Chapter 5
 Workflow Scheduling with Fault Tolerance

Laiping Zhao
Kyushu University, Japan

Kouichi Sakurai
Kyushu University, Japan

ABSTRACT

This chapter describes a study on workflow scheduling with fault tolerance. It starts with an understanding on workflow scheduling and fault tolerance technologies independently. Next, the chapter surveys the related works on the combination field of workflow scheduling and fault tolerance technologies. Generally, these works are classified into six categories corresponding to the six fault tolerance technologies: workflow scheduling with primary/backup, primary/backup with multiple backups, checkpoint, rescheduling, active replication, and active replication with dynamic replicas. An in-depth study on these six topics illustrates the challenge issues explored so far, e.g., overloading conditions, tradeoffs among scheduling criteria, and some future research directions are also identified. As applications are increasingly complex, and failures become a severe problem in large-scale systems, the authors expect to provide a comprehensive review on the problem of workflow scheduling with fault tolerance through this work.

DOI: 10.4018/978-1-4666-1888-6.ch005
acyclic graph (DAG), can be seen as a collection of computational tasks that are processed in a well-defined order to accomplish a specific goal.

Many challenges have been addressed in the field of workflow scheduling. And a major one of them is how to arrange the schedule to satisfy a requested criteria, which could be execution time, reliability, monetary cost or tradeoffs among them. While the scheduling performance on execution time has been studied for years since 2002 (Topcuoglu, Hariri & Wu, 2002), fault tolerance recently attracts a great attention for two main reasons: (1) System scale is growing fast, and failures are consequently becoming popular within large-scale clusters. One example is, according to the failure data from Los Alamos National Laboratory (LANL), annually more than 1,000 number of failures occur at system No. 7, which consists of 2014 nodes in total (Schroeder B. & Gibson G.A., 2006). Assuming some processors, whose MTBF is five years, then a cluster with two thousand such kind of processors, will produce more than one failure per day on average. (2) Heterogeneous systems, which are associated with many flexible and various hardware configurations, increase the complexity of system management, and perform more failures than the homogeneous system. Grid computing, for example, aims to combine together all volunteer machines of the world, presents a significant challenge for resource management. Given this context, fault tolerance technologies, e.g. primary/backup, active replication scheme, have been proposed. And they employ either time redundancy or resource redundancy to ensure the automatic execution of applications. However, most of the existing works, either on workflow scheduling or on fault tolerance, cover the alternative area. And only a few of them considered both sides together. Furthermore, most practical systems, e.g. Hadoop, Condor, have not yet applied a combining consideration into their implementations.

In this study, we have a combining consideration on the workflow scheduling and fault tolerance problem. The rest of this chapter is organized as following. Firstly, we give a brief introduction on the workflow scheduling and fault tolerance problem in section II and section III, respectively. Then, Section IV surveys the proposed fault tolerant workflow scheduling algorithms, which apply the primary/backup into scheduling. Section V presents the related works on rollback-recovery or rescheduling. And section VI discusses the workflow scheduling with active replication. Then the hybrid approach with applying multiple fault tolerance technologies into scheduling is given in section VII. We conclude this chapter in the section VIII, where some major challenge issues and open problems are identified.

BACKGROUND

Workflow Scheduling

Scheduling is one of the core services in distributed systems. It defines a placement policy, following which the tasks are allocated to the processors. Mapping tasks to processors comes with various performance metrics, e.g. execution time, economic cost, load balance, system throughput, etc., which have been richly studied in the literature (Benoit, Hakem & Robert, 2009; Blythe et al. 2005; Topcuoglu et al. 2002; Wieczorek et al., 2009; Yu & Buyya, 2006a). Generally, a static workflow scheduling algorithm follows three steps: (1) deciding the task scheduling order, (2) mapping task to processors, and (3) task submission.

In step (1), all the tasks in a workflow are ordered into a certain sequence, and the tasks will be scheduled on processors one by one. Apparently, there is a disparity among different sequences on the scheduling performances. An appropriate sequence is supposed to be chosen with considering the scheduling criteria. Anyhow, the task ordering methods must share a common constraint that is ensuring the precedence relationship among tasks. To our knowledge, two approaches on task priori-
Related Content

Channel Impairments for V2V Communications in ITS Scenarios
[www.igi-global.com/chapter/channel-impairments-v2v-communications-its/62810?camid=4v1a](www.igi-global.com/chapter/channel-impairments-v2v-communications-its/62810?camid=4v1a)

Smartphones to Access Hybrid Information Spaces
[www.igi-global.com/chapter/smartphones-access-hybrid-information-spaces/45267?camid=4v1a](www.igi-global.com/chapter/smartphones-access-hybrid-information-spaces/45267?camid=4v1a)

Optical Transport Network: A Physical Layer Perspective Part 2
[www.igi-global.com/chapter/optical-transport-network/117810?camid=4v1a](www.igi-global.com/chapter/optical-transport-network/117810?camid=4v1a)

Logical Methods for Self-Configuration of Network Devices
[www.igi-global.com/chapter/logical-methods-self-configuration-network/60449?camid=4v1a](www.igi-global.com/chapter/logical-methods-self-configuration-network/60449?camid=4v1a)