Hierarchical Application Aware Error Detection and Recovery

Ravishankar K. Iyer
Center for Reliable and High-Performance Computing
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
1308 West Main Street, Urbana, IL 61801
+1 217 333 9732, iyer@crhc.uiuc.edu

ABSTRACT

Proposed is a four-tired approach to develop and integrate detection and recovery support at different levels of the system hierarchy. The proposed mechanisms exploit support provided by (i) embedded hardware, (ii) operating system, (iii) compiler, and (iv) application.

Categories and Subject Descriptors

D.2.4 [**Software/Program Verification**]: reliability, assertion checkers and B.8.1 [**Performance and Reliability**]: reliability, testing and fault tolerance.

General Terms: Reliability, Performance.

Keywords: Hierarchical error detection and recovery, embedded hardware, software implemented fault tolerance.

1. INTRODUCTION

The technology scaling together with power reduction become the key contributors to higher error rates and substantially increase the chances of multiple and/or near coincident errors. This poses significant new challenge because most existing recovery techniques while capable of handling single errors are notoriously inefficient in coping with multiple errors. Moreover, with pervasiveness of network environments and increasing system complexity chances of errors to propagate grow. While current studies show that a small percentage of faults propagate across the system hierarchy the system-wide impact of the propagated errors can be catastrophic. The system may hang or crash, the diagnosis and assessment of the system damage can become very difficult, and recovery may require considerable amount of time. These factors are a significant impediment for achieving high availability of the order of 5NINES (less than 5 minutes downtime per year).

The challenge is to design application aware mechanisms for providing tight error containment to prevent error propagation and low-latency detection and rapid recovery to enable high availability applications.

2. APPROACH

We discuss designing hierarchical system of detection and recovery schemes/mechanisms some of which can be embedded into the hardware, e.g., a processor, while others can be integrated with the operating system or application (e.g., via a robust middleware). We present four-tired approach to develop and integrate detection and recovery support at different levels of the system hierarchy.

Embedded Hardware Support. We develop Reliability and Security Engine (RSE) [3], a hardware framework implemented as an integral part of a modern microprocessor. In this framework the hardware modules embedded in the RSE provide error detection and security services and execute in parallel with the core processor pipeline. The application can be instrumented to instruct the processor about the desired level and type of runtime checking. The checking can range from full duplication of the instruction stream, to precise spot-checks of individual instructions and/or results produced by critical code sections.

Operating System Support. We embed detection and recovery mechanisms directly into operating system services to rapidly detect OS hangs/crashes and recover the system in an automated fashion.

Compiler Support. We employ a compiler assisted automated generation of assertions for runtime error detection. The idea is to analyze data generated during the compilation process and to identify data patterns (or data sets), which can be used as signatures of a correct application state or behavior. Assertions for runtime signature checking can be integrated within the application or implemented in hardware.

Application Support. We develop ARMOR architectural framework to provide detection and recovery to applications using flexible and configurable software solutions based on ARMORs. An ARMOR consists of pre-built, reusable software modules, which can be customized to the application needs. The architecture has been formally specified, and employed to deploy a software-implemented, fault-tolerant environment for supporting distributed applications [2].

3. ACKNOWLEDGMENTS

This work was supported by grants from DARPA and MARCO sponsored Gigascale Systems Research Center.

4. REFERENCES

[1] Z. Kalbarczyk, et al., "Hierarchical Simulation Approach to Accurate Fault Modeling for System Dependability Evaluation," IEEE Trans. on Software Engineering, 25(5), 1999.

[2] Z. Kalbarczyk, et al., "Chameleon: A Software Infrastructure for Adaptive Fault Tolerance," IEEE Transactions on Parallel and Distributed Systems, 10(6), 1999.

[3] N. Nakka, et al., "An Architectural Framework for Providing Reliability and Security Support," to appear in Dependable Systems and Networks 2004.

DAC 2004, June 7–11, 2004, San Diego, California, USA