

Towards Unified Mechanisms for Inter-Processor Communication

Manolis G.H. Katevenis*

Foundation for Research & Technology - Hellas
Institute of Computer Science (FORTH-ICS)
and
University of Crete
Dept. of Computer Science Heraklion, Crete, Greece

Abstract. Modern processors no longer work mostly in isolation, but rather communicate continuously with peer processors and other devices; communication is now at least as important as computation. Old architectures had the communication medium –the network– far from the processor, interfacing to it through the I/O bus, which was acceptable when the network was slower than the processor. New architectures need to bring the network close to the processors, at a latency and throughput level equal to that of the cache memories. Coherent caches are good at supporting Implicit Communication, where the communicating threads do not know in advance which input data will be needed or who produced them. On the other hand, in the cases of Explicit Communication, when the input data set is known ahead of time, prefetching yields best performance; furthermore, when the users-to-be of an output data set are known, eager send works even better. Prefetching (pull-communication) works either on top of coherent caches with prefetch engines, or on top of local stores (scratchpad memories) with remote DMA, but consumes much less network traffic –hence energy– in the second case. Eager send (push-communications) works almost only using remote DMA; again, traffic and energy advantages are even more pronounced. Recent advances in parallel programming efficiently support explicit communication, by letting the programmer only identify the input and output data sets, and having the compiler and runtime system do the rest by appropriately placing the data and scheduling the transfers. We conclude that future chip multiprocessors should have local SRAM blocks that are configurable to operate partly as coherent caches and partly as local (scratchpad) memories; it should then be possible and advantageous to merge the cache controller and network interface functions into a single unit. These combined hardware mechanisms will most efficiently support both implicit and explicit communication, leading to a unification of the two traditional camps: shared memory and message passing.

* This work was funded by the EU projects SARC, HiPEAC, and UNiSIX, and was in collaboration with Georgi Gaydadjiev, Alex Ramirez, Dionisios Pnevmatikatos, Dimitrios Nikolopoulos, Angelos Bilas, Stamatis Kavadias, Vassilis Papaefstathiou, and others.