5) Figure 9 shows Node A executing on CPU0, and the subsystem specified as “execution_context”, in order to determine which resource node A is executing on:

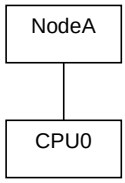

Figure 9 – Data Returned for Node A (Execution Context)

6) Figure 10 shows Node A executing on CPU0, no partition, with the subsystem specified as “memory”:

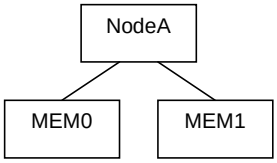

Figure 10 – Data Returned for Node A (Memory)

7) It is possible to register a callback function that is called when a system partition is changed. Suppose the system before re-partitioning is as in case (1) above, and then after re-partitioning CORE0 and CORE1 are on different partitions not visible to each other. A call to `mrapi_resources_get()` following the callback function (after re-partitioning) might yield the following. Using node A on CPU0, after re-partitioning, the results would look like case (3) above. Using node B on CPU1, after re-partitioning, the results would look like case (4) above.