

**EXPANSION OF BUILDING INFORMATION MODEL FOR DISASTER MITIGATION**

Cheng-Ting Chiang, Chia-Ying Lin, and \*Chien-Cheng Chou  
*National Central University*  
*300 Jhongda Rd.*  
*Jhongli, Taoyuan 32001, Taiwan*  
(\*Corresponding author: [ccchou@ncu.edu.tw](mailto:ccchou@ncu.edu.tw))

---

## **EXPANSION OF BUILDING INFORMATION MODEL FOR DISASTER MITIGATION**

### **ABSTRACT**

Information integration and exchange for the entire building life cycle is one of the most important issues in the AEC industry. Although the Building Information Model (BIM) technology is widely utilized to address them, BIM is mainly concerned within the design and construction (D&C) phases. In the operation and maintenance (O&M) phases, however, the information integration and exchange issue becomes more complicated because software applications here require not only BIM but other information outside of the AEC domain, e.g., disaster mitigation. Nowadays, most BIM-related research focuses on the D&C phases and pays less concern on the O&M phases, hence there is a research need that can help seamlessly transfer the information between the D&C and O&M phases. In this research, a software framework using ontology was designed to deal with aforementioned problems. The BIM data from the D&C phases were used as a foundation to generate new models for different software applications for the O&M phases. The ontology technique was employed to assist the model transformation process between the original BIM and the models for applications. Finally, a disaster mitigation example was chosen to validate the framework, in order to make sure that the transformation process is comprehensive and useful from domain experts' perspective.

### **KEYWORDS**

Building information model, Model-driven architecture, Disaster mitigation.

### **INTRODUCTION**

Building Information Model (BIM) is a promising technology that can provide a communication and information exchange platform for all stakeholders involved in a building, as well as create a 3D display environment to clarify a building's virtual representations. Many researchers study BIM-related topics, such as design and engineering, linking to analysis tools, energy innovations, facility management, and so on (Becerik-Gerber and Kensek, 2010). However, most studies are concerned with the design and construction (D&C) phases of a building, few studies deal with issues pertaining to the operation and maintenance (O&M) phases (Akcamete et al., 2010; Vanlande et al., 2008). Because costs associated with the O&M phases of a building are always higher than those with the D&C phases (Gallaher et al., 2004), and because BIM is becoming a mature software tool capable of supporting all kinds of information-intensive activities, applying the BIM technology to the O&M phases is desired.

Although researchers have shown that a building construction project and its subsequent facility maintenance projects are dissociated (Vanlande et al., 2008), it is obvious that the applications in the O&M phases require the information from the D&C phases of a building. In the D&C phases, BIM applications are designed to use parametric and object-oriented modeling techniques to represent each individual building element and help associated processes. Applications in the O&M phases may require not only original BIM-related information but the enhanced one that can be employed to record and manage various activities during O&M. Example applications include facility management systems and disaster management systems. Current practices require contractors provide a building's handover information such as Construction-Operations Building information exchange (COBie) for later usage in the O&M phases, but they usually spend minimal efforts so as to satisfy the contract (Akcamete et al., 2010). Hence,

---

a better information bridge may be needed in order to provide transformation and customization of the handover information from the D&C phases to the O&M phases.

Applications in the O&M phases possess several unique requirements regarding interpretation of BIM. Some BIM information should be accessed and handled via an easier method while the other should be further processed through use of a more complicated interface. In other words, customization of BIM is needed for the O&M phases. In fact, the utilization of BIM should follow the open/closed principle (OCP) – “open for extension, but closed for modification” (Meyer, 1988) as well. In the closed side, for example, structural elements in a building will not be changed during their entire life cycle (Hassanain et al. 2001); and thus, for application developers in the O&M phases, it may be a good way to operate such information at the class level, not at the metaclass level to avoid future modification. By operating BIM at the metaclass level, we mean that O&M application developers regard the structural elements as objects and use generalized classes such as Room, Column, and Wall to access each individual element. When facility managers would like to query one specific room’s maintenance history, application developers need to select all rooms’ maintenance objects and then filter out the records not belonging to the room specified. It is an indirect approach, and because the structural elements are constant, developers actually can regard each element as a distinct class in order to use an object to represent the current condition of an element at a specific time instance. In this way, all BIM or IFC-related functions are encapsulated in one class for an element. Application developers can have a more simple data access interface to manipulate O&M-related information. The traditional approach regards the structural elements or other elements infrequently changed as objects regardless of the D&C or O&M phases; and thus, such O&M applications may use additional redundant codes to retrieve information.

In the open side of the OCP, current materials and equipment information stored in BIM cannot accommodate various needs during the O&M phases, e.g., spare parts list management, which is an essential function from the facility management perspective. An extension mechanism of BIM may be needed and has been proposed by several studies (CRC, 2007). Among the O&M-related activities, the concept of rooms and zones forms the central management unit, which is just one of the derived attributes and is not the focus during D&C. Proper spatial containment of related building elements is essential in the O&M phases (East et al., 2012). In the D&C phases, architects design the form or the placement of each building element. But facility managers need to know the location of a specific device in order to perform maintenance work. A typical room consists of many building elements, and in BIM tools such as Revit, designers need to draw walls, windows, and doors first, and then specify the boundary of a room to link these building elements. Additional furniture or equipment elements may need to be attached to some rooms during the O&M phases. The traditional approach to transferring handover data requires manually create “polylines” of a building instead of directly using the digital building information used in the D&C phases. The traditional approach needs couple days and is labor-intensive. Even if computerized tools can be provided to assist in the transformation process, such tools describe only the geometric information and do not consider the integrity of each building element as a room or zone in the O&M phases.

Finally, several studies have recognized the lack of 3D visualization capabilities in most applications in the O&M phases (Akcemetet et al., 2010). BIM, like a spatial analysis engine (Akcemetet et al., 2010), can assist facility management applications in analysis of spatial relationships between these building elements. The other issue is that the O&M phases consist of many different applications, such as disaster mitigation, security, community care, and so on. In these fields domain-specific requirements exist, and BIM cannot cover all the fields which need building information as a possible input source.

In this research, the encapsulation of BIM (E-BIM) was designed following the model-driven architecture (MDA) technique for automatic and comprehensive transformation and customization of the handover information from the D&C phases to the O&M phases. A metamodel hierarchy of a building was constructed to encapsulate the building information, not simply transforming data into an open format, but playing the role of an “interface” – users can manipulate the building information through the model instead of directly modifying it. Regarding to the designed metamodel hierarchy, we also attempted to

present spatial relationships to reshape data structures of maintenance data. The proposed MDA approach was utilized with the hierarchy to generate codes for further applications of building information.

## RELATED WORK

Nowadays, the software development process is facing more and more complicated situations. Not only requirements become more dynamic but different platforms can be selected as the working environment. To solve these problems, MDA was proposed by Object Management Group (OMG), mainly for the purpose of integration and interoperability (Soley, 2000). This approach basically uses a series of formal models to help the software development process.

To take an overview of MDA, at first the separation of concerns should be introduced. Three MDA viewpoints of a system are described as follows: (1) Computation Independent Viewpoint (CIV), (2) Platform Independent Viewpoint (PIV), and (3) Platform Specific Viewpoint (PSV). CIV contains none of the computer-related processing details and only focuses on the business requirements; PIV deals with the operations of a system but does not contain the details for a specific platform; PSV integrates PIV with the details of using a particular platform as the development environment (Miller and Mukerji, 2003).

As shown in Figure 1, four layers and their transformation mechanisms are identified based on these views. A computation independent model (CIM) is constructed from CIV. Based on the CIM a platform independent model (PIM) is created. With the sufficiently complete and precise PIM, a platform specific model (PSM) can be generated by using model-to-model transformation mechanisms, and the specific code model – which can be viewed as implementation – can be automatically transformed from the PSM.

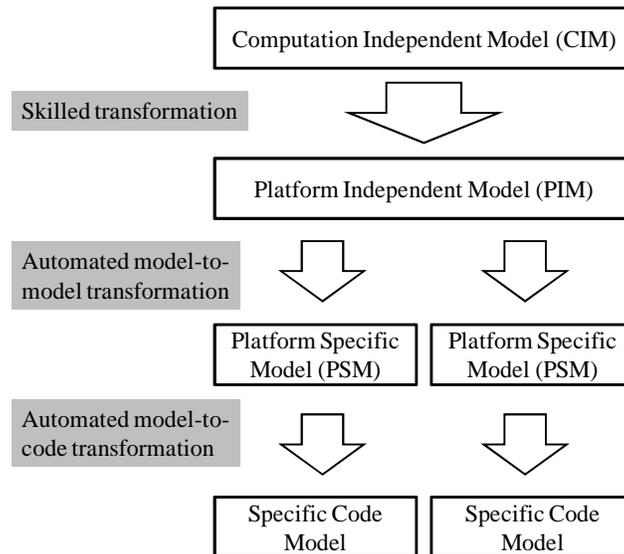


Figure 1 – MDA process

OMG defines a standard – Meta Object Facility – to provide metadata management and modelling language definitions. MOF is used with a metamodel hierarchy shown in Figure 2. A run-time system can be interpreted by a UML model. Since the UML model is designed by stakeholders, there can be many UML models from different perspectives. To describe a UML model, we need to define UML descriptions and notations as a communicating method. The UML definitions are also based on the definitions of MOF. With the descriptions of higher layers, the lower layer can be clearly explained.

## RESEARCH APPROACH

Because of the wide use of BIM, building information is available for stakeholders in the O&M phases. However, the BIM data structure is not easy to be used for users and needs extra efforts for application developers to write programs for further use of such information. In order to offer a better way for the extended use, this research defines a metamodel hierarchy for a building to interpret their various elements.

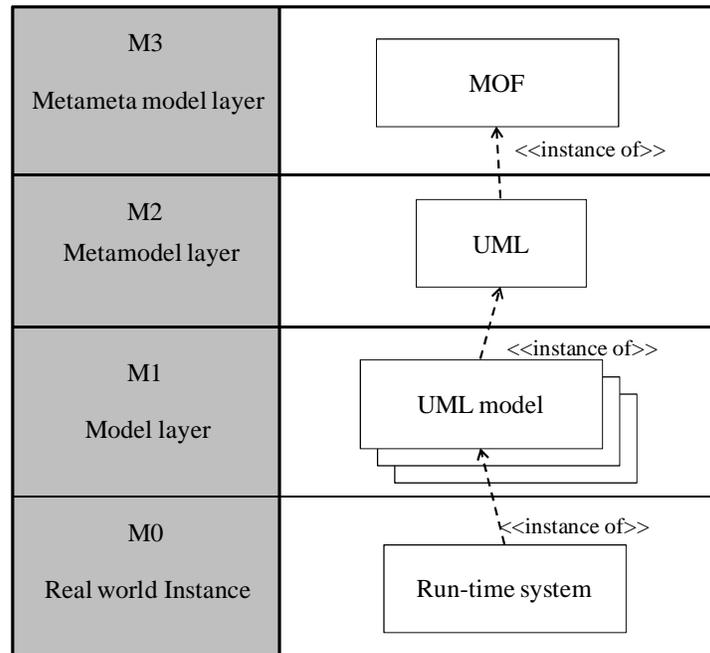


Figure 2 - Metamodel hierarchy

Figure 3 shows the sample definitions of the metamodel hierarchy of a residential building. This hierarchy is designed as the basis of the proposed approach.

- **M3 layer:** For different perspectives, the contents in M3 layer should be viewed as different roles. Inside this hierarchy, the M3 layer describes basic components of a building, such as doors, walls, windows, and so on. These components exist in any type of building, in other words, they are essential for a building; as a result they are designed to be in the M3 layer. Outside this hierarchy – which means extended applications without sufficient understanding of the building internal structure and BIM programming – for them, the layers below and those basic components are encapsulated due to OCP. Building information is available for them through the “BuildingSuperObject” but its data structure cannot be changed from the outside, that is, “BuildingSuperObject” is an interface as a communicating bridge.
- **M2 layer:** This layer is instantiated from basic components from the M3 layer, as well as different types of buildings and their basic units are described here. Different types of buildings may have rooms/zones/spaces for different purposes, and rooms are classified by purposes as a basic unit. For example, an apartment is classified as one type of residential buildings. Bedrooms and living rooms may only exist in a residential building, not in a factory.
- **M1 layer:** In the M1 layer, a “model” means a specific building and is constructed here. Most of the static components are from higher layers. Also, the constant spatial components of a building are described here, since most of the time the floor plan of a building will not be changed after the D&C phases.

- M0 layer: Finally, the instances of M1's building are stretched by time as a timeline and record the whole life cycle, which means the building information covering the D&C phases to the O&M phases exists in this layer.

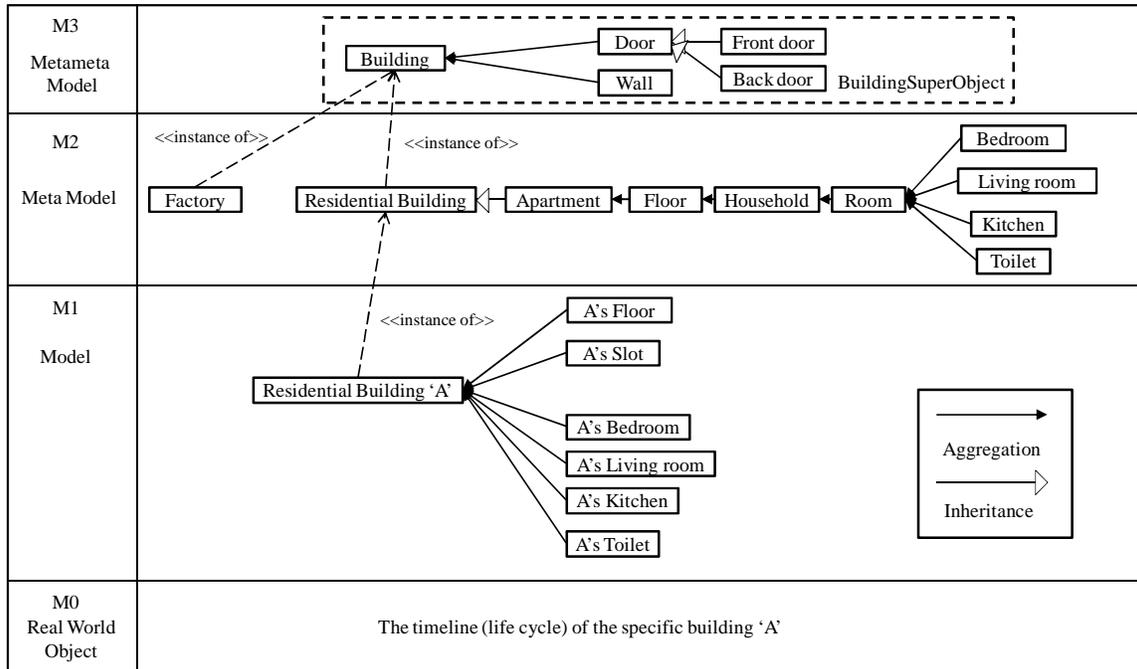


Figure 3 - Metamodel hierarchy of a building

Suppose BIM data of a specific building is available. First its spatial relationship and the usage of spaces are re-drawn as the model shown in the M1 layer of Figure 3. The class library of this building can then be automatically generated by the model. Since attributes are also predesigned to be contained in the model, extended applications can use data they need through the model.

In Figure 4, the E-BIM process was designed referred to the MDA process; the original BIM of the specific building can be viewed as the CIM level, and then it will be transferred to E-BIM form preparing to be extended by different applications. Different application domains are considered PSM in MDA process; therefore, specific domain knowledge is used to be the additional information and taken as the foundation of the specific applications.

### EXAMPLE

Since the more floors and rooms a building contains, the more complicated the problem is. A three-floor apartment is presented in this section as a simple example to show the contents in metamodel hierarchy of an apartment building 'B'. Some details are also omitted for clearly interpreting the proposed approach. The relationship between contents and the expected application of the metamodel hierarchy are also interpreted in this section.

Figure 5 presents a sample model constructed following the metamodel hierarchy in previous section. In apartment building 'B', there are three floors and an elevator. There are two households in each floor, and the rule of the address assignment is the floor number, plus the serial number. Each household is assumed to have three rooms, i.e., living room, bedroom and toilet. In Figure 5, rooms in address '02' are omitted. These rooms define attributes from their designed usage respectively. In addition, from the spatial perspective, 'B' can be viewed in two dimensions: horizontal and vertical. The

horizontal dimension is equivalent to the floor concept; the vertical dimension, named “slot” here, is composed of rooms located on the same vertical line. Usually the floor plan of a household is the same as its neighbour upstairs or downstairs in an apartment; therefore, rooms of the same purpose belong to the same slot. However, the elevator is independent of a floor or a slot since it is movable.

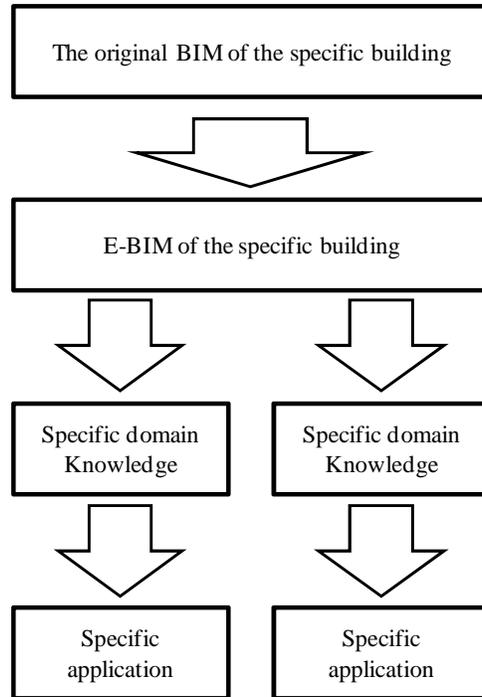


Figure 4 - The E-BIM process referred to the MDA process

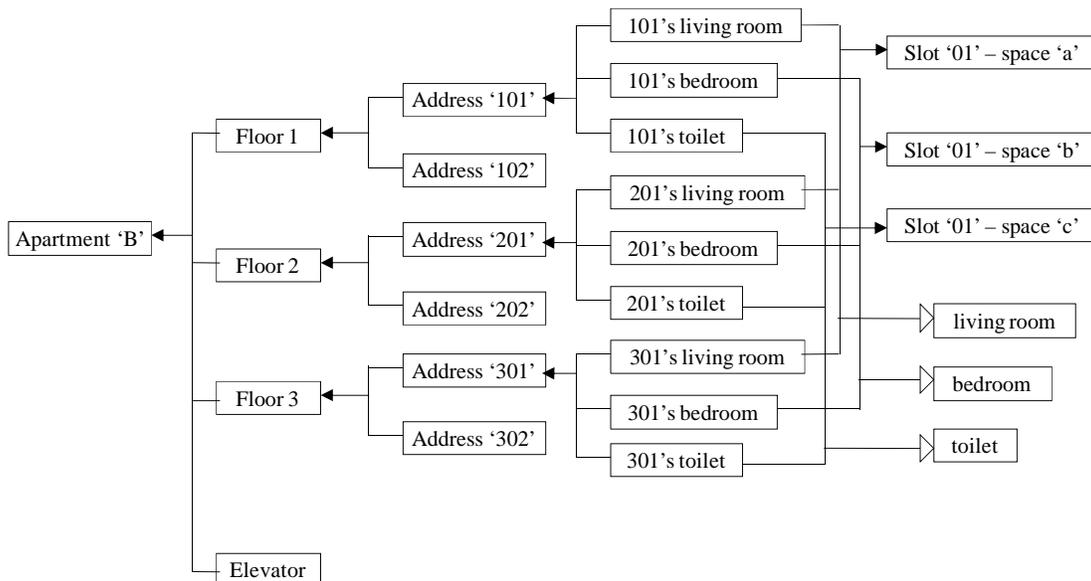


Figure 5 - The model of a specific apartment building 'B'

The model described above becomes a basis for automatic code generation; a class library of 'B' is constructed for extension. The extended applications can use BIM data through this model – more

specifically, users of these applications access BIM data through this model instead of directly operating with BIM. Since it is not easy to cross the threshold of the programming issues for BIM applications, it is believed that using this model will assist the development of extended allocation for BIM.

## CONCLUSIONS

This research has proposed the software architecture to integrate the static (BIM) information with the dynamic, O&M-related data. The MDA technique was utilized to store and synthesize the data. The BIM information is used to show the geometry aspect of the building. MDA has been successfully applied to other industries' applications, including the AEC industry. Further enhancement of the system is needed in order to integrate more dynamic information from different facility management domains and/or other BIM programs.

## REFERENCES

- Akcamete, A., Akinci, B., & Garrett, J.H. (2010). Potential utilization of building information models for planning maintenance activities. In W. Tizani (Ed.), *Proceedings of the International Conference on Computing in Civil and Building Engineering*, Nottingham, UK.
- Becerik-Gerber, B., & Kensek, K. (2010). Building Information Modeling in Architecture, Engineering, and Construction: Emerging Research Directions and Trends, *Journal of Professional Issues in Engineering Education and Practice*, 136(3), 139-147.
- CRC for Construction Innovation. (2007). *Adopting BIM for facilities management: Solutions for managing the Sydney Opera House*, Brisbane, Queensland, Australia.
- East, E. W., Nisbet, N., & Liebich, T. (2013). The Facility Management Handover Model View, *Journal of Computing in Civil Engineering*, 27(1), 61-67.
- Gallaher, M.P., O'Connor, A.C., Dettbarn, Jr., J.L., & Gilday, L.T. (2004). *Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry*, National Institute of Standards and Technology, Gaithersburg, Maryland, USA.
- Hassanain, M.A., Froese, T.M., & Vanier, D.J. (2001). Development of a maintenance management model based on IAI standards, *Artificial Intelligence in Engineering*, 15, 177-193.
- Jardim-Goncalves, R., Grilo, A., & Steiger-Garcia, A. (2006). Challenging the interoperability between computers in industry with MDA and SOA, *Computers in Industry*, 57, 679-689.
- Meyer, B. (1988). *Object-Oriented Software Construction*, New York, NY: Prentice Hall.
- Miller, J., & Mukerji, J. (2003). MDA Guide Version 1.0.1, Object Management Group, Retrieved from <http://www.enterprise-architecture.info/Images/MDA/MDA%20Guide%20v1-0-1.pdf>.
- Soley, R. (2000). *Model Driven Architecture*, White Paper, Draft 3.2, Object Management Group.
- Vanlande, R., Nicolle, C., & Cruz, C. (2008). IFC and Building Lifecycle Management, *Automation in Construction*, 18, 70-78.
-