Lightfleet®

A New Solution to the Networked Coherent Memory Problem

Overview

Early multicomputer systems used a shared memory wherein each endpoint had access to the same physical memory. A shared memory serving more than a few endpoints becomes both very expensive and highly inefficient. Today’s multicomputer systems are based on collections of individual servers or endpoints, each with its own complement of memory, effectively rendering the use of shared memory unmanageable in such systems. The management of a distributed, mirrored memory then becomes one of maintaining identical copies of local memory across the entire cooperating group (which may include all of the endpoints) and granting access in a manner to maintain system-wide memory consistency and coherency. The constraint on each group member is to ensure that read access by any member at any time returns the same result. Such coordination usually requires a mechanism of semaphores and locks to prevent access to local copies until such time that the entire set of memories belonging to the group in question is guaranteed to be consistent. “Coherence” means that each mirrored copy is identical at the time of access; ideally, coherence would be temporal, however since mirrored copies may reside at separate locations, strict temporal coherence is not feasible. The management process becomes one of ensuring that memory access is logically coherent.

Solving the problem

Without native multicast capabilities, updates to multiple memory spaces must be done one-at-a-time. In such a case, latency of memory update operations increases with the number of physical memory copies involved. However, if the communication network in a cluster is able to update mirrored copies simultaneously, a useful virtual shared memory becomes possible, one that supports a much simpler programming model. Lightfleet supports this in its Data Distribution System (DDS) by providing native support or multicasting and self-directed data flow. Unlike other networking implementations that involve control planes, with the multicast capabilities inherent in the DDS, memory coherence can easily be initiated and maintained across the entire set of hosts as well as for each group, which may have members distributed across a network. Physical memory in a given host may contain mirrored segments belonging to multiple groups and coherence is maintained independently for each group supported by a given host. Such multiply-independent, group-coherent memories are often quite difficult to establish and costly to maintain in other environments.

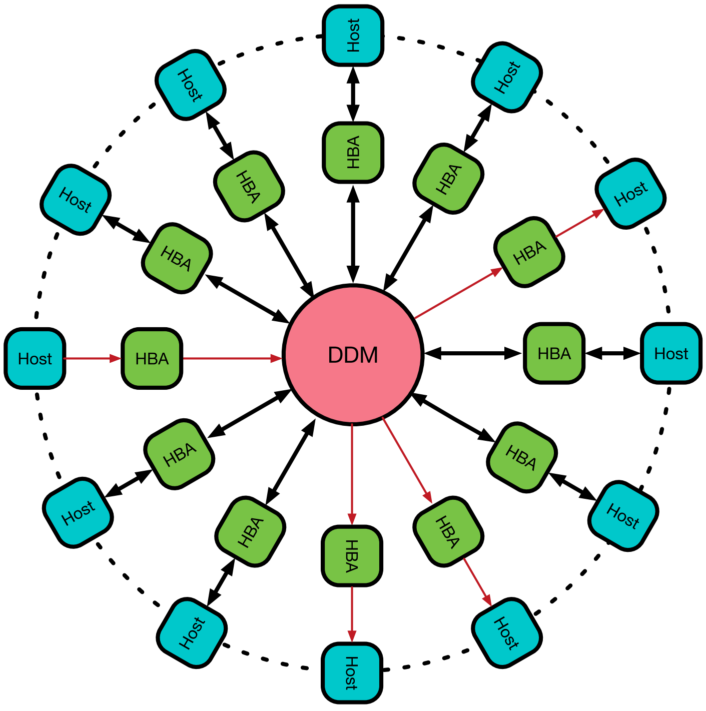
To maintain memory coherence in a multicomputer with distributed shared memory requires an efficient consistency model; Lightfleet supports four consistency models, all of which are dependent on the native multicast transmission mode built in at the architectural level. *Local consistency* (lowest level) is automatically achieved by the multicast process without regard to order of the transmission. The next level is a *general consistency*, where all memory locations are eventually identical after all participating processes have completed their writes. *Release consistency* is achieved by a system of software semaphores (locks) where an updated region of local memory is multicast to all group members when a local process is complete. Finally, a *read-coherent* memory access method (discussed below) is made possible by tables maintained in each interconnect node in the DDS.

Elements of Lightfleet's DDS

The Lightfleet DDS consists of a DataRotor™ data-distribution module (DDM) serving multiple host-bus adapters (HBAs), schematically illustrated in Figure 1, which shows multiple hosts and accompanying HBAs connected to a central DDM. (The dashed circle indicates a plurality of inputs, connections and hosts, while the red arrows illustrate a particular multicast transmission originating in the far-left host.) Each HBA resides in or is closely coupled to an endpoint or host computer or server. Each HBA is connected to the DDM by means of optical fibers which carry information in the form of serialized data streams segmented into frames.

A DDS packet consists of data frames representing a payload with a start-of-packet (SOP)/end-of-packet (EOP) wrapper. Hence, the destination (or subscriber information) travels along with and introduces each packet to the various components of the DDS. All information pertaining to the packet is therefore contained in the wrapping SOP and EOP, with the result that the DDS does not need a supervisor to manage and maintain data pathways. Any supervision or packet control is provided by the interaction of the wrapper with the elements of the DDS. In this manner, the packet "finds its own way" through the DDS to the destinations(s) prescribed by the SOP in an adaptive manner without any outside or top-down supervision.

Figure 1.
The Lightfleet DDS provides native support for multicast operations.



The self-directing properties embedded in a packet allow a DDS fabric to operate without the spanning-tree machinery required by switched systems; such spanning trees both enable and frustrate data traffic in switched fabrics.

Since the destination is a "group" rather than a single location, the Lightfleet DDM is designed as a multicast data transport, with unicast messaging being a special case of multicast. Thus, the DDS fulfills the basic requirements of multicomputer applications from distributed to parallel computing. Coordination of multiple processors across a wide variety of applications is adequately served, with low latency and high efficiency, by a DDS fabric.

Efficient realization of a virtual shared memory allows the DDS interconnect to serve the needs of parallel applications both by means of messages and direct memory access. That is, parallel programming becomes viable for a distributed multicomputer system at a lower cost than a dedicated supercomputer needing hardware support for physical shared memory.

Controlling the flow

The control of data flow within and to the DDS is distributed throughout the fabric and is accomplished by issuing high-priority, single-frame (64 bit) control messages when an input queue in either an HBA or the DDM is nearing capacity. A flow-control frame sends a simple “off” or “stop-sending” command to the source feeding the complaining input queue, halting the transmission until such time as the queue is able to accept more data when an “on” command or “start sending” is sent. This is in sharp contrast to a switch with centralized traffic control. DDS flow control is more responsive to internal conditions both in halting traffic to a module when conditions require it and in allowing traffic to resume.

Handshaking in the form of acknowledgements (and negative acknowledgments) also makes use of these high-priority control frames and provides a point-to-point guarantee of correct message arrival, ensuring that there are no lost packets due to transmission errors.

DDS as a data-moving system

The DDS’ support for group-coherent memory is based on the versatility of the fast priority message and the access to host memory allowed by the HBAs’ direct memory access (DMA) function. In the hosts, memory management is relegated to the kernel which is responsible for updating the mirrored copies when writing to coherent memory and by allowing reads from group memory only when the requested location is consistent across the group, so that all mirrored copies are up to date.

The DDS supports two methods for achieving coherent memory. The first, termed “read-coherent memory access” or RCMA, is optimal for a “light workload” where infrequent calls for memory access are made and randomly distributed across a group. The second, based on a semaphored operating environment (SOE) is designed for heavy or concerted access to a section of group memory in a particular host.

The RCMA method coordinates memory access by notifying the DDM of any read and write requests where write requests are global across the entire group whose mirrored memory copies are to be updated and read requests are strictly local to the mirrored copy of the group memory in the host initiating the read request. The request-grant mechanism is fast as it is based on the fast priority message and the write command efficiently updates group memory by means of the (synchronous) multicast mechanism.

The SOE method makes use of semaphores and locks, mediated by the message-passing and fast priority messages supported by the DDS, to ensure that coherency is maintained across mirrored copies of memory for each group member across the entire set of participating endpoints. The SOE mechanism operates by allowing only one group member in a particular host to carry out whatever memory operations are required at the moment by the local task. When finished, the next group member that requested access is allowed the same access. Any memory that has been altered during the protected access is updated across the entire group when the access lock is released.

In this way, the coherent memory enabled by the DDS bridges the divide between distributed-memory systems (where each processor has its own private memory) and shared-memory systems (where all processors have access to a common memory). This allows both parallel (shared memory) and distributed (local memory) algorithms to run on the same system, expanding the utility of the cluster and enhancing the programming model.

The Lightfleet DDS ensures a lower-latency traffic flow and higher resource utilization (meaning the distributed processes spend less time waiting for data) over that of an Ethernet or InfiniBand system. Installation and operating costs (cooling and power) are lower than with traditional methods and the bottom-up nature inherent in the DDS data-movement process leads to autonomous and efficient operation with no lost packets.

Interested in evaluating Lightfleet technology?
Lightfleet will provide all the details to qualified organizations. Please call Ivy Yap at 360.816.2840 or email iyap@lightfleet.com to order or discuss specific requirements.

Lightfleet®

For more information, visit
www.lightfleet.com

Lightfleet Corporation
4800 NW Camas Meadows Dr.
Camas, WA 98607-7671
USA