

New Java Mechanism for Role-Based Information System Development

Haibin ZHU and MengChu ZHOU

Abstract- Object evolution and separation of concerns are two important issues in building a complex information system using object-oriented design and programming technology. The two issues are closely related to role playing activities in an organization. Roles have been discussed in object modeling for a long time. It is well-accepted that supporting roles in programming languages is required. This paper clarifies the role concept in the sense of object evolution and claims that a role should be taken as a fundamental component in object evolution. It then discusses the mentioned issues in software design and their resolutions. It proposes a principle that keeps the object identity and structure unchanged when passing messages to relevant roles. Next, it introduces the role play regulations, hierarchies and history recording mechanism. A package of classes for role play in Java is developed. Finally, it proposes a method to visualize the object evolution and role hierarchy. The work offers a conceptually solid and practically simple way to have objects in Java play roles. With the proposed methods, it is easy to express the evolution and separation of concerns of objects. The advantages over the previous methods are also discussed. The work assumes fundamental significance in building new information systems.

Index Terms— Role, Object Evolution, Separation of Concerns, Information System, Software Development, Modeling and Simulation

1. INTRODUCTION

People, goals and roles are the three essential components of an organization. Playing roles is a common activity in it. An information system imposes its own logic on a company's organization, strategy and culture [6]. Information systems are tools that help people perform their tasks better and more efficiently. Their goal is to help companies regulate their business processes [7]. Human roles are an important aspect of a system model [21]. It is important to introduce roles in modeling methodologies and role-based methodologies are emerging in different applications [22-25, 27, 28]. Roles have been widely used in organizations [3], project managements, software engineering and object modeling as they are good artifacts to deal with some aspects of these tasks and model natural phenomena more accurately. A common understanding is that when an object plays a role, it accepts messages and provides services related to its role. The introduction of roles to object modeling is to support object evolution and the separation of concerns [2, 5, 8-9, 12-14, 17-28]. They are closely related and always used to express the requirements of the role mechanisms in the modeling areas:

1) Object evolution states that an entity evolves and its status changes. A person plays a role of *student* in a year and may play role *professor* in a later year. It is required to express the person's status at different times.

2) Separation of concerns states that each concern of a given entity should be modeled by a single component in a system. A problem should be decomposed into entities such that each entity has one concern only. People may play roles *consultant* and *professor* during the same period of time. If you ask them "Could you teach a course?" or "Could you develop a system?" they should be able to properly respond to these requests within their roles.

As a tool to describe natural structures and phenomena, an ideal programming language provides direct modeling mechanisms to simplify the work of modelers. Therefore, providing role mechanisms in a programming language is a required task to support system modelers to deal with objects and the separation of concerns. Java is a popular general-purpose modeling and programming language widely used in many fields, especially information system development. Providing available, efficient and easy-to-use role mechanisms in Java is highly demanded by the designers, programmers and modelers of information systems.

Simulating their role playing is important in understanding the behaviors of people in organizations. Although there is some practice in supporting roles with object-oriented programming languages [5, 8, 19], many aspects are left for improvement, e.g., role definition, role playing, and expressing dynamic and traceable states and activities of objects. With the new approach proposed in this paper, it is expected to be easier for system modelers to simulate objects' collaboration activities, especially with roles. The proposed Java mechanisms and package can be directly used by system modelers in their applications requiring the use of roles.

This paper intends to help simulate objects' activities with roles more effectively, i.e., to express easily the separation of concerns of objects and object evolution in Java. A role is emphasized as an entity to express an object's current status in its evolving history. With this emphasis, we can extract many common properties of roles and evolving objects. The first beneficial application of the proposed mechanism is complex software system design. The second application is simulation of complex organizational systems. For example, we can build a role hierarchy of all the possible roles in a company and create role players to play these roles in a system simulation program. By simulation, we can review the past, view the

