An Architecture-Oriented Design Method
For
Interactive Multimedia Systems

Shuh-Ping Sun*、Meng-Chyi Harn**、William S. Chao***

*Professor, Department of Digital Media Design, I-Shou University
(No.1, Sec. 1, Syuecheng Rd., Dashu District, Kaohsiung City 84001, Taiwan, R.O.C.
886-7-6577711 Ext.8451 · spsun@isu.edu.tw)
**Director, C4ISR research center at the Takming University of Science and Technology
***President, Association of Chinese Enterprise Architects

Abstract

In this paper, we propose an architecture-oriented design method for interactive multimedia systems. This design method adopts the structure-behavior coalescence (SBC) architecture as a systems model. SBC architecture design method starts from the preparation phase and then goes through the creative thinking, planning, analysis, and design phases of SBC architecture construction. SBC architecture design method uses six fundamental diagrams to formally design the essence of an interactive multimedia system and its details at the same time. In the planning phase, architecture hierarchy diagram and framework diagram are used. In the analysis phase, component operation diagram and component connection diagram are used. In the design phase, structure-behavior coalescence diagram and interaction flow diagram are used.

With the above six diagrams, we then can effectively design the structure, behavior, and information of interactive multimedia systems; resolve uncertainties and risks caused by those non-architecture-oriented design methods. Overall, SBC architecture design method helps integrate different stakeholders’ works on the same track and unfold the backbone of interactive multimedia systems. The interactive multimedia system design result of SBC architecture can be used as interactive multimedia system blueprints to improve the acceptance and effectiveness of the development of interactive multimedia system.

Keyword: Architecture-Oriented Design Method, SBC Architecture, Interactive Multimedia System
1. Introduction

In general, an interactive multimedia system is exceptionally complex that it includes multiple views such as structure, behavior, and information views. The systems model designs the interactive multimedia system multiple views possibly using two different methods. The first one is the non-architecture-oriented method and the second one is the architecture-oriented method\textsuperscript{1, 6}. Non-architecture-oriented systems model respectively picks a model for each view\textsuperscript{7, 8}. Architecture-oriented systems model, instead of picking many heterogeneous and unrelated models, will use only one single coalescence model\textsuperscript{2, 9}.

An architecture-oriented design method for interactive multimedia systems adopts the SBC architecture\textsuperscript{3, 4, 5} as a systems model. With SBC architecture, we then can effectively design the structure, behavior, and information of interactive multimedia systems; resolve uncertainties and risks caused by those non-architecture-oriented design methods. Overall, SBC architecture design method helps integrate different stakeholders’ works on the same track and unfold the backbone of interactive multimedia systems. The interactive multimedia system design result of SBC architecture can be used as interactive multimedia system blueprints to improve the acceptance and effectiveness of the development of interactive multimedia system.

2. Non-architecture-oriented and Architecture-oriented Systems Models

A systems model is a conceptual system, distinguished from a physical system, used to design either the physical or conceptual systems. A physical system, e.g., house, tree, river, airplane, etc., exists in the physical world. A conceptual system, e.g., symbol, language, diagram, software, virtual reality, thought, etc., exists in the conceptual world.

Figure 1 shows a physical system in which there are two buildings located in the upper left side and right underneath. The upper left building is Seattle Hotel and the right underneath building is Dallas Theater.
To design the physical system in Figure 1, we may then obtain a map as shown in Figure 2. The map is a kind of systems model used to design the physical system.

Besides designing systems in the physical world, a systems model can also design systems in the conceptual world. The conceptual world includes a software system, a virtual reality, or a thought within a person’s mind, etc. Figure 3 demonstrates that a manager is planning a sale task. Planning a sale task, being a thought inside a person’s mind, belongs to the conceptual world.
To design the thought within a person’s mind in Figure 3, we may then use a sale chart as shown in Figure 4. The sale chart is a kind of systems model used to design a person’s thought.

An interactive multimedia system is exceptionally complex that it includes multiple views such as structure, behavior, and information views. The systems model designs the interactive multimedia system multiple views possibly using two different methods. The first one is the non-architecture-oriented method and the second one is the architecture-oriented method.

The non-architecture-oriented method respectively picks a model for each view as shown in Figure 5, the structure view has the structure model; the behavior view has the behavior model; the information view has the information model. These multiple models are heterogeneous and unrelated of each other, thus there is no way to put them into a conformity model.
The architecture-oriented method, instead of picking many heterogeneous and unrelated models, will use only one single coalescence model as shown in Figure 6. The structure, behavior, and information views are all integrated in this multiple view coalescence (MVC) systems model\(^1, 2, 3, 4, 5, 6, 9\).

Figure 5 has many models. Figure 6 has only one model. Comparing Figure 5 with Figure 6, we unquestionably conclude that an integrated, holistic, united, coordinated, coherent, and coalescence model is more favorable than a collection of many heterogeneous and unrelated models.

Since structure and behavior views are the two most prominent ones among multiple views, integrating the structure and behavior views apparently is the best approach of integrating multiple views of a system. In other words, structure-behavior coalescence (SBC) facilitates multiple view coalescence (MVC) as shown in Figure 7. Therefore, we claim that SBC architecture is an architecture-oriented systems model.
3. SBC Architecture Design Method for Interactive Multimedia Systems

SBC architecture design method adopts the SBC architecture as a systems model. SBC architecture design method shall start from the preparation phase and then goes through the creative thinking, planning, analysis, and design phases of SBC architecture construction. Each phase checks with the SBC architecture to make sure the constructed interactive multimedia system is what the users want as shown in Figure 8.

SBC architecture design method uses six fundamental diagrams to formally
design the essence of an interactive multimedia system and its details at the same time. In the planning phase, architecture hierarchy diagram (AHD) and framework diagram (FD) are used. In the analysis phase, component operation diagram (COD) and component connection diagram (CCD) are used. In the design phase, structure-behavior coalescence diagram (SBCD) and interaction flow diagram (IFD) are used.

3.1. Planning Phase

Through the architecture hierarchy diagram, designers shall clearly observe the multi-level decomposition and composition of an interactive multimedia system. As an example, Figure 9 shows that *Multimedia KTV* is composed of *Song_Selection* and *Songs*. *Songs* is composed of *Song_1* and *Song_2*. Among them, *Multimedia KTV* and *Songs* are aggregated systems while *Song_Selection, Song_1* and *Song_2* are non-aggregated systems.

The framework diagram (FD) designs the decomposition and composition of an interactive multimedia system in a multi-layer manner. Only non-aggregated systems will appear in the FD.

As an example, Figure 10 shows the FD of *Multimedia KTV*. In the figure, *Layer_2* contains the component *Song_Selection; Layer_2* contains the components *Song_1* and *Song_2*. 
3.2. Analysis Phase

For an interactive multimedia system, we use component operation diagram (COD) to design all components’ operations. Figure 11 shows the COD of Multimedia KTV. In the figure, component Song Selection has two operations: Select_Song_1 and Select_Song_2; component Song_1 has two operations: Broadcast_Song_1 and Sing_Song_1; component Song Selection has two operations: Broadcast_Song_2 and Sing_Song_2.

We use the component connection diagram (CCD) to design how the components and actors (in the external environment) are connected within an interactive multimedia system. Figure 12 exhibits the CCD of Multimedia KTV.
3.3. Design Phase

In an interactive multimedia system, if the components, and among them and the external environment’s actors to interact, these interactions will lead to the systems behavior. That is, “interaction” plays an important factor in coalescing structures with behaviors for an interactive multimedia system.

We use the structure-behavior coalescence diagram (SBCD) to design how the structure and behavior are integrated within an interactive multimedia system. Figure 13 exhibits the SBCD of Multimedia KTV. In this example, an actor interacting with three components shall represent the overall systems behavior. Interactions among the actor Singer and the components Song_Selection and Song_1 generate the behavior KalaOK_Song_1. Interactions among the actor Singer and the components Song_Selection and Song_2 generate the behavior KalaOK_Song_2.

The overall behavior of an interactive multimedia system is the collection of all of its individual behaviors. All individual behaviors are mutually independent of each other. They tend to be executed concurrently. For example, the overall Multimedia KTV’s behavior includes behaviors KalaOK_Song_1 and KalaOK_Song_2. In other words, the behaviors KalaOK_Song_1 and KalaOK_Song_2 are combined to produce the overall behavior of Multimedia KTV. The major purpose of adopting the SBC architecture design method, instead of separating the structure model from the behavior model, is to achieve one single coalesced model. In Figure 13, designers are able to see that the systems structure and behavior coexist in the SBCD. That is, in the SBCD of Multimedia KTV, designers not only see its systems structure but also see (at the same time) its systems behavior.
The overall behavior of an interactive multimedia system consists of many individual behaviors. Each individual behavior represents an execution path. We use interaction flow diagram (IFD) to design this individual behavior. The overall Multimedia KTV’s behavior includes two behaviors: KalaOK_Song_1 and KalaOK_Song_2.

Figure 14 shows the IFD of the behavior KalaOK_Song_1. First, actor Singer interacts with the component Song_Selection through the operation call interaction Select_Song_1. Next, component Song_Selection interacts with the component Song_1 through the operation call interaction Broadcast_Song_1. Finally, actor Singer interacts with the component Song_1 through the operation call interaction Sing_Song_1.
Figure 15 shows the IFD of the behavior *KalaOK_Song_2*. First, actor *Singer* interacts with the component *Song_Selection* through the operation call interaction *Select_Song_2*. Next, component *Song_Selection* interacts with the component *Song_2* through the operation call interaction *Broadcast_Song_2*. Finally, actor *Singer* interacts with the component *Song_2* through the operation call interaction *Sing_Song_2*.

4. Conclusions

An interactive multimedia system is very complex that it includes multiple views such as structure, behavior, and information views. The systems model designs the interactive multimedia system multiple views possibly using two different methods. The first one is the non-architecture-oriented method and the second one is the architecture-oriented method.

Non-architecture-oriented systems model respectively picks a model for each view. These multiple models are heterogeneous and unrelated of each other, thus there is no way to put them into a conformity model. Architecture-oriented systems model, instead of picking many heterogeneous and unrelated models, will use only one single coalescence model. The structure, behavior, and information views are all integrated in this multiple view coalescence (MVC) systems model.

Since structure and behavior views are the two most prominent ones among multiple views, integrating the structure and behavior views apparently is the best approach of integrating multiple views of a system. In other words, structure-behavior coalescence (SBC) facilitates multiple view coalescence (MVC). Therefore, we claim
that SBC architecture is an architecture-oriented systems model.

SBC architecture design method adopts the SBC architecture as a systems model. SBC architecture design method starts from the preparation phase and then goes through the creative thinking, planning, analysis, and design phases of SBC architecture construction. SBC architecture design method uses six fundamental diagrams to formally design the essence of an interactive multimedia system and its details at the same time. In the planning phase, architecture hierarchy diagram and framework diagram are used. In the analysis phase, component operation diagram and component connection diagram are used. In the design phase, structure-behavior coalescence diagram and interaction flow diagram are used. With these six diagrams, we then can effectively design the structure, behavior, and information of interactive multimedia systems; resolve uncertainties and risks caused by those traditional non-architecture-oriented design methods. Overall, SBC architecture design method helps integrate different stakeholders’ works on the same track and unfold the backbone of interactive multimedia systems. The interactive multimedia system design result of SBC architecture can be used as interactive multimedia system blueprints to improve the acceptance and effectiveness of the development of interactive multimedia system.

5. References