

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320-088X

IJCSMC, Vol. 3, Issue. 3, March 2014, pg.947 – 953

RESEARCH ARTICLE



SECURE AND EFFICIENT ROLLBACK RECOVERY IN GRID ENVIRONMENT

Ms. A. Wisy Shantha

PG Scholar of CSE Department
R.V.S Faculty of Engineering, Coimbatore
India

wisyabraham22@gmail.com

Prof. S. Subashini

Assistant Professor of CSE Department
R.V.S Faculty of Engineering, Coimbatore
India

deancsefoe@rvsgroup.com

Abstract— Large applications executing on Grid or cluster architectures consisting of hundreds or thousands of computational nodes create problems with respect to reliability. The source of the problems is node failures and the need for dynamic configuration over extensive runtime. This paper presents two fault-tolerance mechanisms called Theft-Induced Check pointing and Systematic Event Logging. These are transparent protocols capable of overcoming problems associated with both benign faults, i.e., crash faults, and node or subnet volatility. Specifically, the protocols base the state of the execution on a dataflow graph, allowing for efficient recovery in dynamic heterogeneous systems as well as multithreaded applications. By allowing recovery even under different numbers of processors, the approaches are especially suitable for applications with a need for adaptive or reactionary configuration control. The low-cost protocols offer the capability of controlling or bounding the overhead. A formal cost model is presented, followed by an experimental evaluation. It is shown that the overhead of the protocol is very small, and the maximum work lost by a crashed process is small and bounded. One possible solution to address heterogeneity is to use platform independent abstractions such as the Java Virtual Machine. However, this does not solve the problem in general. There is a large base of existing applications that have been developed in other languages. Reengineering may not be feasible due to performance or cost reasons. Environments like Microsoft .Net address portability but only few scientific applications on Grids or clusters exist.

Full Text: <http://www.ijcsmc.com/docs/papers/March2014/V3I3201499b32.pdf>