Nepal, Madhav, Staub-French, Sheryl, Pottinger, Rachel, & Zhang, Jiemin (2013)
Ontology-based feature modeling for construction information extraction from a building information model.
Journal of Computing in Civil Engineering, *27*(5), pp. 555-569.

https://doi.org/10.1061/(ASCE)CP.1943-5487.0000230

Ontology-Based Feature Modeling for Construction Information Extraction from a Building Information Model

Madhav Prasad Nepal[1] A.M.ASCE, Sheryl Staub-French[2] A.M.ASCE, Rachel Pottinger[3], and Jiemin Zhang[4]

Abstract

A building information model (BIM) provides a rich representation of a building's design. However, there are many challenges in getting construction-specific information from a BIM, limiting the usability of BIM for construction and other downstream processes. This paper describes a novel approach that utilizes ontology-based feature modeling, automatic feature extraction based on ifcXML, and query processing to extract information relevant to construction practitioners from a given BIM. The feature ontology generically represents construction-specific information that is useful for a broad range of construction management functions. The software prototype uses the ontology to transform the designer-focused BIM into a construction-specific feature-based model (FBM). The formal query methods operate on the FBM to further help construction users to quickly extract the necessary information from a BIM. Our tests demonstrate that this approach provides a richer representation of construction-specific information compared to existing BIM tools.

Keywords: *Building Information Modeling (BIM), Construction Management, Building Design, Design Features, Feature-Based Model, Ontology, Information Extraction*

---

[1] Ph.D., Lecturer, School of Civil Engineering and Built Environment, Gardens Point Campus, GPO Box 2434, Queensland University of Technology, Brisbane QLD 4001, Australia; Email: madhav.nepal@qut.edu.au
[2] Ph.D., Associate Professor, Department of Civil Engineering, University of British Columbia, Vancouver, B.C. Canada, V6T 1Z4; Email: sherylsf@civil.ubc.ca
[3] Ph.D., Associate Professor, Department of Computer Science, University of British Columbia, Vancouver, B.C. Canada, V6T 1Z4; Email: rap@cs.ubc.ca
[4] Systems Developer Analyst, Broadridge Financial Solutions, Inc., Vancouver, B.C. Canada; Email: jiemin.zhang@broadridge.com

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 modeling (BIM) has created new challenges as well as new opportunity for construction practitioners. BIMs explicitly represent building components, component properties, and relationships to capture the design perspective. 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. Specifically, designer-focused BIMs do not represent many component properties (e.g., component shape), relationships (e.g., component penetrations), and the spatial context (e.g., the location of penetrations) that are important to construction practitioners. This construction-specific information must be derived by analyzing the topology and geometry of building components (Borrmann and Rank 2009; Haymaker et al. 2004; Katranuschkov et al. 2003). As a result, construction professionals today spend a significant amount of time and effort manually analyzing and interpreting design information (2D drawings, 3D models, etc.) to identify this construction-specific information, which is time consuming and prone to error (Eastman et al. 2008).

Emerging BIM applications provide some support for extracting construction-specific information. *Autodesk® Revit®*, *Innovaya®* and *Solibri Model Checker®* (SMC), for example, can identify explicitly defined geometric and material information in an underlying BIM using ‘schedule’ or ‘information/material takeoff’ tools. SMC provides a constraint set manager (CSM) for managing and configuring constraint sets or rules for checking the design (e.g., interference checking, model pre-checking, space checking, quantity take off). SMC also provides rich support for extracting explicitly defined dimensional and component property information. Conflict detection mechanisms in *Autodesk® Navisworks®* Manage can be used to find hard and soft conflicts between building components, which would help to find penetrations, for example. These programs, however, suffer from many limitations in terms of extracting construction-specific information: they lack sufficient flexibility to extend object properties and parameter values to encode domain-specific (e.g., construction) knowledge (Ding et al. 2004); they do not differentiate between a conflict, an intersection, or a penetration (all soft and hard conflicts are listed); they often identify false positives when performing clash detection (some 'conflicts' are intentional); they do not explicitly represent specific types of intersections (e.g., intersections between drywall and round columns) or related information (e.g., penetrations on a specific type of wall); and they cannot find the location of intersections (e.g., specific walls that have penetrations).

The research challenge in extracting construction-specific information from a given BIM is that construction practitioners have differing preferences, viewpoints, and rationale for expressing when and how a particular design condition impacts construction. They need flexibility to specify the properties of and spatial relationships between building components. Researchers have tried to encode different design conditions to provide computer support for extracting construction information from a 3D building model. The approaches to date, however, have been limited: focusing on a narrow set of design conditions (e.g., Chen et al. 2005), requiring extensive user input (e.g., Nguyen and Oloufa 2002), and lacking user customizability (e.g., Haymaker 2004).

To address the practical and technological challenges mentioned above, we developed a novel approach that involves ontology-based feature modeling, automatic feature extraction based on ifcXML, and query formulation and processing. This approach enriches existing BIMs with construction-specific information (e.g., 'penetrations' are explicit in the model), and provides a method for practitioners to query the enriched BIM (e.g., find 'penetrations' in fire-rated walls). This research makes the following contributions:

- We formalize an *ontology of construction-specific design features* to represent a wide range of construction-specific information.
- We create a project-specific *feature-based model (FBM)* based on an ifcXML input to provide an enriched BIM tailored for construction users.
- We formalize *queries* that can be customized to extract additional information from the FBM.
- We provide an assessment of how feasible it is to use *Industry Foundation Classes* (*IFC*), in particular ifcXML, to extract relevant construction information from a given BIM.

The paper is organized as follows. We first introduce the practical motivation for this research using a few case studies. We then describe related work, which is followed by a brief discussion of the research methodology. Next we provide an overview of our system and describe in detail the ontology of construction-specific design features (the feature ontology), the feature-based model, and the querying process. Finally, we provide an evaluation of our approach followed by a brief discussion of the limitations of our work and potential for future research, and conclusions.

Motivating Case Studies

This section describes two case examples that illustrate the practical motivation for this research. The examples illustrate the kinds of design conditions that impact construction and are based on detailed case studies of two building projects we studied extensively: (1) the Chemical and Biological (Chem-Bio) Building project, and (2) the Engineering Design Center (EDC) project.

The examples in Figure 1 highlight specific design conditions that affect construction. We refer to these design conditions as *construction-specific information*. Figure 1(a) shows the configuration of walls on the fifth floor of the Chem-Bio project. Note the variety and spatial distribution of different wall types that are characterized using various attributes, such as the shape (e.g., curved wall, clipped wall), material (e.g., brick masonry wall), wall function or location in relation to building environment (e.g., exterior wall, interior wall), and other symbolic attributes (fire-rated wall, acoustic wall, etc.) to distinguish their types. Practitioners also distinguish walls based on 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 wall) and their characteristics (e.g., metal stud wall, 15 MPa concrete wall). The spatial distribution and variability of wall types impact productivity in many different ways. For example, different wall heights or ranges of wall heights (e.g., 8, 12 and 15 inch walls) and types require different construction methods and activities. Different wall thicknesses and types of wall shapes (e.g., clipped or curved walls) have different productivity rates. For example, consider the case of a brick veneer wall: RSMeans Inc. (2004) states that if the wall is battered, 30% labor cost should be added to the cost; and if it is a curved wall, an extra 30% is added.

Other construction-specific information highlighted in the literature and confirmed in our second case study (the EDC project) emphasizes the importance of component openings and penetrations and different types of component intersections (Bisharat 2004; Staub-French et al. 2003). Figure 1(b) shows the superintendent's hand drawn sketches indicating the size and location of openings and penetrations in walls for the EDC project. Penetrations are an important design condition for practitioners because they require additional work, which is sometimes dependent on the type of component that is intersected. For example, penetrations through fire-rated walls and slabs must be fire stopped; service penetrations through classrooms with acoustic walls and through block walls between labs must be packed and caulked. Similarly, the use of tunnel forms for concrete wall construction requires uniformity in the size and location of openings (Fischer and Tatum 1997). The intersection of two walls (or wall turns) may necessitate 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 of movement joints.

To make BIMs useful for construction, this construction-specific information must be explicit in the model in a way that practitioners can easily work with to better understand, plan, estimate, and coordinate their work. The next section describes prior research in this field, and how this work advances the state of knowledge.

Relevant Background

We combine and extend previous research on design-specific construction knowledge, ontological modeling, and building product modeling (or BIM) and reasoning. We build on the work of previous researchers that identified different design conditions that impact constructability (e.g., Fischer and Tatum 1997), labor productivity (e.g., Thomas and Zavrski

1999), method selection (e.g., Hanna et al. 1992), and construction costs (e.g., Staub-French et al. 2003). This work helped to define what design conditions are important to different practitioners, and provided the foundation for the scope of the feature ontology. The feature ontology provides a semantic layer between the end users and the BIM and represents knowledge about construction-specific design conditions in a computer-interpretable and unambiguous way.

This research builds on and extends the feature ontology developed by Staub-French et al. (2003). They classified features as Component, Intersection, and Macro features. Component Features were defined as components that are represented consistently with *IFC*-based product models. The attributes of Component Features represented were based on *IFC* properties. Intersection Features were defined as features that result from the intersection of two components. The attributes of Intersection Features represented were based on *IFC* relationships and included 'RelatedComponent' and 'RelatingComponent'. No other attributes related to component intersections were formalized. They defined Macro Features as features that result from pre-specified combinations of other features and focused on the similarity of components of the same type. Our ontology builds on the Component and Intersection Feature classes formalized in this work. We provide a formalism to better characterize the different kinds of intersections that are important to construction practitioners, and we provide a significantly richer characterization of Component and Intersection Features by defining a broad range of attributes that are important to construction practitioners. We also represent this knowledge in a way that supports user interaction and querying of these features.

Semantic or ontological modeling in construction has generally focused on developing a taxonomy, or vocabulary, of building and construction information and conceptualizing it through the use of relations, constraints or axioms (Woestenenk et al. 2000; El-Gohary and El-Diraby 2010, El-Diraby et al. 2005; *ISO* 2001; *OCCS* 2008). Most notably, the *IFC* provides standardized terminologies and model schemas for representing semantically rich information related to the design and other aspects of the AEC/FM industry (*IAI* 2010). *IFC* explicitly defines reusable *Property Sets* for different component types to represent the properties that are important to designers. *Property Sets* consist of the predefined property definitions. *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). *IFC* also indicates that exact definition and calculation rules for quantitative parameters may depend on the method of measurement or application used (*IAI* 2010). We reuse the relevant predefined *IFC* properties and define new attributes that are important from a construction perspective. The defined features and attributes together help to fulfill our requirements for the FBM and subsequently provide a support for the users to flexibly and easily specify queries that meet the unique requirements of construction practitioners.

Pre-defined BIM schemas, such as *IFC*, provide a standardized structure to construct and interpret a BIM. The standard, however, is very complex; an easy to use implementation is needed (Howard and Björk 2008). It is up to the applications to structure the needed information on top of BIM and/or provide reasoning support to facilitate the extraction of construction-specific information. Many research efforts have added representation schemas and employed task-specific reasoning structures in order to construct specific views out of a BIM model. Some related studies use *IFC*-based models or *IFC* Model Servers to generate application-specific views (Taneja et. al. 2011; Chen et al. 2005), and employ spatial query language to extract partial models that fulfill certain spatial constraints (Borrmann and Rank 2009). Other studies develop ontologies on top of *IFC* models to access *IFC* data (Katranuschkov et al. 2003) and support knowledge management (Rezgui 2006; Lima et al. 2005). Beetz et al. (2009) define application or knowledge-based models for transforming *IFC* model information to ontologies which they use for processing building information through generic query and reasoning algorithms. Scherer and Schapke (2011) also use ontologies as the knowledge-based models for the integration of various knowledge resources or application models from different construction domains. Wang et al. (2011) developed an ontology-based framework to represent contextual information and support indexing, retrieving, and accessing information.

Our research developed the ontology of construction-specific design features (the feature ontology) that we formalized to transform a BIM model available as a part of design process into a construction-specific FBM, a view of design information specifically tailored to construction practitioners. We leverage and employ the relevant pre-defined schemas and entities (objects, properties, and relationships) defined in the *IFC* model and BIM authoring tools (*Revit*, in our case) and use queries to identify information relevant to construction from a BIM model.

Research Methodology

Acquiring Knowledge for Developing the Feature Ontology

To develop the feature ontology, we looked carefully at a number of sources to decide what design conditions (i.e., concepts) to represent in our ontology, and to ensure that our ontology was comprehensive, representative and relevant. In particular:

- We gathered relevant terms and concepts through an extensive literature review on design constructability, value engineering, cost estimating, methods selection, construction planning and scheduling.
- We conducted four case studies of construction projects, which included three detailed case studies (two are described in the Motivating Case Studies section) and one 6-month observational study of weekly design coordination meetings.
- We interviewed seven different construction practitioners, including cost estimators, superintendents, project managers, and a concrete foreman. We interviewed several cost estimators and a project manager numerous times to better understand how practitioners describe and characterize the different design conditions that impact various construction trades and CM functions.

Reasoning Methods

We use an XML representation of BIM data for feature extraction and querying. 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. We extracted the required

BIM data from *Revit* in two different ways. First, we made use of ifcXML as much as possible, as it offers the most comprehensive coverage of the relevant features represented in the feature ontology than other XML formats (Zhang et al. 2011). However, much of the spatial information and relationships between features (e.g., location of duct on wall and slabs), and other geometric information (e.g., area, volume of component intersection and penetration) was not available in ifcXML. Such unavailable data was extracted from the *Revit* API, which provides programmers direct access to the internal structure of the *Revit* model. We used the standard XML query language, XQuery, and custom implemented XQuery spatial query predicates to map each concept defined in the feature ontology to a corresponding concept in the standard schema (i.e., ifcXML and *Revit* data) to automatically extract and query features. The next section provides an overview of our proposed system.

System Overview

Figure 2 shows a process diagram of the proposed system. In the first step (Create Feature-based Model), the prototype application, '*Feature Extractor*' that we created abstracts and analyzes the relevant geometric, topological, and other attributes and characteristics of objects in the input *IFC*-based BIM model to identify the feature instances and attributes defined in the feature ontology. It transforms the input *IFC*-based BIM model into a project-specific feature-based model (FBM) that explicitly represents the features that are important to a particular construction practitioner or domain. For this step, we formalized a feature ontology to generically represent construction-specific design conditions. In the second step (Query Features), users configure queries that operate on the project-specific FBM. The system takes the query input from the user and executes the application '*Feature Query*

Analyzer' to process queries. For this step, we developed query specifications to formalize the language and structure of the user-driven queries in relation to a BIM. This paper focuses on basic queries of the FBM, though additional queries have been developed to identify spatial relationships in a BIM, which are described in detail in Nepal (2011) and will be published in a companion paper.

The following subsections describe the ontology of construction-specific design features, feature extraction and the feature-based model, and the process for querying features in more detail. First we describe an ontology of construction-specific design features – a conceptual model of the shared domain knowledge we formalized – to represent construction-specific information.

An Ontology of Construction-Specific Design Features

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). The ontology of construction-specific design features, i.e., the feature ontology, developed in this research formalizes a common vocabulary 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 are general, computer interpretable, and easily understood by practitioners.

We use the manufacturing concept of *features* and *feature-based modeling* (Cunningham and Dixon 1988; Shah 1991) to represent design information that is important from the construction perspective. *Features* refer to meaningful real world entities (objects) to which one can associate construction-specific information. We focused 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 basic building construction components, such as walls and columns, and the relationships between components. As such, many detailed design features (e.g., connections, joints, tolerances), as well as system level features (e.g., frames, floor systems, foundation systems), are currently excluded from the scope of the ontology.

Classifying Features

Classifying features into categories is useful for a number of reasons but in particular it facilitates the development of a library of feature types and associated attributes that can be represented at various levels of abstraction and detail (Shah and Mäntylä 1995). We classify features into two broad categories: *component* and *intersection* features.

A *component* is a feature that is a common building element. Components are further categorized into feature subclasses representing more specific concepts, such as *walls*, *columns*, *slabs*, and *beams*, etc. A "*component*" is similar to *IFC's IfcBuildingElement*, but it is more focused on construction usage than design usage. Not all *IfcBuildingElement*s are good candidates for being "*component*" features, because there are elements that are less significant or relevant from the overall construction viewpoint. For example, *IfcCovering* and *IfcRailing* may not carry as much meaning in construction.

An *intersection* is the physical/geometric interaction between components that results in the formation of different types of component relationships. We further classify *intersection* into three types: *component intersection, opening*, and *penetration*, as described below.

A *component intersection* is the physical/geometric interaction or connectivity between building components. Component intersections may involve conditions, such as a face of one building component overlapping or attaching the face or edge of another building component in a vertical plane, a building component abutting another building component, a building component crossing another building component, a building component supporting or supported by another building component for vertical load transfer function. Component intersections can occur between building components of the same type, such as intersections between walls (wall to wall intersection) or between different types (e.g., wall to column intersection). Other information about the intersecting regions can also be important to construction, such as the type of intersecting building components (e.g., masonry wall intersecting a drywall partition wall), and the relative dimension and characteristics (e.g., fire-rating) of intersecting building components.

An *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). A *penetration* describes design conditions that involve building service elements entering or passing through building components, such as a duct or pipe penetrating a wall or slab.

We classify *intersection* features into different types 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 door and window openings. Also, the type and nature of the host component, on which openings or penetrations exist, can convey different points of view to different practitioners. For example, 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, and the nature and the context of the relationships.

The representation of *intersection* features in current BIM applications varies. For instance, door and window openings are 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 ducts or pipes, is typically not explicit in the BIM model because they are not defined as the models are created but rather emerge based on the models developed by the different disciplines.

While *IFC* defines many spatial relationships (mainly topological) that may occur between objects or elements of a building, *IFC*-based BIM applications do not provide a mechanism to filter for specific types of intersections that are meaningful to construction practitioners.

Defining and Representing the 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. As shown conceptually in Figure 3, various relationships between features can be generalized as being association and specialization relationships. The specialization relationships represented as class sub-class relationships result in the taxonomy of features organized hierarchically and related by a Is-A type relationships. Associative relationships are the most common types of relationships, linking one feature to other features, thereby assisting users to trace relationships between building components during the browsing and querying of the FBM.

Feature-specific properties, on the other hand, are distinct attributes that are generally applied or assigned to a specific feature. Figure 4 shows the class diagram of the ontology with both types of attributes relevant to each feature class. We represent relational attributes of features by attaching them directly to a feature class, as if they are the intrinsic feature properties. 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.

We operationalize the conceptual perspective described thus far to an implementation perspective in which we specifically consider the frame-based knowledge representation to represent features and feature attributes (*Protégé* 2008). Each attribute has a unique name, data (or value) type, cardinality, and if applicable, reference and default values. Tables 1 and 2 provide a general description of attributes for the feature class *component* and its subclass *wall* respectively, including their value type and cardinality. Tables 3 and 4 respectively provide similar information for the feature class *intersection* and its subclass *penetration*.

Feature Extraction and the Feature-Based Model

In the previous section, we generically defined different types of features and feature attributes. The feature-based model (FBM) is the instantiation of these project independent features to a particular project. This means that all predefined features in the feature ontology are extracted or populated with project specific feature instances and their corresponding attribute values for a given BIM. Next, we describe the feature extraction process and discuss some of the challenges involved when extracting features from ifcXML and the underlying BIM model.

Mapping the BIM Model to the Ontology: Automating Feature Extraction

The process of extracting the information required by domain experts is a cumbersome task. Because the data is extracted from the BIM according to some standard data schema (e.g., ifcXML) the concepts needed by the domain experts may be very hard to find. As a result, extraction requires formalizing the mappings from concepts in a domain model (i.e., feature ontology) to the standard data schema. 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. We implemented XQuery query predicates to 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), which is described in more detail in Webster (2010).

We use an XML representation of BIM data from *Autodesk Revit* for feature extraction. 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 5(a) to (c) show 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 resulted in 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 5(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 information must be determined by navigating a complex sequence of references. Furthermore, an element sometimes refers to an ID of another element to describe their relationships. Figure 6 shows a few concepts and the reference paths, displaying their linkage with a wall object. It is 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 (Nepal et al. 2008). Moreover, the most challenging problem is 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 also incomplete.

Examples of Feature Extraction

In the following section, we use specific examples to illustrate the process of automatically extracting design features and attributes defined in the feature ontology. We highlight the

IFC attributes and/or BIM data used to identify them, and the inherent difficulties or complexity involved.

Example 1: 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 because these design conditions impact construction and should be made explicit in the model. Deriving these attributes requires an understanding of how walls are represented in the *IFC* model. IFC allows multiple geometric representation of shape using IfcFaceBasedSurfaceModel and IfcExtrudedAreaSolid attributes and many other subtypes to support the needs of all BIM systems and to facilitate the exchange of geometry in some form. The attribute, IfcExtrudedAreaSolid is heavily used in the parametric modeling of a building design. The two ways of representing walls in ifcXML – "IfcFaceBasedSurfaceModel" and "IfcExtrudedAreaSolid" – are shown in Figure 6(g) and (h), respectively. IfcFaceBasedSurfaceModels contain IfcConnectedFaceSets and, following the referring paths (i.e., ifcFace, ifcFaceOuterBound, ifcPolyLoop), we eventually find the Cartesian 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 6(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. Analyzing the geometric representations of walls in terms of these attribute-driven representations of shape is the key to identifying the shape of a wall. Some attributes are easier to analyze than others. For example, IfcFaceBasedSurfaceModel is more complicated to analyze compared to IfcExtrudedAreaSolid. The standard geometric (body) representation of IfcWallStandardCase is defined by using the 'SweptSolid' representation for wall without clippings or 'Clipping' representation for walls with clippings. The 'SweptSolid' representation requires that a body be represented as IfcExtrudedAreaSolid. The 'Clipping' representation, however requires the use of IfcHalfSpaceSolid (or a subtype of it) in addition to IfcExtrudedAreaSolid.

The attributes reference paths, shown in Figure 6, also provide a clue 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 6 (h)], our system 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 6 (g) or (h) turns out to be a complicated method to derive the property "curved." If space boundaries are defined physically, the wall geometry can be determined from the shape of the boundary surface of a wall. Figure 6(d) shows the ID referring path needed to determine the shape of the boundary surface from a wall. In case of a straight wall, a single geometric representation item of type IfcPolyline (or IfcTrimmedCurve with basis curve of type

IfcLine) is used. For a curved wall, the set of items includes a single geometric representation item of type IfcTrimmedCurve with the basis curve of type IfcCircle.

This example highlights the challenges of working with *IFC* but also demonstrates that sufficient information is provided to characterize important geometric characteristics of components that are useful to construction, specifically for curved and clipped walls.

Example 2: 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 typically modeled with explicit connections in BIM. So we can query ifcXML, following the paths shown in Figure 6(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 6(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. Architects often model walls and columns in the same model, particularly during the early periods of design. In this case, these intersections will be explicit. However, as the structural model gets more developed and becomes the 'master' model for structural components, these explicit relationships no longer exist. Therefore, these types of intersections have to be derived by analyzing the location information of related objects. In the initial phase of this research and as reported in Nepal et al. (2008), 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, 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, including its location (i.e., the boundary points of the region), dimensions (X, Y, Z dimensions), size, area and volume (Webster 2010). Such detailed information about a component intersection is not explicitly defined in both *IFC* and *Revit* because they do not treat "component intersections" as distinct objects.

This example shows how we extended the *IFC* to provide a richer representation of component intersections.

Example 3: 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 'intersection 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), which is described in detail in Webster (2010).

Feature-Based Model Interface

The prototype user interface for the FBM is presented in Figure 7. The left side of the interface 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. 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.

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’), shown in Figure 7 (a), which extends the property information typically represented by the *IFC*. Also evident is the explicit representations of all component intersections (Figure 7 b) and penetrations which are not explicit in a given 3D model and corresponding *IFC* export. The instantiated relationships between features are dynamically linked, which means that the user can trace to the linked or referenced features. The flexibility and expressivity of the FBM is one of its key strengths. Not only does it promote a better understanding of the design features that are of importance to construction practitioners and that are present in the given BIM model, but it also can provide guidance to the user in the formulation of queries.

Querying Features

We use the FBM and formal query specifications to support the processing of user-driven queries on a BIM. In this section, we describe how basic queries on the FBM can be customized to enable the users to filter, group, and aggregate the instantiated features in the FBM. We also implemented spatial queries (e.g., the location, spacing and alignment of features), which are described in a companion paper based on the work of Nepal (2011).

We developed form-based query specification templates to capture practitioners' varying needs, rational or preferences for describing and formulating queries. These templates also provide the users guided assistance to specify queries without knowing the underlying BIM data model or query language (Nepal 2011). We will use the following example to explain the basic querying process.

<u>**Example:** Identify all fire-rated walls that have penetrations.</u>

Figure 8 (a) through (d) illustrate the different query steps – *feature selection*, *property filtration*, *grouping,* and *quantification* – that are useful to manipulate the FBM. The querying process starts with the user selecting a feature as well as a floor to query (Figure 8 a). In the next step, *Property Filtration* allows the user to define properties (or constraints) for the feature selected in the *Feature Selection* process. The properties available correspond to the feature attributes defined in the feature ontology for that feature. For the given example query, the wall properties 'is interior,' 'fire-rated,' 'material,' and their corresponding values, are used to define the wall type (i.e., dry wall), whereas the relational wall property 'Has penetration' further allows the user to constrain the query results to identify walls with penetrations (Figure 8 b).

The *Grouping* option enables the user to group feature instances based on a property or properties, such as grouping the instances of all walls based on fire-rating values, e.g., 1 hr, 2 hr, etc. (Figure 8 c). Oftentimes, practitioners need to quantify or aggregate query results. This functionality is facilitated through the *Quantification* step, which allows the user to specify the numeric property or properties and define the aggregate function(s) to aggregate the query results, such as calculating the total length of walls (Figure 8 d) that fulfill the constraints, as defined in the *Property filtration* and *Grouping* steps.

Queries thus provide an interface and access to extract user-customized information from the enriched FBM. The next section presents an evaluation of our approach.

Evaluation

The evaluation of our approach basically involves two related parts: validating the feature ontology and the resulting FBM, and evaluating our approach for identifying construction-specific information. 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 largely depends on the types of ontologies and their specific purpose (Brank et al. 2005). For our evaluation, we used interviews with construction experts, and conducted a retrospective analysis.

Detailed Interviews with Construction Experts

We conducted interviews with four construction domain experts to examine the developed ontology in terms of its vocabulary. We interviewed the experts in reference to four building projects: The Chem-Bio building and the Engineering Design Center (described in the Motivating Case Studies section), the Discovery Green building, and the Fipke Center for Innovative Research building (Nepal 2011). The experts assessed the degree of relevance (or importance) of different design conditions related to building components in general and walls and columns, in particular, component intersections, penetrations, and openings. They provided expert opinion on how or under what conditions the design conditions would be relevant, and what information or queries they typically ask or look for in a given design. The interviewed construction experts included a project manager, a site superintendent, a formwork manager, and a chief estimator. The project manager played the role of the generalist, surveying the design conditions from the perspectives of component layout, component installation, constructability, cost estimating, methods selection, and construction planning. The formwork manager had the perspective of formwork cost and constructability in the construction and erection of concrete formwork. The interview with the site superintendent reflected the viewpoint of the general contractor for managing construction operations, trade coordination and all aspects of a project on site. The chief estimator that we interviewed represented the general contractor which provides CM services to clients in British Columbia and Alberta.

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.

Rather than describe in detail all the results of the interviews, here we focus on a representative feature: “penetration.” The full interview results are available in Nepal (2011). Figure 9 shows the experts’ assessment about the relevance of design conditions related to the feature “penetration.” We used tables like these to get specific feedback on the features and attributes represented in the feature ontology. Most experts agreed that the existence of penetrations on building components, such as walls and slabs, is important for construction. For the site superintendent, locating the exact size and location of all openings and penetrations on slabs and walls was a time-intensive exercise as they were not always explicit in the drawings (Figure 1b). The site superintendent explained it this way:

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

The construction-specific knowledge formalized in this research was collected from several case studies of actual construction projects and an extensive review of the literature. The detailed interviews with the construction experts provide supporting evidence that the knowledge formalized in this research is *representative* of reality in terms of characterizing design conditions that are relevant to construction practitioners. Four different experts, representing different construction companies, domains and viewpoints and with reference to four different projects, confirmed the importance of the construction-specific information we formalized. The interviews also demonstrated the *generality* of the information formalized because different types of practitioners considered this information to be important.

The next part of the validation provided evidence that our approach is able to provide richer representations of construction-specific information within a BIM compared to existing tools.

Retrospective Analysis

The retrospective analysis assessed the level of support that our approach provides in extracting and querying construction relevant information from a BIM model. The purpose was to demonstrate the *soundness* of our approach in comparison with state-of-the-art tools. In order to conduct the retrospective analysis, we compiled, for each feature type, a list of design conditions that are significant to construction. They were compiled, based on a thorough review of the literature and our detailed interviews with construction experts for the four 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 functions (e.g., cost estimating, method selection, and constructability). We used this compilation as a “gold standard” set of concepts that were desirable from the construction perspective. Then, we checked to see whether our implementation and state-of-the-art tools, *Solibri Model Checker* and *Navisworks* (in aggregate), included the concepts. These tools were selected because they provide the most advanced support for analyzing a BIM from the construction perspective.

In comparing systems against such a gold standard, two metrics are commonly used to determine the value of the system: *precision* and *recall.* In this case, precision measures how many of the concepts in a system are correct, while recall measures what fraction of correct answers from the gold standard are returned by a given system. Because the different systems are made for very different purposes, we did not measure precision – e.g., it makes little sense to penalize the results of *Navisworks* for including all clashes based on the geometry of the building components because *Navisworks* has the ability to work with all the key 3D design file formats, but *Navisworks* doesn't have the functionality to leverage richness of BIM data in meaningful ways. Instead, we concentrated our evaluation using the measure of recall. Figure 10 summarizes the results of the retrospective analysis using this measure. The full details of the results are available in Nepal (2011). These results suggest that our approach provides more significant and flexible support to extract useful construction-specific information than state-of-the art tools. Specifically, state-of-the-art tools lack support for identifying construction-relevant design conditions; the only features on which they have greater than 50% recall 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 current state-of-the-art tools.

Limitations of the Current Research and Future Work

The types of features and feature attributes formalized in this research are not complete. The breadth and depth of concepts formalized in the ontology could be extended to accommodate other types of components and design conditions that are relevant for construction and to provide different levels of abstraction. While many of the design conditions formalized for "wall" and "column" component features could be applicable to other building components, further research is needed to formalize these specific attributes. The "component feature" consists of basic building construction components. There could be other families of components, such as a "connection" family of components to address the needs of steel construction. Even for walls and columns, we do not represent or identify some detailed features, such as the existence of corbels, pilasters on walls, column capitals, column drop heads, etc. They could be defined as an "add-on" family of components. Future research is needed to represent such families of features and develop mechanisms to identify them in a given BIM. More research is also needed to extract detailed information about component intersections. We have not proposed methods that explicitly recognize different types of wall-to-wall intersections (e.g., T, L, end-to-end, and overlap) or wall corners that practitioners would find relevant.

In our current approach, we define feature types and attributes *a priori* and use specialized reasoning mechanisms to extract them. However, allowing 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 explicitly defined *IFC* properties (e.g., such as fire-rating, interior vs. exterior) can be generalized across different components, extracting implicit attributes (e.g., geometric shape) can be highly challenging as the domain knowledge required for their extraction can vary from one component to another.

In the current implementation we do not provide adequate support to visualize the extracted and queried features in the corresponding 2D and 3D design views of a BIM. Additional work is needed to provide effective visualization and management of the extracted information. Our particular interest will also be on the direct integration of our approach with different CM applications, such as cost estimating, construction scheduling, and BIM analysis tools (e.g., clash detection and design checker).

Conclusions

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

A key consideration in developing the ontology is to provide a consistent, unambiguous, and computer-interpretable representation of features that are important from the construction perspective. The ontology formally represents common and important design conditions in terms of features and feature attributes (feature properties and relationships between features). Using specific examples, we described the process of automatically instantiating the feature ontology to create a project-specific FBM, which can also be queried to identify additional construction-specific information. This research could help practitioners to enhance the usefulness of BIM for a variety of construction management functions.

References

Beetz, J., Van Leeuwen, J., and De Vries, B. (2009). "IfcOWL: A case of transforming EXPRESS schemas into ontologies." *Artif. Intell. Eng.Des. Anal. Manuf.*, 23, 89-101.

Bisharat, K. A. (2004). *Construction graphics: A practical guide to interpreting working drawings*, John Wiley & Sons Inc.

Booch, G., Rumbaugh, J., and Jacobson, I. (1999). *The unified modeling language user guide*, Addison-Wesley, MA.

Borrmann, A., and Rank, E. (2009). "Topological analysis of 3D building models using a spatial query language." *Adv. Eng. Inform.*, 23(4), 370-385.

Brank, J., Grobelnik, M., and Mladenic, D. (2005). "A survey of ontology evaluation techniques." *Proc., Conf. on Data Mining and Data Warehouses*, Ljubljana, Slovenia.

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." *Autom. Constr.*, 14, 115-128.

Cunningham, J. J., and Dixon, J. R. (1988). "Designing with features: the origin of features." *ASME Computers in Eng. Conf.*, San Francisco, CA, pp. 237-243.

Ding, L., Drogemuller, R., Jupp, J., Rosenman, M. A., and Gero, J. S. (2004). "Automated code checking." *Proc., CRC for Construction Innovation, Clients Driving Innovation International Conference*, Gold Coast, Australia.

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.” *Int. J. Geographical Inf. Syst.*, 5(2), 161-174.

Ekholm A. (2002). “Principles for classification of properties of construction objects.” *Distributing Knowledge in Building - CIB W78 Conference 2002,* K. Agger, P. Christiansson and R. Howard, eds., Aarhus School of Arch., Aarhus, Denmark.

El-Diraby, T. A., Lima, C., and Feis, B. (2005). “Domain taxonomy for construction concepts: Toward a formal ontology for construction knowledge.” *J. Comput. Civ. Eng.*, 19(4), 394-406.

El-Gohary, N. and El-Diraby, T. (2010). “Domain Ontology for Processes in Infrastructure and Construction.” *J. Constr. Eng. Manage.*, 136(7), 730-744.

Fischer, M., and Tatum, C. B. (1997). “Characteristics of design-relevant constructability knowledge.” *J. Constr. Eng. Manage.*, 123(3), 253-260.

Genesereth, M. R., and Nilsson, N. J. (1987). *Logical foundations of artificial intelligence*, San Mateo, CA; Morgan Kaufmann.

Hanna, A. S., Willenbrock, J. H., and Sanvido, V. E. (1992). “Knowledge acquisition and development for formwork selection system.” *J. Constr. Eng. Manage.*, 118(1), 179-198.

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.

Howard, R., and Björk, B. (2008). “Building information modeling - Experts' views on standardisation and industry deployment.” *Adv. Eng. Inform.*, 22, 271-280.

International Alliance for Interoperability (IAI). (2010). Homepage: http://buildingsmart-tech.org/.

ISO 12006-2. (2001). *Building construction – organization of information about construction works – Part 2: Framework for classification of information*, International Organisation for Standardisation.

Katranuschkov, P., Gehre, A., and Scherer, R. J. (2003). "An ontology framework to access IFC model data." *ITcon*, 8, 413-437.

Lima, C., El-Diraby, T. E., and Stephens, J. (2005). "Ontology-based optimization of knowledge management in e-construction." *ITcon*, 10, 305–327.

Nepal, M. P. (2011). "Automated extraction and querying of construction-specific design features from a building information model." PhD thesis, Dept. of Civil Eng., Univ. of British Columbia, Vancouver, Canada.

Nepal, M. P., Staub-French, S., Zhang, J., Lawrence, M. and Pottinger, R. (2008). "Deriving construction features from an IFC model." *Proc., CSCE 2008 Annual Congress*, Quebec City, Canada.

Nguyen, T., and Oloufa, A. A. (2002). "Spatial information: classification and applications in building design." *Computer-Aided Civ. Inf. Eng.*, 17, 246-255.

Overall Construction Classification System (OCCS). (2008). *OmniClass: A strategy for classifying the built environment*, < http://www.omniclass.org/>

Protégé. (2008). Homepage: <http://protege.stanford.edu/>.

RAPID. (2008). "Robust and accurate polygon interference detection." Homepage: <http://gamma.cs.unc.edu/OBB/>.

Rezgui, Y. (2006). “Ontology-centered knowledge management using information retrieval techniques.” *J. Comput. Civil Eng.*, 20(4), 261-270.

RSMeans Inc. (2004). *RSMeans building construction data*, 62nd annual edition, R. S. Means Co., Inc, Kingston, MA.

Scherer, R. J., and Schapke, S.-E. (2011). “A distributed multi-model-based management information system for simulation and decision-making on construction projects.” *Adv. Eng. Inform.*, 25(4), 582-599.

Shah, J. J. (1991). “Assessment of features technology.” *Comput.-Aided Des.*, 23(5), 331-341.

Shah, J., and Mäntyla, M., (1995). *Parametric and feature-based CAD/CAM: Concepts, techniques and applications*, Wiley & Sons Inc., NY.

Staub-French, S., Fischer, M., Kunz, J., Ishii, K., and Paulson, B. (2003). “A feature ontology to support construction cost estimating.” *Artif. Intell. Eng. Des. Analysis Manuf.*, 17, 133-154.

Taneja, S., Akinci, B., Garrett, J. H., Soibelman, L. and East, B. (2011). “Transforming IFC-based building layout information into a geometric topology network for indoor navigation assistance.” *Proc. of the 2011 ASCE International Workshop on Comput. Civil Eng.*, Miami, Florida.

Thomas, H. R., and Zavrski, I. (1999). “Construction baseline productivity: theory and practice.” *J. Constr. Eng. Manage.*, 125(5), 295-303.

Wang, H.-H., Boukamp, F., and Elghamrawy, T. (2011). “Ontology-based approach to context representation and reasoning for managing context-sensitive construction information.” *J. Comput. Civil Eng.*, 25(5), 331-346.

Webster, A. (2010). “A semantic spatial interoperability framework: A case study in the architecture, engineering and construction (AEC) domain.” M.S. thesis, Dept. of Computer Sci., Univ. of British Columbia, Vancouver, Canada.

Woestenenk, K., van Rees, R., Lima, C., Stephens, J., and Bonsma, P. (2000). *bcTaxonomy,* Report, IST-1999-10303-D501, European Commission, Brussels, Belgium.

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

Table 1 Generic attributes to the feature class "component"

Attribute	Explanation	Value Type	Cardinality
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 2 Generic attributes of the feature class "wall"

Attribute	Explanation	Value Type	Cardinality
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 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 4 Generic attributes for the feature class "penetration"

Attribute	Explanation	Value Type	Cardinality
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

805

Figure Captions:

Figure 1: Case studies showing different construction-specific information: (a) wall types in the Chem-Bio building project, and (b) openings and penetrations hand-sketched on drawings for the Engineering Design Center project

Figure 2: A process model for extracting features from a BIM

Figure 3: Specification diagram of features and their relationships using UML (Methodology adapted from Booch et al. 1999)

Figure 4: Class diagram with feature attributes

Figure 5: A wall with corresponding *IFC* and ifcXML representations

(a) 3D model of a wall in *Autodesk Revit*

(b) Hierarchical representation of a wall in *IFC Viewer*

(c) ifcXML representation of a wall

Figure 6: Attributes and relationships with reference paths showing their linkage to a wall object in ifcXML

(a) Attributes

(b) Openings

(c) Connectivity

(d) Curvature

(e) Insertion Point

(f) Direction

(g) Shape 1

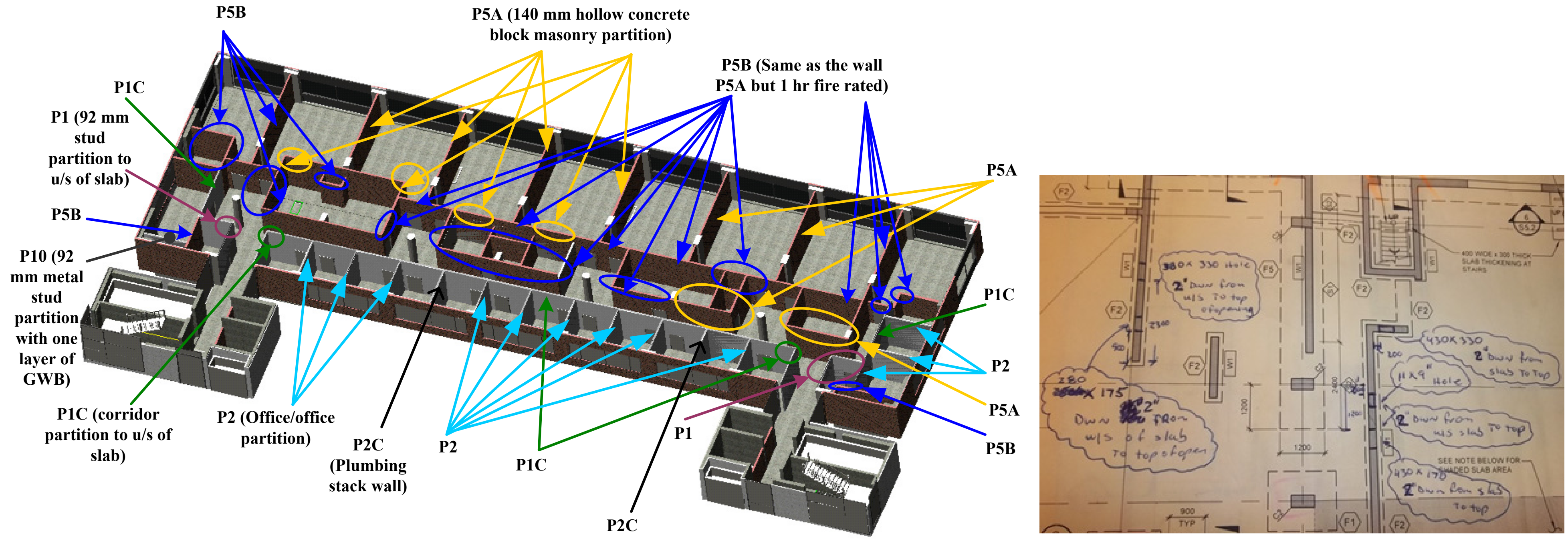
(h) Shape 2

Figure 7: A snapshot of the FBM interface

Figure 8: Screenshots of different query steps for specifying a query

Figure 9 Expert opinion on the relevance/importance of design conditions related to the feature "penetration"

Figure 10: Comparison of level of support (recall) results in percentage: our approach versus state-of-the-art tools



(a) (b)

Figure 1 Case studies showing different construction-specific information: (a) wall types in the Chem-Bio building project, and (b) openings and penetrations hand-sketched on drawings for the Engineering Design Center project

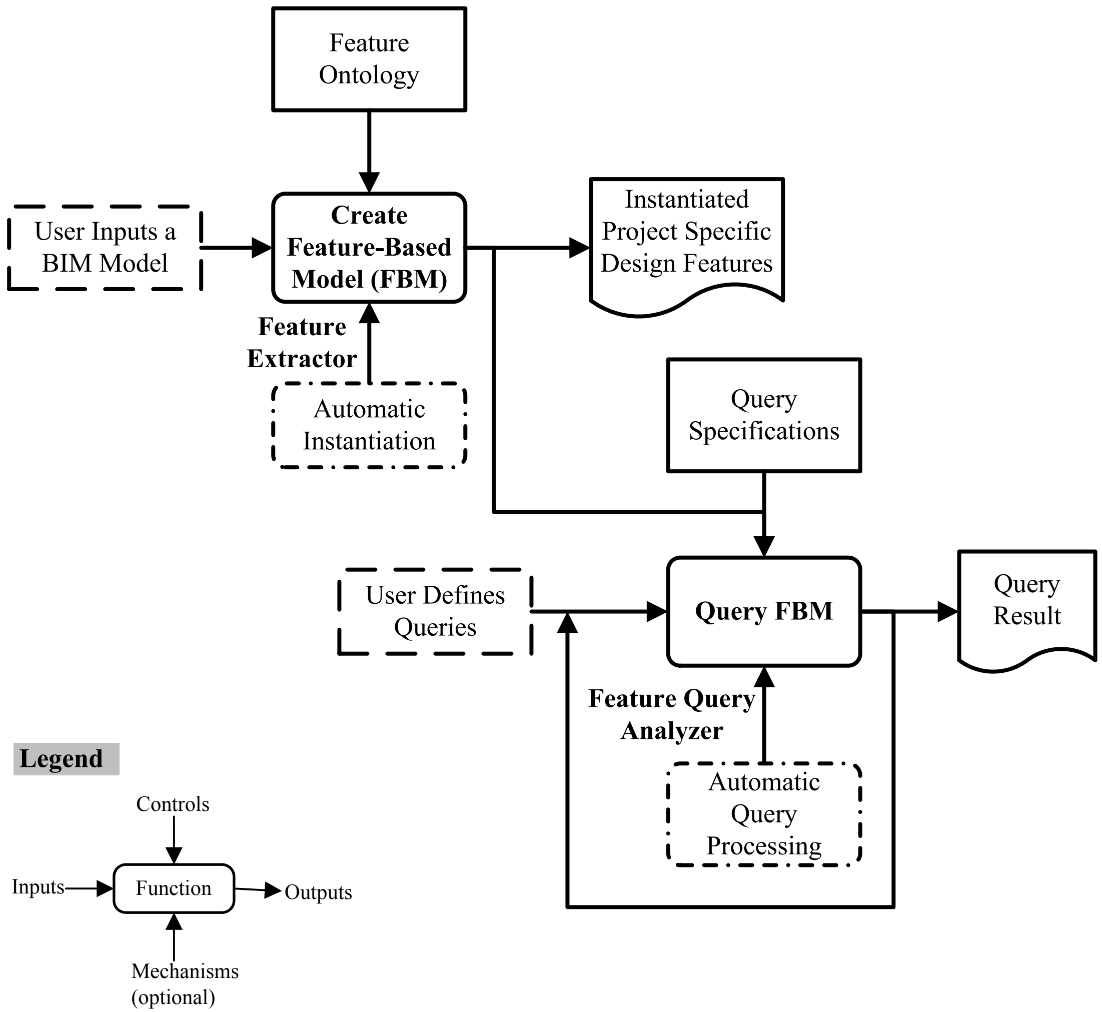


Figure 2 A process diagram for extracting features from a BIM

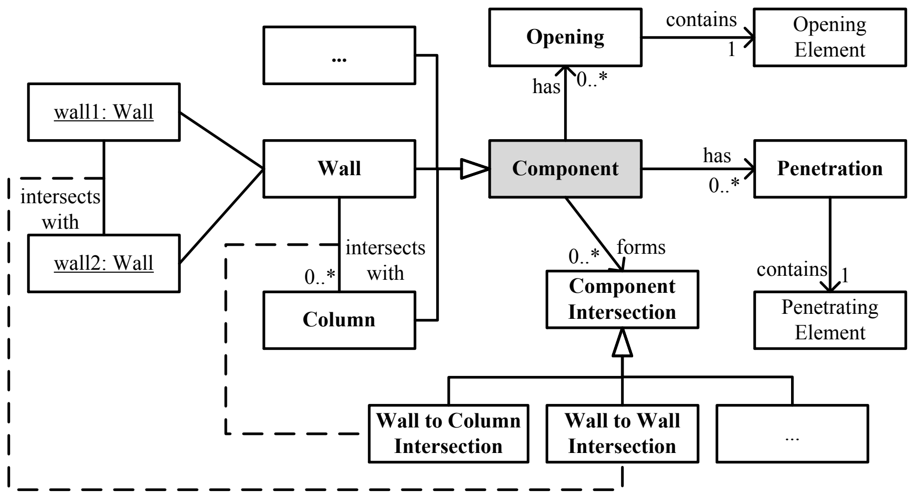


Figure 3 Specification diagram of features and their relationships using UML (Methodology Adapted from Booch et al. 1999)

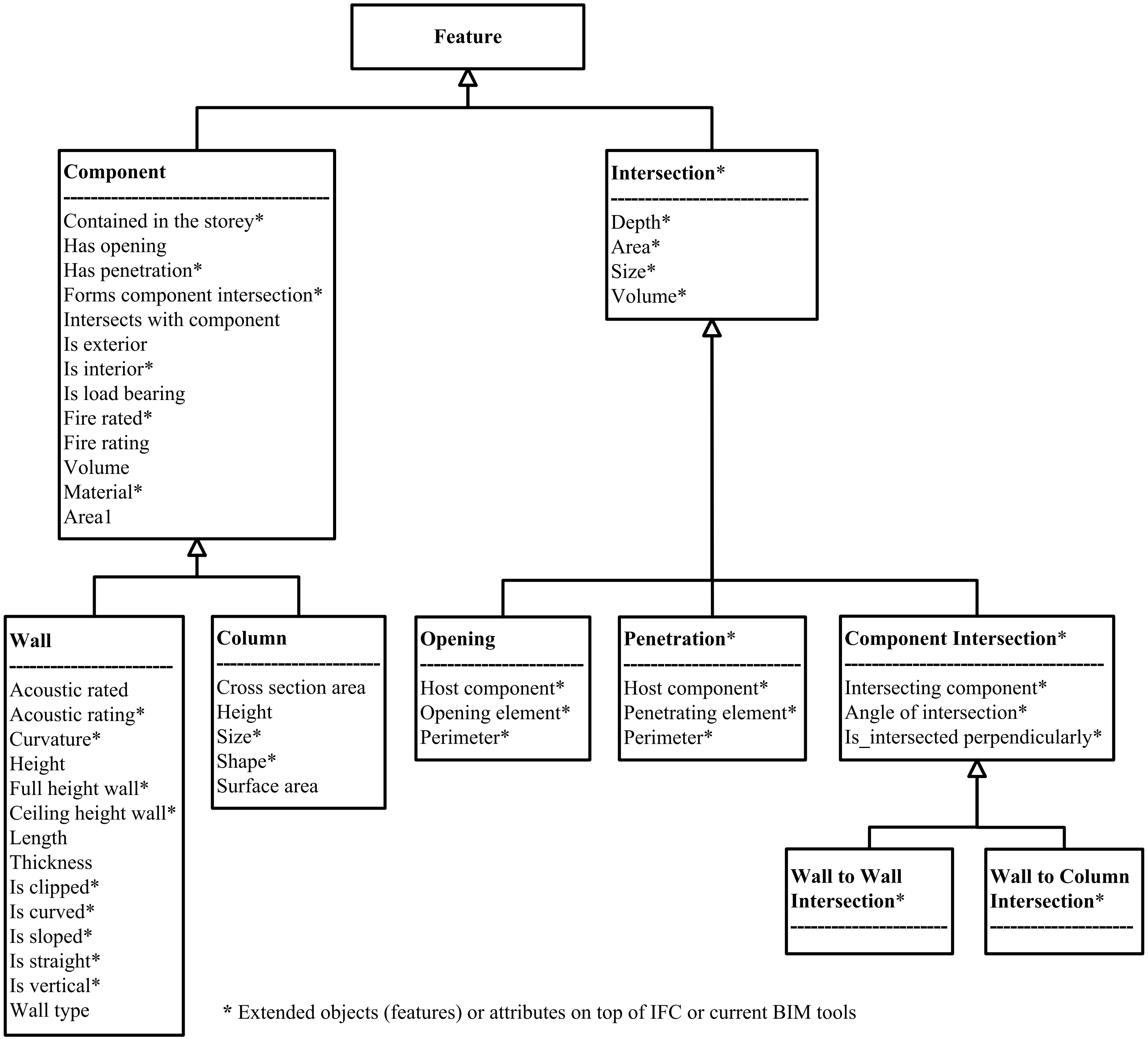
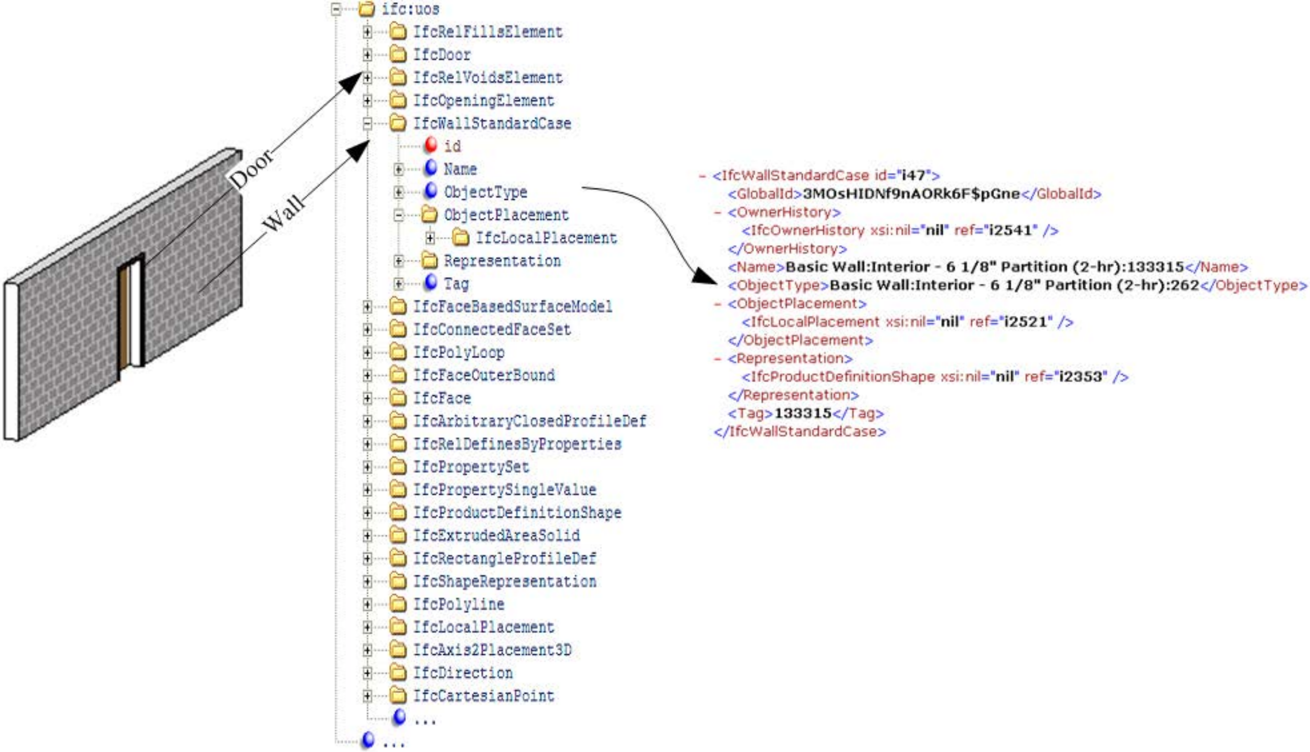


Figure 4 Class diagram with feature attributes



(a) 3D model of a wall in *Autodesk Revit*

(b) Hierarchical representation of a wall in *IFC Viewer*

(c) ifcXML representation of a wall

Figure 5 A wall with corresponding IFC and ifcXML representations

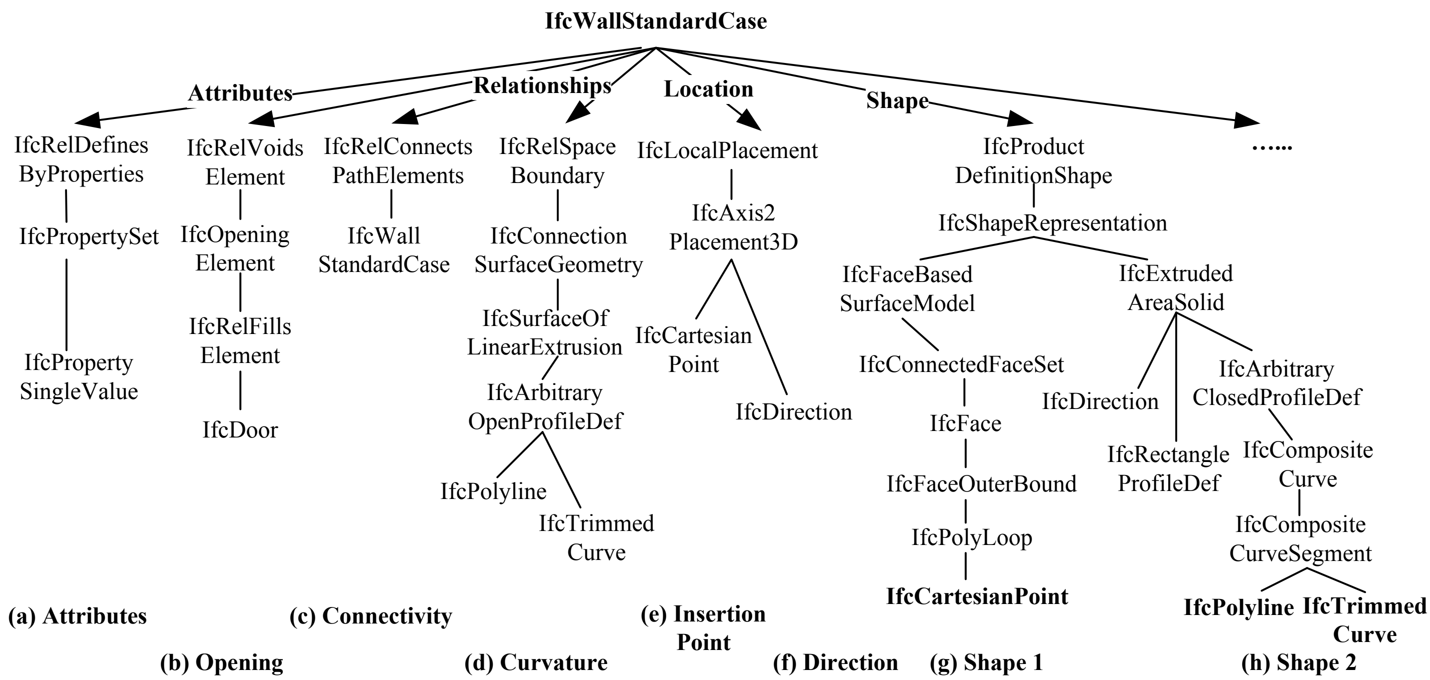


Figure 6 Attributes and relationships with reference paths showing their linkage to a wall object in ifcXML

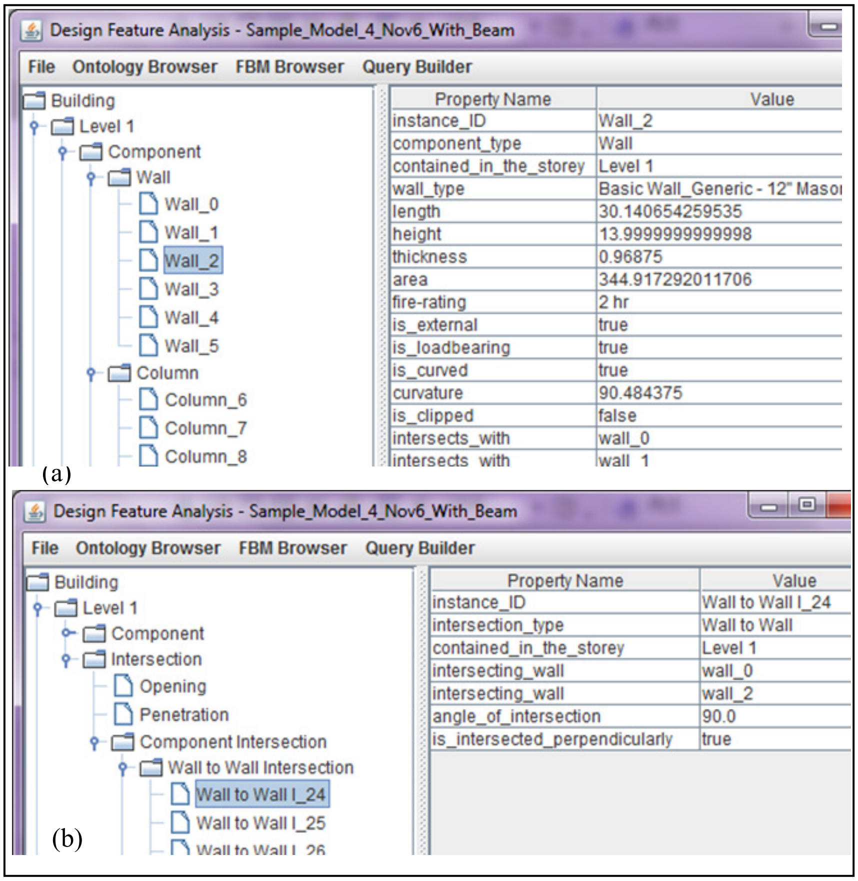


Figure 7 A snapshot of the FBM interface

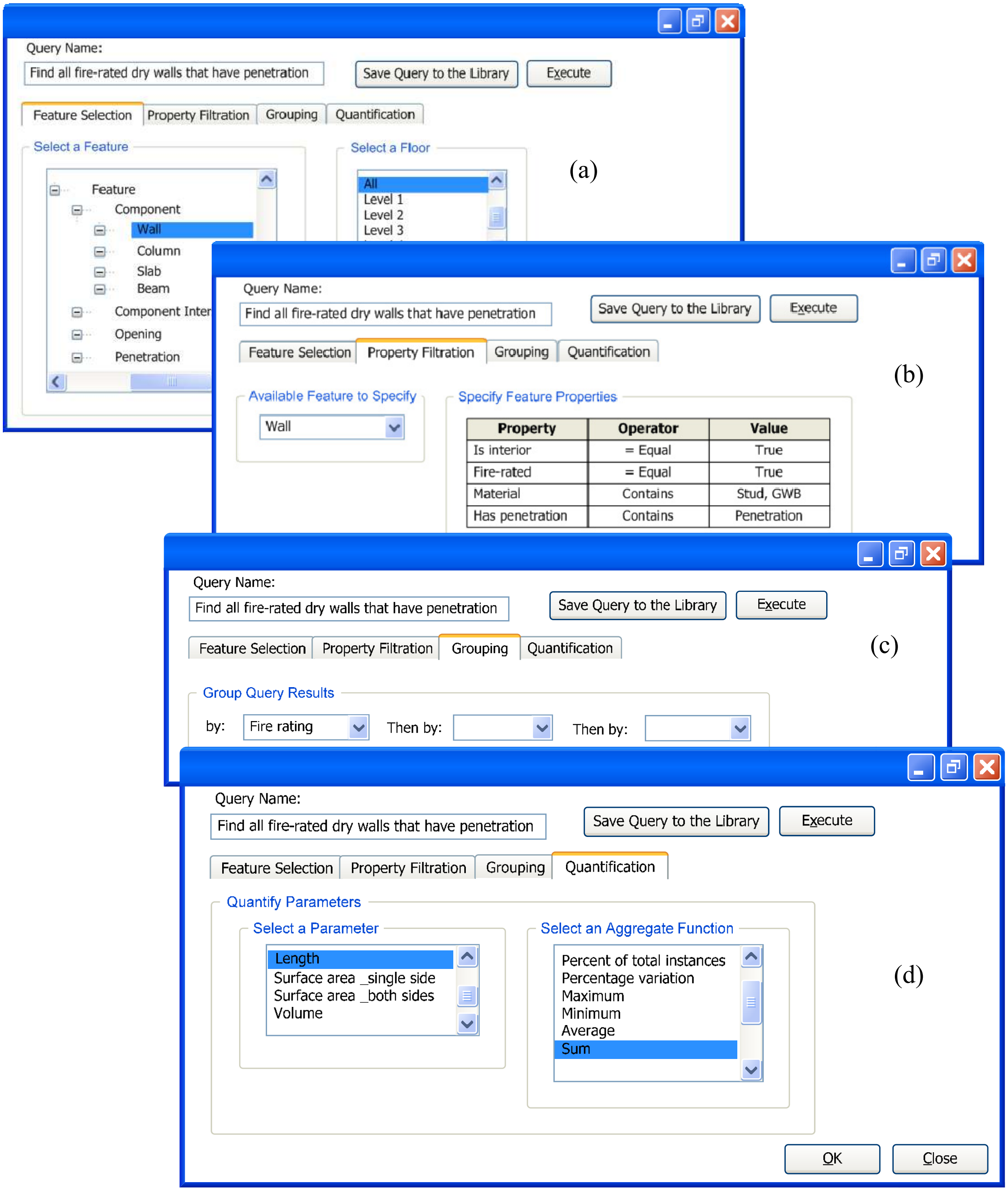


Figure 8: Screenshots of different query steps for specifying a query

Design Conditions	***Relevance/Importance***			
	Significant	***Moderate***	***Little***	***Irrelevant***
Existence of wall/slab penetrations (e.g., duct, conduit, pipe)	■■ □ ○	●		
Size/dimension of penetration	■ □ ○	●		
Depth of penetration		■	■	
Area of penetration		○	□ ■ ●	
Volume of penetration			□	■
Perimeter of penetration		□ ○	■ ●	
Horizontal location of wall penetrations		○ □	■ ●	
Vertical location of wall penetrations	○	■ □	■ ●	
Horizontal location of slab penetrations	■ ○	■ □	●	
Spacing of penetrations			□ ■	●
Uniform size of penetrations	○	□	■ ●	
Uniform location of penetrations	□ ○	■	●	
Uniform spacing of penetrations	■	■ □ ○		●

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

Figure 9 Expert opinion on the relevance/importance of design conditions related to the feature “penetration”

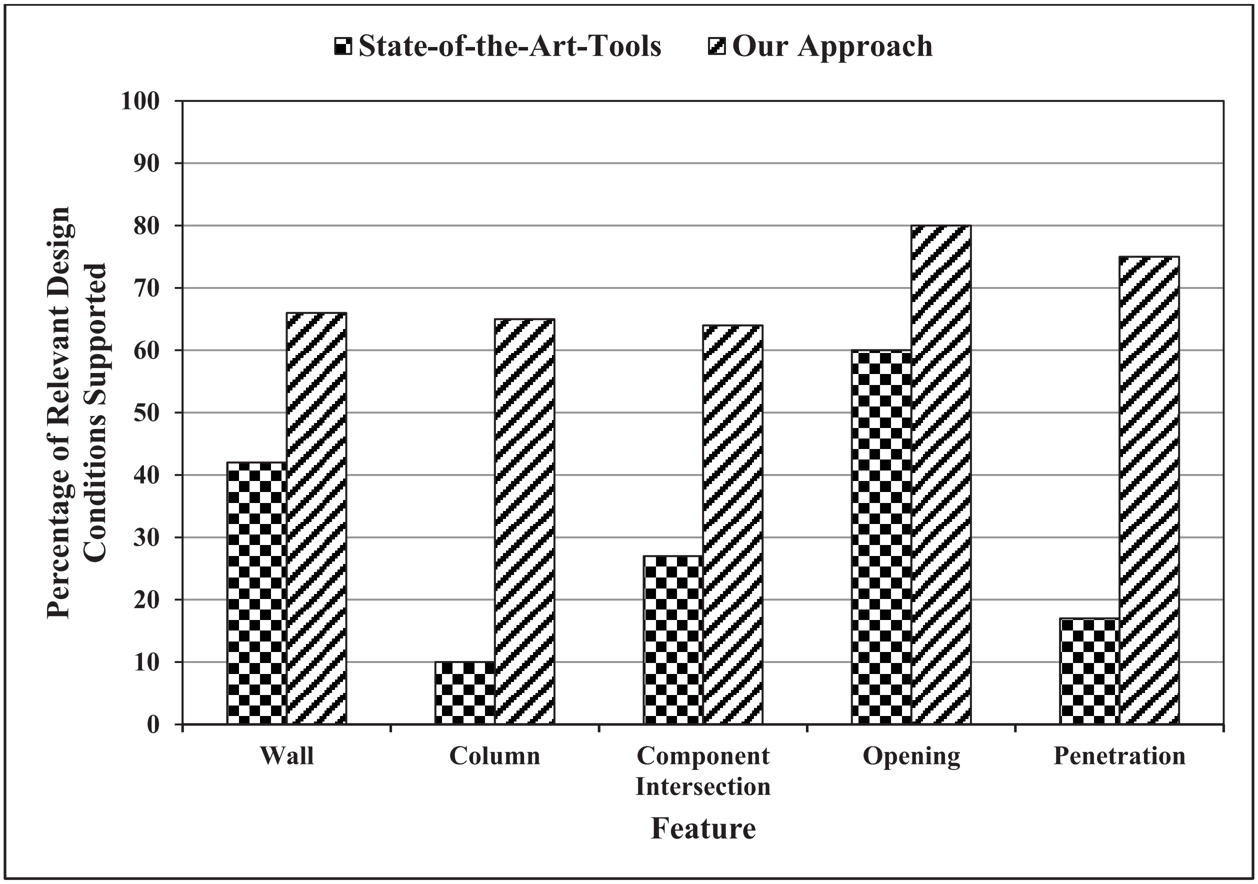


Figure 10: Comparison of level of support (recall) results in percentage: our approach versus state-of-the-art tools