CIFE CENTER FOR INTEGRATED FACILITY ENGINEERING

Practical and Research Issues in using Industry Foundation Classes for Construction Cost Estimating

By

Sheryl Staub-French
Martin Fischer

CIFE Working Paper #56
January, 2000

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

Executive Summary

Industry Foundation Classes (IFC) have made substantial progress in recent years, and many design software companies now provide export capabilities of IFC-based product models. Consequently, product models are starting to become available from commercial design software programs used in the building construction industry. Yet, there has been limited testing of IFC-based product model representations to support the many processes and tasks required from construction professionals. In particular, IFC-based product models contain quantity information that could be used for cost estimating. In this research study, we assess the usefulness of IFC's to meet the practical needs of construction cost estimators.

On average, each item in a cost estimate is created seven times throughout the life of a project (Laitinen 1998). Consequently, it is critical that IFC's do not only support the generation of cost estimates, but also assist estimators with the maintenance of cost estimates throughout the project life cycle. This report shows that an IFC-based product model contains most of the information needed to create a cost estimate. The IFC's provide the information about the types of components and their dimensions and quantities needed for cost estimating. In essence, they help estimators take off quantities automatically and establish an electronic link between a component in a 3D-product model and a cost item in a cost estimating database to create a cost estimate. However, existing estimating software lacks the functionality necessary to assist users in maintaining cost estimates throughout the project, especially in the case of design changes. The reason is that in addition to material quantities, product features, such as repetition and dimensional modularity, also affect construction costs. This report shows that product features determine largely whether a particular cost item from a cost estimating database is the appropriate cost item with the correct unit cost and product rate assumptions. Hence, estimating software must be able to identify product features from an IFC-based product model to assist estimators with creating and maintaining cost estimates. Without computer support to store and use these meta-data about the relationships between cost information and product model components the estimators will need to remember when to adjust the cost information in the case of design changes so that the cost estimate and project scope descriptions are in balance. However, on many projects designs change often and there are thousands of links or relationships between cost items and product model components. Therefore, it is unlikely that estimators will be able to manually update all the electronic links between cost information and 3D components in a timely and complete manner to maintain an accurate computer model with linked cost and scope information throughout a project's life cycle. Hence, we advocate and outline new estimating functionality necessary to leverage the information available in IFC-based product models.

Specifically, there are two objectives for this research study:

1) to assess the *IFC's product model* to support the identification of *product features* that affect construction cost information,
2) to assess the *IFC's project model* to *maintain integration* between product, cost, and resource information as the project evolves.

In summary, our initial findings with respect to the suitability of the IFC's version 2.0 to support cost estimating processes are as follows:

- The only product feature that is explicitly captured as an attribute of the product is openings.
- Most of the product features and properties necessary for estimating can be inferred from an IFC-based product model. The only extension to the IFC's product model that we would encourage would be to explicitly represent the support relationship.
- Estimating software will need to provide new functionality to leverage the information available from an IFC-based product model.
- The IFC's support representation of the products at multiple levels of detail.
- The relationships provided by the IFC's support different types of product decomposition.

- However, the IFC's do not explicitly provide the context for the decomposition.
- The IFC's do not provide a representation to signify what attribute on the product has changed.

We also discuss our vision for an estimating system that analyzes an IFC-based product model to identify relevant construction cost information. We describe the formalisms required to enable estimators to explicitly represent their rationale for relating product and cost information and the mechanisms necessary to automatically identify product features and incorporate their affect on construction costs. We then discuss the related research background. Finally, we conclude this report by describing future research needs.

Table Of Contents

1. Introduction

The International Alliance for Interoperability (IAI) was formed in 1994 with the objective of defining specifications for Industry Foundation Classes (IFC's) to enable information sharing throughout the project life-cycle and across all disciplines and technical applications in the building industry (IAI 1998). The first version of the IFC's was released in 1995 and in subsequent years, many design software companies have added functionality to export IFC-based product models. Consequently, product models are now readily available from commercial design software programs used in the building construction industry. Yet, there has been limited testing of IFC-based product model representations to support the many processes and tasks required from construction professionals. In this study, we provide an initial assessment of the IFC's to support the process of construction cost estimating throughout the project life-cycle.

Construction costs are largely influenced by product features[1], such as repetition and dimensional modularity. It is the estimator's task to determine what product features are important, how the product features affect the construction cost information, and how the cost estimate should be adjusted to incorporate that effect. Cost estimates are created as many as seven times throughout the life of a project so it is critical to understand the effects of product features on construction costs in order to maintain the cost estimate as the project evolves (Laitinen 1998). In this research study, we investigate the representation of product features in an IFC-based product model and the representation of an IFC-based project model to support integration throughout the project life-cycle. Specifically, the objectives of this research study are:

1) to assess the *IFC's product model* to support the identification of *product features* that affect construction cost information,
2) to assess the *IFC's project model* to *maintain integration* between product, cost, and resource information as the project evolves.

The promise of product model based cost estimating is that quantity take-offs can be largely automated and that the estimate can be updated automatically when the design changes. To examine this promise we will start by describing an actual project to illustrate the practical needs of cost estimators and the capabilities of commercial estimating tools.

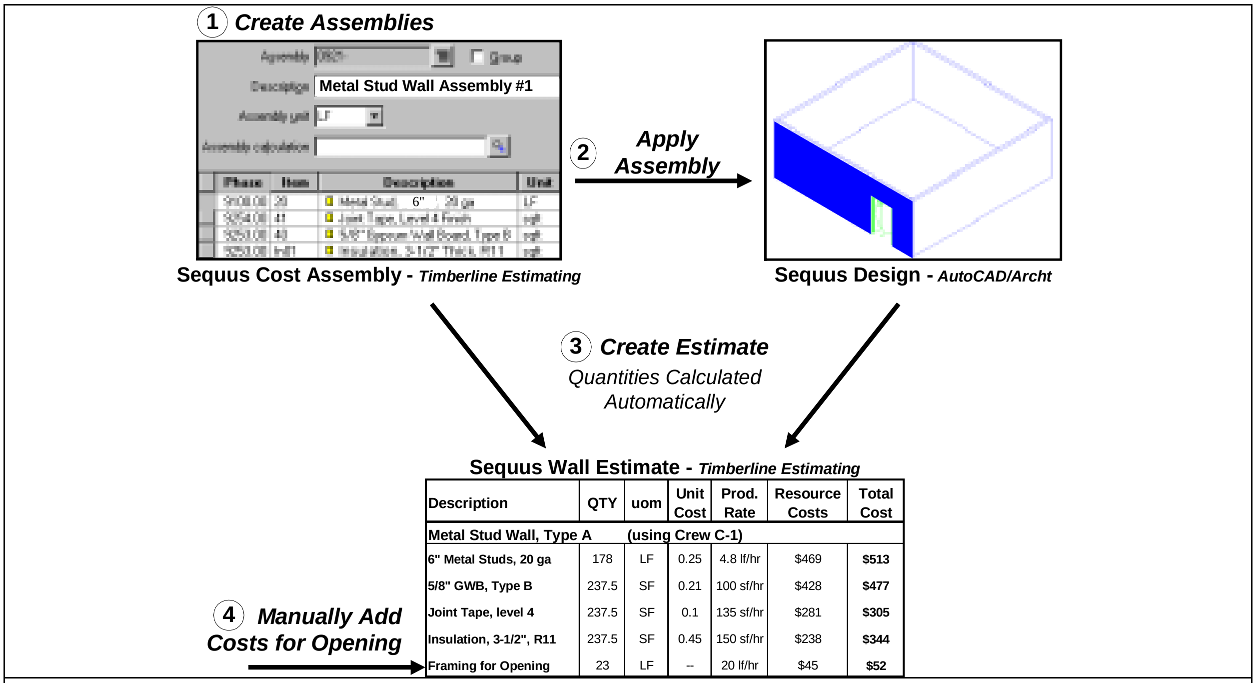
1.1 Motivating Case Example and Current Practice

The practical motivation for this research is based on our experience working with a project team throughout design and construction of a pilot plant facility for Sequus Pharmaceuticals. The project team used a

---

[1] Cunningham and Dixon (1988) define features as "any geometric form or entity that is used in reasoning in one or more design or manufacturing activities" (i.e., fit, function, manufacturability evaluation, analysis interfacing, tool and die design, inspectability, serviceability etc.).

3D CAD model (ArchT by KETIV on top of AutoCAD R14) linked to the cost estimate (Precision Estimating by Timberline) and the schedule (MS Project) (Staub et al. 1999). With these tools, once estimating work packages or cost assemblies are linked to CAD elements the quantity takeoff is automatic. For our example, we will present the drywall subcontractor's estimating process to determine the labor and material costs for all four walls of the room shown in Figure 1. Specifically, we will illustrate what product features are important to the estimator, how the estimator reasons about the product features to determine the relevant construction cost information, and how the existence of product features affects the cost estimate.



Sequus Wall Estimate - ***Timberline Estimating***

Description	QTY	uom	Unit Cost	Prod. Rate	Resource Costs	Total Cost
Metal Stud Wall, Type A	**(using Crew C-1)**					
6" Metal Studs, 20 ga	178	LF	0.25	4.8 lf/hr	$469	**$513**
5/8" GWB, Type B	237.5	SF	0.21	100 sf/hr	$428	**$477**
Joint Tape, level 4	237.5	SF	0.1	135 sf/hr	$281	**$305**
Insulation, 3-1/2", R11	237.5	SF	0.45	150 sf/hr	$238	**$344**
Framing for Opening	23	LF	--	20 lf/hr	$45	**$52**

Figure 1: Overview of the current process of generating estimates by linking design and construction cost information. The estimator performs the following steps:

1) Create a cost assembly that corresponds to the design and specifications of the wall. For every wall that requires different cost information, the estimator creates a different assembly. In this case, the estimator chooses to make one cost assembly for all four walls and manually adds the costs for the opening, as shown in step 4.
2) Attach the cost assemblies to each design object. It is the estimator's responsibility to ensure that the right assembly is attached to the appropriate design object.
3) Order the software to create the estimate and automatically calculate the quantities.
4) Manually add costs for the additional framing required at the door.

Figure 1 shows an overview of the estimating process using commercial estimating software that links with 3D CAD software. In step 1, the estimator creates a cost assembly to aggregate all the estimating items that are needed for each wall in the room (all the walls in this room have the same composition). Each item in the assembly contains information about the material unit costs, labor and equipment resources, and the production rates. The estimator has two options for creating assemblies for the design shown in Figure 1. Since the four walls are essentially identical, the estimator could create one assembly for all four walls and add the opening manually to the estimate. The benefit of this approach is that the estimator has to create and attach only one cost assembly. The shortcoming of this approach is that the estimator has to *manually* add the openings to the cost estimate. Alternatively, the estimator could create two assemblies, one assembly for the wall that has an opening, and one assembly for the three walls that don't have an opening. The benefit of this approach is that there is less chance for the estimator to overlook the additional costs for the openings because they are explicitly included in the cost assembly. The shortcoming of this approach is that the estimator has to create two cost assemblies. On this project, the estimator creates one assembly for all four walls and adds the costs for the opening manually. After the assembly is created, it is attached to all four walls, as shown in step 2. After all the assemblies have been attached, the estimator orders the estimating software to create the estimate, and the software automatically calculates the quantities for each item in the assembly. Finally, the estimator manually adds the costs for the additional production time for framing around the door opening. The estimate shows the selections the estimator made for material unit costs, resources, and productivity rates. However, the estimate does not explain *why* those selections were made. It is not clear why Crew C-1 was chosen or why the productivity rate to install the metal studs was 4.8 lf/hr. To answer these questions, one needs to understand the rationale for how the estimator relates the product to the cost information.

We have divided the estimating process into steps to illustrate the estimator's rationale for relating product and cost information. Figure 2 describes each step in detail.

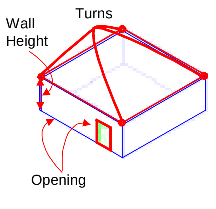
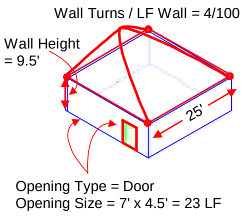
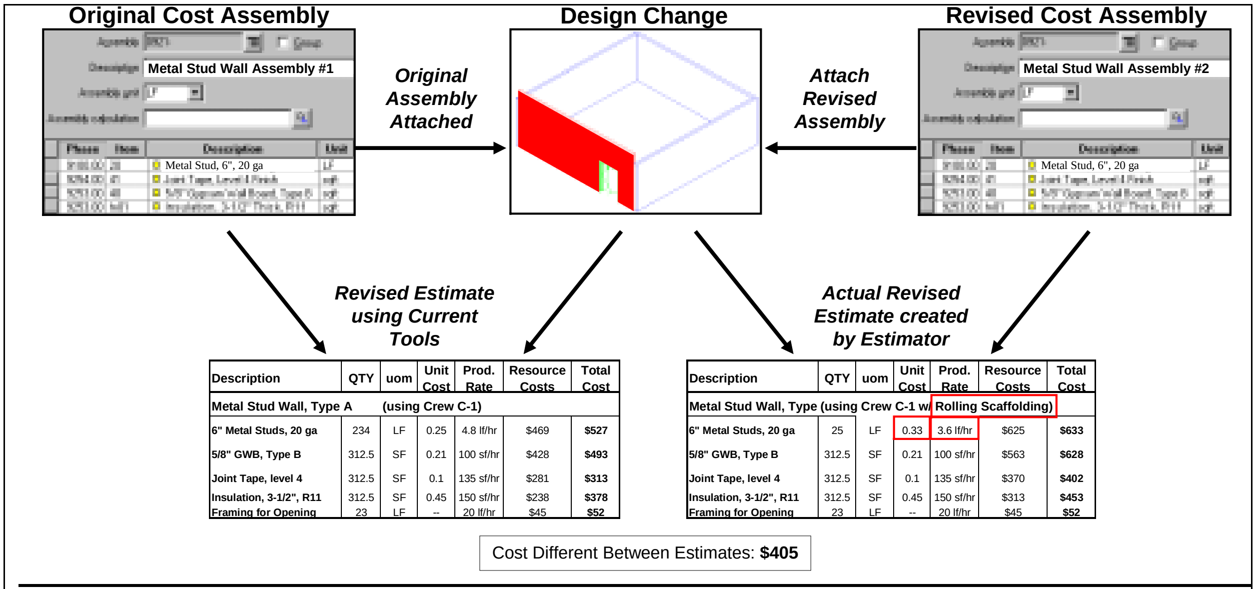
Step 1 ***Identify Product and Cost Information*** Metal Studs and Productivity Rate							
Step 2: ***Identify Product Features & Properties***		**Step 3:** ***Capture Constraints on Product Features***		**Step 4:** ***Incorporate Dependencies***		**Step 5:** ***Adjust Estimate to Incorporate Affect***	
Product Analysis	**Product Feature** **Property**	**Constraints on Product Feature**	**Product Analysis**	**Dependent Product Feature** **Property**	**Constraints on Dependent Product Features**	**Action**	**Result in Example Case**
	Wall Height	Wall Height < 10'				Select	6 LF / Hour
	Turn Quantity	Wall Turns > 4		Wall Length	Wall Length > 100	Modify	Reduce Productivity 6 LF/Hr * 80% = 4.8 LF/Hr
	Opening Type_of	Opening_Type = Door		Door Width	Door Width > 3'	Add	Add Production Time 23 LF / (20 LF Opng/Hr) = 1.15 Hours
Figure 2a		**Figure 2b**		**Figure 2c**		**Figure 2d**	

Figure 2: Steps in the estimating process that illustrate the estimator's rationale for determining the appropriate productivity rate for installing the metal studs:

1) Identify the product and cost information to estimate. In this case, the estimator is estimating the productivity rate for the metal stud installation.
2) Identify the important product features and properties that affect the productivity rate for installing metal studs. This step requires three different levels of analysis on the product model: 1) analysis of the product itself, 2) analysis of multiple products of the same type, and 3) analysis of two connecting products of different types.
3) Determine the constraints on the product features that affect the productivity rate and analyze the product model to determine whether they are satisfied.
4) Incorporate dependencies between product features. For example, the estimator identifies that an opening exists and that the opening type is a door. To create the estimate, however, the estimator also identifies the door size to determine the additional production time needed for the additional framing around the door.
5) Determine how the estimate should be adjusted to incorporate the satisfaction of the constraints on the product features. Figure 2d shows that the estimator first selects an initial productivity rate based on the height of the wall, then modifies the productivity rate to account for number of wall turns per length of wall, and finally adds production time to account for additional framing at the door opening.

Figure 1 shows the estimate that was created for the walls in one room of the Sequus Project, and Figure 2 illustrates the rationale that was used to create a portion of that estimate. But why is it important to understand the estimator's rationale? It is particularly important because estimates are not just created once, and estimating information is not just used on one project. For example, on the Sequus Project, the height of one wall changed from a height of 9.5' to 12.5'. How does this design change affect the estimating information already selected?

As Figure 1 illustrates, current tools allow an estimator to attach cost information to a design object and generate an estimate. Consequently, if the revised design were estimated using the current process, the estimate would reflect the change in design by calculating a new quantity. Figure 3a shows the revised estimate that was created using the original cost assembly. Unfortunately, the changed quantity is not the only impact of this design change. The wall height actually affects the selection of the material unit cost, the resources, and the productivity rate. Figure 3b shows the revised cost assembly and the changes that the estimator made to the metal stud estimating item in the cost assembly as a result of this design change. The estimator had to create a new assembly to estimate the revised wall because the previous assembly was no longer applicable. Specifically, this design change resulted in an increase in the material unit cost, a change in the resource composition by adding rolling scaffolding, and a reduction in the productivity due to the hindered access to the upper portion of the wall from the rolling scaffolding. Therefore, some design changes affect the applicability of the cost information previously selected, requiring the estimator to identify the information that is no longer applicable and create a new cost assembly to estimate the changed design.



Description	QTY	uom	Unit Cost	Prod. Rate	Resource Costs	Total Cost
Metal Stud Wall, Type A	**(using Crew C-1)**					
6" Metal Studs, 20 ga	234	LF	0.25	4.8 lf/hr	$469	**$527**
5/8" GWB, Type B	312.5	SF	0.21	100 sf/hr	$428	**$493**
Joint Tape, level 4	312.5	SF	0.1	135 sf/hr	$281	**$313**
Insulation, 3-1/2", R11	312.5	SF	0.45	150 sf/hr	$238	**$378**
Framing for Opening	23	LF	--	20 lf/hr	$45	**$52**

Description	QTY	uom	Unit Cost	Prod. Rate	Resource Costs	Total Cost
Metal Stud Wall, Type (using Crew C-1 w/ Rolling Scaffolding)						
6" Metal Studs, 20 ga	25	LF	0.33	3.6 lf/hr	$625	**$633**
5/8" GWB, Type B	312.5	SF	0.21	100 sf/hr	$563	**$628**
Joint Tape, level 4	312.5	SF	0.1	135 sf/hr	$370	**$402**
Insulation, 3-1/2", R11	312.5	SF	0.45	150 sf/hr	$313	**$453**
Framing for Opening	23	LF	--	20 lf/hr	$45	**$52**

Cost Different Between Estimates: **$405**

Figure 3a	**Figure 3b**

Figure 3: Comparison of cost assemblies and estimates generated as a result of the design change.
Figure 3a: The revised estimate that was created using current tools and the original link between Assembly #1 and the revised design. Current tools help the estimator to calculate the impact of the design change on the quantities but do not help the estimator to identify cost information that may no longer be applicable to the changed design.
Figure 3b: The actual estimate that was created to account for the impact of the design change. Generating this estimate required the estimator to create another assembly. The cost information affected by the design change includes the material unit cost, the crew composition, and the productivity rate, shown with boxes. The actual estimate was $405 more than the revised estimate calculated using the relationships established by current 3D CAD and estimating tools.

Not all design changes require estimators to revise the cost assembly to update the cost estimate. For example, suppose the room was increased in size by moving one wall 10' away from the other wall. This design change increases the length of two walls by 10'. This design change would not affect the applicability of the cost assembly previously created. As shown in Figure 2, there are no relationships between the wall length and the cost information for the metal studs. Consequently, the estimator can apply the same assembly to the revised design, and the cost impact would accurately be reflected in the estimate with the revised quantities automatically incorporated. Unfortunately, current tools do not help the estimator to identify which design changes require changes to the cost information in the cost assembly and which changes simply require an update of the quantities.

1.2 Limitations of Current Practice

This example illustrates the limitations of existing software tools to aid the estimator during the estimating process and to maintain the estimate throughout the project life-cycle. Specifically, the following limitations exist:

a) Lack of Formalization:

- ***Rationale remains in the minds of estimators and is not formalized:*** Estimating knowledge for relating product and cost information remains in the minds of estimators and is not formalized. For example, the estimator did not formalize the effect of wall turns on the productivity rate as shown in Figure 2. We created the formalization by interviewing the drywall estimator and interpreting his reasoning process.
- ***Current software tools do not capture estimators' rationale:*** Estimators are limited to capturing the context for the cost information in text strings when using current estimating software. Text strings do not provide a computer-interpretable representation. Figure 2 shows that the estimator's rationale for relating product and cost information includes the following information: 1) the product features, 2) the constraints on the product features, 3) dependencies between product features, and 4) the actions that determine how the estimate should be adjusted. Current tools do not capture this information explicitly.

b) Lack of Automated Analysis:

- ***Current software tools do not help the estimator to identify the relevant cost information:*** Consequently, estimating remains a largely manual process. Specifically, the case study illustrated that estimators have to manually identify product features, manually incorporate the effect of product features on the cost information, manually create cost assemblies, and manually determine the cost impacts of design changes.
- ***Incorporating the effects of product features on construction costs is ad hoc:*** The estimator modified the productivity rate for the installation of the metal studs on all four walls to account for the existence of wall turns, as shown in Figure 2. However, the purpose for the modification of the productivity rate was to account for the additional framing that was required at the wall turns. Therefore, modifying the productivity rate for the entire wall is an ad hoc method of incorporating the effect of wall turns that does not explicitly represent the true nature of the impact. A similar *workaround* exists in the cost estimate shown in Figure 1. Like wall turns, openings also require additional framing. In this instance, the estimator

did explicitly represent the opening in the estimate, however, he only accounted for the additional production time and ignored the impact on material costs.

To address these limitations, estimators need a framework to capture their rationale for relating product and cost information. Specifically, this framework should allow estimators to identify product features and their effect on construction cost information in a formalized and systematic way. Moreover, estimators need a computer tool that uses the computer-interpretable representation of their rationale to help the estimator identify relevant cost information given an IFC-based product model and identify the effect of product model changes on the cost information selected.

In the next section, we describe an initial assessment of the IFC's to meet the needs of estimators described above. Specifically, we describe the capabilities of the IFC's product model to represent the product features that affect construction cost information and the capabilities of the IFC's project model to maintain integration between product, cost, and resource information as the design evolves.

2. Initial Assessment of IFC's

The case study demonstrated that two major tasks for estimators are identifying product features and incorporating their effects on construction cost information. Section 2.1 describes the capabilities of IFC'-based *product models* to represent product features affecting construction costs. This section also briefly discusses the capabilities of IFC's to represent products at multiple levels of detail and keep track of changes to the product model. In section 2.2, we discuss the capabilities of the IFC's *project model* to maintain integration between product, cost, and resource information throughout the course of the project.

2.1 Assessment of IFC's Product Model

The section discusses three different issues associated with construction cost estimating using an IFC-based product model. In section 2.1.1, we discuss product features that affect construction cost information and investigate how the IFC's support the identification of product features. Section 2.1.2 discusses the capabilities of IFC's to support estimating at multiple levels of detail. Section 2.1.3 discusses IFC's support of design changes.

2.1.1 Identification of Product Features

Identifying key product features and their properties is an important step in the cost estimating process. In the case example, the key product features and properties that affected the productivity rate for installing the metal studs were the quantity of wall turns, the type and size of the openings, and the height of the wall. After reviewing previous research in this area and interviewing estimators, we have found that the same product features affect many types of work in the building construction industry. For example, turns are important for piping systems; openings are important for formwork systems; and height differences are important for installing ductwork systems. We have identified 18 product features and properties thus far. Table 1 shows a consolidation of the product features and properties that affect construction cost information. We identified these important product features and properties through literature reviews, interviews with the HVAC subcontractor and drywall subcontractor from the Sequus Project, and from R.S. Means and engineering handbooks (R.S. Means Company 1998; Fischer 1991; Hanna and Sanvido 1990; Hendrickson et al. 1987; Peurifoy and Oberlender 1989; Sanders and Thomas 1991; Smith and Hanna 1993; Tah et al. 1991).

		Related Research						Interviews		Means	
		Formwork Method Selection	Masonry Productivity	Masonry Productivity	Piping Cost Estimating	Formwork Method Selection	Formwork Productivity	Metal Stud Walls Cost Estimating	HVAC Cost Estimating	Metal Stud Walls Cost Estimating	Metal Stud Walls Cost Estimating
		Fischer 1991	Hendrickson et al 1987	Sanders & Thomas 1991	Tah et al 1991	Hanna & Sanvido 1990	Smith & Hanna 1993	Drywall Subcontractor	HVAC Subcontractor	R.S. Means	Engineering Handbook
Properties of Features	Dimension	x	x	x	x	x	x	x	x	x	x
	Material Type	x	x	x	x	x	x	x	x	x	x
	Connection Type	x		x	x	x	x	x	x		
	Support Type	x	x	x		x					
	Elevation	x	x				x		x		
	Orientation	x		x			x	x	x		x
	Shape	x				x	x	x	x		x
	Curvature / Sloping			x			x	x			
	Technical Specs (finish, rating)			x				x	x	x	x
Product Features	Distances between Similar Elements	x				x	x	x	x		
	Distances between Dissimilar Elements							x	x		
	Workability	x									
	Durability	x									
	Resistance to Special Conditions	x									
	Repetition	x		x		x	x	x			
	Turns	x		x			x	x	x		
	Changes in Dimensions	x				x	x	x	x	x	
	Openings / Penetrations	x		x		x	x	x	x		x

Table 1: Product features and properties of product features that affect construction cost information in different domains. The product features and properties of product features were identified through various research efforts, interviews with the HVAC subcontractor and drywall subcontractor from the Sequus Project, and in R.S. Means and construction cost estimating handbooks.

There are two options for identifying product features during the estimating process: 1) explicitly represent the product feature in the IFC-based product model, or 2) develop reasoning mechanisms to infer the existence of product features from an IFC-based product model. The first option requires less complex reasoning mechanisms but it requires more information to be identified during design development. Moreover, the IFC's have to provide a way to represent these features explicitly if they are to be included in the product model. Unfortunately, most of the product features identified in Table 1 are not explicitly supported by the IFC's. **The only product feature that is explicitly captured as an attribute of the product is openings.** Therefore, to identify product features in an IFC-based product model, there needs to be reasoning mechanisms that *infer* the existence of product features using the geometric and topological data from the product model. One question remains, however, do the IFC's provide sufficient representation of the product to infer the existence of product features?

Through the course of this research study, **we have found that most of the product features and properties shown in Table 1 can be inferred from an IFC-based product model. Thus far, the only extension to the IFC's product model that we would encourage would be to explicitly represent the support relationship.** This relationship should be explicit because it would require complex reasoning mechanisms to infer the existence of support. In contrast, the IFC's could simply add the attribute

"SupportedBy" to create the relationship explicitly (Froese et al 1999). Currently, the relationships between components supported by the IFC's are: PartOfGroups, Nests, Contains, ConnectedTo, IsAssemblyThrough, and HasOpenings (inverse relationships are not specified). Therefore, **the IFC's support many of the necessary connection relationships between components and geometric information to support the identification of product features affecting construction costs.**

Determining the existence of product features requires different levels of analysis on the product model with increasing complexity. Table 2 shows the three product features and properties from the case example and the corresponding analysis requirements.

Example Product Feature and Property (*Feature* - "Property")	**Description of Analysis**
Wall - "height"	Requires the retrieval of an attribute value on the product being analyzed
Turn - "Quantity"	Requires reasoning about the aggregation of all products of the same type
Opening - "Type_of"	Requires reasoning about the aggregation of two different types of connected products.

Table 2: Product features and properties identified in the case example and corresponding analysis requirements.

Figure 4 demonstrates how a reasoning mechanism could analyze the geometric and topological relationships of the product to infer the existence of product features from an IFC-based product model.

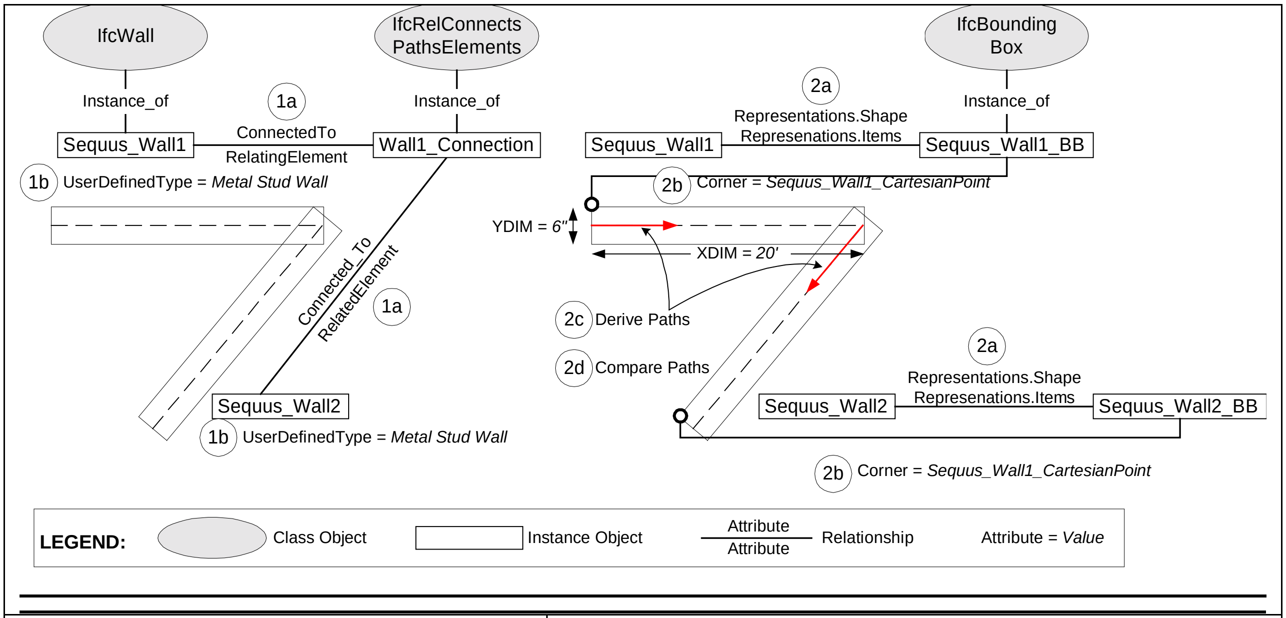


Figure 4a	Figure 4b

Figure 4: Proposed approach for identifying a turn in an IFC-based product model.
Figure 4a: Analyze connected elements and identify elements that are walls of the same type.
1a) Identify all connected elements to Sequus_Wall1 using the attribute "ConnectedTo".
1b) Determine whether the connected element is an instance of the IfcWall class and is of the same type.
Figure 4b: Determine whether the center-line of each wall follows the same path. If the paths are not equal, then the wall is turning.
2a) Find the geometric representation of the wall following the attributes "Representations", "ShapeRepresentations", and "Items".
2b) Using the attribute "corner", identify the Cartesian point for the lower-left corner of the wall.
2c) Derive the path of the wall from the "corner" attribute and the dimensions of the wall, "XDIM" and "YDIM".
2d) Compare the paths to determine whether the wall is turning. The paths can also be used to determine the orientation of the wall turn.

The discussion so far has shown that **estimating software will need to provide new functionality** like outlined in Figure 4, for example, **to leverage the information available from an IFC-based product model.** In the next section, we assess the IFC's product model representation to support estimating at multiple levels of detail.

2.1.2 Multiple Levels of Detail

Product models are represented at different levels of detail throughout design development. Consequently, estimators need to be able to identify the level of information available in the product model to create cost estimates throughout the project. In the case example, the design consisted of four walls, each having the same decomposition. Information about the wall's decomposition significantly affects the cost information selected. For example, the product specifications for the gage and size of the metal stud affected the material unit cost. Therefore, it is critical for the IFC's to support product representations at multiple levels of detail for the corresponding product specifications. **The IFC's support representation of the products at multiple levels of detail** using the following relationships: Nests, Contains, and IsAssemblyThrough. The properties of the product are defined by IFC's through the use of extended properties in the class IfcPropertySet and its relationship to IfcProperty through the HasProperties attribute.

The relationships provided by the IFC's support different types of product decomposition. **However, the IFC's do not explicitly provide the context for the decomposition.** Consequently, it is unclear which relationship the estimator (or the estimating application) should use to determine the product's decomposition. For example, should the estimator use the "IsAssemblyThrough" relationship or the "Nests" relationship? These ambiguities make it difficult for estimators to assess the level of detail in the product model (Froese et al 1999). Therefore, we recommend that the IFC's add an attribute on these reified relationships to clarify the purpose of the relationship.

In the next section, we discuss the capabilities of the IFC's product model to indicate design changes.

2.1.3 Design Changes

Cost estimates need to be continually updated to reflect changes in the product model. Consequently, it is important for the IFC's to explicitly represent product model changes to aid the estimator in identifying their occurrence. Currently, changes to the product model are tracked through the related object IfcOwnerHistory. The attribute "ModifiedFlag" on IfcOwnerHistory has classification types that denote whether a change has occurred. Therefore, the value of the "ModifiedFlag" attribute will reveal whether the product has changed. However, the representation of changes to the product model is limited. Specifically, **the IFC's do not provide a representation to signify what attribute on the product has changed. Consequently, estimating applications that analyze IFC-based product models will have to compare each version of the product representation to determine the specific attribute of the product that has changed.**

In the next section, we discuss the capabilities of IFC's to maintain integration between product, cost and resource information as the project evolves.

2.2 Assessment of the IFC-based Project Model to Maintain Integration

The IFC's formalize the relationships between product, process, cost, and resource information. Figure 5 shows the objects, attributes, and relationships formalized to enable integration in an IFC-based project model. Yet how does this model support design changes like the one presented earlier in the case example, where the wall height was increased from 9.5' to 12.5'? As shown in Figure 5, the productivity rate, duration, and quantity are attributes that would be updated to incorporate this change. However, the IFC model does not provide a way to capture the *context* for the productivity rate, resources, and material unit costs selected. Consequently, the IFC project model will not help in detecting when those selections are no longer applicable, and as a result, are no different in this respect than existing software tools. Because the estimator's rationale for relating product and cost information is not explicit, estimators must manually identify the impact of design changes to determine whether the cost information selected is still applicable.

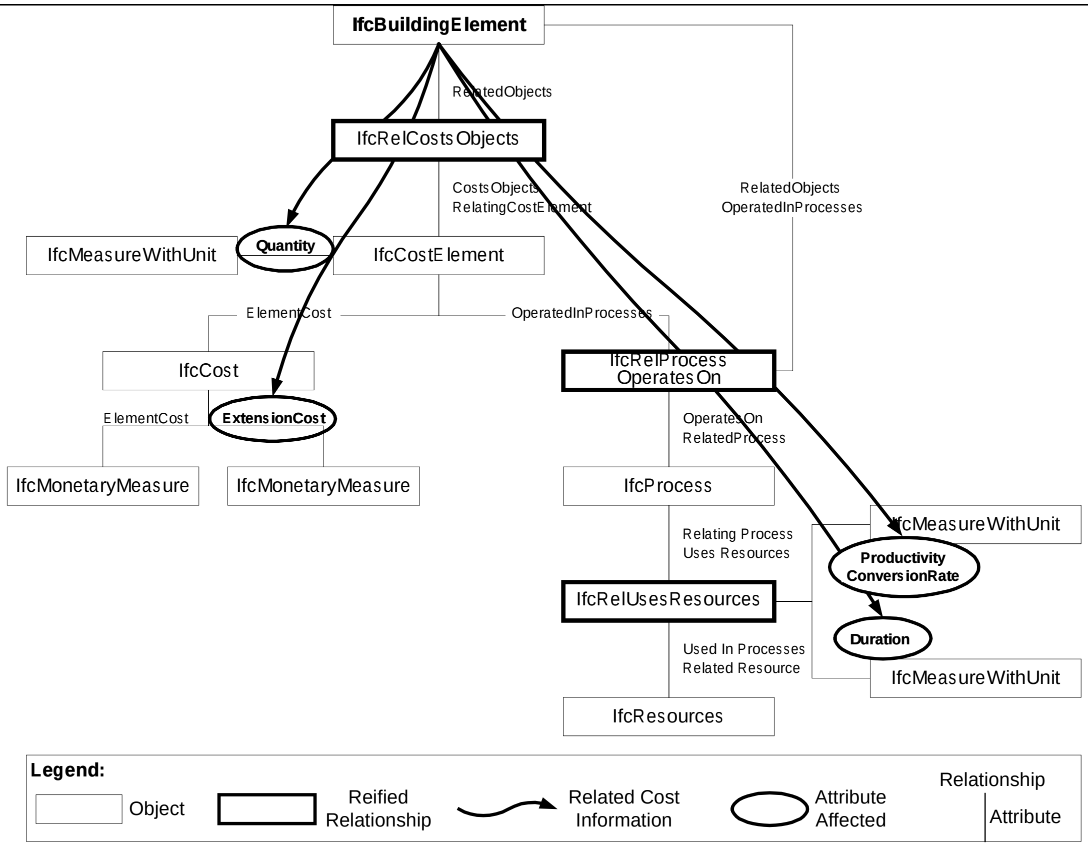


Figure 5: The objects, attributes, and reified relationships defined by the Industry Foundation Classes and information affected by design changes. The heavy lines with arrows show the specific cost information related to a building element that changes when the design changes. Note that an IFC-based project model captures the relationships between scope, cost, and resource information. However, there is no way to capture the *context* of the relationships.

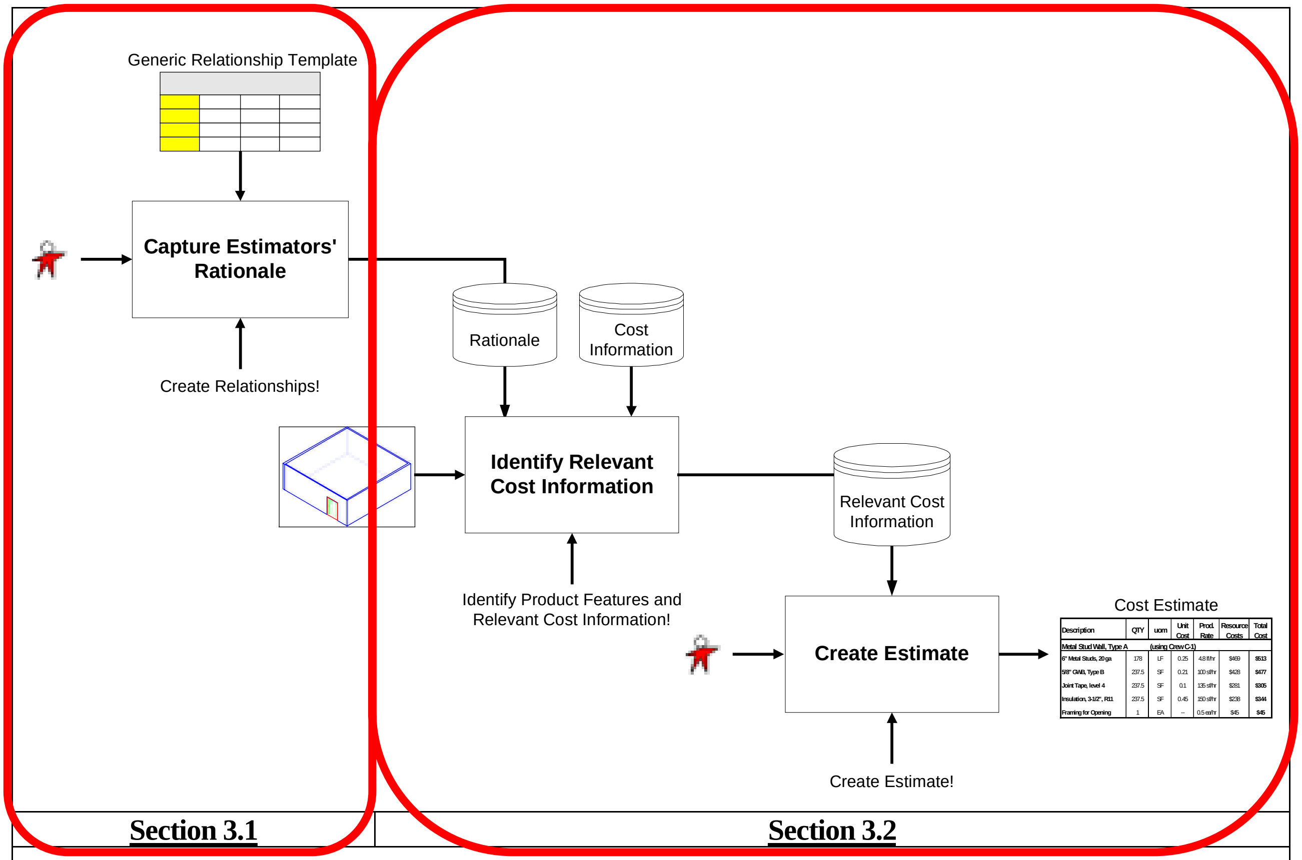
Should the *context* of the cost information be included in the IFC's? We do not have a clear answer to this question. We believe that it depends on the extent that cost information will be shared between different organizations. Based on our experience working for general contractors, we see a clear need for including the context in the IFC's. As a project engineer, we was responsible for reviewing change orders that often resulted from design changes. Many times the unit costs or productivity rates were different from the original quotation. The subcontractor tried to explain why the design change cost more but it was often a source of contention. Capturing the context of the cost information with the original quotation could alleviate this problem. Conversely, construction professionals may be resistant to sharing this information since cost estimating knowledge is typically a core competency of most construction firms. Moreover, it would be a complex task for the IFC's to provide a general representation of this context that supports multiple domains. At this point, we are not proposing that the IFC's should be extended to include the *context* of the construction cost information, but we see the potential benefits of such an inclusion.

In the next section, we discuss our vision of an estimating process that addresses many of the limitations of current tools.

3. Envisioned Estimating Process

Our vision is to develop a system that helps the estimator throughout the estimating process and throughout the project life-cycle. Specifically, there is a need for a flexible system for creating relationships between product and cost information and analysis tools that automatically identify the relevant cost information. The Appendix describes the initial prototype implementation in more detail. In this section, we will demonstrate how such a system would help the estimator to create relationships between product and cost information, to determine the impact of design decisions on the cost information selected, and to create estimates on future projects using the relationships between product and cost information created by the estimator.

Figure 6 shows an IDEF0 view of a possible system architecture. We will discuss each part of the system architecture in the following two sections. Section 3.1 will describe our vision for capturing estimators' rationale. Section 3.2 will describe our vision for using the computer-interpretable representation of estimators' rationale to identify relevant construction cost information and to create a cost estimate.



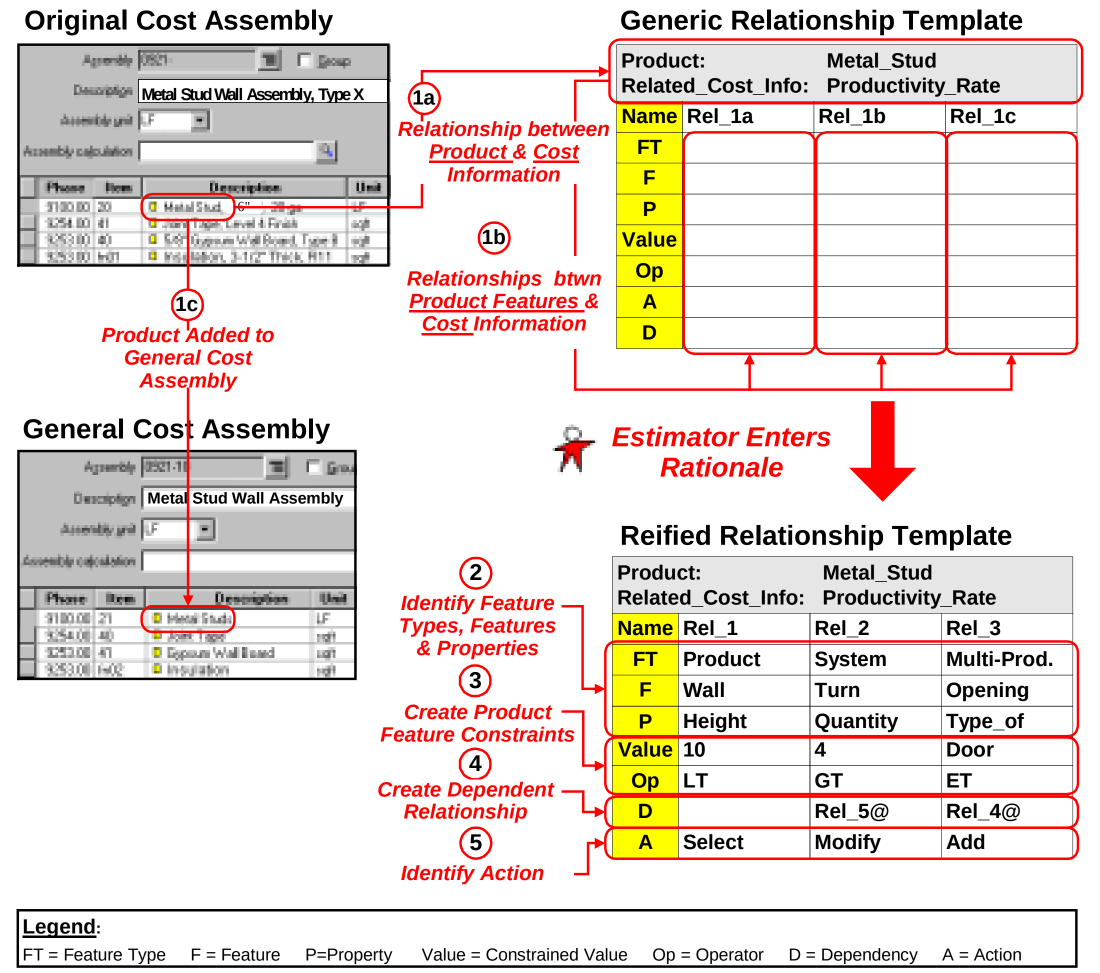
Description	QTY	uom	Unit Cost	Prod. Rate	Resource Costs	Total Cost
Metal Stud Wall, Type A (using Crew C-1)						
6" Metal Studs, 20 ga	178	LF	0.25	4.8 lf/hr	$469	$513
5/8" GWB, Type B	237.5	SF	0.21	100 sf/hr	$428	$477
Joint Tape, level 4	237.5	SF	0.1	135 sf/hr	$281	$305
Insulation, 3-1/2", R11	237.5	SF	0.45	150 sf/hr	$238	$344
Framing for Opening	1	EA	--	0.5 ea/hr	$45	$45

Figure 6: System architecture using an IDEF0 diagram.

3.1 Capturing Estimators' Rationale

Consider the case example discussed in the previous section. After the estimator created the cost assembly, what if the estimating system queried the estimator to ascertain what relationships between the product and the cost information dictated the applicability of that cost information. Figure 7 shows the original cost assembly and the generic relationship template that would be opened to enter the estimator's rationale. The initial implementation of the relationship template is described in section 9.2 of the appendix. Figure 7 also describes in detail the different steps that the estimator or system would execute to create the relationship between product and cost information.

After the estimator creates the relationships that represent their rationale for relating product and cost information, the system can help the estimator to identify when the relationships are no longer applicable in the case of design changes. We describe this process in the next section.



Generic Relationship Template

Product: Metal_Stud
Related_Cost_Info: Productivity_Rate

Name	Rel_1a	Rel_1b	Rel_1c
FT			
F			
P			
Value			
Op			
A			
D			

Reified Relationship Template

Product: Metal_Stud
Related_Cost_Info: Productivity_Rate

Name	Rel_1	Rel_2	Rel_3
FT	Product	System	Multi-Prod.
F	Wall	Turn	Opening
P	Height	Quantity	Type_of
Value	10	4	Door
Op	LT	GT	ET
D		Rel_5@	Rel_4@
A	Select	Modify	Add

Legend:
FT = Feature Type F = Feature P=Property Value = Constrained Value Op = Operator D = Dependency A = Action

Figure 7: Envisioned approach for capturing estimators' rationale using relationship templates. The following describes each step that is required by the estimator or the system:

1a) The estimator identifies the product and specific cost information that is captured in the relationship. In this case, the estimator is creating a relationship between the metal studs product class and the productivity rate.

1b) The estimator creates three relationships between the product features and the productivity rate. The estimator creates three relationships because three product features affect the productivity rate for installing metal studs.

1c) The system automatically creates a general cost assembly to capture the products selected in step 1a. The general cost assembly is needed so the estimator can create cost estimates with more detail than the product model. The general cost assembly for the walls in the example case contains metal studs, joint tape, gypsum wallboard, and insulation.

2) The estimator identifies the product feature, feature type, and property. The system will help with the selection of this information because there are limits to what can be entered. For example, the only properties available for the product feature *turn* in Rel_2 are the "quantity" or "orientation". If the estimator wants to consider both of these properties to obtain the quantity of a certain orientation, he/she would have to create dependent relationships (step 6). If the estimator wants to consider the quantity and the orientation independently, he/she needs to create two independent relationships.

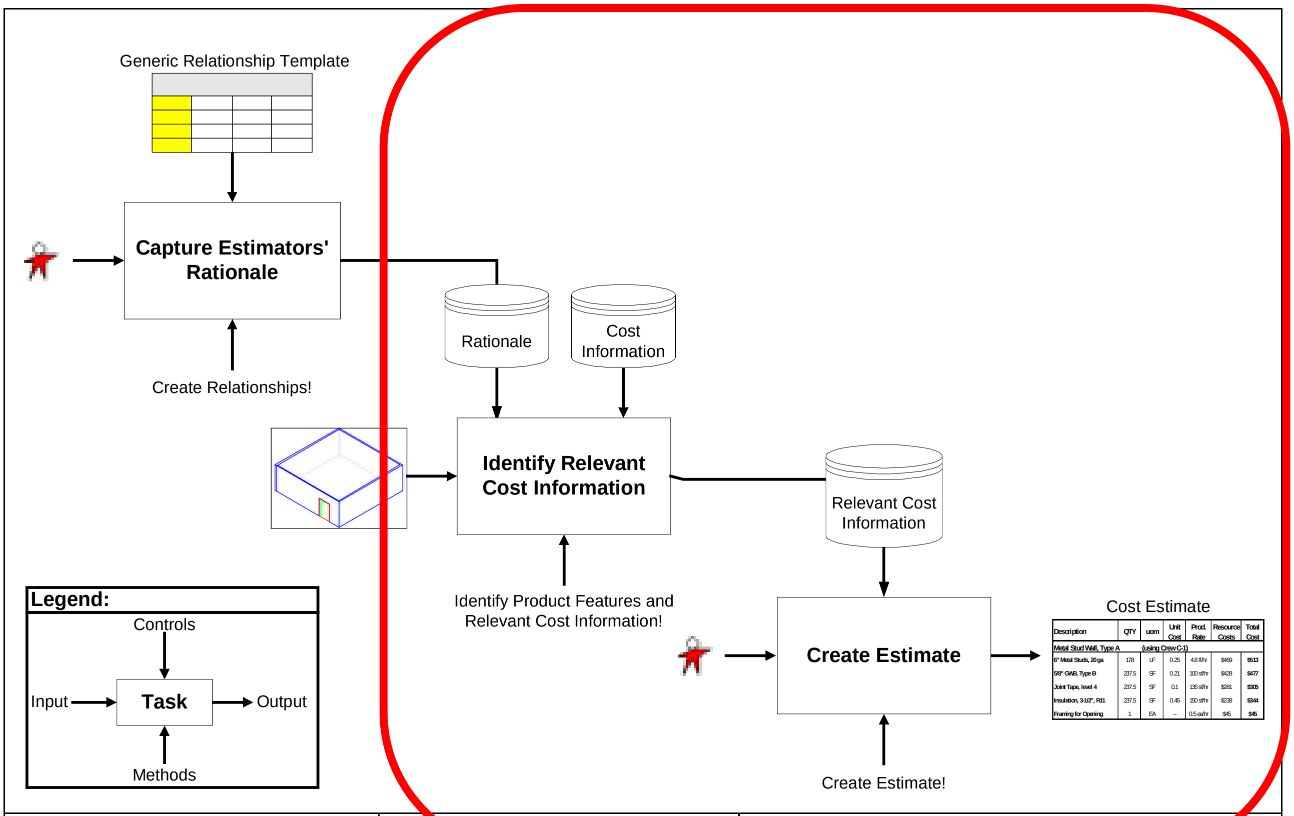
3) The estimator defines the product feature constraints that determine the applicability of the cost information. The operators ("Op") define whether the constraint is Less-than (LT), Greater-than (GT), or Equal-to (ET) the value in the "value" slot.

4) The estimator creates dependent relationships. For example, Rel_4 was identified as a dependent relationship to Rel_3. Rel_4 depends on Rel_3 because it contains the estimating rationale about adding production time based on the size of the opening identified in Rel_3.

5) The estimator selects the action so the system knows how to adjust the estimate to account for the constraint satisfaction. The available actions are select, modify, or add.

3.2 Identifying Relevant Construction Cost Information to Create a Cost Estimate

As estimators create the library of relationships, the system will help the estimator to create estimates on future projects. Our vision is to develop a prototype system that helps the estimator *throughout* the estimating process by using the computer-interpretable representation of the estimator's rationale to not only assist in maintaining a cost estimate as the design changes, but also to identify the relevant construction cost information for a product model. The current prototype system analyzes a product model to automatically identify product features and assist the estimator in identifying relevant cost information when generating cost estimates. Figure 8 describes the inputs, outputs, controls, and mechanisms needed to identify relevant cost information from a given product model and to create an estimate. The input product model is represented with the standard representation defined by the Industry Foundation Classes 2.0 Specification (IAI 1998).



Cost Estimate

Description	QTY	uom	Unit Cost	Prod. Rate	Resource Costs	Total Cost
Metal Stud Wall, Type A	(using Crew C-1)					
6" Metal Studs, 20 ga	178	LF	0.25	4.8 lf/hr	$469	**$513**
5/8" GWB, Type B	237.5	SF	0.21	100 sf/hr	$428	**$477**
Joint Tape, level 4	237.5	SF	0.1	135 sf/hr	$281	**$305**
Insulation, 3-1/2", R11	237.5	SF	0.45	150 sf/hr	$238	**$344**
Framing for Opening	1	EA	--	0.5 ea/hr	$45	**$45**

Figure 8a **Figure 8b** **Figure 8c**

Figure 8: System architecture using an IDEF0 diagram.

Figure 8a: Inputs, controls, methods and output of proposed approach for capturing estimators' rationale.

Figure 8b: Inputs, controls, methods and output of proposed approach for identifying relevant cost information and generating corresponding estimate.

Inputs: The product model represented using the IFC's.

Controls: The controls are the estimator's rationale for relating product and cost information and the corresponding cost information.

Methods: Reasoning mechanisms identify the product features, analyze the relationships to test the applicability of the cost information, and identify the relevant resource, resource productivity, and material cost.

Output: A list of the relevant construction cost information.

Figure 8c: Inputs, controls, methods, and output of for creating an estimate.

Inputs: The estimator selects the most appropriate cost information from lists of relevant cost information identified by the system. The lists of relevant cost information act as a Control in the IDEF diagram.

Controls: The relevant cost information identified by the system using the estimator's rationale captured in the relationship template.

Methods: Reasoning mechanisms calculate the quantities, durations, and corresponding costs for each item.

Output: A cost estimate that is related to the IFC-based product model. The cost estimate also contains links to the relationships that define the context for why that cost information was selected.

The framework presented in Figure 7 suggests the need for several formalisms to capture estimators' rationale for relating product and cost information. First, the relationships between product and cost information need to explicitly represent product feature types and features, constraints on product features, dependencies between features, and actions that determine how the estimate should be adjusted to provide a computer-interpretable representation of the estimator's rationale. Second, product feature types need to be formalized to provide a framework that only permits selections of properties of product features that are *logical*, or make sense, for the particular estimating situation. For example, the logical properties of the product feature "turn" are the *quantity* and *orientation*. Formalizing the relationships between product and cost information and product feature types will enable estimators to enter their rationale in a systematic way and provide a computer-interpretable representation that enables automatic identification of relevant construction cost information. Figure 8 describes processes to automatically identify product features and relevant construction cost information and suggests the need for several formalisms to support these processes. In the next section, we describe in detail each of these research needs.

4. Research Needs

As described in the previous section, several formalisms need to be developed to meet the practical needs of cost estimators. In the following four sections, we describe these formalisms in detail. Section 4.1 describes the classification of product feature types that are necessary to enable the automatic detection of product features. This section also describes the classification of product features and properties to enable the system to prohibit estimators from selecting illogical features and properties. Section 4.2 describes the characteristics of the estimators' rationale for relating product and cost information. Section 4.3 describes the reasoning mechanisms developed to infer the existence of product features from an IFC-based product model. Finally, Section 4.4 describes the reasoning mechanisms needed to identify relevant construction cost information using the computer interpretable representation of estimators' rationale.

4.1 Classification of Product Feature Types and Properties

Identifying feature types is important for enabling the automatic identification of product features from an IFC-based product model. In Figure 7, we presented our proposed framework for capturing estimators' rationale. In Step 2 of the proposed framework, the estimator selects the following information to identify product features: the feature type, the feature, and the property of the feature. The proposed approach will allow the computer system to aid the estimator in selecting product features and properties. The system will identify the appropriate product feature and properties for a given feature type to prevent the estimator from selecting illogical properties. For example, the available properties for the feature "turn" are the *orientation* and the *quantity*. It is imaginable to provide this type of assistance without classifying product feature types and properties. Reasoning mechanisms could be created to check the validity of properties for every feature. However, it would be very cumbersome to maintain these mechanisms since the system would have to check the validity of every property, and estimators would have to add new mechanisms for each new property. Therefore, it would be preferable to give estimators a way to classify product feature types and properties and to store and maintain the classifications instead of relying on reasoning mechanisms to infer the appropriate classifications.

We have classified the product features and properties shown in Table 1 into five different types based on the level of analysis on the product model that is required. The five different feature types and corresponding properties and product features are shown in Figure 9. We have focused on the first three feature types in this research study since they were identified repeatedly in previous research.

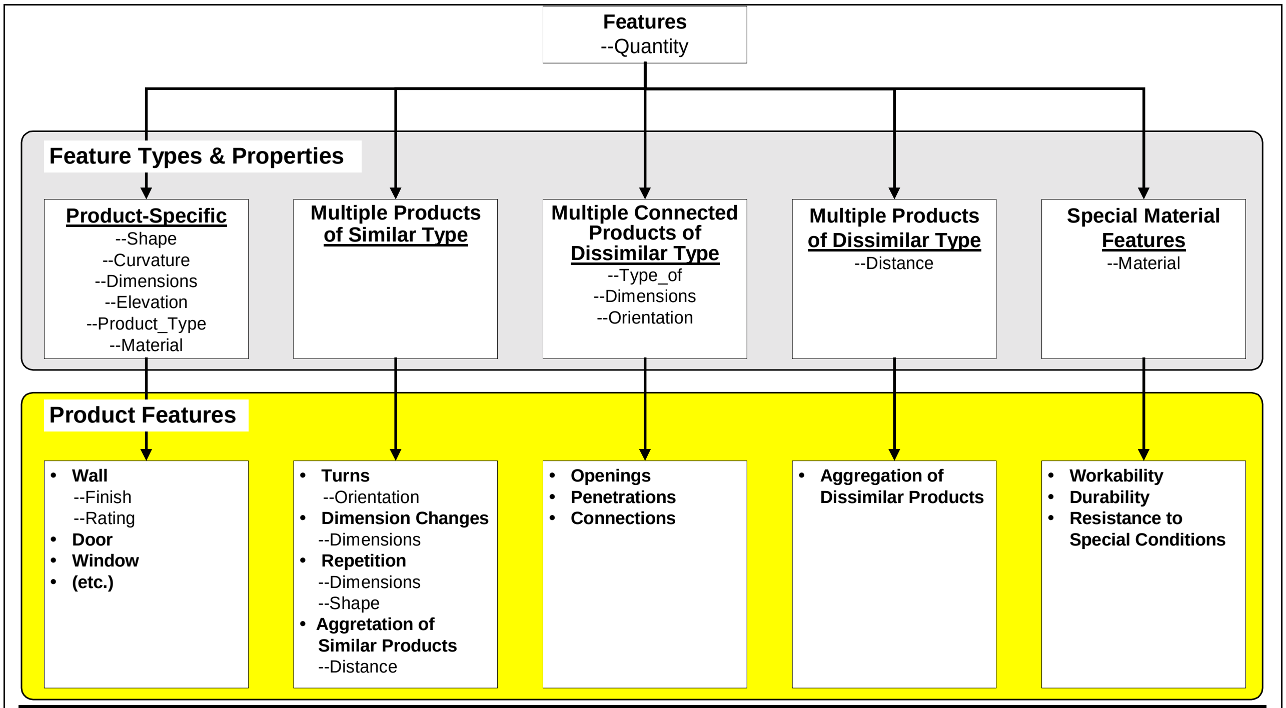


Figure 9: Classification of product feature types and properties. Properties can be specific to feature types and, through inheritance, are appropriate to all product features of that type. For example, the property "dimension" for the *Product-Specific* feature type. Alternatively, properties can be specific to a product feature, such as the property "orientation" for the product feature *turn*.

4.2 Characteristics of Estimators' Rationale

In the case example, we divided the estimating process into five different steps to illustrate the different aspects of the estimator's rationale: 1) identify product and cost information, 2) identify product features and properties, 3) capture constraints on product features, 4) incorporate dependencies between features, and 5) adjust estimate to incorporate the affect of product features by identifying appropriate actions. Step 1 defines the relationship between the product and cost information and steps 2-5 define the characteristics of the estimator's rationale for relating the product and cost information. Hence, the estimators' rationale for relating product and cost information includes at least the following four characteristics:

1) product features,
2) constraints on product features,
3) dependencies between product features, and
4) actions.

The previous section described product features in detail. Now, we will devote the remainder of this section to describing constraints on product features, dependencies between product features, and actions.

Constraints on product features determine the applicability of the relationship. For example, the relationship between the wall height and the productivity rate limited the applicability of the selected cost information to walls with height less than 10'. If the constraints are satisfied then the cost information is selected, modified, or added based on the action identified. In the proposed framework shown in Figure 7, estimators represent constraints by identifying the feature property, values, and operators. The available values are dictated by the feature properties identified. For example, for the feature property wall "height", the available value must be a number. In contrast, for the feature property opening "type", the available values are doors and windows. The available operators are Less-than, Greater-than, and Equal-to.

The case example demonstrates that the combination of multiple product features affects construction cost information and creates dependencies between product features. For example, the quantity of wall turns alone was not useful to the estimator. The quantity of wall turns relative to the total length of wall was the critical information the estimator needed to determine how the productivity rate should be modified, as shown in Figure 2. We represent this occurrence as two dependent relationships between product features. Another example that illustrates the dependencies between features was the opening in the wall. The first relationship identifies the type of opening and the second dependent relationship identifies the size of the opening to determine the production time.

Satisfaction of constraints on product features has different consequences on the cost estimate being generated. For example, the wall height led to the *selection* of the productivity rate for the walls, wall turns led to the *modification* of the productivity information, and the existence of an opening resulted in the *addition* of production time, as shown in Figure 2. In addition, there are also limits to and dependencies between these actions. Resources can not be *modified*, cost information can not be *modified* until it has first been *selected,* and the *addition* of certain cost information requires a *selection*. We have started to implement a flexible framework that allows estimators to select, modify, or add cost information and consequently, adjust the estimate in different ways to account for the different effects of product feature.

Sections 4.1 and 4.2 provided the foundation for developing a formalized representation of estimators' rationale for relating product and cost information. Sections 4.2 and 4.3 build on this foundation by developing reasoning mechanisms that automatically analyze the formalized relationships to identify product features and determine the relevant cost information for a given IFC-based product model.

4.3 Mechanisms to Identify Product Features

Because IFC's do not explicitly represent many product features affecting construction cost information, mechanisms need to be developed to *infer* these product features. Table 1 shows 18 product features and properties identified thus far in this research. This table also demonstrates that many product features affect different types of work. For example, turns affect the costs of piping work and metal-stud wall construction. Consequently, the mechanisms need to be general so that they can identify the same product features for multiple types of work.

The proposed approach for capturing an estimator's rationale allows estimators to identify product features on multiple types of components. For example, estimators can identify "turns" for pipe runs, ductwork, or walls. To provide computer assistance for this task, reasoning blocks for each product feature need to be developed. These reasoning blocks help identify a product feature regardless of the type of product. Therefore, these general reasoning blocks would infer the existence of certain product features. As an example of such a reasoning block, Figure 4 describes the reasoning process to identify a turn in an IFC-based product model.

Identifying the existence and extent of product features is the first step in determining the relevant construction cost information for a given product. The next section describes additional mechanisms that are required to complete this analysis and automatically select, modify, or add the relevant construction cost information.

4.4 Mechanisms to Identify Relevant Construction Cost Information

Two different reasoning mechanisms are needed to enable the automatic selection, modification, or addition of relevant construction cost information: 1) analyzing the constraints on the product features and 2) adjusting the estimate if the constraints are satisfied. To accomplish the first task, we plan to implement constraint-satisfaction reasoning mechanisms to determine whether the constraints on the product features are satisfied for the specific product model. This process essentially analyzes all the cost information and eliminates the information that is not applicable. The second reasoning mechanism figures out how to handle the satisfaction of the constraint. There are two possible analyses that will be triggered if the constraint is satisfied: 1) adjust the estimate by selecting, modifying or adding cost information or 2) analyze a dependent relationship to further refine the search for the specific cost information that is applicable. If the system identifies many possible solutions, the user will have the option of either manually selecting the cost information or having the system generate cost estimates for the possible solutions to identify the low-cost alternative.

In this chapter, we have identified and elaborated the formalisms that are necessary to implement the envisioned estimating process. The next chapter describes the research background and points of departure.

5. Relevant Background And Prior Work

Prior research in the following three areas: (1) Construction Cost Estimating, (2) Product Features, and (3) Capturing Rationale provides background for the envisioned system. The following four sections discuss the research background in each of these areas. Each section concludes with the points of departure and expected contributions.

5.1 Construction Cost Estimating

We have classified the previous research on construction cost estimating into the following two areas: 1) cost estimating with rationale not included and 2) cost estimating with rationale included. Section 5.1.1 discusses previous research on cost estimating that does not include estimators' rationale. Section 5.1.2 describes previous research that captured the estimators' rationale when selecting construction cost information or generating cost estimates. This area of research relates to section 4.2.

5.1.1 Cost Estimating Rationale Not Included

Numerous research efforts have tried to automate different parts of the cost estimating process. Laitinen (1998) developed a Cost and Value Engineering system (COVE) that relates product and process information. Specifically, the estimator manually attaches cost assemblies to the product objects to automate the quantity takeoff process, and to create relationships between product, resource, location, and costs. The Information/Integration for Construction (ICON) project developed a framework for integrating information systems in the construction industry, including the integration of cost and product information (Aouad et al. 1994). The Object-model-based Project Information System (OPIS) was developed to improve the integration of construction information through standardized object-oriented models (Froese 1992). OPIS explicitly related product, activity, resource, and method information. The Open Systems for Construction (OSCON) project integrates 3D CAD information with a resource-based estimating application (Aouad et al. 1997). The Construction Method Modeler (CMM) uses a formal method definition to generate 4D production models. The 4D production models explicitly model the relationships between product, resource, and activity information (Aalami 1998). Similarly, Timberline's Precision Estimating software (Timberline Software Company 1998) links with Ketiv's Archt Architectural Design software (Ketiv Technologies 1997) that was used on the Sequus project discussed in Chapter 1 (Staub et al. 1999).

These research efforts and software tools either require manual assignment of cost information or rely solely on product type information to determine the corresponding costs. However, as the case example illustrates, product features also play a critical role in the cost estimating process. For example, the change in wall height resulted in changes to the appropriate cost information yet the product type didn't change.

Therefore, these systems would not detect that the relevant construction cost information was no longer applicable. Hence, product type information needs to be used as a starting point to determine the relevant cost information. However, as already shown, the selection of cost information needs to be further refined based on the existence of product features and constraints on product features.

5.1.2 Cost Estimating Rationale Included

Table 1 shows the product features and properties affecting construction cost information that were identified in previous research efforts (R.S. Means Company 1998; Fischer 1991; Hanna and Sanvido 1990; Hendrickson et al. 1987; Peurifoy and Oberlender 1989; Sanders and Thomas 1991; Smith and Hanna 1993; Tah et al. 1991). These same research efforts also incorporated the estimator's rationale for how the product features affect the different aspects of construction cost estimating, such as the selection of methods or productivity rates. For example, Fischer (1991) identified that the applicability of flying forms for concrete slabs is limited by a 20' maximum floor-to-floor height.

However, previous research that incorporated estimating rationale did not support this general representation explicitly. Hence there is a need to develop a generic relationship template that allows estimators to enter their rationale in a computer-interpretable form. The relationship template needs to capture the relationship between product and cost information, product features and cost information, constraints on product features, dependencies between product features, and the action for how the estimate should be adjusted, as shown in Figure 4. Table 3 shows five different representation and reasoning approaches used to incorporate estimators' rationale and summarizes the limitations of these approaches.

Object-oriented techniques similar to those used by Fischer (1991) appear to be a useful technology to implement the envisioned system. Statistical modeling, similar to Sanders and Thomas (1991), of product features could be included to identify the relevant productivity rate.

Reasoning Approach	Representation of Rationale	Limitations
AI Techniques ***Tah et al 1991***	Implicit in Code	• Hundreds of production rules • Requires user input of product information at runtime • Difficult to extend
Object-Oriented Programming ***Fischer 1991***	Attributes and Implicit in Code	• Captures product, product feature, and constraints in attributes. An example attribute is "max-floor-to-floor-height". • Complex features and constraints are represented implicitly in the program code. An example is the feature and property "distance between columns". • Reasoning is not general because rationale is represented in single attribute or in computer code. • Can only *select* methods for concrete formwork
Statistical Models ***Sanders & Thomas 1991***	Attributes	• Time consuming to create • Difficult to extend because users have to gather additional information for every new product feature • Requires user input for all the features modeled every time it is used • Representation of rationale is limited to single attributes • Reasoning is not general because rationale is represented in single attributes
Case-based Reasoning ***Yau & Yang 1998***	Attributes	• Only works for a limited number of possibilities • Representation of rationale is limited to single attributes • Reasoning is not general because rationale is represented in single attributes • Can only *select* relevant costs
Neural Networks ***Moselhi & Siqueira 1998***	Attributes	• Representation of rationale is limited to single attributes • Reasoning is not general because rationale is represented in single attributes • Can only *select* relevant costs

Table 3: Approaches used to capture estimators' rationale and to identify relevant construction cost information. None of the approaches identified above provided a framework for estimators to *enter* their rationale explicitly.

In summary, previous approaches do not explicitly capture the four different characteristics of the estimators' rationale. For example, Fischer (1991) captured the product, product features, and constraints on product features. However, he modeled this information as a single attribute. Consequently, the reasoning mechanisms that identified product features were not general and could only be used to find the product feature for the specific product in the attribute. In addition, these research efforts did not explicitly represent the knowledge about when to select, modify, or add construction cost information. Finally, the dependencies between product features were implicit in the computer code.

5.1.3 Summary of Necessary Research and Expected Contributions to Construction Cost Estimating

Practitioners and researchers alike have discussed integrating design and cost information for decades. Standards in the form of Industry Foundation Classes have been developed to facilitate this process. Yet we have demonstrated through the case example that integration can not be achieved without capturing the context of the information. In the case example, the design change of the wall from 9.5' to 12.5' not only increased the quantities of materials in the wall but also required changes to the cost information and a new cost assembly. The impact of design changes on the *applicability* of the cost information can only be understood by explicitly modeling estimators' rationale.

Building on the results of prior work the following contributions are necessary to realize the envisioned system:

A formalized approach to capture estimators' rationale explicitly: Previously, the product, product features, and constraints had to be captured in a single attribute. Additionally, the output of these systems was the selection of a certain part of the cost estimating process, such as method selection. They did not incorporate the knowledge about when to select, modify or add the relevant cost information.

A framework for estimators to enter their rationale as relationships in a computer interpretable form and reasoning mechanisms to analyze when the relationships have been satisfied: Previous approaches have worked more like black boxes that perform some analysis and provide an answer about what cost information to select. However, these approaches are often difficult to integrate into current industry practices because it is difficult to understand why certain cost information was chosen by the system.

5.2 Product Features

The following sections discuss different research efforts on product features. Section 5.2.1 discusses the classification of product features and relates to section 4.1. Section 5.2.2 discusses the reasoning mechanisms developed to automatically recognize product features and relates to section 4.3.

5.2.1 Product Feature Classification

Features that affect manufacturing processes have been classified in the following ways: 1) depression features such as hole, pocket, and slot, 2) protrusion features such as boss, rib, and bridge, 3) transition features such as fillet and chamfer, and 4) special features such as treatment and surface finish (Allada and Anand 1996). Formalizing different types of product features enables the automatic extraction of manufacturing features and mappings to manufacturing processes. Cunningham and Dixon (1988) determined that sets of features could be deduced by a *process-activity pair*. The manufacturing processes they studied were: aluminum extrusion, aluminum casting, plastic injection molding, steel forging, sheet metal stamping, and machining. The activities they studied were: manufacturability evaluation, die/tool design, first order mechanical analysis, and cost analysis. For example, a feature set can be created for the [activity of] cost analysis [in the process of] aluminum casting. A corresponding example from construction would be a feature set for the [activity of] cost estimating [in the process of] installing masonry. However, the process-activity pair does not support the generalization of product features across multiple types of work, which, as shown earlier, is necessary to make a product model based estimating system generally applicable.

5.2.2 Product Feature Recognition

Feature recognition has been extensively researched in the manufacturing industry (Chamberlain et al. 1993; Gadh and Prinz 1992; Henderson 1984; Kung 1984; Nnaji et al. 1991). Feature recognition systems can automatically identify features after the part is modeled by using the geometric and topological data from the CAD model. Typically, the geometry and topology of a particular product is analyzed to infer the presence of a particular type of feature. An alternative approach is to use a feature based design system. This method allows designers to add features as they create the product model. This approach eliminates the need for feature recognition. However, many of the features identified in Table 1 are only relevant to the construction cost estimator. For example, the product feature "repetition" is an estimator's interpretation of the product model. It is unreasonable to expect the designer to identify the existence of "repetition" because it depends on the estimator's criteria, which can be shape, size, or other properties. Consequently, it is unrealistic to expect the designer to add these features to the product model. As illustrated in Figure 4, mechanisms are needed to infer the presence of the particular product features affecting construction cost information identified.

Clayton et al (1996) added features to graphical models to support automated analysis of design behavior and function using the Semantic Modeling Extension (SME). Architects add semantic information about product features to graphical objects to support automated analysis of egress, energy use, and construction costs. SME, however, requires the user to manually link the semantic information about product features to the graphical objects. Moreover, the construction cost analysis links product information to cost information relying solely on the type of component and ignoring the effect of product features on this selection. Haymaker (1999) automated product recognition based on the geometric and topological relationships between products. He uses the notion of 'filters' to infer the existence of a product object, such as a window, thereby eliminating the need to identify all product semantics *a priori*. However, the IFC's have identified the necessary product objects to represent the building elements needed for cost estimating. The envisioned system will use an IFC-based product model to infer the existence of product features rather than infer the existence of product objects.

5.2.3 Summary of Necessary Research and Expected Contributions in Product Features

The following research is necessary to develop the envisioned cost estimating system:

(1) Confirm the generality of the product features identified for specific types of work, such as piping, masonry, and concrete work.

(2) Formalize the different types of product features to support construction cost estimating of building components. We will use the formalisms developed to classify product features that affect manufacturing processes as a starting point.

(3) Create algorithms that infer the existence of product features based on the geometric properties of components and the topological relationships between components, similar to feature recognition in the manufacturing industry.

(4) Test the content of the IFC 2.0 to support feature recognition of product features for construction cost estimating of building components.

5.3 Capturing Rationale

We need to develop a generic way to capture an estimator's rationale for relating product and cost information. The following two sections discuss previous research approaches for capturing rationale for different engineering activities: 1) capturing design rationale to coordinate the design process, 2) capturing planning rationale to automate the planning process.

5.3.1 Capturing Design Rationale to Coordinate the Design Process

Nielsen et al (1991) captured the designer's intent about how geometric forms should be sized and related geometrically to one another. They defined form as consisting of both configuration and geometry information and intent as restrictions on form. The geometric aspect of the design's form is modeled as a collection of all restraints on geometric attributes. The restraints contain qualitative or quantitative information about the attribute's value. Specifically, a restraint is modeled as a collection of scalar restrictions that are associated with an importance level and a degree of certainty. An example of a single restriction described in this research is "the value of an attribute should be less than six inches and more than exactly four inches." There is also a need to represent qualitative and quantitative restrictions on the design's form.

Jeon et al (1999) approached design coordination as a constraint satisfaction problem (CSP). CSP's are defined as "consisting of a set of variables, each associated with a domain of values, and a set of constraints, each of which is expressed as a relation, defined on some subset of variables, whose types are all the simultaneous value assignments to the members of this variable subset that are legal." Variables can be a component or a feature, domain values are possible values the variable can take, and a relation is a constraint expression. We anticipate using the variable-domain-relation combination to represent the constraints on product features for construction cost information. For example, to model the constraint on wall turns, the variable would be the "turn", the domain would be the "orientation" and the "quantity", and the relation would be wall turns "less than four". However, the variable-domain-relation representation does not provide sufficient representation to incorporate all aspects of an estimator's rationale. Therefore, the existing representation needs to be extended to include feature types and properties, dependencies between product features, and actions.

Garcia et al (1993) developed a system that captured design rationale to support analysis of the design process for documentation purposes. The system monitors the design process and determines the expected design solution for a given design based on the design criteria. The design criteria are represented as soft constraints imposed on the design. Each design alternative contains a qualitative evaluation for each criterion, such as "minimum noise in space", which is represented as a single attribute. The representation of constraints needs to be extended by separating the product from the constraint to allow estimators to easily identify product features and constraints independently. In this regard, we can build on Garcia (1993), who also explicitly represents dependencies between constraints when determining the most appropriate design solution.

5.3.2 Capturing Planning Rationale to Automate the Planning Process

Myers (1996) developed an Advisable Planner framework that links an advice-taking interface to Artificial Intelligence planning technology. Her goal was to provide a framework that allows planners to influence the plan generation process in terms that are meaningful to them. Specifically, the Advisable Planner translates advice into the appropriate internal representation for the planner. Advice is entered in a natural language and then goes through two processes to translate the advice to internal constraints that are defined in terms of planner-specific operators, goals, and individuals. A framework is needed that allows the *users* to enter the rationale. However, we will not use the advice translation process to translate user statements into computer-interpretable constraints. Rather, we will develop a framework that allows estimators to enter the constraints explicitly.

Aalami (1998) represented planning rationale in the form of computer-interpretable construction method model templates (CMMT). Each CMMT represents planning knowledge as abstracted skeletal plans that define a set of general activity types and their associated activity elaboration and sequencing knowledge. The Construction Method Modeler (CMM) system can then automatically customize the abstractly represented planning knowledge in a method model to the specific context of the project. The output of the system is a 4D production model that links produce, process, and resources that reflect the application of a particular construction method. The use of templates to capture an estimator's rationale for relating product and cost information, as shown in Figure 4, appears appropriate for the envisioned system.

5.3.3 Summary of Necessary Research and Expected Contributions in Capturing Rationale

The following research is necessary to capture estimators' rationale in a computer-interpretable form:

(1) Provide a general representation of estimators' rationale that represents both the qualitative and quantitative restrictions on the design's form.

(2) Extend the variable-domain-relation representation used to capture design rationale to include feature types and properties, dependencies between product features, and actions to support the representation of estimators' rationale.

(3) Explicitly represent constraints and features independently to allow estimators to easily identify product features and constraints for different types of work.

(4) Develop a template-like representation so estimators can easily input their rationale for relating product and cost information in a computer-interpretable form.

6. Summary of Research Contributions and Practical Implications

Expected contributions of this research are:

- Formalization of relationships between product features and construction cost information.
- Classification of product feature types to support construction cost estimating.
- Algorithms that automatically identify product features affecting construction costs.
- Automatic selection, modification, or addition of relevant construction cost information.
- Confirmation of generality of product features affecting construction cost information that were identified in previous research.
- Extension of construction project models to support the *context* for construction cost information.
- Testing of the IFC 2.0 specification to support feature recognition of product features for construction cost estimating.

Practical implications of this research are:

- Improved reliability of estimates through consistent use of resources and product information.
- Improved reliability of estimating information for use on future projects.
- Improved efficiency in identifying the impact of design changes on construction cost information.
- Formalized approach to incorporate the effect of product features on construction cost information.
- Increased efficiency of estimating process.
- Fewer "workarounds" when creating cost estimates.
- Less information management requirements through explicit representation of the *context* of the cost information.
- More consistent estimating process.
- Better documented estimates.

7. Future Extensions

In addition to product features, project featues also influence project costs. They affect the project as a whole, such as resource availability, environmental effects, and site access (Brown 1986; Oglesby et al. 1989; Ogunlana and Thorpe 1991; Paulson 1975; Skitmore 1988). Future extensions could incorporate the effect of project features on construction costs. This extension would also provide a useful investigation of the IFC's *project feature* representation to support construction cost estimating.

The case example demonstrated the effect of product features on construction costs. It would be useful to extend the proposed framework to provide feedback to designers so they understand the impact of their design decisions on construction costs. The explicit representation of product features and their impact on construction costs would help the designers to identify the specific parts of the design that could be eliminated or revised to provide cost-effective designs.

8. References

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

Allada, V., and Anand, S. (1996). "Machine Understanding of Manufacturing Features." *International Journal of Production Research*, 34(7), 1791-1819.

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.

Bjork, B. C. (1991). "A Unified Approach for Modeling Construction Information." *Building and Environment*, 27(2), 173-194.

Brown, J. "Estimating Labor Productivity." *30th Annual Meeting of the American Association of Cost Engineers,* Chicago, Illinois, L-3.1-L-3.6.

Chamberlain, M. A., Joneja, A., and Chang, T. C. (1993). "Protrusion-features handling in design and manufacturing planning." *Computer Aided Design*, 25, 19-28.

Cunningham, T., Mantripragada, R., Lee, D., Thornton, A., Whitney, D. (1996). "Definition, Analysis, and Planning of a Flexible Assembly Process." *Japan/USA Symposium on Flexible Automation*, 2, 767-778.

Di Marco, P. E., Charles F. ; Ishii, Kos. (1994). "Compatibility Analysis of Product Designfor Recyclability and Reuse." *Computers in Engineering*, 1, 105-112.

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

Froese, T.M. (1996). "Models of Construction Process Information." *Journal of Computing in Civil Engineering*, 10(3), 183-193.

Froese, T. M. (1992). "Integrated Computer-Aided Project Management Through Standard Object-Oriented Models," Ph. D. Thesis, Stanford University, Stanford.

Froese, Thomas; Fischer, Martin; Grobler, Francois; Ritzenthaler, John; Yu, Kevin; Sutherland, Stuart; Staub, Sheryl; Akinci, Burcu; Akbas, Ragip; Koo, Bonsang; Barron, Alex, and Kunz, John. (1999). "Industry Foundation Classes for Project Management-A Trial Implementation." ITCON (Accepted for Publication).

Gadh, R., and Prinz, F. B. (1992). “Recognition of geometric forms using the differential depth filter.” *Computer Aided Design*, 24, 583-598.

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

Haymaker, J. (1999). “Filter Mediated Design: Generating Coherence in (collaborative) Design,” Masters Thesis, Massachusetts Institute of Technology.

Henderson, M. R. (1984). “Extraction of feature information from three dimensional CAD data,” PhD, Purdue University.

Hendrickson, C., Martinelli, D., and Rehak, D. (1987). “Hierarchical Rule-Based Activity Duration Estimation.” *Journal of Construction Engineering and Management*, 113(2), 288-301.

IAI. (1998). “Industry Foundation Classes - Release 2.0 Specifications, IFC Object Models for AEC Projects.” .

Ishii, K., and Mukherjee, S. (1992). “Post Manufacturing Issues in Life-Cycle Design.” *Design for Manufacture*, 51, 49-56.

Ketiv Technolgies. (1997). *Archt, Users Documentation*, Portland, Oregon.

Kung, H. (1984). “An investigation into the development of process plans from solid geometric modeling representation,” PhD, Oklahoma State University.

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

Luiten, B. (1994). “Computer Aided Design for Construction in the Building Industry,” Ph. D. Thesis, Delft University.

Luiten, G. T., Froese, T. M., Bjork, B., Cooper, G., Junge, R., Karstila, K., and Oxman, R. “An Information Reference Model for Architecture, Engineering and Construction.” *First International Conference on the Management of Information Technology for Construction,* Singapore, 391-406.

R. S. Means Company. (1998). *R.S. Means Assemblies Cost Data*, R.S. Means Company, Inc.

Merhar, C., Chong, C., and Ishii, Kos. (1994). “Simultaneous design for manufacturing process selection of engineering plastics.” *International Journal of Materials and Product Technology*, 9(1/2/3), 61-78.

Moselhi, O., and Siqueira, I. “Neural Networks for Cost Estimating of Structural Steel Buildings.” *42nd Annual Meeting of AACE*, Cincinnati, Ohio.

Nnaji, B. O., Kang, T. S., Yeh, S., and Chen, J. P. (1991). “Feature reasoning for sheet metal components.” *International Journal of Production Research*, 29, 1867-1896.

Oglesby, C. H., Parker, H. W., and Howell, G. A. (1989). *Productivity Improvement in Construction*, MC Graw-Hill Inc., New York, N. Y.

Ogunlana, S., and Thorpe, A. (1991). “Nature of estimating accuracy. Developing correct associations.” *Building and Environment*, 26(2), 77-86.

Paulson, B. (1975). “Estimation and Control of Constructoin Labor Costs.” *Journal of the Construction Division*, 101(003), 623-633.

Peurifoy, R. L., and Oberlender, G. D. (1989). *Estimating Construction Costs*, McGraw-Hill.

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

Skitmore, M. (1988). “Factors affecting accuracy of engineers' estimates.” *Cost Engineering*, 30(10), 16-23.

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

Staub, S., Fischer, M., and Spradlin, M. (1999). “Into the Fourth Dimension.” Civil Engineering, ASCE, 69(5), 44-47.

Tah, J. H. M., Thorpe, A., and McCaffer, R. “Decision Support System for Drainage Cost Estimating.” *Second International Conference on the Application of Artificial Intelligence to Civil and Structural Engineering*, Oxford, England, 29-35.

Timberline Software Company (1998). *Precision Estimating Plus, Users Documentation*, Beaverton, Oregon.

Yau, N., and Jyh-Bin, Y. (1998). “Case-Based Reasoning in Construction Management.” *Computer-Aided Civil and Infrastructure Engineering*, 13, 143-150.

9. Appendix: Initial Implementation of Prototype Estimating System

In this chapter, we describe our initial implementation of an estimating system that captures estimators' rationale, identifies relevant construction cost information for a given IFC-based product model, and creates a cost estimate. Specifically, we describe the functionality of the prototype estimating system and use of IFC's to support that functionality. This chapter is organized as follows:

9.1 User Interface to Estimating System

9.2 User Interface to Relationship Template that Captures Estimators' Rationale

9.3 User Interface illustrating IFC's Representation of Products at Multiple Levels of Detail

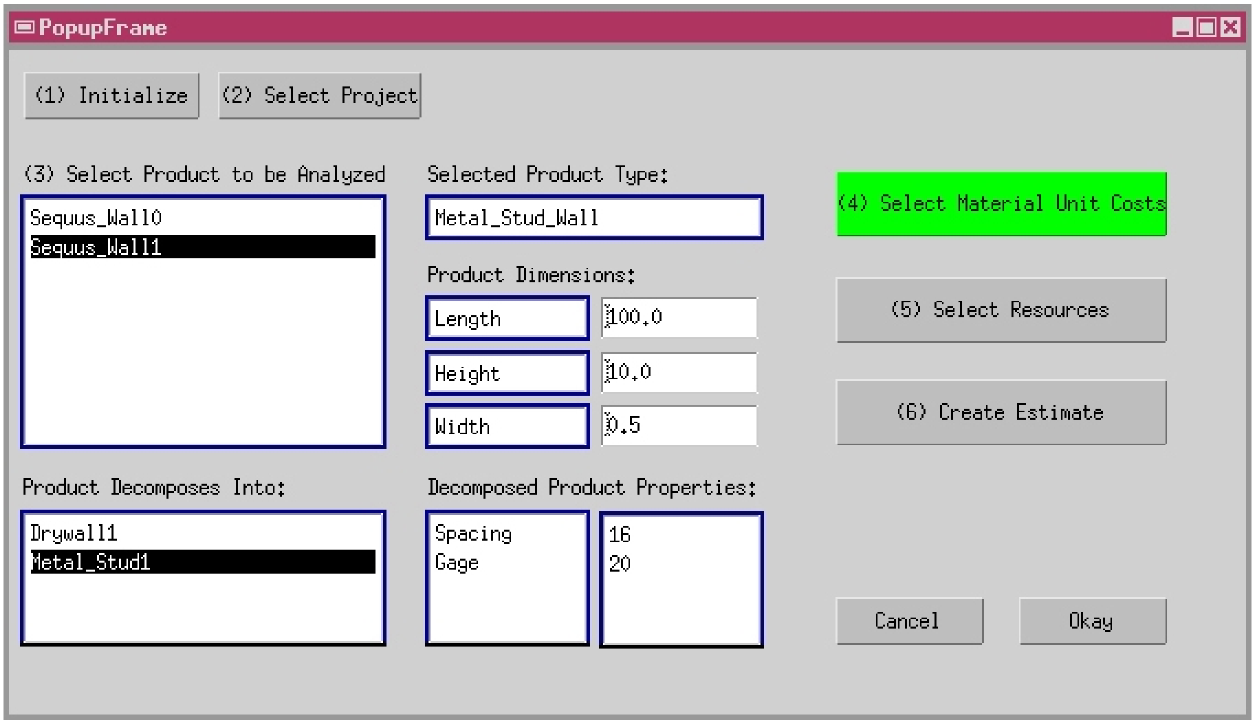
9.4 Initial Implementation of IFC's Product Representation

9.5 Initial Implementation of IFC's Resource Representation

9.6 Initial Implementation of IFC's Cost Representation

9.1 User Interface to Estimating System

User loads the design for which an estimate needs to be made, and the system imports the corresponding IFC-based product model. The product type, product decomposition and properties, and product dimensions are shown. Then the system selects the material unit cost and resources using the estimator's rationale for relating product and cost information. The system provides a list of the *appropriate* cost information and the user selects from the list.



(3) Select Product to be Analyzed
Sequus_Wall0
Sequus_Wall1

Selected Product Type:
Metal_Stud_Wall

Product Dimensions:

Length	100.0
Height	10.0
Width	0.5

Product Decomposes Into:
Drywall1
Metal_Stud1

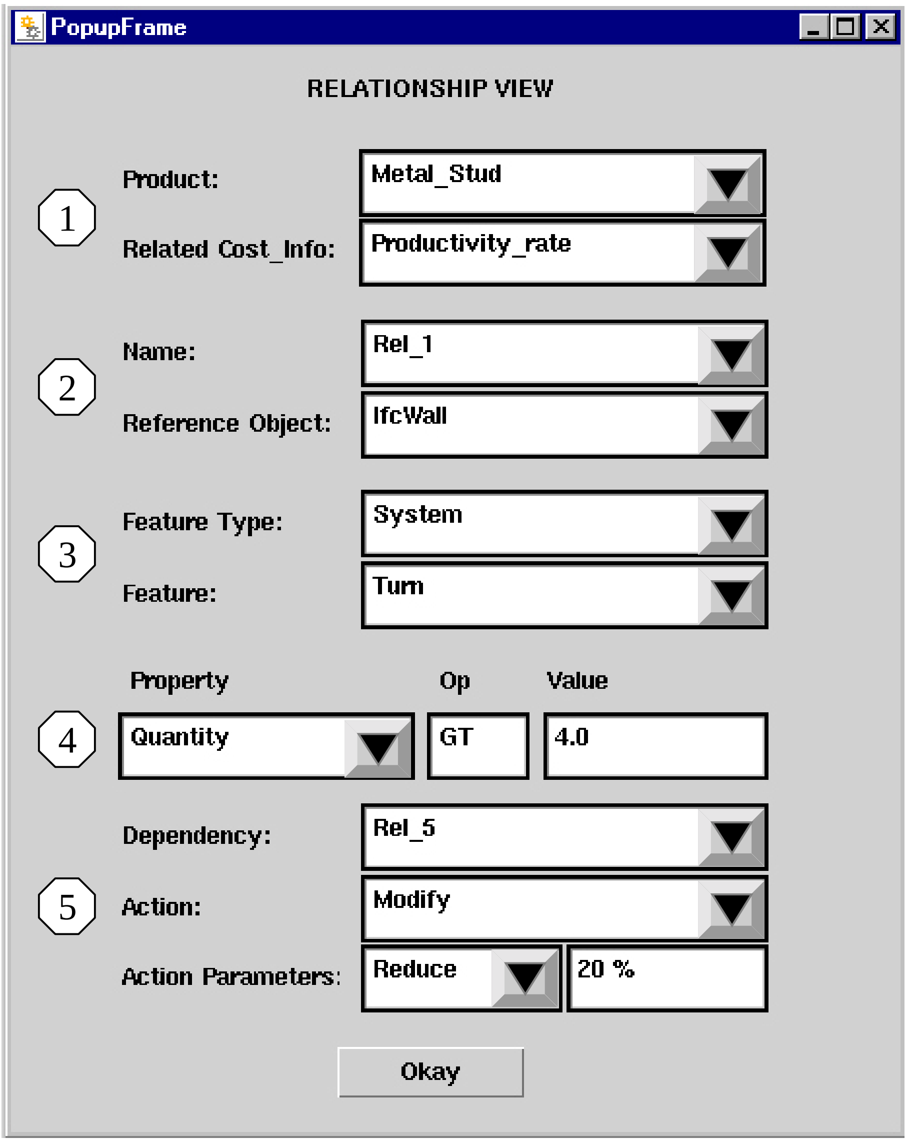
Decomposed Product Properties:

Spacing	16
Gage	20

9.2 User Interface to Relationship Template that Captures Estimators' Rationale

The relationship template provides a way for estimators' to easily enter their rationale for relating product and cost information (also shown in Figure 7). The relationship template only permits selections of properties of product features that are *logical*, or make sense, for the particular estimating situation.

1. The estimator identifies the product and related cost information that is captured in the relationship.
2. The estimator names the relationship and defines the reference object. The reference object allows the estimator to use a related object to analyze in order to determine the product feature. In this case, the "Metal_Stud" product doesn't *turn*, rather the "Wall" *turns*.
3. The estimator identifies the product feature and feature type. The system will help with the selection of this information because there are a limited number of features for a feature type and product.
4. The estimator selects the property and constraint on the property. Only appropriate properties will be available for a given feature. For example, the only properties available for the product feature turn are the "quantity" or "orientation."
5. The estimator creates dependent relationships and defines the appropriate action. For example, Rel_5 was identified as a dependent relationship because Rel_5 defines the dependency on the "length" of wall.



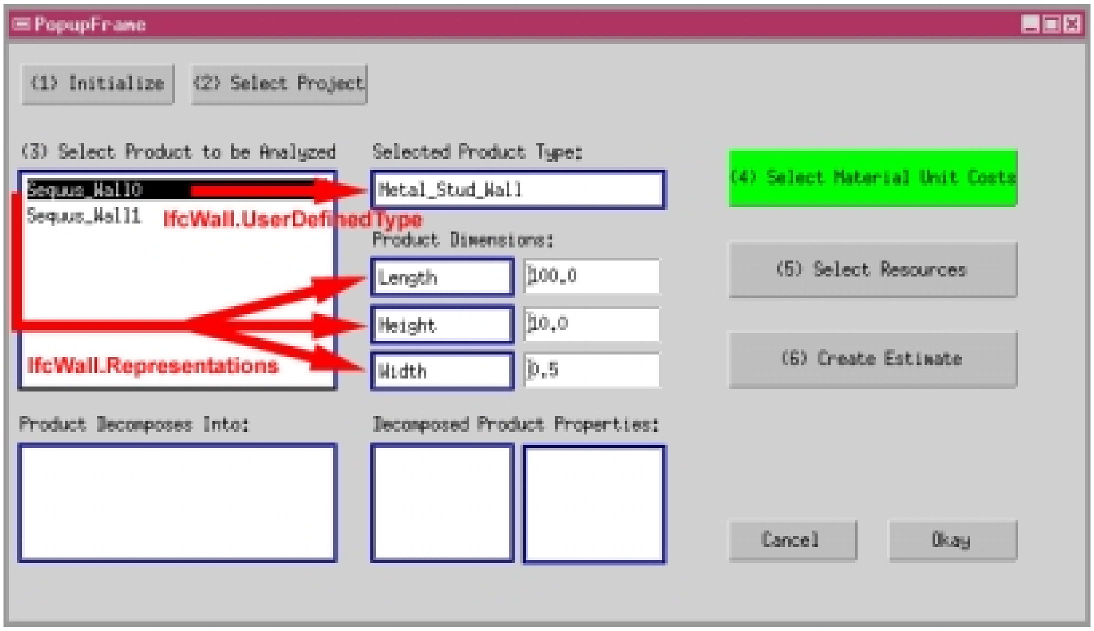
Product: Metal_Stud
Related Cost_Info: Productivity_rate
Name: Rel_1
Reference Object: IfcWall
Feature Type: System
Feature: Turn

Property	Op	Value
Quantity	GT	4.0

Dependency: Rel_5
Action: Modify
Action Parameters: Reduce 20 %

9.3 User Interface illustrating IFC's Representation of Products at Multiple Levels of Detail

Classes and attributes used to determine the product type, dimensions, decomposition of products, and the corresponding properties of the decomposed objects. These classes and attributes will be used to determine the level of design information available when creating cost estimates.



(3) Select Product to be Analyzed
Sequus_Wall0
Sequus_Wall1

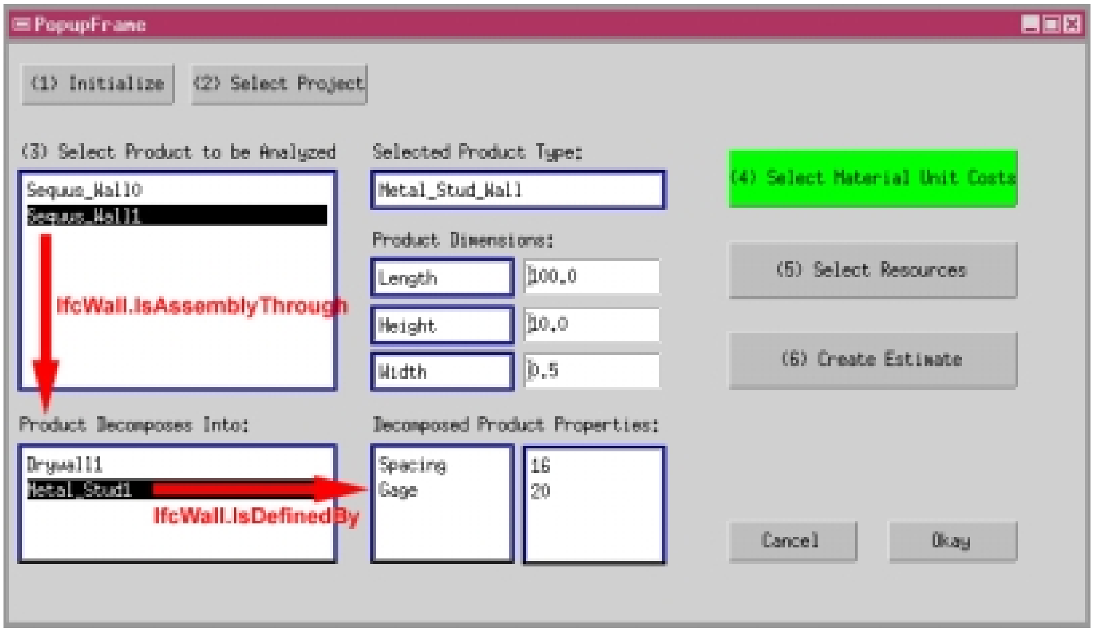
Selected Product Type: Metal_Stud_Wall

Product Dimensions:
Length 100.0
Height 10.0
Width 0.5

Product Decomposes Into:

Decomposed Product Properties:

Sequus_Wall0 shown at conceptual stage when the decomposition is unknown.



(3) Select Product to be Analyzed
Sequus_Wall0
Sequus_Wall1

Selected Product Type: Metal_Stud_Wall

Product Dimensions:
Length 100.0
Height 10.0
Width 0.5

Product Decomposes Into:
Drywall1
Metal_Stud1

Decomposed Product Properties:
Spacing 16
Gage 20

Sequus_Wall1 shown at detailed design stage when decomposition known.

9.4 Initial Implementation of IFC's Product Representation

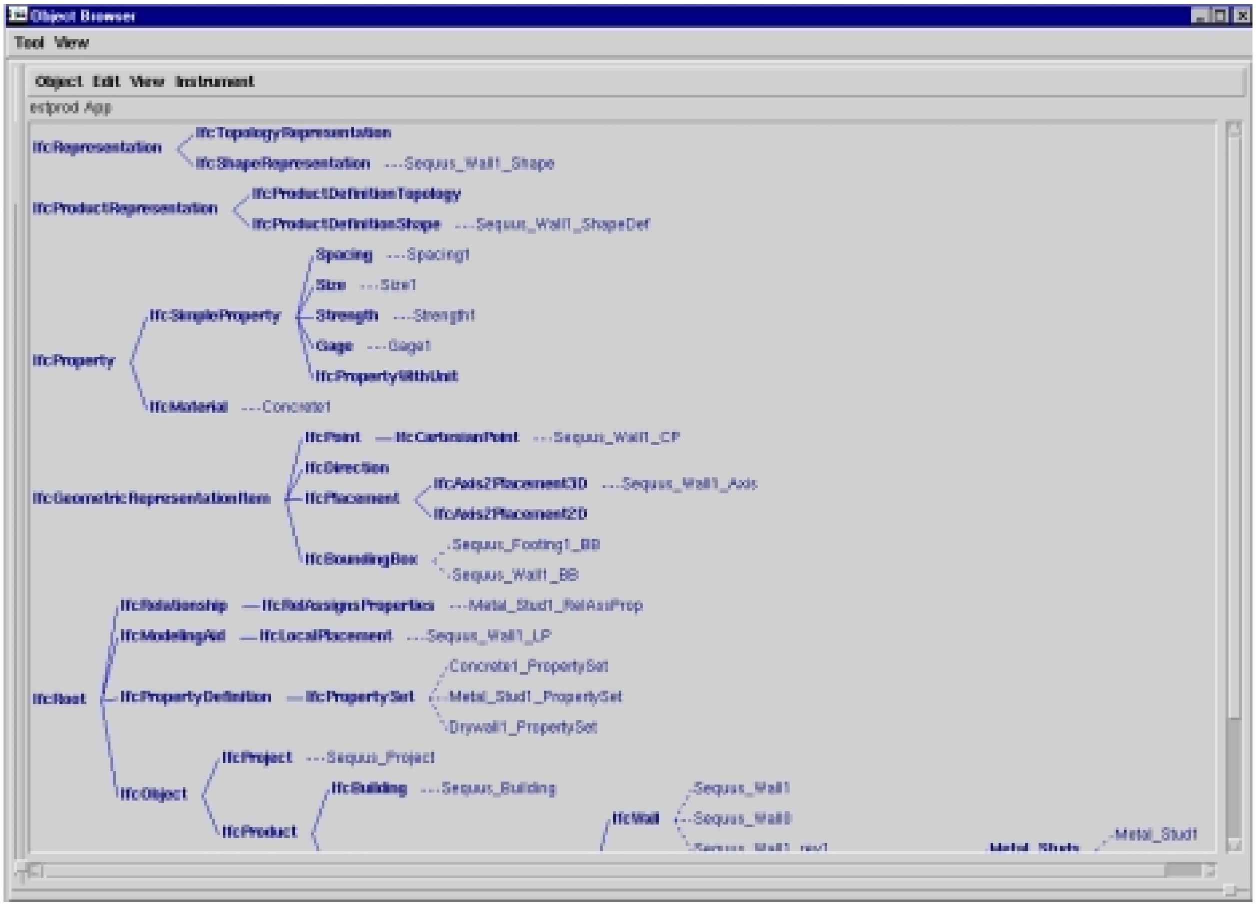
Classes and instances of IFC's product model hierarchy implemented for Sequus Pharmaceuticals Project test case.

The geometric representation of Sequus_Wall_1 can be found by following the attributes to the class IfcBounding Box:

Sequus_Wall1.Representations.ShapeRepresentations.Items

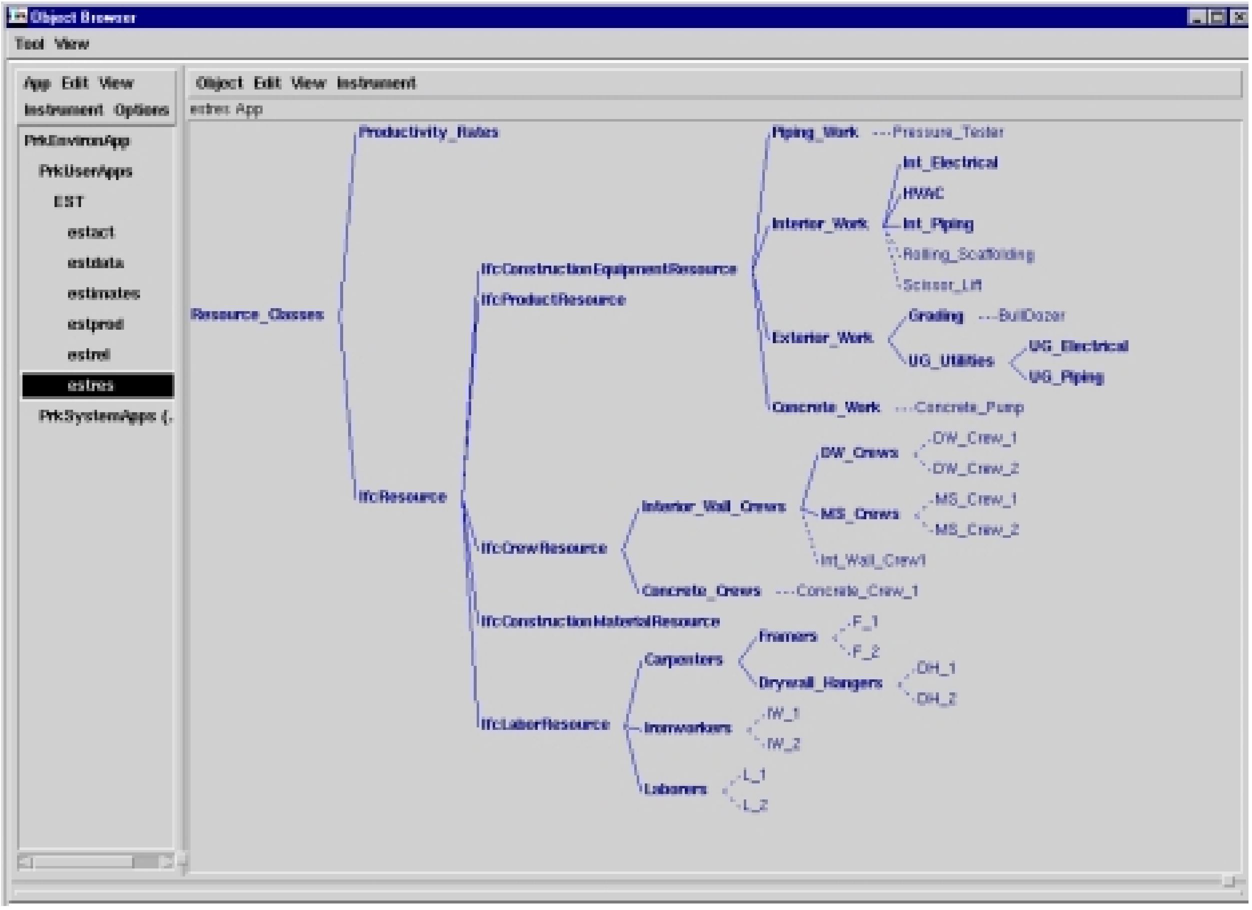
The placement of Sequus_Wall_1 can be found by following the following the attributes to the class IfcAxis2Placement3D:

Sequus_Wall1.LocalPlacement.RelativePlacement



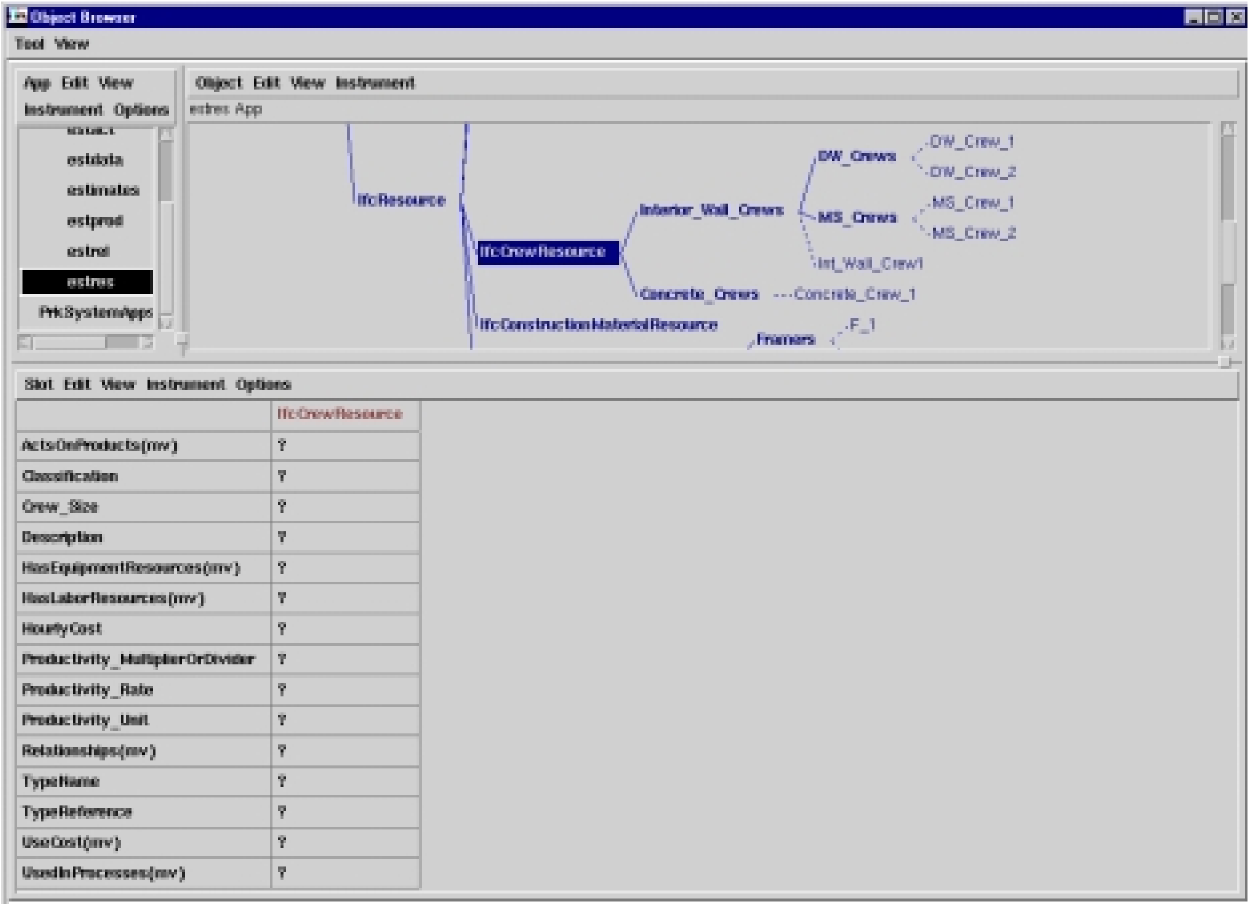
9.5 Initial Implementation of IFC's Resource Representation

The implementation of construction resources inlcuded classes defined by the IFC's, as shown below. Additional classes were added to provide a more detailed representation of resource information. The attributes implemented are shown on the next page.



9.5 Initial Implementation of IFC's Resource Representation (Continued)

Attributes of crews implemented:



	IfcCrewResource
ActsOnProducts(mv)	?
Classification	?
Crew_Size	?
Description	?
HasEquipmentResources(mv)	?
HasLaborResources(mv)	?
HourlyCost	?
Productivity_MultiplierOrDivider	?
Productivity_Rate	?
Productivity_Unit	?
Relationships(mv)	?
TypeName	?
TypeReference	?
UseCost(mv)	?
UsedInProcesses(mv)	?

9.6 Initial Implementation of IFC's Cost Representation

The implementation of costs included the following classes and attributes of the IFC's. Additional classes were added to provide a more detailed representation of cost information

	IfcCostElement
Calculate_Cost!	'?IfcCostElement.Calculate_Cost!
Cost_Container	?
CostSchedule	?
NestedBy(mv)	?
Nests(mv)	?
Product_Class	?
ProjectName	?
Quantity	?
Related_MaterialCost(mv)	?
Related_Product	?
Related_Resources(mv)	?
Total_Cost	0.0