통합설계모델을 위한 프로세스 제어

Process Management for Integrated Design Model

이 창 호*
Lee, Chang-Ho

(논문접수일 : 2001년 3월 5일 ; 심사종료일 : 2001년 5월 14일)

요 지

빌딩골조 구조물의 구조설계는 시스템, 서브시스템, 기본 부재의 여러 설계 단계에 걸쳐서 복잡하게 진행된다. 여러 가지 설계 대안들을 고려하고 비교하여 선택하는 작업을 설계과정에서 행하게 되고, 최종적으로 한 가지의 구조시스템과 이와 관련한 서브시스템 및 기본 부재들을 결정해야한다. 빌딩골조 구조물을 위한 컴퓨터 통합시스템을 구축하기 위해서는 이러한 설계과정을 정형화하여 표현하는 것이 필요하다. 본 논문에서는 시스템, 서브시스템, 기본 부재 단계에서의 설계과정을 표현하기 위한 개체형 통합설계모델의 프로세스 개체들을 소개한 후에 각각의 프로세스개체 내의 설계흐름을 기술하였고, 세 가지 설계 단계에 걸쳐 있는 많은 프로세스 개체간의 설계흐름을 제어를 하기 위한 개념을 설명하고 있다. 프로세스 개체간의 제어는 어떠한 개체내의 프로세스가 끝났는가를 파악하여 구조설계를 완료하는데 필요한 모든 개체의 프로세스가 끝났는가를 확인하기 위함이다. 이러한 프로세스 제어의 개념은 빌딩골조 구조물을 위한 컴퓨터 통합시스템의 개발에 이용될 수 있다.

핵심용어 : 컴퓨터 통합시스템, 개체형 설계모델, 시스템설계, 설계과정, 설계작업

Abstract

The structural design process for a building frame structure at the system, subsystem, and basic component levels can be complicated. Multiple design alternatives are generated, compared, and selected during the design process. One structural system and related subsystems and basic components should be developed at the end of the design process. The representation of the design process is required for developing a computer integrated design system of a building frame structure. This paper presents the process entity categories in an entity-based integrated design model to represent the design process at the system, subsystem, and basic component levels. Then this paper discusses the sequence flow of each process entity category, the completion of the design process in the process entity category, and the management of the overall design process among process entity categories at the three design levels. The management of the overall design process identifies completed design processes and confirms the completion of all design processes of the process entity categories. These concepts can be applied to evaluate the design process of a computer integrated system for a building frame structure.

Keywords : *computer integrated system, entity-based design model, system design, design process, design information*

1. Introduction

A building frame structural system is often decomposed into subsystems such as frame subsystems and floor subsystems during the design process. A subsystem is decomposed into

* 정회원 · 국립한경대학교 건축공학과, 전임강사

• 이 논문에 대한 토론을 2001년 9월 29일까지 본 학회에 보내주시면 2001년 12월호에 그 결과를 게재하겠습니다.

basic structural components such as beams and columns. The design of a building frame structure includes the design of the system, subsystems, and basic components. The design process usually starts with the design of the system which determines the type of the system and the layout of the subsystems. Multiple alternatives for the structural system can be considered, but one structural system is not necessarily selected before the design of the subsystems is performed. The structural system may not be selected until the design of some basic components are initiated. The subsystems and basic components are redesigned if the design requirements are changed. These design processes need to be represented for developing a computer integrated system for structural design.

There have been some research efforts to formally describe dynamic design processes involved in structural and engineering design. The research efforts are for converting an ill-structured problem to a set of well-structured problems.[1] Sanvido et al.[2] developed an integrated building design process model(IBPM) for the process of providing a constructed facility. The model adopted functional modeling methodologies from computer-integrated manufacturing. Sause and Powell[3] proposed a multilevel selection-development(MSD) model for the structural design process. The model organizes the structural design process as a hierarchy of selection and development activities. Baugh and Chadha[4] described the use of process algebras in developing product and process models. The process algebras are used for verifying properties of systems that model concurrent activities. Park and Cutkosky[5] introduced a process modeling tool called Design Readmap(DR) which combines precedence, abstraction, and constraint relations to differentiate parallel and alternative paths. The complexity of managing design processes are implicitly addressed in developing a conceptual design of buildings by Rivard and Fenves.[6]

This paper focuses on the overall management of design processes in an entity-based integrated design model. An entity-based integrated design model uses product and process entities to describe design information and design activities, respectively.[7] Formal concepts for representing flexible sequences among design activities of multiple level process entity categories are described by Lee and Sause.[8] All necessary design activities in process entity categories at the system, subsystem, and basic component levels need to be completed at the end of the design process. The concepts for the overall process management identify uncompleted design activities of the process entity categories and confirm the design process is complete.

This paper begins with an example of the design of a building frame structure at the system, subsystem, and basic component levels. The process entity categories are presented for describing the design activities at the three design levels. Then the management of the design process in one process entity category and the design process among process entity categories at the three design levels are discussed.

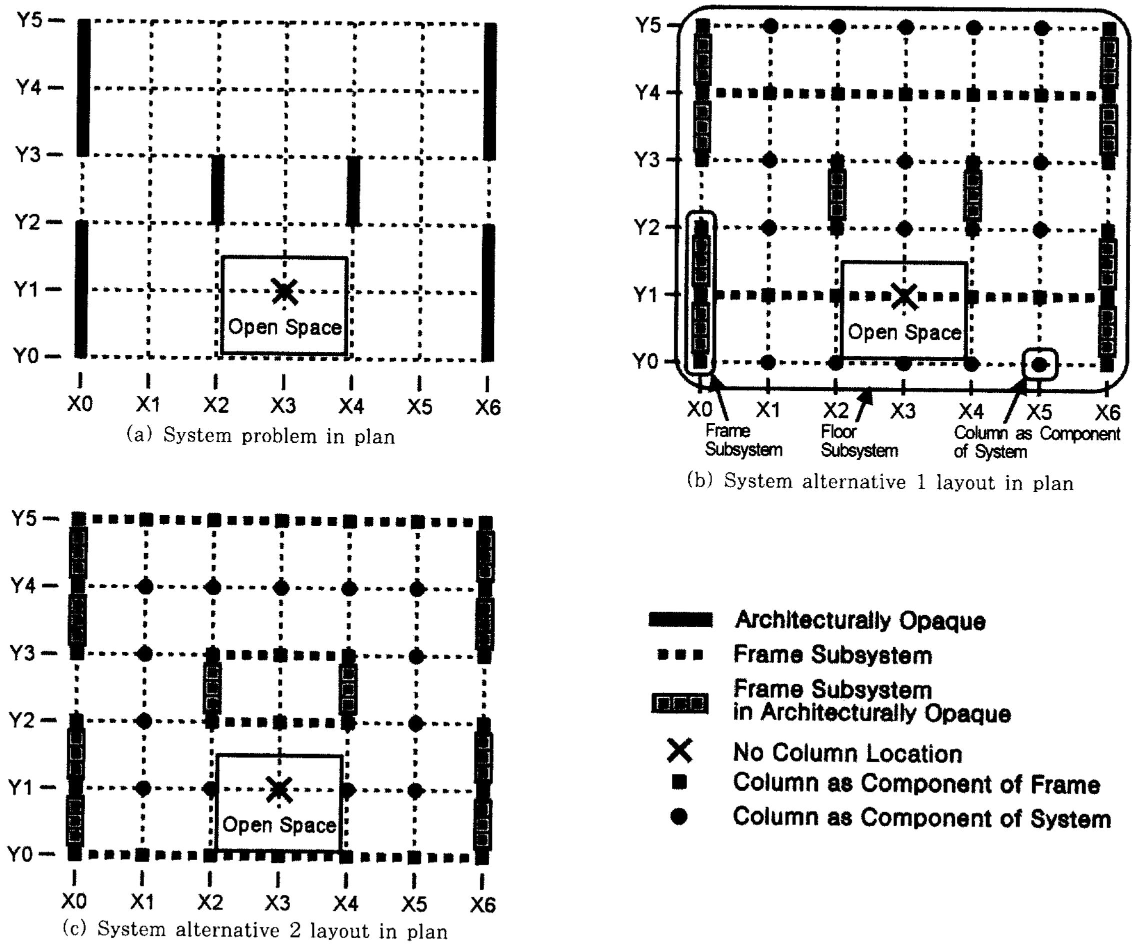
2. Design of Building Frame Structure

A building frame structure can be decomposed into subsystems such as frame subsystems and floor subsystems. A subsystem can be also decomposed into basic components such as beams and columns. An example of a building frame structure at the system, subsystems, and basic component levels is shown in Figs. 1, 2, and 3. This example is used in later sections for discussing the management of design processes.

2.1 Design of Building Frame Structure at System Level

The design of a building frame structure at the system level involves the description of system design problem and the layout of subsystems.[9] Fig. 1(a) represents an example of the system problem in plan of a four story building frame structure. The system problem in the figure includes an open space, no column location, and architecturally opaque planer regions[10] represented as thick solid lines. Examples of architecturally opaque planer regions include walls, floors, and roof. The other regions are transparent. Structurally a planer region is transparent or opaque. For example, simple framing and moment resisting framing are transparent, and braced framing is opaque.

Figs. 1(b) and 1(c) show two system alternative solutions for the system problem shown in Fig. 1 (a). The locations of frame subsystems, represented as thick dotted lines, in each plan are different in Figs. 1(b) and 1(c). Each system alternative has a number of frame subsystems. Some frame subsystems are located in the architecturally opaque regions, and others in the architecturally transparent regions. One floor subsystem is included in each plan. Some columns, represented as squares, are components of frame subsystems. Others, represented as circles, are not components of any frame subsystems. They are components of the system. Thus, a system has been decom-



(a) System problem in plan

(b) System alternative 1 layout in plan

(c) System alternative 2 layout in plan

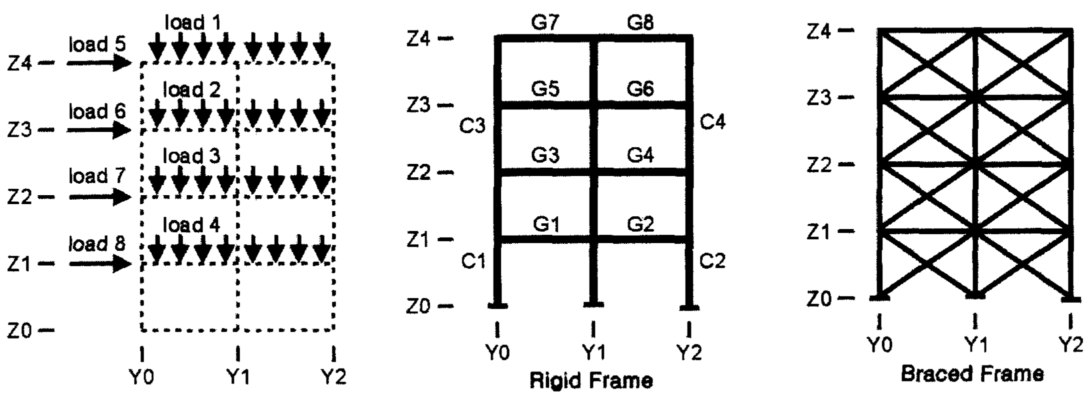
Fig. 1 Design of building frame structure at system level posed into frame subsystems, floor subsystems, and columns as components of the system. Two system alternatives in Figs. 1(b) and 1(c) have been generated. These are compared during the design process, and then one system alternative should be selected and developed by the end of the design process.

2.2 Design of Building Frame Structure at Subsystem Level

System alternative 1 in Fig. 1(b) can be selected as the best solution for the system during the design process. The system alternative includes a number of frame subsystems, floor subsystems, and columns as components of the system. The design of one frame subsystem, located on gird line X0 from row Y0 to row Y2, is shown in Fig. 2. The frame is to be designed to satisfy its design requirements including the loads applied to the frame in Fig. 2(a). Two frames alternatives, rigid frame and braced frame, are shown in Figs. 2(b) and 2(c). Since the frame is located on the architecturally opaque planer region (Fig. 1(a)), the braced frame in Fig. 2(c) is considered as an alternative. Two frame alternatives are compared during the design process, and then one frame alternative should be selected and developed by the end of the design process. The alternatives of other frame subsystems and floor subsystems are similarly generated, and their best alternative solutions should be also selected and developed by the end of the design process.

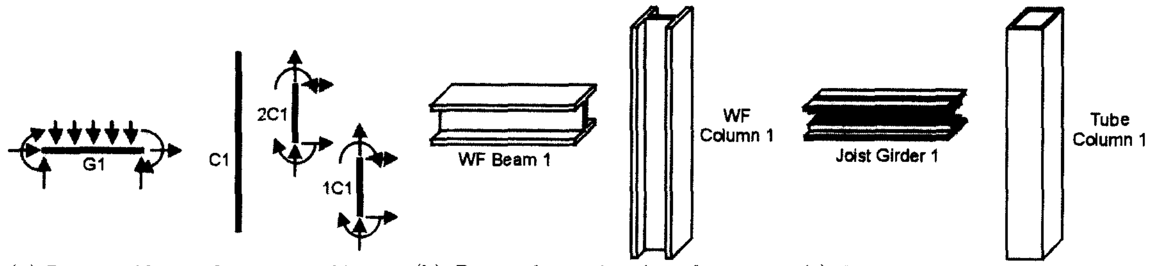
2.3 Design of Building Frame Structure at Basic Component Level

Frame alternative 1 in Fig. 2(b) can be selected as the best solution for the frame problem in Fig. 2(a) during the design process. The rigid frame in Fig. 2(b) includes a number of basic components such as beam G1 and column C1. Given the loads applied to the frame, the structural analysis determines the forces for the beam and the column (Fig. 3(a)). Since column C1 extends over two stories, the analysis may provide the forces for each story of the column. These forces are applied as design requirements for the beam and the column. Two alternative solutions, WF beam 1 and joist girder 1, for the beam, and other two alternative solutions, WF column 1 and tube column 1, for the column are shown in Figs. 3(b) and 3(c). One best solution for a beam and one best solution for the column are selected during the design process.



(a) Frame problem (b) Frame alternative 1 (c) Frame alternative 2

Fig. 2 Design of building frame structure at subsystem level



(a) Beam problem and column problem (b) Beam alternative 1 and coumn alternative 1 (c) Beam alternative 2 and column alternative 2

Fig. 3 Design of building frame structure at basic component level

WF beam 1 and WF column 1 can be selected as the best solutions. The specific materials and dimensions should be also determined by the end of the overall design process.

3. Process Entities and Sequence Flow in Product Entities

The design of a building frame structure can be represented as an entity-based integrated design model. An entity-based integrated design model involves a number of product and process entities.[7),11),12)] The product entities describe design information, and the process entities describe design activities. The product entities are created by the process entities, and the created product entities are used by other process entities. The main design activities associated with the design of a building frame structure at the system, subsystem, and basic component levels shown in Figs. 1, 2, and 3 are represented as process entity categories in Figs. 4, 5, and 6, respectively. The process entity categories are represented using the notation for entity-based integrated design models.[11),12)] Each rectangle in Figs. 4, 5, and 6 indicates a process entity category from which product entities can be initiated when it is applied to a specific design project. The process entity categories include: (1) "selection entity" categories and (2) "development entity" categories. Each selection entity category describes the generation, comparison, and selection of a number of alternative solutions for a design problem, and each development entity category describes the development of an proposed solution to a design problem.[11),12)] Attributes of the entity category are shown below the rectangle with horizontal bars. If the attribute is single-valued, the bar ends with an empty circle, and if the attribute is multiple-valued, the bar ends with the black circle. The value set of an attribute (the set of possible values for the attributes) is represented in square brackets. The attribute type is represented in parentheses. "B" indicates a base attribute, "DI" an internally derived attribute, "DE" an externally derived attribute, and "BS" an subcategory-defined base attribute. "AVA" indicates an "action-valued" attribute whose value is an executable action, and "AEVA" indicates an "activity-valued" attribute whose value refers to other process entity.

There exist sequence relationships among design activities of process entity categories and sequence flow in process entity categories are also included in Figs. 4, 5, and 6. The circles with the letters, I and C, are used to represent the initiation and completion states of the process entities. The sequence in a process entity starts with I, follows one of the paths indicated by the arrows, and completes at C.

3.1 Process Entities and Sequence Flow in Process Entities at System Level

Fig. 1 has showed an example of a building frame structure at the system level. The main process entity categories associated with the design activities are shown in Fig. 4. Some sequence flow, indicated by the arrows, in each process entity category are involved in the figure. The system selection entity category in Fig. 4(a) describes the generation, comparison, and selection of a number of alternative solutions for the system of a building frame structure. The formulation attribute assembles the design requirements of the system, which include architecturally opaque planer regions, no column location, and an open space in Fig. 1(a). Then two alternative system solutions in Figs. 1(b) and 1(c) are created. The alternative development attribute in Fig. 4(a) develops each alternative solution. The alternative solutions are ordered according to the qualitative measures of merit, and system alternative 1 in Fig. 1(b) is selected as the best solution for the system. The selected alternative completion attribute completes the development of the selected alternative solution.

The value set of the selected alternative completion attribute of the system selection entity category shown in Fig. 4(a) refers to the system development entity category in Fig. 4(b). The system development entity category describes the development of a proposed solution to a system design problem. The layout development attribute describes the layout of the components of the system alternative solution. The decomposition development attribute describes the decomposition of the system alternative solution into components. System alternative 1 in Fig. 1 (b) is decomposed into frame subsystems, including the frame in Fig. 2(a), floor subsystems, and columns as components of the system. The detailed development attribute describes the

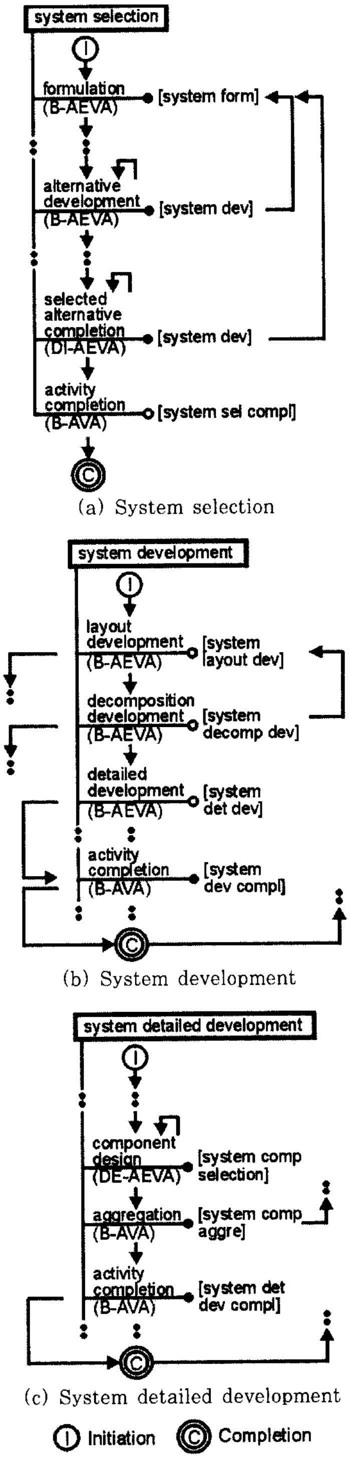


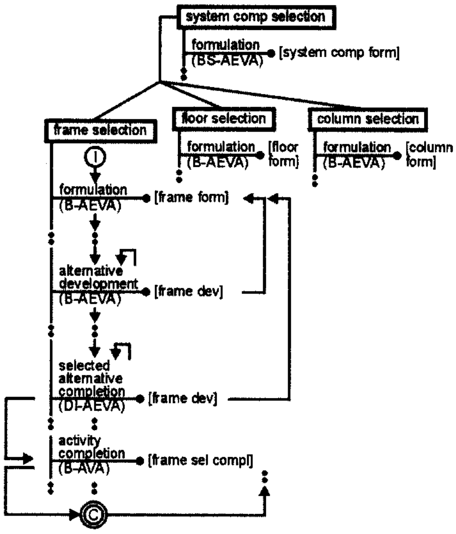
Fig. 4 Process entity categories at system level

detailed development of the system alternative solution. The value set of the detailed development attribute refers to the system detailed development entity category in Fig. 4(c). The component design attribute involves the design of the components of the system. The aggregation attribute collects and combines the design results for the system components into a system solution, which is system alternative 1 in Fig. 1(b).

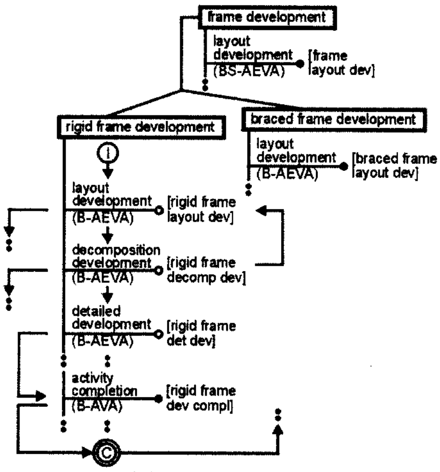
3.2 Process Entities and Sequence Flow in Process Entities at Subsystem Level

The value set of the component design attribute of the system detailed development entity category shown in Fig. 4(c) refers to the system component selection entity category in Fig. 5(a). The system component selection entity category is a generalization of the frame selection, the floor selection, and the column selection entity categories.[11),12)] The frame selection entity category in Fig. 5(a) and other process entity categories in Figs. 5(b) and 5(c) describe the design activities at the subsystem level in Fig. 2. The process entity categories in Fig. 5 involve some sequence flow indicated by the arrows.

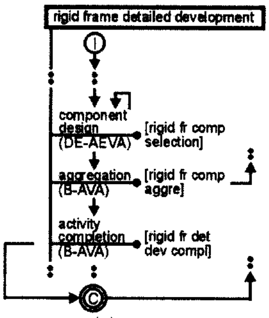
The frame selection entity category in Fig. 5(a) describes the generation, comparison, and selection of a number of alternative solutions for a frame subsystem of a building frame structure. The formulation attribute of the frame selection entity category assembles the design requirements of the frame subsystem, which include the loads applied to the frame in Fig. 2(a). Then two alternative frame solutions in Figs. 2(b) and (c) are created. The alternative development attribute in Fig. 5(a) develops each alternative solution. The alternative solutions are ordered according to the qualitative measures of merit, and frame alternative 1 in Fig. 2(b) is selected as the best solution of the frame subsystem. The selected alternative completion attribute completes the



(a) Frame selection



(b) Rigid frame development



(c) Rigid frame detailed development

Ⓘ Initiation Ⓒ Completion

Fig. 5 Process entity categories at subsystem level development of the selected alternative solution.

The value set of the selected alternative completion attribute of the frame selection entity category shown in Fig. 5(a) refers to the frame development entity category in Fig. 5(b). The frame development entity category describes the development of a proposed solution to a frame subsystem design problem. The frame development entity category is a generalization of the rigid frame and the braced frame development entity categories.[11),12)] The rigid frame development entity category describes the development of a rigid frame solution to a frame subsystem design problem. The layout development attribute of the rigid frame development entity category describes the layout of the components of the rigid frame subsystem solution. The decomposition development attribute describes the decomposition of the rigid frame subsystem alternative solution into components. Frame alternative 1 in Fig. 2(b) is decomposed into beams and columns in Fig. 3(a). The detailed development attribute describes the detailed development of the rigid frame subsystem solution.

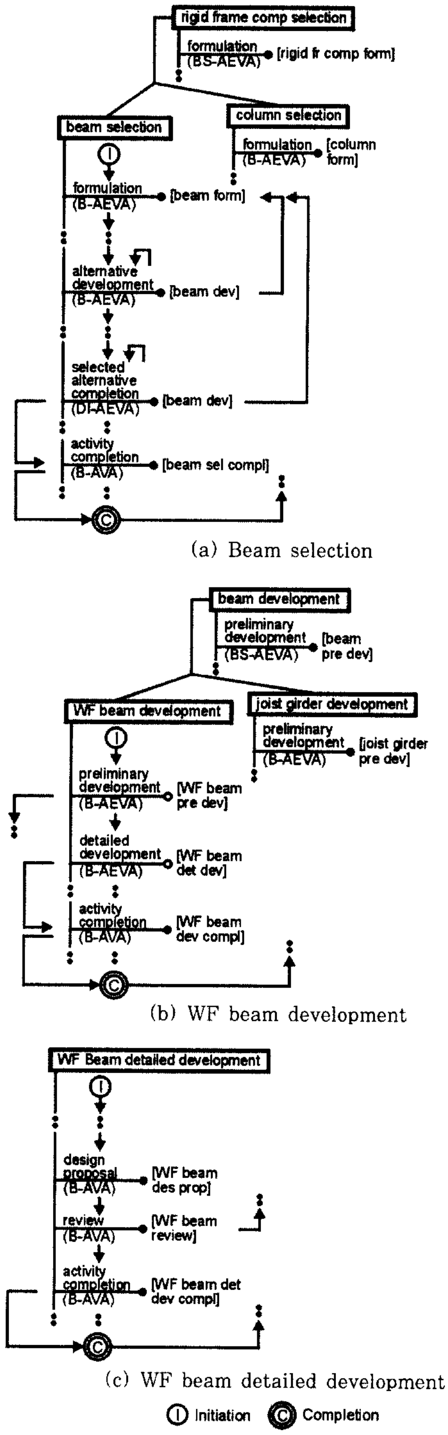
The value set of the detailed development attribute of the rigid frame development entity category shown in Fig. 5(b) refers to the rigid frame detailed development entity category in Fig. 5(c). The component design attribute involves the design of the components of the rigid frame subsystem. The aggregation attribute collects and combines the design results for the components into a frame subsystem solution, which is frame alternative 1 in Fig. 2(b).

3.3 Process Entities and Sequence Flow in Process Entities at Basic Component Level

The value set of the component design attribute of the rigid frame detailed development entity category shown in Fig. 5(c) refers to the rigid frame component selection entity category in Fig. 6(a). The rigid frame component selection entity category is a generalization of the beam selection and the column selection entity categories.[11),12)] The beam selection entity category in Fig. 6(a) and other process entity categories in Figs. 6(b) and 6(c) describe the design activities for a beam at the basic component level in Fig. 2. The process entity categories in Fig. 6 involve some sequence flow indicated by the arrows.

The beam selection entity category in Fig. 6(a) describes the generation, comparison, and selection of a number of alternative solutions for a beam. The formulation attribute of the beam selection entity category assembles the design requirements of the beam, which includes the loads applied to the beam in Fig. 3(a). Then two alternative beam solutions, WF beam 1 in Fig. 3(b) and joist girder 1 in Fig. 3(c), are created. The alternative development attribute in Fig. 6(a) develops each alternative solution. The alternative solutions are ordered according to the qualitative measures of merit, and WF beam 1 in Fig. 3(b) is selected as the best solution for the beam. The selected alternative completion attribute completes the development of the selected alternative solution.

The value set of the selected alternative completion attribute of the beam selection entity category shown in Fig. 6(a) refers to the beam development entity category in Fig. 6(b). The beam development entity category describes the development of a proposed solution to a beam design problem. The beam development entity category is a generalization of the WF beam and the joist girder development entity categories,[11),12)] and the WF beam development entity category describes the development of a WF beam solution to a beam design problem. The preliminary development attribute of the WF beam development entity category describes



(a) Beam selection

(b) WF beam development

(c) WF beam detailed development

Fig. 6 Process entity categories at basic component level

the preliminary development of a WF beam solution. The detailed development attribute describes the detailed development of a WF beam solution.

The value set of the detailed development attribute of the WF beam development entity category shown in Fig. 6(b) refers to the WF beam detailed development entity category in Fig. 6(c). The design proposal attribute proposes values for the design variables(e.g., material and geometry) of the WF beam. The review attribute determines whether the values for the design variables satisfy the design requirements.

4. Overall Process Management

The process entity categories and the sequences in each process entity category in an entity-based integrated design model for a building frame structure have been described. The sequences among the different process entity categories are discussed in this section, and then an approach to ensure completion of the overall sequence of all process entity categories at the system, sub-system, and basic component levels is discussed.

4.1 Sequence Flow Among Process Entity Categories

Figs. 4, 5, and 6 have showed the possible sequence flow among the attributes of the process entity categories. A sequence of attributes in each entity category follows one of the possible paths indicated by arrows. The initiation of an attribute, whose value refers to another process entity, leads to a sequence of attributes in the other process entity. For example, in Fig. 4(a), the system selection entity category includes the alternative development attribute. The value set of the alternative development attribute refers to the system development entity category shown in Fig. 4(b). The initiation of an alternative development attribute of the system selection entity category leads to the initiation of a system development entity from the system development entity category. This results in a sequence of attributes within this system development entity. Similarly, the initiation of a detailed development attribute of the system development entity category shown in Fig. 4(b) leads to the initiation of a system detailed development entity from the system detailed development entity category shown in Fig. 4(c).

When a sequence in a process entity reaches the suspension or the completion state, the sequence in the process entity is suspended or completed. Then the sequence returns to the process entity that initiated the process entity in which the sequence has been suspended or completed. For example, a system detailed development entity(from the category shown in Fig. 4(c)) is initiated by the initiation of a detailed development attribute of a system development entity(from the category shown in Fig. 4(b)). When the sequence in the system detailed development entity(Fig. 4(c)) reaches the completion state, the sequence returns to the system development entity(Fig. 4(b)). Similarly a system development entity(from the category shown in Fig. 4(b)) is initiated by the initiation of an alternative development attribute of a system selection entity(from the category shown in Fig. 4(a)). When the sequence in the system development entity(Fig. 4(b)) reaches the suspension or the completion state, the sequence returns to the system selection entity(Fig. 4(a)).

4.2 Sequence Completion State in Process Entity Category

A design sequence in a process entity is suspended or completed at the suspension or the completion state, and then the sequence returns to the entity from which the sequence was previously initiated. There is a difference between the suspension and completion states. When a sequence in a process entity reaches the suspension state, it means that some of the design activities of the process entity category have been initiated, but the process entity is not completed. When a sequence reaches the completion state, it means that all design activities of the process entity have been initiated. For example, a sequence in a system development entity (Fig. 4(b)) can reach the suspension state after a layout development attribute has been initiated, or after a layout development attribute and a decomposition attribute have been initiated. The completion state can be reached only after a layout development, a decomposition development, and a detailed development attribute have been initiated.

The completion state in a process entity identifies when the design activities described by the process entity category are all initiated. For example, when a design sequence returns from a system development entity(Fig. 4(b)) to a system selection entity(Fig. 4(a)), the designer may check if all of the design activities of the system development entity category have been initiated by checking the state of the system development entity. If the system development entity is at the suspension state, all of the design activities of the system development entity category have not been initiated. If the system development entity is at the completion state, all of the design activities of the system development entity category have been initiated.

4.3 Sequence Completion States in Process Entity Categories

The completion of the overall design process in an entity-based integrated design model can be confirmed by inspecting the completion states of the process entities in the model. The completion states of the selection and development entity categories at the system, subsystem, and basic component levels are shown in Figs. 7, 8, and 9.

The overall sequence for the design of a building frame structure starts and ends in a system selection entity (Fig. 7(a)). The sequence in the system selection entity is at the completion state at the end of the design process. The completion state in a system selection entity is reached after a selected alternative development attribute and an activity completion attribute have been initiated. The initiation of a selected alternative completion attribute leads to the initiation of a system development entity(Fig. 7(b)), because the value set of this attribute refers to the system development entity category. An activity completion attribute(Fig. 7(a)) can be initiated only after the sequence in the system development entity has reached the completion state. When the sequence in a system selection entity reaches at the completion state, it confirms that the sequence in a system development entity has reached the completion state.

The completion state in a system development entity(Fig. 7(b)) is reached after a detailed development attribute and an activity completion attribute have been initiated. The initiation of a detailed development attribute leads to the initiation of a system detailed development entity(Fig. 7(c)), because the value set of this attribute refers to the system detailed development entity category. When the sequence in a system development entity reaches the completion state, it confirms that the sequence in the corresponding system detailed development entity has reached the completion state.

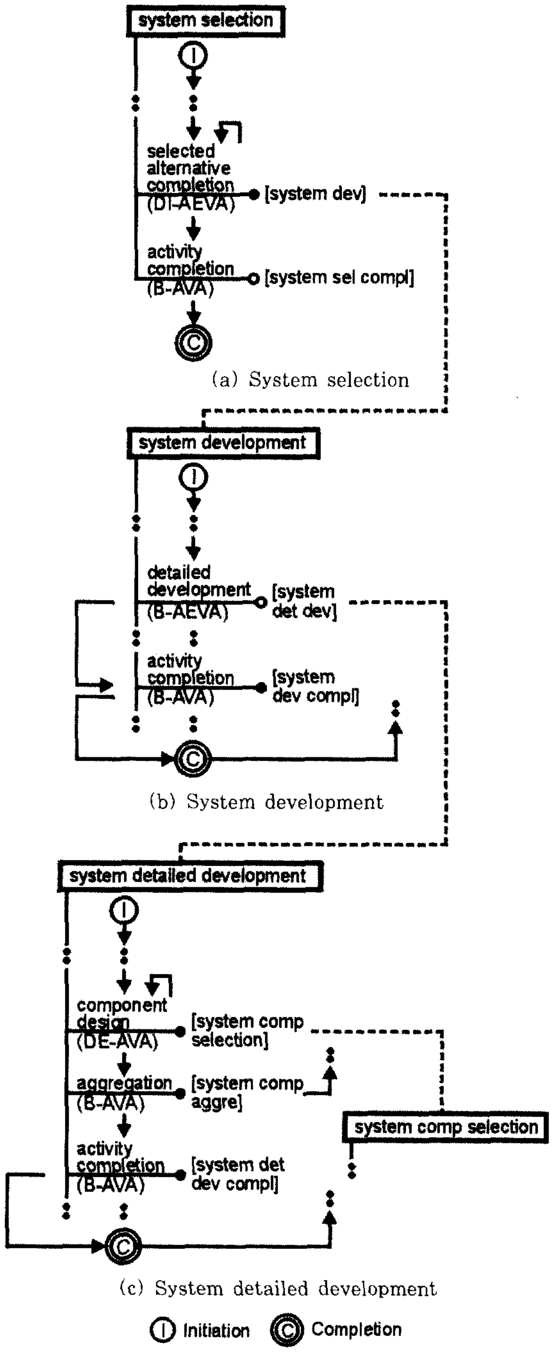


Fig. 7 Completion states at system level

The completion state in a system detailed development entity(Fig. 7(c)) is reached after a component design attribute, an aggregation attribute, and an activity completion attribute have been initiated. The value set of the component design attribute refers to the system component selection entity category which is a generalization of the frame subsystem, floor subsystem, and column selection entity categories.[11),12)] The initiation of a component design attribute leads to the initiation of the frame subsystems, floor subsystems, and column selection entities. After these selection entities have reached the completion states, an activity completion attribute can be initiated in the system detailed development entity. Therefore, when the sequence in a system detailed development entity has reached the completion state, it confirms that the sequences in the corresponding frame subsystem, floor subsystem, and column selection entities have reached the completion states.

The completion state in a frame selection entity(Fig. 8(a)) is reached after a selected alternative development attribute and an activity completion attribute have been initiated. The value set of the selected alternative completion attribute refers to the frame development entity category which is a generalization of the rigid frame development and the braced frame development entity categories.[11),12)] When the selected alternative solution is a rigid frame, the initiation of a selected alternative development attribute leads to initiation of a rigid frame development entity(Fig. 8(b)). An activity completion attribute(Fig. 8(a)) can be initiated only after the sequence in the rigid frame development entity has reached the completion state. When the sequence in a frame selection entity reaches the completion state, it confirms that the sequence in a frame development entity (in this example, a rigid frame development entity) has reached the completion state.

The completion state in a rigid frame development entity(Fig. 8(b)) is reached after a detailed development attribute and an activity completion attribute have been initiated. The initiation of a detailed development attribute leads to the initiation of the rigid frame detailed development entity(Fig. 8(c)), because the value set of this

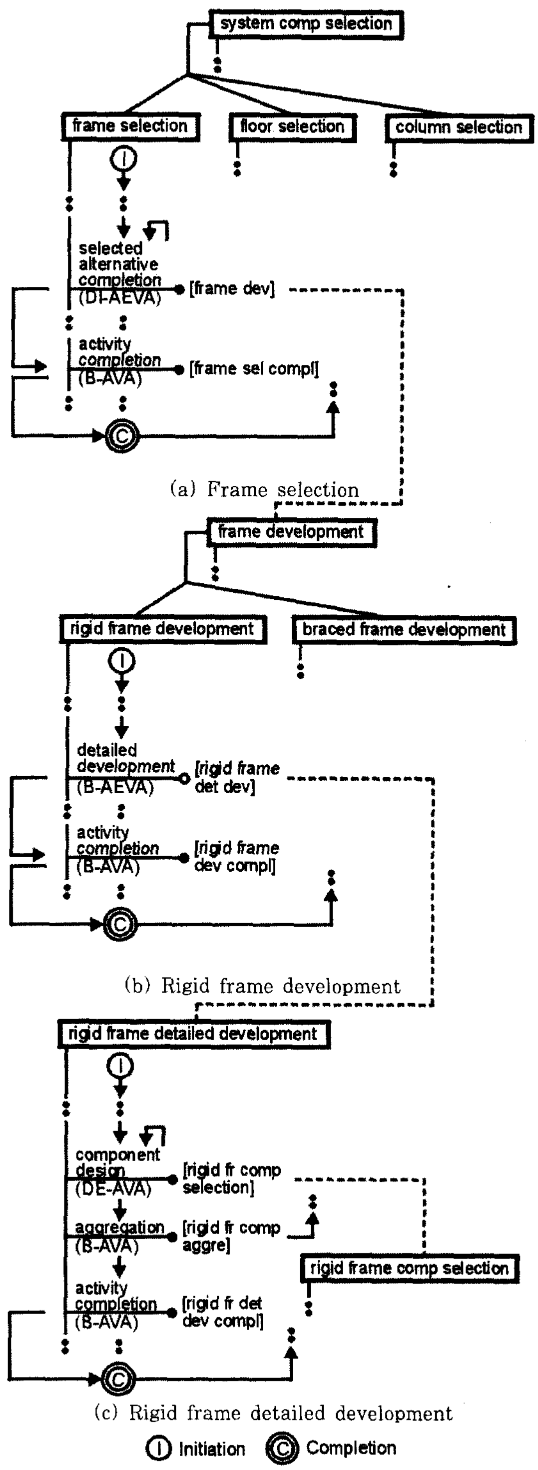


Fig. 8 Completion states at subsystem level

attribute refers to the rigid frame detailed development entity category. When the sequence in a rigid frame development entity reaches the completion state, it confirms that the sequence in the corresponding rigid frame detailed development entity has reached the completion state.

The completion state in a rigid frame detailed development entity(Fig. 8(c)) is reached after a component design attribute, an aggregation attribute, and an activity completion attribute have been initiated. The value set of the component design attribute refers to the rigid frame component selection entity category which is a generalization of the beam selection and the column selection entity categories.[11),12)] The initiation of a component design attribute leads to the initiation of beam and column selection entities. After these selection entities have reached the completion states, an activity completion attribute in the rigid frame detailed development entity can be initiated. Therefore, when the sequence in a rigid frame detailed development entity has reached the completion state, it confirms that the sequences in the corresponding beam and selection entities have reached the completion states.

The completion state in a beam selection entity (Fig. 9(a)) is reached after a selected alternative development attribute and an activity completion attribute have been initiated. The value set of the selected alternative completion attribute refers to the beam development entity category which is a generalization of the WF beam development and the joist girder development entity categories.[11),12)] When the selected alternative solution is a WF beam, the initiation of a selected alternative development attribute leads to initiation of a WF beam development entity(Fig. 9(b)). An activity completion attribute (Fig. 9(a)) can be initiated only after the sequence in the WF beam development entity has

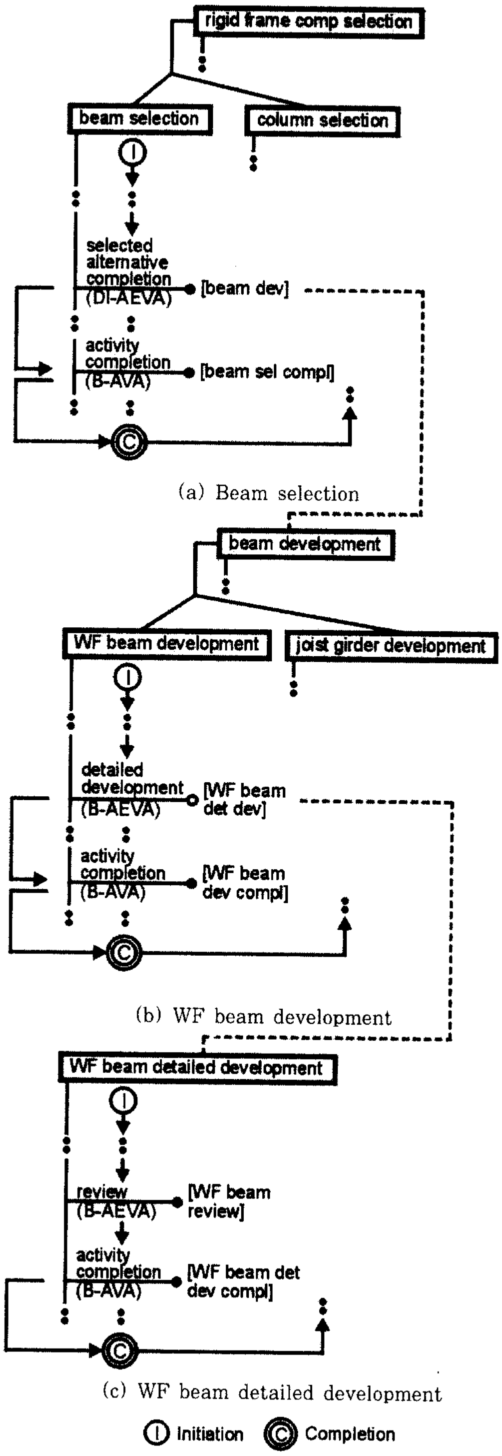


Fig. 9 Completion states at basic component level

reached the completion state. When the sequence in a beam selection entity reaches the completion state, it confirms that the sequence in a beam development entity(in this example, a WF beam development entity) has reached the completion state.

The completion state in the WF beam development entity(Fig. 9(b)) is reached after a detailed development attribute and an activity completion attribute have been initiated. The initiation of a detailed development attribute leads to the initiation of the WF beam detailed development entity(Fig. 9(c)), because the value set of this attribute refers to the WF beam detailed development entity category. When the sequence in a WF beam development entity reaches the completion state, it confirms that the sequence in the corresponding WF beam detailed development entity has reached the completion state. The completion state in a WF beam detailed development entity(Fig. 9(c)) is reached after a review attribute and an activity completion attribute have been initiated.

The completion of the overall sequence for the design of a building frame structure requires the completion of the all required design activities. In an entity-based integrated design model, the completion of the overall sequence requires the completion of the selection and development for the system, the components of the system (e.g., frame and floor subsystems), and the components of the subsystems(e.g., beams and columns). This means that the sequences in the process entities for the selection and development of the system, the components of the system, and the components of the subsystems should be at the completion states. The overall sequence starts and ends in a system selection entity. When a sequence in the system selection entity reaches the completion state, it confirms that sequences are at the completion states in the selection and development entity for the system, the components of the system, and the components of the subsystems, as shown in Figs. 7, 8, and 9. Therefore, the completion of the overall sequence for the design of a building frame structure is confirmed.

5. Conclusions

This paper described the overall process management in an entity-based integrated design model for a building frame structure. The design activities for a building frame structure at the system, subsystem, and basic component levels are represented using process entity categories in an entity-based integrated design model. There exists possible sequence flow among the attributes of a process entity category. The sequence in a process entity reaches its completion state when all design activities of the process entity category have been initiated. An entity-based integrated design model for a building frame structure includes a number of process entity categories. The system selection entity category includes the system development entity category, which includes the system detailed development entity category, which includes other process entity categories, and so on. When the sequence in a system selection entity category reaches the completion state, it confirms that the sequences of all of the related process entity categories at the system, subsystem, and basic component levels have reached the completion states.

The completion of the overall sequence for the design of a building frame structure is confirmed by introducing the concepts for the completion states where can be reached only after all design activities have been initiated, with the aids of the organization of an entity-based integrated design model such as the aggregation and the generalization of design activities. However, the underlying concepts for the completion of the overall sequence can be applicable to other kinds of design process models. Since the concepts are explained using an entity-based integrated design model whose organization is similar to those of objected-oriented programming, the concepts can be implemented in the design model using object-oriented programming languages such as C++ and Smalltalk, or using other developing tools. For actual implementation, the corresponding design information should be also considered with the concepts for the completion of the overall sequence. Once the concepts are implemented, they can be used for evaluating a computer system which is implemented using the design process model. If the completion of the overall sequence is identified but the actual design results are not obtained, then the design process model needs to be modified and the computer system model needs to be reimplemented. There is a gap between the actual design process and the design process in a design process model. The research efforts on design process models have tried to capture and represent the actual design process as completely as possible. This paper is one of the efforts to diminish the gap for developing an integrated design system that fully supports structural engineers.

Acknowledgments

This research was supported by the Engineering Research Center for Advanced Technology for Large Structural Systems(ATLSS) at Lehigh University and by the Advanced Structure Research Station(STRESS) at Hanyang University. The opinions, findings, and conclusions in the paper are those of the author and do not necessarily reflect the views of the ATLSS Center and the STRESS Center.

References

1. Simon, H. A., "The Structure of Ill-Structured Problems," *Artificial Intelligence*, Vol. 4, 1973, pp.181~265
2. Sanvido, V. E., Kumara, S., Ham, I., "A Top-Down Approach to Integrating the Building Process," *Engineering with Computers*, Vol. 5, 1989, pp.91~103
3. Sause, R. and Powell, G. H., "A Design Process Model for Computer Integrated Structural Engineering," *Engineering with Computers*, Vol. 6, 1990, pp.129~143
4. Baugh. J. W. and Chadha, H. S., "Semantic Validation of Product and Process Models," *Journal of Computing in Civil Engineering*, ASCE, Vol. 11, No. 1, 1997, pp.26~36
5. Park, H. and Cutkosky, M., "Framework for Modeling Dependencies in Collaborative Engineering Processes," *Research in Engineering Design*, Vol. 11, 1999, pp.84~102
6. Rivard, H. and Fenves, S. J., "A Representation for Conceptual Design of Buildings," *Journal of Computing in Civil Engineering*, ASCE, Vol. 14, No. 3, 2000, pp.151~159
7. Lee, C.-H., Sause, R., and Hong, N. K., "Overview of Entity-Based Integrated Design Product and Process Models," *Advances in Engineering Software*, Elsevier Science, Oxford, United Kingdom, Vol. 29, No. 10, 1998, pp.809~823
8. Lee, C.-H. and Sause, R., "Sequence Control for Integrated Structural Design Models," *Journal of Computing in Civil Engineering*, ASCE, Vol. 10, No. 3, 1996, pp.213~225
9. Lee, C.-H., "Representation of System Design Information for Developing Computer-Aided Structural Design System," *Journal of the Computational Structural Engineering Institute of Korea*, The Computational Structural Institute of Korea, Vol. 11, No. 4, 1998, pp.371~382
10. Rigopopulus, D. R. and Oppenheim, I. J.,

"Intelligent Objects for Synthesis of Structural Systems," *Journal of Computing in Civil Engineering*, ASCE, Vol. 6, No. 3, 1992, pp. 266~281

11. Hong, N. K. and Sause, R., "Concepts and Notation for Integrated Structural Design: Product and Process Models," ATLSS Report No. 94-13, Lehigh University, Bethlehem, PA, USA, 1994
12. Hong, N. K. and Hong, S., "Entity-Based Models for Computer-Aided Design Systems," *Journal of Computing in Civil Engineering*, ASCE, Vol. 12, No. 1, 1998, pp.1~12