4.4.2 Remote Memory

The Remote Memory API's allow buffers in separate memory subsystems that are not directly accessible to be shared buffers. This can be accomplished if either CPU can see both memory regions, or if a DMA engine can provide a datapath to move the memory, or through some other form of communication that can perform the data transfer. These methods can optionally include a software cache.

If a CPU in the system can see both memory regions, then it can directly perform the memory transfers between memory spaces. A remote CPU node may not have access and must request the CPU that has access to perform any synchronization requests.

In a DMA transfer method, the DMA must have access to both memory regions. This entails set up of the buffer to be initially transferred between memory regions. The initial buffer and the copy are ready for access by either node. DMA can be used independently of a software cache or in conjunction with a software cache.

A software cache is similar to a hardware cache, and gives the ability to synchronize between different CPU's accessing the same memory structure which makes the accesses by both CPU's coherent. For example, when any write access is performed on a remote memory buffer, the result can be immediately stored in the software cache. If another CPU does a read or write access to the same region of the buffer, the software cache must communicate between CPU's and synchronize the buffer between remote memory regions prior to performing the buffer access. A sync command will force the remotely shared memory region to be synchronized.