6. ROAD BRIDGE PLANNING AND DESIGN SYSTEM BASED ON THE OBJECT-ORIENTED APPROACH

Chiyenori INOUE1, Yoshito ITOH2

Abstract: An object-oriented approach is applied to the road bridge planning and design in this research because it is difficult to model the whole process of road bridge planning and design using the conventional structured systems development method. In this paper, problems that arise from applying the object-oriented approach to it are pointed out, and the outline of the prototype system is described.

Keywords: Bridge, System of Planning and Design, Object-Oriented Approach

1. Introduction

With the recent development of computers, several systems that support planning and design processes of road bridges have been developed. These systems improved the processes on a large scale but they deal with only the simple routine works such as structural analysis, structural drawing, quantity calculation, and so on. They have not been able to accomplish the computer-aided design. Though these systems for designing superstructures, substructures or foundations of a bridge have been developed, true interactive systems for a whole bridge are still very few. This problem implies that there is no established modeling method which unifies the planning design processes of a whole bridge.

A feature of the conventional system development method that originates from Structured Programming is to subdivide a complicated and abstract function into simple and concrete functions hierarchically. On the other hand, a design work of each bridge component can be subdivided into simple routine work. Therefore the method is certainly suitable for developing the design system.

However, the method does not cover sufficient ground for modeling the planning and design processes of a whole bridge. The processes consist of a lot of objects such as superstructures, substructures, foundations, roads and topography. A bridge planning and design project is developed by manipulating the objects and the relationships among them. Therefore a system development method for modeling the objects and the relationships is needed but the conventional method does not provide an architecture to model them. On the other hand, almost all object-oriented approaches provide the architecture. In this research, an object-oriented approach based on the Object Modeling Technique (OMT) is employed in the development. OMT was developed by a team led by James Rumbaugh at General Electric Research and Development Center.

2. Object-Oriented Bridge Planning and Design

This chapter presents main concepts of object-oriented approach used to model the planning and design process of a whole bridge in this research. Simple examples are used in the following description to explain easily.

2.1 Object-Based Modeling and Class-Based Modeling

The object-based modeling could be divided into two parts, encapsulation and information hiding as shown in Fig.1. Encapsulation is an architecture to combine data and functions associated with an object into a capsule. The data of the encapsulated object describe its attributes and the functions describe its behavior. For example, an encapsulated abutment of a bridge has data such as height, width, etc., and functions such as stability calculation, stress calculation, etc. By encapsulating the

1: Member of JSCE, Visiting Research Associate, Nagoya University, E-mail: inoue@fureai.or.jp
2: Fellow of JSCE, Dr.Eng., Nagoya University, E-mail: itoh@cirsc.nagoya-u.ac.jp

— 217 —
The inheritance is an architecture in which a child class object shares the data and the functions of its parent class object. The child class object may add its own data and functions for having differences among its parents. The parent class object is called a super-class and the child class object is called a sub-class. For example, because abutments of all the structure types such as a reversed T-type abutment and a rigid frame abutment can be classified into an abstract abutment, all the abutment classes can be generated by inheriting an abstract abutment class as shown in Fig.2.

2.3 Association and Polymorphism

Association is an architecture for associating an object class with other object classes. For example, an association between a superstructure as a super-class and an abutment as a sub-class describes that the abutment supports the superstructure as shown in Fig.3.

Fig.3 Association

Polymorphism is an architecture in which all subclasses inheriting a super-class are treated as the super-class. For example, if the association in which the abutment supports the superstructure exists as shown in Fig.4, all the subclass abutments can support all the subclass superstructures.

Fig.4 Polymorphism

2.4 Aggregation

Aggregation is a specialized association in which an object class includes other object classes as parts. For example, an object class of road bridge design includes superstructures, substructures, foundations, roads, and topography. Added to this, an object class of bridge includes superstructures, substructures and foundations. By the hierarchal structure of objects like this, it could be emphasized that an operation to a system is coped with like a waterfall type.

Fig.2 Inheritance

Fig.1 Object-Based Modeling and Class-Based Modeling

data and the functions into a capsule like this, it could simplify most of the complicated models in the real system. Information hiding is an architecture to isolate the internal aspects of an object from others. The data of an object could be accessed only through the allowed internal functions. By this architecture, the internal data could be protected from illegal accesses including bugs.

The class-based modeling is an architecture to generate object instances using an object as a pattern or a class. For example, an object-class of an abutment generates object-instances which have structure dimensions of real abutments.

2.2 Inheritance

Fig.2 Inheritance

The inheritance is an architecture in which a child class object shares the data and the functions of its parent class object. The child class object may add its own data and functions for having differences among its parents. The parent class object is called a super-class and the child class object is called a sub-class. For example, because abutments of all the structure types such as a reversed T-type abutment and a rigid frame abutment can be classified into an abstract abutment, all the abutment classes can be generated by inheriting an abstract abutment class as shown in Fig.2.

2.3 Association and Polymorphism

Association is an architecture for associating an object class with other object classes. For example, an association between a superstructure as a super-class and an abutment as a sub-class describes that the abutment supports the superstructure as shown in Fig.3.

Fig.3 Association

Polymorphism is an architecture in which all subclasses inheriting a super-class are treated as the super-class. For example, if the association in which the abutment supports the superstructure exists as shown in Fig.4, all the subclass abutments can support all the subclass superstructures.

Fig.4 Polymorphism

2.4 Aggregation

Aggregation is a specialized association in which an object class includes other object classes as parts. For example, an object class of road bridge design includes superstructures, substructures, foundations, roads, and topography. Added to this, an object class of bridge includes superstructures, substructures and foundations. By the hierarchal structure of objects like this, it could be emphasized that an operation to a system is coped with like a waterfall type.
3. Domain Analysis

The basic flow of road bridge planning and design is shown in Fig. 5 although it may differ in each public corporation according to its own project process. In this flow, Alignment Planning and Road Planning are mainly developed with the administrative judgement, and Preliminary Design and Detailed Design are developed with the technical judgement. The purpose of this research is to model Preliminary Design.

The basic flow of Preliminary Design is shown in Fig. 6 including three stages: (1) Preliminary Bridge Type Selection-Phase1, (2) Inferred Design and Preliminary Bridge Type Selection-Phase2, and (3) Comparative Design and Optimal Selection of Bridge Type. In these phases, an engineer has to consider all possible bridge types and span arrangements to decide the most suitable combination.

The first issue of bridge type selection is span arrangements. By considering how to arrange substructures to erection space between a topography and a road, at most, ten to fifteen candidates for bridge types to examine are selected in Preliminary Bridge Type Selection-Phase1.

In Inferred Design and Preliminary Bridge Type Selection-Phase2, outlines of these plans are inferred and simply compared on some viewpoints such as cost, maintenance, landscape, etc. As a result, three to five plans to examine are selected. Then, sizes of primary members of the bridge structure and total cost of the bridge construction are only inferred with the accumulation of the design data because most of experts think that realizing an image of the whole bridge structure is more important than its detail. In Comparative Design, calculations for design and quantity survey are actually done. To select the most suitable plan in the following phase, construction cost of each plan that is correct as much as possible, must be grasped in this phase. Therefore, in this phase, the engineer calculates to decide sizes of primary members of bridge structure of each plan, and to grasp the cost. In Optimal Selection of Bridge Type, each plan is compared on all viewpoints in detail, and a most suitable plan is selected. More often than not a most important viewpoint for bridge type selection is the construction cost because a bridge is constructed for public benefit. However other viewpoints should be also considered in this system development because they have been beginning to become important by degrees.

These stages consist of simplified procedures of Detailed Design. All plans proposed in Preliminary Bridge Type Selection-Phase1 are investigated while passing through each phase. The plans that are not suited for the project must be excluded at each step. The reason that Preliminary Design is such a spiral structure with multi-layers, is that it is actually impossible to investigate all plans by calculations. Instead it uses to think that an expert might get a good result without calculations. However, on such result, ability and arbitrary judgement of each engineer influence too much. So it is not desirable on the publicity of the bridge planning and design project. Generally, the precision of Preliminary Design becomes
problem, and it is also pointed out by each public corporation. Therefore, all plans proposed in Preliminary Bridge Type Selection-Phase1 should be investigated by designs and quantity surveys based on calculations. In Comparative Design, if bridge type and span arrangement can be changed easily and interactively, Inferred Design and Preliminary Bridge Type Selection-Phase2 should be omitted in Preliminary Design.

4. System Analysis

In general, object-oriented approach models can be classified into three types: the object model, the dynamic model and the functional model. The object model is the most important of the three models and the system analysis is mainly developed to realize the real world objects and relationships among them.

The dynamic aspect does not influence the system development so much because this system is not the real-time system. Therefore, the dynamic model excludes from this system analysis. The functional model is explained as Comparative Design and Bridge Type Selection-Phase2 as shown in Fig. 7. In this system analysis, only the object model of the bridge planning and design system is adopted.

4.1 Modeling Objects

Important nouns in the application domain often correspond to objects of the object-oriented approach. So such nouns as candidate objects should be extracted from documents of the domain analysis. In the preceding chapter, this research describes that ten to fifteen candidates for bridge types to examine are selected by considering how to arrange substructures to erection space between topography and road. In the statement, important nouns would be Bridge, Topography, and Road. Actually, regardless of the object-oriented approach, an engineer who plans and designs road bridges does realize that these nouns are key words of importance. For example, when the engineer looks over a bridge side view as shown in Fig. 8, he/she pays attention to the bridge structure, the road alignment, and the topography. Therefore, important objects would be these three key words in this research as shown in Fig. 9.

Fig. 7 Comparative Design and Optimal Selection of Bridge Type

Fig. 8 Example of Road Bridge

Fig. 9 Main Objects of Road Bridge
These objects consist of a large number of objects. Bridge consists of Superstructure, Substructure, and Foundation. Therefore, important objects of Bridge object would be these key words. Similarly, Road object consists of Horizontal Curve Alignment object, Vertical Curve Alignment object, and Composition of Cross Section object. However this research analyzes Road object in the range which is necessary to analyze Bridge object, because its main purpose is to model the objects of the bridge planning and design system. Topography is only defined as an object to return a ground height and a geological column corresponded to an optional point on the plan, and this research does not analyze it for the same reason as Road object.

4.2 Modeling Associations

Stative verbs or verb phrases in the application domain often correspond to associations among objects. So such verbs as candidate associations should be extracted from documents of the domain analysis. This is very useful for a system engineer who is not an expert of the domain. However it is not necessary for civil engineers who plan and design bridges to use it, because they already realize true associations among objects of the domain consciously or unconsciously. Therefore, the following analysis result seen from the viewpoint of a bridge planner is explained without detailed explanation based on general object-oriented approaches.

For example, associations among objects included in the Bridge object are shown in Fig. 10. In this association model, there is a significant point that Node object as virtual object exists between every two objects. It improves independence of all objects, and enables plug-and-play of all objects. By providing a broker object like Node object among primary objects, most troubles of associations would be solved easily. Node object not only is a broker object of association inside Bridge object, but plays the same role between other object categories. In addition, Node object plays a role as receptor for events of a system operator.

In the example shown in Fig. 11, the superstructure type is a girder bridge. In case that the superstructure type is integrated with substructure like a rigid-frame bridge, it is considered that a Substructure object moves from Substructure object category to Superstructure object category.
4.3 Modeling Inheritance Structures and Association Structures

An engineer unfamiliar with the object-oriented approach tends to use the inheritance structure too many for confiding in its absolute possibility. However, the inheritance is a mere modeling technique that consists of primitive genetics and abstraction based on vague feeling of the human being. Without both of the complete dominance and the incomplete dominance, this primitive genetics consists. It used to think that the trouble of two opposite genes does not appear, because there are two mechanisms: (1) by the complete dominance, one of two is hidden, and (2) by the incomplete dominance, middle character of two is appeared. In virtual inheritances of the real world, such mechanisms exist. For example, though an escalator has both features of stairs and belt conveyor, it does not have all of both features. The features which are necessary for an escalator and are not conflicted, are selected.

Because the inheritance of object-oriented approach does not involve such mechanisms, we should pay a prudent attention to eliminate conflicts among features of an inheritance structure. In addition, because the use of multiple inheritance causes such conflicts, we should especially avoid using it. However, as a rule, objects of the real world have many aspects, so a system should be often developed from several viewpoints.

For example, there are some primary viewpoints to divide superstructures: (1) topography to erect a bridge such as a river or a city area, (2) bridge material such as steel or prestressed concrete, (3) bridge type such as girder bridge or rigid-frame bridge, and (4) bridge purpose such as a road or a railway or a aqueduct (though the purpose of this research is a system development of road bridge, we should consider its viewpoint for improving generality of this research.) On the system development, there are many viewpoints in spite of considering only superstructure. If such a system based on many viewpoints was modeled by multiple inheritance, it would be too difficult to develop the class library and to maintain the system.

The system analysis in this paper is based on the bridge type only as a viewpoint to classify superstructures in order to avoid such problems, though Substructure object and Foundation object are modeled in a consistent manner. Other viewpoints are defined as indirect
attributes of Bridge object and the class library structure is a single inheritance. By realizing associations from the object to other objects, it would possess these attributes indirectly. The indirect attributes are changed by changing an object concerned with the Bridge object: (1) topography to erect a bridge can be changed by changing Topography object, (2) bridge material can be changed by changing Bridge Material object, (3) bridge type can be changed by changing Bridge Type object, and (4) bridge purpose can be changed by changing from Road object to such Railway object or Aqueduct object. It is essential to learn the limit of capability of inheritance structure and to restrict using it. Instead, we are much more interested in the challenge of investigating association structure. Fig. 12 shows Bridge object described in the graphical notation of the Object Modeling Technique.

The association structure of Road Bridge Planning and Design object is shown in Fig. 13. This association structure has a remarkable point that is a virtual object to aggregate Bridge object and Road object. Because this system has to manipulate some bridges and roads, a mechanism to manage them is needed. For example, the system for an over-bridge has to manipulate some roads passed under it, in addition to the road which includes it. Added to this, a planning bridge may pass under some existing bridges. In these cases, the engineer should check clearances between an over-bridge and roads that pass under (or over) it. Because the system development is confused in such circumstances, an object is needed to reveal association between them. Finally, Road object structure is shown in Fig. 13.

5. Development of Prototype System

The Software Engineers Association indicated that about 75% of obstructions in all system developments is caused by the upper processes (system analysis:49%, system design:25%)\(^9\). Even through our experiences in the system developments, it has been clear that the obstructions in the upper processes are more serious to get over. Because the number of steps of the bridge planning system is estimated to be over 1,000,000 steps, a lot of hidden obstructions will appear in the final stage. Therefore, by developing a prototype system from system analysis phase, this research has been trying to expose most of the serious obstructions and to resolve them in advance. This method is called the spiral approach for refining the analysis model repeating alternately both system analysis and implement phases.

By developing the prototype system, some serious obstructions hidden in an initial analysis model are found out, for example, an obstruction occurred by using multiple inheritance to model a bridge with some points of analysis view during the development of the prototype system. This problem was solved by turning its object structures into association and aggregation structures after reconsidering the analysis model.

However, a feedback approach like this is not only an advantage but also a cause of problems. Because it is not clear how many spirals we should repeat in a system analysis, we often lose track of time for refining a model. Therefore, the spiral approach should be rationalized by using object-oriented computer-aided software engineering (OOCASE) tools.

However, our recent investigation indicated that a CASE tool which we required did not exist in the marketplace. Therefore, with manual work, we had to change code of the prototype system every spiral. Because it would not cause only the increase of manpower but also a decline problem of reliability of the system, it was the most serious problem in the development of the prototype system.

We have been contemplating to provide requirements for the development of a new OOCASE tool that we required through the development of this prototype system. Because several OOCASE tools have entered the marketplace, new one we required may be developed in the near future.

A three dimensional graphics of a bridge as an example of output of the prototype system is shown in Fig.14. In General, bridge planners have always needed to figure out their span arrangements and bridge type selections in three dimensions. Since the prototype system can provide three dimensional graphics of a planning bridge quickly it can fulfill their requirement in real time.
6. Conclusions

This paper could be concluded as follows:
1) The prototype system of road bridge planning and design system has been developed using the object-oriented approach.
2) Most of objects and associations among them needed for realizing that a bridge planning system could be shown definitely.
3) By developing a prototype system, serious obstructions in the development of a large and complicated bridge planning system could be resolved in advance.

Reference