Chapter XIII
A Formal Verification and Approach for Real-Time Databases

Pedro Fernandes Ribeiro Neto
Universidad do Estado do Rio Grande do Norte, Brazil

Maria Lígia Barbosa Perkusich
Universidade Católica de Pernambuco, Brazil

Hyggo Oliveira De Almeida
Federal University of Campina Grande, Brazil

Angelo Perkusich
Federal University of Campina Grande, Brazil

ABSTRACT

Real-time database-management systems provide efficient support for applications with data and transactions that have temporal constraints, such as industrial automation, aviation, and sensor networks, among others. Many issues in real-time databases have brought interest to research in this area, such as: concurrence control mechanisms, scheduling policy, and quality of services management. However, considering the complexity of these applications, it is of fundamental importance to conceive formal verification and validation techniques for real-time database systems. This chapter presents a formal verification and validation method for real-time databases. Such a method can be applied to database systems developed for computer integrated manufacturing, stock exchange, network-management, and command-and-control applications and multimedia systems. In this chapter, we describe a case study that considers sensor networks.

INTRODUCTION

Nowadays, the heterogeneity of platforms, distributed execution, real-time constraints, and other features are increasingly making software development a more complex activity. Besides, the amount of data to be managed is increasing as well. Taken together, complexity and data management are causing both risk and cost of software projects to get higher.

Database management systems are used to manage and store large amounts of data efficiently. However, when both data and transactions have timing restrictions, real-time databases (RTDB) are required to deal with real-time constraints (Ribeiro-Neto, Perkusich, & Perkusich, 2004). For an RTDB, the goal is to complete transactions on time, while maintaining logical and temporal consistency of the data. For real-time systems, correct system functionality depends on logical as well as on temporal correctness. Static analysis alone is not sufficient to verify the temporal behavior of real-time systems. To satisfy logical and temporal consistency, concurrency control techniques and time-cognizant transactions processing can be used, respectively. The last occurs by tailoring transaction management techniques to explicitly deal with time.

The real-time ability defines nonfunctional requirements of the system that must be considered during the software development. The quality assurance of real-time systems is necessary to assure that the real-time ability has been correctly specified. Imprecise computation is used as a technique for real-time systems where precise outputs are traded off for timely responses to system events. For that, formal models can be created to verify the requirement specifications, including the real-time specifications (Ribeiro-Neto, Perkusich, & Perkusich, 2003).

Validation as well as verification can be carried out by simulation model. With the simulation model, a random sample will be selected from the input domain of the test object, which is then simulated with these chosen input values. After that, the results obtained by this execution are compared with the expected values. Thus, a simulation model is as a dynamic technique, that is a technique that contains the execution of the test object. One major objective of simulation models is error detection (Herrmann, 2001).

The main motivation for this research is the fact that methods to describe conceptual models of conventional database systems cannot be directly applied to describe models of real-time database systems. It occurs because these models do not provide mechanisms to represent temporal restrictions that are inherent to real-time systems. Also, most of the available models focus on the representation of static properties of the data. On the other hand, complex systems, such as real-time databases, also require the modeling of dynamic properties for data and information. Therefore, the development of methods to design real-time databases with support for both static and dynamic modeling is an important issue.

In the literature, there are few works for real-time database modeling that allow a formal analysis, considering verification and validation characteristics. The existing tools for supporting modeling process especially do not present simulation capacity. The unified modeling language (UML) approach presents a number of favorable characteristics for modeling complex real-time systems, as described in Selic and Rumbaugh (1998) and Douglass (2004). UML also is used for modeling object-oriented database systems. However, the existing tools for UML modeling do not present simulation capacity.

This chapter describes a formal approach to verify and validate real-time database systems. The approach consists of the application of the five steps: (1) building an object model; (2) building a process model; (3) generating an occurrence graph; (4) generating a message-sequence chart; and (5) generating a timing diagram. The two first steps include static and dynamic analysis, respectively. The following steps allow the user to validate the model. Hierarchical coloured Petri nets (HCPNs) are used as the formal language to describe RTDB models (Jensen, 1998). The proposed approach can be applied to different domains, such as computer-integrated manufac-

Related Content