COMPACT HIERARCHICAL PRODUCT AND PROCESS REPRESENTATION STRUCTURES FOR CONSTRUCTION PLANNING

Asad Udaipurwala[1] **and Alan D. Russell**[2]

ABSTRACT

This paper presents a set of complementary product and process representation structures that utilize a generalization of the mereological part-of relation along with the concept of spatial or work locations to enable an expert planner to express his knowledge in a compact semantically predefined manner that can later be utilized in a modular fashion to develop a project plan either manually or semi-automatically. Our approach reflects the observation that industry practitioners tend to think in more abstract terms, for example rather than reason about each and every column in a building, they group them by location, dimension and so on and reason on these aggregate groupings while making decisions regarding the construction methodology to adopt, the resource levels required, estimating activity durations, etc. We contrast our approach to other research work to develop computer-interpretable product and process models which adopt a more fine-grained representation of a project.

KEY WORDS

knowledge representation, product models, process models, part-whole relationship

INTRODUCTION

This paper presents a set of complementary product and process representation structures that utilize a generalization of the mereological part-of relation along with the concept of spatial or work locations to enable an expert planner to express his knowledge in a compact semantically predefined manner that can later be utilized in a modular fashion to develop a project plan either manually or semi-automatically.

We start the paper by explaining the product and process models. The next section describes the kinds of knowledge representation and reasoning that can be performed using these structures by casting them within the framework of the five distinct roles they can play as identified by Davis et al. (Davis et al. 1993). Next we compare and contrast our structures to IFC and show that we can easily map most of the constructs from our structures to that of IFC thus enabling us to achieve data exchange. As well, we indicate some of the limitations

[1] Ph.D. Candidate and Graduate Research Assistant, Department of Civil Engineering, University of British Columbia, Vancouver, Canada; E-mail: hussain@interchange.ubc.ca

[2] Professor and Chair, Computer Integrated Design and Construction, University of British Columbia, Vancouver, Canada; E-mail: adr@civil.ubc.ca

of a purely object-oriented approach similar to IFC. We end with an example of the use of these structures for construction Methods Feasibility Analysis.

PCBS AND M&RBS – COMPONENTS AND THEIR COMPOSITION

PHYSICAL COMPONENT BREAKDOWN STRUCTURE

The PCBS is primarily a decomposition hierarchy that models the successive breakdown of an overall construction project into its constituent parts. It consists of two distinct sub-trees rooted at a single Project component. The first sub-tree – which we will designate the physical hierarchy for ease of reference – consists of the following components: SubProject, System, SubSystem, Element, SubElement, SubSubElement, Content and Material in that topological order. Thus a Project breaks down into SubProjects that further break down into Systems and so on. This allows the user to communicate his understanding of a facility's physical configuration regardless of the particular spatial location of each component. For example, a dam can have multiple spillways (main spillway, emergency spillway) which can be broken down as Dam (Project component) → Spillways (Element). The second sub-tree – the location hierarchy – contains the Location Set, Location and SubLocation components in that topological order. Locations are available work faces which might be physical spaces, that is those where components from the first sub-tree might be located (for example, right embankment of a dam which is the location of the emergency spillway); process steps (for example, the fabrication stage in a procurement process) or global (for example, project site). The assignment of a component in the physical hierarchy to a Location in the location hierarchy is achieved through attributes (see below). This basic organization of the PCBS can be modified to tailor it to the needs of a particular project or planning task by either truncating the tree or skipping intermediate levels. Here are a few examples:

1. The simplest example is the case where a Project does not consist of any SubProjects – A single high-rise condominium project as opposed to a multi-building project. In this case, one simply omits the SubProject level and places the System component directly under the Project component. Depending on user viewpoint, all or only a subset of a building's systems may be modeled.

2. The planner might choose to plan/reason at an aggregated or detailed level. For example, in planning the prefabrication of pipe modules the planner might decide that simply tracking productivity for each type of pipe module is sufficient, or on the other hand he might decide to model each component of the pipe module separately. In the former case, the PCBS only consists of each pipe module (Element component) directly under the Project component. The latter case can be modeled as a Project component divided into System component for each pipe module type further broken down into Elements for pipes, insulation, welding and so on.

3. Instead of modeling the individual units on every floor of an apartment complex (SubLocation component) the planner might choose to restrict the modeling to the floors (Location component).

To perform reasoning or other useful management functions such as productivity analysis we need information on the characteristics of these components such as their dimensions, numbers, relative location in the facility, working space available and a lot more. To capture these characteristics every component in the PCBS can be described using a set of user-defined attributes. Attributes defined for a parent component can be inherited by its children. This not only facilitates data-entry, but also provides for an efficient means of computing an aggregate value for the parent component's attribute based on the value of inherited attributes of the children[3]. The attributes can have quantitative, boolean or linguistic values. Only quantitative values can be aggregated. Quantitative values can optionally have an associated unit of measure which may be a singular unit such as meters, or compound such as kg/m^2. Values for an attribute of a component are assigned on a per location basis, and at present cannot be assigned on a SubLocation basis. For example, component C has a value V at a location L; L in this case coming from the location hierarchy. This serves as a compact way of saying that component C exists at location L and for that instance of the component its value is V. Attributes of components of type Location can only have a self-located value.

Knowledge management and reuse is afforded by an ability to abstract fragments of the physical hierarchy as Templates and catalog them in a central library called Standard PCBS (as opposed to the project-specific variety above known as the Project PCBS). A Template can be thought of as a named branch of the physical hierarchy starting at Project, System, Element, Content or Material level. Components in the Standards have similar representation schema as their Project counterparts except that the attribute values are not assigned on a location basis and although the attributes can be inherited their values cannot be aggregated. These Templates can then be composed together to form other Templates or to form a Project PCBS. In doing so, a Template of a particular type can be inserted as a subtype as long as the PCBS syntax is not violated. When Standard components are used to form a Project PCBS the values assigned to any attributes are retained and made available to the user for reference.

METHOD AND RESOURCE BREAKDOWN STRUCTURE

As opposed to the PCBS, the M&RBS is best thought of as goal-directed tree with the root component Method Statement providing the goal, for example, Construction of Typical Floor with Central Core. To attain this goal one needs to perform a number of Operations (or sub-goals), for example, Build Columns and Walls, Build Slabs, etc. There is a set of construction Methods which can be used to achieve each of these new objectives, for example, Gang forming, and Pouring concrete using crane and bucket. Each of these Methods involves the use of one or more Resources, for example, formwork, carpenters, and a crane for Gang forming. Thus an M&RBS contains Method Statement, Operation, Method and Resource components in that topological order. The M&RBS for a project can be composed of multiple Method Statements. Similar to the PCBS, it is possible to capture the characteristics of each component by assigning it a set of user-defined Parameters, which can again have

---

[3] This notion of inheritance of attributes is conceptually very similar to that of OO models. However, in the latter, inheritance is only supported from super-class to sub-class. In PCBS we are not defining a child component type as a sub-class of the parent. In fact, this is not possible given the morphability of the PCBS. Also, implementing inheritance of dynamically defined attributes in an OO model is cumbersome.

quantitative, boolean or linguistic values. Parameters refer to inherent properties of a component such as production rate, bucket capacity, etc. Parameters of a parent component can be inherited by its children. However, aggregation of values of inherited Parameters is not allowed. For example, production rate of a Method is not necessarily equal to the sum of the production rates of its Resources. In addition, for each component one can also assign a set of Conditions which are criteria that if left unsatisfied would negate the feasibility of the component. Conditions also provide for values to be specified in quantitative, boolean or linguistic terms but cannot be inherited. For the Method component one can also specify a list of partially ordered tasks called a Fragnet that the Resources must perform to use it for an Operation. Fragnets can nest other Fragnets. A Fragnet for the Operation is then automatically derived from its constituent Methods. This derived Fragnet can be further refined by the user, for example, by introducing additional tasks and/or logic relationships.

The M&RBS also provides for knowledge management capability in the form of Standard M&RBS. Companies can garner knowledge about existing construction strategies, and newly introduced methods and resources from various sources and catalog these for future reference in classification hierarchies for Resources (Resource Class → Resource SubClasses → Resources) and Methods (Method Class → Methods). These can then be used to compose Method Statements either as Templates for use on future projects or directly for a particular project. If Methods or Resources from one of these classifications is used in forming Method Statements, a reference to the original source classification item is always maintained so that one can track changes or developments. Similar to the PCBS, the values of Parameters and Conditions of Standard M&RBS components are assigned in a nonspecific manner, and can then be given more appropriate values on a per-location basis when used in a project setting, though unlike PCBS this would be more an exception then the norm. Another knowledge management capability in the M&RBS is the ability to assign URIs to Methods and Resources so that additional information about them can be obtained right from the source such as supplier's websites, standardized corporate testing procedures, etc. Finally Fragnet information can also be stored in a Standard Fragnet library. These can then be associated with Method components or used directly as sub-networks in project schedules.

SEMANTICS

The rules for formulating the PCBS and M&RBS enunciated in the previous section are best applied when the user has a firm understanding of the roles these representation structures are designed to serve. Davis et al. (Davis et al. 1993) have identified five distinct roles for a knowledge representation which we will use as a framework for the PCBS and M&RBS.

Role I: A KR is a Surrogate – The PCBS is a surrogate computational representation for a non-existing facility. Clearly any model of a physical designed facility will have a lower fidelity than the actual structure – that is it is an imperfect surrogate (Davis et al. 1993). In fact, in the case of the PCBS this is mostly a desirable trait to enable one to perform computational tasks using it with some acceptable efficiency (Bingi at al. 1995). One example of the efficiency afforded by our representation is the ability to leverage the order in which the Locations are defined in Hierarchical Scheduling (Russell et al. 2003); for example, in a residential building Location "1st Floor" would precede Location "2nd Floor".

However, while analyzing Methods Feasibility it cannot necessarily be interpreted that subsequent Locations adjacent to each other; for example, in a bridge project one might group all piers together followed by the spans. In addition to adjacency, Methods Feasibility Reasoning would benefit greatly from a mechanism to represent the relationship among Locations; for example, in a residential building a vertical elevator shaft (ELEVATOR Location component) passes through every floor. It is extremely difficult to allow specification of such relationships (in a general way) in the PCBS without sacrificing the intuitive nature of the breakdown (Bingi at al. 1995)[4].

Role II: A KR is a Set of Ontological Commitments – The ontological basis for PCBS and some of M&RBS is Mereology[5]. Methods Feasibility Reasoning makes explicit use of these commitments in performing preliminary semantic checks. Table 1 lists the meronymic relationships at various levels in the PCBS and M&RBS.

Table 1 PCBS/M&RBS Components best suited for particular Meronymic Relationships

Whole – Part Relationship	**PCBS/M&RBS level**
Collection – Elements[A]	Location Set – Location; Location – SubLocations
Collection – Elements	Element – SubElement; SubElement – SubSubElements[B]
Complex – Components	Project – SubProjects; Project – Systems; Project – Elements; SubProject – Systems; SubProject – Elements; System – SubSystems; System – Elements; SubSystem – Elements; Element – SubElements; SubElement – SubSubElements
Object – Stuff[C]	Element – Content; SubElement – Content; SubSubElement – Content; Element – Material; SubElement – Material; SubSubElement – Material
Mass – Quantities	Content – Material
Feature – Activity	Method Statement – Operation

[A] (Gerstl and Pribbenow 1996)
[B] A TypeOf relationship can be modeled as a Collection – Elements relationship under certain interpretations For example, Column (Element) with Rectangular and Circular Column (SubElement) sub-components.
[C] (Artale et al. 1996)

Role III: A KR is a Fragmentary Theory of Intelligent Reasoning – As noted, the order of Locations in the PCBS influences a planner's reasoning process during Scheduling. Similarly, the M&RBS facilitates Methods Feasibility Reasoning as follows:

1. The breakdown structure of the M&RBS can be used to guide the reasoning process in a natural bottom-up order. That is, for a Method to be feasible, all its key Resources have to be feasible for the Operation under consideration. For a Method Statement to be feasible all its Methods need to be feasible. Thus if a Resource is infeasible, one can simply terminate the reasoning process if so desired. This decreases the time needed to develop a feasible construction strategy while promoting compositionality.

[4] In our opinion, other than Predicate Logic, there is no other representation scheme capable of generally modeling such relationships.

[5] Mereology – The formal study of the relations between parts and wholes. [Oxford English Dictionary]

2. All other Method Selection approaches we have found in the literature concentrate on the selection of individual methods. However, methods might be feasible in themselves but incompatible with each other. The Method Statement captures an expert's knowledge of the compatibility of various methods in reaching a common objective.
3. The aggregation of Method Fragnets into an Operation Fragnet provides an avenue for aggregate reasoning such that if one simply desires to know whether a set of Methods fulfills a time constraint, planned productivity rate, etc. it can be determined from the start and finish times of the activity corresponding to the Operation.

Role IV: A KR is a Medium for Efficient Computation – As noted previously, the Project PCBS provides an efficient means to calculate aggregate Values for Attributes of parent components from those of their children. A similar calculation in an object-based system would involve for every attribute of a parent component at a particular location, enumerating all its children, iterating through the child's Attributes, identifying the right attribute, collecting its value and assigning it to the parent Attribute.

Role V: A KR is a Medium of Human Expression – The PCBS and M&RBS are designed to be very modular (Bingi at al. 1995). As such they do not always require an expert to formulate. The intuitive decomposition structure can be used by a novice to formulate a PCBS and M&RBS for a project directly, with help partially/fully from Templates developed by experts. Referring to the Conditions associated with a Method or Resource one can determine its feasibility manually. At the same time experts have the flexibility of directly assembling an innovative construction strategy by pulling Methods and Resources directly from the Standard Classes. The PCBS and M&RBS themselves can be formulated independently so that if one wishes to formulate only a physical model of the facility for planning purposes one can do so. Alternatively, one can have a minimal PCBS but develop a detailed construction strategy using Method Statements from the Standards.

COMPARISON WITH OTHER DATA MODELS

There have been a variety of schemes used by researchers to represent a facility's physical configuration and construction method knowledge such as rule-based systems, semantic nets, flat lists (text or spreadsheets), neural nets (the net represents knowledge in some manner, even though it is not explicitly known), hierarchical structures, object-oriented systems, CAD systems, and others. In this section, we will compare the PCBS and M&RBS with IFC from IAI, as these represent a consensus opinion on modeling data. Note that IAI does not claim that IFC are a means of knowledge representation but positions it as a data-exchange standard for exchange of data among software applications (International Alliance for Interoperability 2000, pp.10). However, in practice a number of research efforts have chosen to use the IFC as a database to support their applications. In addition, as noted by Sowa (Sowa 1981), both databases and artificial intelligence systems represent and process knowledge about the real world. The primary difference between the two fields lies in the volume of data that they process and the complexity of the representations.

Table 2 presents a listing of some of the key PCBS and M&RBS constructs that can be mapped onto a corresponding IFC construct along with relevant comments for each mapping.

The PCBS and M&RBS are designed to be used across all construction domains, while the IFC support only the AEC/FM domain. This is reflected in some of the comments. Space considerations prohibit us from including a complete mapping. Table 3 lists key PCBS and M&RBS constructs that do not have a counterpart in IFC. Since the IFC are designed for use by a number of disciplines such as architects, specialty consultants, contractors, facility managers, etc. as opposed to the PCBS and M&RBS which are primarily for use for and during the construction phase a reverse mapping has not been developed as the number of IFC constructs not having any counterpart in the PCBS and M&RBS would be considerable.

Table 2 PCBS/M&RBS – IFC Mapping[A]

PCBS/M&RBS Concept	IFC Mapping
Physical Component Breakdown Structure (PCBS)	
Component (Project, SubProject)	IfcProject[B]
Project – SubProject Relationship	IfcRelNests[C]
Component (A) (System, SubSystem, Element, SubElement, SubSubElement)	IfcBuildingElementProxy
Project – System Relationship SubProject – System Relationship Project –Element Relationship SubProject – Element Relationship All valid Component Relationships involving components of (A) above.	IfcRelAggregates[D]
Component (Location Set)	IfcSpace[E] (CompositionType: COMPLEX)
Component (Location)	IfcSpace[F] (CompositionType: ELEMENT)
Component (SubLocation)	IfcSpace (CompositionType: PARTIAL)
Location Set – Location Relationship	IfcRelAggregates
Location – SubLocation Relationship	IfcRelAggregates
Project – Location Set Relationship SubProject – Location Set Relationship	IfcRelAggregates
Component (B) (Content, Material)	IfcConstructionMaterialResource[G]
Content – Material Relationship	IfcRelNests
All valid relationships between components of group (A) and group (B)	IfcRelAssociates
Component type	IfcTypeObject[H]
Method & Resource Breakdown Structure (M&RBS)	
Component (C) (Method Statement, Operation, Method)	IfcTask[I]

All valid Component Relationships involving components of (C) above.	IfcRelNests
Component (Resource)	IfcCrewResource[J]
Operation – Resource Relationship Method – Resource Relationship	IfcRelAssignsToResource
Parameter/Condition	IfcComplexProperty with a constituent IfcPropertyEnumeratedValue indicating whether this is a parameter or condition
Parameter/Condition Class	IfcClassificationItem
Parameter/Condition – Parameter/Condition Class Relationship	An IfcPropertyReferenceValue member of Parameter/Condition IfcComplexProperty

[A] For all components based on IfcObject, component code maps to Name and component description maps to Description.

[B] Some of the required attributes of IfcProject are not relevant in our context - specifically RepresentationContexts and UnitsInContext. These can be set to any valid value. The use of IfcProject for the SubProject component will cause the IFCSINGLEPROJECTINSTANCE rule to fail. However, this is not fatal. The reason why IFC fail to allow such a relationship as PCBS is not clear.

[C] part-of relationship.

[D] part-of relationship. Semantically this relationship is closer to that present in the PCBS, although one could use IfcRelNests.

[E] Alternatively one can use IfcGroup.

[F] Here one might want to tweak the exported data so that certain location components such as “SITE” for example are exported as IfcSite.

[G] Ideally one might want to use IfcMaterial instead of IfcConstructionMaterialResource. However, IfcMaterial is defined as a homogeneous substance which might not always be the case with PCBS Content. For example, “Reinforced Masonry”

[H] This would be done by defining an IfcPropertyEnumeration for all the component types and then associating an IfcPropertySet with an IfcPropertyEnumeratedValue with this IfcTypeObject. IfcRelDefinesByType will used to associate the IfcTypeObject with the relevant object.

[I] This is a non-ideal mapping, as it provides a very narrow semantic view of the corresponding M&RBS component, but it is the only non-abstract class available in the current IFC specification suitable for the purpose. One could more suitably utilize IfcProcess if it were not abstract.

[J] Again, this is a non-ideal mapping for the same reason as for components of (C) above.

Table 3 PCBS/M&RBS Concepts that cannot be mapped to IFC

REPCON Concept	Mapping Notes
Physical Component Breakdown Structure (PCBS)	
Attribute Inheritance	IFC currently do not support the notion of property set inheritance, i.e., there exists no mechanism to define a set of properties without values that can be assigned to multiple objects and then assigned values particular to that object. A workaround that utilizes only existing IFC entities is possible, but it would require user to refer documentation external to the Standard.
Project PCBS Values (Planned and Actual)	Planned and Actual project properties in the PCBS are defined on a per location basis. IfcPropertyReferenceValue cannot have a reference to an IfcSpace object. This would have permitted the use of an IfcComplexProperty

	to define planned and actual project PCBS values. The alternative provided by the IFC is to use an IfcRelContainedInSpatialStructure objectified relationship. However, relationships cannot have property sets associated with them.
Attribute Value Aggregation	Since attribute inheritance and project PCBS values cannot be modeled in IFC there is no possibility of aggregating the value of inherited attributes for a parent. Even if one could model these, the mechanism of IfcPropertyDependencyRelationship as suggested in the IFC specification would be inadequate as it allows only a one-to-one dependency relationship between properties. PCBS attribute value aggregation requires a one-to-many relationship. A possible workaround using IfcGroup is also not possible as these entities are not permitted to be the value of an IfcPropertyReferenceValue.
Method & Resource Breakdown Structure (M&RBS)	
Parameter Inheritance	This is not possible for the same reason as the PCBS.

RULE-BASED METHODS FEASIBILITY REASONING

In this section, we illustrate some of the advantages afforded by the PCBS and M&RBS design in formulating Methods Feasibility Rules. In addition to the following, the PCBS and M&RBS design enables easy consistency checking, composition of rulesets from existing Methods and formulation of an initial project schedule from a feasible Methods Statement. These aspects are covered in more detail in Udaipurwala and Russell (2003).

A sample feasibility rule is expressed in Figure using the CLIPS syntax (Riley 2003). The experts in this case are expressing their best judgment on the minimum number of reuses that would be required for the method – forming using Flying Truss Forms – to be economical along with the variation in length and width it can handle without regard to a specific configuration for the building.

Parts 1 and 2 illustrate how the hierarchical nature of the PCBS and M&RBS enables one to restrict the context in which the rule applies. That is for the case of forming superstructure SlabBays only. Part 3 shows the ease with which all the instances of a particular SlabBay type can be retrieved. This is primarily because of the compact representation of attribute values we adopt (that is Attributes → Location → Values). In part 4 we retrieve the minimum number of reuses required for our forms. Part 5 then collects the values for Attributes Width and Length into lists for further reasoning. Part 6 then determines how many reuses are potentially available within the allowable variations in Width and Length that the Flying Truss Forms can accommodate. This is only possible because we know the Location of existence of each of the SlabBays and thus can reason about the length and breadth of a SlabBay at a Location in unison.

SUMMARY AND CONTRIBUTIONS

This paper described the knowledge representation capabilities of two breakdown structures – PCBS and M&RBS. The PCBS models the meronymic (whole - part) structure of a project in a modular, concise, computer-interpretable structure (Bingi et al. 1995) that provides a great deal of flexibility to enable the user to adapt it to different construction domains and planning requirements. The M&RBS enables organizational learning by providing a

```
(defrule ROOT.FormSlab.FlTruss::rule2
  (ancestor (ancestor_component ?ancestor1) (descendant_component ?descendant1))        1
             (pcbs_component (name ?ancestor1) (code ?code1&:(eq "SupSTR" ?code1)))

  (ancestor (ancestor_component ?ancestor)   (descendant_component ?descendant))        2
  (mrbs_component (name ?ancestor)   (code ?code3&:(eq "FormSlab" ?code3)))

  (pcbs_component (name ?descendant1) (code ?code2)
    (description ?desc1)
    (attributes $?ah1 "Width" $?)                                                       3
    (attribute_values $?avh1&:
      (eq (length$ $?ah1)(length $?avh1)) ?val2 $?))

  (mrbs_component (name ?descendant) (code ?code4&:(eq "FlTruss" ?code4))
    (description ?desc2)
    (attributes $?ah3 "Min. Reuse Required" $?)                                         4
    (attribute_values $?avh3&:
      (eq (length$ $?ah3)(length $?avh3)) ?val4 $?))

  =>

  (bind ?a1 (send ?val4 get-value1))

  (bind ?clist1 (create$))
  (bind ?llist (send ?val1 get-location_list))
  (bind ?avlist (send ?val1 get-attribute_value_list))
  (progn$ (?field ?llist)
    (bind ?clist1 (insert$ (create$) ?field-index ?clist1 ?field                        5
      (send (nth$ ?field-index ?avlist) get-value1))))

  (bind ?clist2 (create$))
  (bind ?llist (send ?val2 get-location_list))
  (bind ?avlist (send ?val2 get-attribute_value_list))
  (progn$ (?field ?llist)
    (bind ?clist2 (insert$ (create$) ?field-index ?clist2 ?field
      (send (nth$ ?field-index ?avlist) get-value1))))

  (bind ?a2 (common$ (get-largest-cluster (cluster_data ?clist1 10))
      (get-largest-cluster (cluster_data ?clist2 10))))

  (if (not(>= ?a2 ?a1))
    then (printout t "The Method: " ?desc2 " for PCBS component " ?desc1                6
      " is infeasible as the number of reuses: " ?a2
      ", for the formwork is not economically feasible." t)
    else (printout t "The Method: " ?desc2 " for PCBS component " ?desc1
      " affords sufficient reuses: " ?a2 ", for the formwork." t)))
```

Figure 1: Rule for Assessing Feasibility of Flying Truss Formwork System

catalogued store of construction methods and resources and the capability to utilize these in formulating construction strategies for new projects. The discussion included the syntax and detailed review of the semantic content of the knowledge that can be encoded in these structures and subsequently used by the system. We have also documented the roles that these structures play and their semantic strengths and weaknesses. By providing a detailed comparison to IFC from IAI, we have hopefully provided some avenues for further enhancements to the capabilities of these Standards.

Two lines of inquiry are being pursued for future research. One involves integrating these structures with other project data such as risks, records, organizational information, etc. thereby making the system more aware of the physical and process environment while analyzing risks, retrieving relevant records for claims, assessing performance of a project participant and so on. The second is to enrich these structures further by making them shareable across an enterprise.

REFERENCES

Artale, A., Franconi, E., Guarino, N., and Pazzi, L. (1996). "Part-whole relations in object-centered systems: An overview" *Data and Knowledge Engineering*, 20, 347-383.

Bingi, R., Khazanchi, D., and Yadav, S.B. (1995). "A Framework for the Comparative Analysis and Evaluation of Knowledge Representation Schemes" *Information Processing & Management*, 31 (2), 233-247.

Davis, R., Shrobe, H., and Szolovits, P. (1993). "What is a Knowledge Representation?" *AI Magazine*, 14 (1), 17-33.

Gerstl, P., and Pribbenow, S. (1996). "A conceptual theory of part-whole relations and its applications" *Data & Knowledge Engineering*, 20, 305-322.

International Alliance for Interoperability (2000). "IFC Technical Guide" (available at http://www.iai-international.org/Model/documentation/IFC_2x_Technical_Guide.pdf).

Riley, G. (2003). "CLIPS: A Tool for Building Expert Systems." (available at http://www.ghg.net/clips/CLIPS.html).

Russell A.D., Udaipurwala A.H. and Wong, W. (2003). "A Generalized Paradigm for Planning and Scheduling" *Proc. of 2003 Construction Research Congress*, Honolulu, HI.

Sowa, J.F. (1981). "A Conceptual Schema for Knowledge-Based Systems" *Proc. of the 1980 workshop on Data abstraction, databases and conceptual modeling*, Pingree Park, CO, 193-195.

Udaipurwala, A.H. and Russell, A.D. (2003). "Construction Methods Feasibility Reasoning in an Integrated Environment" *Proc. of 2003 Construction Research Congress*, Honolulu, HI.