Chapter 1
Causes, Effects, and Consequences of Priority Inversion in Transaction Processing

Sarvesh Pandey
https://orcid.org/0000-0002-3014-9792
Madan Mohan Malaviya University of Technology, India

Udai Shanker
https://orcid.org/0000-0002-4083-7046
Madan Mohan Malaviya University of Technology, India

ABSTRACT

The problem of priority inversion occurs when a high priority task is required to wait for completion of some other task with low priority as a result of conflict in accessing the shared system resource(s). This problem is discussed by many researchers covering a wide range of research areas. Some of the key research areas are real-time operating systems, real-time systems, real-time databases, and distributed real-time databases. Irrespective of the application area, however, the problem lies with the fact that priority inversion can only be controlled with no method available to eliminate it entirely. In this chapter, the priority inversion-related scheduling issues and research efforts in this direction are discussed. Different approaches and their effectiveness to resolve this problem are analytically compared. Finally, major research accomplishments to date have been summarized and several unanswered research questions have also been listed.

DOI: 10.4018/978-1-7998-2491-6.ch001

Copyright © 2020, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

INTRODUCTION

Today, the systems/ applications are not only supposed to provide correct results, but these results should also come on or before some predefined time. Consequently, it has become very critical to design application-specific and time-constraint aware scheduling algorithms. The problem of priority inversion is a most talked about topic which requires wider researchers' attention as there are not vital solutions proposed till date to resolve this problem completely. It is a situation where task with high priority has been put on hold so that a low priority task may finish its execution. In the beginning of the research in this direction, priority inheritance approach was tried to resolve this problem. However, this approach does not actually eliminate the priority inversion completely but reduces its negative impact to some extent on the system by reducing the duration of it.

Let us suppose that two tasks are involved in a priority inversion problem. One is low priority task and another one is high priority task. Low priority task is currently holding the shared resource which is being requested by the high priority task. High priority task is waiting for completion of low priority task since low priority task is already holding the required resource. This is the case of priority inversion. It is done by upgrading the priority of low priority task to that of high priority task. The debate on usefulness of priority inheritance approach as a solution to the priority inversion problem is more than three decades old. Interestingly, researchers still do not agree and come to a conclusion whether this approach has a potential to address the ever-changing today's complex system requirements or not. More specifically, in some environments, this approach was proven to be an effective one while in others the completely contrary results were obtained.

Real-time computing systems are vital to a wide range of applications such as in the control of nuclear reactors & automated manufacturing facilities, in controlling & tracking air traffic, in advanced aircraft and in communication systems etc. All such systems control, monitor or perform critical operations, and must respond quickly to emergency events in a wide range of embedded applications (Rajkumar, 1989). They are, therefore, required to process tasks with stringent timing requirements and must perform these tasks in a way that these timing requirements are guaranteed to be met. Real-time scheduling algorithms attempt to ensure that system timing behavior meets its specifications, but typically assume that tasks do not share logical or physical resources. Since resource sharing cannot be eliminated, synchronization primitives must be used to ensure that resource consistency constraints are not violated.

Later, Lui Sha et al. analyzed the general priority inheritance class and proposed two protocols — the basic priority inheritance protocol and the priority ceiling protocol (Sha, Rajkumar, & Lehoczky, 1990). The objective of both the protocols was to solve the uncontrolled priority inversion problem. It has been claimed that the priority ceiling protocol solves this uncontrolled priority inversion problem particularly well and reduces the worst-case task blocking time to at most the duration of execution of a single critical section of a lower-priority task. This protocol also prevents the occurrences of deadlocks.

In software systems using preemptive scheduling based on task priorities, it is desirable to include a priority inheritance mechanism. This is an arrangement by which a task's priority is temporarily increased when it is blocking a task of higher priority. Although, it is easy to work out the time and way to increase a task's priority, the subsequent reduction of that task's priority involves some hidden traps. It is shown that the "obvious" solutions are flawed, in that they can reduce the priority either too early or too late (Moylan, Betz, & Middleton, 1993).

11 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/chapter/causes-effects-and-consequences-of-priority-inversion-in-transaction-processing/249420

Related Content

Understanding One-Handed Use of Mobile Devices
Amy K. Karlson, Benjamin B. Bedersonand Jose L. Contreras-Vidal (2008). *Handbook of Research on User Interface Design and Evaluation for Mobile Technology (pp. 86-101).*
www.irma-international.org/chapter/understanding-one-handed-use-mobile/21825

Expressive Audiovisual Message Presenter for Mobile Devices
Alex Garcia Gonçalvesand José Mario De Martino (2013). *International Journal of Handheld Computing Research (pp. 70-83).*
www.irma-international.org/article/expressive-audiovisual-message-presenter-mobile/76310

The M-Health Reference Model: An Organizing Framework for Conceptualizing Mobile Health Systems
Phillip Ollaand Joseph Tan (2009). *Mobile Computing: Concepts, Methodologies, Tools, and Applications (pp. 432-450).*
www.irma-international.org/chapter/health-reference-model/26519

Evolution of Mobile Services: An Analysis
Sunil Jose Gregory (2013). *Mobile Services Industries, Technologies, and Applications in the Global Economy (pp. 104-119).*
www.irma-international.org/chapter/evolution-mobile-services/68654

Rolopanel: Tracking Game Behaviour through Mobile Analytics
Monika Rajendra Astonkarand Amar Buchade (2014). *International Journal of Handheld Computing Research (pp. 48-59).*
www.irma-international.org/article/rolopanel/137120