Exo: Atomic Broadcast for the Rack-Scale Computer

Matthew P. Grosvenor
University of Cambridge

Marwan Fayed
University of Stirling

Andrew W. Moore
University of Cambridge

Extended Abstract

Agreement is a crucial component of many distributed systems. It is the feature that is at the center of critical algorithms that provide consensus, election, and failure detection, among others. Intuitively, agreement between processes is possible only in the presence of uninterrupted or well ordered operations. One such powerful primitive is atomic, or 'total order,' broadcast.

The way in which atomic broadcasts are implemented depends on the underlying communications infrastructure. On a single device atomicity is relatively unambiguous, often due to hardware support. For example, single processors can enforce atomic reads and writes to shared memory. In multiprocessors, cache coherency is maintained by way of distributed MESI/MOESI bus protocols or centralised directory-based schemes. Atomicity within these environments is facilitated by the presence of special purpose, low latency, highly reliable interconnects. Consequently, atomic operations inside a single machine are fast, and can be completed in just a few CPU cycles.

Across devices few assumptions can be made: The communication infrastructure that connects devices is general purpose, higher latency and less reliable. Given the absence of hardware support, agreement is reached by way of software state replication and consistency algorithms such as Paxos[2] and Zookeeper (Zab)[5]. These are effective but complex solutions. As a consequence, atomic broadcast via software is slow. Previous studies have shown that atomic broadcasts can take milliseconds to complete [4].

The racks-scale computer (RSC) falls somewhere between these two worlds. On the one hand, we would like to be able to program the RSC as if it were a single multi-processor machine, with hardware supported fast atomic primitives. On the other, we would like for individual components in this machine to be able to fail without affecting the operation of the machine as a whole. Our work is motivated by this apparent contradiction, and the observation that closely co-located devices in RSCs present an opportunity to re-envision network support for distributed operations.

In response we are engineering Exo, a fast and efficient network architecture and protocol for atomic broadcasts at the rack scale. Exo employs a special purpose network, constructed from general purpose Ethernet networking components. The Exo physical infrastructure comprises a broadcast/aggregate network similar to Hubnet [3] shown in Figure 1. We envisage Exo network as one of many (potentially special purpose) networks present in the RSC.

Logically, Exo implements a token ring protocol similar to Totem [1]. Token rings are well a understood mechanism for building atomic broadcast systems. They are cheap to build, and run at predictably high speeds. The Exo protocol is accelerated by a simple offload engine in the NIC and the presence of the low-latency broadcast network described above. As a result, it provides total ordered sequential consistency with deterministic delivery times of a few hundred nanoseconds, even in the presence of failures.

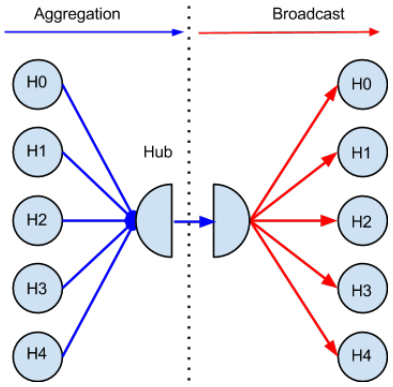

Figure 1: Physical Exo architecture.

At time of writing, Exo is running in our lab at modest prototype scale. We have implemented the Exo network using a commodity layer one matrix switch and layer 2 ethernet switch respectively. We have implemented the Exo offload engine in FPGA enabled NICs. Once validated we will measure performance in experiments at much larger scale. Our expectation is that Exo will achieve consensus messaging in loosely coupled rack scale computers at rates that are an order of magnitude faster than current software based systems.

1. REFERENCES

[1] Amir, Y., Moser, L. E., et al. The Totem Single-ring Ordering and Membership Protocol. *ACM Trans. Comput. Syst.* (1995).

[2] Lamport, L. The Part-time Parliament. *ACM Trans. Comput. Syst.* (1998).

[3] Lee, E., and Boulton, P. The Principles and Performance of Hubnet: A 50 Mbit/s Glass Fiber Local Area Network. *Selected Areas in Communications* (1983).

[4] Marandi, P., Benz, S., et al. The Performance of Paxos in the Cloud. In *Proceedings of Reliable Distributed Systems (SRDS)* (2014).

[5] Reed, B., and Junqueira, F. P. A Simple Totally Ordered Broadcast Protocol. In *Proceedings of LADIS'08* (2008).