

# Interleaving Semantics: a Filter Mediated Communication Model to Support Collaboration in Multi-Disciplinary Design

Yongwook Jeong<sup>1</sup>, Armando Trento<sup>2</sup>

<sup>1</sup>Department of Architecture, University of California at Berkeley - USA, <sup>2</sup>Department of Architettura e Urbanistica per l'Ingegneria, University of Rome, La Sapienza - Italy

<sup>1</sup>ywjeong@berkeley.edu, <sup>2</sup>armando.trento@uniroma1.it

**Abstract.** *The purpose of collaboration is to integrate the separate knowledge possessed by the participants in the design process into one meaningful whole. Centralized data structures, showed that the AEC industry is so fragmented that the data-centric approach is not feasible, for technical and procedural reasons. The shared database, minimizing the complexity of translating different form of representation, quickly becomes too large and unwieldy to support the dynamic process of multi-disciplinary collaborative design. In this paper, we propose a distributed model that includes a mechanism to facilitate the participants' intentions more effectively by incorporating semantics into their representations. We also present how the semantics would be authored and published through the mechanism so that a higher level of shared understanding among the participants would be achieved.*

**Keywords:** *Multidisciplinary Collaboration; Semantic Network; Ontology; Artificial Agent.*

## Introduction

The legacy of design as problem solving has been to consider collaboration a problem of effective communication where massive amounts of data must be shared among heterogeneous participants. Therefore, achieving interoperability among different CAD systems by way of organizing efficient databases has been the core research issue. The initial effort started with standardizing product descriptions including geometric information and constructing databases

to organize them. Following the standardization of product model data, (Eastman, et al.1991) proposed Engineering Data Model (EDM) to manage heterogeneous information carried by different design and engineering applications.

The underlying theoretical assumption of these efforts is that a building is a product composed of the heterogeneous products. This assumption has been relatively valid and even successfully realized in several related industries, such as the automotive and shipbuilding manufacturing industries. However,

the building and construction industry continues to lag behind in this development (Tolman, F. P. 1999) In the following sections, we will examine the Building Product Model (e.g., ISO-STEP, IAI-IFC) and its problems, and discuss the Filter Mediated Communication Model as an alternative approach.

## **Building Information Modelling**

### **Building Product Models**

In the 1970s, the National Institute of Standards and Technology (NIST) proposed a standard for the exchange of massive geometric data, called the Initial Graphics Exchange Standard (IGES), which defined a neutral data format as the lowest common denominator among CAD systems that use it. While IGES has been developed and maintained by a governmental agency, Data eXchange Format (DXF) has been developed by Autodesk in order to meet customers' needs. However, as CAD systems grew more diverse and sophisticated, this approach soon revealed its limitation. The most notorious problem is that when one application's data is translated into one of the neutral file formats, the translated data is no longer consistent with the original data because it loses the semantic information that was relevant to the application.

The complexity of the file formats is another problem. Both IGES and DXF require solid professional programming skills to understand and manage. These weaknesses were not easily corrected so that searching for a new method was accelerated. With this motivation, the International Standards Organization (ISO) developed an international standard in 1984, known as the Standard for the Exchange of Product Model Data (STEP) for computer-based description and exchange of physical and functional characteristics of products throughout their life cycle, independent of any particular system. A product model is an information model that implicitly contains data regarding form and function of a product and aims at describing the targeted product through

its life cycle. A building product model is an example of a product model in the AEC industry, which can describe the form (e.g., geometric information and its relationships) and function (e.g., energy performance) of a building through its life cycle. Recently, these efforts were reincarnated as Building Information Modeling (BIM) driven by several CAD system vendors (Autodesk, GraphiSoft, Bentley, etc.).

### **Industry Foundation Classes**

The newest and largest effort is collectively known as the Industry Foundation Classes (IFCs), published by the International Alliance for Interoperability (IAI) in 1995. The IFC is an open and non-proprietary data model specification in the AEC industry representing a fixed set of objects commonly used for the built environment. IFCs are used by computer applications (not intended for humans) to assemble a computer processable model of the facility that contains all the information of the parts and their relationships to be shared among project participants.

The philosophy of the IFCs is to electronically represent all possible aspects of a building including products (e.g., doors, walls, fans, etc.) and abstract concepts (e.g., space, organization, process etc.). These specifications represent a data structure supporting an electronic project model useful in sharing data across applications. Each specification is called a 'class.' The word 'class' is used to describe a range of things that have common characteristics. For instance, every door has the characteristic of an opening to allow entry to a space; every window has the characteristic of transparency so that it can be seen through. Door and window are names of classes.

### **Problems and Limitations of BIM**

For the past few decades, it appeared that such product models could serve the purpose of collaborative design. Eastman, C., Jeng, T. S. et al.(1997) and (1998) proposed a universal building model, which was an attempt to support multiple designers using different applications forming an effective multi-user collaborative design environment. It is based on a

central model that can be either partially or entirely shared by participants, and specific design ‘views’ which are defined as units of organization specific to each participant.

Although his approach supports different views, it is focused on converting and updating the integrated model from multiple sources at the level of the applications themselves into a generalized description of the entire building. Eastman’s, and other centralized data models, were well defined and tractable enough to support a limited number of participants. However, the complexity of the architectural product has generated more problems than the shared data model could solve. The IAI have also identified the most time consuming aspects of their IFC definition problems as follows:

- Ensuring as wide an agreement as possible within the industry on semantic definitions.
- Obtaining model reviews, handling the issues that result from review and ensuring that issues resolution is open for all members to see.
- Integrating all domain developments into a single model that is internally self consistent.

The listed difficulties would be true of all the integrated model approaches. It has become clear that a single data model would not be able to serve all the requirements of all the participants. In addition, the sheer magnitude of the combined data often exceeds the capability of its management by any one domain. Although an integrated model was expected to achieve interoperability among different domains of expertise, it actually exacerbated their fragmentation and the symmetry of ignorance. The nature of architectural design has proven to be an obstacle to the kind of utopianism embodied in the data-centric approach. It can be likened to the making of a puzzle, where designers search for individual solutions (spatial, structural, material, economical, etc.) that can fit together in some spatio-temporal context. It is an iterative, dynamic process, where propositions are made and tested against goals and constraints that are both internal to each domain of expertise, as well external to them (i.e., originating in

other domains).

## **The Filter Mediated Communication Model**

When the participants in a design process make decisions and negotiate with one another, they use their own representations, knowledge, methods, and resources. They must satisfy two sets of goals: shared (or common) project goals, and their own individual goals, which derive from their role, personality, culture, education, interests, etc. Each participant’s primary source of input is the other participants’ most recent outputs. Since the representations used by the participants may be different, the ‘input’ may have to be translated into each participant’s own form of representation before it can be processed. In practice, that explains, for example, why structural engineers re-draw, or re-model the plans given to them by architects, and why architects ‘translate’ the clients’ requirements into their own form of representation.

To illustrate these issues, let us take three participants in the design of a house: an architect, a builder, and a client. The client makes one or more lists of requirements and constraints, using his/her own representational methods. The client ‘publishes’ the most desirable requirements, including such information as the list of desired rooms, their sizes, and the desired adjacencies among them, in a way that can be read by the architect and the builder. Based on the client’s input, the architect designs a house, using his/her knowledge and method of representation. Like the client, the architect most likely will design a number of alternatives with varying parameters (size, materials, complexity of design and so on). S/he ‘publishes’ the one that seems most appropriate, from his/her point of view. The client may review the design, and ask for some changes, or ask the builder for an estimated cost of construction. The builder calculates the cost of construction, based on the most recent version of the architect’s design and the client’s requirements, which s/he then ‘publishes’.

The client may reject the estimated cost, asking the builder and/or the architect for modifications.

This scenario is, of course, only a brief description of a rather lengthy and iterative process. But it suffices to show how each participant in the process alternates between his/her 'private' representations, used during his/her own, internal design process, and the 'public' version which s/he 'publishes' for the benefit and use of the other participants. We assume a kind of 'filtering' mechanism mediates between the private (domain-specific) workspace and the shared and public workspace (Gruber, T. R. 1993). This filter strips the published version from each of the participants' private notations, sketches, calculations, and other design representations that the participant uses during the design process, and which would be of no use to the other participants. A similar but inverse filtering occurs when each participant receives the input generated by the others, and interprets and translates it into his/her domain-specific representation. It is the process that adds disciplinary knowledge to inputs received from other participants, and which makes each participant's own discipline-specific representations easier to comprehend by the others (much like a perspective drawing is a form of representation that makes it easier for the client to understand the architect's design). The 'filter' is thus a bridge between the private workspace of each participant, and the public, shared workspace

This approach matches conventional multi-disciplinary design processes, and differs from computer-assisted collaborative design as described earlier. In conventional design processes, such 'filters' are the knowledge used by each of the participating professionals. It differs from the data-centric models of collaboration in that it uses no centralized database. Rather, the project data is a collection of the individual contributions of the participants. If any one of the participants wishes to see data produced by another participant, s/he has to retrieve the relevant information from that participant, translate it into its own representational form within its own workspace. The entire project can thus be regarded

as a compilation of each individual's own representations. Each one of the participants is responsible for authoring their own representations, and for retrieving the latest version of the design produced by the other participants. There is no centralized control over the overall compilation.

## Computational Implementation of the Filter

### Semantics in the Model

Building semantics among the participants is the first step towards inter-domain communication. It adds to the shared data the conceptual ontology that any one domain expert may take for granted, but which would be viewed differently by another domain expert. It thus provides a more explicit, but abstract way to describe information, encapsulating both conceptual and domain-specific data models (Carrara G. 2002). **The conceptual models may include elements such as generalization, aggregation, and cardinality constraints about the objects (e.g., that a door is a kind of opening, and belongs to a specific wall).** The domain models deal with vocabularies defined by domain-specific ontologies, such as architecture, structural engineering, mechanical engineering, and general contractor to name a few.

### Implementation of the model

The filter of each participant needs to translate the incoming representations from other participants based on his or her domain specific knowledge, and to publish his or her representations with semantics. We created a central repository with an XML database (e.g., eXist, [exist.sourceforge.net](http://exist.sourceforge.net): May 2008) that allows us to gather ontological information from the participants without having any modification. This central repository was intended to be the shared design workspace where every participant with proper permission can publish their own representations in XML format. For this matter, we chose the X3D ([www.w3.org/DesignIssues/Notation3.html](http://www.w3.org/DesignIssues/Notation3.html): May 2008) for

Figure 2  
A RDF triple based on graph model and its application to an X3D representation

storing both geometric data and ontological data. The main interaction with the shared design workspace will be done through queries especially in the format of XPath ([www.w3.org/TR/xpath](http://www.w3.org/TR/xpath): May 2008). The primary benefit of using the XML database for a central repository is that the representations that each participant contributes can be queried by a proper mechanism although we cannot guarantee that any results can be retrieved or not.

Figure 1  
The shared design workspace and private workspaces

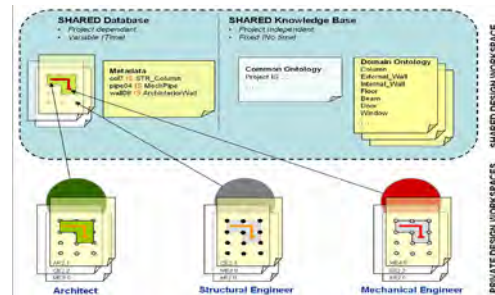
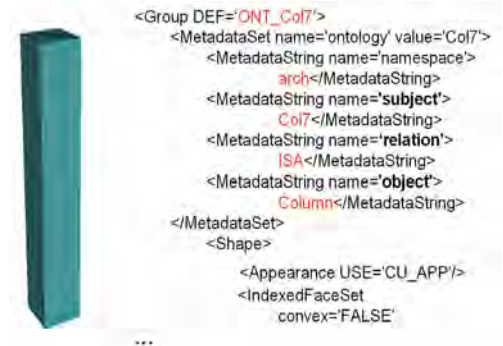


Figure 1 shows a detailed description of the shared design workspace where an architect, structural engineer and mechanical engineer have to work together as an example. In the shared knowledge base, there are common ontology and domain-specific ontologies that are project-independent and reusable through the projects. To make it working in the real world situations, we identified four units including Building Unit (e.g., hospital, outpatient department, intensive care department, etc.), Space Unit (e.g., ward, bedroom, bathroom, etc.), Construction Unit (e.g., partition, floor, wall, window, door, etc.), and Functional Unit (e.g., furniture, equipments, etc.). If there needs additional semantic objects, then the participants can add them upon agreement. The shared database is project-dependent and keeps the latest versions contributed by the participants through their own filter along with semantic descriptions. Each entity in the shared database is a derived object from the shared knowledge base. This description can be embedded in any geometric



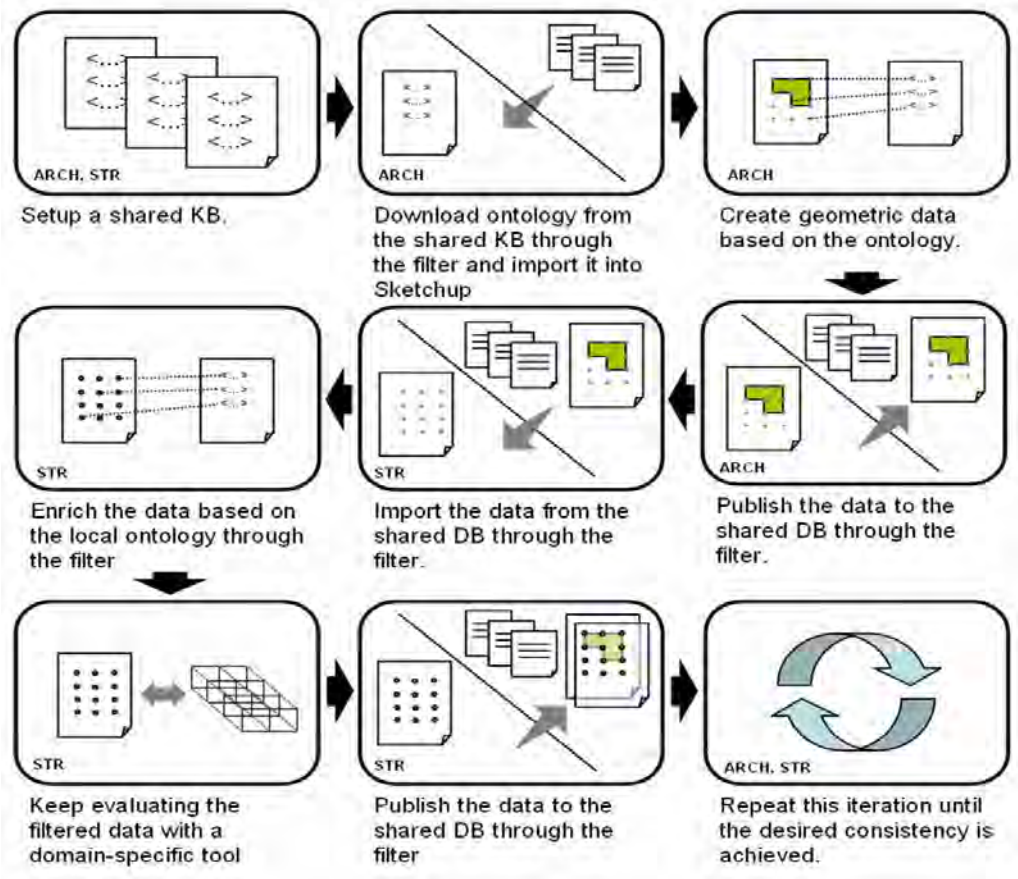
formats as long as they support metadata description. For instance, a column is a Construction Unit semantically and can be described in X3D format at the same time as shown in Figure 2. The format of the semantics incorporated in the model follows Notation 3 or N3 ([www.w3.org/DesignIssues/Notation3.html](http://www.w3.org/DesignIssues/Notation3.html): May 2008) that is a shorthand non-XML serialization of Resource Description Framework (RDF) ([www.w3.org/RDF](http://www.w3.org/RDF): May 2008).

### Tools supporting the model

In order to make the model usable in the real situation, we concocted a pipeline that includes Google's Sketchup ([www.sketchup.com](http://www.sketchup.com): May 2008) in conjunction with the XML database. We used Sketchup as a semantic authoring tool as well as a geometry manipulation tool in private workspace. Based on Sketchup's Ruby programming language extension ([www.ruby-lang.org/en](http://www.ruby-lang.org/en): May 2008), we made several plug-ins to create semantics according to the participant's domain knowledge and share it through the filter mechanism. The image on the right in Figure 3 shows a semantic relationship based on the geometries. Each elliptical shape represents a class and



Figure 5  
An illustration of the process  
in the model



again and the situation that a structural engineer has to verify the architect's structural durability. 'Col7' is an object that belongs to the architect's dictionary at first. The structural engineer has to take the architect's object from the repository and conduct necessary analysis. After having several iterative analyses, the structural engineer returns his representation to the repository with the same 'Col7'. Now, 'Col7' in the repository belongs to not only the architect but also the structural engineer. If there are other participants who are interested in 'Col7', then they have

to maintain this convention to keep consistency among them.

### Processes in the model

It would not be feasible to describe the whole process in a project. Figure 5, however, shows an iterative process possibly between the architect and the structural engineer through the shared design workspace implementing the filter mediated communication

## Conclusions and Future Work

Every participant creates their own objects by specifying geometric and non-geometric information as well as ontological information (or semantics). Consider that the architect designs a building with his own objects. In the schematic design phase, his objects would be generic in that they only describe the function or performance and some of the simple geometry specific to that object. It would be used to define spaces, enclosures and openings while he develops some architectural plans of the building. In the course of subsequent phases, more detailed specifications could be determined and added to the objects. For example, a door in the preliminary design phase would be used to define an entering point to analyze circulation. At this point, no specifications are needed. The colors, material, finish, sound, fire-rating, and price might not be taken into account. In the design development, the architect publishes the plans and the same door object would be specified in more detail by the participating vendors. A participant-oriented representation of the architect's door can be incrementally enriched by the pieces of the other participant's descriptive knowledge. Once the participants contribute their knowledge to the representation, it would be possible that each of them could see the other's point of view. The dynamic and semantically-rich representation would allow the participants to make alternatives reflecting their intents more effectively, which eventually leads to a state of shared understanding.

The published data from other participants can be treated as a collection of documents. It is natural that the participant may want to retrieve some information from the published data (e.g., floor plans). The participant can specify some queries (e.g., 'find all doors', 'show all doors owned by both the architect and the fire marshal', and so on), which can be performed by the participant's own filter. Since this type of queries is formulated by the participant intentionally, the filter will show the results only when asked. When an object appears in more than one

data set, with different ontologies (e.g., the architect's wall and the structural engineer's wall), the filter will recognize this duplicity and perform consistency-management, possibly formulating queries on behalf of the architect. Suppose that the structural engineer designated an architectural wall object as a bearing wall. The architect may try to put a door in that wall. His filter, recognizing that the wall has multiple anthologies, will automatically make query the structural engineer's ontology to verify that the door may be placed in that wall. The filter will alert the architect that the proposed action may pose a problem. The architect may select another wall for placing the door, thereby avoiding a conflict, or continue with the original action, which will require additional action on the part of the engineer. Through this process, the architect may reduce design errors and delivery time without sending his/her designs to the structural engineer and achieve his/her design goal more efficiently.

The impact of a network-based collaborative design transforms a hierarchical/linear partitioned process into a distributed and interleaved one (Kalay, Y. E. 2004). In the filter mediated communication model, the participating professionals can affect one another bi- or multi-directionally. The participants do not have to share a large and heavy integrated model. Rather, their intelligent filter will access the information at the object level (geometric/non-geometric information and ontologies) which resides at its own location and translate it into their own representation using user-defined ontologies. They would fill the gap between the heterogeneous representations preserving semantics as long as they are based on the syntactical agreement (e.g., XML). While producing and consuming the information, the participants and their filters construct a 'knowledge chain' in which subassemblies of the information are passed from one filter to another, each one contributing its own piece of knowledge.

## References

- Carrara, G. and Fioravanti, A.: 2002, 'Private Space' and 'Shared Space' Dialectics in Collaborative Architectural Design, Collaborative Decision-Support Systems, Focus Symposium and InterSymp-2002 Conference Proceedings, Baden-Baden, pp. 27-44.
- Eastman, C., Bond, A. and Chase, S.: 1991, A Data Model for Engineering Design Databases. Paper presented at the 1st International Conference on A.I. in Design, Edinburgh, Scotland.
- Eastman, C., Jeng, T. S., Chowdbury, R. and Jacobsen, K.: 1997, Integration of Design Applications with Building Models, Paper presented at the CAAD Futures '97.
- Gruber, T. R.: 1993, Toward Principles for the Design of Ontologies Used for Knowledge Sharing: Knowledge System Laboratory, Stanford University, CA, USA.
- Jeng, T.-S. and Eastman, C. M.: 1998, A database architecture for design collaboration, *Automation in Construction*, 7(6), pp. 475-483.
- Kalay, Y. E.: 2004, *Architecture's new media : principles, theories, and methods of computer-aided design*, MIT Press, Cambridge, Mass.
- Tolman, F. P.: 1999, Product modeling standards for the building and construction industry: past, present and future, *Automation in Construction*, 8(3), pp. 227-235.