CIFE CENTER FOR INTEGRATED FACILITY ENGINEERING

An Ontology for Relating Features of Building Product Models with Construction Activities to Support Cost Estimating

By

Sheryl Staub-French, Martin Fischer, John Kunz,
Boyd Paulson, and Kos Ishii

CIFE Working Paper #70
July 2002

STANFORD UNIVERSITY

If you would like to contact the authors, please write to:

c/o CIFE, Civil and Environmental Engineering Dept.,
Stanford University
Terman Engineering Center
Mail Code: 4020
Stanford, CA 94305-4020

WORKING PAPER #70

AN ONTOLOGY FOR RELATING FEATURES OF BUILDING PRODUCT MODELS WITH CONSTRUCTION ACTIVITIES TO SUPPORT COST ESTIMATING

Sheryl Staub-French[1], Martin Fischer[2], John Kunz[3], Boyd Paulson[4], Kos Ishii[5]

Abstract

It is the cost estimator's task to determine how the building design influences construction costs. Estimators must account for how and when features of building product models affect the project's activities, resources, and resource productivity rates that form the basis of a cost estimate. Without explicitly representing estimators' rationale for relating product and cost information in the computer, estimators must manually detect when and how particular features in a product model affect construction costs, which is a time consuming and error-prone process. Previous research efforts represent the relationships between product, activity, resource, and cost information, and they formalize construction knowledge about how the building design influences construction resources and their productivity. However, they do not provide a common vocabulary and a formal way for estimators to represent their rationale for relating product and cost information to enable automated support of the cost estimating process. This paper presents the ontology we formalized to represent estimators' rationale for relating product and cost information. The ontology represents the features of building product models that are important to estimators and the estimators' rationale about how the features affect the activities, resources, and resource productivity rates to calculate construction costs. A software prototype that uses the ontology helps estimators to represent their knowledge formally, and to generate and maintain cost estimates quickly and consistently for feature-based product models. Validation studies of use of the prototype system provide evidence for the power and generality of the ontology.

[1] Assistant Professor, Department of Civil Engineering, University of British Columbia, Vancouver, BC, Canada, V6T 1Z4, ssf@interchange.ubc.ca
[2] Associate Professor, Department of Civil and Environmental Engineering and (by Courtesy) Computer Science, Director, Center for Integrated Facility Engineering, Stanford University, Stanford, CA 94305, fischer@stanford.edu
[3] Executive Director, Center for Integrated Facility Engineering, Stanford University, Stanford, CA 94305, kunz@stanford.edu
[4] Professor, Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305, paulson@stanford.edu
[5] Associate Professor, Department of Mechanical Engineering, Stanford University, Stanford, CA 94305, ishii@cdr.stanford.edu

1. Introduction

Understanding how the building design influences construction costs is a challenging task for estimators. Estimators must determine *what* design conditions are important (i.e., incur a cost), *when* they are important, and *how* they affect construction costs. The estimator's knowledge about when and how different design conditions affect construction costs represents the *estimator's rationale* for relating design and cost information. Without explicitly representing the estimator's rationale in the computer, how would cost estimating software know how cost information applies on future projects, or how design changes will affect the existing cost estimate?

Current estimating software helps estimators establish a relationship between a component in a 3D-product model and a cost item in a cost-estimating database to take off quantities automatically when creating a cost estimate (Timberline 2001). However, these quantity-based relationships do not explicitly represent the estimator's rationale for relating the design and cost information. Without automated support to store and use the estimator's rationale, estimators must manually detect when and how particular design conditions affect the project's activities and resources that form the basis of the cost estimate, which is a time consuming and error-prone process. Consequently, the cost impacts of many design conditions may either go undetected or get handled in ad hoc ways, resulting in inefficiencies and inconsistencies in the cost estimating process and the resulting cost estimate. Hence, estimators need a vocabulary for describing their rationale for relating design and cost information, and a formal way to represent that estimating knowledge in the computer to enable automated support of the cost estimating process.

Previous research efforts formally relate product, activity, and resource information to calculate construction costs (Laitinen 1998; Aouad et al. 1994; Froese 1996: Aouad et al. 1997; Stumpf et al. 1996; Slaughter 2000). However, these researchers do not provide a vocabulary or method to represent the practitioners' rationale for when and how to relate this information. Other researchers represent practitioners' rationale for relating product and resource information (Hanna and Sanvido 1990; Fischer 1991; Thomas and Zavrski 2000; Thomas and Sackrakan 1994; Smith and Hanna 1993; Sanders and Thomas 1991). They generalize practitioners' rationale about when and how different design conditions influence construction resources and their productivity. However, the researchers represent this construction knowledge implicitly in computer code. Consequently, they do not allow practitioners to represent their rationale for relating product and resource information. Hence, previous research efforts do not provide a vocabulary to describe practitioners' rationale for relating product and cost information or a framework for practitioners to represent their rationale in the computer.

We formalized a vocabulary in a computer model, i.e., an ontology, to represent estimators' rationale for relating design and cost information. We use the concept of *features* to describe the part of the design that estimators care about and *design conditions* to describe when features are important to estimators. We provide a formal way for estimators to relate features of a building product model with construction activities to enable automated support of the cost estimating process. The ontology helps estimators to establish a richer relationship between product and cost information. Figure 1 shows how the ontology conceptually represents the relationship between product and cost information and contrasts it with the product and cost relationship established using current cost estimating software (the specific example will be explained in the next section). Current cost estimating software represents the relationship between product and cost information by representing the component properties that affect construction costs (Figure 1a). This representation is incomplete because it does not represent the estimator's rationale for how the component properties affect specific cost information, and it does not represent features of the component that are important and how they affect the component's cost. The ontology enriches the representation of the relationship between product and cost information by representing the features of the building product model that are important, when they are important, and how they affect construction activities and their resources to calculate construction costs (Figure 1b).

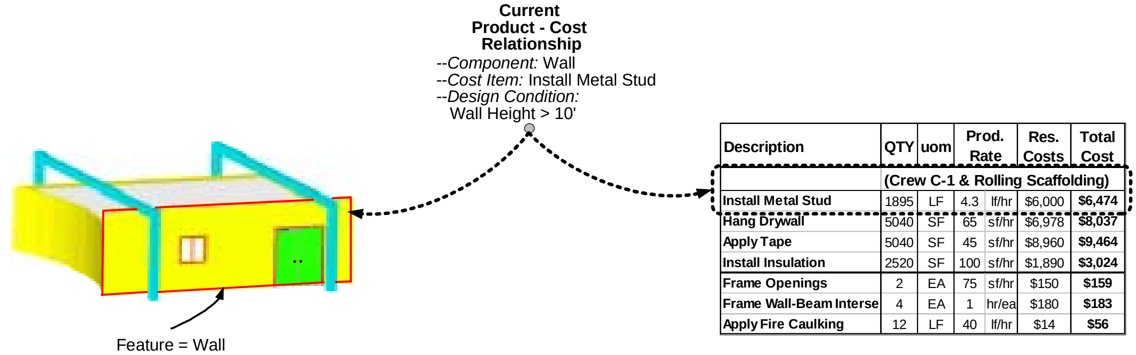


Figure 1a: Current product-cost relationship that represents estimators' rationale by representing the component properties that affect construction costs (e.g., the wall height). This relationship does not represent estimators' rationale about how the component properties affect specific cost information, and how and when features of the component affect construction costs.

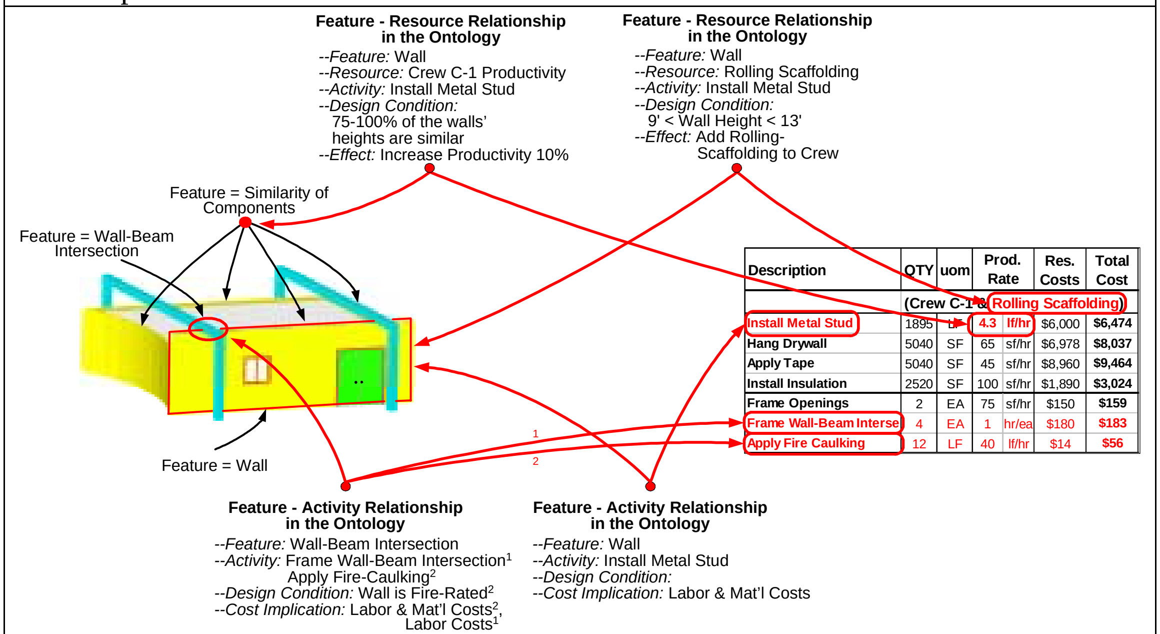


Figure 1b: The ontology represents estimators' rationale for relating product and cost information by representing the features of the building product model that affect construction cost and how the features affect the project's activities, resources, and resources' productivity rates. For example, the ontology represents the relationships between the feature 'Wall-Beam Intersection' and the two activities that are needed to construct the feature: (1) "Frame Wall-Beam Intersection" and (2) "Apply Fire Caulking."

Figure 1: Representation of the product-cost relationship in current cost estimating software and in the ontology formalized in our research. The ontology provides a richer representation of the relationship between product and cost information.

This paper describes the ontology we formalized to provide a common vocabulary for representing estimators' rationale for relating product and cost information. We describe the framework we developed that uses the attributes of the ontology to represent this estimating knowledge independent of a particular project and in a computer-interpretable way to enable automated support of the cost estimating process. The paper concludes with a detailed discussion of how we tested the ontology. The main contributions of the paper lie in the ontology that represents the different types of relationships between features and activities and the framework developed that uses the attributes of the ontology to capture this knowledge from estimators.

1.1 Motivating Case: Current Practice

This section describes a use case that illustrates the requirements of the ontology to represent estimators' rationale for relating product and cost information. The case study is based on drywall estimators' rationale for estimating the labor costs for one of the rooms in an office project shown in Figure 2. The building components in the room are annotated in Figure 2.

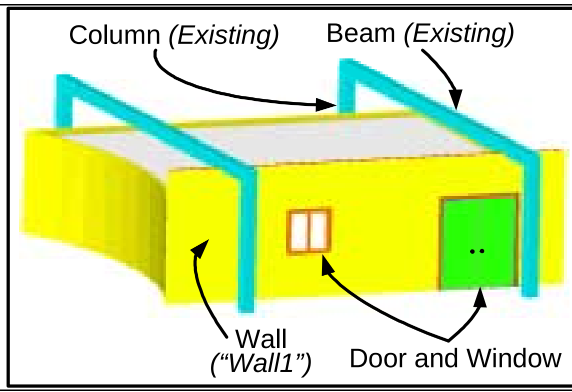


Figure 2: Building components in the office project case study. The drywall estimator is estimating the costs for constructing the four walls shown.

Drywall estimators must identify the design conditions that affect the cost of constructing walls and adjust the activities, resources, and the resources' productivity rates that form the basis of a cost estimate for a particular design. Figure 3 shows the rationale of two drywall estimators illustrating how they adjusted the cost estimate to account for the cost impact of different design conditions.

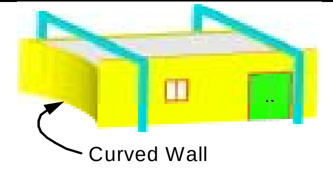
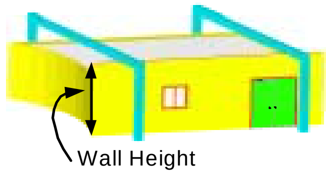
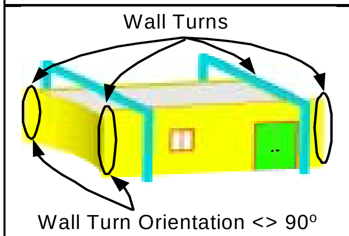
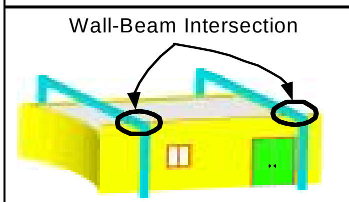
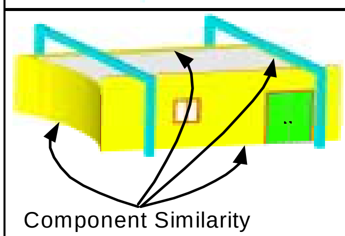
Relevant Design Conditions	*Estimator #1's Rationale*	*Estimator #2's Rationale*
	Reduces the crew productivity rate for all the "Install Metal Studs" activities.	Adds activity "Layout Wall" using a survey crew.
	Uses Rolling Scaffolding in the "Install Metal Studs" activity if the wall height is between 9' - 13', and uses a Scissor-lift if the wall height is greater than 13'. Reduces crew productivity when Rolling Scaffolding or Scissor-lifts are used.	Uses Rolling Scaffolding in the "Install Metal Studs" activity if the wall height is greater than 10'. Reduces crew productivity when Rolling Scaffolding is used.
	Reduces the crew productivity rate for the "Install Metal Studs" activity.	Adds activities "Install Metal Studs" to account for wall turns and "Layout Wall" to account for non-90° wall turns.
	Adds activities "Frame Wall-beam Intersection" to account for the additional labor costs for the unusual framing condition. Adds activity "Apply Caulk" if the intersected wall is fire-rated, which has labor and material cost implications.	Usually reduces the crew productivity rate for the "Install Metal Studs" activity. However, did not notice that the wall intersected the beam when estimating this design.
	If most of the walls have the same height, increases the crew productivity for the "Install Metal Studs" activity by 20%.	If the majority of the walls have the same height and wall type, increases the crew productivity for the "Install Metal Studs" activity by 10%.

Figure 3: Two drywall estimators' rationale for relating product and cost information. Estimators adjust a project's activities, resources, and resources' productivity rates to reflect the cost impact of specific design conditions in the cost estimate. In some cases, the estimators account for the cost impact of design conditions using ad hoc methods. For example, Estimator #2 adjusts the crew productivity rate to account for wall turns and non-90° wall turns, and Estimator #1 adjusts the crew productivity rates of all the "Install Metal Studs" activities to account for one wall's curvature.

The motivating case shows the different design conditions estimators consider and the different ways they adjust the cost estimate to account for them. To provide a formal way to represent estimator's rationale for relating design and cost information, we abstracted the design conditions estimators consider and the different ways estimators adjust activities and resources to reflect their cost impact in the cost estimate.

We use the concept of *features* to describe the design information estimators care about. We refer to components in a building product model, such as walls and columns, as “component features.” Throughout the remainder of this paper, the terms “component feature” and “component” will be used interchangeably. We refer to features that result from the intersection of two components, such as openings and turns, as “intersection features.” We refer to features that result from the similarity of components as “macro features.”

Design conditions describe when a particular feature affects construction costs. Design conditions can be based on properties of component features (e.g., the curvature and height of the wall), groupings of component features (e.g., the grouping of walls based on component similarity), the existence of intersection features (e.g., the existence of turns and openings), and properties of intersection features (e.g., the orientation of wall turns).

The use case demonstrates that features may have multiple effects on construction costs. Features drive the requirement for activities, features affect when a resource is appropriate in an activity, and features affect a resource’s ability to execute an activity effectively. Estimators have different preferences for what features are important, when features are important, and how they affect the project’s activities, resources, and resources’ productivity rates. Activities also have different cost implications. For example, Estimator #1 accounts for the labor costs of the “Frame Wall-beam Intersection” activity but excludes the material costs because the material costs are included in the “Install Metal Stud” activity (Figure 3). On the other hand, the activity “Apply Caulk” has material and labor cost implications.

To represent estimators’ rationale for relating product and cost information, estimators need a vocabulary for describing:

- The component and intersection features and feature properties that affect construction costs.
- The representation of component similarity, including the properties of the components that need to be similar and the degree of similarity to be achieved.
- When activities are required to construct a specific feature and the labor and material cost implications of the activity.

- When features limit the applicability of resources in an activity.
- When features influence resources' productivity rates and how the productivity rates should be adjusted to account for them.

Current cost estimating software does not represent the different aspects of estimators' rationale for relating product and cost information. Without representing the estimators' rationale explicitly in the computer, estimators have to manually identify most relevant design conditions and adjust the project's activities and resources accordingly. For a large project, this is a time-consuming and error prone task that must be repeated each time the design changes. Consequently, estimators often employ ad hoc methods (e.g., Estimator #1 reduces the crew productivity to account for wall turns and non-90° wall turns), overlook the cost impact of features and feature properties (e.g., Estimator #2 does not account for the cost impact of the wall-beam intersection), or account for the cost impact of features and feature properties inconsistently (e.g., Estimator #1 must remember how much she reduced crew productivity to account for wall curvature and how she represented and accounted for component similarity to consistently estimate the next project). Hence, the lack of a formal and computer-interpretable representation of estimators' rationale limits the automated support estimators receive and prevents them from accounting for the cost impacts of features explicitly, leading to inconsistencies and inefficiencies in the cost estimating process.

Estimators need automated support to account for the cost impact of features and feature properties explicitly and consistently when generating and maintaining cost estimates from 3D product models. To enable automated support, estimators' rationale needs to be represented formally and in a computer-interpretable way. Our research addresses this need by providing a vocabulary to describe estimators' rationale for relating product and cost information and by formalizing the representation of that knowledge in the computer. The ontology enables estimators to establish a richer relationship between product and cost information by representing estimators' rationale about how features of building product models affect the requirement for activities and the execution of resources in an activity. The next section describes the research goals in more detail.

1.2 Research Goals

The goals of this research were to formalize an ontology that abstracts the common attributes of estimators' rationale for relating product and cost information and to provide a formal and computer-interpretable representation of estimators' rationale that enables computer-based support of the cost estimating process. The use case illustrates that the ontology needs to be formal, general, and flexible to represent estimators' rationale:

(1) Formal: Estimators need a structured way to relate features of building product models to construction activities and their resources to represent their rationale in the computer. The formal representation should include all the attributes necessary for estimators to describe their rationale (Figure 3), and prevent estimators from using ad hoc methods to account for the cost impact of features. Hence, the formal representation should enable estimators to represent their rationale for when and how features affect the requirement for activities and the execution of resources in an activity. By representing this knowledge formally, estimators should be able to account for the cost impact of features explicitly and consistently.

(2) General: Estimators need to represent their rationale independent of a specific project or product model. A generic and computer-interpretable representation of estimators' rationale enables estimators to leverage their estimating knowledge to generate and maintain cost estimates for a given product model. The generic representation of estimators' rationale can be leveraged to automatically generate a project-specific estimate for a given product model or product model change. Estimators can represent their rationale once and reuse it to consistently generate new estimates for a given product model or maintain estimates by identifying the cost information affected by a given product model change. Finally, the representation of estimators' rationale also has to be general enough to support cost estimating of different domains.

(3) Flexible: The use case demonstrated that estimators have different preferences for accounting for the cost impacts of different features (Figure 3). The ontology must be flexible enough to represent estimators' varied preferences for relating product and cost information. Estimators should be able to represent their preferences for what features are important to them, when they are important, and how they affect the project's activities and resources.

The ontology formalized in this research to represent estimators' rationale for relating product and cost information meets these criteria. The ontology provides a vocabulary that abstracts estimators' rationale for relating features of building product models and construction activities and represents this estimating knowledge independent of a particular project. The next section describes the related research background.

2. Related Research Background

To represent estimators' rationale about when and how particular product features influence construction activities and their resources to calculate construction costs, this research combines and extends previous research in construction cost estimating, activity modeling, and product modeling.

2.1 Prior Research on Construction Cost Estimating

The motivating case demonstrated that estimators need a vocabulary for describing their rationale and a formal way to represent their rationale for relating product and cost information in the computer. Many research efforts have developed formal systems that relate components, activities, and resources to calculate construction costs from a given product model (Laitinen 1998; Aouad et al. 1994; Froese 1996; Aouad et al. 1997; Stumpf et al. 1996; Slaughter 2000). These research efforts demonstrate the feasibility of representing the relationships between product and cost information generically and instantiating them for a given product model, which is a goal of this research. However, they are limited in their ability to represent estimators' rationale for how and when to relate product, activity, and resource information. Moreover, they do not represent intersection features and component similarity and their affects on a project's activities and associated resources. Hence, they do not provide a vocabulary and a framework for estimators to represent their rationale for relating features of building product models with construction activities and their resources.

Many research efforts have focused on understanding how the building design influences the applicability of certain construction resources in executing an activity (Hanna and Sanvido 1990; Fischer 1991; Udaipurwala and Russell 2000; Fischer and

Tatum 1997). For example, Fischer (1991) identified that the applicability of flying forms for concrete slabs is limited by a 20' maximum floor-to-floor height. Hanna and Sanvido (1990) provide guidelines that limit the applicability of different formwork systems based on component similarity, such as “conventional form systems…can handle variation of column wall/size and location.” They identify the design conditions that influence the applicability of a resource in an activity similar to this research. However, they do not provide a flexible approach that allows estimators to represent their preferences for when a construction resource is appropriate in an activity. Hence, previous research efforts do not provide a vocabulary or a framework for estimators to represent their rationale for when a resource is appropriate in an activity.

Other research studies have focused on the influence of the building design on resource productivity (Thomas and Zavrski 2000; Thomas and Sackrakan 1994; de Sousa and Thomas 1996; Smith and Hanna 1993; Sanders and Thomas 1991; Akbas and Fischer 1999). Thomas and Zavrski (2000) use a work content (WC) scale ranging from 1-5 to represent design complexity, with 5 being the most complex. A WC-1 rating represents long straight walls with few openings, while a WC-5 rating includes walls with numerous corners and walls not at 90° with limited similarity in the scope of work. These research efforts demonstrate that component similarity, intersection features, such as openings and turns, and component feature properties, such as height and curvature, affect production parameters. However, they do not represent and account for the production impact of these features explicitly. They adjusted the resources’ productivity rates for the entire project based on the design complexity captured in the work content rating. The rating criteria are ambiguous and does not account for the production impact of each feature explicitly. Moreover, they do not allow estimators to represent their preferences for how resource productivity rates should be adjusted for component similarity, intersection features, and component feature properties. Hence, they do not provide a vocabulary or a framework that allows estimators to represent their rationale for how and when a resource’s productivity rate should be adjusted for component feature properties, intersection features, and component similarity.

2.2 Prior Research on Activity Modeling

The case demonstrated that estimators need to represent their preferences for how features influence the requirement for activities and the execution of resources in an activity. The case also showed that some activities have material and resource cost implications, such as the “Apply Caulk” activity, and others only have a resource cost, such as the “Frame Wall-Beam Intersection” activity. Consequently, estimators need a vocabulary for describing when an activity is required and how an activity impacts material and resource costs and a formal way to represent that knowledge in the computer.

Prior research efforts in activity modeling provide a formal way to relate the required activities to the component feature being installed and the method being used (Aalami 1998; Darwiche et al. 1988; Froese and Rankin 1998; Jagbeck 1994; Stumpf et al. 1996; Udaipurwala and Russell 2000). They also represent construction activities generically using product objects, actions, and resources, which our research builds on. However, these research efforts do not provide a formal way for practitioners to represent their preferences for the resource and material cost implications of the activity, or the activities required for intersection features. Moreover, they do provide a formal way for practitioners to represent their rationale for how certain features constrain a specific resource and limit its applicability or impact its efficient execution in an activity. Hence, previous research in activity modeling does not provide a formal way for practitioners to represent their rationale for when intersection features require activities, and when features affect the applicability of resources in an activity or impact resources’ productivity rates when executing an activity.

2.3 Prior Research on Product Modeling

The motivating case demonstrated that different features influence construction costs. The intersection features “openings”, “turns”, and “wall-beam intersections” required the execution of additional activities. Component similarity affected the crew productivity of the “Install Metal Stud” activity. The features of building product models that affect the cost of constructing a specific component feature need to be represented explicitly. Product features are used extensively in manufacturing to describe the geometric forms or entities in a product model that are important in some aspect of the manufacturing process (i.e., manufacturability evaluation and flexibility analysis) (Cunningham and Dixon 1988; Shah 1991). Our research represents the product features that are important for cost estimators of building construction.

In summary, previous research efforts explicitly relate components in a product model to construction activities and their resources to calculate construction costs. These researchers also identify the design conditions that influence the applicability of construction resources and their productivity. They formalize project-independent construction knowledge to reason about a given product model. However, they tightly link product model representation and knowledge representation. Consequently, estimators cannot represent their rationale for relating product and cost information. Hence, they do not provide a vocabulary and a formal way for estimators to represent their rationale for how and when features affect the requirement for activities and the execution of resources in an activity.

3. An Ontology for Relating Features and Activities

So far the paper has demonstrated the need to provide a vocabulary for representing estimators' rationale for relating product and cost information. Estimators also need a framework for representing their rationale formally and in a computer-interpretable way to enable automated support of the cost estimating process. As our tests have shown (Staub-French 2002), the formal and computer-interpretable representation of estimators' rationale enables estimators to generate and maintain construction cost estimates more completely (i.e., less ad hoc and fewer omissions) and more consistently than the current product-cost relationship represented in current tools.

We developed the ontology by identifying the common attributes estimators use to describe their rationale for how the building design influences construction costs. We reviewed previous research in this area and interviewed 14 different cost estimators from five different domains. We interviewed two general contractors and twelve subcontractors that self-perform construction work on drywall, structural concrete, ductwork, process piping, and electrical systems. We performed three case studies on two drywall construction projects and one concrete column construction project. We formalized the different vocabularies used by estimators to describe the design conditions that affect construction costs and to specify the adjustments to a project's activities, resources, and resources' productivity rates to account for the design conditions.

The ontology extends existing formalisms of construction processes that define activities as *objects* <O>, *actions* <A>, and *resources* <R> (Darwiche et al. 1988; Aalami 1998). We extend this formalism by representing the *feature* <F> that drives the requirement for an activity (Staub-French et al. 2002a). The ontology represents estimators' rationale about how and when to assemble activities consisting of <FOAR> tuples for a given product model to calculate construction costs, as shown in Figure 4. *Activity Specifications* represent estimators' rationale about how and when activities, represented as <AO> pairs, are required for different types of features <F>. *Resource Specifications* represent estimators' rationale about when resources <R> are required for a given activity <AO> based on features <F>, and when and how to adjust resources' productivity rates based on features <F>. Appendix B contains a detailed diagram of all the classes, attributes and relationships represented in the ontology.

We implemented the ontology in a software prototype called Activity-based Cost Estimating (ACE). ACE provides the framework that uses the ontology to represent estimators' rationale for how and when features influence construction costs independent of a particular project. The computer-interpretable representation of estimators' rationale enables estimators using ACE to generate and maintain cost estimates for a given product model. ACE analyzes a given product model to generate activities that know what feature requires their execution, what resources are being used and why, and how much the activities' execution costs (Staub-French et al. 2002b).

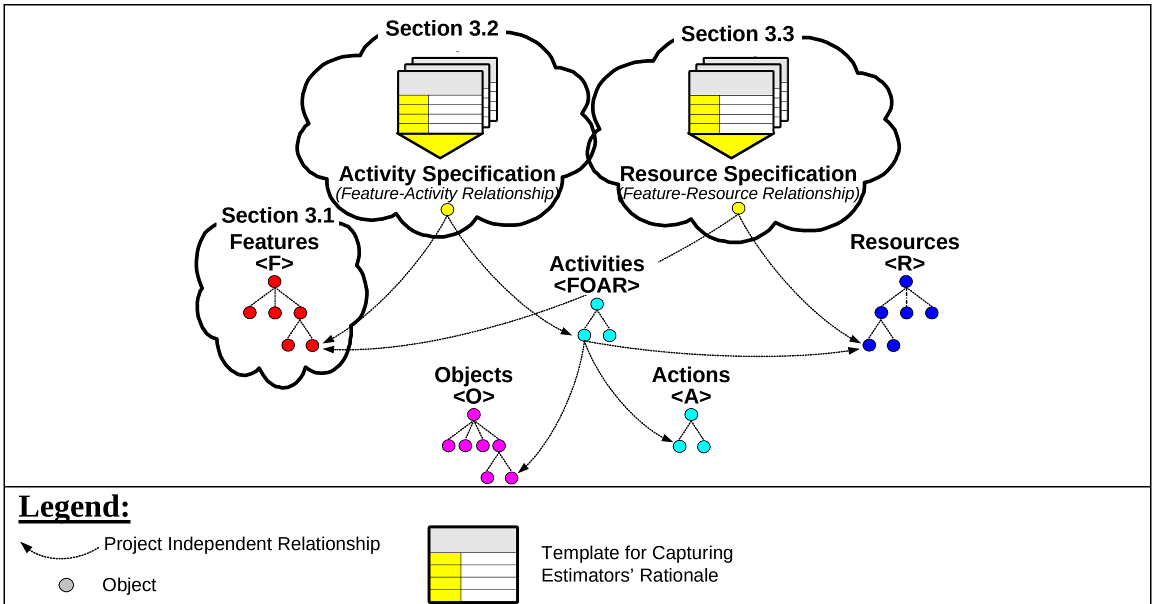


Figure 4: The ontology represents activities consisting of <FOAR> tuples to calculate construction costs for a given product model. The ontology represents the estimators' rationale about how and when to assemble <FOAR> tuples generically using Activity and Resource Specifications. Each activity knows what feature <F> requires its execution and why (Activity Specification), what object <O> it is acting on, what action <A> is being performed, what resources <R> are executing the activity and when they are required (Resource Specification), and how to adjust the resources' productivity rates for particular features (Resource Specification).

The next sections describe the vocabularies formalized in the ontology to represent features and estimators' rationale about how and when features influence construction costs. We describe the common attributes of estimators' rationale and the framework developed for ACE to capture this knowledge from estimators. Section 3.1 describes how we represent features to support feature-based cost analysis of building construction. Section 3.2 describes how Activity Specifications are represented in the ontology to describe estimators' rationale about the activities required for a given feature and the framework developed to capture that knowledge from estimators. Section 3.3 describes how Resource Specifications are represented in the ontology to describe estimators' rationale about the affect of features on resources executing an activity and the framework developed to capture that knowledge from estimators.

3.1 Representing the Features that Affect Construction Costs

The case study demonstrated that different design conditions affect the requirement for activities and the execution of resources in an activity. The "wall-beam intersection" created an unusual framing condition and the *orientation* of the "wall turn" affected the wall layout. To help estimators generate and maintain cost estimates, cost estimating software must represent the design conditions that are important to cost estimators. To represent the design conditions that affect construction costs, we abstracted the different vocabularies used by estimators to describe the different design conditions that are important to cost estimators (Staub-French et al. 2002a).

The motivating case showed that different types of design conditions affect construction costs. Estimators consider properties of components, groupings of components, intersections of components, and properties of component intersections when creating cost estimates (Figure 3). We formalized a feature ontology that classifies features and represents the sets of features and properties that affect costs for a specific construction domain. Using the feature ontology, estimators can create features and specify the features and properties that affect a particular component's construction costs. The ontology represents these features and feature properties independent of a particular product model. By representing the features that affect construction costs generically, estimators can select the features that matter to them and represent their rationale for how and when features affect construction costs independent of a particular project.

Estimators select the features and feature properties that affect construction costs for the component feature they are estimating. The feature ontology represents these features and feature properties in *feature sets* and *property sets* respectively. Each feature has the attribute *property set* that represents the feature's properties that affect construction costs. Consequently, each component knows what features and properties influence the cost of its construction. Figure 5 shows the features and properties that were important to drywall estimators estimating the walls in the motivating case. The main contributions lie in the formalization of the feature ontology and the framework developed to capture this knowledge from estimators.

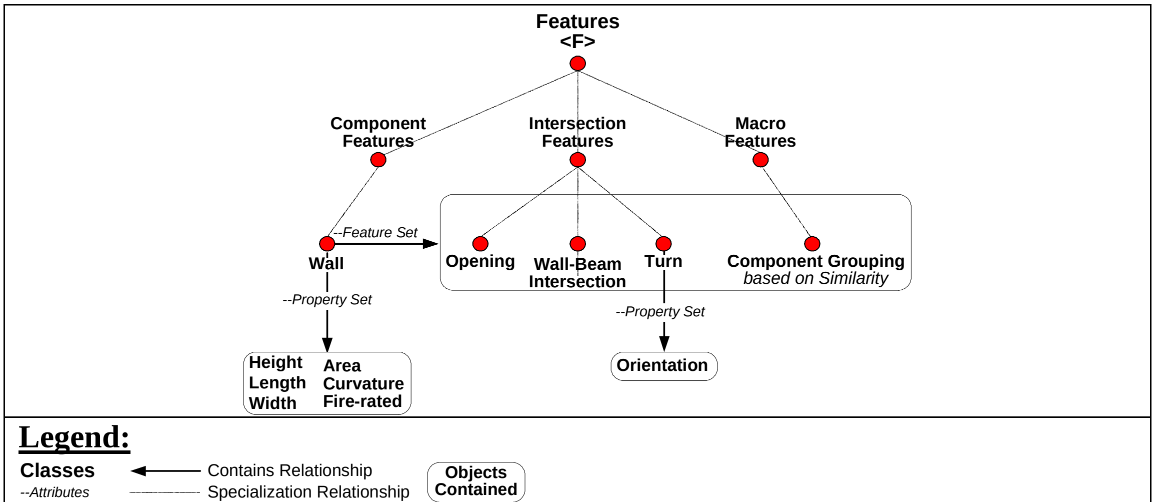


Figure 5: The ontology classifies features and represents the sets of features and properties that affect costs for a specific construction domain. Estimators leverage the feature ontology to create features and specify the features and properties that affect a particular component's construction costs, as shown for the drywall estimator in the motivating case.

The next sections describe the common attributes of Activity Specifications and how estimators represent their rationale for relating features and activities using Activity Specification templates.

3.2 Activity Specifications Relate Features to Activities

The case study demonstrated that different design conditions require the field execution of different construction activities. In the motivating case, the estimators identified activities for constructing the component feature "wall" and the intersection feature "wall-beam intersection", and for laying out the wall when the *orientation* of the "turn" was not 90° (Figure 3). Consequently, component and intersection features drive the need for activities, and feature properties can limit the requirement for activities when constructing a particular feature. Moreover, the motivating case showed that different activities can have different cost implications. For example, the "Apply Caulk" activity has material and resource costs, while the "frame wall-beam intersection" activity only has resource costs. Therefore, estimators need to represent the component or intersection feature that drives the activity, the activity that should be instantiated if the feature exists, the design condition that limits the applicability of the activity, and how the activity affects the material and resource costs of executing the activity.

3.2.1 *Common Attributes of Activity Specifications*

Figure 6 shows the common attributes of Activity Specifications represented in the ontology and examples from the motivating case. The common attributes of Activity Specifications are described below.

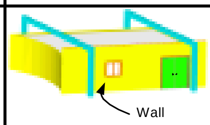
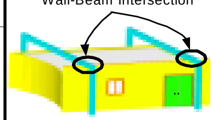
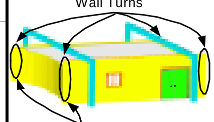
	Attributes of Activity Specifications						
	Component Feature	**Driving Feature**	**Action**	**Object**	**Cost Implication**	**Product Required**	**Design Condition**
Examples of Activity Specifications	Wall		Install	Metal Stud	Material and Resource	Metal Stud	
			Hang	Drywall	Material and Resource	Drywall	
			Install	Insulation	Material and Resource	Insulation	
			Apply	Tape	Material and Resource	Tape	
	Wall	Wall-Beam Intersection	Frame	Wall-Beam Intersection	Resource		
	Wall		Apply	Fire Caulking	Material and Resource	Caulk	***Constrained Feature:*** Wall ***Feature Property:*** Fire-rated **Operator:** = **Value:** True
	Wall	Wall Turns	Install	Metal Stud	Material and Resource	Metal Stud	
	Wall	 Wall Turn Orientation <> 90°	Layout	Wall	Resource		***Constrained Feature:*** Turn ***Feature Property:*** Orientation ***Operator:*** <> ***Value:*** 90°

Figure 6: Attributes of Activity Specifications and examples from the motivating case. Activity Specifications represent estimators' rationale about how and when to relate features <F> to activities represented as <AO> pairs. ACE uses this estimating knowledge to determine when to add activities for features in a given product model, and how to calculate an activity's costs.

(1) ***Component Feature:*** The component feature being estimated. The "wall" is the component feature being estimated in the motivating case.

(2) ***Driving Feature*:** The component or intersection feature that requires the activity. Examples of driving features from the motivating case are the component feature "wall" and the intersection feature "wall-beam intersection." The intersection features for this attribute are based on the feature set specified for the component feature (1). Future extensions of the ontology may need to represent other types of features that require the execution of additional activities. For example, the proximity of a duct run to a pipe run can require the execution of additional activities due to increased congestion. These types of "congestion" features are not represented in the current ontology.

(3) ***Object*:** The physical building object being acted on in the activity (Darwiche et al. 1988; Aalami 1998). Examples of objects being acted on in activities from the

motivating case are the *metal studs* in the "Install Metal Studs" activity, and the *wall-beam intersection* in the "Frame Wall-beam Intersection" activity. Objects can be component features (e.g., "wall"), intersection features (e.g., "wall-beam intersection"), subcomponents of component features (e.g., "metal stud"), and product resources (e.g., "formwork"). Our research extends the types of objects represented in activities by representing intersection features as objects that can be acted on in an activity.

(4) ***Action*****:** The construction action or operation being performed in the activity (Darwiche et al. 1988; Aalami 1998). In the motivating case, "install" and "frame" were the actions performed on the "metal stud" and "wall-beam intersection" objects respectively.

(5) ***Cost Implication*****:** The material and resource cost implications of the activity. In the motivating case, the "Apply Caulk" activity has material and resource cost implications, while the "Frame Wall-beam Intersection" activity has resource cost implications.

(6) ***Design Condition*****:** The design condition represents when the driving feature (2) requires the activity. In the motivating case, the "layout wall" activity is only required for non-90° wall turns. The common attributes for representing design conditions for intersection and component features are:

a) ***Constrained Feature*****:** The feature that limits the requirement for the activity, which can either be the component feature (1) or the driving feature (2). In the example above, the constrained feature is the wall "turn."

b) ***Constrained Feature Property*****:** The property of the constrained feature (6a) that is limiting the applicability of the activity. In the example above, the "orientation" of the wall "turn" is constrained. The feature properties for this attribute are based on the property set specified for the constrained feature (6a).

c) ***Operator*****:** The operator represents how the constrained feature property will be compared. In the example above, the operator is "not equal to" because the "layout wall" activity is needed when the orientation of the wall turn is "not equal to" 90°. The ontology currently represents the following operator values: less than (<), greater than (>), not equal to (<>), equal to (=), less than or equal to (<=), greater than or equal to (>=).

d) ***Value*:** Value of constrained feature property that will be compared when analyzing a given product model. From the example above, the value of the property is "90°" because the "layout wall" activity is needed when the orientation of the wall turn is not equal to "90°."

The common attributes of Activity Specifications represent estimators' rationale about the activities required for different features, when the activity is required to construct the feature, and the material and resource cost implications of the activity. ACE uses the attributes of Activity Specifications to provide a flexible environment that helps estimators represent their estimating knowledge consistently and in a project-independent and computer-interpretable way.

3.2.2 Activity Specification Templates

We developed a framework to represent and leverage estimators' rationale for relating product and cost information to provide automated support for the cost estimating process. Figure 7 shows the Activity Specification templates implemented in ACE that use the attributes of the ontology to help estimators represent their rationale for when features affect the requirement for activities and how the activities impact construction costs.

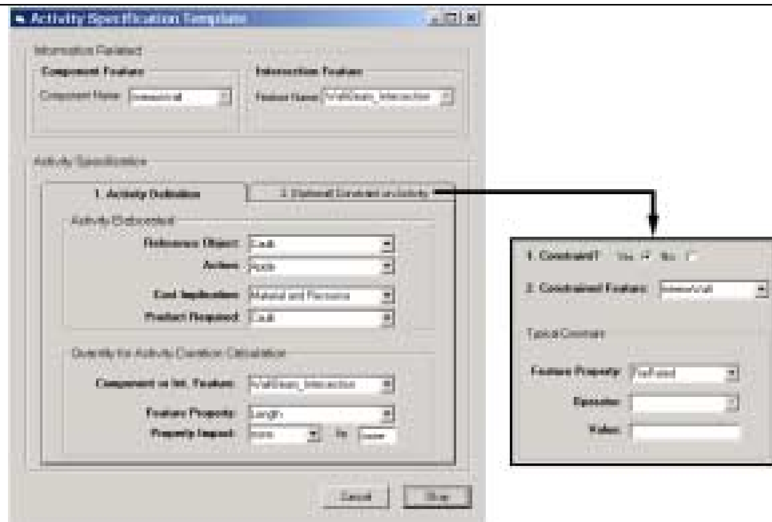


Figure 7: In the ACE user interface, Activity Specification templates use the attributes of the ontology to represent an estimator's rationale for when an activity is required for a specific feature and how it affects material and resource costs. The example Activity Specification template shows an estimator's preference to add the activity "Apply Caulk" for the feature "Wall-beam Intersection" if the intersected wall is fire-rated.

Estimators represent their preferences for the activities to add for a particular feature by filling out the Activity Specification template. In this way, ACE represents this estimating knowledge in a project-independent and consistent way. Estimators input their rationale once, and ACE reuses this knowledge to create project-specific activities when generating and maintaining cost estimates from 3D product models. Consequently, ACE enables estimators to account for the cost impact of features consistently throughout a project and across projects. Estimators representing their rationale in ACE cannot use the ad hoc methods described in the motivating case. For example, estimators cannot account for the production impact of wall turns by fudging the crew's productivity (Figure 3). Rather, estimators using ACE must account for intersection features explicitly by representing the activities that need to be executed. ACE uses this knowledge to determine when to create activities for particular features and how to calculate their construction costs for a given product model or product model change. In consequence, each activity generated by ACE knows what feature requires its execution and why, and what the material and resource cost implications of the activity are.

The estimators' rationale represented in the instances of Activity Specifications enables ACE to generate and maintain activities for particular features in a given product model. ACE then assigns the appropriate resources to the activities based on the Resource Specifications, which will be discussed next.

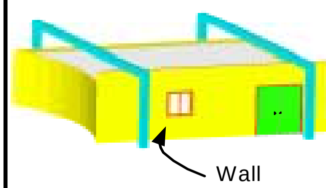
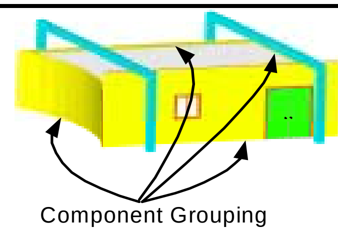
3.3 Resource Specifications Relate Resources to Activities

Many resources can be appropriate for executing a given activity. For example, crews performing drywall construction could use rolling scaffolding or a scissor-lift to construct the upper portion of walls. However, estimators often have a preference for when a specific resource should be used in an activity, and that preference is often based on the design. Estimators also have different preferences for when a crew's productivity rate is appropriate in a given activity and how it should be adjusted for different design conditions. In the motivating case, the drywall estimator selected the crew's base productivity rate for the "install metal stud" activity based on the wall height and then adjusted the productivity to account for wall curvature and component similarity. Resource Specifications provide a way to represent estimating knowledge about when a resource is appropriate in an activity, and when to select and adjust a resource's productivity rate.

Resource Specifications represent estimators' rationale for when resources are appropriate in an activity and when to select and adjust a resource's productivity rate based on features. The next sections describe the common attributes of Resource Specifications and the framework developed to capture this estimating knowledge from estimators in ACE.

3.3.1 Common Attributes of Resource Specifications

Figure 8 shows the common attributes of Resources Specifications represented in the ontology and examples from the motivating case. The common attributes of Resource Specifications are described below. Some of the attributes of Resource Specifications are also represented in Activity Specifications and are described in detail in Section 3.2.1.

Attributes of Resource Specifications	Examples of Resource Specifications from the Motivating Case	
Object	Metal Stud	Metal Stud
Action	Install	Install
Resource	Rolling Scaffolding	Crew C-1
Resource Productivity	--	Crew C-1 Productivity Rate = 8 lf/hr
Productivity Adjustment	N/A	Increase
Adjustment Amount	N/A	20%
Design Condition – ***Constrained Feature***		
Design Condition – ***Constraint***	***Property:*** Height ***Operator:*** > ***Value:*** 10'	***Constrained Feature Property:*** Similarity ***Component Feature Grouped:*** Wall ***Direction:*** Horizontal ***Component Variation:*** 90 - 100% ***Similar Component Property:*** Height ***Property Variation:*** +/- 2"

<u>Figure 8a</u>: Common Attributes of Resource Specifications.

<u>Figure 8b</u>: Resource Specification that represents the estimator's preference to use Rolling Scaffolding when the wall heights are greater than 10'.

<u>Figure 8c</u>: Resource Specification representing the estimator's preference that Crew C-1's productivity be increased by 20% for walls with 90-100% similar wall heights ± 2".

<u>Figure 8</u>: Common attributes of Resource Specifications and examples from the motivating case. Resource Specifications represent estimators' rationale for when a resource is appropriate in an activity and when and how a resource's base productivity should be adjusted for specific features. ACE uses this estimating knowledge to help estimators assign resources and select and adjust resource productivity for activities needed to construct a given product model.

(1) ***Object*:** Refer to Section 3.2.1.

(2) ***Action*:** Refer to Section 3.2.1.

(3) ***Resource*:** The labor (e.g., "Crew C-1"), equipment (e.g., "Rolling-Scaffolding") or product resource (e.g., "Formwork") used in the activity. The resource can also be a crew consisting of labor and equipment. (Darwiche et al. 1988; Aalami 1998)

(4) ***Resource Productivity:*** The productivity rate for the resource selected in (3), such as Crew C-1's productivity of 8 lf/hr. If this attribute is not instantiated, then the Resource Specification applies to the resource specified in (3).

(5) ***Productivity Adjustment:*** This attribute represents how the resource productivity (4) will be adjusted. The resource productivity can either be "increased" or "decreased." In the motivating case, the estimator increases the crew productivity in the "Install Metal Stud" activity to account for component similarity and decreases the crew productivity to account for wall curvature.

(6) ***Adjustment Amount:*** The percentage amount that the resource productivity (4) will be increased or decreased. In the motivating case, the estimator increases the productivity by 20% to account for the productivity gains associated with component similarity (Figure 8c).

(7) ***Design Condition:*** The design condition represents estimators' rationale for when a resource should be used in an activity (Figure 8b), and when to select or adjust the resource's productivity (Figure 8c). Section 3.2.1 describes how the ontology formally represents design conditions that constrain component and intersection features. The ontology represents the common attributes estimators use to describe component similarity, including the properties of the component that need to be similar and the degree of similarity that needs to be achieved for component similarity to exist. The following attributes represent estimators' rationale for defining component similarity:

   a) ***Constrained Feature:*** The ontology represents the concept of component similarity by representing "groupings" of component features based on the feature property *similarity*. In the motivating case, the estimator evaluates the grouping of wall components to evaluate the "wall" component feature for component similarity. The current ontology does not represent the similarity of intersection features.

   b) ***Constrained Feature Property:*** The property of the constrained feature (7a) that is limiting the applicability of the resource (3) or resource productivity (4). The current ontology only represents the *similarity* of "groupings" of component features.

   c) ***Component Feature Grouped*:** The component feature that the estimator wants to evaluate for similarity. In the motivating case, the estimator evaluates the "wall" component feature for component similarity. The component feature grouped can be different from the component feature being estimated. For example, the similarity of "columns" can affect the formwork operations for constructing the "slab" component feature.

d) ***Direction:*** The ontology represents the direction for which component similarity will be assessed as either "horizontal" or "vertical." The horizontal direction represents similarity across a single floor. The vertical direction represents similarity across multiple floors. In the motivating case, the estimator evaluates the similarity of the wall components in the horizontal direction. To represent component similarity in both directions, estimators need to create two Resource Specifications where each Resource Specification represents component similarity in a different direction.

e) ***Component Variation*****:** The overall variation in the value of the component property allowed among similar components. This attribute is needed because estimators have different preferences for the degree of similarity allowed for component similarity to exist. In the motivating case, the estimators use the phrases "the majority of walls" and "most walls" to describe their preference for the degree of similarity that needs to exist. Figure 8c shows how the ontology represents this concept quantitatively, explicitly, and consistently by representing the estimator's preference that component similarity is achieved if 90-100% of the wall heights are similar. Hanna and Sanvido (1990) represent this concept in guidelines that specify "the system can handle moderate variation of wall size." We represent this concept explicitly and consistently using the component variation attribute.

f) ***Similar Component Property:*** The component property (or properties) of the component feature grouped (7c) that the estimator wants to compare to determine whether similarity exists. In the motivating case, Estimator #1 analyzes the property "height" and Estimator #2 analyzes the properties "height" and "type" to assess the similarity of wall components.

g) ***Property Variation:*** The variation in the value for the similar component property (7e) allowed for similarity. In Figure 8c, the estimator specifies her preference that component similarity applies if 90-100% of the wall heights are within ± 2."

For example, wall #1 is classified as similar to wall #2 if its height within 2" of wall #2.

The common attributes of Resource Specifications represent estimators' rationale about how features influence the applicability of resources in an activity and impact resources' productivity rates when executing the activities.

3.3.2 Resource Specification Templates

ACE provides a framework to enable estimators to represent and leverage their knowledge about the influence of features on activities and resources to calculate construction costs. In ACE, estimators represent their rationale for how features affect the execution of resources in an activity by filling out the Resource Specification template, as shown in Figure 9. Estimators specify their preferences for when resources are appropriate in an activity (Figure 9a) and when to select and adjust a resource's productivity rate for a particular feature (Figure 9b). ACE allows estimators to represent this estimating knowledge consistently and in a project-independent way.

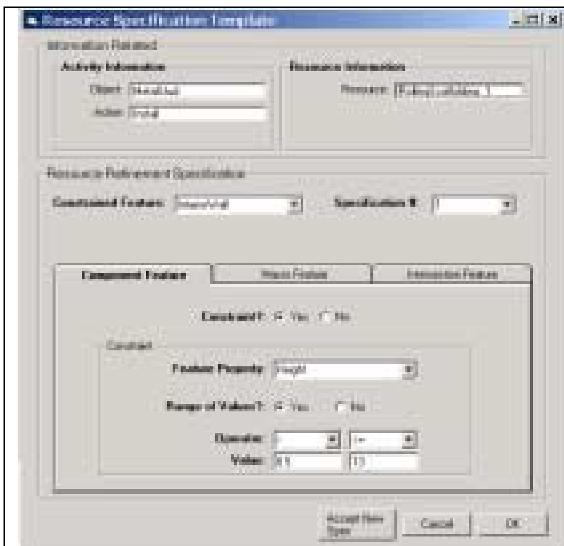


Figure 9a: Resource Specification representing an estimator's preference to use "Rolling Scaffolding" in the "Install Metal Stud" activity if the wall height is greater than 8.5' and less than 12'.

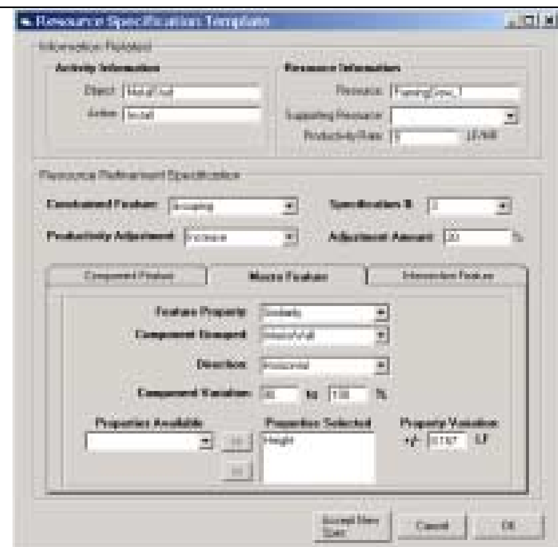


Figure 9b: Resource Specification representing an estimator's preference to increase the framing crew's productivity by 20% if 90-100% of the walls have similar wall heights.

Figure 9: Resource Specification template that allows estimators to represent their rationale for when a resource is appropriate in an activity and when and how a resource's productivity should be selected and adjusted based on features.

Estimators input their rationale once using the Resource Specification templates and ACE reuses this knowledge to assign resources and adjust resource productivity when generating and maintaining cost estimates for a given product model. Consequently, ACE enables estimators to account for the cost impact of features consistently throughout a project and across projects. Estimators representing their rationale in ACE cannot use the ad hoc methods described in the motivating case. For example, estimators cannot use the ad hoc method of adjusting a crew's productivity rate for all the walls when only one wall is curved (Figure 3). ACE uses the estimator's knowledge to determine when to assign resources to an activity and when and how to select and adjust resources' productivity rates for particular features in a given product model. For activities generated by ACE, each resource knows why it was assigned to an activity. Each resource's productivity rate knows why it was selected and how it was adjusted for particular features. With the quantities from the related object and the resource assignments and productivity rate adjustments, ACE calculates the resource's duration in an activity to determine the activity's construction costs.

4. Validation

Our research formalized an ontology for relating features and activities to represent cost estimators' rationale and developed a framework that we implemented in ACE to represent estimators' rationale in the computer. As discussed, design goals for the ontology were to be:

(1) *formal* to provide a structured way to represent the attributes necessary for estimators to describe their rationale and prevent estimators from using ad hoc methods to account for the cost impacts of features,

(2) *general* to represent estimating knowledge independent of a particular project and to support cost estimating of different domains, and

(3) *flexible* to represent different estimators' preferences for how and when features affect the project's activities and resources.

Our tests show that use of ACE allows estimators to generate and maintain cost estimates more completely, consistently and quickly than using state-of-the-art software tools.

The next sections describe the validation method, results, and research conclusions.

4.1 Validation Study Design

To demonstrate that the ontology is *formal*, we evaluated the level of completeness of estimates generated by 13 estimators using ACE and compared them to estimates generated by the same estimators using Timberline's state-of-the-art Precision Estimating (PE) software (Timberline 2001) that links with product models represented using the industry standard Industry Foundation Classes (IFC) (IAI 2001). We used level of completeness as a way to measure the extent to which estimators accounted for the cost impacts of features explicitly. If estimators used ad hoc methods or overlooked the cost impact of features, they received a lower score for completeness than if they explicitly accounted for the impacts of specific features. We also evaluated the ability of ACE and PE to capture and reuse the estimators' rationale to maintain the cost estimates when the design changes by recognizing repeated changes to relevant features and feature properties and their associated cost impacts based on the estimators' rationale. We performed this analysis to evaluate the extent to which the estimators' rationale for generating the estimate could be represented in ACE and PE to help the estimators maintain their estimates for specific design changes. We compared the level of completeness of the revised estimates generated by ACE and PE based on the estimators' rationale for generating the estimate. We also evaluated the consistency of the estimates generated by estimators using ACE and PE.

To demonstrate that the ontology is *general*, we demonstrated that ACE could generate project-specific estimates for two different components types.

To demonstrate that the ontology is *flexible*, we demonstrated that ACE could represent 13 different estimators' preferences for what features are important to them, when they are important, and how they affect the project's activities and resources.

4.2 Validation Tests

We performed a charrette test (Clayton et al. 1998) and three retrospective tests to validate the formal, general, and flexible representation of estimators' rationale in the ontology. Because ACE implements the ontology, we used ACE to perform each of the four validation tests:

(1) Charrette test with eight industry practitioners estimating interior wall construction costs for three design changes: (1) increase wall height from 8’ to 10.5’, (2) add new wall, and (3) move northern wall 5’.

(2) Retrospective test case of estimating interior wall construction costs on the Sequus Pharmaceuticals project (Staub-French and Fischer 2001).

(3) Retrospective test case of estimating interior wall construction costs on a DPR Office project.

(4) Retrospective test case of estimating concrete column construction costs on DPR’s Bay Street Emeryville Project.

4.3 Validation Method

To demonstrate the power and generality of the contributions, we wanted to show that estimators using ACE could generate and maintain cost estimates more completely, consistently, and quickly for different component types and estimator preferences. We evaluated the accuracy of the estimates for each of the four test cases by measuring the level of completeness. We evaluated the level of completeness of estimates generated by 8 estimators in the charrette test and 5 estimators in the retrospective test cases. We evaluated the consistency of estimates generated by practitioners for the charrette test. We also evaluated practitioners estimating time in the charrette test based on the time it took charrette participants to identify the cost impacts for each of the three design changes.

The level of completeness of estimates was the most important metric considered because it was applied in each of the four test cases. To evaluate the level of completeness of estimates in the four validation tests, we performed three steps:

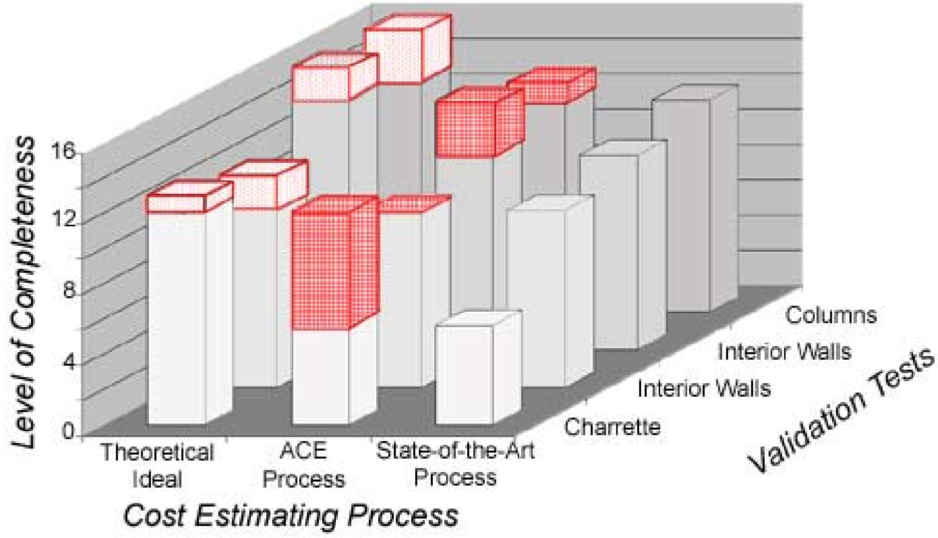
(1) Create Theoretical Ideal: We defined a theoretical ideal to represent the “most complete” estimate for each test case. We crafted the theoretical ideal based on interviews with estimating experts of interior wall and concrete column construction. The theoretical ideal represents cost impacts explicitly and excludes ad hoc methods used by estimators. For example, the theoretical ideal from the motivating case includes activities for wall turns and non-90° wall turns rather than the ad hoc method of adjusting crew productivity to account for these design conditions (Figure 3).

(2) Evaluate Completeness of Estimates: We evaluated the *level of completeness* of estimates generated by practitioners using ACE and PE based on how well they conformed to the theoretical ideal. We also evaluated the level of completeness of the revised estimates generated by ACE and PE for specific design changes that were relevant to the estimators based on their rationale for generating the estimate.

(3) Compare Level of Completeness of Estimates: We compared the level of completeness of estimates generated by practitioners using ACE and PE to the theoretical ideal. We also compared the level of completeness of the revised estimates generated by ACE and PE with the theoretical ideal for specific design changes based on the estimators' rationale for generating the estimate. For the charrette test, we compared the statistical significance and consistency of the estimates' level of completeness scores and the time it took for the eight charrette participants to identify the cost impacts for the three design changes.

4.4 Results from Validation Tests

Figure 10 shows the results of the four validation tests for practitioners using ACE and PE relative to the theoretical ideal. The height of each bar indicates the level of completeness of the estimates for each of the four test cases. The height difference between ACE and PE indicates that practitioners using ACE were able to generate substantially more complete estimates. The height difference between ACE and the theoretical ideal indicates that ACE approaches the theoretical ideal, but it also shows the limitations of the ontology to generate estimates as completely as the theoretical ideal.



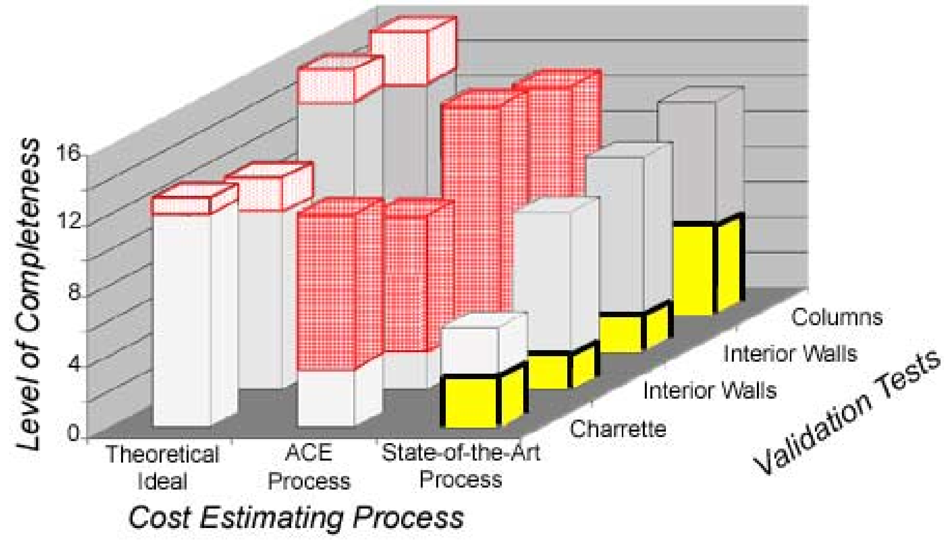
Legend:

Improvement in *level of completeness* of estimates generated by practitioners using ACE when compared with a state-of-the-art process (PE).

Disparity in *level of completeness* of estimates generated by practitioners using ACE when compared with the theoretical ideal.

Figure 10: Summary of the results for the level of completeness of estimates for the four tests. I tested the level of completeness of estimates generated by estimators using ACE compared with a state-of-the-art process (PE). I evaluated the level of completeness based on how well the estimates conformed to the theoretical ideal. The height of each bar indicates the level of completeness of estimates generated by practitioners using ACE and PE relative to the theoretical ideal for each validation test. The results show that the level of completeness of estimates generated by estimators using ACE is significantly greater than using PE and approaches the theoretical ideal.

Figure 11 shows the level of completeness of revised estimates generated using ACE and PE for specific design changes that were relevant to the estimators when generating the estimate. The height of each bar indicates the level of completeness of the estimates for each of the four test cases. The height difference between ACE and PE indicates that practitioners using ACE could identify the cost impact of specific design changes significantly more completely. The height difference between ACE and the theoretical ideal indicates that ACE approaches the theoretical ideal, but it also shows the limitations of the ontology to maintain estimates as completely as the theoretical ideal.



Legend:

Level of completeness of revised estimates for specific design changes.

Improvement in *level of completeness* of revised estimates generated by ACE when compared with state-of-the-art software.

Disparity in *level of completeness* of revised estimates generated by ACE when compared with the theoretical ideal.

Figure 11: Summary of results for the level of completeness of revised estimates for the four tests. I tested the level of completeness of the revised estimates generated by ACE and PE for specific design changes that were relevant to the estimators when generating the estimate. I evaluated the level of completeness based on how well the revised estimates conformed to the theoretical ideal. The height of each bar indicates the level of completeness of revised estimates created by ACE and a state-of-the-art process relative to the theoretical ideal for specific design changes for each validation test. The results show that the level of completeness of revised estimates generated by ACE is significantly greater than PE and approaches the theoretical ideal.

We evaluated the statistical significance of the completeness results for the estimates generated by practitioners during the charrette test. The results provide evidence that practitioners generate more complete estimates using ACE with 99% confidence. We compared the time it took for practitioners to estimate construction costs and demonstrated that practitioners using ACE could calculate construction costs 17 % faster than practitioners using PE. For a larger project, we speculate that estimating time could be reduced as much as 50% by implementing the theories we described in this paper in cost estimating software. We evaluated the consistency of the completeness scores for the estimates generated by the charrette participants using ACE and PE. The results suggest that practitioners using PE were more consistent than practitioners using ACE. However, the practitioners using PE had a significantly lower mean score for completeness because they were unable to identify many of the cost impacts for the specified design changes. Consequently, they produced less complete estimates more consistently. When considering consistency relative to completeness, we showed that practitioners using ACE were able to more consistently generate more complete estimates than practitioners using the state-of-the-art process.

4.5 Conclusions from Validation Tests

The goal of our research was to provide a formal, general, and flexible way to represent cost estimators' rationale to enable estimators to generate and maintain construction costs more completely and consistently. We demonstrated that the ontology of features and activities formalized in this research and implemented in ACE meets these criteria.

The results of the validation tests demonstrate that the formal representation of cost estimating knowledge in the ontology enabled practitioners using ACE to represent cost estimating knowledge more completely than using PE (Figure 10). These results suggest that practitioners and estimators using ACE could represent costs more explicitly (less ad hoc and fewer omissions) than estimators using PE. These results also demonstrate that estimators using ACE could represent costs more explicitly for both the generation (Figure 10a) and maintenance (Figure 10b) of cost information. Therefore, the four validation tests demonstrate the power of the ontology of features and activities to generate and maintain more complete estimates.

The four validation tests also demonstrate that the ontology is sufficiently general to represent cost estimators' rationale for a variety of domains. We modeled costs for two different component types in three retrospective test cases. Specifically, we modeled the costs for drywall construction on the DPR Office Project and Sequus Project and concrete column construction on the Bay Street Project. The construction of these two different component types required different activities, methods, and equipment. Moreover, different features and feature properties impacted costs for these two component types. The ability of practitioners to represent different component types and associated activities and features in ACE demonstrates the generality of the ontology.

The retrospective tests and charrette test demonstrate the flexibility of the ontology in representing different estimator preferences. Two of the retrospective tests evaluated estimators from different companies estimating the same component type. The different estimators for drywall construction were able to represent their preferences in ACE on the Sequus Project and the DPR Office Project. In addition, the eight practitioners in the charrette test were able to represent their preferences in ACE. These tests demonstrate generality across user types and suggest that ACE is flexible enough to represent different estimator preferences.

The charrette test demonstrates that practitioners using ACE were able to more consistently identify the correct cost impact when compared with a state-of-the-art process. Moreover, practitioners identified the cost impacts 17% faster using ACE when compared with a state-of-the-art process. The ability of practitioners to identify the cost impacts more consistently and quickly using ACE demonstrates the power of the ontology.

5. Conclusions

Our research formalized an ontology for relating features and activities, implemented the ontology in a software prototype to support the cost estimation of building components, and validated its power and generality. Without automated support to help estimators generate and maintain construction costs, estimators rely on ad hoc methods that are prone to error and lead to inconsistencies and inefficiencies in the cost estimating process. The validation tests demonstrate that by formally, generally, and flexibly representing cost estimating knowledge in the computer, practitioners are able to generate and maintain construction cost estimates more completely, consistently, and quickly than using state-of-the-art cost estimating software.

We limited the scope of our research in many ways. This research represents the cost impacts of features explicitly by representing how they affect production in terms of the activities performed and the resources required for their execution. This representation does not capture the field planning knowledge about how to build the components. For example, I provide a way to represent that an ‘opening’ in a wall requires field crews to install additional framing around the opening. I do not provide a way to represent field planning knowledge about how to frame the opening. This research focused on components as features, features that result from intersections of components, and features that result from the similarity of components. We excluded similarity of intersection features and other parameters for defining similarity. We also excluded other types of features. For example, we excluded material features, such as workability, and features that result from dissimilar component types that are not connected, such as the proximity of a duct run to a pipe run. This research also excluded factors exogenous to product design that affect construction costs, such as site characteristics, environmental conditions, and resource skill and availability.

Our research combines and extends previous research in cost estimating, activity modeling, and product modeling. Our research demonstrates that features and activities provide a theoretical framework to represent cost estimating knowledge about the specific impacts of different design conditions on construction costs. The ontology formally and generally represents cost estimating knowledge as activities consisting of a <FOAR> tuple where each activity knows when it is required, why resources were assigned to it, and how to adjust the resources' productivity rates. The representation of activities consisting of a <FOAR> tuple and the estimating knowledge about how and when to assemble activities consisting of <FOAR> tuples for a given product model provides a theoretical foundation for representing a project's scope (features), schedule (activities and durations), and corresponding cost.

This research takes an essential step in formalizing the theoretical framework needed to develop software tools that can help project teams to maintain integrated models of a project's scope, schedule, and cost. Our research has advanced the current state of knowledge about the relationships between product, activities, and costs. Understanding these relationships is critical to managing the design and construction process. Software tools that systematically implement these relationships can help project teams perform what-if analyses to develop cost effective and constructable designs. Hence, by formalizing the knowledge about the different types of relationships between scope, cost, and time in the computer, project teams should be able to avoid many of the inefficiencies that often result in cost overruns and schedule delays and better manage and control the design and construction process.

6. Acknowledgements

We gratefully acknowledge the support of the Center for Integrated Facility Engineering at Stanford University. We also thank the National Science Foundation for the partial support of this work under grant 9625228. We also acknowledge the financial support of the Future Professors of Manufacturing Program, Hathaway-Dinwiddie Construction Company, Flad & Associates, Mazzetti & Associates, Rosendin Electric, Paragon Mechanical, and Rountree Plumbing and Heating. We also thank the following people and their respective companies for their participation in this research:

- Melody Spradlin, Gregg Thoman, and Jeff Lindell from Hathaway Dinwiddie Construction Company,
- Chris Crouse from Rountree Plumbing and Heating,
- Chris Sorauf from Rosendin Electric,
- Jim Brady from Paragon Mechanical,
- Craig Vargas from California Drywall,
- Christopher Stewart from Flad and Associates,
- Dean Read, Jim Ray, Mark Qualcomm, and Jay Wilson from DPR Construction,
- John Sempek from Newcon Construction, Inc.,
- Bill Russell from Vance Brown, Inc., and
- Jim Dick from Pankow.

7. References

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

Akbas, R., and Fischer, M. (1999). "Examples of Product Model Transformations in Construction." *8dbmc, Durability of Building Materials & Components, Information Technology in Construction*, CIB W78 Workshop, Vancouver, BC, Canada, Volume 4, 2737-2746.

Aouad, G., Betts, M., Brandon, P., Brown, F., Child, T., Cooper, G., Ford, S., Kirkham, J., Oxma, R., Sarshar, M., and Young, B. (1994). "*ICON: Integration of*

Construction Information.” Department of Surveying and Information Technology Institute, University of Salford, Salford.

Aouad, G., Child, T., Marir, F., and Brandon, P. (1997). “*Open Systems for Construction (OSCON), draft industry report*.” Department of Surveying, University of Salford, Salford.

Clayton, M.J., Kunz, J.C., and Fischer, M.A. (1998). "The Charrette Test Method." *Technical Report 120*, Center for Integrated Facility Engineering, Stanford, CA.

Cunningham, J.J. and Dixon, J.R. (1988). “Designing with Features: The Origin of Features,” *ASME Computers in Engineering Conference*, San Francisco, CA, Aug. 1988, 237-243.

Darwiche, A., Levitt, R., and Hayes-Roth, B. (1989). “OARPLAN: Generating Project Plans by Reasoning about Objects, Actions and Resources.” *AI EDAM*, 2(3), 169-181.

de Sousa, U.E.L. and Thomas, H.R. (1996). “Development of an Explanatory Model for Concrete Formwork Productivity.” *7th International Symposium on the Organization and Management of Construction*, CIB, W-65, Glasgow, Scotland, Aug. 28, 1996, (2)27-38.

Fischer, M. (1991). “Constructibility Input to Preliminary Design of Reinforced Concrete Structures.” *Technical Report 64*, Center for Integrated Facility Engineering, Stanford.

Fischer, M. and Tatum, C.B. (1997). “Characteristics of Design-Relevant Constructability Knowledge.” *Journal of Construction Engineering and Management*, ASCE, 123(3), 253-260.

Froese, T. and Rankin, J. (1998). "Representation of Construction Methods in Total Project Systems", *Computing in Civil Engineering: Proceedings of the International Computing Congress*, ASCE, Boston, MA, October 19-21, 1998, 383-394.

Froese, T., Yu, K., and Shahid, S. (1996). "Project Modeling in Construction Applications," *Computing in Civil Engineering: Proceedings of the Third Congress*, ASCE, Anaheim, June 1996, 572-578.

Hanna, A. S., and Sanvido, V. E. (1990). “Interactive Vertical Formwork Selection.” *Concrete International: Design and Construction*, 12(4), 26-32.

International Alliance of Interoperability (IAI) (2001). “IFC 2x Extension Modeling Guide”, Available from http://www.iai.org.uk

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

Laitinen, J. (1998). “Model Based Construction Process Management,” *PhD Thesis*, Royal Institute of Technology. Stockholm, Sweden.

Sanders, S., and Thomas, R. (1991). “Factors Affecting Masonry-Labor Productivity.” *Journal of Construction Engineering and Management*, 117(4), 626-644.

Shah, J.J. (1991). “Assessment of Features Technology,” *Computer-Aided Design*, 23(5), 331-343.

Slaughter, S. (2000). “The Link Between Design and Process: Simulation Modeling of Construction Activities.” *Proceedings of the Construction Congress VI*, Orlando, Florida, 1051-1057.

Smith, G. R., and Hanna, A. S. (1993). “Factors Influencing Formwork Productivity.” *Canadian Journal of Civil Engineering*, 20(1), 144-153.

Staub-French, S., and Fischer, M. (2001). “Industrial Case Study of Electronic Design, Cost, and Schedule Integration." *Technical Report 122*, Center for Integrated Facility Engineering, Stanford.

Staub-French, Sheryl (2002) “Feature-Driven Activity-based Cost Estimating.” *PhD Thesis*, Stanford University, Stanford.

Staub-French, S., Fischer, M., Kunz, J., Paulson, B. and Ishii, K. (2002a). “A Feature Ontology to Support Construction Cost Estimating.” *Working Paper #69*, Center for Integrated Facility Engineering, Stanford.

Staub-French, S., Fischer, M., Kunz, J., Paulson, B. and Ishii, K. (2002b). “A Formal Process to Create Resource-loaded and Cost-loaded Activities Related to Feature-based Product Models.” *Working Paper #71*, Center for Integrated Facility Engineering, Stanford.

Stumpf, A., Ganeshan, R., Chin, S. and Liu, L. (1996). "Object-Oriented Model for Integrating Construction Product and Process Information." *Journal of Computing in Civil Engineering*, 10(3), 204-212.

Thomas, H.R. and Sackrakan, A. (1994). "Forecasting Labor Productivity Using the Factor Model." *Journal of Construction Engineering and Management*, American Society of Civil Engineers, 120(1), 229-239.

Thomas, H.R. and Zavrski, I. (1999). "Construction Baseline Productivity: Theory and Practice." *Journal of Construction Engineering and Management*, American Society of Civil Engineers, 125(5), 295-303.

Timberline Software Company (2001). Precision Estimating Extended and CAD Integrator, Users Documentation, Beaverton, Oregon.

Udaipurwala, A. and Russell, A. D. (2000). "Reasoning about Construction Methods", *Proceedings of Construction Congress VI*, Orlando Florida, February 2000, 386-395.