Transparent Intra-machine Full-Software-Stack Replication for Fault Tolerance

Giuliano Losa, Antonio Barbalace, Yuzhong Wen, Marina Sadini, Binoy Ravindran
ECE Department, Virginia Tech, Blacksburg, USA
{giuliano.losa, antoniob, wyz2014, sadini, binoy}@vt.edu

We propose FT-Popcorn, the first operating system providing transparent fault-tolerance using full-software-stack replication within a single machine, similarly to lockstep processors but using commodity hardware. FT-Popcorn addresses the growing concern about hardware faults in data-centers.

FT-Popcorn partitions the hardware resources of a multiprocessor machine among isolated replicas of the full software-stack, including the operating system, achieving a distributed design where no software component is a single point of failure (virtualization is not used).

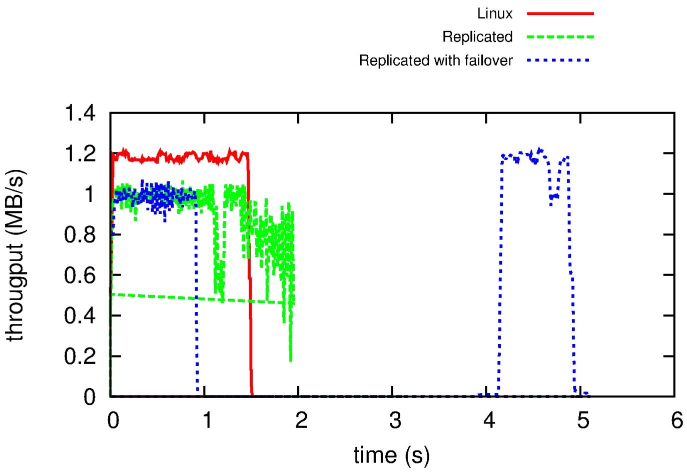
With FT-Popcorn, we explore a new software architecture for transparent fault-tolerance. Compared to hardware-implemented lock-step execution, FT-Popcorn has cost advantages, offers a wider choice of hardware, and protects against a similar range of faults; compared to existing software solutions, FT-Popcorn has lower overhead than full-machine replications because it uses shared memory for replica communication, does not require duplicating entire machines, saving costs and space, and is more resilient than other single-machine replication solutions (hypervisor-based or user-space only) because all the software stack is replicated. FT-Popcorn is based on Popcorn Linux [1] and runs unmodified Linux applications, enabling evaluation of existing applications.

Our goal is to assess whether single machine full-software-stack replication is a practical fault-tolerance solution for the data-center by building and experimentally evaluating a prototype. Multiple design and implementation challenges must be addressed to reach this goal, such as running multiple synchronized Linux kernel replicas on the same hardware with minimal overhead, and adapting replication algorithms built for a networked environment to achieve high performance in shared-memory.

FT-Popcorn coordinates replicas by combining primary-backup replication with deterministic replica execution in order to minimize synchronization points while ensuring consistent replication.

Deterministic User-space. A custom deterministic scheduler makes multi-threaded and multi-process userspace applications deterministic using a deterministic logical-time algorithm inspired by [2] which enforces a unique total order on inter-thread and inter-process communication. The primary replica resolves all remaining non-determinism, such as the logical time of I/O delivery, and forces the other replica to follow its choices.

Deterministic Kernel-space. Because we replicate the full software stack, including the OS, a major difficulty is to deterministically execute the operating system. However, for transparent application failover, only application-visible non-determinism needs to be eliminated. Kernel subsystems holding application-visible state, such the TCP stack or the VFS, run exclusively on the primary replica. Other replicas instead maintain a logical representation of the state of those subsystems which can be used upon failover to initialize the subsystems of the new primary. The implementation of the logical TCP stack is based on the work of Alvisi et al. [3].



I/O Replication. The primary has exclusive ownership of all the I/O devices accessed by replicated applications and broadcasts I/O data to the other replicas. Upon failover, a device-ownership transfer mechanism is activated.

Running Multiple Replicas. FT-Popcorn isolates replicas by partitioning the hardware and dedicating special memory areas to be used as mail-boxes for inter-replica communication. The FT-Popcorn prototype relies on timely detection of hardware faults to turn them into crash-stop faults. On incompatible hardware, the gap between a hardware fault and its detection increases the likelihood that FT-Popcorn will crash because of cross-replica contamination. However, future use of a Byzantine Fault-Tolerant synchronization protocol could decrease cross-replica contamination chances.

Initial Results. Our experimental evaluation shows that FT-Popcorn achieves fault tolerance on a selection of server applications and incurs a moderate performance overhead compared to an unmodified Linux kernel running on the resource partition of an FT-Popcorn replica. Figure 1 shows the throughput obtained with FT-Popcorn, without failure and with a failure, compared to Linux running the Apache HTTP server benchmarking tool repeatedly requesting a 32Kb page with 20 concurrent requests at a time from a Mongoose server running 20 server threads.

Acknowledgements. This work is supported by the Air Force Office of Scientific Research under the grant FA9550-14-1-0163.

References

[1] Barbalace, A., Murray, A., Lyerly, R., and Ravindran, B. Popcorn: a replicated-kernel os based on linux.

[2] Merrifield, T., and Eriksson, J. Increasing concurrency in deterministic runtimes with conversion.

[3] Zagorodnov, D., Marzullo, K., Alvisi, L., and Bressoud, T. C. Practical and low-overhead masking of failures of tcp-based servers. *TOCS 27*, 2 (2009).