Abstract

Various communication and computing tasks in the fields can be integrated and applied in a distributed system. However, those resources are heterogeneous and dynamic in nature, connecting a broad range of resources. This study proposed a hybrid load balancing policy to maintain performance and stability of distributed system in Mobile services. Proposed work suggests to opt the use of some of advanced and efficient technologies like Multiagent. In this proposed implementation a two models are developed. The worker model is used to describe the workload and its current distribution within the system. The master model defines for a given algorithm at any instant of time and shows the formal context for obtaining and evaluating the load distribution decisions. Multiagent computing on a cluster of workstations is widely envisioned to be a powerful paradigm for building useful distributed applications. The Mobile agents of the system span across all the machines of a cluster. Just like the case of traditional distributed systems. With different characteristics between ordinary processes and agents, it is interesting and useful to investigate whether conventional load-balancing strategies are also applicable and sufficient to cope with the newly emerging needs, such as coping with temporally continuous agents, devising a performance metric for multi agent systems, and taking into account the vast amount of communication and interaction among agent. This work discusses the above issues with reference to agent properties and load balancing techniques and outlines
the space of load-balancing design choices in the arena of multi agent computing. The proposed algorithm works by associating a credit value with each agent. The credit of an agent depends on its affinity to a machine, its current workload, its communication behavior, and mobility. When a load imbalance occurs, the credits of all agents are examined and an agent with a lower credit value is migrated to relatively lightly loaded machine in the system. Proposed work consider the problem of load balancing to minimize the cost of dynamic computations, including the cost of migrations. We propose the Ripple load balancing paradigm, The load balancing service presented is a generic tool for enhancing performance of accessing distributed objects from the WAP interface.

Reference

Multi-agent Optimized Load Balancing Using Spanning Tree for Mobile Services

Workshops (ICDCSW’03) 0-7695-1921-0/03 © 2003 IEEE


[20] Xiaodong Zhang, Yanxia Qu, Li Xiao “Improving Distributed Workload Performance by Sharing Both CPU and Memory Resources”

[21] Xiaodong Lu, Yi Zhou and Kinji Mori “Agent-Based Rating Oriented Information Provision and Reallocation for High-Assurance in Open and Dynamic Environments” Proceedings of the 19th International Conference on Distributed Computing Systems Workshops (ICDCSW’04) 0-7695-2087-1/04 $20.00 © 2004 IEEE


[29] Bhardwaj, Ghosh , Mani” Scheduling Divisible loads in parallel and distributed system”

[30] Shiraji, Hurson Kav “Scheduling and Load Balancing in parallel and distributed System”


[32] “Web Service Scalability and Performance with Optimising Intermediaries” Mark Nottingham mnot@akamai.com

[33] Ananya Das, Charles Martel, Biswanath Mukherjee, and Smita Rai “A Better Approach to Reliable Multi-Path Provisioning” Department of Computer Science, University of California, Davis, CA 95616 Email: fdas, martel, mukherje, raig@cs.ucdavis.edu
Multi-agent Optimized Load Balancing Using Spanning Tree for Mobile Services

[34] “Works in Progress: The 2nd International Middleware Doctoral Symposium” IEEE DISTRIBUTED SYSTEMS ONLINE 1541-4922 © 2006 Published by the IEEE Computer Society Vol. 7, No. 3; March 2006
[35] VALERIA CARDELLINI Tor Vergata PHILIP S. YU IBM T.J “DYNAMIC LOAD BALANCING ON WEB-SERVER SYSTEMS “.
[36] Wang Fangxiong, Jiang Zhiyong “ Research on A Distributed Architecture of Mobile GIS Based on WAP”, State Key Laboratory of Information engineering in Surveying, Wuhan University

Index Terms

Computer Science Distributes Systems

Key words

Load balancing Multiagent
WAP
Spanning Tree