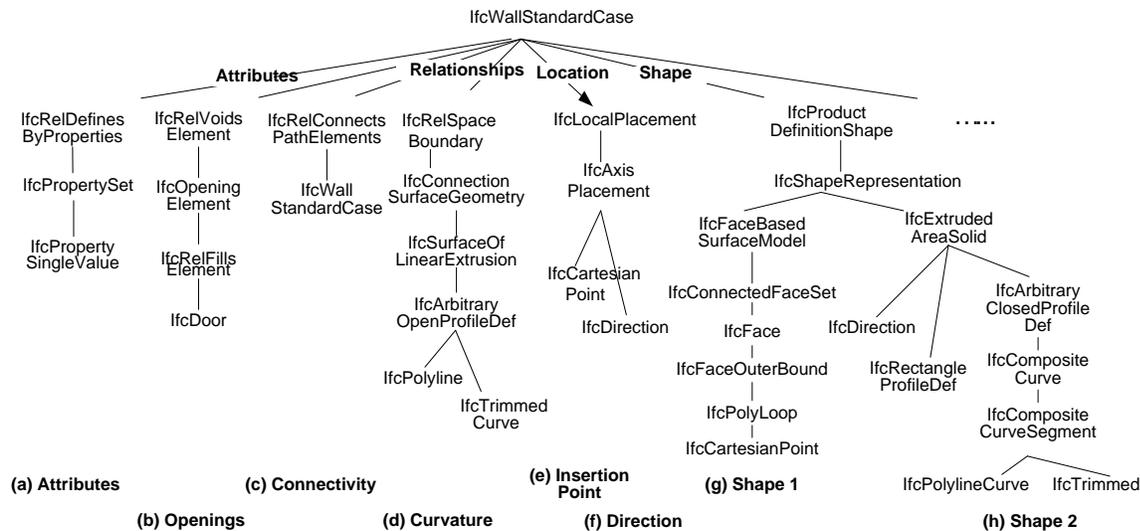
Webster (2010). In order to extract ifcXML data from *Revit*, we exported BIM data in native file format from *Revit* to an *IFC* file, and then the *IFC* file was converted to ifcXML file by using a convertor based on the *IFC2x2/ifcXML2x2* specifications. Figure 3.8 (a) to (c) shows the different representations of a 3D wall at each step in this process.



**Figure 3.8** A wall with corresponding ifc and ifcXML representations

An XML document contains a set of elements, where each element may have a sub-element (e.g., a wall element may have a sub-element describing its name) to describe relationships to other objects, or an attribute to describe a simple property (e.g., a wall has an ID). Figure 3.8(a) to (c) shows a 3D wall component, a hierarchical representation of some ifcXML elements in an XML viewer, and the actual ifcXML, respectively. While ifcXML is the best choice given the export mechanism considered in this study, due to its higher expressivity, it, however, has a much more complex schema and significantly larger file sizes (Zhang et al. 2011).

There are several challenges in working with ifcXML data. In particular, the properties of an element are often not directly attached to it, as sub-elements or attributes, (as is generally the case with highly-related XML data), but indirectly through ID references. This convoluted structure is demonstrated in Figure 3.8(c). The element “IfcWallStandardCase” has a limited amount of information explicitly attached to it: only the name and the history of the object are directly represented; all other related design conditions must be determined by navigating a complex sequence of references. Figure 3.9 shows a few concepts and the reference paths, displaying their linkage with a wall object. It becomes apparent that analyzing how objects are linked with different attributes and relationships is often the first, complicated, yet necessary step to extract features from ifcXML data. Moreover, the most challenging problem is such that much of the spatial information, such as feature locations and relationships (which are essentially derived from the spatial location of features) is not available in the exported ifcXML file. Information about MEP components, such as ducts, is incomplete. They are also unavailable in DWF or a relational database.



**Figure 3.9** Attributes and relationships with reference paths showing their linkage to a wall object in ifcXML

In order to address the lack of spatial data in ifcXML, the relevant data was extracted through the *Revit API* (Webster 2010). *The Revit API* is an application programming interface

that provides programmers direct access to the internal structure of the *Revit* model. However, it does require more time and effort to understand the *Revit API* and *Revit's* representation of spatial data. *Microsoft Visual Studio* using C#, a .NET-compliant language, a requirement for working with the *Revit API* (Autodesk 2009) was used to extract the required data. This information is output in an XML file (Webster 2010).

The process of extracting the information required by domain experts is a cumbersome task. It is necessary to formalize the mappings from concepts in a domain model (i.e., feature ontology) to the underlying schemas in the standard data model. In our research, mappings are created from the ifcXML schema to each of the concepts defined in the feature ontology, using XQuery, the standard query language for XML. Additionally implemented XQuery spatial query predicates extract features and attributes not adequately represented in ifcXML. These query predicates operate on data extracted from *Revit API* and are represented in a GML application schema (XML vocabulary), as described in Webster (2010). The following illustrates the process of automatically extracting some selected design features, defined in the feature ontology.

### 1: Identifying the Wall Type - Interior vs. Exterior

An ifcXML contains a Boolean property named “IsExternal.” It is set as “1” and “0” for external and interior walls, respectively. However, while it might be expected that this property would be explicitly attached to a wall (i.e., it would be its attribute or sub-element), determining the relationship is, in fact, much more complicated. In ifcXML, each object, property and property set is treated as an *IFC* element having an “ifcID.” Most properties, including “interior or exterior,” are linked to an object by attaching to it their ifcIDs, through the path shown in Figure 3.9(a). An XQuery expression is used to manage this process and find the appropriate property. Other properties that are explicitly defined in the *IFC* model can be similarly extracted.

### 2: Identifying the Wall Shape - Clipped Walls and Curved Walls

This example illustrates the identification of feature attributes that are important, but not defined explicitly in ifcXML. We defined ‘clipped’ and ‘curved’ as wall attributes. Deriving them requires an understanding of how walls are represented in the *IFC* model. There are two ways of representing walls in ifcXML: “IfcFaceBasedSurfaceModel” and “IfcExtrudedAreaSolid” are shown in Figure 3.9(g) and (h), respectively.

IfcFaceBasedSurfaceModels contain IfcConnectedFaceSets and, following several more referring steps, we eventually find the points that are used to define a wall. The faces, however, are not in consistent shape or order, which makes it too difficult to decipher if a wall is clipped or curved. The second representation, as shown in Figure 3.9(h), defines a shape by sweeping a bounded planar surface. The planar area, as well as the direction and the length of the extrusion, are given. The planar area can be a rectangle or it can be composed of lines or curves. Based on our observations, it appears that all non-clipped walls and clipped walls are represented using shape attributes IfcExtrudedAreaSolid and IfcFaceBasedSurfaceModel, respectively. The latter attribute is more complicated to analyze. We use the IFC attribute IfcExtrudedAreaSolid to determine the clipped or non-clipped wall. If a wall is represented by IfcExtrudedAreaSolid, then it is not clipped, otherwise it is clipped.

The attributes reference paths, shown in Figure 3.9, also provide a foundation to reason about curved walls, based on their representation. If the wall is represented using an IfcFaceBasedSurfaceModel, it requires complicated analysis of all wall faces. On the other hand, if the wall is represented using an IfcExtrudedAreaSolid, although the planar area can be composed of curves [the path on the right in Figure 3.9 (h)], the user would need to determine if those curves are the longer edges, since only walls whose longer edges are curved, are defined as curved walls. In either case, using the information listed in Figure 3.9 (g) or (h) turns out to be a complicated method to derive the property “curved.” The alternative is to determine the shape of the boundary surface of a wall. Figure 3.9(d) shows the ID referring path needed to determine the shape of the boundary surface from a wall. If the path ends with an “IfcPolyline,” then the wall is not curved; otherwise (i.e., if it ends in IfcTrimmedCurve), the wall is curved.

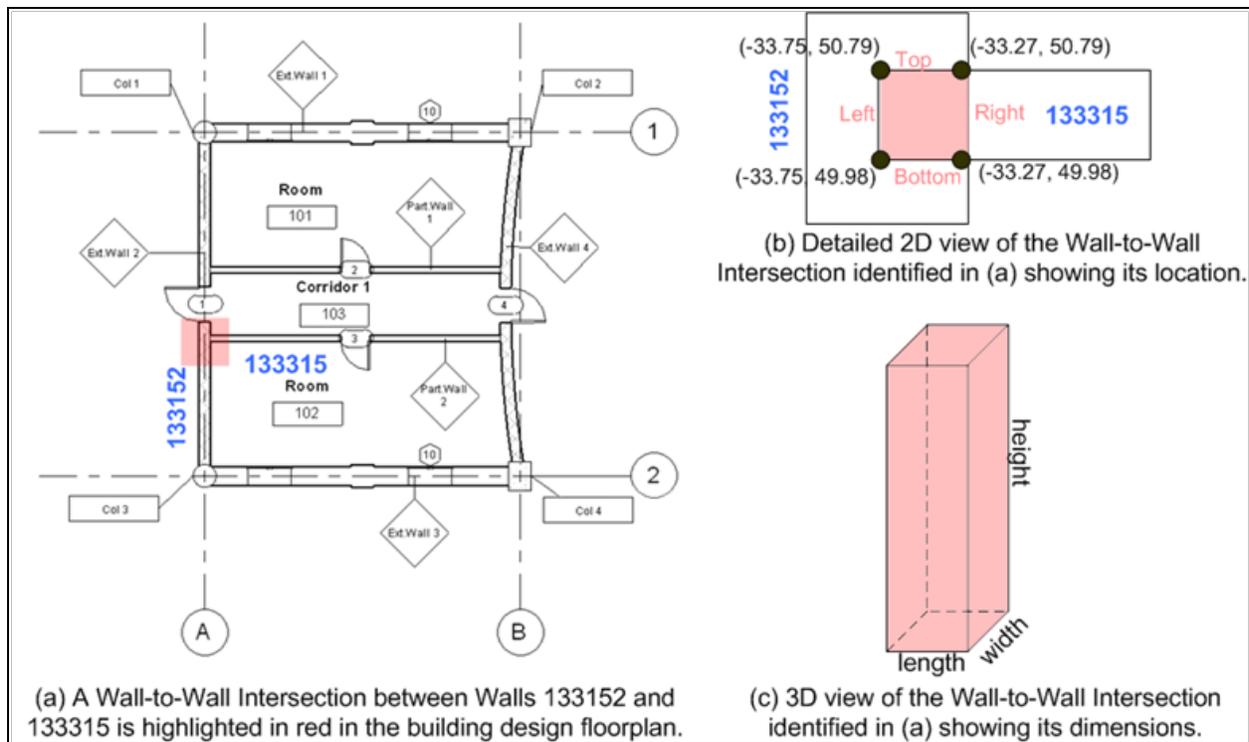
### 3: Identifying Intersecting Components – Explicit and Implicit

We consider two kinds of intersections: wall to wall intersections, and wall to column intersections. Wall to wall intersections are typically explicit in ifcXML because they are modeled explicitly. So we can query ifcXML, following the paths shown in Figure 3.9(c), to identify those component intersections. Determining which wall intersections are non-perpendicular, however, cannot be identified directly using ifcXML. They can be derived using the orientations of related walls, as shown in Figure 3.9(f). We first extract these two

orientations, which are represented by unit vectors, and then use a geometric formula to calculate the angle between them.

Wall to column intersections are often not explicit in ifcXML, because they are not typically modeled explicitly in 3D. Therefore, these types of intersections have to be derived by analyzing the location information of related objects. In the initial phase, we used an open source collision detection library called *RAPID* (RAPID 2008), which deduces the connectivity of two objects defined as triangular meshes.

As we needed more detailed information about component intersections (see the attributes of intersection features in Section 3.4.3.2), beyond whether building components simply intersect or not, the initial intersection prototype application was further extended. Where an intersection exists, the ‘intersection query’ predicate based on the 9-IM model (Egenhofer and Franzosa 1991) extracts more detailed information about the intersecting region: its location (i.e., the boundary points of the region), dimensions (X, Y, Z dimensions), size, area and volume. Such detailed information is not explicitly defined in both *IFC* and *Revit* because they do not treat “component intersection” as distinct objects. Figure 3.10 shows an instance of a wall to wall intersection with some of the corresponding details highlighted. Webster (2010) describes different query predicates including the ‘intersection query’ predicate and the processes followed to extract different spatial relationships between features formalized in this research.



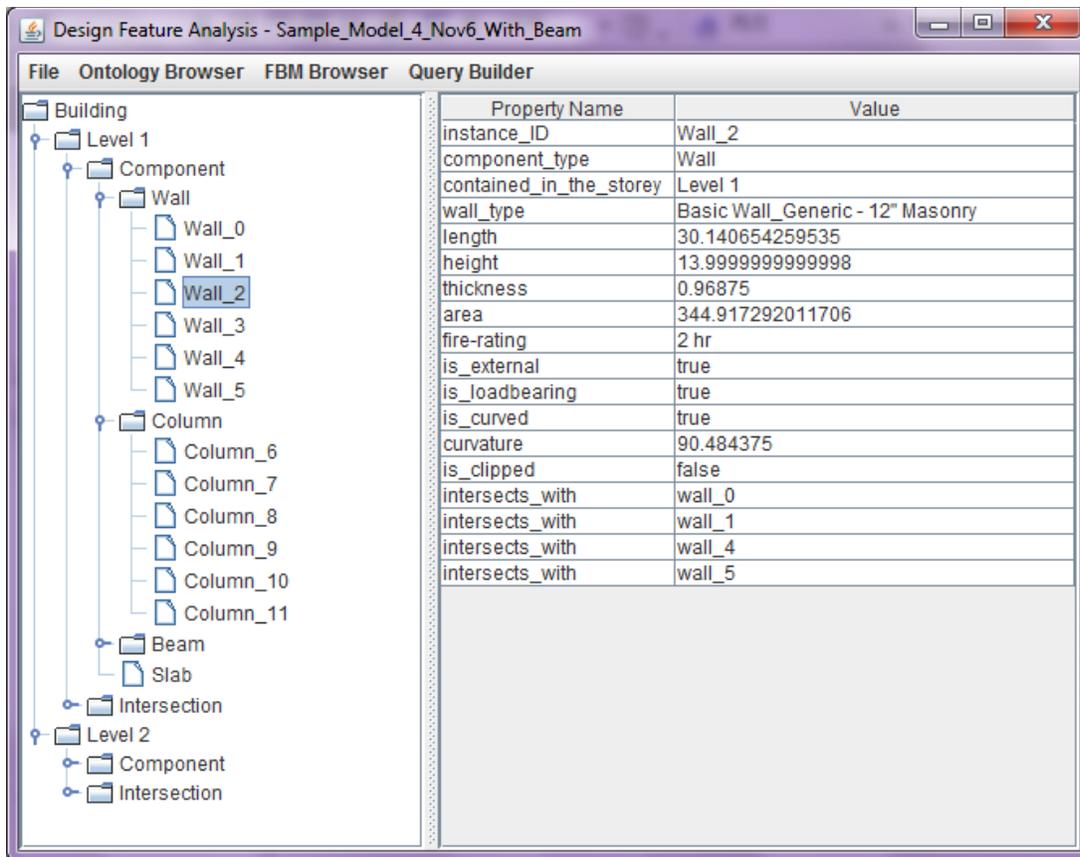
**Figure 3.10** Example of a wall-to-wall intersection and the details provided by the ‘intersection query’ predicate

#### 4: Identifying Penetrations - Duct Penetrations on Walls

As previously mentioned, duct information is limited in ifcXML. A few properties, such as type, insertion point, and shape can be found in ifcXML, but dimension and relationship data is not indicated, even though it was defined in the 3D model. DWF and relational exports also provide type and dimension parameters for ducts, but not location or relationship information. To address this, we extract the required data from the *Autodesk Revit API*. It should be noted that a duct penetration on a wall is essentially an intersection, where one of the intersecting components is a duct and the other is a wall. Moreover, additional information above and beyond what is reported for a standard intersection is needed, such as the location, area, and volume of penetration. To extract this information, we initially employ the “intersects” query predicate to determine if a penetration occurs. Where a penetration exists, the ‘penetration query predicate’ provides the location, area, and volume of penetration(s) (Webster 2010).

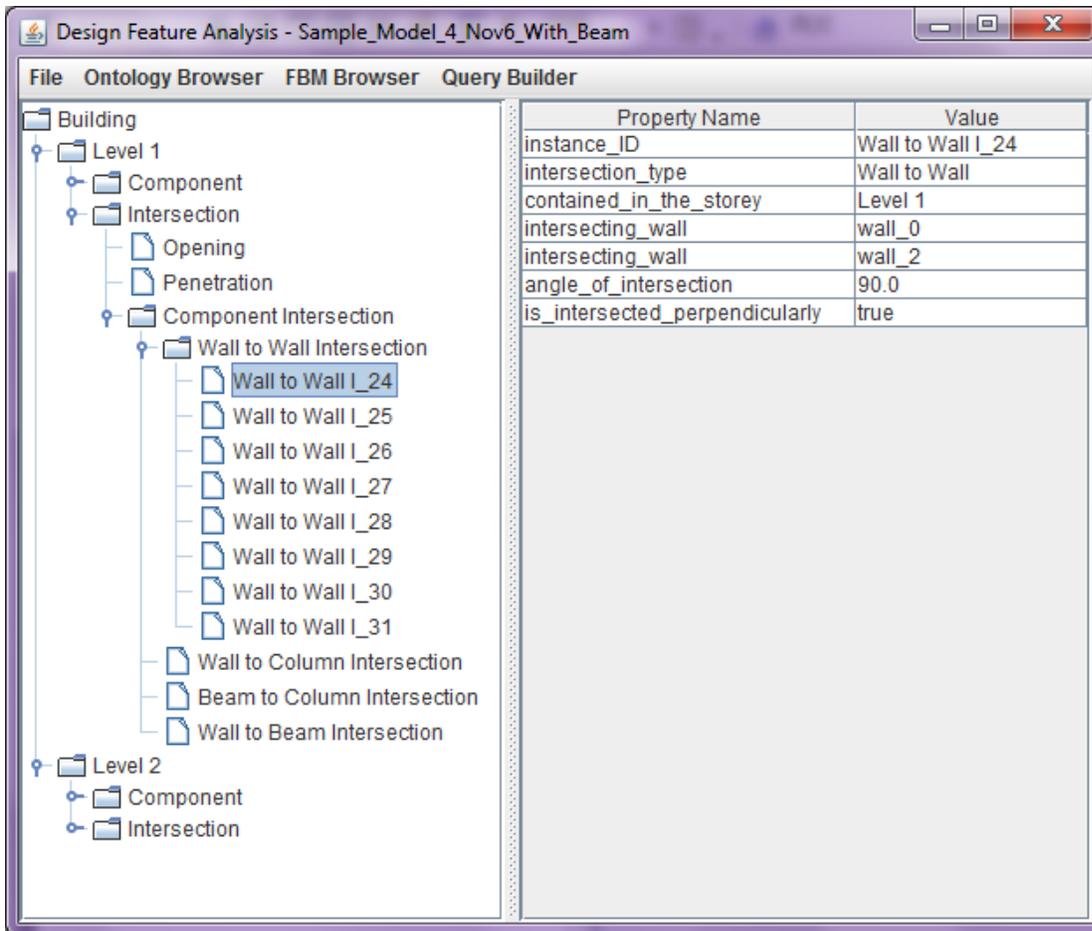
### **3.5.2 Feature-Based Model Interface and its Navigability**

In order to provide an easy readability and comprehensibility of the project-specific features we have developed the FBM interface. The interface includes basic browsing and navigation capabilities for features and feature attributes. It organizes feature classes/subclasses and their instances floor-wise. The prototype user interface for browsing the FBM is presented in Figure 3.11. The left side shows a hierarchical view of the instantiated features, which are organized by level (or floor), and then by the hierarchy specified in the feature ontology. The user can navigate feature instances in the left panel. These features include building components, as well as other features defined by the user, such as openings and penetrations. The associated attributes (properties/relationships) of a feature selected in the left panel are displayed in the right panel. They provide detailed information about that feature, which is derived directly from the characterization of that feature in the feature ontology, with corresponding values instantiated from a given BIM model.



**Figure 3.11 Browsing construction-specific features in the FBM**

The FBM explicitly shows attributes which are otherwise implicit in a BIM. Note the explicit representation of wall curvature ('is\_curved') and shape ('is\_clipped'), which extends the property information typically represented by the *IFC*. Also evident is the explicit representations of all component intersections (Figure 3.12) - these intersections are not always explicit in a given 3D model and corresponding *IFC* export. The instantiated relationships between features are dynamically linked which means that the user can navigate to the linked or referenced features. For example, the user in Figure 3.12 can click the "intersecting wall" property and its specific instance, such as "wall\_2," and see this feature's attributes displayed in the pop-up window. In this way, users can trace the relationships between features, such as related walls or ducts.



**Figure 3.12** Explicit representation and browsing of component intersections in the FBM

The flexibility and expressivity of the FBM browser is one of its key strengths. Not only does it promote a better understanding of the design features present in the given BIM model, but it also provides guidance to the user in the formulation of queries. The FBM browser enhances and complements existing technologies, such as *IFC* viewers, by providing additional support for navigating design features that are of importance to construction practitioners. Ideally, the FBM browser would be an information visualization tool that would allow the user to view the design features, both in data and graphical views.

### 3.6 Evaluating the Feature Ontology

Validating an ontology is an open problem and a difficult task. What constitutes a common vocabulary or shared definition for one individual may not be the same for another user. Various approaches to the evaluation of the ontologies have been considered in the literature, and

the choice depends on the types of ontologies and their specific purpose (Brank et al. 2005). In this research, we evaluate the ontology in terms of its vocabulary and applications, or tasks, for which the ontology is developed.

The validation process is described in detail in Chapter 5 of the thesis. Here we highlight some of the tests performed that are particularly relevant for evaluating the feature ontology and resulting FBM. Specifically, we provide evidence for: (1) the *content and representativeness* of the concepts formalized, which were validated through extensive interviews with construction practitioners, and (2) the *soundness* of our approach, which was validated by comparing the level of support provided by our approach with state-of-the-art tools in terms of representing or identifying information that is relevant to construction practitioners.

We interviewed four construction experts (a project manager, a site superintendent, a formwork contractor, and a chief estimator) and solicited their input about the degree of relevance (or importance) of the type of features and attributes formalized in the feature ontology. Table 3.8 shows experts' opinion on some of the design conditions related to the feature "wall". Results of these interviews indicate that the knowledge formalized in this research is sufficiently relevant and represents reality. The interview results also demonstrate some level of certainty about the credibility, assurance and generalizability of the knowledge formalized in this research, because four different experts representing different construction companies, domains and viewpoints and with reference to four different projects provided their useful input on what design conditions matter to them, the degree to which they matter, reasons for their importance, and how (or under what conditions) they would matter. Section 5.2.1 of Chapter 5 provides a detailed description, and full discussion of the results of the interviews.

**Table 3.8 Expert opinion on design conditions related to the feature “Wall”**

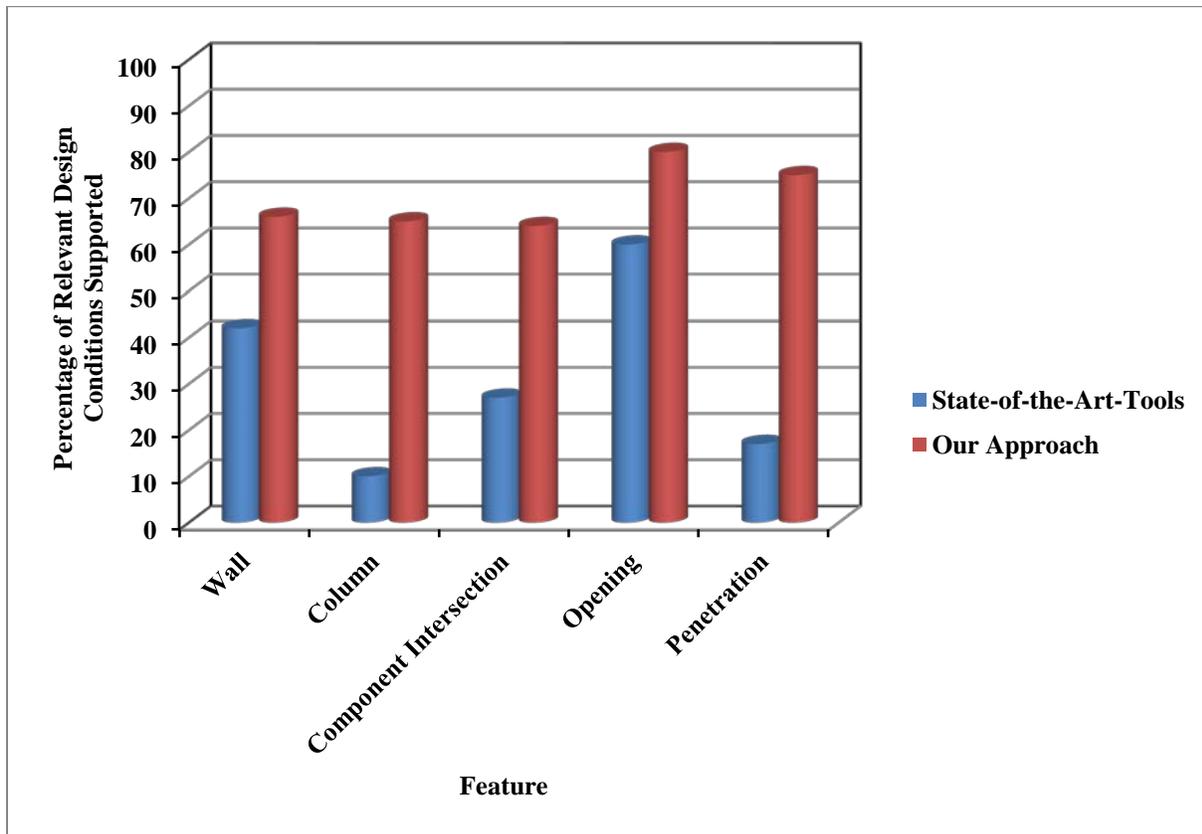
<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
Wall type	■ □ ●				No wall type is really generic; even two walls presumed to be of same type can actually have differing characteristics.
Wall shape:					
Straight vs. curved walls	■ □ ○	●			Curved walls are of concern to both estimator and tradesman, as they are significantly more labour-intensive than straight walls; Expensive formwork to build curved walls.
Vertical vs. sloped walls	□ ○	■ ●			Battered walls can have a slightly lower productivity rate when installing studs/ bridging/blocking, but not so much when installing sheathing/wallboard/insulation.
Clipped vs. non-clipped walls		■	■ ●		Not so relevant cost-wise, but can be significant layout concern to assure chipped portion(s) are correct.
Exterior vs. interior walls	■ □ ●	□	○		Speaks to same issues as does wall type.
Existence of blockouts, bulkheads, pilasters	■ □ ○	□	●		Added features/details aren’t nearly as costly as mistakes. Retrofitting/rework can be tedious if not done correctly, initially.
Constituent parts of a wall	■	■ ●			Level of concern, depends on uniqueness of certain parts constituting a particular wall.
Wall material	■ ●				The fewer number of types of materials, the better, as this involves fewer sub-trades.
Wall finish type	□ ○ ●	■			
Wall dimensions (height, length, thickness)	○ □ ●	■			
Gross surface area_single side	●		■		
Gross surface area_double side			■		
Net surface area_single side (i.e., without		■ ●			Useful to compare the ratio of net surface area to gross surface area;

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
openings and penetrations)					suggests special features, which can be concerning.
Net surface area_double side (i.e., net without openings and penetrations)		■			
Volume of a wall		●	■		More relevant to concrete walls.
Wall curvature	■ □				Tighter the radius, mistakes very costly if wall geometry isn't perfect to adequate – fixing mistakes requires do-overs. If you are building a radius wall you are not working quickly. What do you with radius cuts and plywood that someone took hours to build, but after one use will be unusable?
Fire-rated vs. non fire-rated walls			○ ●	□	
Fire-rating of a wall (e.g., a wall with 2-hr fire-rating)	■		○	□	Code requires fire-resistant walls to be generally compliant with published, prequalified U.L. tested assemblies.
Acoustic-rated vs. non acoustic-rated walls			●	○ □	
Acoustic-rating of a wall (e.g., a wall with STC value of 50)		■		○ □	
Load-bearing vs. non-load-bearing wall	●	○			Load bearing wall needs special steel gauge studs.
Full height vs. ceiling height wall	●	■ ○	□		“Full height” walls are a different type than ceiling height walls, which may be either braced or not above the ceiling. It is important to identify which walls adhere to the underside of the ceiling, as opposed to those that extend to the slab above. These two different “wall heights” incur different costs due to differences in productivity rate.
Wall turns/corners	■ □ ○	■ ●			“Wraps” if highly detailed can be quite concerning.
Non-perpendicular (or orientation of) wall	□ ■	■ ○ ●			

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
turns					
Spatial location of walls within a floor (e.g., corridor walls, classroom walls)		■ ●	■		
Spacing of walls			□ ■ ○	●	

■ Project Manager; □ Formwork Manager; ○ Site Superintendent; ● Chief Estimator

We also conducted retrospective analysis using the four construction projects studied to demonstrate the level of support that the FBM and subsequent query mechanisms are able to extract from a BIM model. We evaluated the level of support that our approach could support to extract design conditions that are relevant or important to practitioners, and compared them to the state-of-the-art tools. The results are presented graphically in Figure 3.13.



**Figure 3.13 Comparison of level of support results: our approach and state-of-the-art tools**

The results of our analysis suggest that our ontology-based approach provides significant support when compared to state-of-the-art tools. The state-of-the-art tools lack substantial support for identifying construction-relevant design conditions; the only features that they can identify more than 50% of are “openings.” In contrast, our approach finds roughly 80% of “opening” and 75% of “penetration” related design conditions. While we still fail to identify around 35% of construction-specific design conditions related to “wall,” “column,” and “component intersection” features, this is still considerably better than the current state-of-the-art tools.

### 3.7 Conclusions

Extracting the most relevant and useful information out of a BIM model is both challenging and time consuming. Previous research and existing tools provide limited support for automatically extracting construction-specific information from a given BIM. In this chapter, we described the development of: (a) the feature ontology, which generically formalizes design-related construction knowledge about building components; and (b) the project-specific feature-based model that explicitly represents features that are relevant for a given construction practitioner or domain and customized for a particular project.

A key consideration in developing the ontology is to provide a consistent, unambiguous, and computer-interpretable representation of features which are important from the construction perspective. The ontology formally represents the common and important design conditions, in terms of features and feature attributes (feature properties and relationships between features). We described using specific examples, the process of automatically instantiating concepts formalized in the feature ontology and creating the project-specific feature-based model, and discussed key features of the FBM interface. This research, in part, could help to identify useful and construction-relevant information for a variety of construction domains.

Additional work is required to explore other types of features that are relevant for construction, and for effective visualization and management of the extracted information in the FBM. In addition, the breadth and depth of concepts formalized in the ontology could be extended to accommodate other types of components, design conditions, and levels of abstraction. Our current work instantiates predefined features and feature attributes. Further study is also needed to investigate the potential for providing enough flexibility for users to compose new features or attributes on the fly. Finally, additional research is needed to integrate our approach in support of different construction management applications.

## Chapter 4: Query Specifications and Templates for Specifying Queries

### 4.1 Introduction

The continuing development of Building Information Models (BIMs) and BIM technology facilitated by *Industry foundation Classes (IFC)* has enabled the sharing, exchange and reuse of building information across multiple disciplines and software applications. BIMs contain a richness of intelligent information (geometric, topology and semantic details) related to the life cycle of a facility, and enable enhanced communication, coordination, analysis, and quality control (McGraw-Hill Construction 2008). This results in a faster and more cost-effective project delivery process, and higher quality buildings that perform at reduced costs (Hardin 2009; Eastman et al. 2008). BIM is gaining momentum in the architectural, engineering and construction (AEC) industry. Although much focus has been given to designer's use of BIM, contractors are also using BIM to support various construction management functions, including site and safety planning (Kristiina et al. 2009), scheduling (Russell et al. 2009), construction coordination (Khanzode et al. 2008), cost estimating (Staub-French et al. 2003), and constructability analysis (Wissam et al. 2009).

There remain, however, enormous challenges to fully leverage BIM for construction. The reality is that construction practitioners view a project differently from designers, and hence, require a different type of model; in effect, one specifically tailored to construction practitioners (McGraw-Hill Construction 2008). In particular, BIMs do not explicitly represent many design conditions that are relevant for construction (buildingSMART alliance & OGC Inc. 2008). For example, the location and dimensions of penetrations is important for concrete and drywall construction (Bisharat 2004; O'Connor et al. 1987), the spacing of columns is important for formwork selection (Fischer and Tatum 1997), and the variability of wall sizes impacts productivity (Thomas and Zavrski 1999). These kinds of design conditions result from complex spatial relationships between components, which are not explicitly represented in BIM today. Emerging BIM applications are addressing aspects of this problem. *Solibri Model Checker*® provides analysis tools to evaluate a given BIM's compliance with a set of design requirements. *Autodesk*® *Navisworks*® Manage software identifies interferences and component clashes between building systems in a given BIM or set of BIMs (Autodesk 2010). These tools provide

sophisticated mechanisms for navigating product data that is explicitly defined in the design model, but offer limited support for practitioners needing to identify construction-specific design conditions from a BIM.

There are many ways to address these challenges. Some suggest constructing separate BIM models with a construction focus (*Graphisoft* 2004). Others prefer populating the designer-specified BIM model with construction-relevant information to make it suitable for construction (Khemlani 2004). Many construction researchers have investigated mechanisms to transform or enrich design-focused product models to derive construction-useful information (Akbas 2004; Akinci et al. 2002; Navon et al. 2000, Haymaker et al. 2004). Others argue for providing query facilities (especially spatial queries) in order to rapidly derive the required information and views of a design that satisfy the needs of different BIM users (Wong and Sriram 1999; Borrmann and Rank 2009). The reality is such that current BIM-based tools (e.g., *Solibri Model Checker*) do not provide sufficient support to derive production relevant information at the level of detail necessary for practitioners (Haymaker et al. 2004; Katranuschkov et al. 2003; Borrmann and Rank 2009).

This research aims to address these limitations by formalizing query specifications that provide a formal yet flexible structure for practitioners to formulate queries on a given BIM. We use the manufacturing concept of ‘features’ (Cunningham and Dixon 1988; Shah 1991) to characterize design conditions relevant to construction practitioners. The queries operate on the features identified in the given BIM (explained in detail in Chapter 3). They relate to the manipulation of component, opening, penetration, and component intersection features and their attributes. We also provide query support for higher-level spatial concepts and phenomena, which currently includes *spacing*, *location*, *alignment*, and *design uniformity* of features.

The remainder of this chapter is structured as follows: next we describe several examples from projects we have studied that illustrate the different kinds of design conditions practitioners consider, then we describe the related research background, followed by sections on the formalization of the query specifications and the associated query templates developed, and finally the summary and conclusions.

## 4.2 Motivating Case Examples

In this section, we describe several case studies that illustrate the variety of design conditions that are important to practitioners when constructing concrete columns, walls, slabs and building service components. Construction practitioners look for these design conditions to accomplish a variety of construction management tasks, including cost estimating, method selection, scheduling, productivity analysis, subcontractor coordination and project management. Here we try to highlight those design conditions that are relevant for multiple domains, characterize them in a general way, and describe why they are important for specific construction applications.

*Openings* in building components (such as doors and windows, and even empty openings) and their properties (e.g., the location and size) impact construction productivity and methods of construction. For example, to make use of tunnel forms for concrete wall construction, there must be uniformity in the size and location of openings (Fischer and Tatum 1997). Similarly, *penetrations* of building components by building services are an important design condition occurring frequently in components such as walls and slabs. Wall penetrations, for example, often require special construction procedures, such as fire stopping, weather resistance, sound insulation, and the application of penetration seals. Failure to detect these design conditions can result in a considerable amount of rework as well as site coordination problems (Haymaker et al. 2004). Figure 4.1 shows one such condition that we observed on one of the construction projects we studied, the *Centre for Interactive Research and Sustainability (CIRS)* project at the University of British Columbia (UBC). On this project, a mechanical contractor was not able to route the principal air supply duct through the wall opening provided. As a result, the site superintendent had to order the concrete crews to dismantle the already-cast concrete portion. Construction practitioners often spend a significant amount of time analyzing and interpreting the different drawings to identify these kinds of design conditions. Figure 4.2 shows annotated drawings created by the site superintendent documenting the size and location of openings/penetrations on walls (left) and slabs (right).

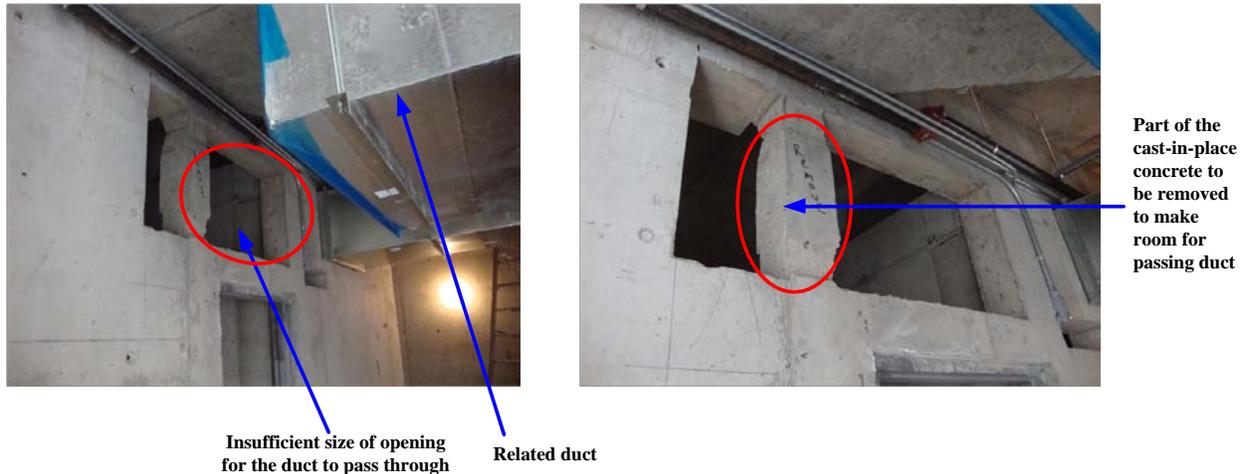


Figure 4.1 Example from the *CIRS* project showing a design conflict in which a duct cannot fit in the allocated wall opening

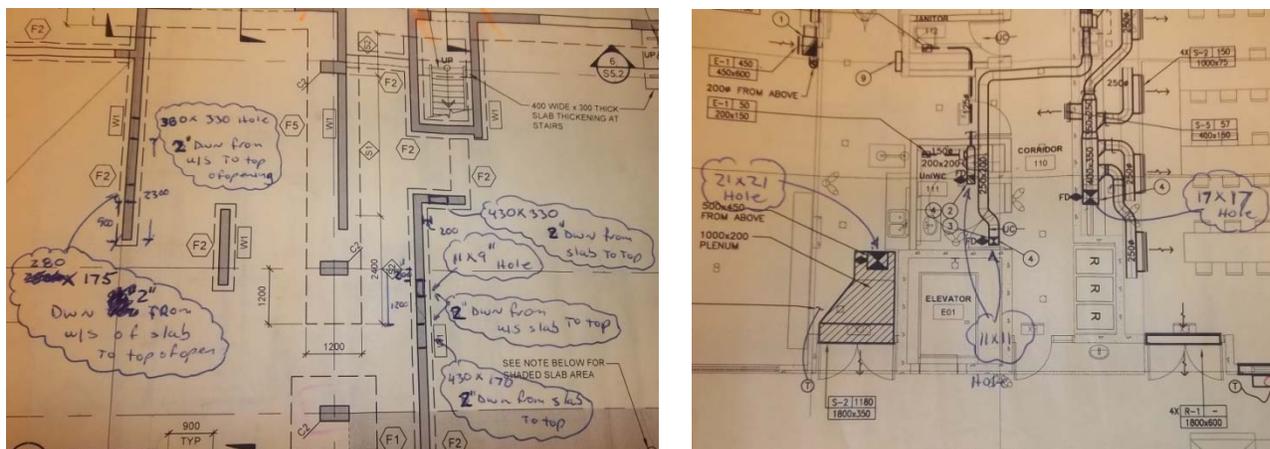


Figure 4.2 Annotated drawings created by the superintendent on the Engineering Design Centre (EDC) project identifying the size and location of openings on walls (left) and penetrations on a slab (right)

Current BIM analysis tools provide some support to check for openings and penetrations in a given BIM. For example, clash detection mechanisms in *Autodesk® Navisworks® Manage* (*Autodesk* 2010) can be used to find penetrations on building components. These programs, however, do not differentiate between a conflict, an intersection, or a penetration; they often identify false positives when performing clash detection; they cannot find specific types of intersections, such as intersections between drywall and round columns; and they are unable to

provide information about the location and uniformity of these design conditions, which are relevant from a construction and constructability perspective.

*Uniformity* in the design helps to standardize and select proper construction means and methods, and enables workers to learn the job fast faster, thereby increasing output and decreasing cost. For example, the uniformity of column spacing allows regular bay size and a regular grid of columns and frames that facilitates more efficient construction of other components, such as beams, slabs, walls, or cladding. Figure 4.3 shows the layout of columns in the first three floors of the Chemical and Biological Engineering (Chem-Bio) building project constructed at UBC, a project studied extensively for this research. Notice the consistency (or lack thereof) in size, shape, and location of columns within a single floor and across multiple floors. Practitioners today have a difficult time identifying this condition manually, typically analyzing variations in column schedules or using overlays of 2D drawings for the different floors to identify design variations. Current BIM analysis tools provide little support for practitioners to specify uniformity according to their preferences and identify the corresponding design condition from a given BIM.

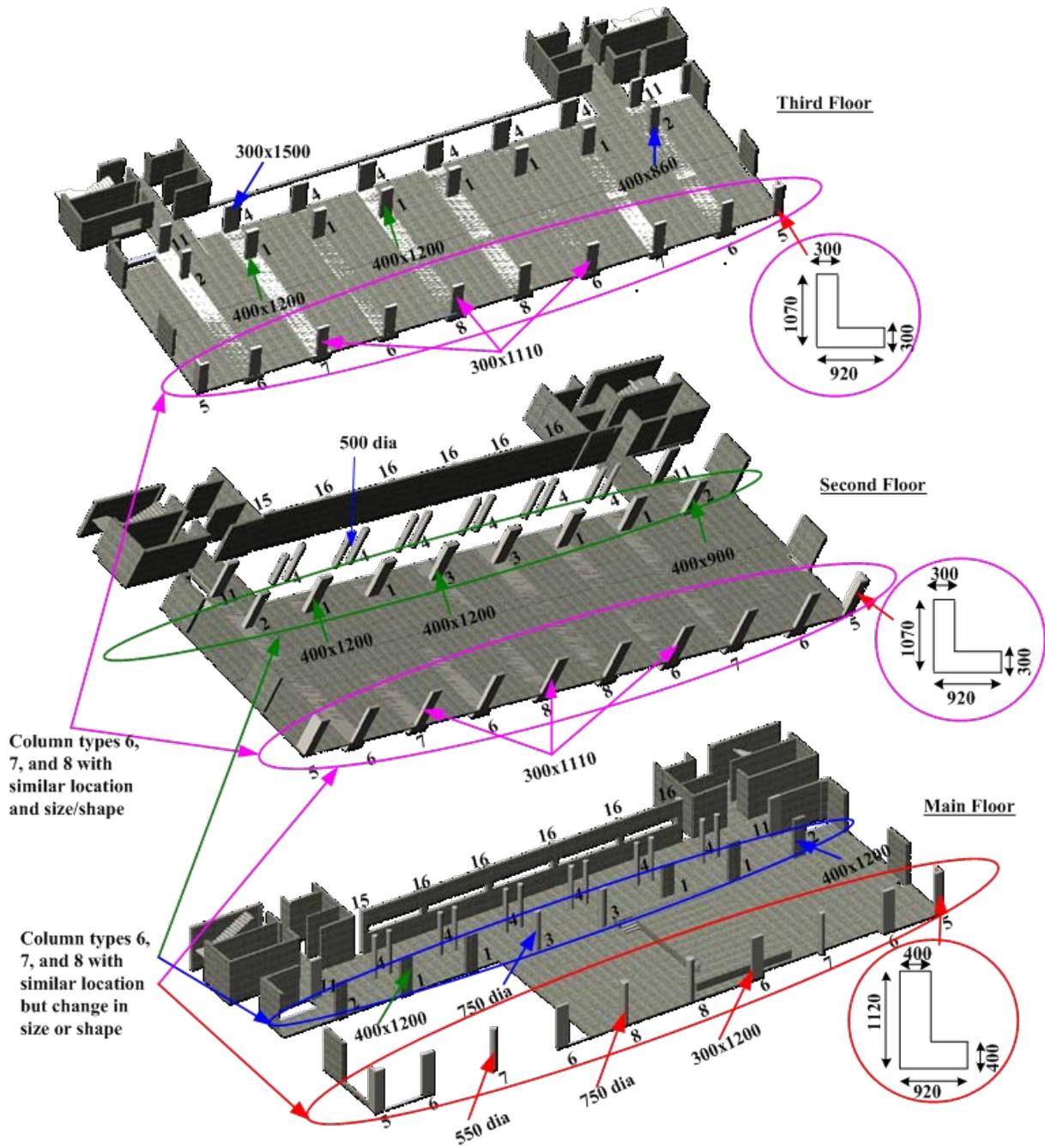
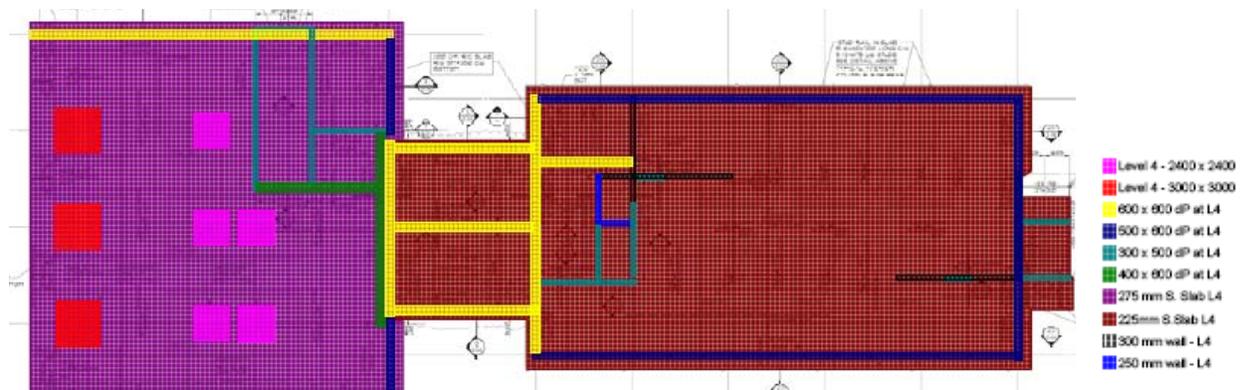


Figure 4.3 Variation in Column size, shape, and location in a floor, and from floor to floor, in the Chem-Bio building project, UBC

**Grouping components** is another type of important task required by construction practitioners, particularly cost estimators. For example, estimators working in a construction company in Vancouver, BC spend a significant amount of time and effort colour-marking

appropriate conditions on 2D drawings (in PDF format) of the building plans to group and categorize components by material and other component properties. Figure 4.4 shows an example from the new Arts and Science Building constructed at the UBC Okanagan campus. Note that the legend on the right indicates the estimator's rationale for defining appropriate groupings based on component-specific conditions (shown with different colors). This process typically involves identifying components by level (or floor), component type (wall, beam, slab, etc.), size, thickness, etc., and quantifying the results accordingly. This process is typically done manually using tools like *On Centre's On-Screen Takeoff*, which is inefficient and prone to error. Emerging BIM tools, such as *Solibri Model Checker*®, provide some support for grouping components but lack sufficient flexibility to meet the varied needs of practitioners, such as cost estimators.



**Figure 4.4** Colour marking of different design conditions by an estimator on a 2D drawing.

These case study examples illustrate the diversity of design conditions that are important to construction practitioners. Practitioners look for the aforementioned design conditions (and others) in every building project because they are critical for assessing construction methods, productivity, costs, etc. Emerging BIM tools provide some support for identifying these kinds of construction-specific design conditions (e.g., clash detection can be used to find penetrations), but lack the flexibility, comprehensiveness, and formal structure to support the requirements of construction practitioners.

Our goal is to formalize query specifications to support automated processing of user-driven and customizable queries related to these different design conditions on BIMs. Query

specifications provide a controlled and structured query vocabulary to specify different types of spatial and non-spatial queries, including the ones highlighted in these case study examples. Our approach is flexible as it provides a medium for practitioners, who typically have little expertise in BIM or computer programming, to formulate queries without requiring them to write query expression(s), programs, or possess prior knowledge of the underlying BIM database and query language. The next section describes the related research background.

### **4.3 Related Research Background**

We organize related research into three areas: (1) we provide an overview of the type and nature of building information and its representation; (2) we present a brief account of previous research in the context of reasoning about building product models; (3) we highlight some of the state-of-the-art BIM analysis tools and their support for identifying design conditions that are important for construction.

#### **4.3.1 Building Information and its Representation**

Building information consists of non-spatial and spatial information. Non-spatial information relates to the geometry, material and other characteristics of the components. Spatial information describes various spatial relationships between building components, or between components and spatial elements, i.e., site, building, storey, and space of a building. Much of the research on spatial/topological relationships relates to the concepts and technologies developed in the area of *Geographic Information Systems (GIS)*. According to Clementini and Di Felice (1996), spatial relationships fall into one of the following three basic categories:

- Topological relationships (e.g., adjacent, overlap): these describe whether or not two objects intersect, and, in the former case, how they intersect.
- Orientation relationships (e.g., north-of, south-of): these describe object location with respect to a reference.
- Distance relationships (e.g., very close, close): these describe the distance of an object with respect to a reference.

Topological relations are among the most extensively researched spatial relationships in the field of *GIS*. Egenhofer and Franzosa (1991) have formalized nine topological spatial relations, called the “9-intersection model” (9IM) that occur between polygonal areas in the

plane (spatial regions). They are: *disjoint*, *touch*, *equals*, *inside* or *contains*, *covers* or *is covered by*, *overlap with disjoint boundary*, and *overlap with intersecting boundary*. These relationships are defined in terms of the intersections of the boundaries and the interiors of two sets. The 9IM was later extended to the *Dimensionally Extended 9-Intersection Model* (DE+9IM) by Clementini and Felice (1995). The DE+9IM forms the basis for the formal definitions of topological relationships in the *Open GIS Consortium Standard* (OGC 2010). An important prerequisite for applying the 9IM or DE+9IM is, however, the formal specification of the interior/boundary/exterior of spatial objects (Borrmann and Rank 2009). There are also hindrances to fully use *GIS*-based tools and formalisms to spatial reasoning about a BIM, due to geometric and semantic differences which exist between BIM and *GIS* models (Isikdag and Zlatanova 2009).

Spatial relationships play a critical role in the AEC industry. Reasoning structures to infer spatial relationships of components in a building, however, are significantly limited. Researchers have recognized many important spatial relations between building components (Rush 1986; Nguyen and Oloufa 2002; Zamanian and Fenves 1994; Borrmann et al. 2006). Such relationships include topological relationships, such as, adjacency, intersection and containment relationships, orientation or directional relationships, and distance relationships between components. *IFC* defines multiple spatial relationships (mainly topological) that may occur between the objects or elements of a building. A connectivity relationship (*IfcRelConnects*) is a generalized objectified relationship that connects elements under some criteria. Various connectivity relationships or subtypes of *IfcRelConnects* are defined in the *IFC* model. The *IFC* product model, however, has largely ignored directional and distance relationships between objects (Borrmann et al. 2006).

A number of research efforts and tools have defined generic conceptual schemas and constructs to represent different types of spatial and non-spatial information about a building (Bjork 1989; Gielingh 1988; Augenbroe 1994; Watson 1995). For the past decade and a half, *IFC* has undertaken a global effort to develop a model schema that is able to support a semantically-rich representation of information pertaining to the life cycle of a building. *IFC* technology promises to offer seamless data exchange between different AEC/FM computer applications through the definition and standardization of data models. Standard schemas, such as *IFC*, provide a standardized structure to construct and interpret a BIM. However, they do not provide sufficient reasoning mechanisms to reason about the spatial and non-spatial information

of a building. Some researchers develop a structured set of terms and definitions, containing lexical, construction-related information about building products, materials and services to support eBusiness and eCommerce transactions (Woestenenk et al. 2000; Tolman et al. 2001). Others have developed domain taxonomy of construction concepts to assist organizations index and retrieve knowledge items (El-Diraby et al. 2005). Feature-based modeling can also be considered as a building product modeling approach, to represent information about a building, using the specialized modeling entities “features.” Many researchers use “features” to represent a building design (Zamanian et al. 1991; van Leeuwen and Wagter 1997; Fenves 2002) as instances of feature classes and attributes. Others use features to construct task-specific construction views from a project-specific geometric model (Haymaker et al. 2004; Boukamp 2006; McKinney et al. 1998).

Previous research identifies different types of spatial and non-spatial building-related information. They also formalize generic and specific schemas and constructs to facilitate reasoning. We leverage and build on those research efforts to provide simple, generic, and flexible query support to identify construction-relevant information. We use “features” to represent construction semantics of a building design, and we develop an ontology of features representing construction-relevant information, as described in Chapter 3, and leverage it to automatically extract and query features from an *IFC*-based BIM model.

#### **4.3.2 Reasoning about Building Product Models**

Many research efforts involving reasoning about building models have been undertaken. They add representation schemas and utilize task-specific reasoning structures in order to construct specific views using building product models. Some related work in this area includes the constructability adviser system (Fischer 1991), automated architectural code checking (Dym et al. 1988), automatic generation of project plans from a 3D model (Darwiche et al. 1988), time-space conflicts analysis (Akinci et al. 2002), handicapped accessibility analysis of an *IFC*-based product model (Han et al. 2000), MEP coordination (Korman et al. 2003), and generation and evaluation of daily construction work zones (Akbas 2004). This considerable body of work signifies the need for developing reasoning tools to leverage building product models to support various design assists or construction tasks. Many of these current approaches are inflexible and

hard-coded, leaving very little room for users to define their own requirements and customize their own rationale in the reasoning process.

Other studies have attempted to derive different topological or spatial relationships among building components from a 3D solid model (Nguyen and Oloufa, 2002, Haymaker et al., 2004) and the *IFC* model server (Chen et al. 2005). Some work has also been undertaken to add reasoning dimensions to the *IFC* model in order to access, browse, and navigate *IFC* data, explicitly defined in the *IFC* model (Katranuschkov et al. 2003).

Researchers in the mechanical and manufacturing domain use “features” to describe the shape and geometry of parts which constitute an assembly. They use them for reasoning about the topology and geometry of designed artifacts in order to support various design and manufacturing activities (Shah 1991; Cunningham and Dixon 1988). The extensive body of research in that domain explores recognizing or extracting features from the already-designed artifacts, or geometric models, from an analysis and interpretation of the product data, normally solid modeling (Shah and Mantyla 1995; Dixon and Poli 1995). Haymaker et al. (2004) and Staub-French et al. (2003) apply these concepts to construction.

Query-based approaches provide increased generic support to rapidly generate task-specific views of a product model (Haymaker et al. 2004). They act on the predefined model schemas or they support the definition of schemas to query a product model database. The research in mechanical engineering (Lou et al. 2003) investigates generic CAD query languages that enable engineers to query a model for geometric features. However, it does not provide simple, generic and explicit support to derive construction-relevant features from building models. Recent research has extended the application of spatial concepts and language developed in the *GIS* community to the AEC sector to develop a 3D spatial query language to enable the spatial analysis of objects in a BIM (Borrmann and Rank 2009). This work, however, provides limited support to extract construction-relevant information. It also excludes the richness of design information contained in the BIM and *IFC* models.

Some research efforts provide a general and flexible means to describe query information to handle partial model data from the *IFC* Model Server. Two such efforts are *Partial Model Query Language* (Adachi 2003) of the *Secom IFC Model Server* and the *Product Model Product Model Query Language* of the *EuroStep Model Server*. They provide query support for the retrieval of explicitly defined *IFC* properties and relationships, limited to simple containment

relationships from the *IFC Model Server*. However, existing query-based approaches and languages are not widely used in AEC practice today (Haymaker et al. 2004). One possible reason for this is due to the absence of simple, generic, formal and expressive framework which enables practitioners to explicitly define these queries.

In this research, we combine feature extraction and a query-based approach to reason about a BIM. We build on and extend previous research efforts to extend the breadth of reasoning dimensions supported. We develop a query vocabulary, query specification templates, and formalize the process for specifying queries. Our proposal involves a rich, expressive and flexible query support approach to query for a variety of design conditions that are not currently represented explicitly in a BIM.

### **4.3.3 State-of-the-Art BIM Analysis Tools**

Many tools leverage BIM and CAD models for checking and analyzing a building design. *Solibri Model Checker*© (*SMC*) can be used to visualize and analyze a BIM for its integrity, quality, and for compliance with a given set of design requirements, such as code checking (*SMC Inc.* 2010), using its object-based rule engines. It does not currently provide, however, support to flexibly query a BIM to derive construction-relevant information. *NavisWorks*© provides a suite of applications which includes interference checking and clash detection of geometric objects, etc. As well as additional time and efforts required to manually determine false positive interference and clash results between components, *NavisWorks* offers limited support in helping construction practitioners to query for other construction-relevant design conditions.

Many state-of-the-art tools provide quantity takeoff and BIM-based estimating. *On Centre*, which has yet to embrace BIM technology, provides some flexibility to take off quantities by allowing users to define different material conditions and apply them manually to digital representations (*On Centre Inc.* 2009). *Revit* also provides add-on internal support to takeoff material schedules and enables filtering or grouping of results, based on pre-defined data fields. *Innovaya* has BIM-based estimating solutions, which allows the estimator to visualize and analyze a BIM model, and provides object quantification and pricing (*Innovaya Inc.* 2009). While *Innovaya* provides tools to define conditions that specify quantities to be calculated and how objects should be grouped for quantification, it does not provide query support to identify

features that are of concern to construction practitioners. We build on current state-of-the-art tools to enable users to flexibly specify design conditions and query them from a given BIM.

#### **4.4 System Architecture for Querying Construction-Specific Design Information**

The overall process for querying features is illustrated in Figure 4.5 as an *IDEF0* model (Colquhoun et al. 1993), which shows the user and computer actions, the controls, and the results for each step or function. In the first step (Create Feature-based Model), the input *IFC*-based BIM model is transformed into a project-specific feature-based model (FBM) which explicitly represents the features that are important to a particular construction practitioner or domain (see Chapter 3 for details). In the second step (Query Features), the main focus of this chapter, users configure queries that operate on the input feature-based model by leveraging *query specifications*. Query specifications formalize the language and structure required to build and answer user-driven queries in relation to a BIM. In this study, we describe in detail, the query specifications, including the underlying formalized domain knowledge, and subsequent query interfaces, designed to support query formulation.

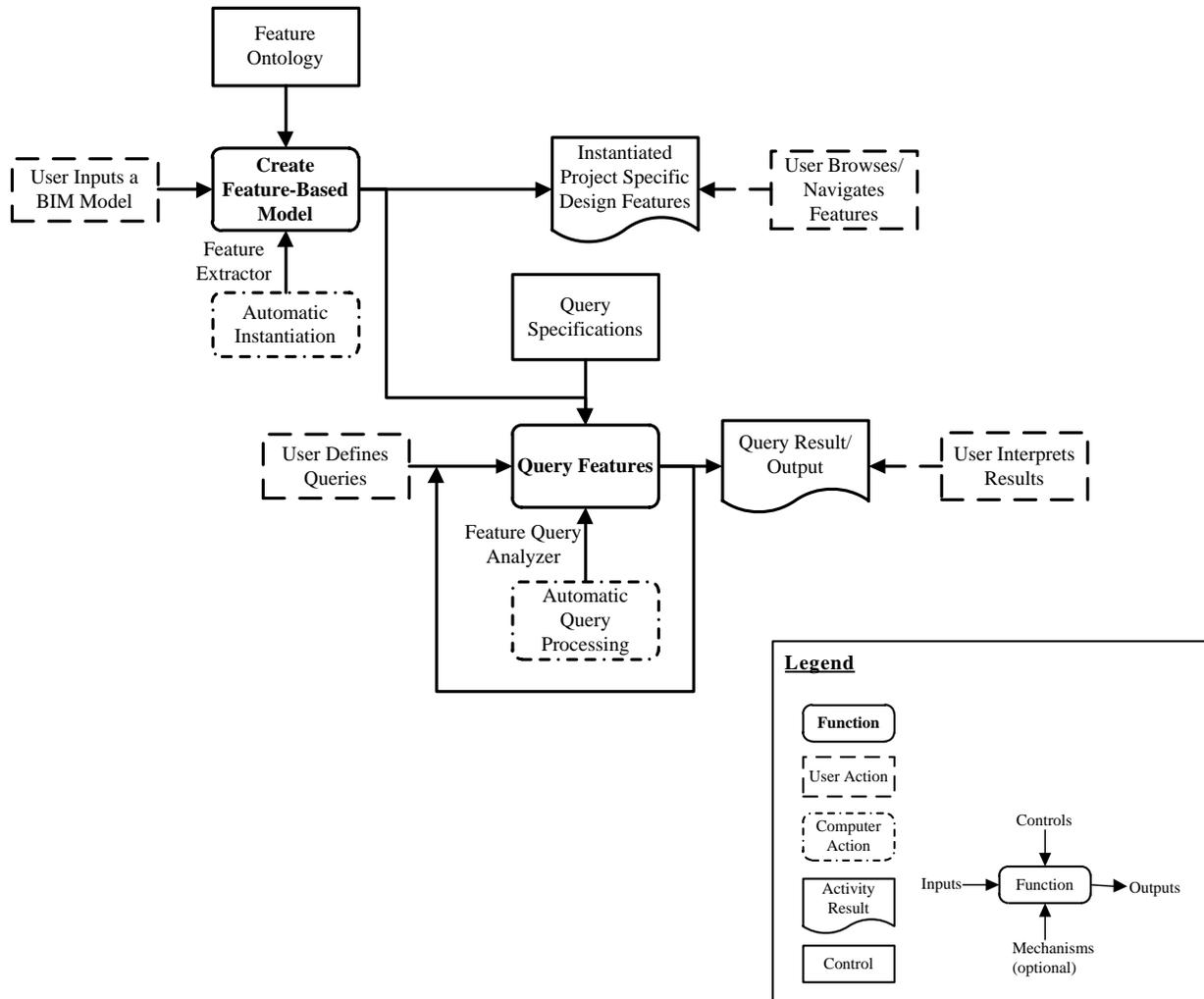


Figure 4.5 An IDEF0 model architecture for automatically querying features

#### 4.5 Specification of Construction-Relevant Queries

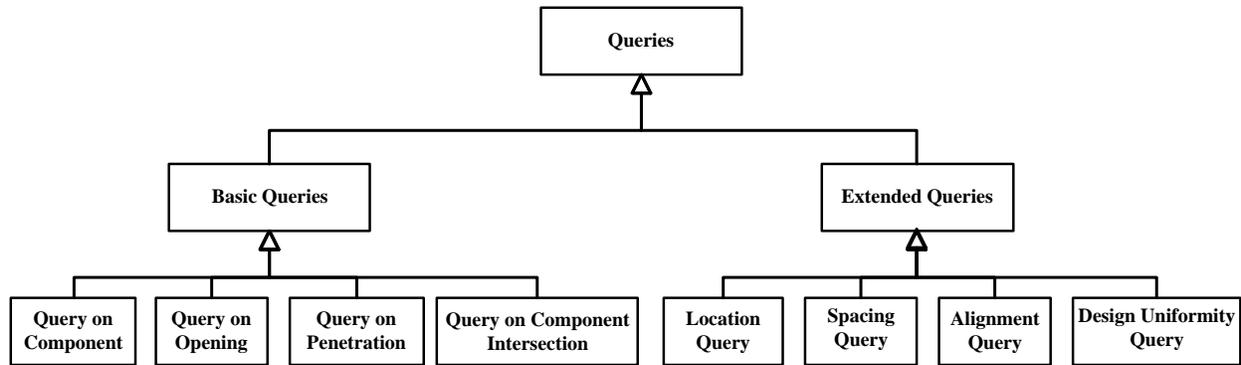
Query specifications provide a formal and structured query vocabulary to specify queries on features, formalized in the feature ontology and instantiated in the FBM. The vocabulary defines different types of queries and formalizes relevant attributes and knowledge necessary to specify queries. The research challenge with respect to formalizing query specifications is that construction practitioners require different types of queries, use different means to express queries, and require different levels of knowledge specifications for describing queries. Moreover, query specifications should ideally allow practitioners to easily and intuitively define queries without having BIM experience or requiring programming skills.

In an effort to address these challenges, this research develops a series of reusable, customizable and computer interpretable query specification templates. The templates leverage the feature ontology and query attributes to characterize each query type. The templates, which use form-based interfaces, are customizable, expressive and easy-to-use to interactively specify queries. They are designed to be visually intuitive to the end user who is familiar with the domain concepts but normally does not have sufficient knowledge of the *IFC* model, underlying data models of a BIM, or query languages. Completing the forms is a simple task; the end user requires little technical training to learn how to correctly complete a form. It is not surprising that a form-based query interface is the preferred means by which to provide an unsophisticated query environment to average users in many database applications (Jayapandian and Jagadish 2008). For each type of query, a form or a collection of query specification forms, allows the user to select a feature, filter feature properties, and specify attributes, and other query related parameters, to precisely define a query.

In the next section, we highlight different types of queries and describe the attributes used to specify queries in the query specification templates.

#### **4.5.1 Classifying Queries**

We classify queries into two categories: basic queries, and extended queries (see Figure 4.6). Basic queries relate to the manipulation of feature attributes, exclusively defined in the feature ontology and project specific values instantiated in the FBM. These enable users to query component, opening, penetration and component intersection features. Extended queries contain the underlying concepts of basic queries, but also incorporate additional concepts, not represented explicitly in the feature ontology, and the resulting FBM. The queries belonging to this category relate to higher-level spatial concepts and phenomena, which in the current scope of our work, includes queries on spacing, location, alignment and uniformity of features.



**Figure 4.6** A taxonomy of queries

### 4.5.2 Specifications of Basic Queries

This research provides a generic vocabulary and query interfaces to query components, component intersections, openings, and penetration features, based on the feature attributes, defined in the feature ontology, or constrained property values, instantiated in the FBM. Examples of ‘basic’ queries include: (1) identifying the existence (or type) of penetrations in a slab, similar to the annotated drawings shown in Figure 4.2 for slab penetrations on the EDC project; and (2) grouping and quantifying (or aggregating) the walls on a particular floor based on wall type and size, similar to the ‘highlighted’ drawing shown in Figure 4.4 that was created by cost estimators for the UBC-O project. Figure 4.7 shows the list of attributes applicable to all basic queries and those that are specific to a particular type. These attributes provide a language for practitioners to query for features, or for specific aspects of the features in a given FBM.

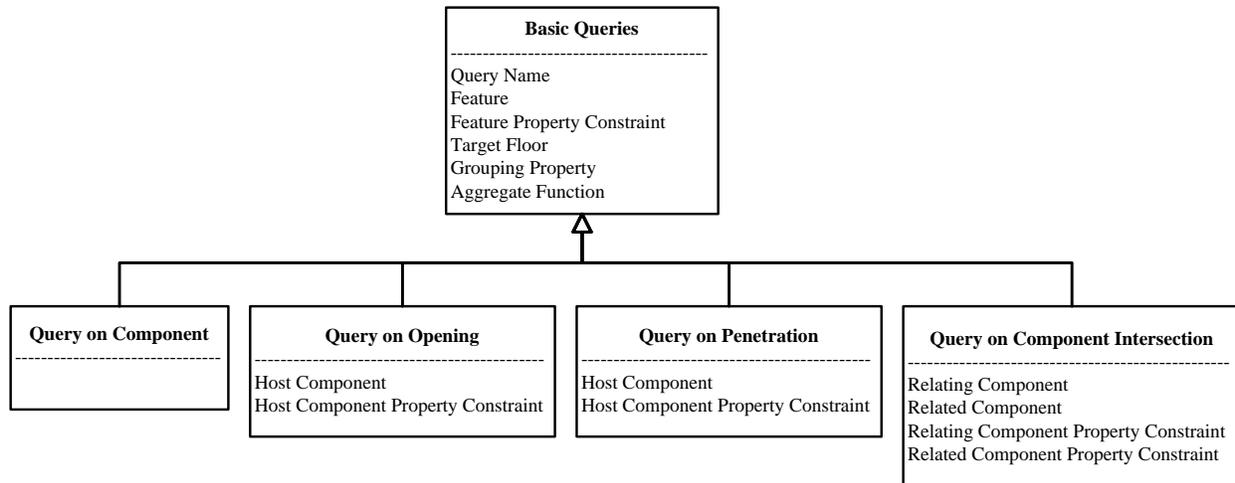


Figure 4.7 Basic queries and associated attributes formalized in this research

#### 4.5.2.1 Common Attributes for Basic Queries

A number of attributes are common to the specification of all the basic queries. These common attributes are briefly explained below:

1. *Query Name*: This represents a practitioner’s preference for naming the query. It is a simple, yet powerful way to communicate about the problem domain, using natural language. The naming of the query is also important, as it can reside in the query library for later retrieval and use.
2. *Feature*: Each query has a feature associated with it. The practitioner selects a feature to query from the list of available features, such as “wall,” under the component feature.
3. *Feature Property Constraint*: This attribute allows practitioners to filter the properties of a feature to define a query. Users can pick a set of properties available for the selected feature, and assign constraints to these properties to precisely define a query. For example, specifying the ‘fire-rating’ and ‘material’ property of a wall, such as 2-hr fire-rated dry wall, to focus on fire-rated dry wall.
4. *Target Floor*: Users choose a floor, or a set of floors, applicable to the given BIM model for which a query has to be run. It should be mentioned here that query results are always shown by floor.

5. *Grouping Property*: This attribute allows practitioners to select a grouping property, or properties, corresponding to a feature. It helps them to categorize the feature instances into user-defined groupings, such as groupings of walls by type, or columns by size.
6. *Aggregate Function*: We use this attribute to represent simple quantitative measures, such as count, maximum, minimum, percent count, percentage variation, summation to allow users to quantify query results, such as the total length of walls, or maximum wall height.

#### 4.5.2.2 Attributes Specific to Opening and Penetration Queries

In addition to the above-described common attributes, the specification of queries on openings and penetrations requires the following two additional attributes to characterize the type of opening or penetration.

1. *Host Component*: This attribute allows the user to define the component for which the instances of openings or penetrations are queried. For example, by selecting the host component ‘slab,’ when querying for a penetration feature, the user will identify all slab penetrations in a given model.
2. *Host Component Property Constraint*: This attribute allows the user to further constrain opening or penetration queries by the type of host component, such as constraining the query to identify duct penetrations on fire-rated dry walls.

#### 4.5.2.3 Attributes Specific to Component Intersection Queries

Practitioners have different preferences as to which component intersections are important for their domain. In addition to the common attributes described earlier, the following additional attributes allow the user to represent a preference in defining the intersections between components. The definition of relating and related components follows the *IFC* model specifications. *IFC* defines two sides of the objectified relationships as: Relating + <name of the relating object> and Related + <name of related object, to handle relationships among objects>. We use a similar representation to specify component intersection queries.

1. *Relating Component*: Reference to the first component that is chosen for specifying a component intersection query. For example, in the query for identifying all component intersections involving masonry walls, the feature “wall” is specified as *relating component*.

2. *Related Component*: Reference to the second component that is chosen for specifying a component intersection query. In the above-mentioned query example, the features such as “column,” and “slab” are chosen as the *related component*. Even the same feature “wall” can be chosen as the *related component*, if it is also required to find intersections between masonry walls and another type of wall, such as dry wall.
3. *Relating Component Property Constraint*: Defines a particular type of relating component by restricting properties of the relating component. For example, in order to define a masonry wall in the example query, the relating component’s material property is defined as the one containing the brick and concrete masonry unit (CMU).
4. *Related Component Property Constraint*: Defines a particular type of related component by restricting properties of the related component. For instance, for the same example query, the related column’s shape property can be defined as a round column, in order to find the instances of intersections between masonry walls and round columns.

### 4.5.3 Extended Queries

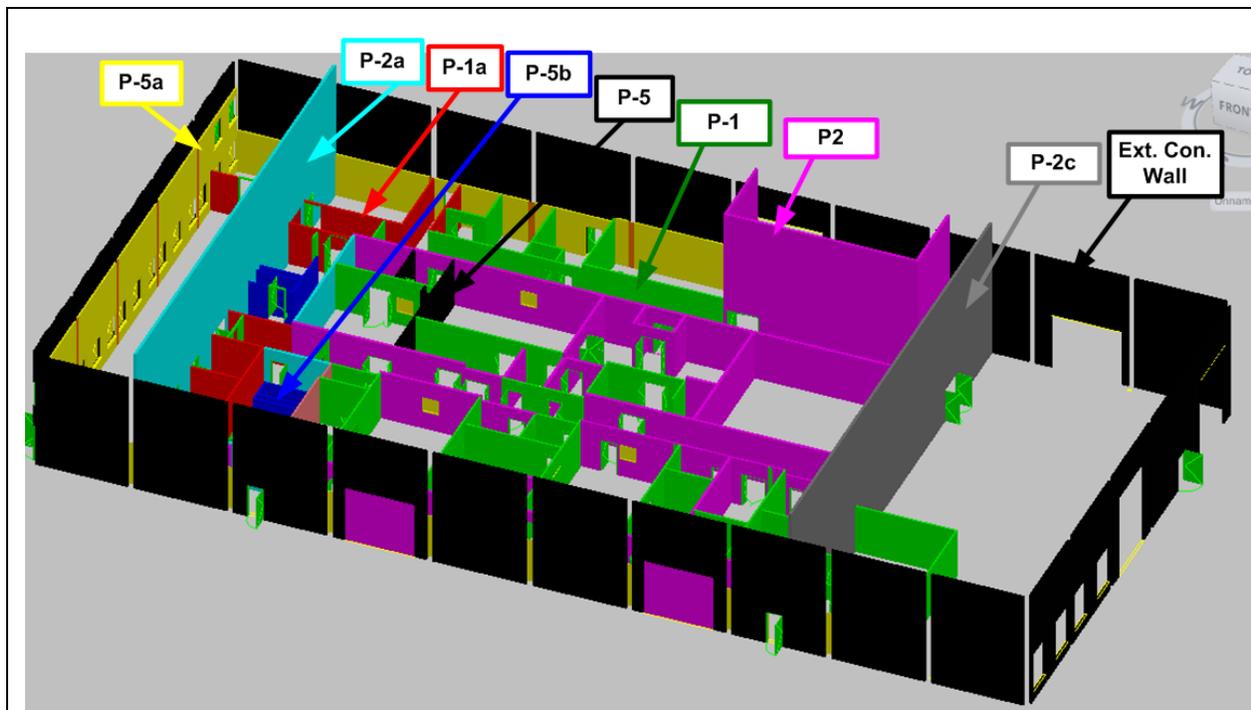
Extended queries build on and extend the basic queries to incorporate spatial and other higher-level concepts of features. These queries emerge from the spatial location of features, spatial relationships between features, or feature geometry. Specifically, the extended queries formalized in this research are limited to queries on feature location, spacing, alignment and design uniformity. These are common types of concepts, identified in the literature, and that we observed in the projects studied and had confirmed by industry experts. We do not propose that these are the only types of extended queries required by construction practitioners. However, in this research, we limited our scope to the above mentioned types because our intention was to show how the feature ontology and query specifications framework could help to process challenging and higher order reasoning about a BIM. Next, we provide an overview of the extended queries, and then we describe the specific attributes formalized for each query type.

- a) ***Location Queries***: Location queries are used to identify the location of features relative to some frame of reference, such as the location of columns with respect to related grid lines. Being able to quickly identify the location of features is an important consideration for construction planning, constructability assessment, and facility management (Allen and Iano,

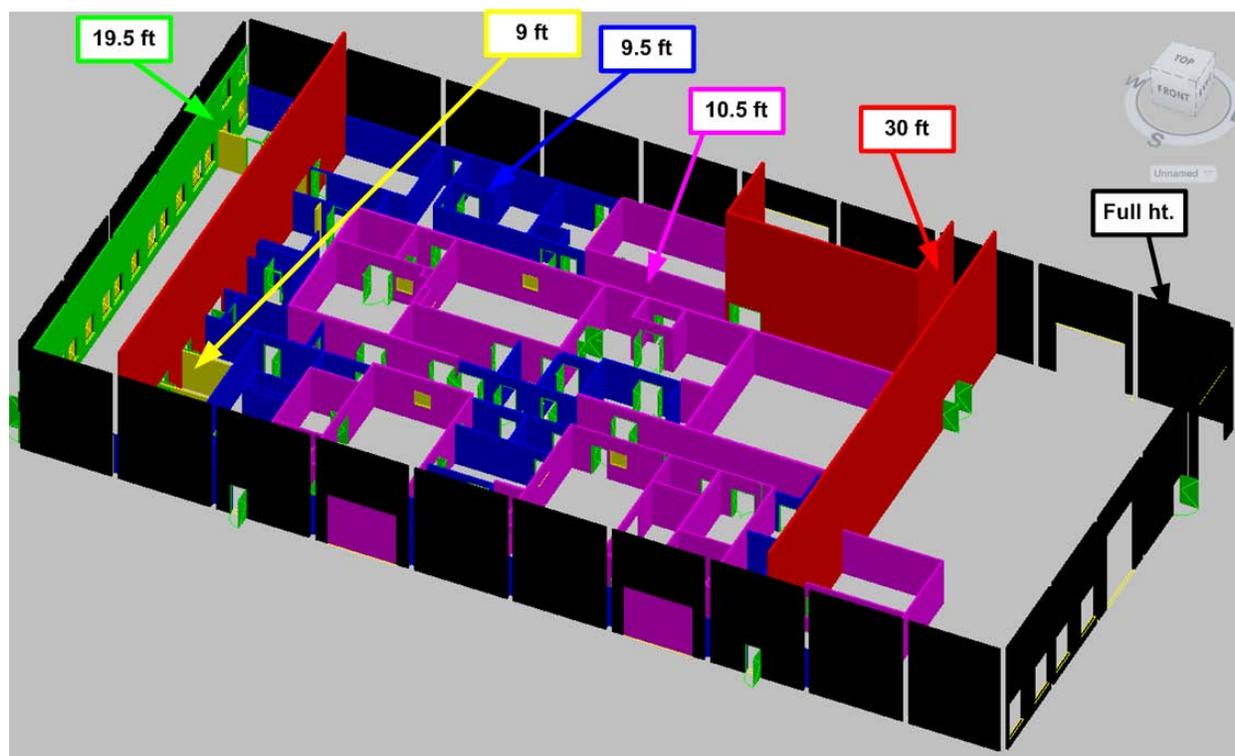
2002). For instance, knowing where all penetrations occur, will result in more accurate cost estimates of the work (Bisharat 2004). An example location query that practitioners might ask is: “*Identify the location and size of all penetrations on walls.*”

- b) **Spacing Queries:** Spacing queries are used to identify the distance between (or spacing of) proximate features of the same type or different types. Spacing between similar features, such as the spacing of columns or the spacing of openings can impact construction planning, constructability assessment and formwork selection (Fischer and Tatum 1997). Spacing between dissimilar features (e.g., spacing between walls and adjacent ducts) can be important during the construction and operation of the facility (Korman et al. 2003; Tabesh and Staub-French 2006). An example spacing query that practitioners might ask is: “*Identify the maximum and minimum clear spacing between proximate columns.*”
- c) **Alignment Queries:** Alignment queries are used to identify the orientation and/or placement of the instances of a feature with respect to some criteria. The purpose is to find unaligned features, if any, present in a given design. The proper alignment of features, such as the column alignment in a floor, or from floor to floor, is crucial for the constructability of a design and the installation of façade and curtain walls (Fischer and Tatum 1997; Allen and Iano 2002). An example alignment query that practitioners might ask is: “*Identify columns that do not align horizontally or vertically.*”
- d) **Design Uniformity Queries:** Design uniformity is an important factor for practitioners as it can influence the method selection, constructability of a structure (Fischer and Tatum 1997; Hanna et al. 1992), and technical and economic feasibility of using a particular construction method (Udaipurwala and Russell 2005). Design conditions that are normally relevant from a design uniformity perspective include those that relate to feature properties, such as size and/or dimension (width, thickness, diameter, length, depth, height) or other extended properties, such as location and spacing of features (e.g., the variation in the openings’ size and location on walls, change in column size, and location from floor to floor). Design uniformity queries are used to identify or gain insights about the consistency or variation of design features in a particular floor, or from floor to floor, in a building. They generally fall

into two categories: (i) identifying the cluster of similar components or features; and (ii) recognizing non-uniform features. The first category identifies or creates a grouping of similar features based on simple geometric or nominal attributes, and calculates some measure of variation, such as count, percent count, etc. An example query that practitioners might ask is: “*Show me the variation in wall types and height in a building.*” Figure 4.8 shows examples of how uniformity might be assessed for walls based on (i) wall type (Figure 4.8 a), and (ii) wall height (Figure 4.8 b). As the figure shows, the process of filtration and grouping of components based on uniformity creates clusters of similar walls, which is useful information to practitioners. Additionally, component similarity can also be evaluated by combining nominal and quantitative attributes and by using a simple matching approach (Staub-French and Nepal 2007). The second category of design uniformity involves assessing the uniformity of features in terms of their nominal (e.g., material), geometric (e.g., size, shape) or topological attributes (e.g., location, spacing). A typical query of this type that practitioners might ask is: “*Identify columns that have a change in size, shape or location from floor to floor.*”



(a) Variation in wall types



(b) Variation in wall height

Figure 4.8 Groupings of uniform walls based on: (a) wall types and (b) wall height

#### 4.5.4 Specifications of Extended Queries

In order to assist construction practitioners to precisely formulate extended queries on a BIM, a general, consistent, and precise specification of a vocabulary of extended queries is needed. In order to address this, we define several attributes that characterize extended queries. Figure 4.9 shows the taxonomy of the extended queries with the relevant attributes formalized in this research. We formalized these attributes based on the critical review and analysis of the literature case studies, and our interviews with construction experts. Users employ formalized query attributes to precisely specify queries that reflect their preferences and specific rationale.

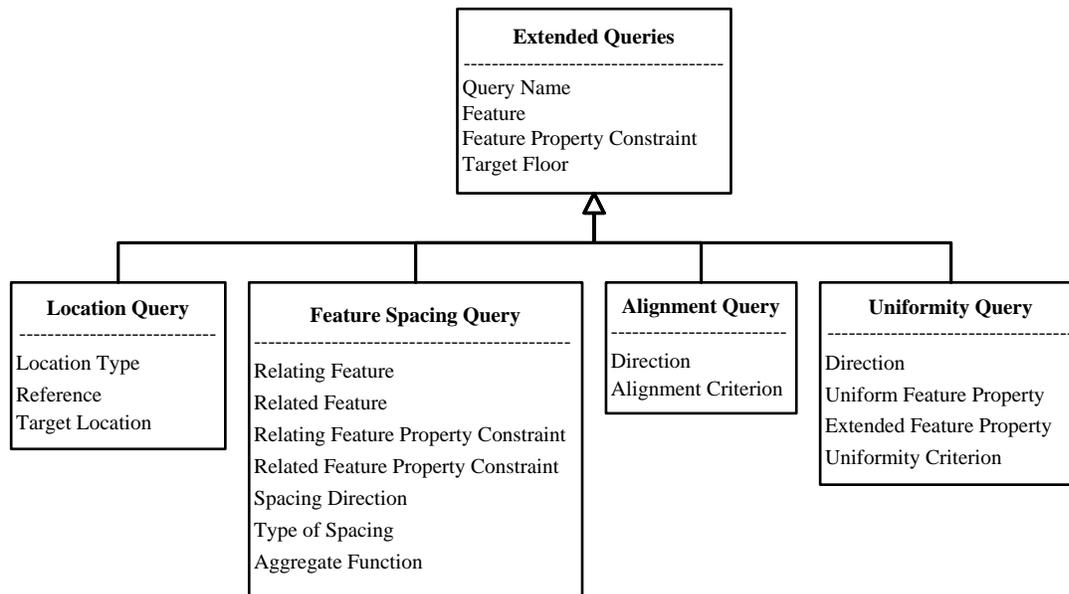


Figure 4.9 Extended query types and associated attributes

##### 4.5.4.1 Common attributes of Extended Queries

The attributes that are commonly applicable to specify extended queries discussed in this research include:

1. *Query Name*: This represents a practitioner’s preference for naming an extended query in the natural language form.
2. *Feature*: Extended queries act on features. This attribute allows a practitioner to select a feature of their interest, for which an extended query is run. For example, in the query to

find the maximum and minimum spacing of columns, the component feature “column” is selected.

3. *Feature Property Constraint*: This attribute allows a practitioner to filter the properties of the selected feature in order to define an extended query. Users can choose a set of properties, relevant to the selected feature and assign constraints to these properties, to precisely define an extended query. In the above-mentioned query on spacing, the practitioner, for instance, can specify the column’s property as “exterior” to identify the spacing of exterior columns only.
4. *Target Floor(s)*: This allows the user to specify a floor or a set of floors, corresponding to the given BIM model for which a query has to be run.

Other attributes are specific to different query types, as shown in Figure 4.9. The following sections describe these query-specific attributes in further detail.

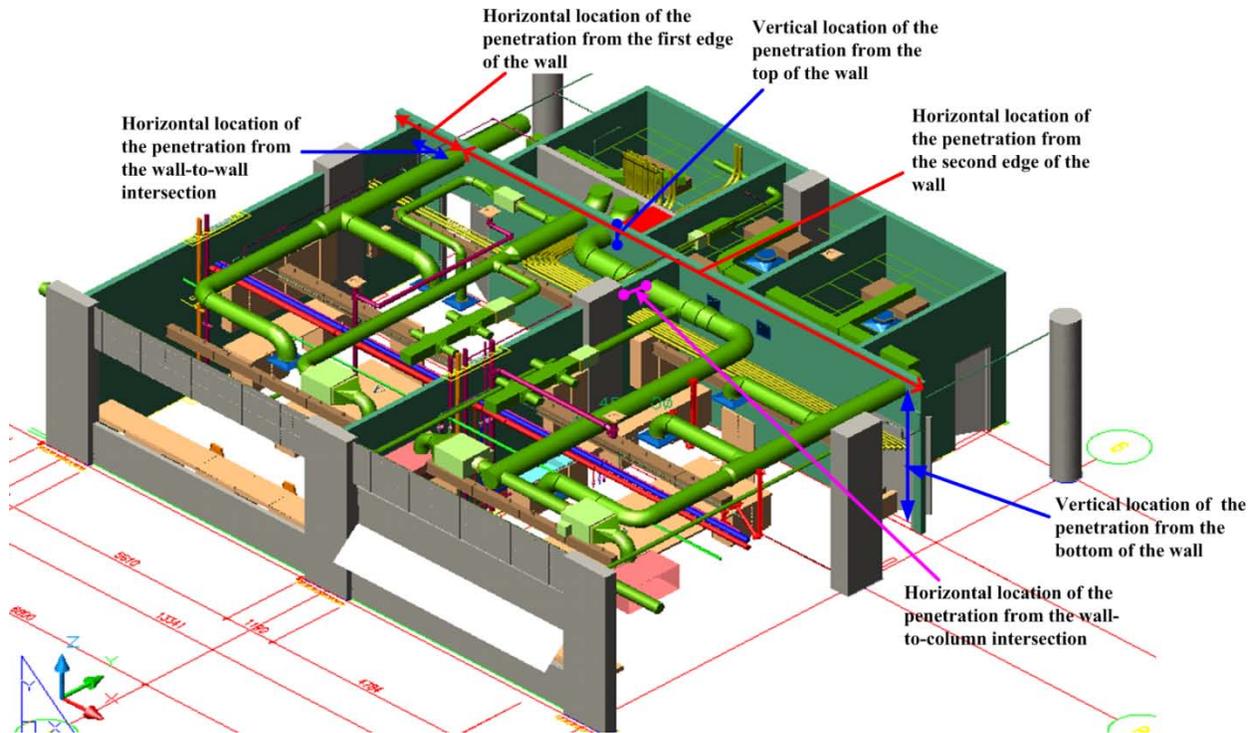
#### **4.5.4.2 Attributes Specific to Location Queries**

In addition to the common attributes described above for specifying extended queries, the following attributes are required to specify location queries. Some common applications of these types of queries include, identifying the location of penetrations and openings, and locating columns with respect to specific grid lines.

1. *Location Type*: This represents a practitioner’s preference for querying either the ‘horizontal’ or ‘vertical’ location of features. The location of openings and penetrations on components can be characterized as either horizontal or vertical assessed relative to a stated reference. The location of columns, for example, is generally assessed horizontally, relative to related grid lines.
2. *Relative Reference*: This attribute allows practitioners to specify the reference for the horizontal or vertical location of a feature. For instance, as shown in Figure 4.10, the horizontal and vertical location of duct penetrations on walls (similar terminology applies for openings on components) can be specified in a number of ways. The vertical location of a penetration can be defined based on: the distance from the top of the host wall, or from the bottom of the host wall, and the distance to the floor level. The horizontal location of a penetration can be designated from either edge of the host wall. It can also

be referenced from the intersection of the host component with other components, such as a wall to wall intersection or a wall to column intersection.

3. *Target Location*: This attribute allows practitioners to specify the location of interest of the selected feature, either as the ‘feature centre’ or ‘feature boundary,’ referred with respect to the selected *relative reference*. For instance, the user defines the ‘*target location*’ of duct penetration as ‘feature centre’ to specify that the location of all duct penetrations be measured up to the centre of each penetration. If specified as ‘feature boundary,’ the location of duct penetrations is measured to the proximate boundary of the penetration from the relative reference.



**Figure 4.10** Illustration of the different attributes required to locate penetrations on walls, based on a lab in the Chem-Bio Building project, UBC

#### 4.5.4.3 Attributes Specific to Spacing Queries

The following attributes are relevant to specify feature spacing queries. Figure 4.11 illustrates some of the spacing attributes for specifying the spacing between proximate columns.

1. *Relating Feature*: Reference to the first feature that is chosen for querying spacing. For example, to identify the spacing between proximate columns, the *relating feature* is specified as “column.”
2. *Related Feature*: Reference to the second feature that is chosen for querying spacing in relation to the first feature. This attribute is needed to provide sufficient flexibility so that practitioners can identify the spacing between different kinds of features (e.g., the spacing between columns and walls), and similar kinds of features (e.g., the spacing between columns). For the above-mentioned query, the related feature is also a “column” because we are querying for the spacing between proximate columns.
3. *Relating Feature Property Constraint*: Defines a particular type of *relating feature* by restricting a property (or properties) of the relating feature.
4. *Related Feature Property Constraint*: Defines a particular type of related feature by restricting a property (or properties) of the relating feature.
5. *Spacing Direction*: This attribute designates the direction of spacing, depending on the type of relating and related feature, the spacing will be assessed in the ‘horizontal’ direction (horizontal spacing) or ‘vertical’ direction (vertical spacing). For components, such as columns, which are normally placed relative to rectangular grid lines (Figure 4.12), the spacing is assessed in the horizontal direction, and can be further evaluated as spacing along the X- or Y-axis. The spacing of openings or penetrations could be assessed in the horizontal or vertical direction.
6. *Type of Spacing*: This attribute denotes the spacing between features as ‘centre to centre,’ or ‘clear’ spacing, which is illustrated in Figure 4.11.
7. *Aggregate Function*: This attribute is used to specify the desired quantification of the query results and includes functions such as ‘maximum’, ‘minimum’, ‘average’, ‘percentage variation’, etc. For instance, the user, who is interested in identifying the maximum and minimum spacing of proximate columns, specifies the aggregate function as ‘maximum’ and ‘minimum’.

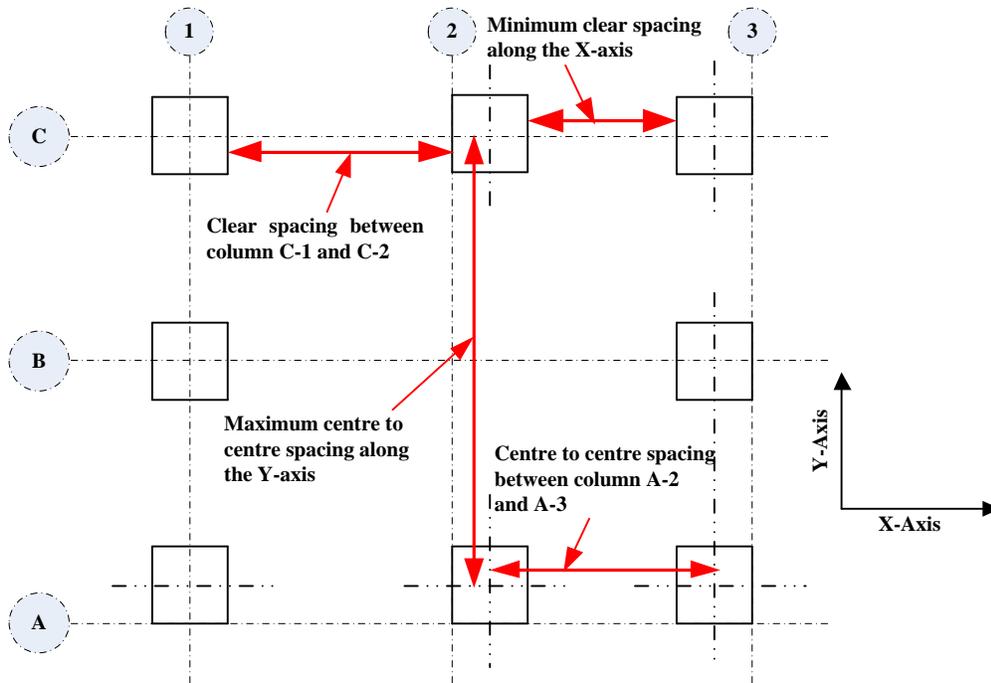


Figure 4.11 Illustration of some attributes for column spacing

#### 4.5.4.4 Attributes Specific to Alignment Queries

The following set of attributes is generally relevant to specify alignment queries of features, such as identifying the horizontal and vertical alignment of columns.

1. *Direction*: The direction to which alignment has to be assessed, either as 'horizontal' (across the floor) or 'vertical' (from floor to floor). Horizontal alignment, depending on the feature, can be further categorized as alignment along the X- or Y-axis. For example, *column alignment* can be used to assess whether a group of columns are aligned in the X- or Y-direction.
2. *Feature Alignment Reference*: The reference to the feature's geometry used to assess alignment. This attribute is generally applicable to component, opening and penetration features. The alignment of these features may be assessed along their centres (or centre lines) or edge(s) or face(s) (see Figure 4.12 for illustration of these terms).
3. *Alignment Criterion*: This represents the definition of conditions or criteria used to define the alignment of a feature. For example, criteria that can be used to identify unaligned columns, includes the following: 'on-grid column location,' 'colinearity of columns'

centre’, ‘edge orientation’, and ‘identical size/dimension.’ Practitioners can choose a single or a combination of criteria to identify columns that do not align in a floor, or from floor to floor. Some of these criteria that can be used to define the horizontal alignment of columns are discussed in Section 4.6.2.3.

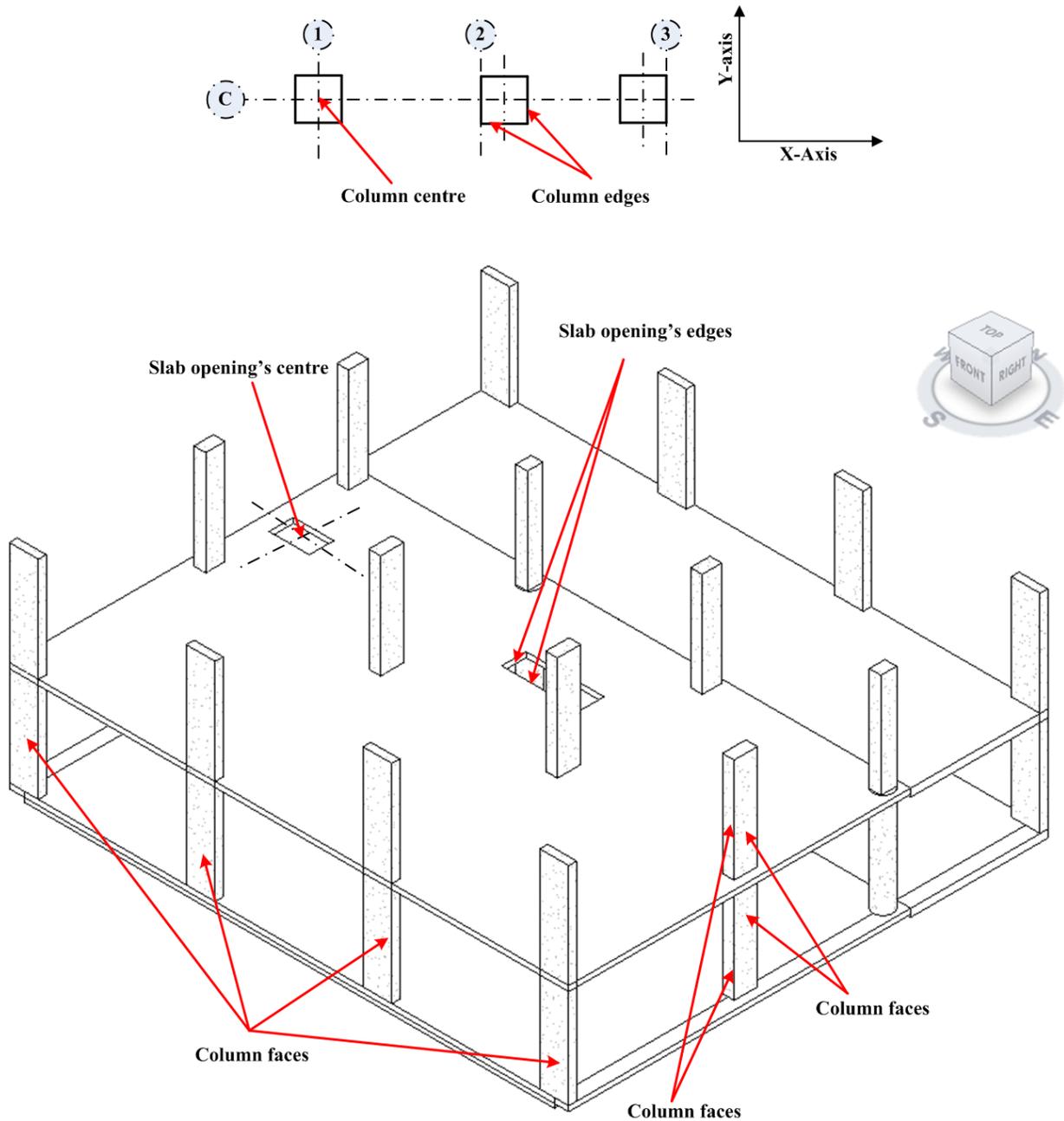


Figure 4.12 Illustrations of the “feature alignment reference” attribute for opening and column features

#### 4.5.4.5 Attributes Specific to Design Uniformity Queries

We have highlighted some characteristics of design uniformity queries in Section 4.5.3. The attributes that characterize the uniformity queries generally include the following:

1. *Direction*: This is a common attribute for assessing design uniformity queries. It represents the direction for which the uniformity has to be assessed, and consists of uniformity either in the horizontal direction or vertical direction. A horizontal direction represents uniformity across a floor; and the vertical direction represents uniformity from floor to floor (Hanna et al. 1992).
2. *Uniform Feature Property*: This is the property (or properties) of a feature that will be considered in the assessment of a design uniformity query. For example, grouping of walls based on wall ‘thickness’ and assessing the similarity of walls based on wall ‘height’ and ‘wall type.’
3. *Extended Feature Property*: This attribute allows users to specify a spatial property, if any, to be considered in the design uniformity query. Common extended properties include *spacing* and *location* of features. The following query would involve assessing the uniformity of the location of columns from floor to floor: “*Identify the columns in a base floor that have different locations in the floor above.*” This would involve specifying ‘location’ as the *extended feature property*.
4. *Aggregate Function*: The attribute ‘aggregate function’ lets users quantify all feature instances, or instances within each group, in terms of user-defined quantitative measures, such as ‘count,’ ‘maximum,’ ‘minimum,’ ‘percent count,’ and ‘percentage variation.’ It, thus, provides a general indication of the variation of features expressed in terms of simple aggregate functions. This attribute is mostly relevant for uniformity queries for groupings of similar feature instances. For instance, when grouping walls based on wall height, the user selects ‘count,’ ‘maximum,’ ‘minimum,’ and ‘percent count’ aggregate functions, the query output identifies the total instances of walls for each wall height, maximum and minimum wall height from all wall instances, and percentage of wall instances for each wall height.
5. *Permissible Percentage Variation*: This attribute allows the user to input an acceptable percentage variation for the uniform feature property (or properties), or extended property, in the uniformity assessment, and can affect the formation of clusters of similar

feature instances. For example, in an expert's opinion, a six percent variability in the dimensions of slab-bays is acceptable for use in flying truss formwork as it can be accommodated with infill or hinge panels (Udaipurwala and Russell 2005). Flying truss forms can also handle about 20% variation in opening size and location (Hanna et al. 1992).

6. *Uniformity Criterion*: This attribute is needed to define conditions or criteria (or criterion) that can be used to further specify uniformity queries. This can involve a set of rules, or constraints, that are considered for assessing the design uniformity of features. For instance, walls should be of the same size, location and height from floor to floor to use a tunnel formwork system when building slabs and roofs (Hanna et al. 1992).

## **4.6 Re-Usable and Computer Interpretable Query Templates**

We developed project-independent and computer interpretable query templates for specifying each type of basic and extended query. The templates provide user-friendly interfaces for specifying each type of query. These form-based templates leverage the feature ontology and query attributes to provide the user with structured and guided help, and a visual interface to specify queries. We structured query attributes into common query methods (they are represented as different query tabs in the templates), to guide the user in a step-by-step query formulation process. The intent is to enable users to specify queries in an interactive and intuitive manner. This section also shows the results from some of the example queries in XML format.

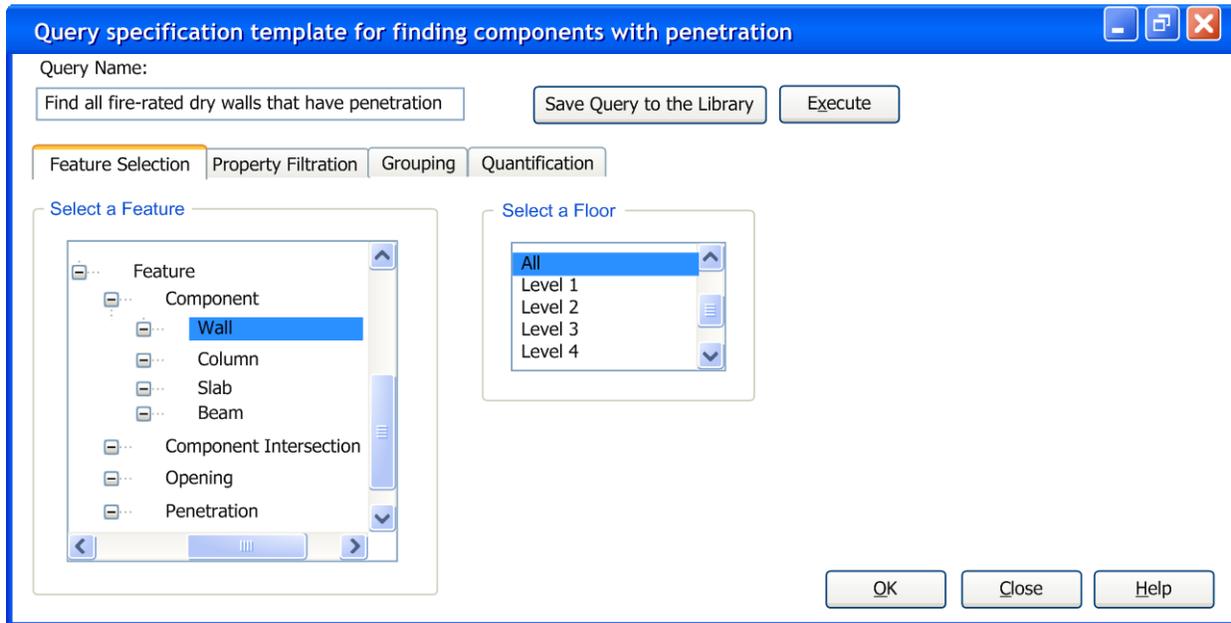
### **4.6.1 Query Templates for Formulating Basic Queries**

The main steps for formulating basic queries include: (1) Feature Selection; (2) Property Filtration; (3) Grouping; and (4) Aggregation. We used different query examples to illustrate these steps and to highlight the use of query-specific attributes.

#### **4.6.1.1 Example # 1 Query on a Component Feature**

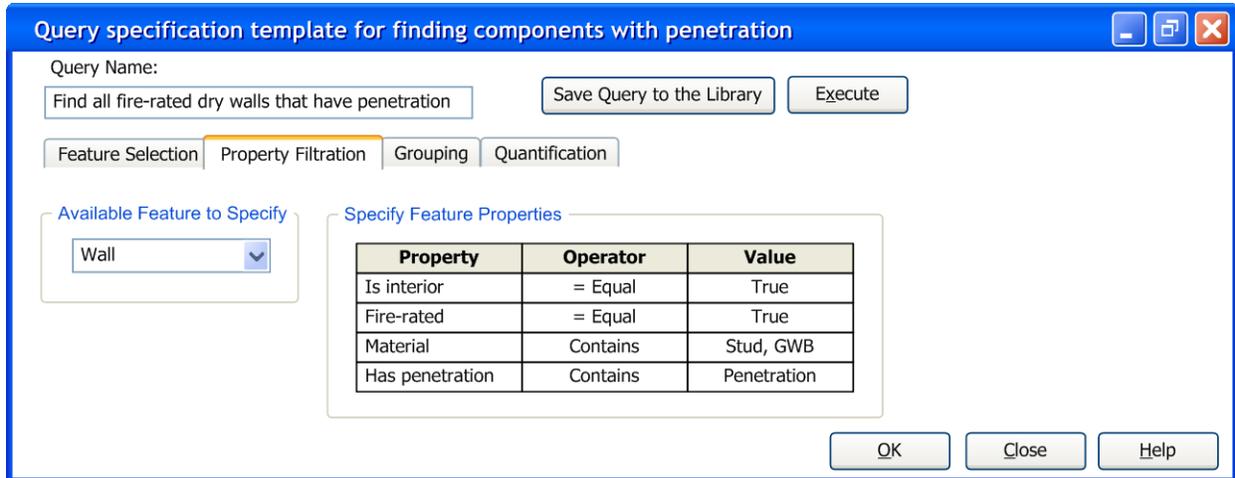
Consider the query: "*Identify all fire-rated walls that have penetrations.*" This query is relevant to drywall contractors because penetrations on fire-rated dry walls require additional work for fire stopping, weather resistance, sound insulation, and penetration sealant. Four steps are required to input this query.

**Step 1: Feature Selection** The first step in query formulation is the selection of a feature, such as a “wall.” The user also specifies a floor (using the *target floor(s)* attribute) for which the query will use in processing (Figure 4.13).



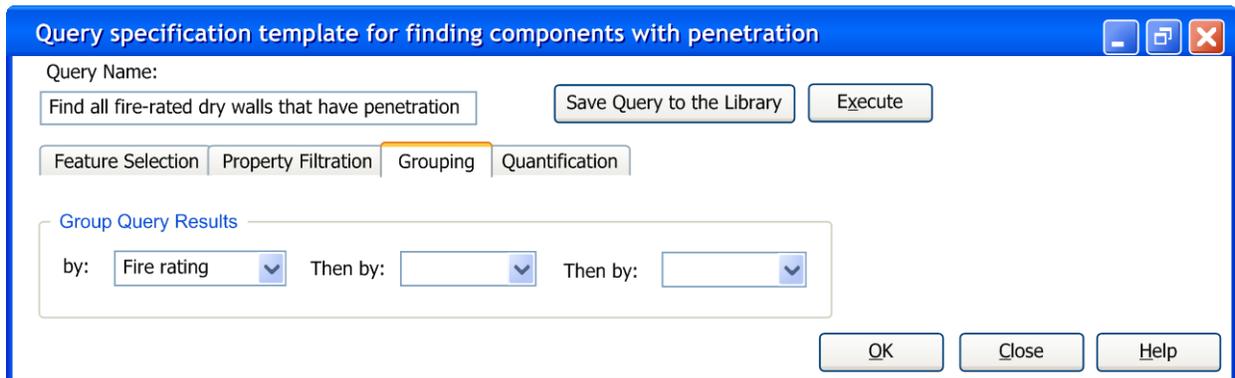
**Figure 4.13 Step 1 – Feature selection**

**Step 2: Property Filtration** In the next step, the user filters properties (or defines constraints) according to the feature selected in Step 1. For the given example query, the user defines the wall properties ‘is interior,’ ‘fire-rated,’ ‘material,’ and their corresponding values, to define the dry wall, and the relational property ‘Has penetration’ to designate walls with penetrations (Figure 4.14).



**Figure 4.14 Step 2 - Property filtration**

**Step 3: Grouping** After property filtration, the user can define a grouping property (or properties), such as grouping the instances of all dry wall, based on fire-rating values, e.g., 1 hr, 2 hr, etc. (Figure 4.15). The user can define other grouping criteria as well, such as grouping walls based on wall height. When multiple grouping criteria are specified, the sequence of grouping follows the order in which the grouping parameters are specified.



**Figure 4.15 Step 3 – Grouping**

**Step 4: Aggregation** Oftentimes, practitioners need to quantify or aggregate query results. This functionality is facilitated through the *Quantification* method, which allows the user to select numeric parameter(s) and define the aggregate functions to quantify feature properties, such as calculating the total length of walls (see Figure 4.16) that fulfill the constraints, as defined in the

property filtration and grouping steps. It is noteworthy to mention that the grouping criteria takes precedence over the quantification in that the quantification is applied to each group of features defined in the grouping criteria. It is not mandatory, however, that the grouping or quantification criteria be specified; the user may not use any grouping criterion at all, but use quantification only, or use the grouping, but not the quantification. The query templates provide the user with these options to manipulate the features and their attributes according to their needs and preferences.

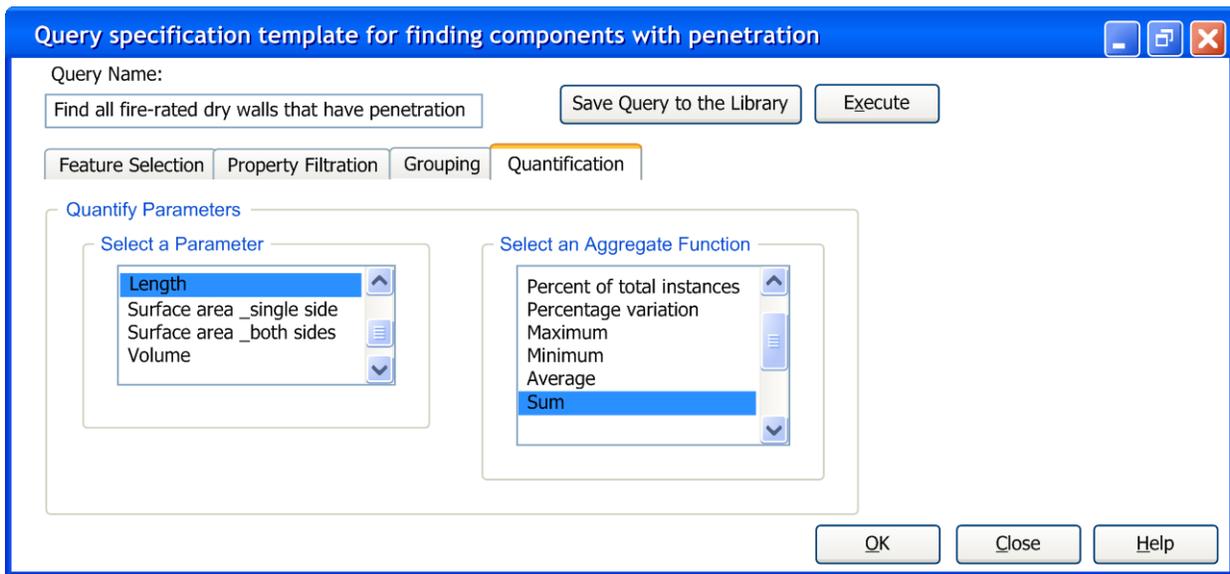


Figure 4.16 Step 4: Quantification

#### 4.6.1.2 Example # 2 Query on Opening (or Penetration) Feature

Consider the query: “Identify all openings on interior non-load bearing walls, and group them by type and then by size.” This query is of particular interest for trades involved in interior construction for framing openings. While the specification of this query follows similar steps as described for the query in Example # 1, it also illustrates the different ways a query can be expressed. Identifying openings (or penetrations) on walls may sound similar to identifying walls with openings (or penetrations). But the former query has a different focus and intention than the latter, where the focus is on identifying the instances of walls, as opposed to the instances of openings (or penetrations). Four query steps are involved in formulating query Example # 2, which are briefly highlighted next.

**Step 1: Feature Selection** As in the previous example, the specification of this query begins with the user selecting a feature, in this case the “opening” feature, as well as the floor for which the query will be processed.

**Step 2: Property Filtration** Openings (as is the case for penetrations) are hosted by components. As such, the user needs to define the “host component” in the ‘*property filtration*’ step (see Figure 4.17 Step 2a). The host component for this example is specified as “wall.” The user could define, if necessary, additional available properties for the feature “opening” to specify a specific type of opening, such as the opening of a certain size range.



**Figure 4.17 Step 2(a) – Filtration - Specifying properties for the feature “Opening”**

It should be noted that users can also specify properties for any feature that is referenced or linked in the query specification process, in order to precisely define a query. As illustrated in Figure 4.18 and Step 2(b), the user can choose the feature “wall” from the ‘Feature Available to Specify’ dialog box and define wall’s properties as ‘Is Interior,’ and ‘Is Load Bearing,’ as true and false, respectively, in order to define the wall type as an interior, non-load bearing wall.

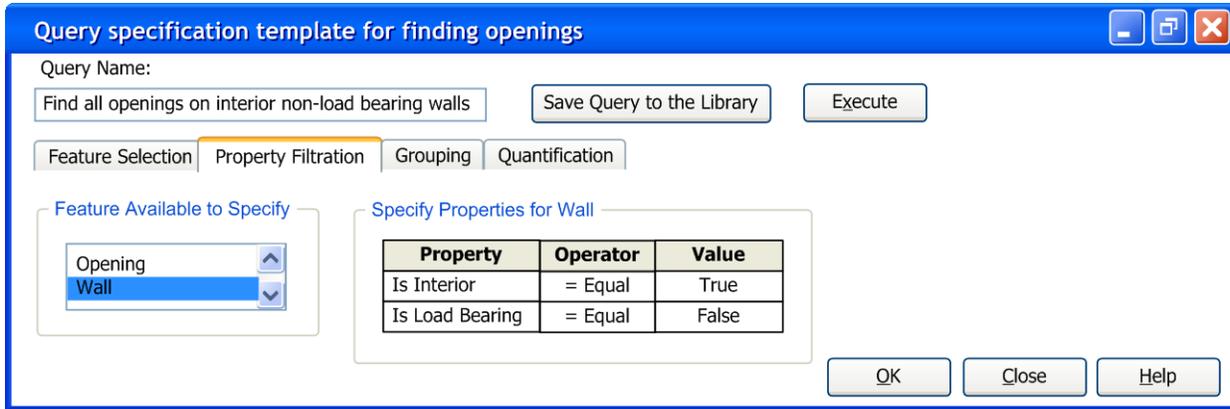


Figure 4.18 Step 2 (b) – Filtration - Specifying properties for the host feature “Wall”

**Step 3: Grouping** In the next step, the user specifies the grouping criteria, such as grouping by ‘opening type,’ and then by ‘size’ of the opening (Figure 4.19).

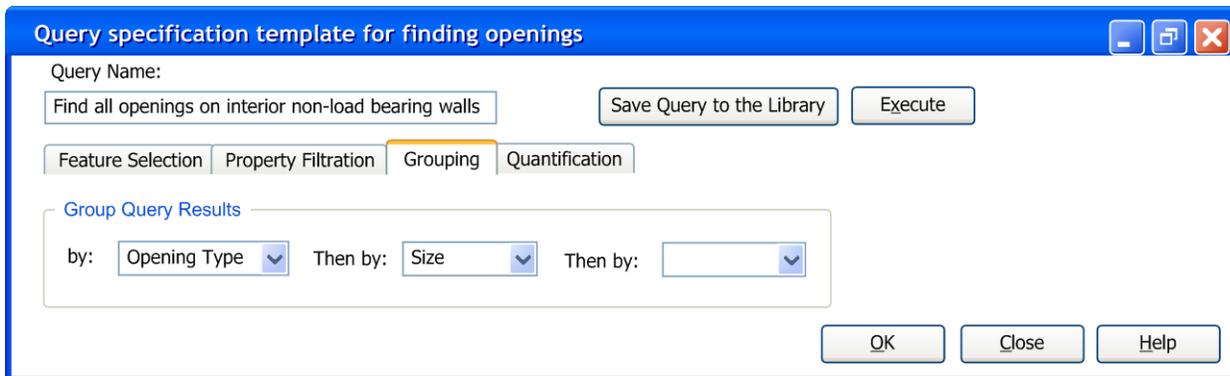
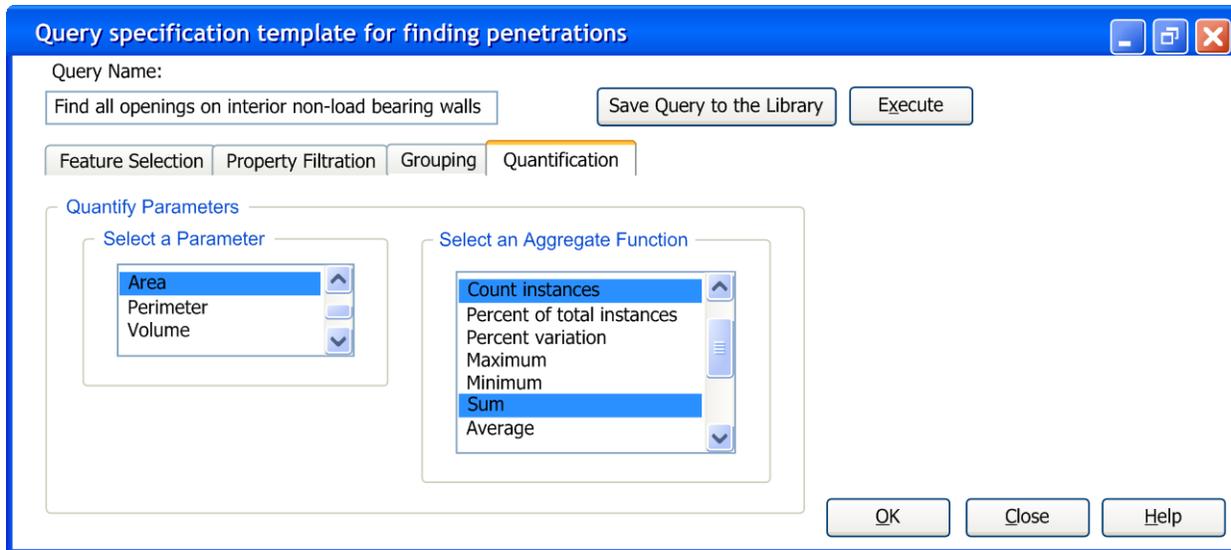


Figure 4.19 Step 3 – Grouping - Specifying grouping criteria for “Opening”

**Step 4: Aggregation** Finally, the user specifies the parameters for quantification as ‘area’ and aggregate functions as ‘count instances’ and ‘sum’ (see Figure 4.20) to identify the number of openings of each type and size and to calculate the corresponding total areas.

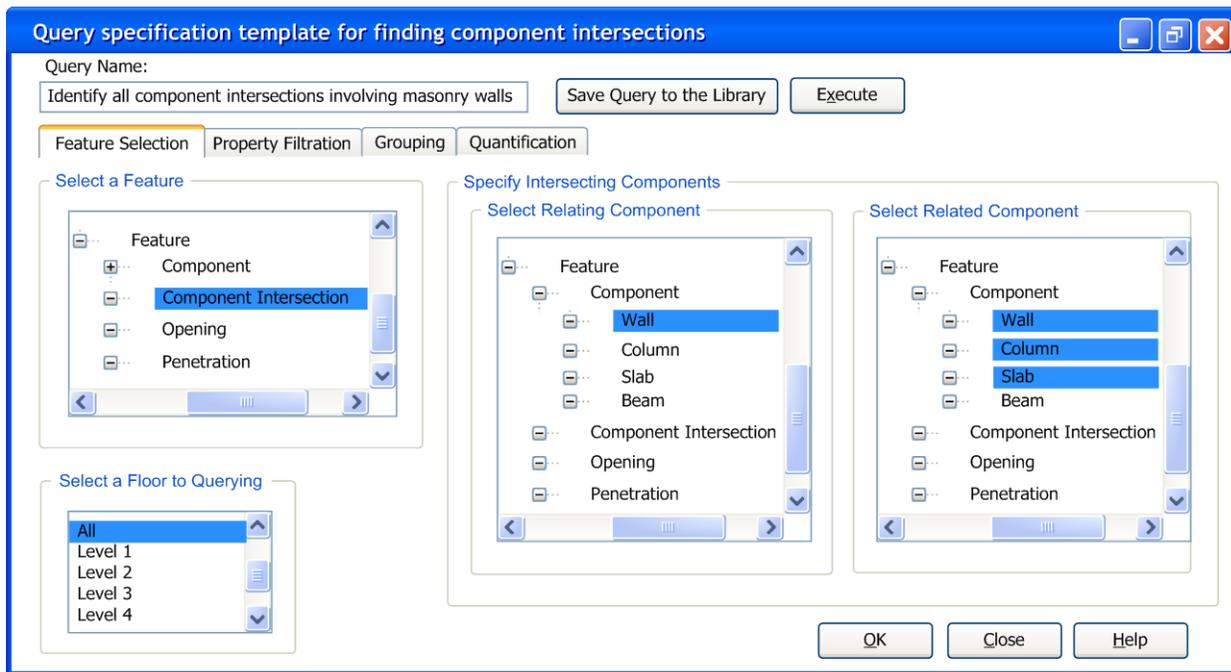


**Figure 4.20 Step 4 – Quantification - Specifying the parameter(s) and aggregate function(s) to be used in quantifying feature properties**

#### 4.6.1.3 Example # 3 Query on Component Intersection

Consider the query: “*Identify all component intersections involving masonry walls.*” This query incorporates the intersection of masonry walls with other types of walls (masonry to dry walls, masonry to concrete walls, etc.), and intersection of masonry walls with other component features, such as columns, slabs, etc. The intersections of masonry walls with other types of walls, impact productivity and may require additional work for framing, forming, layout, and detailing. Intersection of masonry walls to other structural components, such as beams, slabs and columns, may need to fulfill specific design and construction requirements, such as for connection, interface, fire separation, fire stopping, bond beam and lateral support. Query Example # 3 also highlights the use of different attributes in the ‘*feature selection*’ and ‘*property filtration*’ steps than the previous two query examples.

**Step 1: Feature Selection** In order to specify the above-mentioned intersection query, the user has to specify the type of intersecting components, i.e., ‘relating component’ and ‘related component’ attributes. In the example query, the user specifies the feature “wall” as the relating component and features “wall,” “column,” and “slab” as related components in the feature selection step in order to identify the instances of wall to wall, wall to column, and wall to slab intersections (Figure 4.21).



**Figure 4.21 Step 1 – Feature selection process to specify a component intersection query**

**Step 2: Property Filtration** The *property filtration* step allows the user to further specify the properties of the intersecting components, at a more detailed level. For instance, in the example query, consider that the user wishes to ascertain if there are any intersections of brick and concrete masonry walls with other components, which is specified by defining the properties of the relating component (wall, in this case), as shown in Figure 4.22.

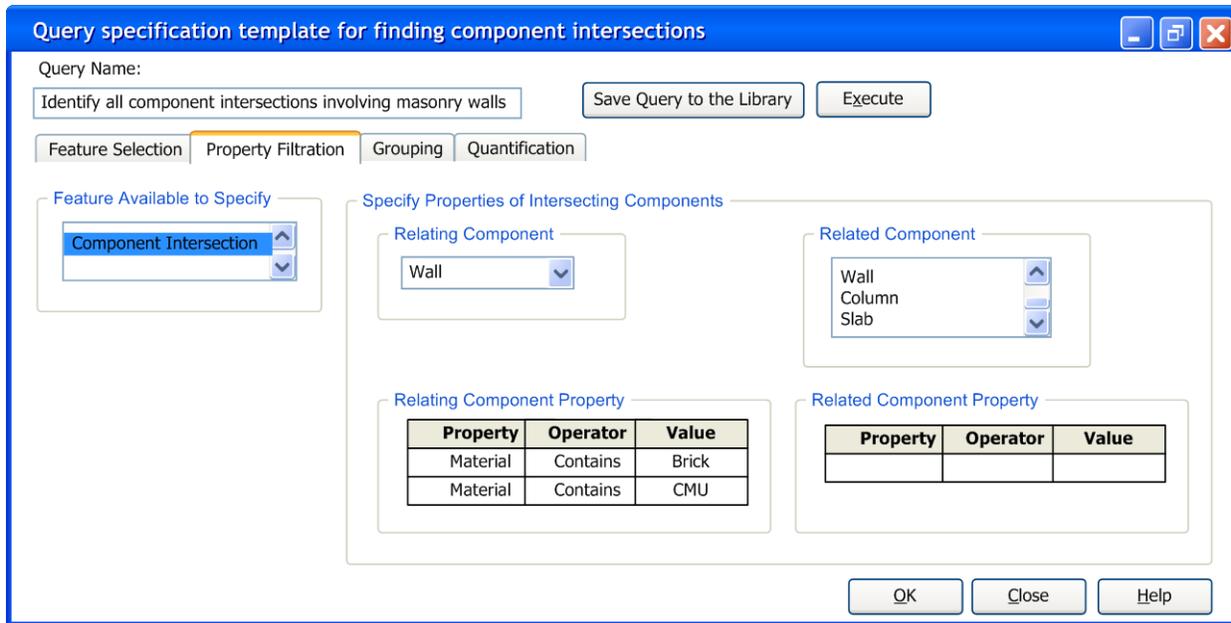


Figure 4.22 Step 2 – The process of property filtration in specifying a component intersection query

**Steps 3 and 4: Grouping and Aggregation** The ‘grouping’ and ‘aggregation’ steps are consistent with the previous examples previously discussed. These allow the user to group and quantify instances of component intersections.

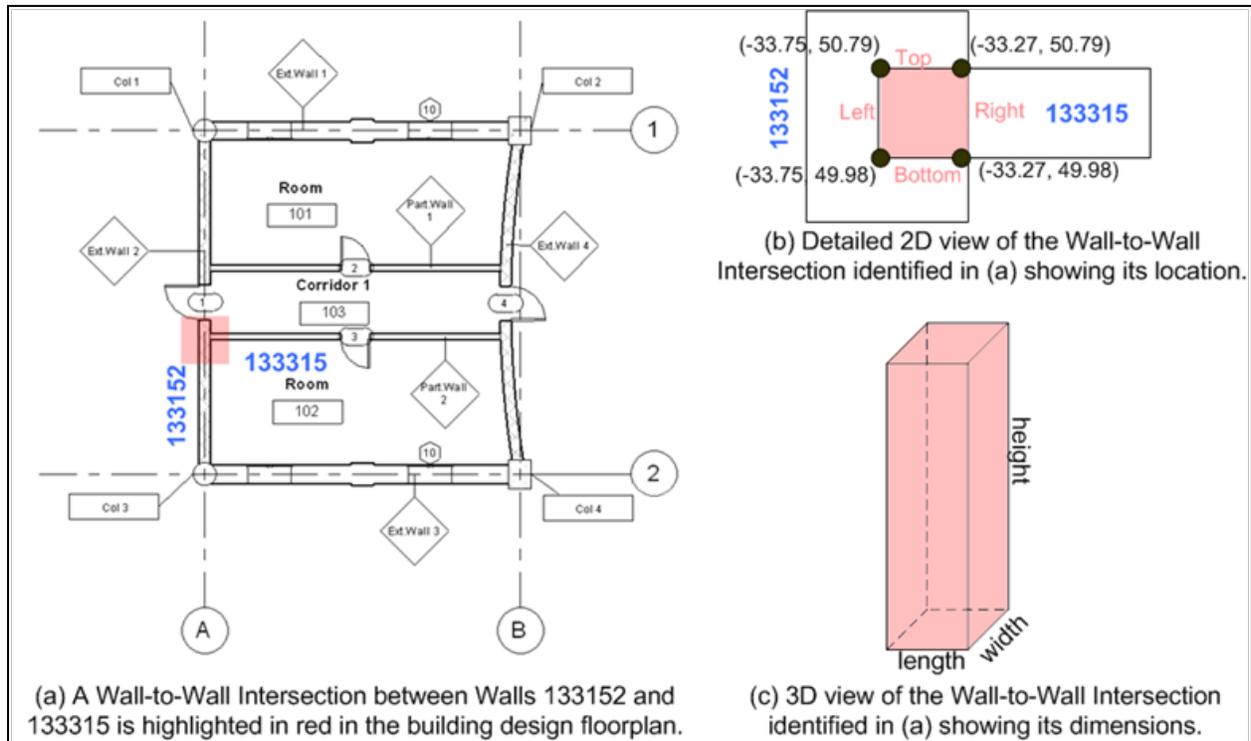
A sample output from the ‘component intersection’ query is provided below. Figure 4.23 shows the instance of wall to wall intersection graphically in 2D with some of the relevant details highlighted.

```
<Intersection>
  <Wall gml:id="133315"/>
  <Wall gml:id="133152"/>
  <location>
    <gml:Point srsName="gml:CartesianCS">
      <gml:pos>-33.75 50.79 0.00</gml:pos>
    </gml:Point>
    <gml:Point srsName="gml:CartesianCS">
      <gml:pos>-33.75 49.98 0.00</gml:pos>
    </gml:Point>
    <gml:Point srsName="gml:CartesianCS">
      <gml:pos>-33.27 50.79 0.00</gml:pos>
    </gml:Point>
    <gml:Point srsName="gml:CartesianCS">
```

```

        <gml:pos>-33.27 49.98 0.00</gml:pos>
    </gml:Point>
</location>
<area uom=#sq-ft>0.39</area>
<volume uom=#cu-ft>3.95</volume>
</Intersection>

```



**Figure 4.23** Example of a wall-to-wall intersection and the details provided by the “intersection query”

We have thus far described, with specific examples, how the user can formulate different types of basic queries using customizable and easy-to-use templates. Next, we describe the general characteristics of templates designed to specify extended queries, and we discuss how they leverage query attributes defined in Section 4.5.4.

#### 4.6.2 Query Templates for Formulating Extended Queries

Query templates for extended queries incorporate three query specification steps: (1) feature selection; (2) property filtration; and (3) parameter specification. The steps ‘*feature selection*’ and ‘*property filtration*’ have similar purposes as seen in the templates for specifying

basic queries, described earlier. The step, ‘*parameters specification*’ explicitly represents the underlying domain knowledge, or query parameters, of the related extended query. We use the following query examples to illustrate these steps.

#### 4.6.2.1 Example # 1 Query on Feature Location

Consider the query: “*Identify the location and size of duct penetrations on walls.*” Penetrations on walls may require special procedures, such as fire-stopping and weather resistance. Steps for the above query are briefly highlighted below.

**Step 1 and 2: Feature Selection and Property Filtration** In order to specify the above query, the user selects the feature “penetration” (Figure 4.24) in the first step, and next defines constraints to specify its properties, which include the specification of the host component, and penetrating element (Figure 4.25).

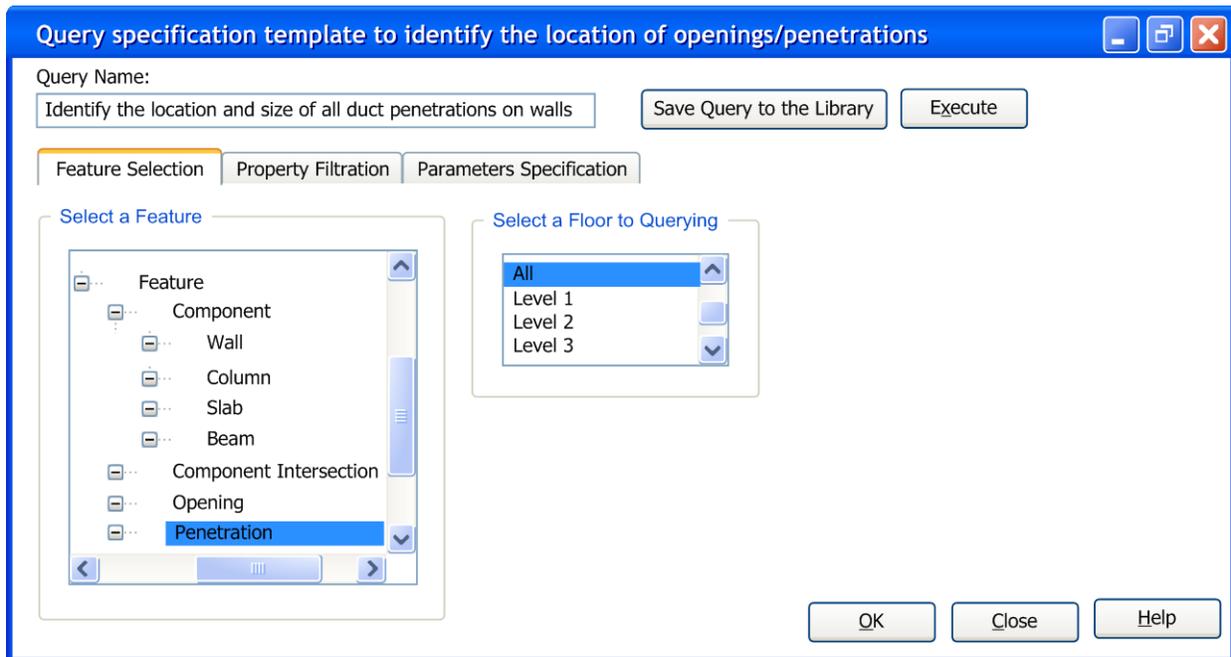
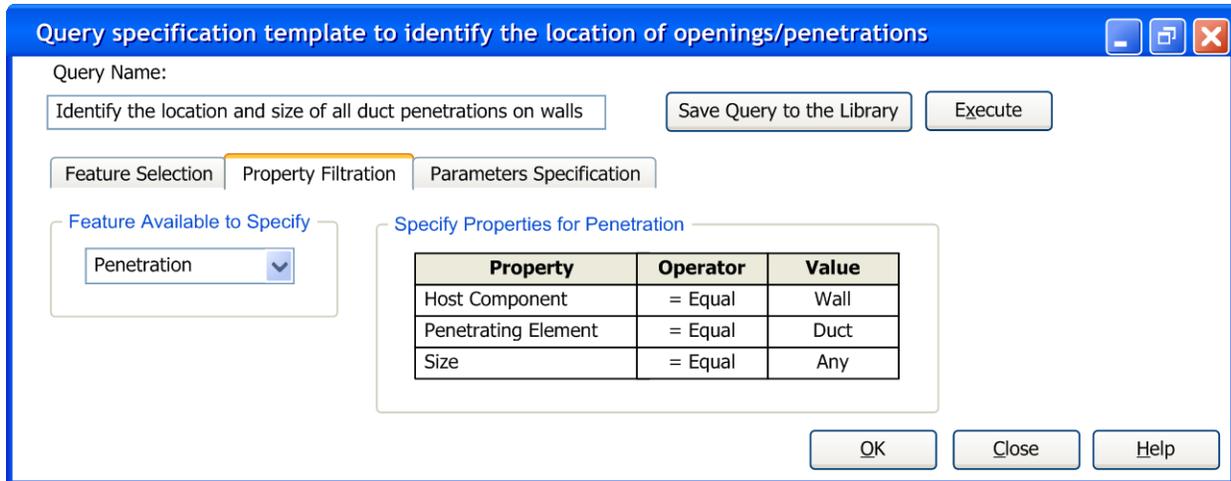
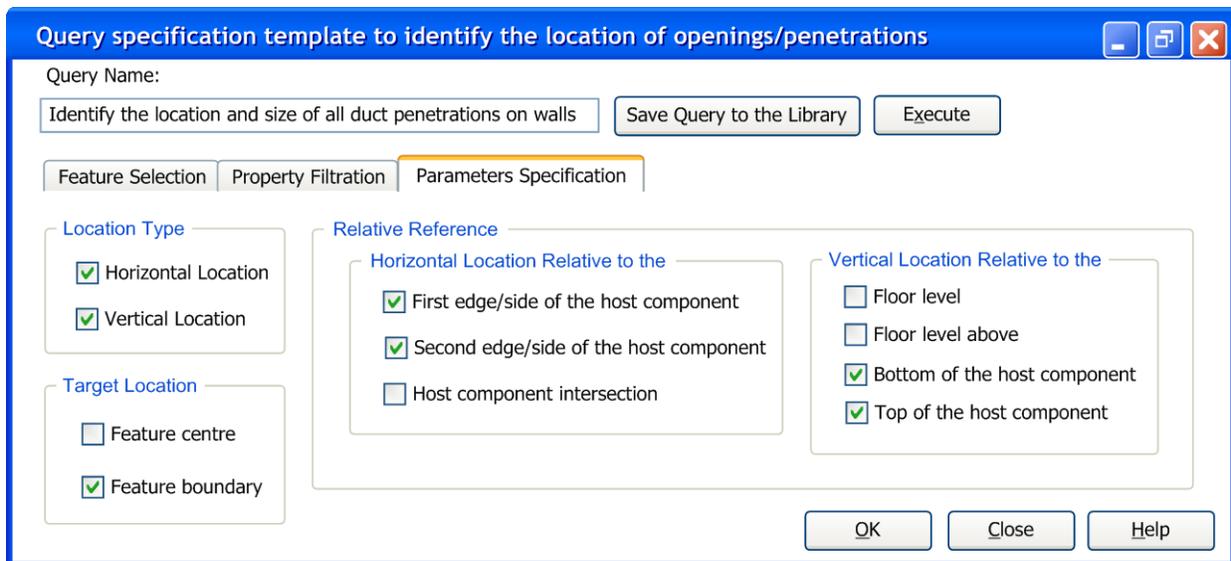


Figure 4.24 Step 1 – Feature selection for specifying the location of penetration



**Figure 4.25 Step 2 – Property filtration**

**Step 3: Parameters Specification** In Step 3, the user selects the ‘*parameter specification*’ tab to specify parameters for the “penetration” feature. The user specifies the attributes ‘location type’ as *horizontal* or *vertical* and ‘relative reference,’ and the corresponding references for horizontal or vertical location of a feature, as shown in Figure 4.26 (refer to Figure 4.10 for illustration on how practitioners commonly refer to the location of penetrations). The user also specifies the attribute, ‘target location,’ designating the location point of interest, as the *feature centre*, or *feature boundary*.



**Figure 4.26 Step 3 – Parameters specification process for specifying the location of penetration**

It should be noted that the type of parameters, or information to be incorporated under the ‘*parameters specification*’ step may vary from one feature to the next. For example, unlike penetrations or openings, which are referenced relative to the host component, the location of columns is normally referenced relative to grid lines.

A sample output from the ‘penetration query’ is provided below. Figure 4.27 shows the instance of the duct penetration on the wall and its location relative to the wall boundaries (or edges).

```
<Penetration>
  <Wall gml:id="133152"/>
  <Duct gml:id="149164"/>
  <area uom=#sq-ft>0.39</area>
  <volume uom=#cu-ft>3.95</volume>
  <location>
    <gml:Point srsName="gml:CartesianCS">
      <gml:pos>-34.24 42.04</gml:pos>
    </gml:Point>
    <gml:Point srsName="gml:CartesianCS">
      <gml:pos>-34.24 40.54</gml:pos>
    </gml:Point>
    <gml:Point srsName="gml:CartesianCS">
      <gml:pos>-33.27 42.04</gml:pos>
    </gml:Point>
    <gml:Point srsName="gml:CartesianCS">
      <gml:pos>-33.27 40.54</gml:pos>
    </gml:Point>
  </location>
  <locationOnWall>
    <distanceFromWallTop uom="#ft">2.25
      </distanceFromWallTop>
    <distanceFromWallBottom uom="#ft">6.25
      </distanceFromWallBottom>
    <distanceFromWallLeft uom="#ft">8.75
      </distanceFromWallLeft>
    <distanceFromWallRight uom="#ft">19.75
      </distanceFromWallRight>
  </locationOnWall>
</Penetration>
```

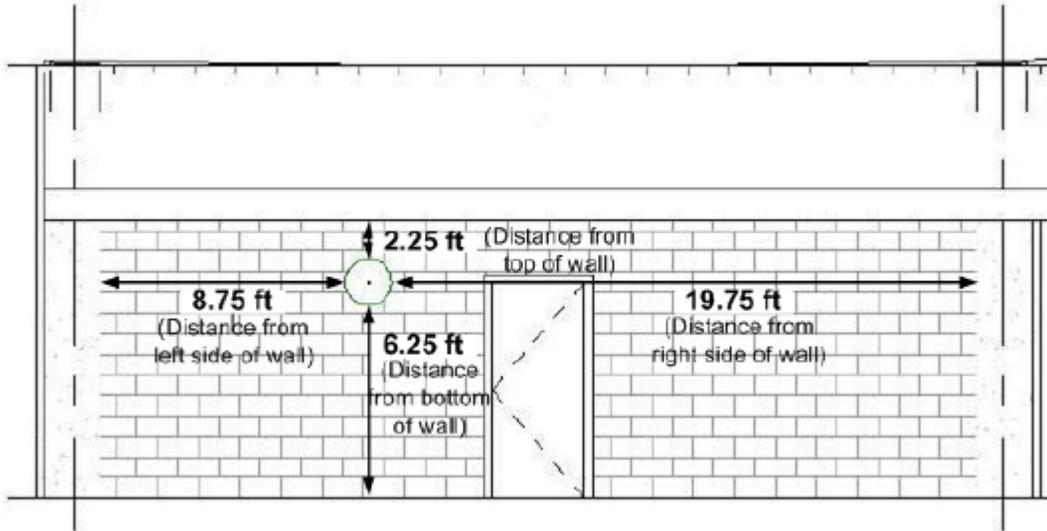


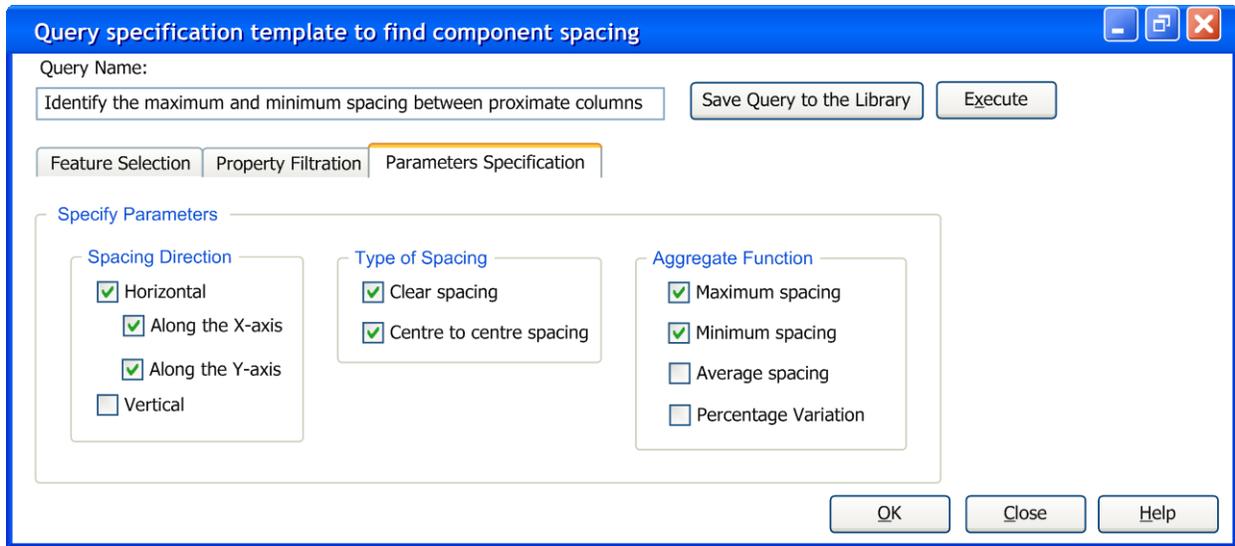
Figure 4.27 Example of duct penetration and its location in relation to the wall boundaries (sides) indicated

#### 4.6.2.2 Example # 2 Query on Feature Spacing

Consider the query: “*Identify the maximum and minimum spacing between proximate columns.*” The spacing query allows the user to identify the spacing between proximate features, such as columns, as well as other relevant information, such as maximum spacing, minimum spacing, average spacing, and percentage variation.

**Step 1 and 2: Feature Selection and Property Filtration** Similar to the two queries described above, the ‘*feature selection*’ and ‘*property filtration*’ steps allow the user to select a feature and specify the properties of a selected feature to define the spacing between the designated type of feature instances, such as the spacing of proximate columns.

**Step 3: Parameters Specification** In this step, the user specifies the spacing attributes and their value types. The visual encoding of relevant query-specific domain knowledge in the ‘*parameter specification*’ step in the query templates, allows the user to expressively and conveniently formulate extended queries. In regards to the example extended Query # 2, the user can specify a number of parameters that describe the attributes ‘type of spacing,’ ‘spacing direction,’ and desired ‘aggregate function’ (see Figure 4.28) to obtain the required and specific query results.



**Figure 4.28 Step 3 – Parameters specification process for specifying the column spacing query**

The 'spacing query' first identifies both the clear (or face-to-face) and centre-to-centre spacing between a column and its proximate columns on a level-by-level basis. Figure 4.29 shows the centre-to-centre spacing between all the proximate on-grid columns on the first level of a building plan. The 'spacing query' then identifies the minimum and maximum spacing between proximate columns for each level. Figure 4.29 shows an instance of the minimum centre-to-centre spacing between the two proximate columns (columns 80671 and 83412) as 16.4 ft.

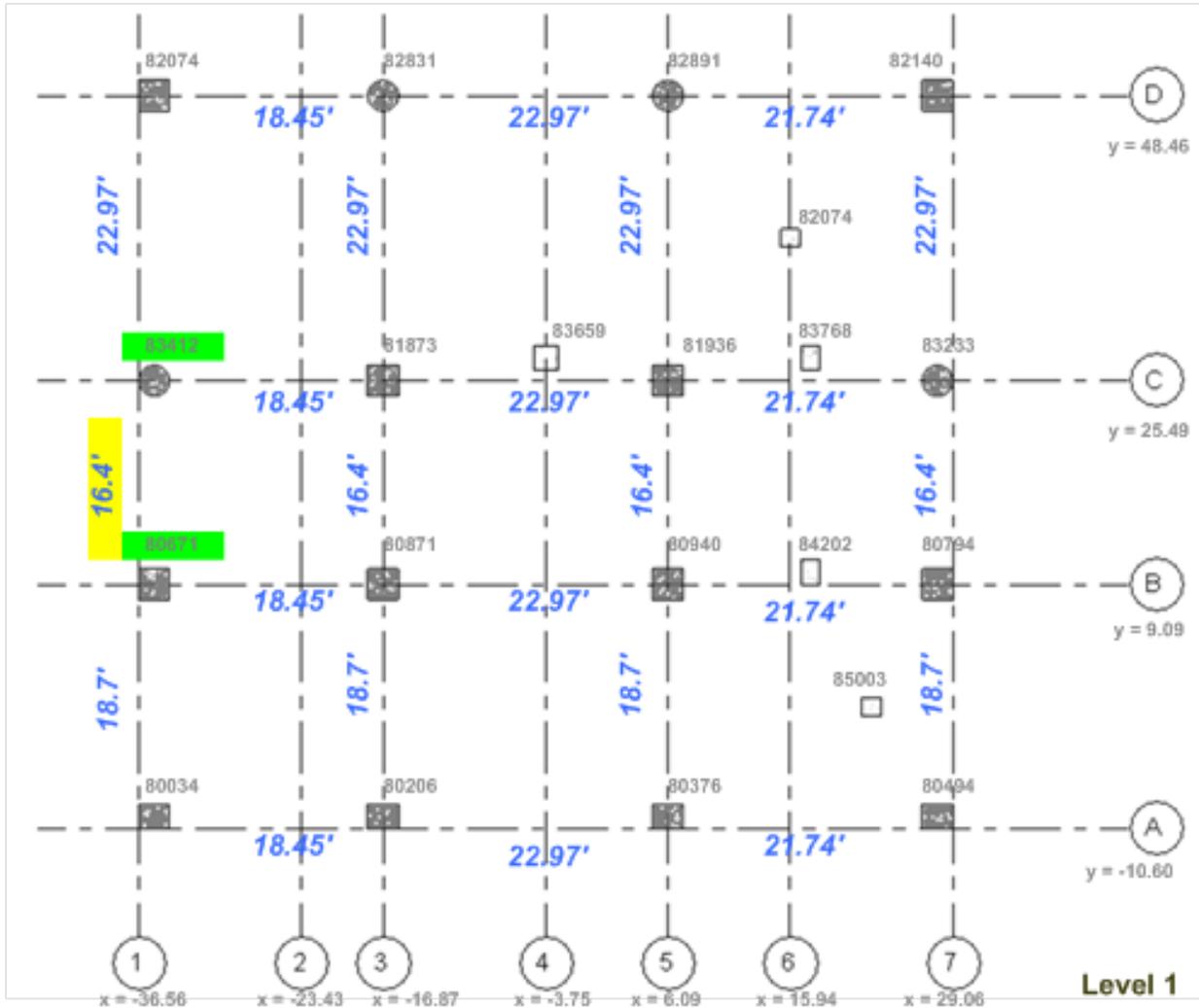


Figure 4.29 Spacing of proximate, on-grid columns in the first floor plan

The output from the ‘spacing query’ is provided below in the XML excerpt. The query returns, for each level of a building, the minimum and maximum clear spacing and centre-to-centre spacing.

```
<spacings>
  <Spacing>
    <level>1</level>
    <faceToFace>
      <minimum uom="#ft">11.48</minimum>
      <maximum uom="#ft">18.04</maximum>
    </faceToFace>
    <centerToCenter>
      <minimum uom="#ft">16.40</minimum>
    </centerToCenter>
  </Spacing>
</spacings>
```

```

                <maximum uom="#ft">22.97</maximum>
            </centerToCenter>
        </Spacing>
    <Spacing>
        <level>2</level>
        <faceToFace>
            <minimum uom="#ft">11.98</minimum>
            <maximum uom="#ft">37.73</maximum>
        </faceToFace>
        <centerToCenter>
            <minimum uom="#ft">16.40</minimum>
            <maximum uom="#ft">41.67</maximum>
        </centerToCenter>
    </Spacing>
</spacings>

```

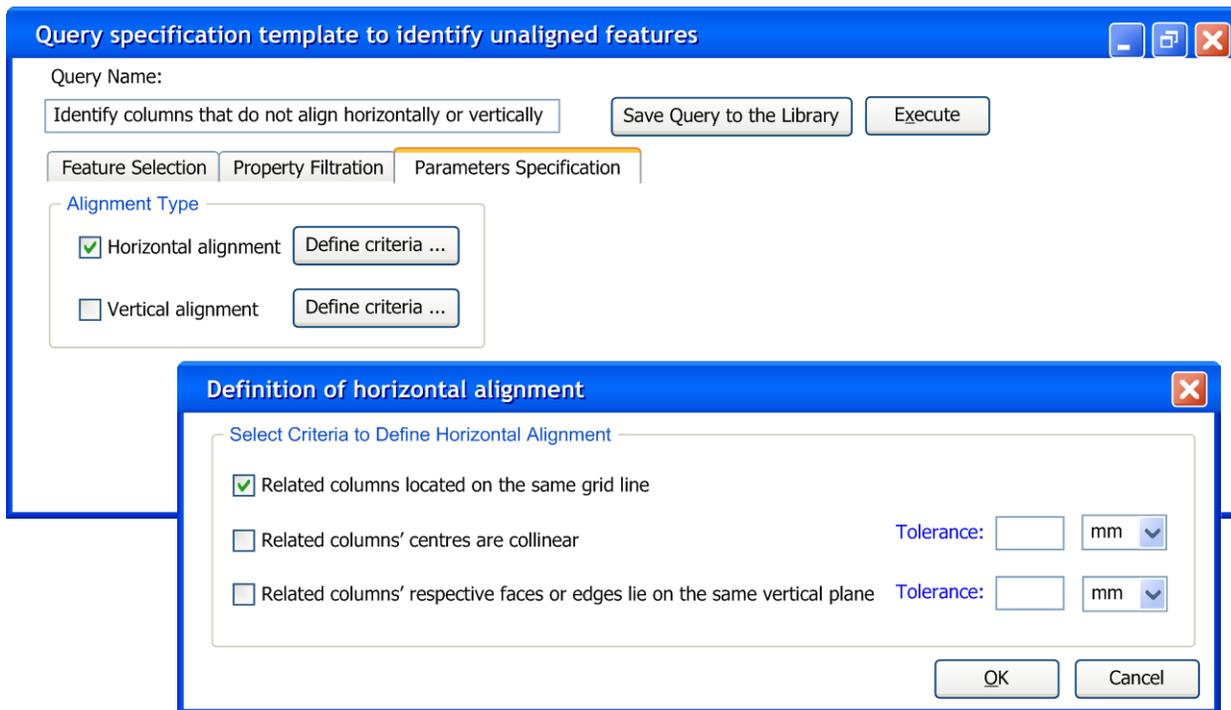
#### 4.6.2.3 Example # 3 Query on Feature Alignment

Consider the query: “*Identify columns that are not aligned horizontally or vertically.*” This query illustrates the requirement for encoding a set of query-specific domain rules, or criteria, in the templates to formulate certain types of extended queries, such as the feature alignment query in the ‘*parameter specification*’ step. The purpose of this query is to identify the unaligned columns horizontally along a floor, or vertically from floor to floor. The queries on column alignment seek to answer questions about specific aspects of layout, position, or orientation of columns in a floor (horizontal alignment) or from floor to floor (vertical alignment). This requires specific knowledge of practitioners and how they might define what it means for columns to be aligned. For instance, the following three rules, or criteria, are captured in the ‘*parameter specification*’ step (Figure 4.30), and the user can choose one or a combination of these criteria to define the horizontal alignment of columns.

*Criteria 1:* Related columns located on the same grid line

*Criteria 2:* Related columns’ centres are collinear

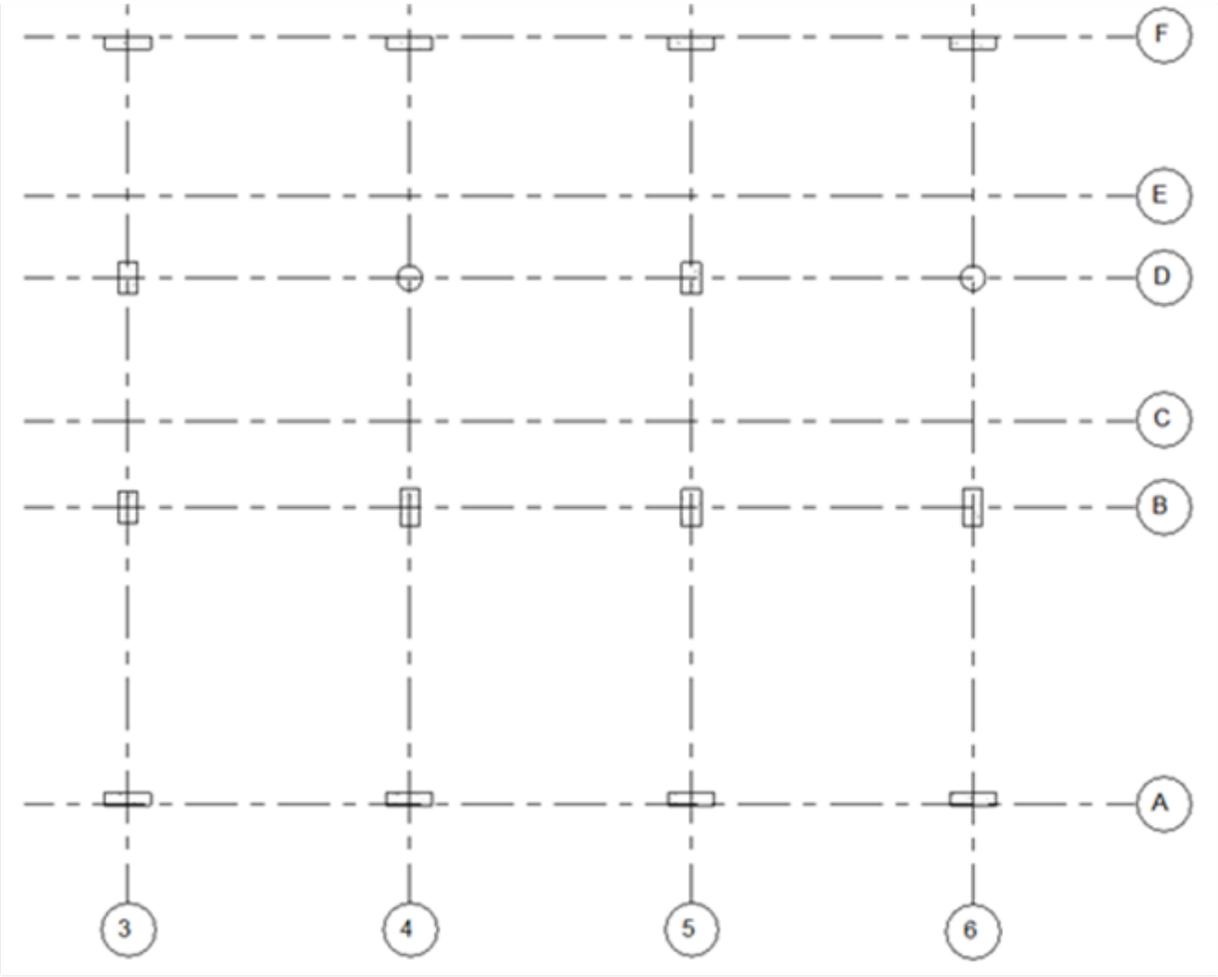
*Criteria 3:* Related columns’ respective faces, or edges, are equidistant from the relevant column reference line



**Figure 4.30 Parameters specification process for specifying the horizontal alignment of columns**

According to the first criterion, if the same grid line intersects a set of columns, then the columns are considered to be horizontally aligned along the direction of that grid line. From this definition, all columns A-3 to F-6, in Figure 4.31, are horizontally aligned along both the X - and the Y-axes, as they are all located on, or intersect with, the related rectangular grid lines.

The second criterion relates to the collinearity of the columns' centre. According to this definition, if the centre of the related columns are collinear, or lie on the same line, connecting their individual centre, then the columns are considered horizontally aligned along the axis of the grid line, to which such columns belong. In Figure 4.31, the centres of columns D-3 to D-6 and B-3 to B-6 are horizontally aligned along the X-axis because they are collinear along the grid lines D and B, respectively. Moreover, all columns are horizontally aligned along the Y-axis, i.e., along the grid lines 3 through 6.



**Figure 4.31** A portion of the column layout plan in a typical floor of the Chem-bio building project, UBC

The third criterion stipulates that, if related columns’ respective faces or edges are the same distance from the relevant column reference line, then these columns are horizontally aligned along the direction of that line. The reference line could be the related grid line. For example, in Figure 4.31, the columns B-3, B-4, B-5 and B-6 are *horizontally aligned* along the X-axis, or grid line B, because their corresponding faces along the X-axis are equidistant from grid line B. Similarly, all columns along grid line A, i.e., A-3 to A-6, exhibit the conditions for satisfying the third criterion, even though their centres are offset, relative to grid line A.

It should be noted that provisions can also be made to allow users to specify an appropriate tolerance value (e.g., +/- 25 mm) when checking for the alignment query. Rule sets can similarly be encoded in the query templates to check for the vertical alignment of columns.

Next, we provide a brief discussion of the reasoning process implemented for identifying the horizontal alignment of columns based on the first criteria discussed above (Webster 2010). We make use of the ‘on-grid’ query predicate to determine if columns are on- or off-grid, and consequently use that information to identify the horizontal alignment of columns. If a column is on-grid, the grid intersection to which it is aligned is returned. The ‘alignedToGridIntersection’ query predicate that provides this information is shown below. The output of this query predicate is the grid intersection that a column is horizontally aligned to in the format ‘x-gridline – y-gridline.’ Figure 4.32 shows exterior and interior on-grid columns that are horizontally aligned.

```
declare function artifact:alignedToGridIntersection($column)
{
    let $alignedToGridIntersection := (
        if (artifact:ongrid($column))
        then concat(artifact:alignedToGridline($column,"x"),
            "-",artifact:alignedToGridline($column,"y"))
        else "null"
    )
    return $alignedToGridIntersection
};
```

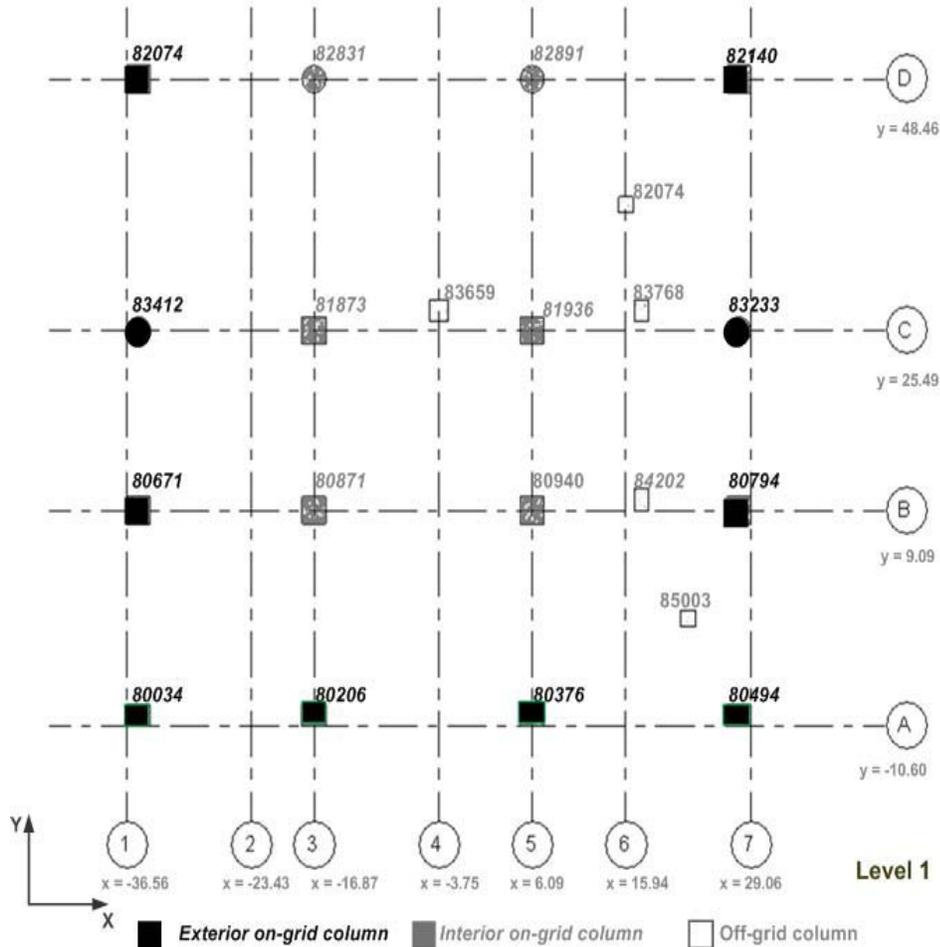


Figure 4.32 On-grid and off-grid columns on part of Level 1

If a column is not on-grid, it is described as off-grid, and the ‘alignment query’ provides additional related information, such as the closest x- and y-gridlines as well as the distance from them. In the first step, the ‘closestGridline’ query predicate, which is shown below, provides the closest x- or y-gridlines to an off-grid column.

```

declare function artifact:closestGridline($column,$axis)
{
  let $distToGridlines := (
    (: for all gridlines, find distance from column
    to the gridline :)
    for $gl in doc($gridsXMLFile)/grids/
      gridline[gridlineAxis=$axis]
    return
  )
}

```

```

        <closestGridline>
            <gridline>{$gl/name/text()}</gridline>
            <distance uom="#ft">
                {artifact:distanceToClosestGridline
                 ($column,$gl,$axis)}
            </distance>
        </closestGridline>
    ),
    $minDistance := min(($distToGridlines/dist)),
    $closestGridlines := (for $g in $distToGridlines
        where $g/dist = $minDistance
        return $g/gridline/text())
    return $closestGridlines
};

```

Next, the ‘distanceToClosestGridline’ query predicate, shown below, is used in the ‘closestGridline’ query predicate to help determine which gridline is closest to the off-grid column.

```

declare function artifact:distanceToClosestGridline($column,
    $gl, $axis)
{
    let $distance := round(fn:abs(round(number($column/
        centerpoint[@axis=$axis])*10000) div 10000
        - round(number($gl/gridlineValue)*10000) div
10000)*100)
        div 100
    return $distance
}

```

An example of the output of the ‘alignment query’ for an off-grid column 84202 in Figure 4.32 is presented below. This column is at the offset distance of 1.64 ft. from the grid line 6.

```

<column id="84202">
    <closestXGridline>
        <xGridline>6</xGridline>
        <distance >1.64</distance>
    </closestXGridline>
</column>

```

The above-mentioned ‘extended’ query examples and corresponding templates describe common query steps and the types of attributes and knowledge required to specify these more complex queries. While the query steps and attributes formalized can also be applied to other types of extended queries, the parameters and knowledge required to represent the ‘*parameter specification*’ step, however, can be query specific. The explicit representation of query-specific knowledge, and the use of query templates, provides intuitive, structured, and guided help to users when formulating queries.

#### **4.7 Summary and Conclusions**

In recent years, there has been a growing interest from the design and construction community to adopt BIM. The emergence of BIM has created new challenges as well as new hope for construction practitioners. While the richness of design information offered by BIM is evident, there are still tremendous challenges in getting construction-specific information out of BIM, limiting the usability of these models for construction and other downstream processes. Construction practitioners today have an increasing need for quick and easy access to information in a BIM, delivered in a way that meets their expectations. In an effort to address these deficiencies, this thesis has formalized the necessary representation and reasoning mechanisms to support the automatic extraction and querying of construction-specific design information in a given BIM.

In this chapter, we described query specifications that provide a query vocabulary, or language, for specifying spatial and non-spatial queries on design features. We developed reusable, customizable and computer interpretable query specification templates, and described generic query formulation steps for different types of queries. Query templates provide a visually intuitive environment to easily and quickly formulate queries by non-expert BIM users, familiar with the domain concepts, but not possessing sufficient knowledge of the *IFC* model, underlying data models of a BIM, or query languages. The queries and underlying concepts formalized in this research were captured from an extensive review of the academic literature, case studies, interviews and discussions with working practitioners.

Our approach helps to extract information that is normally hidden or implicitly represented in a BIM. It allows construction practitioners to query for specific types of component features, and filter, group and aggregate information that meet their specific

requirements. It also provides a comprehensive support system to formulate queries for identifying component intersections, which are not always explicit in BIM models. It enables the user to specify queries, to find the specific type of intersections (e.g., intersections between concrete block wall and slab, and intersections between dry wall and round columns), based on specific properties or characteristics of intersecting components. Our approach provides support to help practitioners specify queries for identifying component penetrations and openings and offers other insightful information about these features. It allows the user flexibility to define the host component, where the penetrations (or openings) have to be identified, the type of penetrating elements (e.g., duct, pipe, conduit, etc.) and to specify relevant properties of interest (e.g., size, area, perimeter, etc.).

We provide support for extended queries, which analyze BIMs for higher-level spatial concepts or relationships among features, which include the concepts of ‘location,’ ‘spacing,’ ‘alignment,’ and ‘design uniformity’ of features. We identified commonly applicable attributes required to specify these extended queries, and described the steps required for their formulation. This knowledge has been formalized in a way that captures the nuance for how practitioners think about the design, and that provides sufficient flexibility for practitioners to express this knowledge in a variety of ways. For example, consider the formalization of alignment knowledge. Practitioners can express queries on column alignment in terms of grid-lines, the collinearity of column centres, or by using the column faces or edges. Similarly, practitioners have significant flexibility in expressing uniformity queries, in terms of feature properties (e.g., walls types, size, height, fire-rating), in terms of quantifying the degree of uniformity (e.g., based on counts or a percentage), and in terms of specifying the degree of permissible variation (e.g., 20% variation in opening size and location).

The descriptive and interpretative analysis, described in detail in Section 5.2.3 of Chapter 5, demonstrates that our approach is simple, general, expressive, and flexible. We provide evidence for the relevance and importance of the concepts formalized in our research based on interviews with numerous construction professionals. We provide evidence for the usefulness of our approach in generating actionable and insightful information relevant to construction practitioners. We also provide evidence that our approach generates a richer representation of construction-specific design information when compared to existing tools. We provide evidence for the generality of our approach by demonstrating that it supports multiple construction

management functions (e.g., method selection and cost estimating), multiple perspectives (e.g., project managers and site superintendents use different terminology), and multiple component types (e.g., different features and properties impact column and wall construction).

This research takes an essential step in formalizing the theoretical framework needed to develop software tools capable of extracting required information from a given BIM in a way that is customized for and by construction practitioners. Our research has the potential to facilitate improved understanding of a given design and support decision making in a broad range of construction management functions, enabling practitioners to better plan, coordinate, estimate, and execute their work.

## Chapter 5: Conclusions

In this dissertation, we developed and implemented a framework for the automated extraction and querying of construction-specific design information from a Building Information Model (BIM). To employ this framework, we formalized: (a) a feature ontology to represent design features relevant to construction; (b) query specifications and templates to specify spatial and non-spatial queries on features; and (c) an integrated approach that automatically creates a construction-specific feature-based model (FBM) for a particular BIM, and answers user-driven and customizable queries on the FBM.

We interviewed construction experts and studied several building projects to determine the relevance and representativeness of the concepts formalized. This was followed by an evaluation of the extent to which the developed research tools could identify the kinds of design conditions that are relevant to construction practitioners. Finally, we provided a critical evaluation of the intellectual merits of the research for generating practical, insightful and rich construction-specific information. The combination of validation techniques employed in this research demonstrates the content, representativeness, generalizability, soundness, and intellectual merits of the developed research. We believe that a richer representation of construction-specific design conditions, coupled with the extraction of desired knowledge from a BIM, can make a BIM more accessible and effective for construction use.

The remainder of this chapter is organized as follows: Section 5.1 describes the main contributions of this research; Section 5.2 explains the validation tests performed; Section 5.3 briefly outlines practical implications of the research; and Section 5.4 describes some limitations of the current research, and directions for future research.

### 5.1 Research Contributions

In this research, we developed and implemented a framework to extract and query construction-relevant design information from a BIM. The key contributions are:

1. The formalization of a feature ontology that explicitly represents design conditions that are relevant to construction practitioners and supports the generation of a construction-specific feature-based model;

2. The formalization of a query specification vocabulary which characterizes spatial and non-spatial queries, and the development of query templates to guide non-expert BIM users to specify queries, and;
3. The formalization of an integrated approach that combines model-based reasoning, feature extraction, and query processing to automatically extract design features to create a project-specific feature-based model (FBM) and provide support for answering queries on the FBM.

**Contribution 1: The formalization of a feature ontology that explicitly represents design conditions that are relevant to construction practitioners and supports the generation of a construction-specific feature-based model.**

BIMs provide rich information about a building and its components. However, they do not explicitly represent many design conditions that are of importance to construction practitioners. Moreover, construction practitioners today have limited computer support to query a BIM for relevant design conditions. In order to address this need, we developed a feature ontology that provides a language to characterize the design conditions that are important to construction in a generic and computer-interpretable way. Features in the context of this research are used to explicitly describe different design conditions that practitioners would need to identify in a given BIM. The developed ontology provides a consistent, less ambiguous, and computer-interpretable vocabulary of concepts, in terms of features and feature attributes (properties and relationships), representing an array of design conditions.

Many prior studies have identified important design conditions in the domain of cost estimating (Bisharat 2004; Staub-French et al. 2003); constructability (Burkhart et al. 1987; Boeke 1990; Skibniewsk et al. 1997); productivity analysis (Thomas and Zavrski 1999; Sanders and Thomas 1991); methods selection (Hanna et al. 1992; Fisher and Tatum 1997) and design and construction knowledge integration (Allen and Iano 2002; Nguyen and Oloufa 2002; Haymaker et al. 2004). This research builds on prior research that has identified the importance of different design conditions to construction. In particular, it builds on the ontology developed by Staub-French et al. (2003). It also augments previous research in the area of building product data modeling. The augmented model allows for access of product data information in the *IFC* model, and third party BIM tools, through a domain-specific model, or ontology.

We formalized the feature ontology as an object hierarchy consisting of feature classes (or features), their properties, and the relationships among them. The feature ontology is represented generically so that it can be reused from project to project. The ontology represents building components as *component* features and intersections of building components as *intersection* features. The intersection feature is further classified into *component intersection*, *opening*, and *penetration* features. The object-based hierarchical structure of the ontology enables domain-specific knowledge to be extended or modified. The ontology acts as a knowledge repository that enables the automatic generation of a project-specific FBM. This FBM explicitly represents the features relevant to particular construction practitioners or domains, and provides the foundation upon which to query construction features from a BIM.

**Contribution 2: The formalization of a query specification vocabulary which characterizes spatial and non-spatial queries, and the development of query templates to guide non-expert BIM users to specify queries.**

Construction practitioners have specific requirements for design-related construction information. The previous research has captured an extensive body of knowledge of different design conditions relevant to construction or that impact various construction management functions. It, however, does not provide a vocabulary of different types of queries and attributes which characterize them. Existing approaches for identifying design conditions are inadequate, as they are narrowly focused, inflexible and difficult for seasoned construction practitioners, or novice BIM users. Emerging BIM tools do not provide sufficient support to easily and flexibly specify queries to derive production relevant information to support the requirements of construction practitioners. In order to address this need, we formalized query specifications and developed query templates to specify user-driven and customizable queries.

Query specifications provide a formal and structured query vocabulary, or language, for specifying spatial and non-spatial queries on design features, formalized in the feature ontology and instantiated in the FBM. The vocabulary defines different types of queries and formalizes relevant attributes and domain knowledge necessary to specify queries. The queries and underlying domain knowledge formalized in this research were captured from an extensive review of the academic literature, case studies, interviews and discussions with working practitioners.

We developed reusable, customizable and computer interpretable query specification templates, and formalized generic query formulation steps for different types of queries. Query templates provide a visually intuitive environment to easily and quickly formulate queries by non-expert BIM users, familiar with the domain concepts, but not possessing sufficient knowledge of the *IFC* model, underlying data models of a BIM, or query languages (e.g., *XQuery* or *SQL*). For each type of query, a template or a collection of templates, allows the users to select a feature, filter feature properties, and specify attributes, and other query related parameters, to precisely define a query that meet their construction requirements.

**Contribution 3: The formalization of an integrated approach that combines model-based reasoning, feature extraction, and query processing to automatically extract design features to create a project-specific feature-based model (FBM) and provide support for answering queries on the FBM.**

For BIM analysis tools to be useful for construction, they must provide data as well as information in a way that is meaningful to practitioners. They must also provide flexibility in terms of retrieving necessary information, based on the needs and preferences of practitioners. To this end, this research formalized an integrated approach for the automated extraction of design features and querying of a BIM.

Our integrated approach combines model-based reasoning and query-based approach. The model-based reasoning leverages a prototype application, the *Feature Extractor*, which automatically extracts the features in a given BIM based on the feature ontology to create the Feature-Based Model (FBM). The FBM is a project-specific model which includes instantiated features, feature properties, and the relationships among them. The FBM interface allows the user to interactively browse and navigate a FBM in order to gain insights into features that exist in a particular building project. The query-based approach leverages a prototype application, the *Feature Query Analyzer*, which provides further support for answering user-specified spatial and non-spatial queries on the FBM. The *Feature Query Analyzer* takes the user-defined query specifications as input to automatically process queries on features extracted from a given BIM.

Building information extraction, browsing, navigation, and the visualization of the extracted information, are all active research areas (Katranuschkov et al. 2003; Tolman et al. 2001; Woestenenk et al. 2000). Many *IFC* complaints software tools, such as *IFC* viewers (e.g.,

DDCIfcViewer, and IfcViewer) and *IFC* file browsers, facilitate 2D and 3D visualization of *IFC* data, and browsing through *IFC* files (IAI 2010). Other third party applications, such as *Solibri Model Checker*, extend the capabilities of *IFC* viewers and *IFC* browsers to extract and browse other types of information, such as location and quantities of different components. Other researchers have also added task-specific reasoning dimensions to product models in order to derive different types of information, such as identifying objects and their spatial and topological relationships (Stiny 1985; Cherneff et al. 1992; Nguyen and Oloufa 2002; Chen et al. 2005, Fischer 1991; Haymaker et al. 2004). Borrmann and Rank (2009) developed *3D Spatial Query Language* to support the spatial analysis of a BIM.

This research builds upon and extends previous research and state-of-the-art tools, to automatically identify information that matters most to a broad range of construction practitioners. Our approach not only automatically extracts features of a design from a given BIM, but also provides flexibility to specify queries in order to identify rich, insightful and useful information about the extracted features. Our approach is also reasonably generic and scalable to identify a wide-range of construction-specific design information, both spatial and non-spatial, across different building components. It has practical significance and implications in terms of leveraging BIM for construction purposes. It is expected that our research contributes towards improving the efficiency, consistency and effectiveness of the decision making process in construction.

## 5.2 Validation Studies

Given that construction works are project-based and interdisciplinary, construction research validation in real-life settings face inherently difficult challenges (Lucko and Rojas 2010). Fischer (2006) accurately argues that researchers in project-based industries, such as construction, often need to validate their research by triangulating different methods, such as field observations, researching theory in related literature, reviewing predictions, and investigating insights from domain experts, expert opinion, and case studies.

In this research, we use a combination of non-statistical approaches to validate our research findings. The approaches used are collectively known as *external*, *content*, and *face validity* approaches, as defined in the scientific community.

- *External validity* relates to the generalizability of results. It measures the extent to which the research results or conclusions can be applied or generalized to situations beyond the study itself (Leedy and Ormrod 2005).
- *Face validity* (sometimes referred to as *representational validity*) determines whether a study represents its intended measure, and if it represents reality (Leedy and Ormrod 2005; Lucko and Rojas 2010).
- *Content validity* is another non-statistical approach that focuses on determining whether the content of a research study makes sense in its context (domain) and is representative of reality.

*Face validity* and *content validity* seem to measure similar aspects, but the former is coined more so for the process by which it is conducted. For example, Leedy and Ormrod (2005) define *face validity* as the subjective judgment of a non-statistical nature that seeks the opinion of non-researchers, or non-domain experts, regarding the validity of a particular study. They argue that having representative samples, or data, and using triangulation, such as case studies, in-field observation, and interviews with domain experts, while collecting data and designing the system increases the content validity of a study.

We performed the following tests to collect evidence for the validity of our research. These tests demonstrate the content, representativeness, soundness, intellectual merits, and generality of the developed research.

1. ***Detailed Interviews:*** We interviewed construction experts and studied a variety of projects to validate the *content* and *representativeness* of the concepts formalized. Conducting personal interviews with domain experts is the most valuable and effective way to acquire domain vocabulary, or knowledge, and receive immediate data and feedback to verify the implemented system (Carrico et al. 1989). The evidence gathered from interviews, also shows the *generality* of the knowledge formalized in this research. This is based on the fact that four different experts, representing different construction domains and viewpoints, provided their input on the design conditions that are relevant to them, the criteria for why they matter, and/or the conditions under which they would matter.

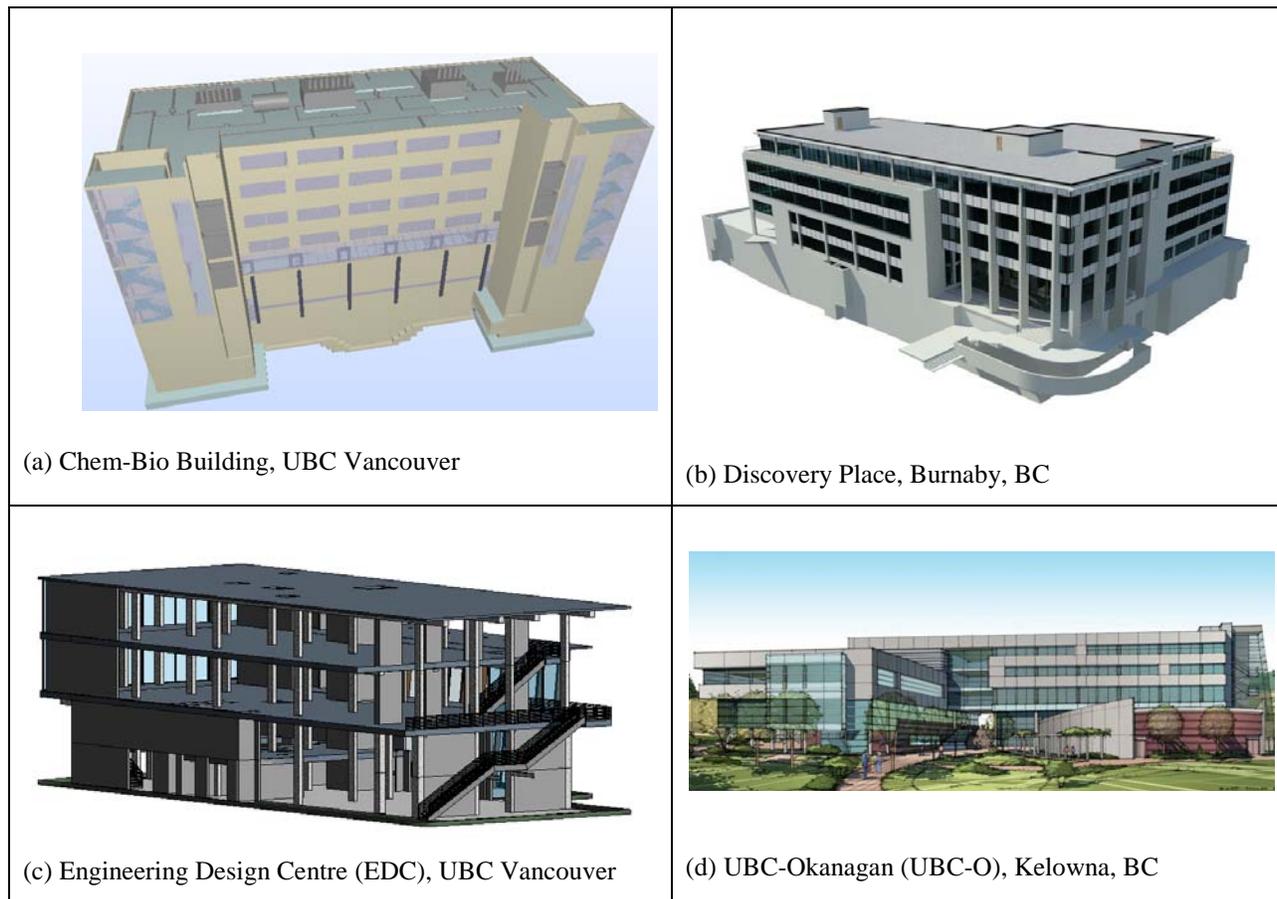
2. ***Retrospective Analysis:*** We conducted retrospective analysis of several design conditions to demonstrate the *soundness* of our approach. Specifically, we analyzed the level of support provided by our approach, in relation to the state-of-the-art tools, for identifying the design conditions that construction practitioners are concerned with. We designed retrospective analysis of design conditions related to a “component” in general, and “wall” and column” components in particular, and “component intersection,” “opening,” and “penetration” features. This test also demonstrates the *generality* of our approach as our research enables the user to extract, or query different design conditions for different components, intersections, openings, and penetration features.
3. ***Descriptive and Interpretative Analysis:*** We performed descriptive and interpretative analysis to evaluate the *intellectual merits* of our research for generating *useful*, *actionable* and *insightful* information. The sections that follow describe each of these tests in detail.

The projects studied and used for validation in this research are shown in Figure 5.1. They are concrete structures. All these projects are institutional buildings, with the exception of the Discovery Place project. They are briefly described below.

- a) The Chemical and Biological Engineering Building (referred to as the Chem-Bio project) was constructed at UBC’s Vancouver campus. It is a 123,000 square foot building that includes two integrated components, a replacement facility for the Chemistry and Biological Engineering Department, and a new facility for the Clean Energy Research Centre. We focused on the replacement facility in our case study.
- b) The Discovery Green Building (referred to as Discovery Place project), located at Discovery Place Research Park, Burnaby, BC, is a LEED Platinum certificated state-of-the-art office complex with a floor area of 153,000 square feet. It features sustainable design principles in a modern facility.
- c) The Wayne & William White Engineering Design Centre (referred to as the EDC project) is located at UBC’s Vancouver Campus and has a total floor area of 20,232 sq. ft. It provides spaces for direct instruction, extracurricular activities, learning skills support,

and student study spaces. One of the main focuses of this project is to enhance delivery of engineering design courses and student's engineering design activities and collaborations.

- d) The Fipke Centre for Innovative Research Project (referred to as the UBC-O project) with a total floor area of 73,466 sq. ft, is located in Kelowna, B.C. It incorporates generic research and teaching space, ensuring flexibility for the changes in floor spaces between those two activities. The Fipke Centre has earned an unprecedented five Green Globe awards in recognition of sustainable design initiatives.



**Figure 5.1** 3D models of the case study projects used in this research

The following sections provide more detailed discussions of our validation tests.

### 5.2.1 Interviews with Construction Experts

We conducted interviews with four construction experts to assess whether the concepts formalized in our research were relevant to construction practitioners. The experts were identified wholly at arm's length, by the author of this thesis, and by the author's supervisor, via networking with personal contacts. The interviews helped to verify the credibility and confirm the knowledge formalized in the feature ontology and query specifications. It also provided the opportunity to better understand the experts' rationale for why and how design conditions impact construction. Another purpose of conducting the interviews was to further understand the important additional concepts that carry relevance for construction experts, and the domains that may be worth considering within our framework for further improvement and extension. The interviewed construction experts included a Project Manager, a Formwork Manager, a Site Superintendent, and a Chief Estimator. The following summarizes their particular expertise:

- ***Project Manager:*** The project manager is working with one of the world's leading planning, engineering, and construction management organizations. He holds professional degrees in structural, architectural, and construction management, with more than thirty years of experience in planning, scheduling, and estimating. For more than twenty years, he has worked in the capacity of project manager, and construction manager, for a range of infrastructures and facilities. We used examples of design conditions and scenarios from the Chem-Bio project at UBC (Figure 5.1a), as reference to steer the interview with this expert. He played the role of the generalist, surveying the design conditions from the perspectives of component layout, component installation, constructability, cost estimating, construction planning, and scheduling.
- ***Formwork Manager:*** The formwork manager works for a company that specializes in the construction and erection of concrete formwork, serving the commercial, high-rise and high-end residential homes industries. He has over 20 years of construction experience. He has significant experience in estimating and managing concrete formwork. The author of this thesis used several examples and scenarios from the Chem-Bio Project to lead the discussion with him. He often used examples of the design conditions from the Discovery Place project (Figure 5.1b) in the interview.

- **Site Superintendent:** The site superintendent works with a medium-sized construction company, specializing in multi-residential, commercial, institutional and mixed-use projects. He has over 12 years of experience in construction site supervision and operation. He displays an outstanding and proactive ability to decipher design issues that impact construction operation and then provide constructability feedback to his clients. He specifically used examples from the EDC project (Figure 5.1c) in our discussions.
- **Chief Estimator:** The Chief Estimator is working for a general contractor that provides construction management services to clients. He has extensive experience in quantity surveying and project controls. He provided examples of relevant design conditions from a cost-estimating perspective, with particular reference to the UBC-O project (Figure 5.1d).

We asked these experts about the impact, or relevance, of different design conditions related to specific building components (walls and columns, in particular), component intersections, penetrations, and openings. We used sets of close-ended questions to interview the Project Manager and asked him to indicate the relevance of each design condition (or factor). We also sought open-ended explanations about the rationale for each factor, or any other factors not incorporated in the questions. We conducted face-to-face interviews, and directed open-ended questions to the three other experts, to understand the relevance of different design conditions and gathered detailed information about the specific design conditions that were present, or of particular concern, in the referenced projects. We used visual aids, probing questions, example scenarios, and structured sets of questions to guide the interviews, and to reduce any potential misunderstanding in terms of our questioning. We recorded all interviews and later analyzed the transcripts of these interviews.

The results of the interviews with all four experts are presented in Table 5.1 through Table 5.7. The tables show the degree of relevance of different design conditions related to the features formalized in our research. It should be noted that, for certain design conditions, the same expert identified multiple levels of relevance (e.g., can be significantly or moderately relevant). This suggests that the influence of design conditions can vary, and is likely to depend on the project context and the interactions with other design conditions. The tables also indicate

the comments, if any, explaining the rationale as to how and why a particular design condition is relevant to construction. The following sections highlight the results of the interviews.

#### **5.2.1.1 Expert Opinion on the Feature “Wall”**

Table 5.1 presents the experts’ response on different design conditions related to the feature “wall.” Practitioners considered many design conditions related to walls as very or moderately important. ‘Wall type’ is typically relevant to all experts because it is an important factor in terms of estimating, construction planning, and execution. We found that the shape of the wall (in particular, the curved and sloped walls), and added details (such as pilasters, wall reveals, and wall finishes), are of concern to both cost estimators and those in the building trades. With the Discovery Place project, the formwork expert emphasized that the existence of clipped walls, battered walls, and wall pilasters were of concern. He also mentioned that the exposed concrete wall and the existence of wall reveals have considerable impact on formwork productivity and workmanship.

Some design conditions, such as the fire- and acoustic-rating, were deemed less relevant to the formwork contractor and site superintendent. The formwork expert mentioned that “concrete” is a material with superior fire-rating and acoustic-rating characteristics, and it is the designer who should specify the specific type and grade of concrete required.

Practitioners perceived the attribute “wall turns” or “corners” as having an impact on productivity. For example, the site superintendent mentioned that the crew productivity in laying out formwork and pouring concrete is impacted by wall turns, and in particular, by T- and L-turns of concrete walls. He further mentioned that a change in wall dimensions from floor to floor impacted the reuse of formwork in the EDC project. He also stated that wall finish textures and details (e.g., exposed concrete and square corners) and the existence of wall reveals impact construction.

**Table 5.1 Expert opinion on design conditions related to the feature “Wall”**

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
Wall type	■ □ ●				No wall type is really generic; even two walls presumed to be of same type can actually have differing characteristics.
Wall shape:					
Straight vs. curved walls	■ □ ○	●			Curved walls are of concern to both estimator and tradesman, as they are significantly more labour-intensive than straight walls; Expensive formwork to build curved walls.
Vertical vs. sloped walls	□ ○	■ ●			Battered walls can have a slightly lower productivity rate when installing studs/ bridging/blocking, but not so much when installing sheathing/wallboard/insulation.
Clipped vs. non-clipped walls		■	■ ●		Not so relevant cost-wise, but can be significant layout concern to assure chipped portion(s) are correct.
Exterior vs. interior walls	■ □ ●	□	○		Speaks to same issues as does wall type.
Existence of blockouts, bulkheads, pilasters	■ □ ○	□	●		Added features/details aren’t nearly as costly as mistakes. Retrofitting/rework can be tedious if not done correctly, initially.
Constituent parts of a wall	■	■ ●			Level of concern, depends on uniqueness of certain parts constituting a particular wall
Wall material	■ ●				The fewer number of types of materials, the better, as this involves fewer sub-trades.
Wall finish type	□ ○ ●	■			
Wall dimensions (height, length, thickness)	○ □ ●	■			
Gross surface area_single side	●		■		
Gross surface area_double side			■		
Net surface area_single side (i.e., without		■ ●			Useful to compare the ratio of net surface area to gross surface area;

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
openings and penetrations)					suggests special features, which can be concerning.
Net surface area_double side (i.e., net without openings and penetrations)		■			
Volume of a wall		●	■		More relevant to concrete walls.
Wall curvature	■ □				Tighter the radius, mistakes very costly if wall geometry isn't perfect to adequate – fixing mistakes requires do-overs. If you are building a radius wall you are not working quickly. What do you with radius cuts and plywood that someone took hours to build, but after one use will be unusable?
Fire-rated vs. non fire-rated walls			○ ●	□	
Fire-rating of a wall (e.g., a wall with 2-hr fire-rating)	■		○	□	Code requires fire-resistant walls to be generally compliant with published, prequalified U.L. tested assemblies.
Acoustic-rated vs. non acoustic-rated walls			●	○ □	
Acoustic-rating of a wall (e.g., a wall with STC value of 50)		■		○ □	
Load-bearing vs. non-load-bearing wall	●	○			Load bearing wall needs special steel gauge studs.
Full height vs. ceiling height wall	●	■ ○	□		“Full height” walls are a different type than ceiling height walls, which may be either braced or not above the ceiling. It is important to identify which walls adhere to the underside of the ceiling, as opposed to those that extend to the slab above. These two different “wall heights” incur different costs due to differences in productivity rate.
Wall turns/corners	■ □ ○	■ ●			“Wraps” if highly detailed can be quite concerning.
Non-perpendicular (or orientation of) wall	□ ■	■ ○ ●			

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
turns					
Spatial location of walls within a floor (e.g., corridor walls, classroom walls)		■ ●	■		
Spacing of walls			□ ■ ○	●	

■ Project Manager; □ Formwork Manager; ○ Site Superintendent; ● Chief Estimator

### 5.2.1.2 Expert Opinion on the Feature “Column”

Table 5.2 presents design conditions related to the feature “column” and their relevance to construction practitioners. For the site superintendent, spacing of columns is more of an engineering issue. However, on some jobs, he said that he tends to question what will work and what will not, in order to make sure that the design is structurally safe, even though this issue is more the designer’s responsibility. He explained that he provides feedback to the architect and to the design engineer. He also mentioned that he also consults with rebar trades to make sure that they are confident in terms of the spacing. The formwork expert noted that the bigger the span, the easier the job, notwithstanding the fact that one needs to consider the dead load of the building. Specifically, he said:

*“If every grid bay is the same, it is far easier to build. Obviously you want to maximize the spacing, but that is more (in the domain) of engineering.”*

While many column-related design conditions listed in Table 5.2 are of concern to practitioners, formwork and site personnel are more concerned with the consistency of design. For example, the formwork expert we interviewed expressed his level of concern as follows:

*“I don’t care much whether columns are on grid or off-grid. What is really important is that grids remain consistent. If you can get the grid line to stay the same or add up to the same value all the time, it is easier for the trades to build and easier to design scaffolding for suspended slabs, because it is always the same load. When grids are consistent, you can move fly tables from one area to the next one, because that table is the same. In other words, if you keep the building consistent, the costs drop. Same thing applies to floor height. If the columns are changing all the time, you have to adjust column heights because it can’t be too high; when you go to pour the concrete, you’ve got to be able see inside the column to get it to the perfect elevation. If you get the same floor every time, you don’t have to change formwork. If you’re changing formwork, it costs money. The more consistent the design is, the cheaper it is to build. If you’ve got a building that goes around a circle or oval, and you want to do the glazing and do the concrete, how long do you think the guys would take to put the slab edging? Of course, it takes way longer. Change in column sizes costs money. You’ve got to design the load for every one of these redesign columns; you got rebar issues. For every floor, the detailer has to change the*

*detailing. For every different size of column, I have to build different column forms. I have to pay someone to change the forms or build the form. Contractors also like aesthetically pleasing buildings. If there are changes in size/shape of columns, to save or reduce concrete volume, and the volume is not that much, that is not worth it.”*

The relevance of grid lines to construction was somewhat ambiguous. The site superintendent explained that he doesn't have much concern about grid lines and off-grid vs. on grid columns. He noted that sometimes grid lines can pass the centre of the beam, and walls may not align on grid lines. However, he further explained that he always measures off-sets, and he dislikes offsetting columns. This indicates that he cares about whether or not columns are on-grid, or aligned properly. Another issue that was raised by one of the experts relates to the placement of columns. For example, a formwork expert is always keen to know whether columns are located right on the slab edge or projecting past the slab edge. Practically speaking, formwork practitioners like to see the columns inside of the building (from slab edge) by about a foot. That is a safe design, he remarked.

The superintendent on the EDC project mentioned column height, and change in column height from floor to floor, as the most relevant of the design conditions. Some of the columns in this project were constructed as per specifications, all the way from the foundation to the second floor, without obstruction. While the experts considered the size/dimension and shape of columns as very or moderately important, it was specifically the change in size, shape, and location of columns that was the more important issue for the project manager, site superintendent and formwork expert, than it was for the estimator. Existence of some project-specific design conditions, such as column drop panels, column mats, and off-grid columns, were particularly relevant to the formwork expert for the Discovery Place project. The formwork expert on this project was also concerned about the spacing of façade columns, as well as the maximum spacing of columns. There were changes in height, size, shape and location of the columns from floor to floor. Due to the variation in column shape, size, and location, the horizontal and vertical alignment of the columns was a significant issue in the Discovery Place project. In the EDC project, however, the alignment of columns was not challenging for the site superintendent since there were only two types of column size, of rectangular shape, and with uniform locations.

**Table 5.2 Expert opinion on design conditions related to the feature “Column”**

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
Column size/dimensions	■ □ ○	●			
Column shape	■ □ ○	■ ●			Estimators understand that forming circular columns may be more costly than square columns of a certain size range. Tradesmen may be concerned about the shape of smaller, round columns to a lesser degree.
Existence of column head/drop panels	■ □	■ □ ●			The material volume of ‘drop heads,’ insignificant compared to the added labour to form such details. Estimators diligently account for such additional costs.
Exterior column	■ □		●		Exterior columns are much more difficult to build; everything has to be plumb, perfectly plumb/ perfect, and the slab has to match up with the next floor.
Interior column		■ □	●		Conceivably, an exterior column might cost more when it is intended to be finished differently than an interior column of like dimension/shape. Actually, depending upon the detailing at perimeter edges of floor/roof slabs, layout precision for an exterior column might require a bit more time than does an interior column.
Off-grid vs. on-grid columns		□ ■	□ ○	○ ●	Avoiding costly mistakes requires a more precise layout for off-grid columns than do on-grid columns – layout might not cost significantly more, but mitigating higher risk makes it a key concern.
Spacing of columns:					Estimator relatively less concerned than practitioner(s)

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
					assuring accurate layout of work.
Spacing of columns in the X-direction		□	■ ○ ●		
Spacing of columns in the Y-direction		□	■ ○ ●		
Maximum spacing of columns	□	○	■ ●		
Minimum spacing of columns		■	□ ○ ●		Conceivably, slab forming systems may be impacted if columns are too closely spaced.
Centre-to-centre spacing between columns		■ □	○ ●		ditto
Clear spacing between columns		■ □	■ ○ ●		ditto
Alignment of columns:					Alignment of columns is a significant issue. If columns are on-line and consistent, I can use my fly forms.
Horizontal alignment of columns	■	■ □ ○	●		
Vertical alignment of columns	■ □ ○		●		
Design uniformity (or consistency):					Repetition enhances constructability, reducing risks of costly layout errors when “everything” is the same. Repetition enhances work efficiency.
Uniform size/shape of columns in a floor	■ □ ○	●			Changing column size will not save money.
Uniform size/shape of columns from floor to floor	■ □ ○	●			
Uniformity in the location of columns from floor to floor	■ □ ○	■	●		
Uniform spacing of columns in a floor	□	■ ○	● ■		
Uniform spacing of columns from floor to floor	□	■ ○	●		

■ Project Manager; □ Formwork Manager; ○ Site Superintendent; ● Chief Estimator

### 5.2.1.3 Expert Opinion on the Feature “Component Intersection”

Table 5.3 shows the experts’ response when interviewed about design conditions related to the “component intersection” feature. We could not acquire adequate input from the site superintendent and formwork expert on every detail listed in Table 5.3. The estimating expert explained that he does not normally work at that level of detail as it is not cost effective. This is probably due to the fact that it is incredibly tedious to manually account for these conditions in an estimate. The project manager provided comprehensive feedback, along with information about pertinent conditions/situations whereby the component intersection can be of great concern. One of the relevant design conditions of which we were unaware is the location of the component intersection. According to the project manager:

*“The location of the component intersection, for instance, whether it is centred on a wall or offset from the wall/column face can be an issue of concern. It is simpler to flush one face with another, than to evenly back set wall faces on both sides, relative to the thicker wall/column.”*

We further consulted with the experts about the relevance of typical types of wall-to-wall intersections (see Table 5.4). The experts, in general, were all very concerned about these intersection types as they agreed that they can impact the layout and detailing (or constructability), labour productivity, and construction costs. For instance, the formwork expert articulated his concerns as:

*“When you have corners, such as T, L, or whatever it is, it does look simple on paper, but it is really complicated to build. Because I have to pour for a section of it, and (then) come back and build a section tomorrow. So, I can’t pour all sections at the same time, which means I have lost a day. Every time you introduce corners, keys, etc., it doesn’t look difficult on paper, but it is difficult to build and costs more money. I have to keep this thing square and plumb.”*

The project manager highlighted a few types of intersections that are practically important to mention. They are: dry wall/masonry wall to column intersection, masonry wall to slab intersection, block wall to beam intersection. The site superintendent on the EDC project was very concerned about the nature and extent of the intersections of concrete walls with other components, such as pilaster, column, beam and slab, and column to beam intersection. Table 5.3

also shows the fact that the project manager rates the importance of component intersections, based on the relative dimension of the components, the component properties, and the component types.

**Table 5.3 Expert opinion on design conditions related to the feature “Component Intersection”**

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
A wall attached or otherwise connected to another type of wall (i.e., wall-to-wall intersection)	□	■			If the intersecting walls are of same type of construction, then detailing may not be as “special” as when types are different.
No. of walls attached or connected to a given wall	□		■		
Type of intersecting walls (e.g., masonry wall intersecting dry wall)		■			Transitioning dissimilar finish materials invoke trim details.
Property (e.g., size/dimensions, fire-rating, etc. ) of intersecting walls		■			
A wall attached or connected to a column (i.e., wall-to-column intersection)	○	■ □	■		
No. of columns or beams a wall is attached, or connected, to	■ ○	■ □			A single but complicated juncture is a more concerning issue than several simple junctures.
Type of intersecting wall and column (e.g., dry wall intersecting the round column)		■			Drywall partitions joining square shapes are slightly less of a concern than round shapes.
Relative dimensions of intersecting wall and column (or beam)			■		
Depth of intersection (inward intrusion perpendicular to the plane of intersection)	■	■			Keying can be straightforward as well as complex – depth dimension isn’t as concerning as when sides of key skew to form a “dovetail” slot.
Size of intersection		■	■		Layout and constructability is of more concern than cost.
Area of intersection				■	Practically irrelevant.
Volume of intersection				■	Practically irrelevant.

■ Project Manager; □ Formwork Manager; ○ Site Superintendent

**Table 5.4 Expert opinion on different types of “Wall-to-Wall Intersections”**

<i>Type of Wall- to- Wall Intersection</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
T-intersection	□ ○	●	■		Practically-speaking, the easiest type of wall-to-wall intersection, unless materials are dissimilar, which may not require special trim detailing.
L-intersection	□ ○	■ ●			Getting an L-shaped juncture ‘right’ could require more attention/care to layout, frame, and trim install on exposed edge –radiused corner slightly more labour than 90-degrees.
End-to-end intersection	□	■ ○	■ ●		
Overlapping type of intersection		■ ○			
Non-perpendicular intersection	■ □ ○	■ ●			Skewed intersections are “special,” whereas perpendicular intersections are “regular.” Depending on transitioning materials, skewed trim detailing can become complicated.

■ Project Manager; □ Formwork Manager; ○ Site Superintendent; ● Chief Estimator

#### 5.2.1.4 Expert Opinion on the Features “Opening” and “Penetration”

Table 5.5 and 5.6 present the experts’ feedback on different design conditions related to the features “opening” and “penetration” respectively. All of the experts agreed that the existence of openings and penetrations is important for construction. However, they expressed varying perspectives on the importance of different opening and penetration-related design conditions. The formwork expert acknowledged such divergence of opinions among the professionals and remarked:

*“Many people in the formwork business think that when there are openings on slab, there should not be contact footage area for formwork in the building, because there is no concrete. But, actually, sometimes that kind of contact footage can be very difficult, because it would take an incredible amount of scaffolding to take point loads.”*

For the site superintendent, locating the exact size and location of all openings and penetrations on slabs and walls was a painful exercise as they were not always explicit in the drawings. This was an issue of concern because he consistently needed to instruct the work crews as to their locations. Also, prefabricating the rebar cages had to be put on proper places. The site superintendent explained:

*“If you don’t know where the penetrations or openings are going, it creates site coordination problems.”*

The relevance or impact of design conditions on construction can also vary depending on the nature and existence of other design conditions. For instance, the project manager revealed that a vertical location of wall penetrations is aesthetically critical only when it is located below the suspended ceiling finish. On the other hand, it is functionally critical when interstitial/attic space is “tight.” He further stated that a horizontal location of slab penetrations is more critical when they are located nearer to columns/walls.

Project environmental factors, such as the type of building (e.g., industrial, commercial, residential, institutional, etc.) and the parties involved, also provide a preliminary indication to practitioners about the likelihood of occurrences of certain important design conditions. As illustrated by the formwork expert in the following comment:

*“If it is an industrial building or pump station, you would expect lots of problems with penetrations.”*

The site superintendent explained it this way:

*“It may sound odd, but we know to some extent, without even looking at all design drawings, which design conditions to expect in a project designed by certain architects and structural engineers, because they have a distinctive style in terms of designing a building.”*

**Table 5.5 Expert opinion on design conditions related to the feature “Opening”**

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
Existence of openings on walls or slabs	■ □ ○ ●				Every time you have an opening, you have to put reinforcement around it to protect it from cracking. For large holes on the slab in the upper floors, sufficient work must be done to take a point load to the ground.
Size/dimension of opening	□ ○ ●	■			Small openings, not much problem. Large openings on slabs may require re-shoring in order to make a false deck. Slab opening require slab edging.
Depth of opening		■			By definition, if zero depth, then this is not an opening; if partial depth (wall thickness), then this is a recessed “niche.” By default, if no mention of a specified depth of opening, then opening is presumed equal to full wall/slab thickness.
Area of opening	●	□ ○	■		
Volume of opening				■	Practically irrelevant.
Perimeter of opening		□ ○	■ ●		Estimators may care more than other practitioners as they need to calculate the length of running form.
Horizontal location of wall openings		○ ■	□ ●		
Vertical location of wall openings		○ ■ ●	□		
Horizontal location of slab openings	■	■ □ ○ ●			
Spacing of openings on walls and slabs		■	□ ○	●	Designer must specify the required spacing to avoid random placements.
Uniform size of openings	○	□ ●	■		Repetition is helpful and, ultimately, cost-effective.
Uniform location of openings	□ ○		■ ●		Pre-planning opening locations strategically is advantageous.
Uniform spacing of openings		○	■	●	

■ Project Manager; □ Formwork Manager; ○ Site Superintendent; ● Chief Estimator

**Table 5.6 Expert opinion on design conditions related to the feature “Penetration”**

<i>Design Conditions</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
Existence of wall/slab penetrations (e.g., duct, conduit, pipe)	■ ■ □ ○	●			
Size/dimension of penetration	■ □ ○	●			
Depth of penetration		■	■		By definition, if not a through-wall penetration, it really isn't a “penetration,” but merely an indented surface feature.
Area of penetration		○	□ ■ ●		Can be less critical, unless the ratio of hole area-to-wall area is greater than a relevant threshold value of, for instance, 25%.
Volume of penetration			□	■	
Perimeter of penetration		□ ○	■ ●		Can be less relevant, unless the sleeve detail is costly/critical.
Horizontal location of wall penetrations		○ □	■ ●		
Vertical location of wall penetrations	○	■ □	■ ●		Aesthetically critical only when located below suspended-finish ceiling; functionally critical when interstitial/attic space is “tight.”
Horizontal location of slab penetrations	■ ○	■ □	●		More critical when nearer to columns/walls.
Spacing of penetrations			□ ■	●	Designer must specify if important; otherwise random placements might arise. Generally more concerning/critical for slab penetrations than for wall penetrations.
Uniform size of penetrations	○	□	■ ●		Repetition may be preferred, but in practical terms, shapes/sizes tend to vary depending upon purpose for a particular hole.
Uniform location of penetrations	□ ○	■	●		Penetrations tend to be customized rather than generic placements.
Uniform spacing of penetrations	■	■ □ ○		●	Designer should specify minimum clearance distance between adjacent holes.

■ Project Manager; □ Formwork Manager; ○ Site Superintendent; ● Chief Estimator

Notwithstanding, a difference of professional opinion, there is a common understanding about what opening/penetration-related design conditions are relevant to practitioners. For example, in the Discovery Place Project, the formwork contractor was particularly concerned about the size, area, perimeter, and location of the openings and penetrations and their consistency on the walls and slabs. Similarly, the site superintendent in the EDC project was also very critical about the existence, size and location of penetrations. The estimating expert emphasized the fact that wall and slab penetrations can be of concern for both drywall contactors and formwork contractors. This suggests that identifying the openings, penetrations, and related specific information, such as size and location, is of great concern in every building project. The extent of this concern, however, may vary depending on the type of project, such as hospital, institutional, residential, pharmaceutical projects.

We received valuable feedback from the project manager in particular, as to the relevance and practical significance of different location-specific parameters for designating wall openings/penetrations. The results are shown in Table 5.7 and indicate a significant or moderate relevance of factors, in general. The project manager pointed out that the horizontal location of openings/penetrations tends to be less relevant as the percentage of clearance distance increases. He provided useful suggestions about including certain parameters in our specification. In addition to the location of wall openings/penetrations from “floor level above,” as formalized in the specifications, the project manager also recommended using “beam/ceiling soffit above,” as an additional reference measure. In addition to characterizing the location of openings/penetrations from “wall-to-wall intersections” and “wall-to-column intersections,” he also advised using other parameters, such as “distance to the nearest wall opening,” such as a doorway, in the approach.

**Table 5.7 Expert opinion on design conditions related to the “Horizontal and Vertical Locations of Openings and Penetrations”**

<i>Location of Penetrations/Openings on wall measured from the:</i>	<i>Relevance/Importance</i>				<i>Comments, if any</i>
	<i>Significant</i>	<i>Moderate</i>	<i>Little</i>	<i>Irrelevant</i>	
Left edge of the wall (or first edge of the wall)		■	■		Less significant as magnitude of clearance dimension increases.
Right edge of the wall (or second edge of the wall)		■	■		Ditto
Top (or top edge ) of the wall	■	■			Vertical clearance to obstruction above equally concerning as horizontal clearance to the side wall, perpendicular to subject wall.
Bottom (or bottom edge) of the wall	■				
Floor level	■				
Floor above		■			Minimum vertical clearance should be specified by designer – rule of thumb might be: not less than, for instance, 0.5 diameter of penetrating hole (even if circle fits inside square/rectangle).
Wall-to-wall intersection	■	■			
Wall-to-column intersection	■				Columns “stair-stepping” outward at intersecting wall, corners are of concern, in that the effective clearance dimension can be significantly reduced. Minimum horizontal clearance should be specified by designer.

■ Project Manager

### 5.2.2 Retrospective Analysis

We conducted retrospective analysis of the design conditions identified in the literature and interviews to demonstrate the soundness of the developed research. The retrospective analysis provides evidence that our approach provides significant support for identifying design conditions of interest to construction practitioners in comparison to the state-of-the-art tools.

The retrospective analysis focused on “component” features in general, and “wall” and “column” features in particular, and includes “opening,” “penetration,” and “component intersection” features. In order to conduct the retrospective analysis, we compiled a list of design conditions that are significant to construction, based on a thorough review of the literature and confirmation from our interviews with construction experts for the projects studied. The compiled sets of design conditions represent generally useful information for different construction domains, trades, (e.g., construction planning, concrete construction, interior construction, MEP coordination, and site layout) and applications (e.g., cost estimating, method selection, and constructability). For each design condition, we evaluated the extent to which our research tools, in comparison to the state-of-the-art tools, provide support in identifying a given design condition from a BIM model. The state-of-the-art tools evaluated were *Solibri Model Checker* and *Navisworks*. These tools were selected because they provide the most advanced support for analyzing a BIM from the construction perspective. In an effort to evaluate the effectiveness of the developed research approach and the existing tools more precisely and unequivocally, we differentiate the degree of support provided into three categories: “full,” “partial,” and “none.” Tables 5.8 to 5.13 summarize our evaluation results. The corresponding numbered references used in these Tables are listed in Appendix A.

Table 5.8 lists the level of support results for identifying design conditions related to “component” features in general. State-of-the-art tools provide support to extract component dimensions, spatial locations, clearances, and the component type. They provide partial support in identifying most of the component material-related information, compared to our research tool. Our tool provides partial support in identifying many design conditions that state-of-the-art tools do not.

Table 5.9 lists the results of our evaluation for identifying wall-related design conditions. Except for wall finish type and net surface area, our research tool also provides the support to identify design conditions, as do the state-of-the-art tools. Moreover, our research tool can be

applied to identify a number of other design conditions that are important to construction, especially those related to wall shape, curvature, wall turns, the angle of wall turns, and some forms of aggregated information, such as the average wall height, maximum or minimum wall length, and the variation in the dimensions/size, shape of the walls.

We evaluated the extent to which our research supports the identification of column-related design conditions about which practitioners are concerned, compared with the state-of-the-art tools. Table 5.10 shows the results of the evaluation.

Table 5.11 provides detailed analysis results of the evaluation for the level of support for the feature “component intersection.” Our research has the ability to identify component intersections based on a wide range of criteria. State-of-the-art tools do not provide explicit support to identify the existence of component intersections and related design conditions, as existing tools are geared towards identifying clashes of supporting conflict detection, and evaluating the BIM model for errors or interferences.

Table 5.12 and Table 5.13 present the results of the evaluation for the level of support to identify the existence of openings and penetrations, and their specific parameters and conditions, respectively. State-of-the-art tools, such as the *SMC*, provide support for identifying component openings, their dimensions, area, and perimeter. They also identify some aspects of their variations, and some location-specific data. The location data, however, deals mainly with global coordinates and elevation data, and places less emphasis on the relationship between boundaries of the host component, and other references of concern. Existing tools provide no direct support for identifying design conditions related to “penetrations.”

**Table 5.8 Level of support results for the feature “Component” in general**

Relevant Design Conditions	Reference	State-of-the-Art Tools			Our Approach		
		Full Support	Partial Support	No Support	Full Support	Partial Support	No Support
Component dimensions (e.g., height, depth, width, thickness, length)	[1,2,3,4]	√			√		
Maximum/minimum dimension of components	[5]			√	√		
Component location in a floor (e.g., below grade, main floor)	[6,7]	√			√		
Modular layout of components	[8]			√			√
Adjacency of building components	[9,10]		√				√
Repetition of component dimensions/sizes and distances in a floor	[1,2,3,8,11]			√		√	
Repetition of distances between components from floor to floor	[1,2,3,8,11]			√			√
Shape of components	[2,3]			√		√	
Irregularity in the shape of components	[2,3,4]			√			√
Existence of blockouts, bulkheads, pilasters, drop heads	[2,3]			√			√
Changes in dimensions, size/cross sections, shape of components	[1,2,3]			√		√	
Variation in the size and location of components	[1,2,3,15]			√		√	
Component spacing	[1,16]			√		√	
Maximum and minimum spacing of components	[5]			√		√	
Component clearances	[26,27]	√					√
No. of components attached, or connected to, a component	[17]		√		√		
Type of component material(s)	[12,13,14]	√				√	
Material characteristics of a component	[1]		√				√
Component parts	[12,14]		√			√	
Characteristics of component parts (e.g., size of masonry units)	[5, 6, 14,17,18]		√				√
Component finish type	[4,12]		√				√
Type of connections	[13]			√			√
<b>No. of Relevant Design Conditions Treated: (22)</b>	<b>Count</b>	<b>(4)</b>	<b>(6)</b>	<b>(12)</b>	<b>(4)</b>	<b>(8)</b>	<b>(10)</b>

**Table 5.9 Level of support results for the feature “Wall”**

Relevant Design Conditions	Reference	State-of-the-Art Tools			Our Approach		
		Full Support	Partial Support	No Support	Full Support	Partial Support	No Support
Wall type (as defined in the BIM application)	[4,6,16,22]	√			√		
Wall type by specific function (e.g., fire wall/shaft wall/core wall)	Interviews			√			√
Wall type by spatial location in a floor (e.g., corridor/theatre wall)	Interviews		√				√
Sloped walls	[12]			√	√		
Straight walls	[6,12]			√	√		
Curved walls	[1]			√	√		
Clipped walls	Interviews			√	√		
Exterior or interior walls	[6]	√			√		
Wall height	[1,4,16,22]	√			√		
Average wall height	Interviews			√	√		
Ceiling or full height walls	[16]		√			√	
Change in wall thickness and height from floor to floor	Interviews			√			√
Length of wall	[1, 4,6,12]	√			√		
Minimum and maximum wall length	[1,6,12]			√	√		
Wall curvature	[4,12]			√	√		
Existence of wall corbels, ledges, and pilasters	[12,23]			√			√
Variation in the size, height, and shape of walls	[12]			√	√		
Wall turns (corners)/ no. of wall turns	[6,22,23]			√		√	
Non-perpendicular wall turns (orientation of wall turns)	[6,12,22]			√		√	
Type of wall turns/corners (T, L, or non-perpendicular corners)	Interviews			√			√
Load bearing vs. non-load bearing walls	[28]; Interviews	√			√		
Acoustically-rated walls	[28]; Interviews	√			√		
Fire-rated walls	[28]; Interviews	√			√		

Relevant Design Conditions	Reference	State-of-the-Art Tools			Our Approach		
		Full Support	Partial Support	No Support	Full Support	Partial Support	No Support
Net surface area (single or double side)	[28]; Interviews		√				√
Gross surface area (single or double side)	[28]; Interviews		√			√	
Wall finish type	Interviews	√					√
Sides of wall finishes	Interviews			√			√
Exposed concrete wall	Interviews			√			√
Wall reveals/finish texture details	Interviews			√			√
<b>No. of Relevant Design Conditions Treated: (29)</b>	<b>Count</b>	<b>(8)</b>	<b>(4)</b>	<b>(17)</b>	<b>(15)</b>	<b>(4)</b>	<b>(10)</b>

**Table 5.10 Level of support results for the feature “Column”**

Relevant Design Conditions	Reference	State-of-the-Art Tools			Our Approach		
		Full Support	Partial Support	No Support	Full Support	Partial Support	No Support
Column height	Interviews	√			√		
Change in column height from floor to floor	Interviews			√		√	
Column size/shape	[1,2,3, 8]	√			√		
Change in column size/shape from floor to floor	[15]			√		√	
Change in column location from floor to floor	[15]			√		√	
Existence of column head/drop panels	[2,8]			√			√
Uniformity in the layout of columns	[20]			√			√
Uniformity in the spacing of columns in a floor	[20]			√		√	
Uniformity in the spacing of columns from floor to floor	Interviews			√			√
Horizontal and vertical alignment of columns	[5,21]			√		√	
Uniformity in the size of column head/drop panels	Interviews			√			√
Spacing of façade columns	Interviews			√	√		
Off-grid vs. on-grid columns	Interviews			√	√		
Uniformity of off-grid columns from floor to floor	Interviews			√	√		
Offset columns	Interviews			√			√
Column offset distance	Interviews			√			√
Location of exterior columns from the slab edge	Interviews			√			√
Maximum and minimum spacing between columns	Interviews			√	√		
Clear vs. centre-to-centre spacing of columns	Interviews			√	√		
Exterior vs. interior columns	Interviews			√	√		
<b>No. of Relevant Design Conditions Treated: (20)</b>	<b>Count</b>	<b>(2)</b>	<b>(0)</b>	<b>(18)</b>	<b>(8)</b>	<b>(5)</b>	<b>(7)</b>

**Table 5.11 Level of support results for the feature “Component Intersection”**

Relevant Design Conditions	Reference	State-of-the-Art Tools			Our Approach		
		Full Support	Partial Support	No Support	Full Support	Partial Support	No Support
Intersection or connectivity of building components	[3,5,9,12,17,24]	√			√		
Wall-to-wall intersections	[4]	√			√		
Intersection of masonry wall with structural steel elements	[6,12]			√	√		
No. of intersecting components	[6,17]		√		√		
Property of intersecting components (e.g., material type)	[12]		√		√		
Type of intersecting components	[12]		√		√		
Relative dimension of intersecting components	[5,2,13,25]			√		√	
Complexity of intersection (e.g., complex slab-beam intersection)	[3]			√			√
Reinforcement ratio of the intersecting components	[17]			√			√
Depth of intersection	Interviews		√		√		
Size of intersection	Interviews			√	√		
No. of beams or columns wall is attached to	Interviews			√	√		
Type of intersecting wall and column (e.g., a dry wall intersecting a round column)	Interviews			√	√		
Relative dimensions of intersecting wall and column	Interviews			√			√
Type of intersecting walls (e.g., a masonry wall intersecting a dry wall)	Interviews			√	√		
Property (e.g., size/dimensions, fire rating, etc.) of intersecting walls	Interviews			√	√		
Location of intersection	Interviews			√			√
T-intersection of walls	Interviews			√			√
L-intersection of walls	Interviews			√			√

Relevant Design Conditions	Reference	State-of-the-Art Tools			Our Approach		
		Full Support	Partial Support	No Support	Full Support	Partial Support	No Support
End-to-end intersection of walls	Interviews			√			√
Overlapping wall intersection	Interviews			√			√
Non-perpendicular intersection of walls	Interviews			√	√		
<b>No. of Relevant Design Conditions Treated: (22)</b>	<b>Count</b>	<b>(2)</b>	<b>(4)</b>	<b>(16)</b>	<b>(13)</b>	<b>(1)</b>	<b>(8)</b>

**Table 5.12 Level of support results for the feature “Opening”**

Relevant Design Conditions	Reference	State-of-the-Art Tools			Our Approach		
		Full Support	Partial Support	No Support	Full Support	Partial Support	No Support
Existence of openings	[6]	√			√		
No. of wall openings	[12]	√			√		
Dimensions/size of component openings	[1]	√			√		
Horizontal location of wall openings	Interviews			√	√		
Vertical location of wall openings	Interviews		√		√		
Horizontal location of slab openings	Interviews			√			√
Uniformity in the size of openings	[1,4,12]		√		√		
Uniformity in the size and location of wall/slab openings	[1,4,12]		√			√	
Spacing of openings	Interviews			√			√
Uniformity in the spacing of openings	[12,19]			√			√
Openings on certain type of walls (e.g., fire-rated stud walls)	Interviews			√	√		
Opening type (e.g., window, door)	Interviews		√			√	
Area of opening	Interviews	√			√		
Perimeter of opening	Interviews	√			√		
Depth of opening	Interviews			√		√	
<b>No. of Relevant Design Conditions Treated: (15)</b>	<b>Count</b>	<b>(5)</b>	<b>(4)</b>	<b>(6)</b>	<b>(9)</b>	<b>(3)</b>	<b>(3)</b>

**Table 5.13 Level of support results for the feature “Penetration”**

Relevant Design Conditions	Reference	State-of-the-Art Tools			Our Approach		
		Full Support	Partial Support	No Support	Full Support	Partial Support	No Support
Existence and extent of wall/slab penetrations	[12,16]	√			√		
Penetration on certain types of component (e.g., fire-rated assemblies)	Interviews			√	√		
Type of penetrating element	Interviews	√			√		
Vertical location of wall penetrations	[11,16]			√	√		
Horizontal location of wall penetrations	Interviews			√	√		
Horizontal location of slab penetrations	Interviews			√			√
Uniformity in the size and location of wall/slab penetrations	[11]			√		√	
Size/dimension of penetration	Interviews			√	√		
Area of penetration	Interviews			√	√		
Perimeter of penetration	Interviews			√	√		
Spacing of slab penetrations	Interviews			√			√
Uniform spacing of penetrations	Interviews			√			√
<b>No. of Relevant Design Conditions Treated: (12)</b>	<b>Count</b>	<b>(2)</b>	<b>(0)</b>	<b>(10)</b>	<b>(8)</b>	<b>(1)</b>	<b>(3)</b>

### **5.2.2.1 Summary of the Retrospective Analysis**

Table 5.14 presents the summary of the retrospective analysis. Figure 5.2 displays the summarized results in terms of level of support expressed in percentage. While the state-of-the-art tools provide comparable support to identify design conditions related to components in general, our approach provides increased support for identifying design conditions related to the features “wall,” “column,” “component intersection,” “opening,” and “penetration.” The results indicate that the state-of-the-art tools lack substantial support for identifying construction-relevant design conditions; the only features that they can identify more than 50% of are “openings.” In contrast, our approach finds roughly 80% of “opening” and 75% of “penetration” related design conditions. While we still fail to identify around 35% of construction-specific design conditions related to “wall,” “column,” and “component intersection” features, this is still considerably better than the current state-of-the-art tools.

While our research approach provides significant support for identifying design conditions that are relevant to practitioners, further work needs to be carried out to provide support for those design conditions that are not yet supported or only partially supported, and to further extend our approach to identifying design conditions for other components. In the section that follows, we describe in further detail, the soundness and usefulness of our approach, and we highlight some areas for future research.

**Table 5.14 Summary of the level of support results for identifying different design conditions**

Feature	Relevant Number of Design Conditions Treated	State-of-the-Art Tools						Our Approach					
		Full Support		Partial Support		No Support		Full Support		Partial Support		No Support	
		Count	Percent (%)	Count	Percent (%)	Count	Percent (%)	Count	Percent (%)	Count	Percent (%)	Count	Percent (%)
Components in general	22	4	18	6	27	12	55	4	18	8	36	10	45
Wall	29	8	28	4	14	17	59	15	52	4	14	10	34
Column	20	2	10	0	0	18	90	8	40	5	25	7	35
Component intersection	22	2	9	4	18	16	73	13	59	1	5	8	36
Opening	15	5	33	4	27	6	40	9	60	3	20	3	20
Penetration	12	2	17	0	0	10	83	8	67	1	8	3	25

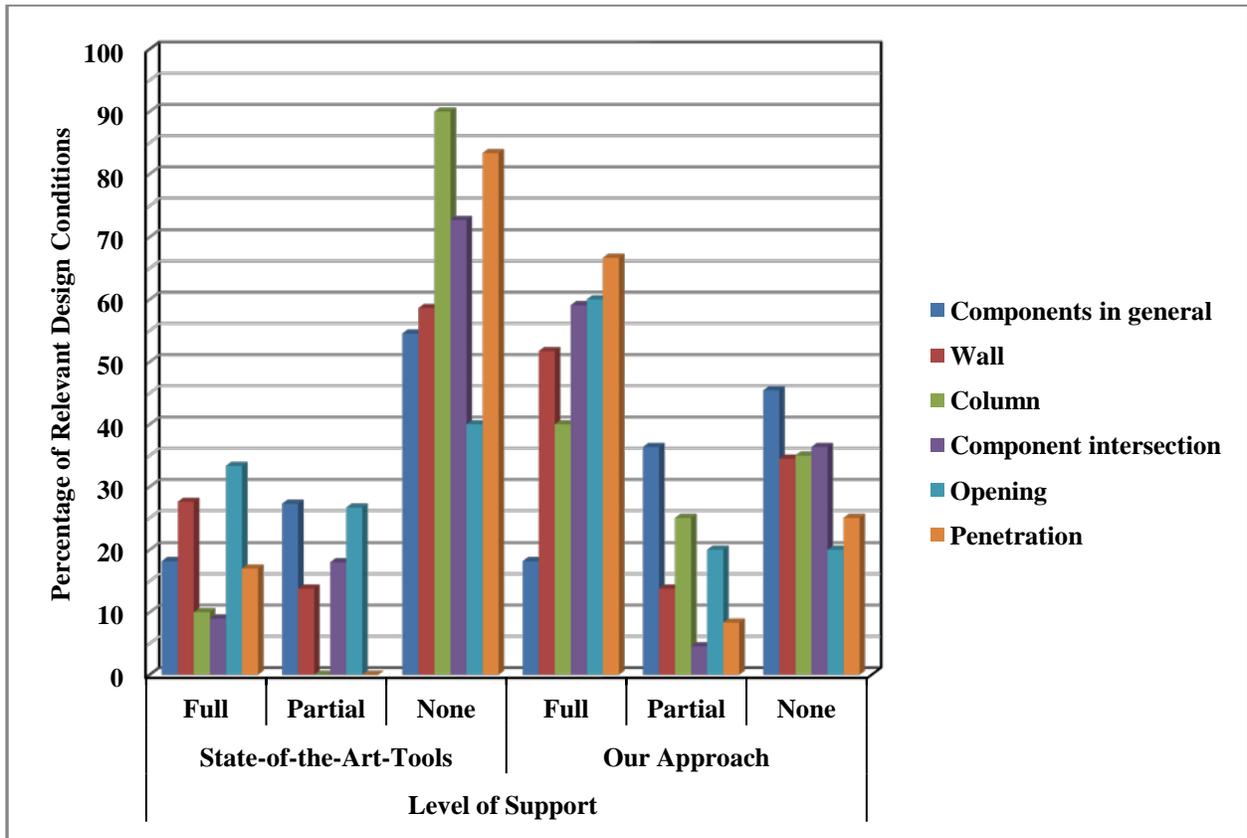


Figure 5.2 Summary of the results for the level (or degree) of support for different construction-specific features

### 5.2.3 Descriptive and Interpretative Analysis

In the previous section, we showed the extent to which the developed research tool can identify different kinds of design conditions that are of concern to practitioners from a BIM. In this section, we critically evaluate the *intellectual merits* of the research, and the quality of information provided to the practitioner. We also discuss the types of *actionable* and *insightful* information that can be generated using our approach. Where applicable, we critically and categorically examine the extent to which existing tools provide comparable support in terms of their ability to: (i) query different types of basic and extended queries on a BIM, (ii) inquire for very specific and focused types of information, and (iii) demonstrate flexibility with which such information can be queried by users to meet their specific construction requirements and preferences. The intent is to demonstrate the potential value of the developed research in relation to existing tools and current work practices.

#### 5.2.3.1 Richer Representation of Component Types

The cost estimators, such as those described in the UBC-O project, have different ways of defining component types, grouping or categorizing the required component-related information, and using various types of quantitative information. Table 5.15 shows some of the criteria that cost estimators use to define wall types. Architects or designers define wall types generically, on the premise that practitioners can unambiguously extract relevant information and use it according to their requirements. Using traditional 2D drawings they employ symbols, assembly details, and descriptions, as shown in Figure 5.3. They annotate the drawings using pre-defined symbols to convey design intent and requirements (see Figure 5.4). In BIM applications, such as *Revit*, designers use pre-defined wall types, or define new wall types, derived from the family of wall types already defined, and add assembly details, and type properties of walls (see Figure 5.5). Notwithstanding how components are represented, cost estimators, schedulers, and other practitioners must manually work to filter and group similar wall types, based on criteria, such as those listed in Table 5.15.

**Table 5.15 Different criteria for defining wall types by cost estimators**

<b>Wall typing criteria</b>	<b>Example/s (with type italicized)</b>
Generic wall name	<i>Masonry wall, Drywall, Concrete wall</i>
Constituent materials	<i>Steel stud drywall, Brick veneer wall</i>
Material properties	<i>5/8" drywall, Wall concrete-35Mpa</i>
Location in relation to interior or exterior of a building	<i>Interior steel stud walls, Exterior wall 8' to 16 high</i>
Shape (plan view)	<i>Straight wall, Curved wall</i>
Shape (elevation view)	<i>Vertical wall, Battered wall</i>
Change in height	<i>Clipped wall, Non-clipped wall</i>
Dimensions (height/length/thickness)	<i>190 mm concrete block wall</i>
Wall height relative to slab and ceiling	<i>Full height wall, Ceiling height wall</i>
Location on the floor	<i>Basement wall-300 mm, Foundation wall-concrete block</i>
Location on the floor space	<i>Classroom wall, Corridor wall, Theater wall-300 mm</i>
Generic wall properties	<i>Fire-rated wall, Acoustically-rated wall, Load-bearing wall</i>
Type of construction	<i>Precast wall panel, CIP concrete wall</i>
Wall function/usage	<i>Shaft wall, Core wall, Fire wall</i>

## EXTERIOR WALL TYP SCHEDULE

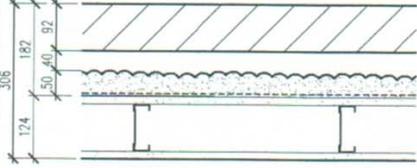
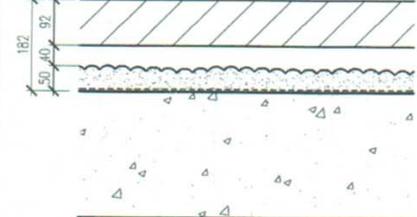
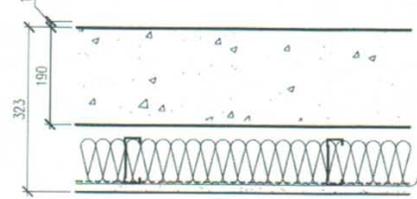
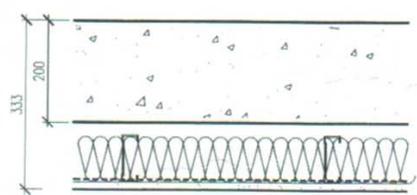
NO.	ILLUSTRATION	ASSEMBLY
		<p><b>MASONRY WALL ASSEMBLY</b>            92mm BRICK VENEER            40mm AIR SPACE            50mm SPRAY FOAM INSULATION - R-12            AIR/VAPOUR BARRIER MEMBRANE            12mm EXTERIOR SHEATHING            92mm STRUCTURAL METAL STUDS (18 GAUGE)            16mm GWB</p>
		<p>AS PER W1A BUT W/ 152mm (18 GAUGE) STRUCTURAL METAL STUDS</p>
		<p><b>MASONRY WALL ASSEMBLY</b>            92mm BRICK VENEER            40mm AIR SPACE            50mm SPRAY FOAM INSULATION - R12            AIR/VAPOUR BARRIER MEMBRANE @ OPENINGS, TRANSITIONS AND CONSTRUCTION JOINTS            CAST IN PLACE CONCRETE WALL</p>
		<p>AS PER W1A BUT W/ 190 mm CONCRETE BLOCK IN LIEU OF STRUCTURAL METAL STUDS</p>
		<p><b>TILT-UP WALL ASSEMBLY</b>            190mm THICK TILT-UP CONCRETE PANEL            89mm BATT INSULATION R14            92mm METAL STUDS (20 GAUGE) OFFSET 25mm FROM CONCRETE FACE            VAPOUR BARRIER MEMBRANE            16mm STANDARD ABUSE RESISTANT GWB TO 2400mm            16mm GWB ABOVE 2400 MM</p>
		<p><b>TILT-UP WALL ASSEMBLY</b>            200mm THICK TILT-UP CONCRETE PANEL            89MM BATT INSULATION R14            92mm METAL STUDS (20 GAUGE)            25mm OFFSET FROM CONCRETE FACE            VAPOUR BARRIER MEMBRANE            16mm STANDARD ABUSE RESISTANT GWB TO 2400mm            16mm GWB ABOVE 2400 MM</p>

Figure 5.3 Style of designating and defining wall types in traditional work practice

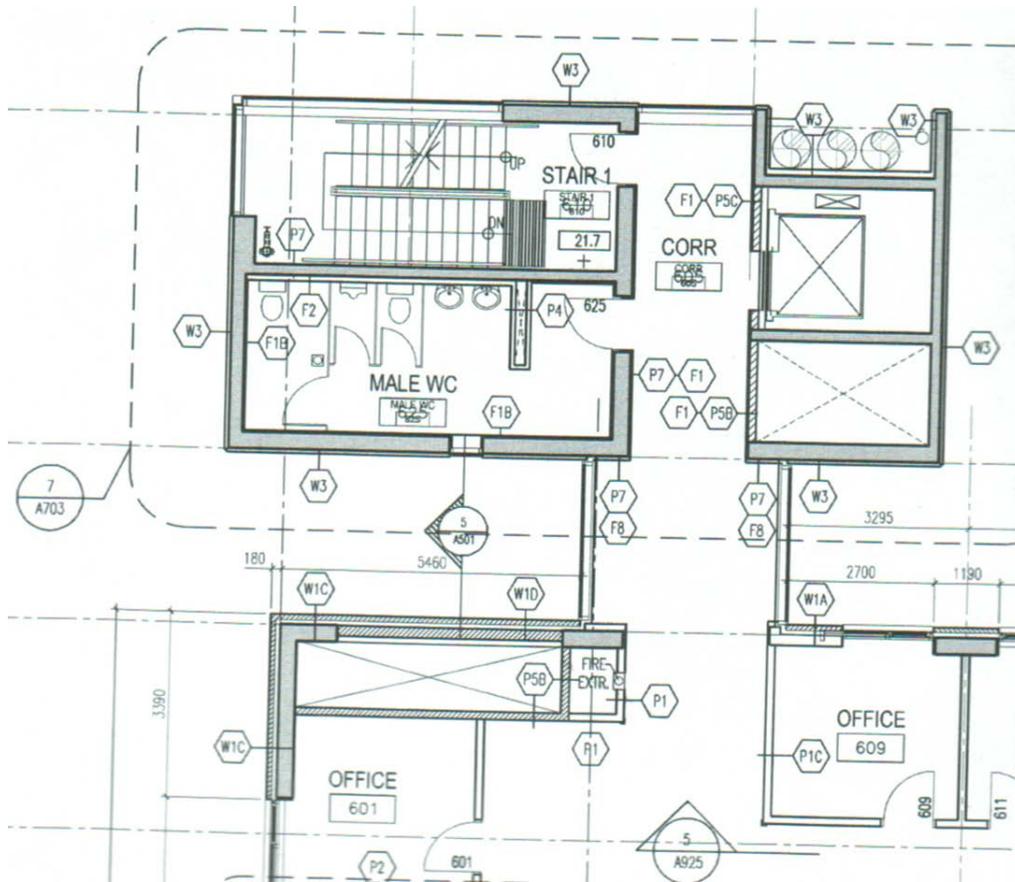
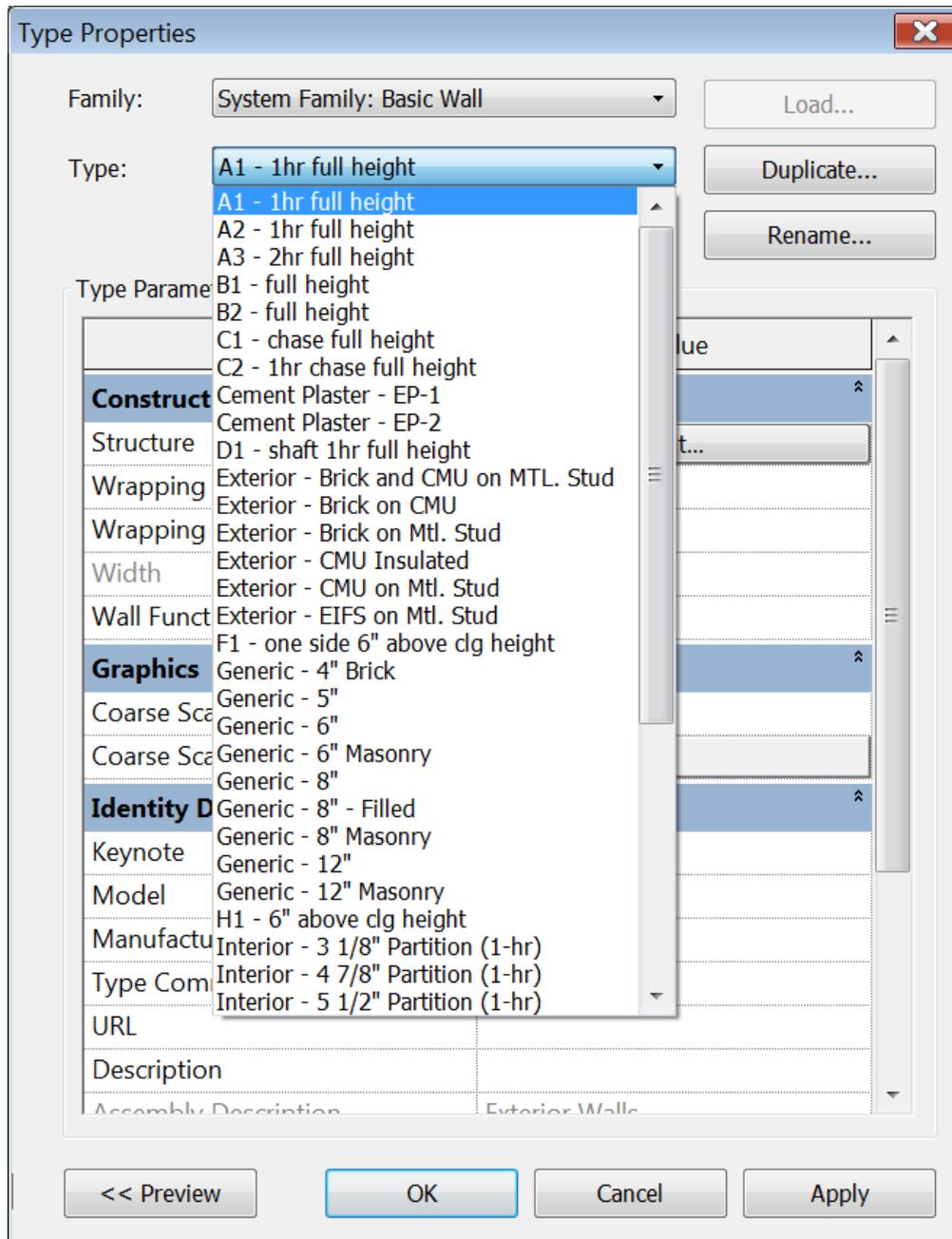


Figure 5.4 An example of applying the wall definition from Figure 5.3 to the actual design



**Figure 5.5** Methods of defining wall types in *Revit Architecture*

Practitioners use different methods to categorize walls. They also require different types of queries based on wall types. For instance, estimators might want to create a query to: *“Group all walls by type and height, and then calculate the total linear foot and area for each wall type.”* This query is of interest to note, because different wall types may require different construction operations. For example, estimators use different productivity rates or

equipment for different ranges of wall heights. Estimators need adequate computer support to flexibly define wall types and extract information, as in the above query. In this research, we define attributes that characterize components, such as “wall” in the feature ontology, and we instantiate their values in the FBM. Cost estimators can use these predefined attributes to characterize a specific type of wall, define grouping criteria, such as the wall material type and height, and define quantitative functions, such as ‘sum,’ to calculate the total wall area of each type and height category. The estimator can use a range of wall properties, such as dimensions (thickness, length, height), wall type materials (masonry, stud, concrete), shape (curved, clipped, straight, battered), and other properties (load-bearing vs. non-load-bearing, exterior vs. interior, fire-rating, acoustic-rating, etc.) to define wall types. A variety of quantitative functions enable estimators to obtain the required information, such as count, percent count, sum (total), average, maximum, minimum, and range. When the estimating expert on the UBC-O Project was asked about the value of such information, and the query process, he was impressed with the query steps involved in our approach and the actionable results that it is capable of generating.

Estimators in current practice group walls and take-off quantities using non-BIM based interactive and semi-automated processes. For instance, the estimators that we interviewed spend a significant amount of time colour-marking appropriate conditions onto pdf drawings of the building plans using computer tools, such as On-screen Take-off in order to categorize components by material and other component properties. For example, on the UBC-O Project, estimators marked the types of building components on floor plans (Figure 5.6), different types of finishes on components, such as on floors/slabs (Figure 5.7), different ceiling types and finishes in the reflected ceiling plan (Figure 5.8), and different types of exterior elements, such as curtain walls, sunscreen systems, and vents, in elevation views (Figure 5.9). When a design changed, the estimator had to repeat the process again for the components affected.

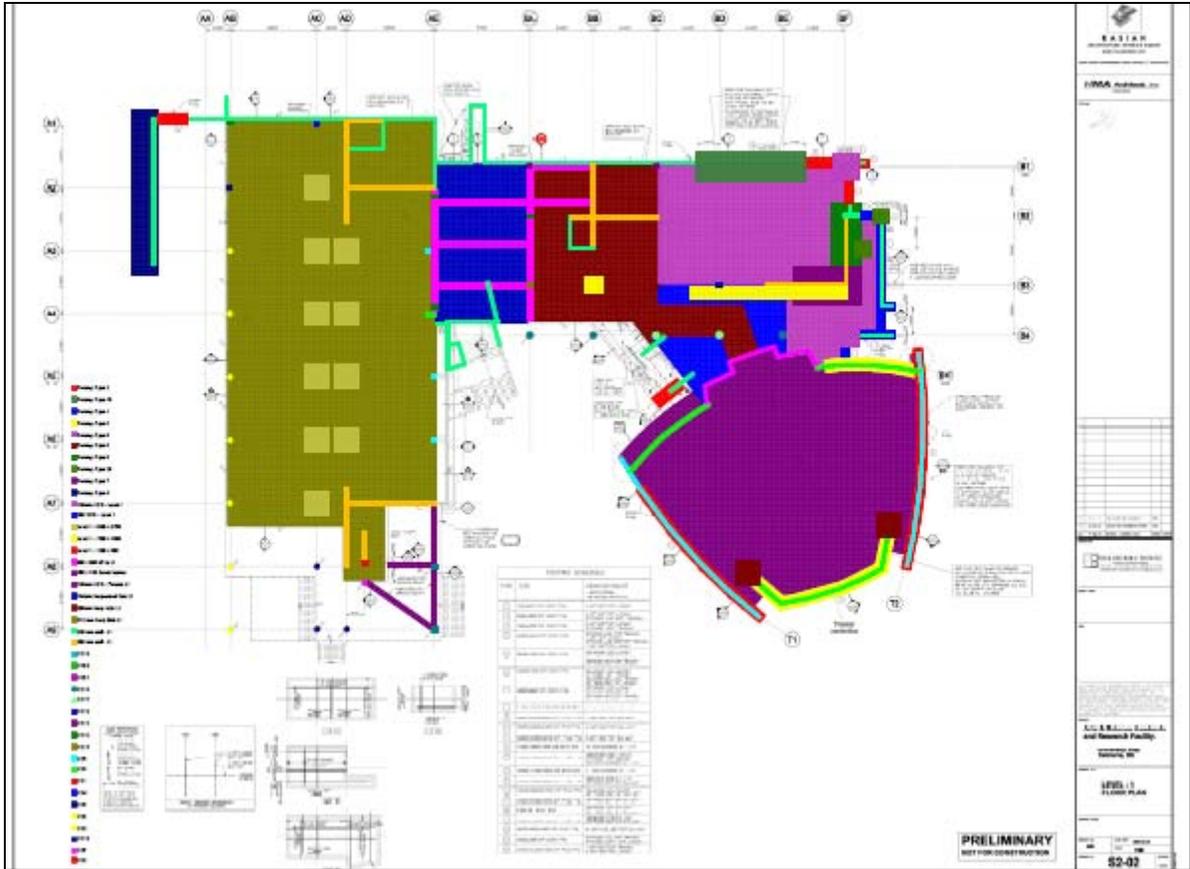


Figure 5.6 Colour-marking the appropriate type of component – walls, columns, slabs on grade, suspended slabs – on Level 1



Figure 5.7 Colour-marking of different floor finishes on Level 1

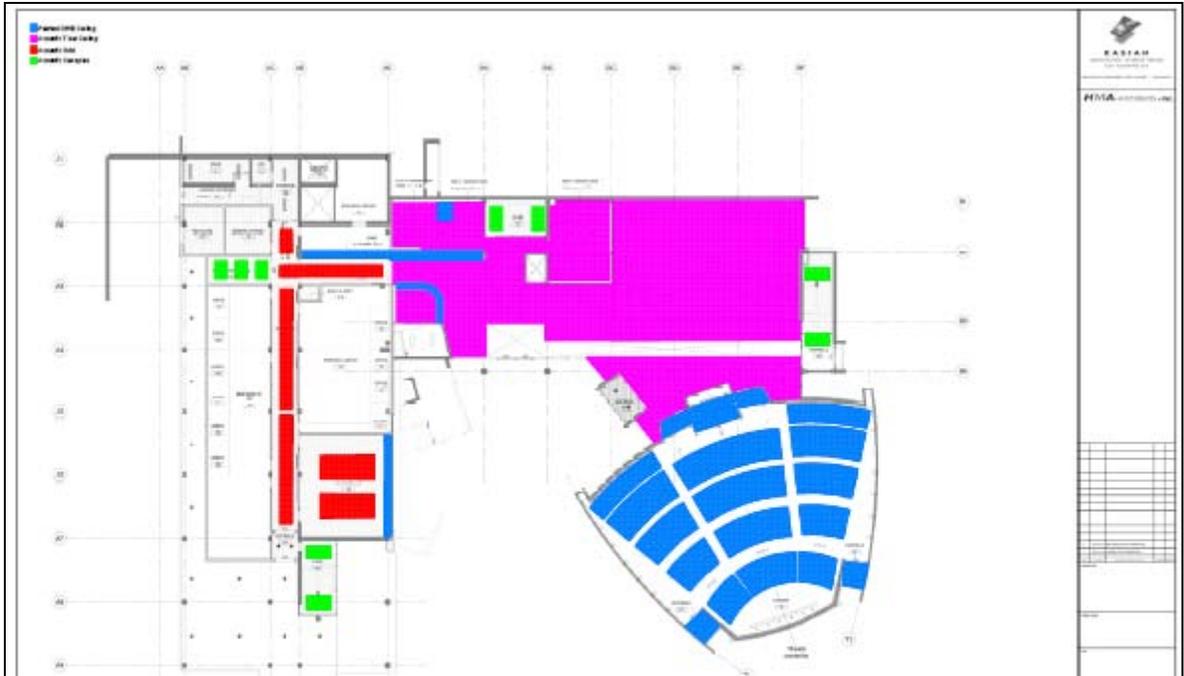


Figure 5.8 Colour-marking of different ceiling types and finishes in the reflected ceiling plan of Level 1

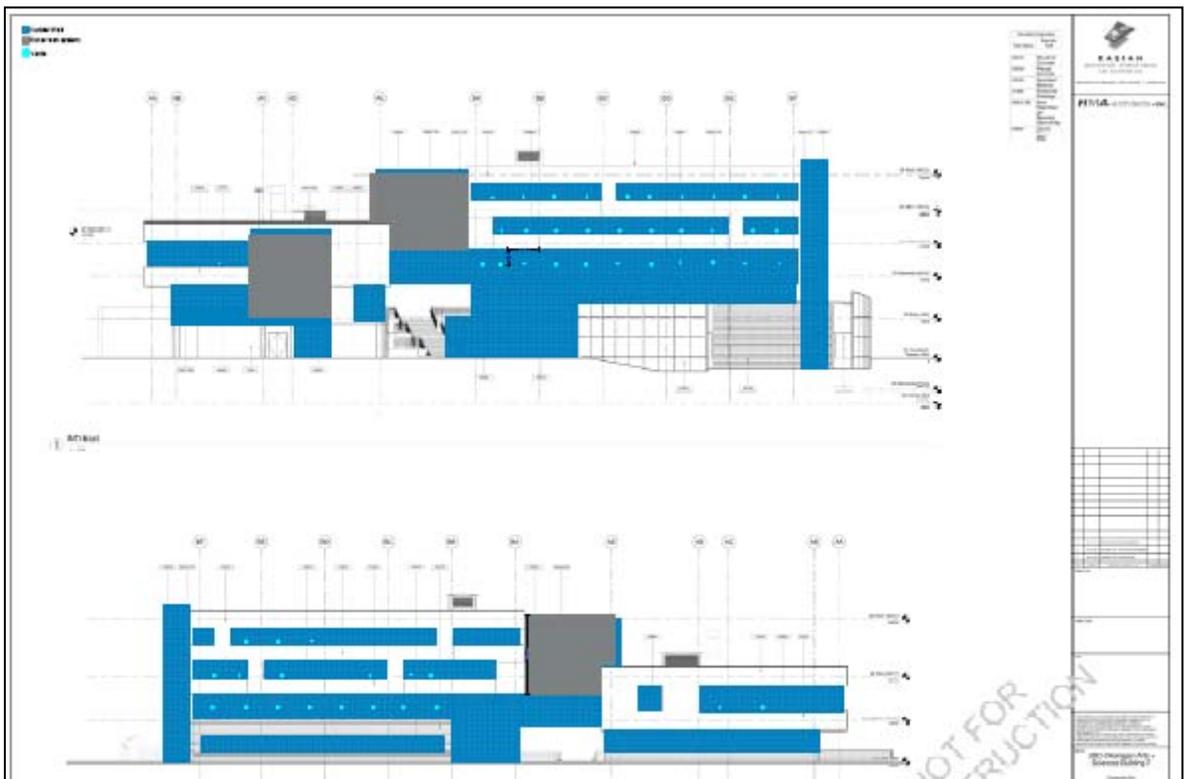


Figure 5.9 Colour-marking of curtain walls, sunsreen system, and vents, in elevation view

Using our approach, estimators need not mark conditions on each and every object of a design, but rather they can use the attributes of features to automatically apply appropriate conditions on components to identify useful information. This approach is significantly more efficient, more intelligent, and user-friendly. For example, estimators can categorize different types of walls (Figure 5.10 a), since it affects the specific items needed in the cost estimate and base crew productivity. They can organize walls based on wall heights (Figure 5.10 b), since it affects crew productivity and the methods required. They can identify the instances of wall connections with columns and exterior walls (Figure 5.10 c), since these require additional set-up and framing time. Note that the graphical representations depicted in Figure 10 are for illustration purposes only – the current implementation does not show results in the BIM interface.

State-of-the-art BIM tools do provide limited support to flexibly query component types. *Solibri Model Checker (SMC)* provides some support to identify instances of components. However, the parameters defined in *SMC* are narrow and the knowledge encoded is not adequate to support a wide range of construction-relevant queries dealing with components. *SMC* populates components with properties or attributes that are explicitly defined in the BIM application. *SMC* is however unable to identify component features with specific properties and relational properties that are important to practitioners. The lowest level that a user can work with on *SMC* is at the generic component level, such as wall, column, beam, duct, pipe, etc. However, it is not easy to specify a particular type of component based on its characteristics, such as material, dimensions, geometric shape, etc.

While *IFC* and *Revit* represent much information, many design conditions are either not explicitly represented or are missing. For instance, the designation of a ‘full height wall’ or ‘ceiling height wall’, and the shape of wall are not explicitly defined. The categorization of components, such as wall type (or any building component typing, in general, for that matter), in the *IFC* model or *Revit* are inadequate for construction use. *IFC* largely leaves the definition of a component type (e.g., wall type) completely to the modeling application, and to other model extension software, through type definitions (*IfcDefinesByType*) and through the definition of properties.

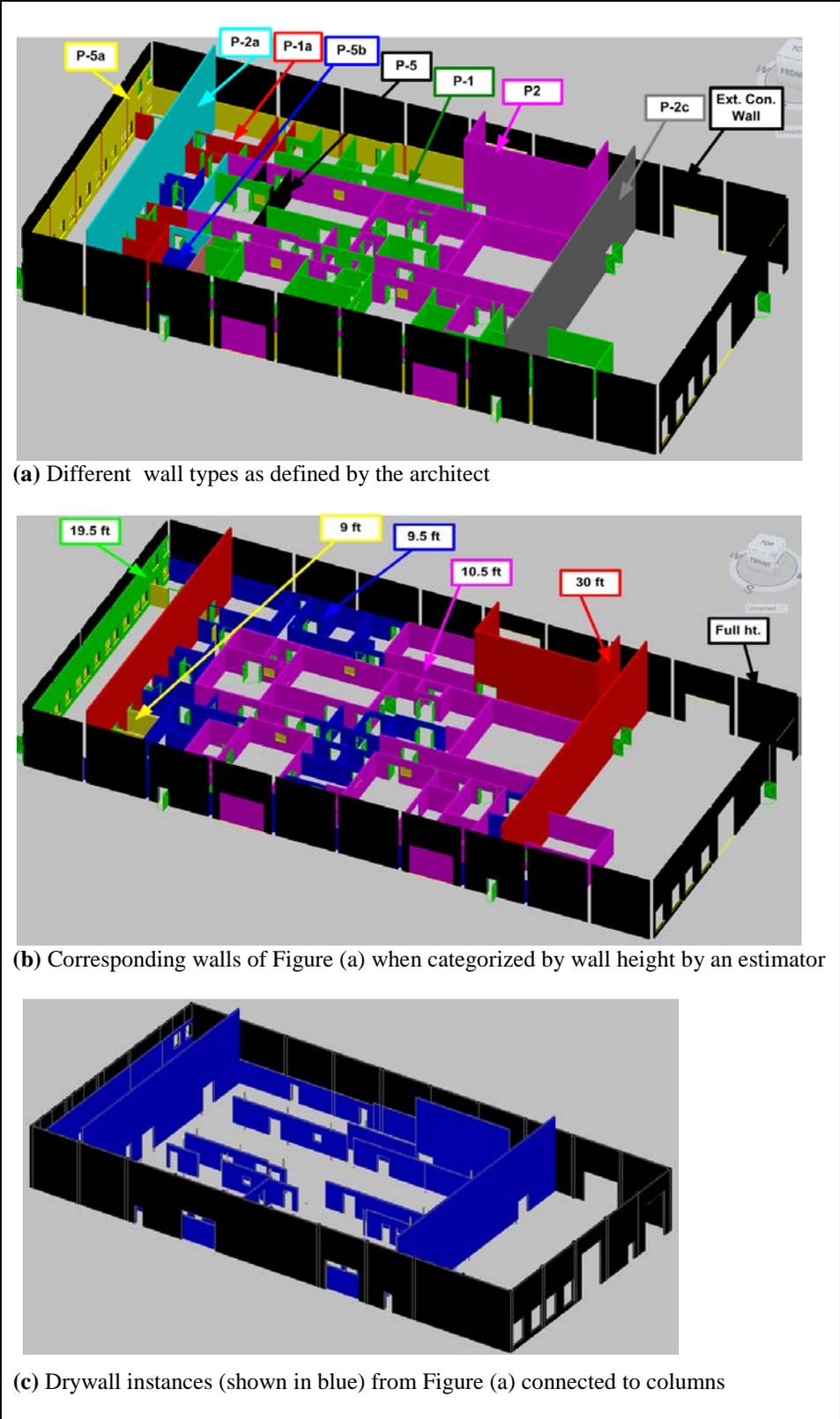


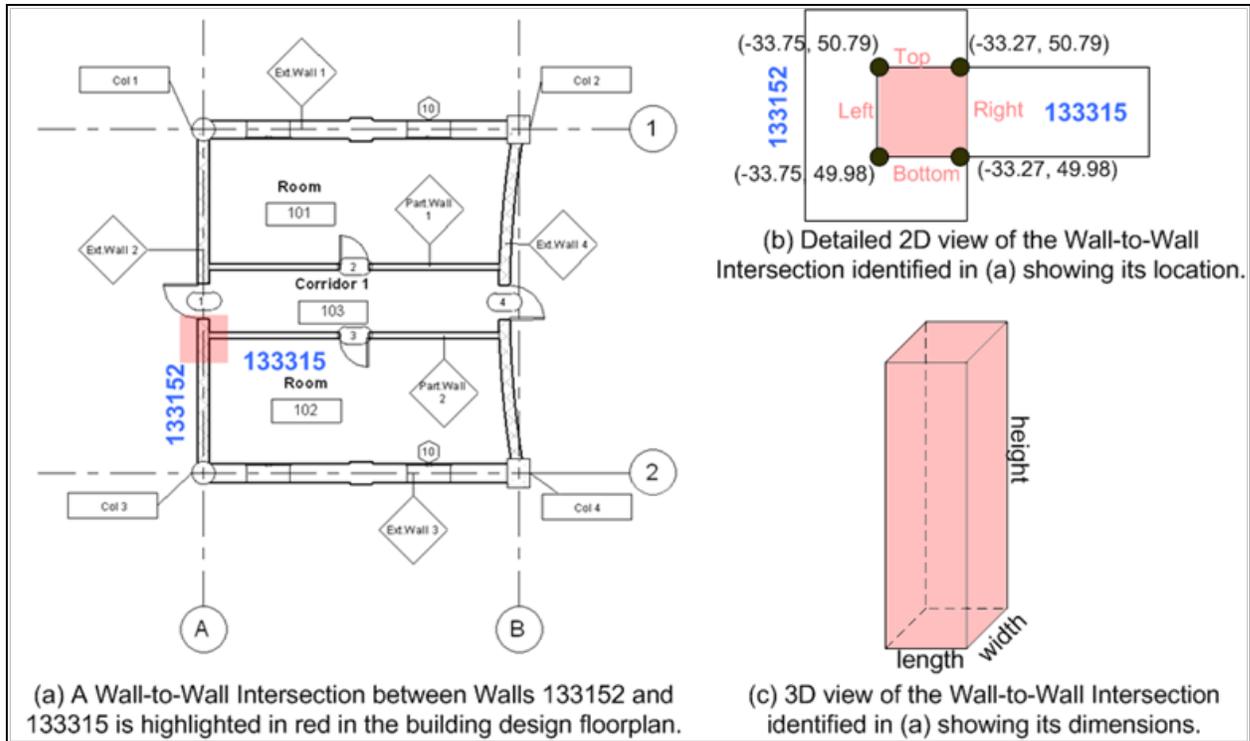
Figure 5.10 Grouping of walls using different criteria

*Revit* has some limited pre-defined wall types, which the user can edit to define instances of new wall types. Such user-defined typing, however, is intended mostly for verbal communication and for interpretation of design information among the users. The feature ontology in our research, through explicit definition and representation of attributes (e.g., wall properties and relationships), enables the user to define component types at query run time to provide design information in way that is tailored to construction practitioners..

### **5.2.3.2 Richer Support for Identifying Component Intersections**

The existence of component intersections and characteristics of intersecting components are of interest to construction practitioners because they impact constructability, construction productivity, and costs, as noted in Section 5.2.1.3. For example, the masonry and dry-wall contractors on the Chem-Bio Project at UBC were interested in identifying dry-wall to column, masonry wall to column, masonry wall to slab, and block wall to beam intersections. Similarly, the site superintendent on the EDC project wanted to know the intersections of concrete walls with other components, such as columns, pilasters, beams and slabs. The formwork contractor on the Discovery Place Project, Burnaby, BC, wanted to know the numbers of T- and L- intersections of the concrete walls.

This research provides support to find the intersections between components of interest according to different practitioners' preferences. Component intersections are not always explicit in BIM models, such as *Revit*, especially when the components are modeled by different professional domains, such as architects and structural engineers. While the *IFC* provides objectified relationships to connect components in some way, it is up to the *IFC*-complaint software, or the modellers, to explicitly define these conditions in the 3D model. Where an intersection exists, the 'intersection query' predicate implemented in our research, returns more detailed and insightful information about the intersecting region: its location (i.e., the boundary points of the region), dimensions (X, Y, Z dimensions), size, area and volume. Such detailed information is not explicitly defined in either *IFC* or *Revit*. Figure 5.11 shows the instance of a wall-to-wall intersection with some of the corresponding details highlighted.



**Figure 5.11** Example of a wall-to-wall intersection and the details provided by the ‘intersection query’ predicate

BIM analysis software provides excellent support to check for interferences and clashes of components, and helps to ensure the integrity of the 3D model. For example, *SMC* analyzes a BIM for its integrity and quality to ensure that there are no pertinent design errors, such as overlapping components. *NavisWorks*<sup>®</sup> provides a suite of applications that includes interference checking and clash detection of 3D models. The interference checking and clash detection module allows users to select the elements, or groupings of elements, that are to be checked against each other, to specify a tolerance value, and set the options for clash type and interference method. It thus helps practitioners to detect errors and coordinate designs, prior to construction. The clash detection mechanism in *Solibri* and *NavisWorks*<sup>®</sup> is meant to detect errors, or interferences, such as a duct passing through a beam, and is useful to ensure the quality of the given model. These tools also provide support to check for clearances, and find hard and soft conflicts. They, however, do not provide sufficient support to locate the intersections, which are not conflicts themselves, but rather genuine intersections of interest to practitioners.

The ‘intersection query’, as formalized in our research, enables the user to find a specific type of intersection, based on specific properties or characteristics of intersecting components (e.g., intersections between concrete block wall and slab, intersections between dry wall and round columns, etc.). As mentioned earlier, we also provide additional relevant information such as the size of the intersection, and the area and volume of the intersection. *NavisWorks*<sup>®</sup> provides an extensive set of options for selecting individual elements, or element groups, within the model. Objects can be selected at different levels (an individual object, a group, such as a block or cell, a layer or level, or an entire model), as defined by the CAD application in which the model was first created. Unless the building components in the authored CAD application(s) are clearly and properly differentiated into different categories, the clash detective in *NavisWorks*<sup>®</sup> does not have the intelligence nor the knowledge about building components, other than the geometry of CAD objects. From that perspective, *NavisWorks*<sup>®</sup> falls squarely into the CAD category, rather than the BIM category (Khemlani 2008). *SMC* is a state-of-the-art BIM-based application and contains this sophisticated level of intelligence on building components. It, however, does not allow the properties of a component to be filtered in order to specify the intersecting component type.

### **5.2.3.3 Richer Support for Identifying Component Penetrations and Openings and Other Insightful Information**

Building components contain many openings and penetrations. It is a major challenge to find different kinds of penetrations on walls, slabs, etc. on each floor of the building. Practitioners spend a significant amount of time manually identifying the location, size, type and other pertinent details, of these penetrations. For instance, the site superintendent on the EDC Project marked and manually-located penetrations, and other details, on several drawing sets. He marked slab penetrations by ducts (Figure 5.12), identified wall penetrations and openings for pipes, ducts, and cable trays (Figure 5.13), and highlighted openings in structural walls (Figure 5.14). One can imagine the amount of time and effort required to locate all penetrations in complex, or multi-storey buildings, particularly for projects with extensive mechanical, plumbing, electrical, and communication services.

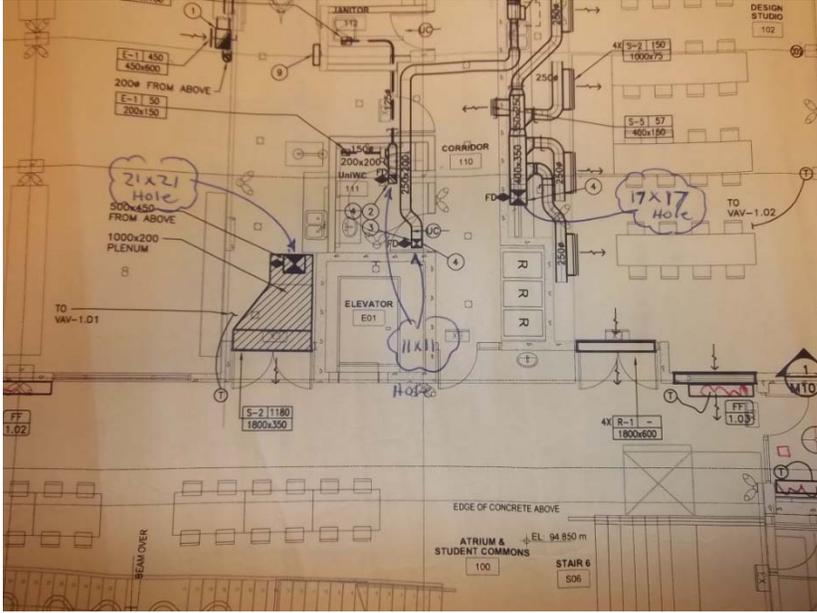


Figure 5.12 Site superintendent marking by hand, the size of duct penetrations, on a portion of the slab on the EDC project

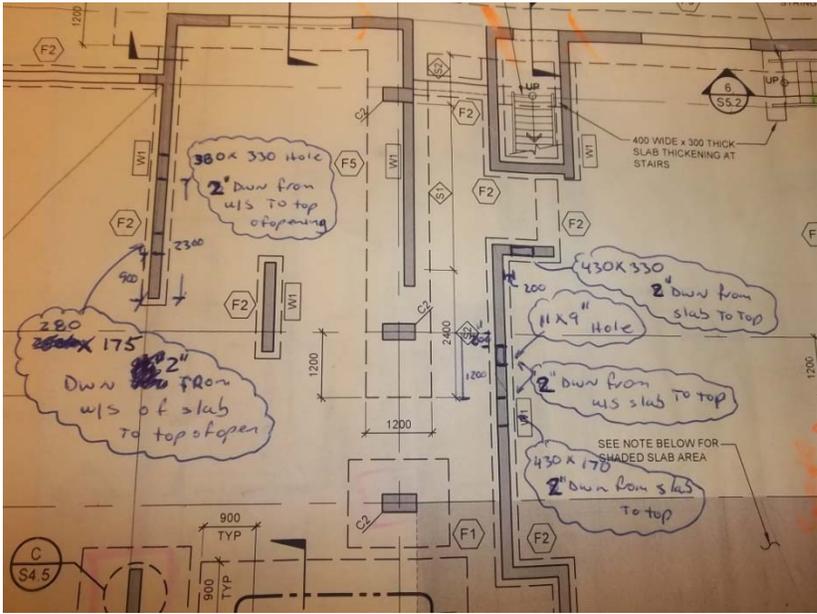
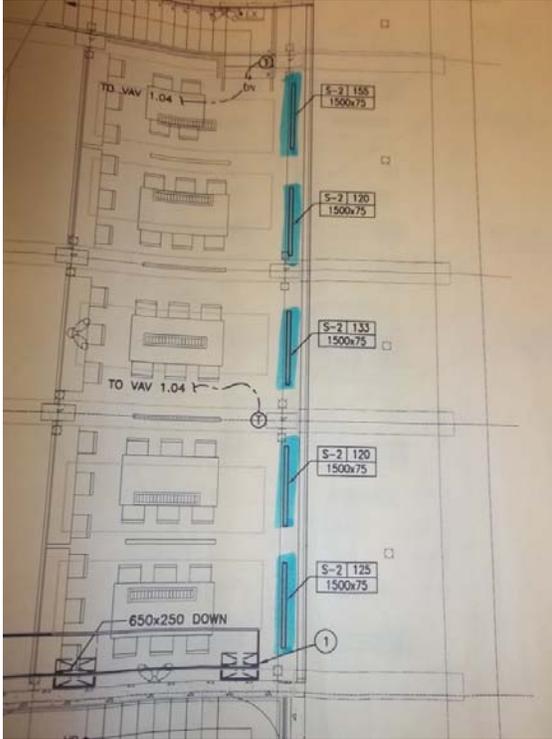


Figure 5.13 Site superintendent manually locating the openings on walls on the EDC project

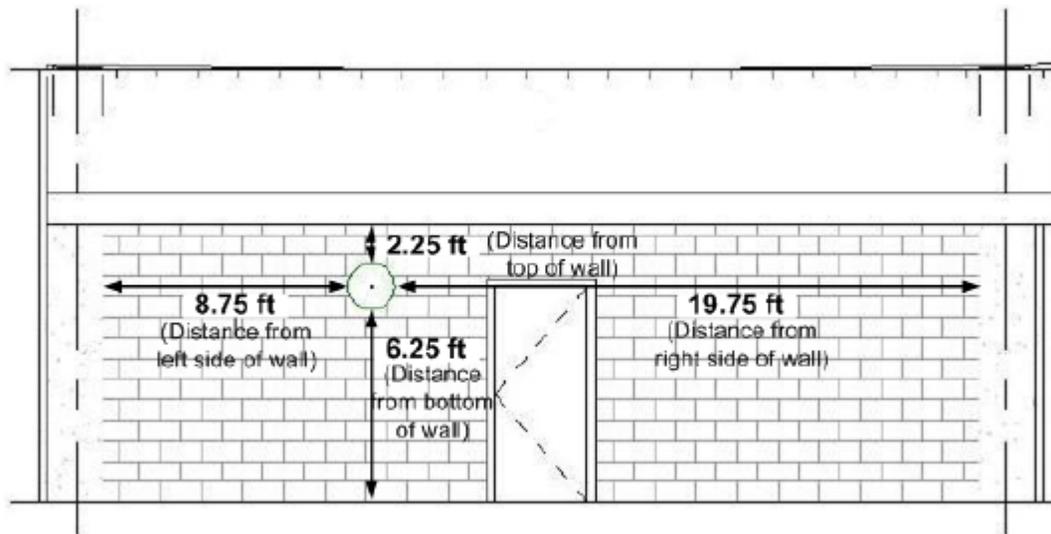


**Figure 5.14** Site superintendent marking the openings on structural drawings on the EDC project

Our research assists users to locate penetrations/openings and detailed information about them, from a given BIM. It allows users the flexibility to define the host component, where the penetrations (or openings) have to be found, specify the type of penetrating elements, (i.e., duct, pipe, conduit, etc.) and specify relevant properties of interest, such as the dimension (size, area, perimeter, etc.) of each penetration. Furthermore, users can specify the properties of the host component so as to locate the penetrations on a specific type of the host component. For instance, a drywall contractor can find penetrations on fire-rated walls; a roofing contractor can rapidly identify all existing roof penetrations, and their pertinent details.

The clash detection mechanisms in *NavisWorks*<sup>®</sup> and *SMC*, can be used to identify penetrations on building components. They do not, however, provide the support to specify the host component, or the properties of the host component, in the locating process. Moreover, these mechanisms do not differentiate between a conflict, an intersection or a penetration. In our system, the ‘penetration query’ employs the ‘intersection query’ predicate” to first determine if a penetration has occurred. Where a penetration exists, the

‘penetration query’ predicate provides detailed information about the penetration: its location (i.e., the corner points of the penetration region in the x-y plane, as well as its location on the wall), and its area and volume. The practitioners that we interviewed confirmed the importance of this type of comprehensive information. Figure 5.15 shows the instance of duct penetration on a wall, and its location, relative to the wall boundaries (or edges).



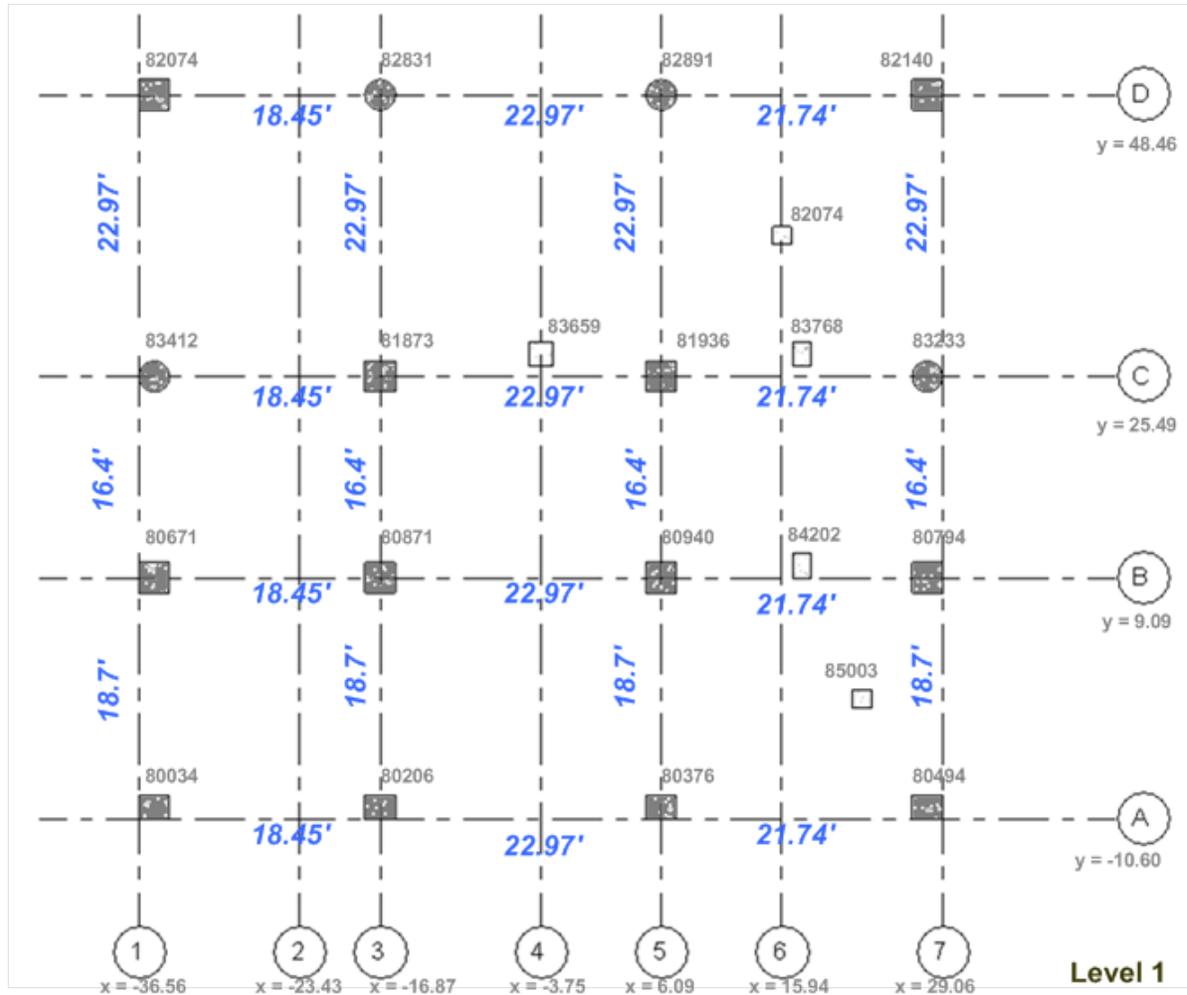
**Figure 5.15** Example of duct penetration, and its location in relation to the wall boundaries (sides) in one of the masonry walls on the Chem-Bio building, UBC

*SMC* is able to identify the vertical location of wall and window openings. However, it does not provide the support to identify the horizontal and vertical location of penetrations, and the horizontal location of openings on components. It provides basic location data of components, such as columns, with respect to a global origin and storey reference. Similarly, *NavisWorks*<sup>®</sup> provides almost no support for extracting location-specific information about penetrations.

#### 5.2.3.4 Richer Support for Identifying the Spacing of Features

Our research provides the support to reason about many types of spatial queries, such as feature spacing, alignment, and design uniformity. In this section, we examine some representative query examples and use them to illustrate some of the insights that can be gained by those using our system.

“Spacing” between components, or features, is an important concept that is useful for selecting appropriate construction methods, and can impact the constructability of a design (Fischer and Tatum 1997). Consistency, or uniformity, of the spacing of components, reduces field errors and costs in the building stage. We use an example of the spacing of proximate columns to illustrate how our system provides added value when querying about component spacing (Figure 5.16). The ‘spacing query’ first finds proximate columns, i.e., those that are adjacent to each other, and meet the following constraints. A target column is proximate to a source column (an input column), if it satisfies two criteria: (1) the column is aligned to one of the gridlines that the source column is aligned to; and (2) it is the closest such column along that gridline in a given direction (i.e., the positive or negative x- or y-direction from the input column). A column will have at least two, and at most, four proximate columns. An algorithm, known as “ray tracing,” a common technique used in graphics to render an object, identifies proximate column candidates (Webster 2010). Once the proximate columns for each column, on each level of the building, are identified, the ‘spacing query’ calculates the face-to-face (or clear) spacing, and centre-to-centre spacing, between a column and its proximate columns. Figure 5.16 presents the face- to-face spacing for all proximate, on-grid columns, on Level 1 of a building.



**Figure 5.16 Spacing of proximate, on-grid columns on Level 1**

The key concepts formalized in this research are also relevant and general enough for querying many other components, or features, such as the spacing of openings (or penetrations), walls, beams, and other features of the design.

### 5.2.3.5 Richer Support to Query for Feature Alignment

Another important aspect of spatial reasoning of features formalized in this research, is to query for alignment. The alignment query is used to identify the orientation and/or placement of the feature instances, with respect to a given criteria. The purpose is to find unaligned features, if any, present in a given design. The proper alignment of features, such as the column alignment in a floor, or from floor to floor, is crucial for the constructability of

a design and the installation of façade and curtain walls (Fischer and Tatum 1997; Allen and Iano 2002). An example alignment query for practitioners could be: “*Identify columns that do not align horizontally (i.e., in a particular floor) or vertically (i.e., from floor to floor).*” As described in Section 4.6.2.3 of Chapter 4, the alignment of columns can be assessed using different criteria. One of the criteria for defining the horizontal alignment of columns is based on the location of columns in the x-y plane (i.e., the floor), in a building and with respect to the grid lines. Based on the location data extracted from *Revit*, the ‘alignment query’ first determines if columns in a given design are on- or off-grid, and consequently, provides additional related information, such as the distance from related grid lines. For a building design based on a rectangular grid layout, a column is said to be “on-grid,” if it intersects two perpendicular gridlines (i.e., an x- and y-gridline). Figure 5.17 shows the sample results of on-grid and off-grid columns on part of a floor plan view. If a column is on-grid, the x- and y-gridlines to which the on-grid column is aligned, are reported in the query results.

If a column is not on-grid, it is described as “off-grid,” and the closest x- and y-gridlines to it, as well as the distance from them to the column, are reported. The query accepts as input, both the column and the axis (i.e., x or y), for which the closest gridline is desired, and reports the corresponding distance from the off-grid column. The identification of column locations in relation to gridlines, can thus provide insightful information about the horizontal alignment of columns.

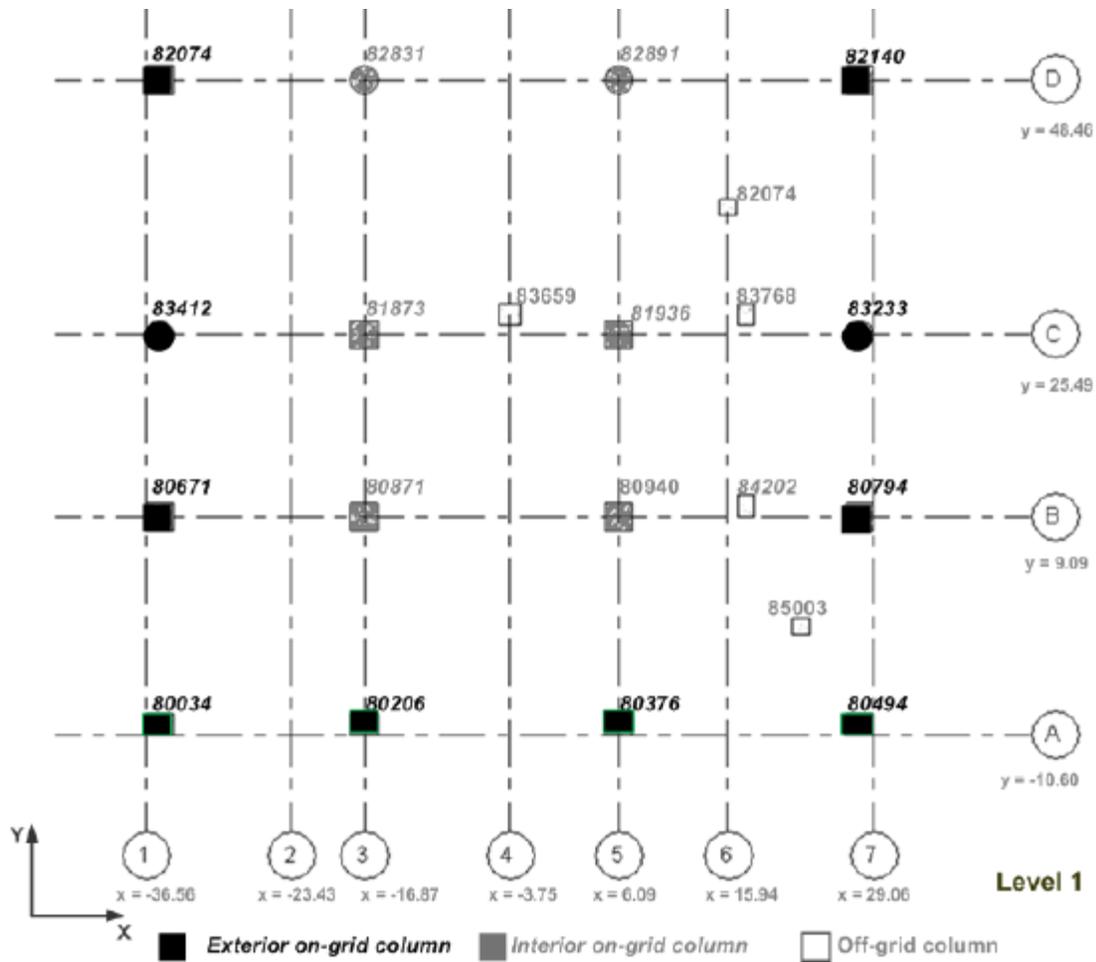


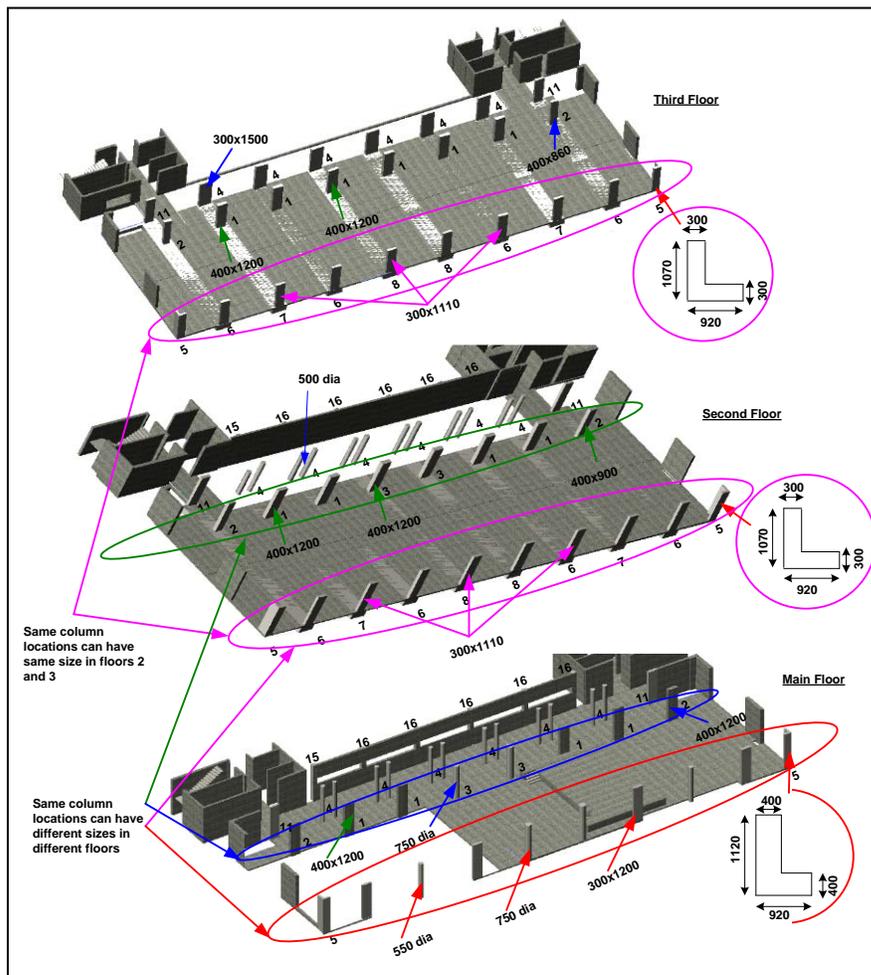
Figure 5.17 The designation of on-grid and off-grid columns on part of a floor plan

### 5.2.3.6 Richer Support for Identifying the Design Uniformity of Features

Assessing the design uniformity of features is promising, and yet a challenging aspect of the spatial reasoning of features, particularly across multiple floors (i.e., from floor to floor). Practitioners are especially interested in isolating where variations in design and uniformity exist. As indicated by the construction experts in our interviews, non-uniformity in column locations and shape/size from floor to floor are critical issues because they impact the constructability of design, and thereby, can prevent the use of more economical construction methods, reduce labour productivity, and ultimately increase costs. For instance, in the Chem-Bio Project at UBC, significant variation in the column locations, size, or shape, occurs from one floor to the next. Hanna and Sanvido (1990) also confirm that certain types

of formwork require uniformity of location, and size of columns, and can accommodate up to a 10% variation in column location and size.

It is a complex, time consuming, and error-prone process to manually identify the variation of different design features in each and every building project, particularly in complex, multi storey projects. As shown in Figure 5.18, one has to overlay different designs floor by floor, identify their locations, and the changes in their characteristics, such as size, shape, and other relevant properties.



**Figure 5.18** Variation in column size/shape, location in a floor, and from floor to floor in the first three floors of the Chem-Bio building, UBC

Querying for uniformity of design features from floor to floor is a step-by-step process, whereby the spatial location of features has to be determined first, followed by the

corresponding analysis of other feature properties, or parameters of concern. For instance, in order to analyze the variation in column location and size from floor to floor, the ‘uniformity’ query first locates the non-uniform columns from floor to floor. In other words, it identifies those columns in the base floor, (e.g., Level 1) whose location changes in the floor(s) above.

Once the uniformity of column locations is determined, the next step in the query process is to assess variation in other aspects (e.g., size, shape, or height) of columns. For the above-mentioned example, it identifies those columns in the base floor with change in size in the floor(s) above. This richer representation of uniformity can provide previously hidden and implicit information in the underlying BIM model in a significantly more meaningful and explicit way. This enables construction practitioners to better understand the design, and provides more support for their decision making process in the planning and execution of construction projects.

Current state-of-the-art tools provide very limited support for reasoning about spacing, alignment, and design uniformity of design features, which are valuable to construction practitioners.

#### **5.2.4 Conclusions from Validation Studies**

Our detailed interviews with the construction experts provides evidence that the knowledge formalized in this research is representative of reality in terms of representing design conditions that are relevant to construction. The interview results also show the generality of the knowledge formalized in this research, as four different experts representing different construction companies, domains, and viewpoints related to different projects, confirmed that the design conditions represented are important to them. They also provided information on the extent and criteria under which they are relevant.

Our retrospective analysis demonstrates that our research provides a more thorough level of support for identifying design conditions relevant to construction practitioners from a given BIM when compared to current state-of-the-art tools. Our descriptive and interpretative analysis illustrates that our approach is capable of providing richer, insightful and useful information and therefore could complement existing BIM analysis tools and/or work practice. These tests also demonstrate the flexibility of our approach to represent and identify information that is relevant to a broad range of practitioners. Our approach is also reasonably

generic and scalable to represent and identify a wide-range of construction-specific design information, both spatial and non-spatial, across different building components.

While spatial reasoning, in itself, is a challenging research issue, this research adds some novel work towards the spatial analysis of a BIM, and offers practical significance and implications in terms of leveraging BIM for construction purposes.

### **5.3 Practical Implications**

Our research has many practical implications to the construction industry. In addition, our approach could be useful to design analysis and facility management operations as well. Our approach not only automatically extracts and answers queries on construction-specific design information but also provides rich, actionable and insightful information about features of a design. It thus contributes towards improving efficiency, consistency and effectiveness in decision making process in construction by eliminating the tedious manual approach of identifying design features.

We anticipate that this work could have an impact in the following ways:

- i. Quickly identify cost incurring features of a design to support cost estimating.
- ii. Improve the consistency and accuracy of information extracted from a BIM.
- iii. Identify constructability issues prior to construction and provide constructability feedback to designers and owners.
- iv. Improve construction efficiency and productivity by improving the speed and ease with which construction information can be obtained.
- v. Support decision-making tasked related to purchasing and methods selection.
- vi. Provide information in a form that helps practitioners to manage the construction process and coordinate trades.
- vii. Help to make informed decisions and avoid costly mistakes in the layout and installation of components.
- viii. Make BIMs more accessible to construction.

#### 5.4 Limitations of the Current Research and Future Research Directions

This research developed a generic approach to automatically extract and query construction-relevant information from a BIM. It formalized construction knowledge and developed methods to identify meaningful construction information from a given design. We validated the content, soundness, intellectual merits, and generality of the developed research. This research, however, is not free from limitations. It is recommended that further studies build on this research and extend it to a number of areas. The following describes the limitations of this research, and suggested directions for further study.

1. The feature ontology developed in this research focuses on component and intersection features and their subtypes. It specifically formalizes design conditions for “wall” and “column” component features. The ontology formalizes the intersection feature as “component intersection,” “opening,” and “penetration,” and defines their attributes. While many of the design conditions formalized for “wall” and “column” features are applicable to other components, such as beams and slabs, further research is needed to formalize the specific attributes that are relevant to other types of component.
2. In this research, we do not represent or identify some detailed features, such as the existence of corbels, pilasters on walls, column capitals, column drop heads, slab bands, etc. Future research is needed to represent such features on components, and to develop mechanisms to identify them in a given BIM. One possible option is to define these features as “add-ons” and subtypes (e.g., “column add-ons”) and to define relationships according to the “component feature” in order to facilitate their extraction and querying.
3. In our current approach, we define feature types and attributes *a priori* and use specialized reasoning mechanisms to extract and query them. However, strategies to allow the user to rapidly define new feature types on-the-fly, and use the developed generic reasoning mechanisms to automatically instantiate their instances and attributes, remains a challenge. While the identification of explicit feature attributes

- (e.g., such as fire-rating and interior vs. exterior dimensions) can be generalized across different components, extracting implicit attributes (e.g., geometric shape(s) and other spatial concepts) can be highly challenging as the domain knowledge required for their extraction can vary from one component to the next.
4. The current implementation does not provide adequate support to visualize the extracted features in the corresponding 2D and 3D design views of a BIM. Further research is needed to effectively integrate the browsing and navigation of FBM data with corresponding objects in the BIM. Additional research work is also needed to formalize the structure and format of query results, or outputs, in order to provide more effective information support for practitioners.
  5. The success of this research also lies in the ability to enhance links to related construction management applications, such as cost estimating, construction scheduling, and BIM analysis tools (e.g., clash detection and design checker). Automated query support could be an integral part of BIM tools or specific construction applications that leverage a BIM.
  6. We extracted feature attributes (properties and relationships) and implemented multiple representative queries. However, there are number of queries which we have not yet implemented. Section 5.2.2 provides a comprehensive list containing relevant information and comparative analyses of our developed research with current state-of-the-art tools. In principle, more research is needed to provide the support for identifying information that is partially supported or not supported at all. For instance, we have not proposed methods that explicitly recognize different types of wall-to-wall intersections (e.g., T, L, end-to-end, and overlap methods) that practitioners would find relevant. In addition, the spacing query needs to be extended to identify the spacing between openings and penetrations. The identification of wall corners and types of wall corners (T or L) also requires further study. We have identified component finishes and related details as one of the more subtle, yet important features of a design that can impact cost, workmanship, and productivity in the

construction of the components. This issue needs further consideration, both in terms of its representation and to provide query support. The types of design conditions and queries formalized in this research are not complete, but the developed research does propose a sound and thorough basis for its extension by providing additional, more comprehensive support to BIM users.

7. The spatial reasoning about the features of a building is a complicated research effort. While we define generic attributes to assess the uniformity of design, we do not illustrate the specific knowledge as to how to assess uniformity for the location, spacing, and alignment of some features, such as uniformity in the location of wall and slab openings, uniformity in the spacing of columns, etc. Assessing the uniformity of features from floor to floor for spatial concepts, such as the spacing, location and alignment of features is very challenging, especially when there are discontinuities in the layout of floors and components. Further work is needed to address these challenges in assessing the uniformity of design features. The application of spatial clustering could provide support for the spatial analysis of features, and consequently, provide BIM users with an improved understanding of the spatial distribution of components and their variation.

## References

- Adachi, Y. (2003). Overview of partial model query language. *Proceedings of the 10<sup>th</sup> ISPE International Conference on Concurrent Engineering Research Applications*, pp. 549-555.
- Akbas, R. (2004). *Geometry-Based Modeling and Simulation of Construction Processes*. Technical Report No. 151, Centre for Integrated Facility Engineering, Stanford University, CA.
- Akinci, B., Fischer M., Levitt, R., and Carlson, B. (2002). Formalization and automation of time-space conflict analysis. *Journal of Computing in Civil Engineering*, 16(2), 24-134
- Allen, E., and Iano, J. (2002). *The Architect's Studio Companion: Rules of Thumb for Preliminary Design*, 3<sup>rd</sup> Edition, Willey, NY.
- ASCE Construction Division (1991). Constructability and constructability programs: White paper. *Journal of Construction Engineering and Management*, 117(1), 67-89.
- Augenbroe, G. (1994). An overview of the COMBINE project. *Proceeding of the First European Conference on Product and Process Modeling in the Building Industry*, Dresden, Germany, pp. 547–554.
- Autodesk. (2009). *Revit 2009 API Developer's Guide Version 1.0*.
- Autodesk. (2010). *Autodesk Navisworks Products*. Homepage: <http://usa.autodesk.com/navisworks/>
- Bisharat, K. A. (2004). *Construction Graphics: A Practical Guide to Interpreting Working Drawings*, John Wiley & Sons Inc.
- Björk, B.-C. (1989). Basic structure of a building product model. *Computer-aided Design*, 21(2), 71–78.
- Boeke, E. H. (1990). Design for constructability: A contractor's view. *Concrete Construction Magazine*, issue 35. Retrieved from: <http://www.structuremag.org/>
- Booch, G., Rumbaugh, J., and Jacobson, I. (1999). *The Unified Modeling Language User Guide*, Addison-Wesley, MA.
- Borrman, A., van Treeck, C., and Rank, E. (2006). Towards a 3D spatial query languages for building information models. *Proceedings of the Joint International Conference of*

- Computing and Decision Making in Civil and Building Engineering (ICCCBE-XI)*, Montreal, Canada.
- Borrmann, A., and Rank, E. (2009). Topological analysis of 3D building models using a spatial query language. *Advanced Engineering Informatics*, 23(4), 370-385.
- Boukamp, F. (2006). Modeling of and Reasoning about Construction Specifications to Support Automated Defect Detection, Ph.D. Thesis, Carnegie Mellon University.
- Brank, J., Grobelnik, M., and Mladenic, D. (2005). A survey of ontology evaluation techniques. *Proceedings of the Conference on Data Mining and Data Warehouses*, Ljubljana, Slovenia.
- Building and Construction Authority (BCA). (2001). *Guide to the Buildable Design Appraisal System*, Singapore.
- buildingSMART Alliance & The Open Geospatial Consortium, Inc. (2008). *A Request for Technology in Support of an AECCO Testbed*, Reference No. bSa/OGC RFT 08-001, Indiana, USA.
- Burkhart, A. (1987). Repeating formwork greatly reduces costs. *Concrete Construction Magazine*. Retrieved from: <http://www.structuremag.org/>
- Carrico, M. A., Girard, J. E., and Jones, J. P. (1989). *Building Knowledge Systems: Developing and Managing Rule-Based Applications*, McGraw-Hill Book Company.
- Chen, P.-H., Cui, L., Wan, C., Yang, Q., Ting, S. K., and Tiong, R. L. K. (2005). Implementation of IFC-based web server for collaborative building design between architects and structural engineers. *Automation in Construction*, 14, 115-128.
- Cherneff, J., Logcher, R., Connor, J., and Patrikalakis, N. (1992). Knowledge-based interpretation of architectural drawings. *Research in Engineering Design*, 3(4), 195-210.
- CII (1986). *Constructability: A Primer*. Publication 3-1, Construction Industry Institute, Austin, Texas.
- Clementini, E., and Di Felice, P. (1995) A comparison of methods for representing topological relationships, *Information Sciences*, 3(3), 149-178.
- Clementini, E., and Di Felice, P. (1996). A model for representing topological relationships between complex geometric features in spatial databases. *Information Sciences*, 90(1-4), 121-136.

- Colquhoun, G. J., Baines, R. W., and Crossley, R. (1993). A state of the art review of IDEF0, *International Journal of Computer Integrated Manufacturing*, 6, 252-264.
- Construction Specifications Institute (CSI). (2008). *Standards and Formats*. Homepage: <http://www.csinet.org/>
- Cunningham, J. J., and Dixon, J. R. (1988). Designing with features: the origin of features. *ASME Computers in Engineering Conference*, San Francisco, CA, pp. 237-243.
- Darwiche, A., Levitt, R. E., and Hayes-Roth, B. (1988). OARPLAN. Generating project plans in a blackboard system by reasoning about objects, actions, and resources. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 2(3), 169–81.
- Dixon, J., and Poli, C. (1995). *Engineering Design and Design for Manufacturing: A Structured Approach*, Field Stone Publishers, MA.
- Dym, C. L., Henchey, R. P., and Gonick, S. (1988). A knowledge-based system for automated architectural code checking. *Computer-Aided Design*, 20(3):137–45.
- Eastman, C. M., Teicholz P. Sacks, R., and Liston, K. (2008). *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, designers, Engineers, and Contractors*, Wiley and & Sons, Inc.
- Egenhofer, M., and Franzosa, R. (1991). Point-set topological spatial relations. *International Journal of Geographical information Systems*, 5(2), 161-174.
- Ekholm, A. (2002). Principles for classification of properties of construction objects. In (Agger K., Christiansson P. and Howard R. Eds), *Distributing Knowledge in Building - CIB W78 Conference 2002*. Aarhus School of Architecture, Aarhus, Denmark.
- El-Diraby, T. A., Lima, C., and Feis, B. (2005). Domain taxonomy for construction concepts: Toward a formal ontology for construction knowledge. *Journal of Computing in Civil Engineering*, 19(4), 394-406.
- Fellows, R., and Liu, A. (2003). *Research Methods for Construction*, 2<sup>nd</sup> Edition, Blackwell Publishing Company.
- Fenves, S. J. (2002). *A Core Product Model for Representing Design Information*, NISTIR 6736, National Institute for Standards and Technology, Gaithersburg MD.

- Fischer, M. (1991). *Constructability Input to Preliminary Design of Reinforced Concrete Structures*. Technical Report, Centre for Integrated Facility Eng., Stanford University, CA.
- Fischer, M. (2006). Formalizing construction knowledge for concurrent performance-based design. *Intelligent Computing in Engineering and Architecture*, Springer-Verlag Berlin, pp. 186-205
- Fischer, M., and Tatum, C. B. (1997). Characteristics of design-relevant constructability knowledge. *Journal of Construction Engineering and Management*, 123(3), 253-260.
- Galambos, J. A. (1986). Knowledge structures for common activities. In: J. A. Galambos, R. P. Abelson, J. B. Black (Eds.), *Knowledge Structures*, Lawrence Erlbaum Associates, NJ, pp. 21-47.
- Genesereth, M. R., and Nilsson, N. J. 1987. *Logical Foundations of Artificial Intelligence*, San Mateo, CA; Morgan Kaufmann.
- Gielingh, W. (1988). General AEC reference model (GARM). In: P. Christiansson, H. Karlsson (eds.) *Conceptual Modeling of Buildings, CIB Proceedings*, Publication 126, Lund, Sweden, pp. 165-178.
- Glavinich, T. E. (1995). Improving constructability during design phase. *Journal of Architectural Engineering*, 1(2), 73-76.
- Graphisoft. (2004). *The New Heroes in the Building Industry*, Graphisoft Envisions Newsletter.
- Gruber, T. (1995). Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human – Computer Studies*, 43, 907–928.
- Han, C. S., Law, K., and Kunz, J. (2000). *Computer Models and Methods for a Disabled Access Analysis Design Environment*. CIFE Technical Report No. 123, Stanford University.
- Hanna, A. S., and Sanvido, V. E. (1990). Interactive vertical formwork selection. *Concrete International Design and Construction*, 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*, 118(1), 179-198.

- Hardin, B. (2009). *BIM and Construction Management: Proven Tools, Methods, and Workflows*, John Wiley & Sons, Inc.
- Haymaker, J., Fischer, M., Kunz, J., and Suter, B. (2004). Engineering test cases to motivate the formalization of an AEC project model as a directed acyclic graph of views and dependencies. *ITcon*, 9, 419-441. Available online at: <http://www.itcon.org/2004/30/>
- Howard, R., and Bjork, B. (2008). Building information modeling - Experts' views on standardisation and industry deployment. *Advanced Engineering Informatics*, 22, 271-280.
- Innovaya Inc. (2009). Homepage: <http://www.innovaya.com> [Accessed April 10, 2009]
- International Alliance for Interoperability (IAI). (2000). *Industry Foundation Classes - Release 2x: IFC Technical Guide*.
- International Alliance for Interoperability (IAI). (2010). Available at: <http://buildingsmart-tech.org/>
- Isikdag, U., and Zlatanova, S. (2009). A SWOT analysis on the implementation of building information models within the geospatial environment. In: A. Krek, M. Rumor, S. Zlatanova and E. M. Fendel (Eds.), *Urban and Regional Data Management*, CRC Press, The Netherlands, pp. 15-30.
- ISO 12006-2. (2001). *Building Construction – Organization of Information about Construction Works – Part 2: Framework for Classification of Information*. International Organisation for Standardisation.
- Jayapandian, M., and Jagadish, H. V. (2008). Expressive query specification through form customization, *Proc. of the 11<sup>th</sup> International Conference on Extending Database Technology (EDBT)*, Nantes, France, pp. 416-427.
- Katranuschkov, P., Gehre, A., and Scherer, R. J. (2003). An ontology framework to access IFC model data, *ITcon*, 8, 413-437. Available online at: <http://www.itcon.org/2003/29/>
- Khanzode, A., Fischer, M., and Reed, D. (2008). Benefits and lessons learned of implementing building virtual design and construction (VDC) technologies for coordination of mechanical, electrical, and plumbing (MEP) systems on a large healthcare project. *ITcon*, 13, 324-342. Retrieved from <http://www.itcon.org/2008/22>.

- Khemlani, L. (2004). BIM goes mainstream: Graphisoft's new virtual construction Solutions. *ACEbytes Newsletter #15*. Available online at:  
[http://www.aecbytes.com/review/newsletter/2004/issue\\_15.html](http://www.aecbytes.com/review/newsletter/2004/issue_15.html).
- Khemlani, L. (2008). Autodesk NavisWorks 2009. *AECbytes Product Review*. Available online at: <http://www.aecbytes.com/review/2008/NavisWorks2009.html>.
- Korman, T. M., Fischer, M. A., and Tatum, C. B. (2003). Knowledge and reasoning for MEP coordination. *Journal of Construction Engineering and Management*, 129(6), 627-634.
- Kristiina, S., Mäkelä, T., and Kiviniemi, M. (2009). BIM-based site layout and safety planning. *1<sup>st</sup> International Conference on Improving Construction and Use through Integrated Design Solutions, CIB IDS*. Espoo, Finland, pp. 125-140.
- Kunz, J. C., Jin, Y., Levitt, R. E., Lin, S.-D., and Teicholz, P. M. (1996). Support for integrated value-based maintenance planning. *IEEE Expert: Intelligent Systems and their Applications*, 11(4), 35-44.
- Laitinen, J., (1998). Model Based Construction Process Management, Ph.D. thesis, Royal Institute of Technology, Stockholm.
- Lawson, B., and Roberts, S. (1991). Modes and features: the organization of data in CAD supporting the early phase of design. *Design Studies*, 12(2), 102-108.
- Leedy, P. D., and Ormrod, J. E. (2005). *Practical Research: Planning and Design*, 8th ed., Prentice-Hall.
- Li, M. (2009). Diagnosing Construction Performance by Using Causal Models, Ph.D. Thesis, Dept. of Civil Engineering, University of British Columbia, Vancouver, Canada.
- Liebich, T. (2009). *IFC 2x Edition 3 Model Implementation Guide*.
- Lou, K., Subramaniam J., Iyer, N., Kalyanaraman Y., Prabhakar S., and Ramani K. (2003). A reconfigurable, intelligent 3D engineering shape search system part II: database indexing, retrieval, and clustering. *ASME DETC 2003 Computers and Information in Engineering (CIE) Conference*, Chicago, IL.
- Lucko, G., and Rojas, E. M. (2010). Research validation in the construction domain: challenges and opportunities. *Journal of Construction Engineering and Management* 136(1), 127-135.

- Mäntylä, M., Nau, D., and Shah, J. (1996). Challenges in feature-based manufacturing research. *Communications of the ACM*, 39(2), 77-85.
- McGraw-Hill Construction. (2008). *Building Information Modeling: Transforming Design and Construction to Achieve Greater Industry Productivity*. SmartMarket Report, McGraw-Hill Construction, NY.
- McKinney, K., Fischer, M., and Kunz, J. (1998). Visualization of construction planning information. *Proceedings of Intelligent user Interfaces*, San Francisco, CA.
- Navon, R., Shapira, A., and Shechori, Y. (2000). Automated rebar constructability diagnosis. *Journal of Construction Engineering and Management*, 126(5), 389-397.
- Neuman, W. L. (2006). *Social Research Methods: Qualitative and Quantitative Approaches*, Sixth Edition, Pearson Education Inc.
- Nguyen, T., and Oloufa, A. A. (2002). Spatial information: classification and applications in building design. *Computer-Aided Civil and Infrastructure Engineering*, 17, 246-255.
- Noort, A., Bidarra, R., and Bronsvort, W. F., (2000). Satisfying interaction constraints. In: Cugini, U., and Wozny, M., (Eds.), *Seventh IFIP WG 5.2 Workshop on Geometric Modeling: Fundamentals and Applications - GEO7*, pp. 51– 64. University of Parma, Parma.
- Noy, N. F., and McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology. Retrieved from:  
[http://protege.stanford.edu/publications/ontology\\_development/ontology101.pdf](http://protege.stanford.edu/publications/ontology_development/ontology101.pdf)
- O'Connor, J. T., Rusch, S. E., and Schulz, M. J. (1987). Constructability concepts for engineering and procurement. *Journal of Construction Engineering and Management*, 113(2), 235-248.
- On Centre Software (2009). *On-Screen Take-off*. Homepage: <http://www.oncenter.com>  
 [Accessed February 11, 2009]
- Open Geospatial Consortium (OGC). (2010). *OGC® Standards and Specifications*. Available online at: <http://www.opengeospatial.org/standards/> [Accessed Jan 10, 2010]
- Overall Construction Classification System (OCCS). (2008). Homepage: <http://www.omniclass.org> [Accessed Nov 22, 2008]

- Paulson, B. C., Jr. (1976). Designing to reduce construction costs, *Journal of Construction Division*, 102(4), 587-592.
- Protégé. (2008). Homepage: <http://protege.stanford.edu/> [Accessed Jan 8, 2008]
- RAPID - Robust and Accurate Polygon Interference Detection. (2008). Homepage: <http://gamma.cs.unc.edu/OBB/> [Accessed Feb 6, 2008]
- Rosen, W. R., Dixon, J. R., and Dong, X. (1991). A methodology for conversions of feature-based representations. *Design Theory and Methodology*, ASME, 31, 45-51.
- RSMeans Inc. (2004). *RSMeans Building Construction Data*. 62nd annual edition, R. S. Means Co., Inc, Kingston, MA.
- Rush, R. D. (1986). *The Building Systems Integration Handbook*, The American Institute of Architects, Reed Publishing, Inc., Stoneham, MA.
- Russell, A., Staub-French, S., Tran, N., and Wong, W. (2009). Visualizing high-rise building construction strategies using linear scheduling and 4D CAD. *Automation in Construction*, 18(2), 219-236.
- Sanders, S. R., and Thomas, H. R. (1991). Factors affecting masonry labour productivity. *Journal of Construction Engineering and Management*, 117(4), 626-644.
- Shah, J. J. (1991). Assessment of features technology. *Journal of Computer-Aided Design*, 23(5), 331-341.
- Shah, J. J., and Rogers, M. T. (1988). Functional requirements and conceptual design of the feature-based modeling system. *Computer-Aided Engineering Journal*, 9-15.
- Shah, J., and Mäntylä, M., (1995). *Parametric and Feature-Based CAD/CAM: Concepts, Techniques and Applications*, Wiley & Sons Inc., New York,
- Skibniewski, M., Arciszewski, T., and Lueprasert, K. (1997). Constructability analysis: a machine learning approach. *Journal of Computing in Civil Engineering*, 2(1), 8-17.
- Smith, G. R., and Hanna, A. S. (1993). Factors influencing formwork productivity. *Canadian Journal of Civil Engineering*, 20(1), 144-153.
- Solibri Model Checker (SMC) Inc. (2010). Homepage: <http://www.solibri.com/> [Accessed August 15, 2010, April 2006, Feb 2008]
- Staub-French, S., and Nepal, M. P. (2007). Reasoning about component similarity in building product models from the construction perspective. *Automation in Construction*, 17(1), 11-21.

- Staub-French, S., Fischer, M., Kunz, J., Ishii, K., and Paulson, B. (2003). A Feature ontology to support construction cost estimating. *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing*, 17, 133-154.
- Stiny, G. (1985). Computing with form and meaning in architecture. *Journal of Architectural Education*, 39(1), 7-19.
- Tabesh, R., and Staub-French, S. (2006). Modeling and coordinating building systems in three dimensions: A case study. *Canadian Journal of Civil Engineering*, 33(12), 1490–1504.
- Thomas, H. R., and Sakarcan, A. S. (1994). Forecasting labour productivity using factor model. *Journal of Construction Engineering and Management*, 120(1), 228–239.
- Thomas, H. R., and Zavrski, I. (1999). Construction baseline productivity: Theory and practice. *Journal of Construction Engineering and Management*, 125(5), 295-303.
- Tolman, F., Böhms, M., Lima, C., Van Rees, R., Fleuren, J., and Stephens, J. (2001). Econstruct: Expectations, solutions, and results, *ITCON*, 6, 175-197. Available online at: <http://www.itcon.org/2001/13/>.
- Turk, Z. (2001). Phenomenological foundations of conceptual product modeling in architecture, engineering and construction. *Artificial Intelligence in Engineering*, 15(2), 83-92.
- Turner, J. (1990). *AEC Building Systems Model, ISO TC184/SC4/WG1*, Doc. N363. Working paper, University of Michigan.
- Udaipurwala, A., and Russell, A. D. (2005). Hierarchical clustering for interpretation of spatial configuration. *Proceedings of the 2005 ASCE Construction Research Congress*, San Diego, CA, pp. 556-571.
- Ugwu, O. O., Anumba C. J., and Thorpe, A. (2005). Ontological foundations for agent support in constructability assessment of steel structures - a case study. *Automation in Construction*, 14, 99-114.
- Ugwu, O. O., Anumba, C. J., and Thorpe, A. (2004). The development of cognitive models for constructability assessment in steel frame structures. *Advances in Engineering Software*, 35, 191-203.
- University of British Columbia (UBC). (2011). *The Building*. Available online at: <http://www.sustain.ubc.ca/hubs/cirs/building> [Accessed Jan 10 2011]

- Uschold, M., King M., Moralee, S., and, Zorgios, Y. (1998). The enterprise ontology. *The Knowledge Engineering Review*, 13(1), 31-89.
- van Leeuwen, J. P. (1999). Modeling Architectural Design Information by Features, Ph.D. Thesis, Eindhoven University of Technology, Eindhoven, The Netherlands.
- van Leeuwen, J. P., and Wagter, H. (1997). Architectural design by features. *Proceedings of the 7th Intl. Conference on Computer Aided Architectural Design Futures*, Munich, Germany, pp. 97-115.
- Watson A. (1995). To product models and beyond. In: P. Brandon and M. Betts (eds.) *Integrated construction information*, London: E&F Spon.
- Webster, A. (2010). A Semantic Spatial Interoperability Framework: A Case Study in the Architecture, Engineering and Construction (AEC) Domain, Master's Thesis, Department of Computer Science, University of British Columbia, Vancouver, Canada.
- Wilson, B. (1984). *Systems: Concepts, Methodologies and Applications*, Wiley, Chichester.
- Wissam, H., Alkass, S., and Zayed, T. (2009). Constructability assessment using BIM/4D CAD simulation model. *AACE International 53rd Annual Meeting*, Seattle WA.
- Woestenenk, K., van Rees, R., Lima, C., Stephens, J., and Bonsma, P. (2000). *bcTaxonomy*. Report, IST-1999-10303-D501, European Commission, Brussels, Belgium.
- Wong, A., and Sriram, D. (1993). SHARED: An information model for cooperative product development. *Research in Engineering Design*, 5(1), 21-39.
- World Wide Web Consortium (W3C). (2010). XQuery 1.0: An XML Query Language. Homepage: [www.w3.org/TR/xquery](http://www.w3.org/TR/xquery) [Accessed May 5, 2010, 2007].
- Yin, R. K. (2009). *Case Study Research: Design and Methods*, Fourth Edition, Sage Inc., CA.
- Zamanian, M. K., and Fenves, S. J. (1994). A referential scheme for modeling and identifying spatial attributes of entities in constructed facilities. *Research in Engineering Design*, 6,142-168
- Zamanian, M. K., Fenves, S. J., Thewalt, C. R., and Finger, S. (1991). A feature-based approach to structural design. *Engineering with Computers*, 7, 1-9.

Zhang, J. (2008). Evaluations of XML Standards for Actual Applications, Master's Thesis, Department of Computer Science, University of British Columbia, Vancouver, Canada.

Zhang, J., Webster, A., Lawrence, M., Nepal, M., Pottinger, R., Staub-French, S., and Tory, M. (2011). Improving the usability of standard schemas. *Information Systems*, 36, 209-221.

## Appendix A: Numbered References Cited in Tables 2.3 and 5.8 to 5.13

- [1] Fischer, M., and Tatum, C. B. (1997). Characteristics of design-relevant constructability knowledge. *Journal of Construction Engineering and Management*, 123(3), 253-260.
- [2] Boeke, E. H. (1990). Design for constructability: A contractor's view. *Concrete Construction Magazine*, issue 35. Retrieved from: <http://www.structuremag.org/>
- [3] Burkhart, A. (1987). Repeating formwork greatly reduces costs. *Concrete Construction Magazine*. Retrieved from: <http://www.structuremag.org/>
- [4] Smith, G. R., and Hanna, A. S. (1993). Factors influencing formwork productivity. *Canadian Journal of Civil Engineering*, 20(1), 144-153.
- [5] Fischer, M. (1991). *Constructability Input to Preliminary Design of Reinforced Concrete Structures*. Technical Report, Centre for Integrated Facility Engineering, Stanford University, CA.
- [6] Sanders, S. R., and Thomas, H. R. (1991). Factors affecting masonry labour productivity. *Journal of Construction Engineering and Management*, 117(4), 626-644.
- [7] Udaipurwala, A., and Russell, A. D. (2005). Hierarchical clustering for interpretation of spatial configuration. *Proceedings of the 2005 ASCE Construction Research Congress*, San Diego, CA, pp. 556-571.
- [8] Building and Construction Authority (BCA). (2001). *Guide to the Buildable Design Appraisal System*, Singapore.
- [9] Nguyen, T., and Oloufa, A. A. (2002). Spatial information: classification and applications in building design. *Computer-Aided Civil and Infrastructure Engineering*, 17, 246-255.
- [10] Shaked, O., and Warszawski. (1995). A knowledge-based system for construction planning of high-rise buildings, *Journal of Construction Engineering and Management*, 21(2), 172-182.
- [11] O'Connor, J. T., Rusch, S. E., and Schulz, M. J. (1987). Constructability concepts for engineering and procurement. *Journal of Construction Engineering and Management*, 113(2), 235-248.

- [12] Thomas, H. R., and Zavrski, I. (1999). Construction baseline productivity: theory and practice. *Journal of Construction Engineering and Management*, 125(5), 295-303
- [13] Ruby, D. I. (2006). Award winning constructability: Lansing Community College. *Structure Magazine*. Retrieved from <http://www.structuremag.org/>.
- [14] RSMMeans Inc. (2004). *RSMMeans Building Construction Data*. 62nd annual edition, R. S. Means Co., Inc, Kingston, MA.
- [15] 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*, 118(1), 179-198.
- [16] Bisharat, K. A. (2004). *Construction Graphics: A Practical Guide to Interpreting Working Drawings*, John Wiley & Sons Inc.
- [17] Skibniewski, M., Arciszewski, T., Lueprasert, K. (1997). Constructability analysis: a machine learning approach. *Journal of Computing in Civil Engineering*, 2(1), 8-17.
- [18] Navon, R., Shapira, A., and Shechori, Y. (2000). Automated rebar constructability diagnosis". *Journal of Construction Engineering and Management*, 126(5), 389-397.
- [19] Hanna, A. S., and Sanvido, V. E. (1990). Interactive vertical formwork selection. *Concrete International Design and Construction*, 12(4), 26-32.
- [20] Ugwu, O. O., Anumba, C. J., and Thorpe, A. (2004). The development of cognitive models for constructability assessment in steel frame structures. *Advances in Engineering Software*, 35, 191-203.
- [21] Allen, E., and Iano, J. (2002). *The Architect's Studio Companion: Rules of Thumb for Preliminary Design*, 3<sup>rd</sup> Edition, Willey, NY.
- [22] Staub-French, S., Fischer, M., Kunz, J., Ishii, K., and Paulson, B. (2003). A feature ontology to support construction cost estimating. *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing*, 17, 133-154.
- [23] Peurifoy, R. L., and Oberlender, G. D. (1996). *Formwork for Concrete Structures*, 3<sup>rd</sup> ed., McGraw Hill.
- [24] Haymaker J., Fischer M., Kunz J., and Suter, B. (2004). Engineering test cases to motivate the formalization of an AEC project model as a directed acyclic graph of views and dependencies. *ITcon*, 9, 419-441. Available online at: <http://www.itcon.org/2004/30/>

- [25] Luth, G. P., Jain, D., Krawinkler, H., and Law, K. H. (1991). A formal approach to automating conceptual structural design, part I: Methodology. *Engineering with Computers*, 7, 79-89.
- [26] Korman, T. M., Fischer, M. A., and Tatum, C. B. (2003), Knowledge and reasoning for MEP coordination, *Journal of Construction Engineering and Management*, 129, 627-634.
- [27] Tabesh, A., and Staub-French, S. (2006). Modeling and coordinating building systems in three dimensions: A case study. *Canadian Journal of Civil Engineering*, 33(12), 1490–1504.
- [28] International Alliance for Interoperability (IAI). (2010). Available online at: <http://buildingsmart-tech.org/>