Study of Communication Model for Object-Oriented Methodology

Masahiro Akagi

School of Information Science,
Japan Advanced Institute of Science and Technology

February 15, 2001

Keywords: Communication Model, event communication, Object-oriented methodology, ObTS/ObCL, Formal CASE .

Recently, software become large scale and complex. It is said that Object-Oriented methodologies are useful solutions for them. Statecharts are commonly used for Dynamic Model which expresses the behavior of software in Object-Oriented methodologies. And recently, embedded systems are used in many area. By having many functions, embedded systems become large scale and complex, too. In development of reactive system such as embedded systems, we often use Statecharts. So, Dynamic Model is one of most important factors in case Object-Oriented methodologies are applied to development of reactive system.

ObTS is one of the model for expressing reactive systems. ObTS is a formal specification description model which relates Dynamic Model with structures of Object Model. In ObTS, the target system is described with hierarchy of objects, and objects executed in parallel is connected by event communications. And, ObCL is proposed as a formal specification description language based on ObTS. We can dynamically analyze ObCL codes by executing them on simulator ObML.

However it is difficult to compare the behavior of the target systems in ObML with the behavior of them in real world. Because inter-object communication in ObML based on Statechart-type communication model is synchronous, while communication in real world is usually asynchronous.

So, we focus the event communication which is used for expressing inter-object communication in many Object-Oriented methodologies. We divide computational model for ObTS into the communication model and execution model. Communication model expresses computation behaviors of events. Execution model expresses computation behaviors of objects. The communication model is the main subject of this study.

First, we identify concepts of event communication from Object-Oriented methodologies. By less arguments to treat events monistically in the past methodologies, we anew define event. And we give a definition of communication by the event. Then, we give a definition of the communication model. This communication model gives monistic definition of event communication.

Second, we extend ObCL/ObML to treat communication model flexibly. By this extention, we can describe systems containing two or more different communication models.

Next, we propose a methodology consistently based on the monistic definition of event communication. It is the software development process which is clear and consistent concerning to treat event communication. We can smoothly develop using ObTS/ObCL by the methodology.

Finally, we show a design of virtual elevator system as a case study using our communication model for evaluating the methodology and the extended ObCL/ObML. We find that the monistic definition of event communication is effective and that our methodology is useful for smooth and clear development using ObTS/ObCL.