NUMA Quiz 2

Given a NUMA multiprocessor system with 4 nodes, each node with one processor, 64 KB of private cache and 8 GB of main memory. Coherence among nodes is kept with a directory-based MSU protocol, using the transactions explained in class.

Assume that the home node for variable *var* is NUMAnode0 and that at a given moment there exist clean copies of that variable in the cache memories in NUMAnode1 and NUMAnode2, but no copies in NUMAnode0 and NUMAnode3. If the processor in NUMAnode1 wants to update (write to) variable *var*, which of the following coherence actions do NOT occur?

Trieu-ne una o més:

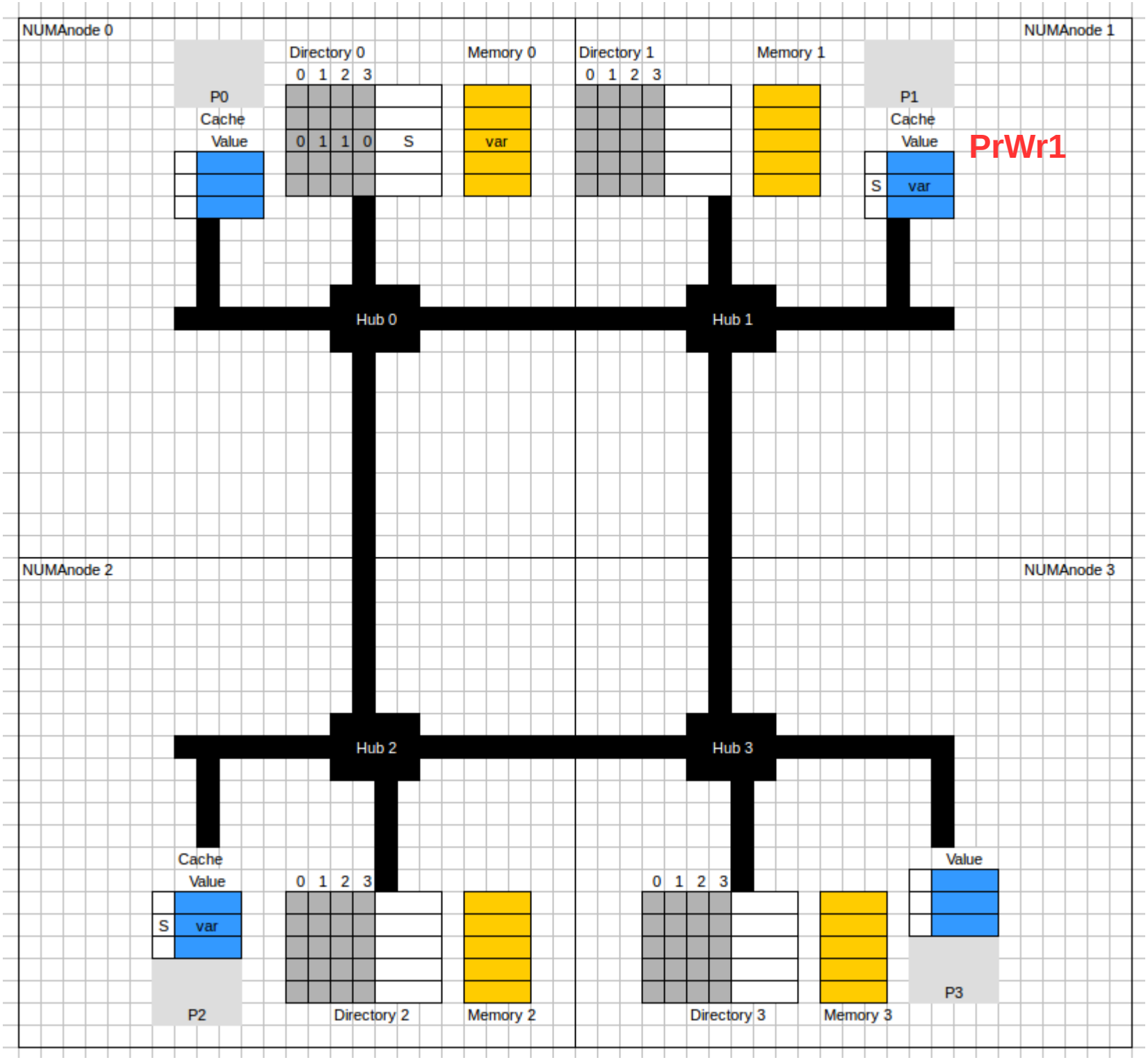
- [ ] The processor in the local node (NUMAnode1) issues PrWr.
- [ ] The local node sends a UpgrReq command to the home node (NUMAnode0).
- [ ] Since there is another copy of the line in the cache of a remote node (NUMAnode2), the home node sends an Invalidate command to it.
- [ ] Since there is another copy of the line in the cache of a remote node (NUMAnode2), the local node sends an Invalidate command to it.

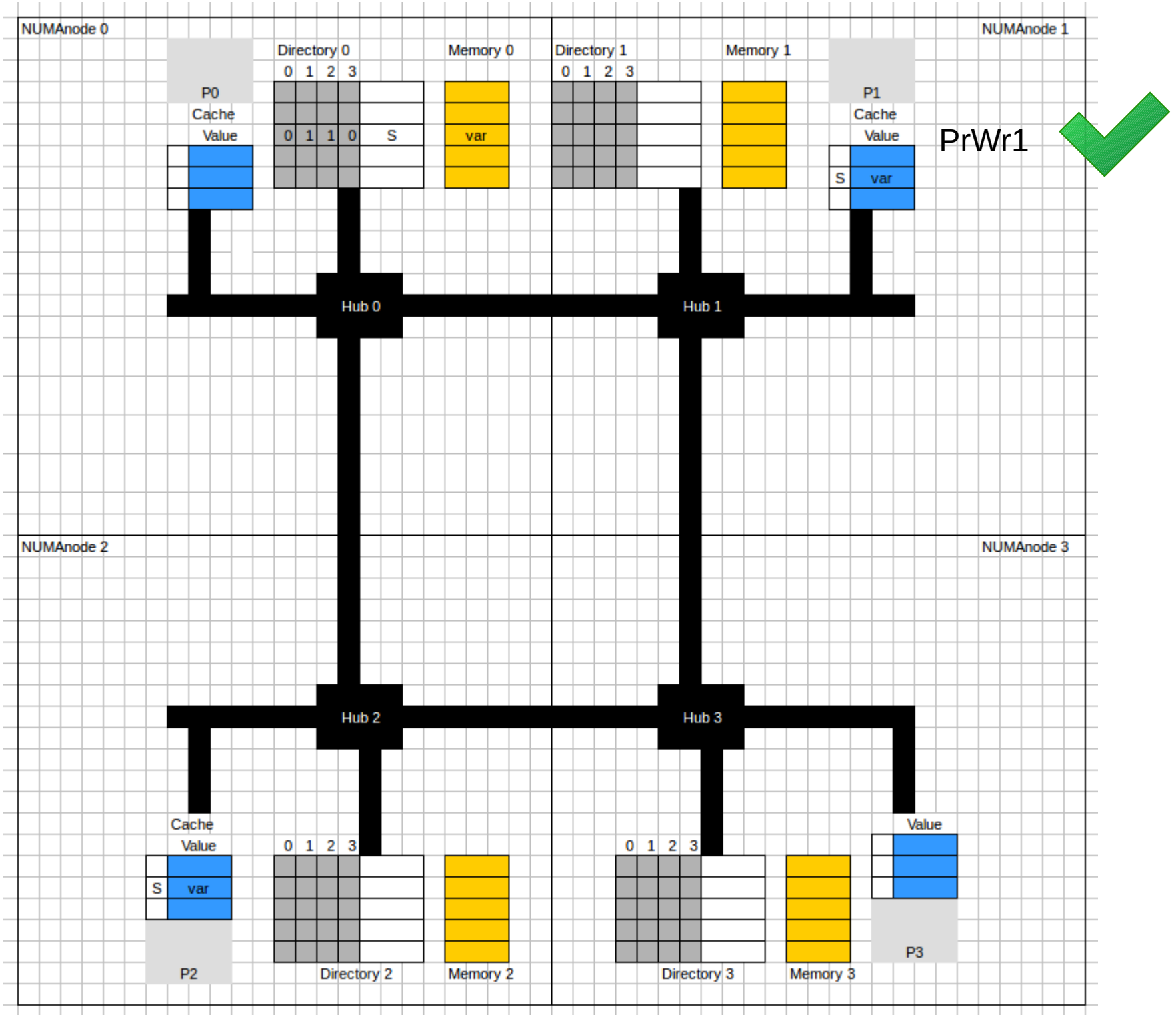
Given a NUMA multiprocessor system with 4 nodes, each node with one processor, 64 KB of private cache and 8 GB of main memory. Coherence among nodes is kept with a directory-based MSU protocol, using the transactions explained in class.

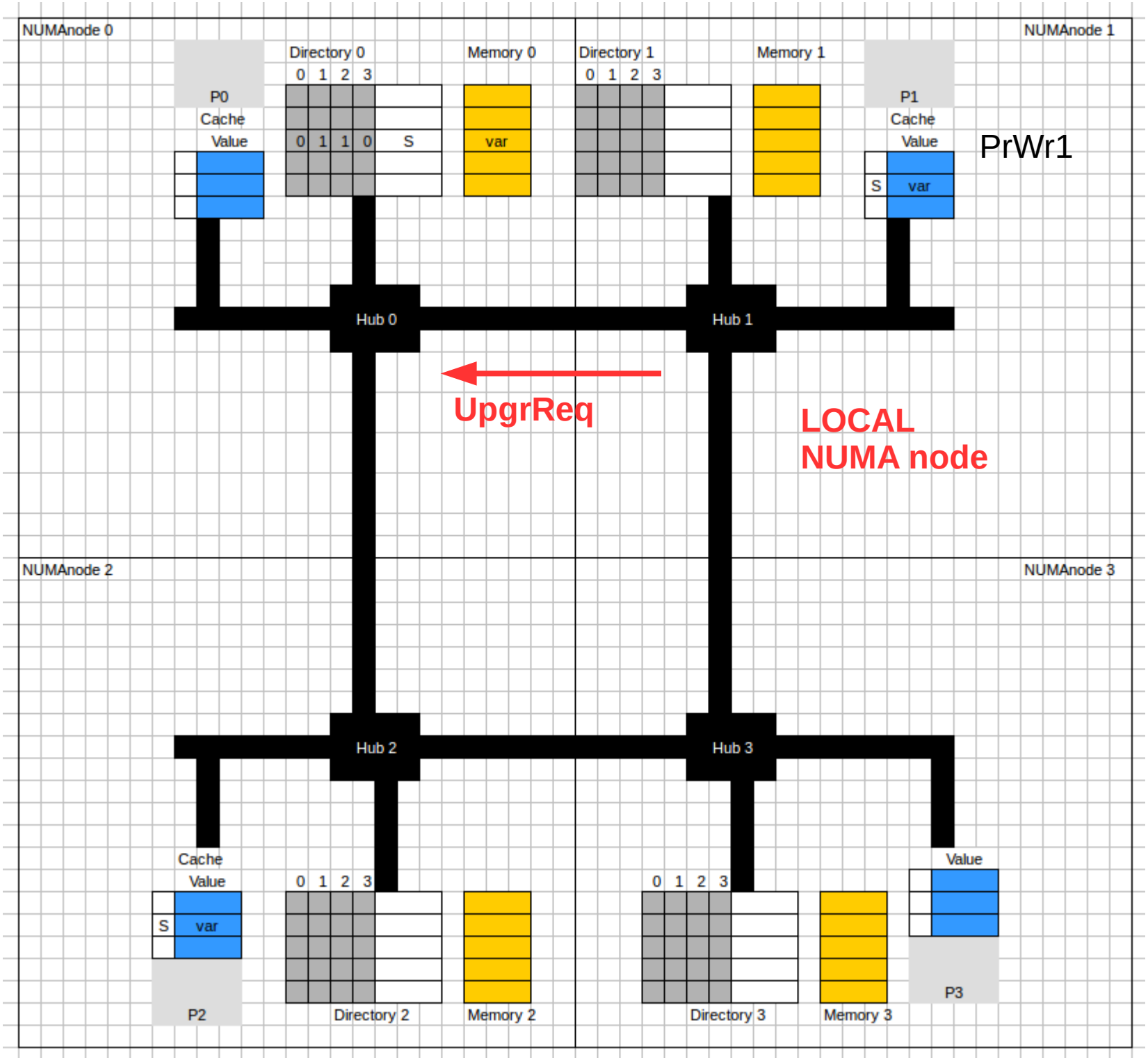
Assume that the home node for variable *var* is NUMAnode0 and that at a given moment there exist clean copies of that variable in the cache memories in NUMAnode1 and NUMAnode2, but no copies in NUMAnode0 and NUMAnode3. If the processor in NUMAnode1 wants to update (write to) variable *var*, which of the following coherence actions do NOT occur?

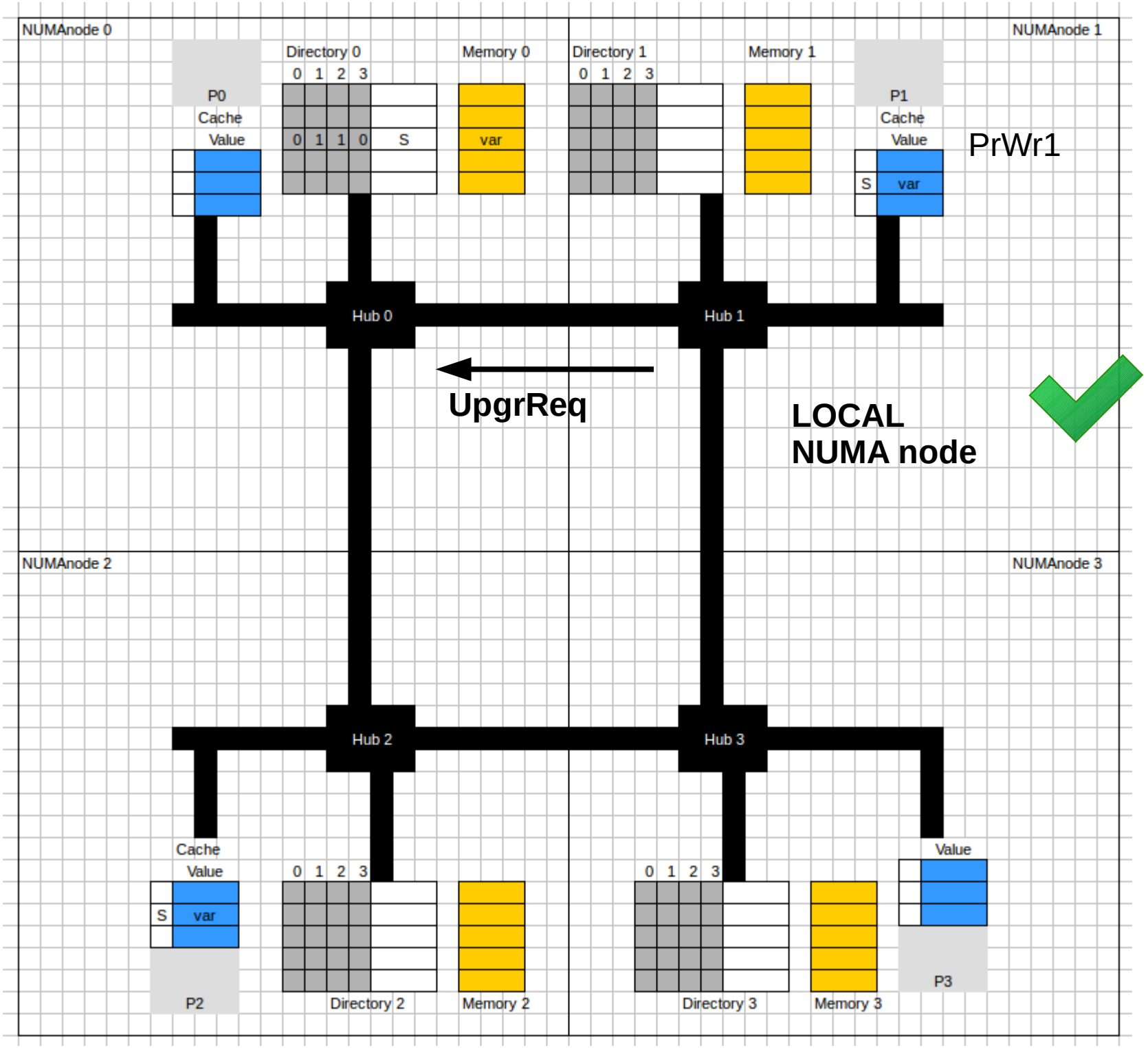
Trieu-ne una o més:

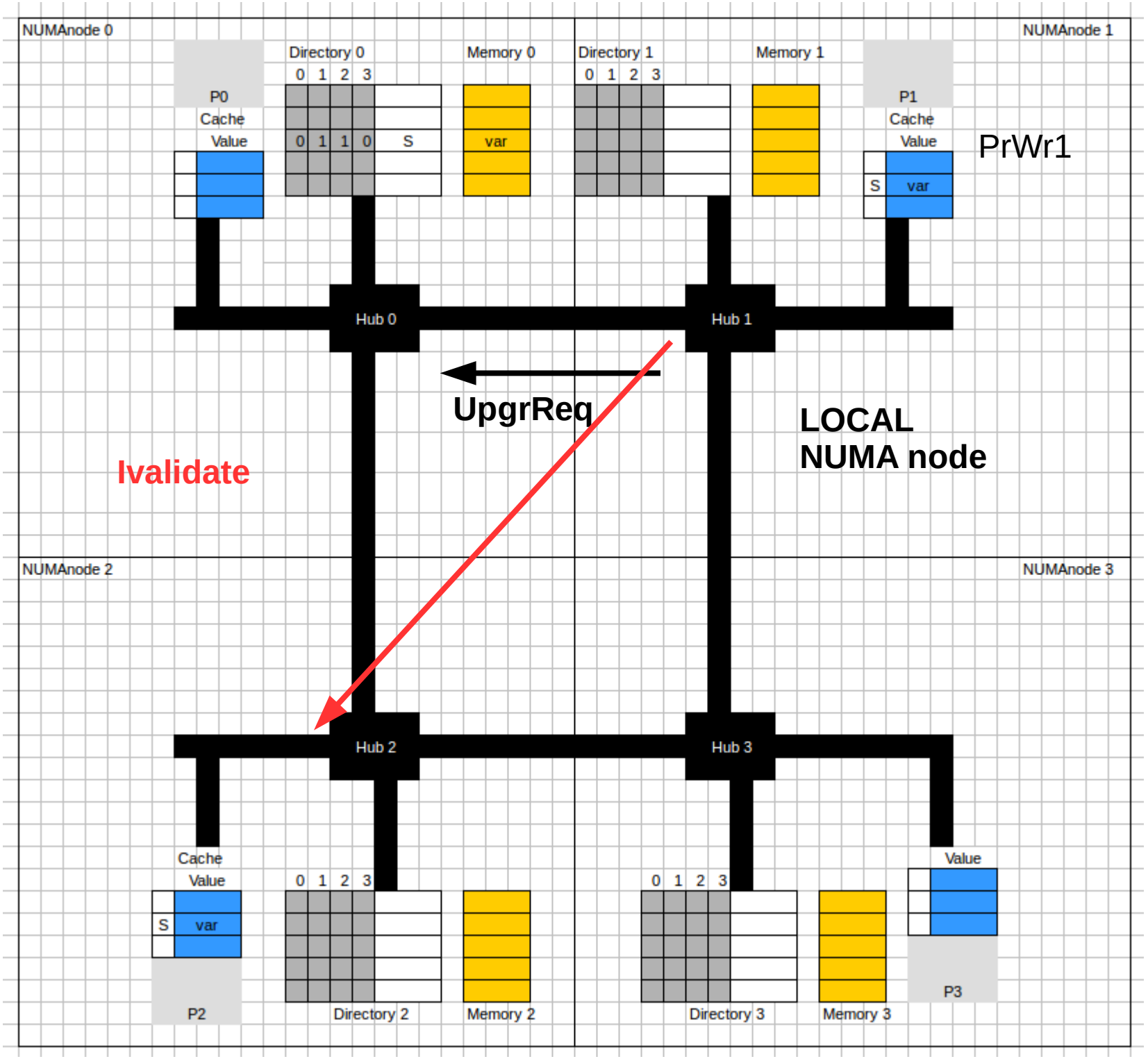
- [ ] The processor in the local node (NUMAnode1) issues PrWr.
- [ ] The local node sends a UpgrReq command to the home node (NUMAnode0).
- [ ] Since there is another copy of the line in the cache of a remote node (NUMAnode2), the home node sends an Invalidate command to it.
- [ ] Since there is another copy of the line in the cache of a remote node (NUMAnode2), the local node sends an Invalidate command to it.

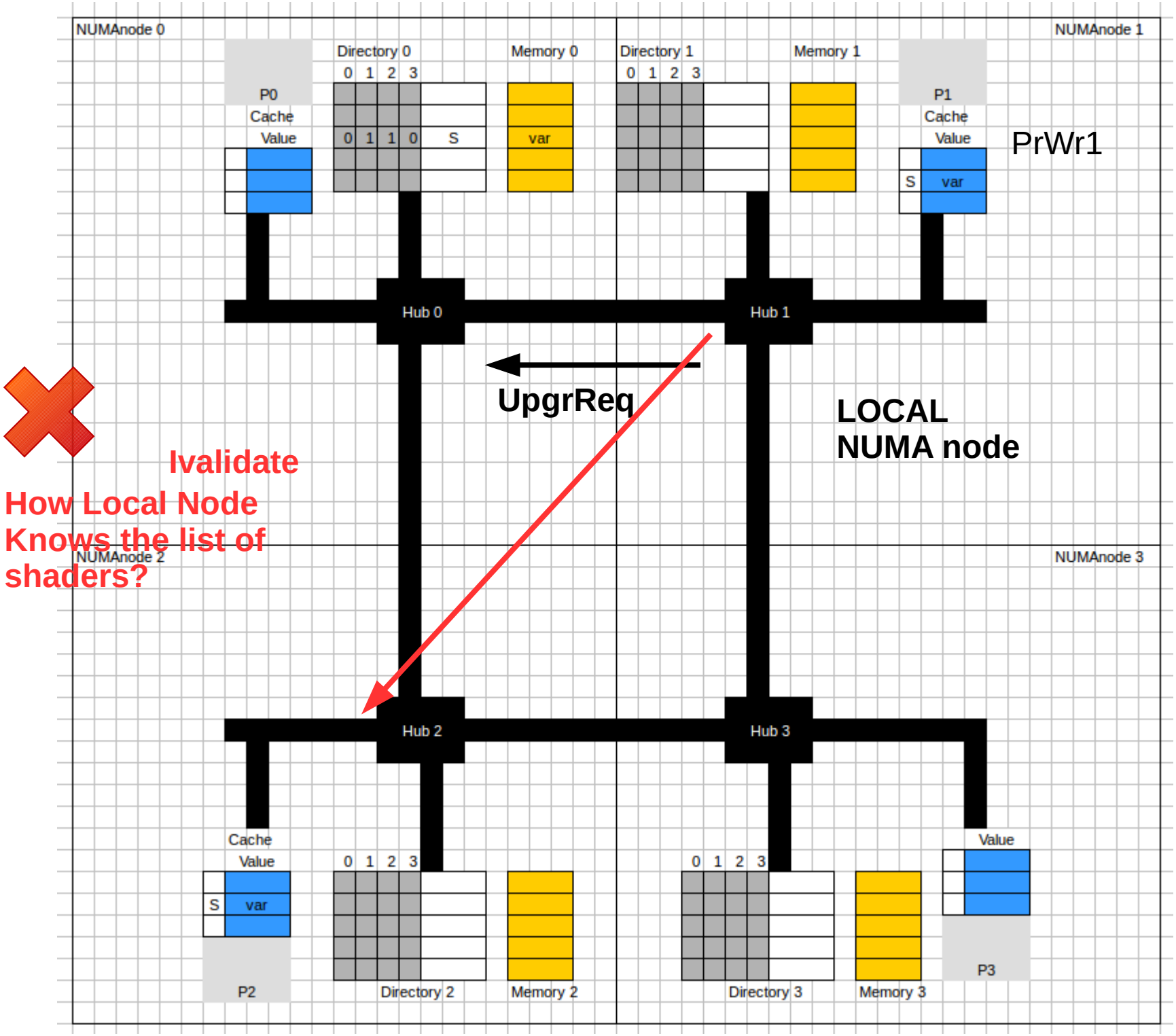

NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 1 0 S
var
P1
S var
PrWr1
Hub 0
Hub 1
Hub 2
Hub 3
Cache
Value
S var
P2
Directory 2
Memory 2
Directory 3
Memory 3
Value
P3

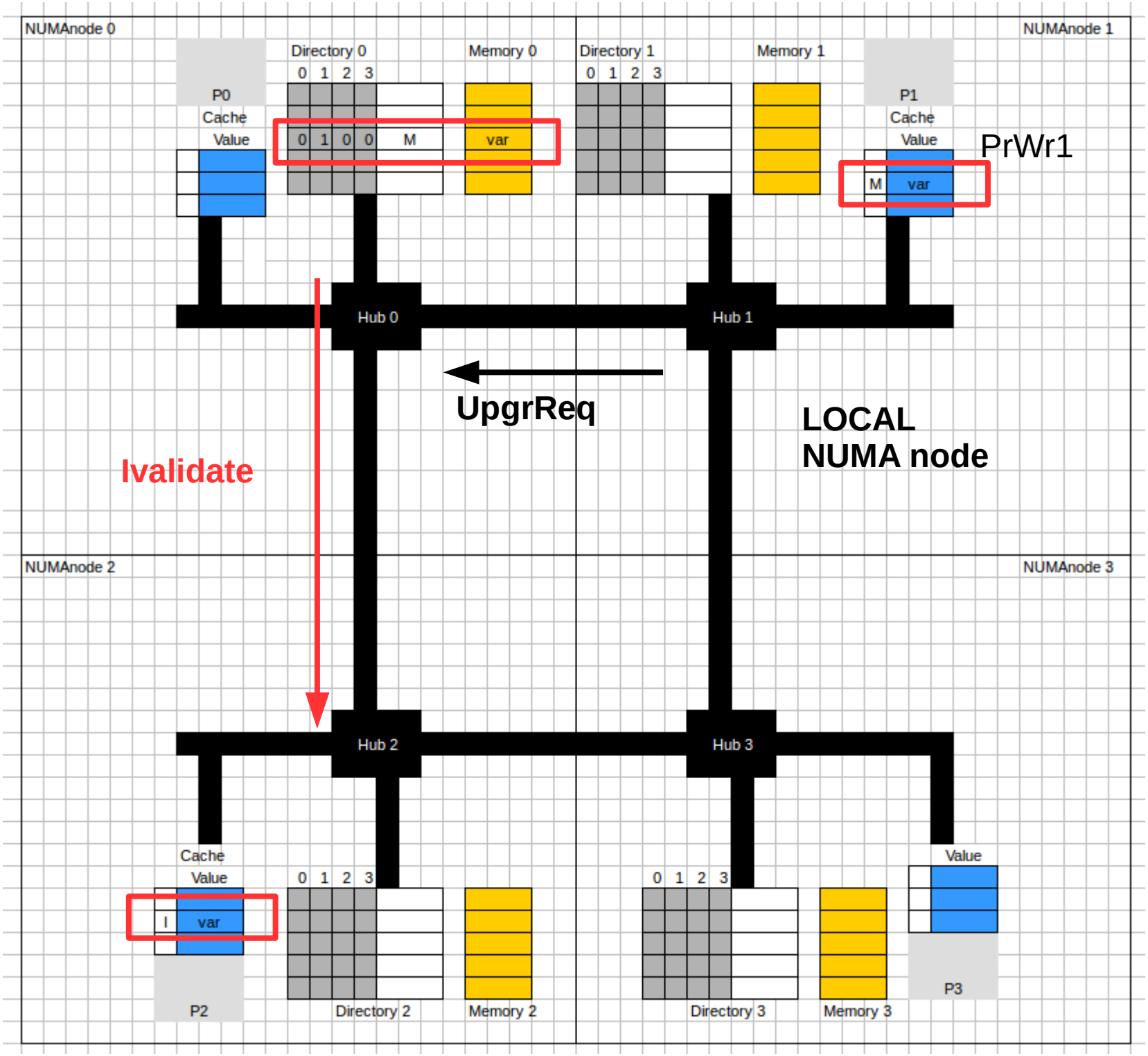

NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 1 0 S
var
P1
S
Hub 0
Hub 1
Hub 2
Hub 3
P2
P3
Directory 2
Memory 2
Directory 3
Memory 3
PrWr1

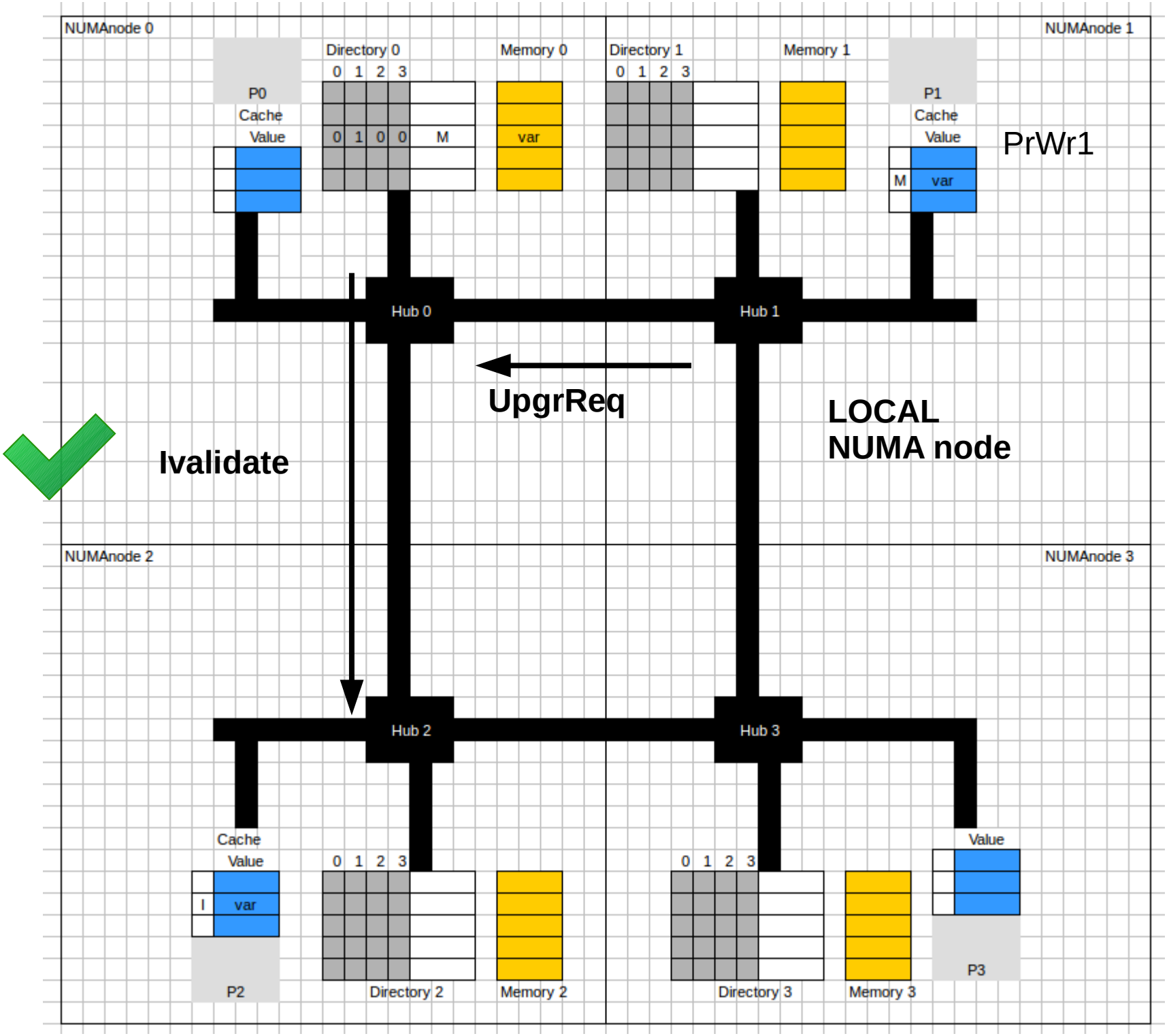

NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 1 0 S
var
P1
Cache
Value
S var
PrWr1
Hub 0
Hub 1
Hub 2
Hub 3
UpgrReq
LOCAL
NUMA node
Cache
Value
S var
P2
Directory 2
Memory 2
Directory 3
Memory 3
Value
P3


NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 1 0 S
var
P1
Cache
Value
S var
PrWr1
Hub 0
Hub 1
UpgrReq
LOCAL
NUMA node
Hub 2
Hub 3
Cache
Value
S var
P2
Directory 2
Memory 2
Directory 3
Memory 3
Value
P3


NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 1 0 S
var
P1
Cache
Value
S var
PrWr1
Hub 0
Hub 1
UpgrReq
Ivalidate
LOCAL
NUMA node
Hub 2
Hub 3
Cache
Value
S var
P2
Directory 2
Memory 2
Directory 3
Memory 3
Value
P3


NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 1 0 S
var
P1
Cache
Value
S var
PrWr1
Hub 0
Hub 1
Hub 2
Hub 3
UpgrReq
LOCAL
NUMA node
Ivalidate
How Local Node
Knows the list of
shaders?
Cache
Value
S var
P2
Directory 2
Memory 2
Directory 3
Memory 3
Value
P3


NUMAnode 0
NUMAnode 1
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 0 0 M
var
P1
Cache
Value
M var
PrWr1
Hub 0
Hub 1
UpgrReq
LOCAL
NUMA node
Ivalidate
NUMAnode 2
NUMAnode 3
Hub 2
Hub 3
Cache
Value
I var
P2
Directory 2
Memory 2
Directory 3
Memory 3
Value
P3


NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 0 0 M
var
P1
M var
PrWr1
Hub 0
Hub 1
Hub 2
Hub 3
UpgrReq
Ivalidate
LOCAL
NUMA node
I var
P2
Directory 2
Memory 2
Directory 3
Memory 3
P3

Given a NUMA multiprocessor system with 4 nodes, each node with one processor, 64 KB of private cache and 8 GB of main memory. Coherence among nodes is kept with a directory-based MSU protocol, using the transactions explained in class.

Assume that the home node for variable *var* is NUMAnode0 and that at a given moment there exist clean copies of that variable in the cache memories in NUMAnode1 and NUMAnode2, but no copies in NUMAnode0 and NUMAnode3. If the processor in NUMAnode1 wants to update (write to) variable *var*, which of the following coherence actions do NOT occur?

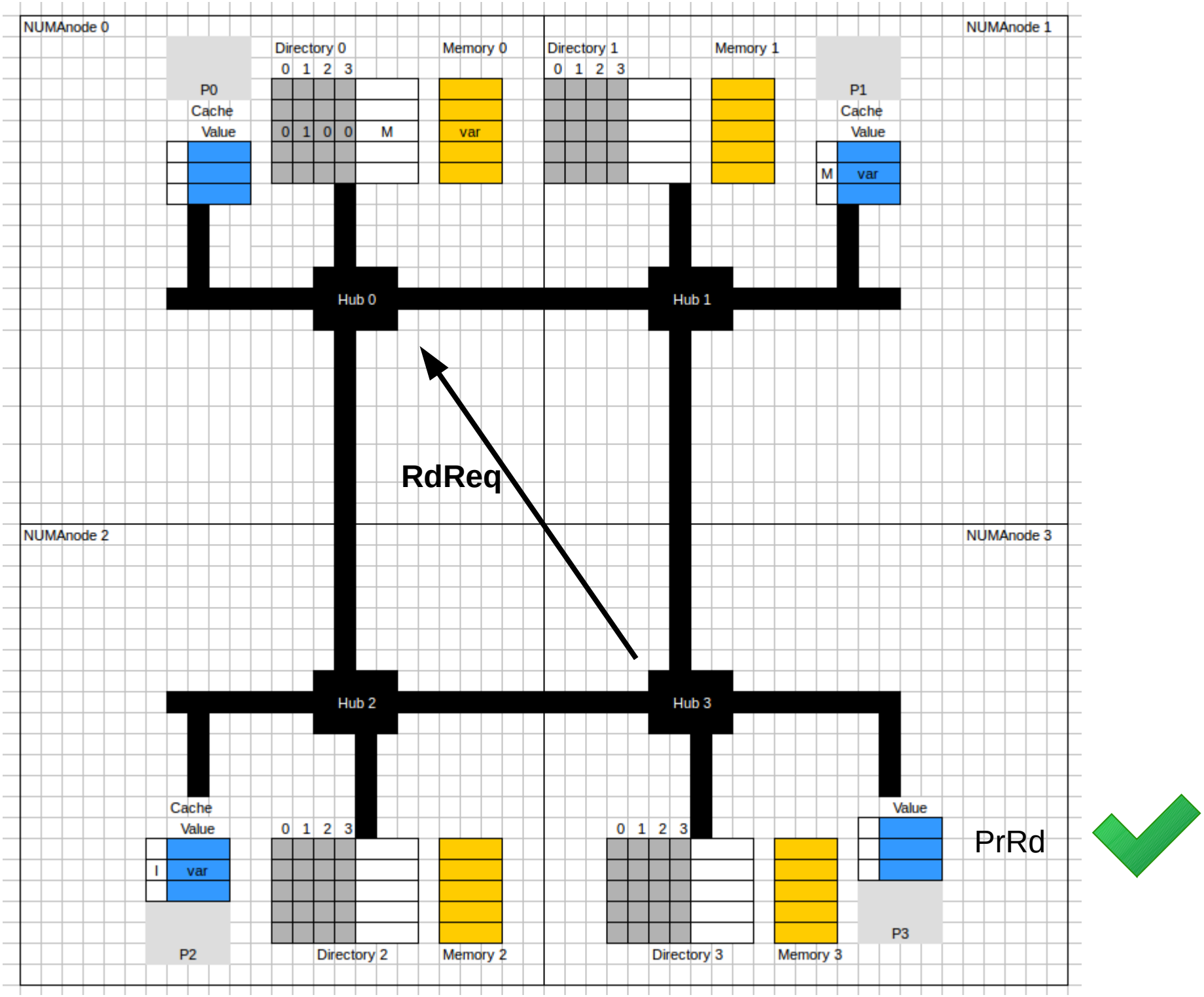
Trieu-ne una o més:

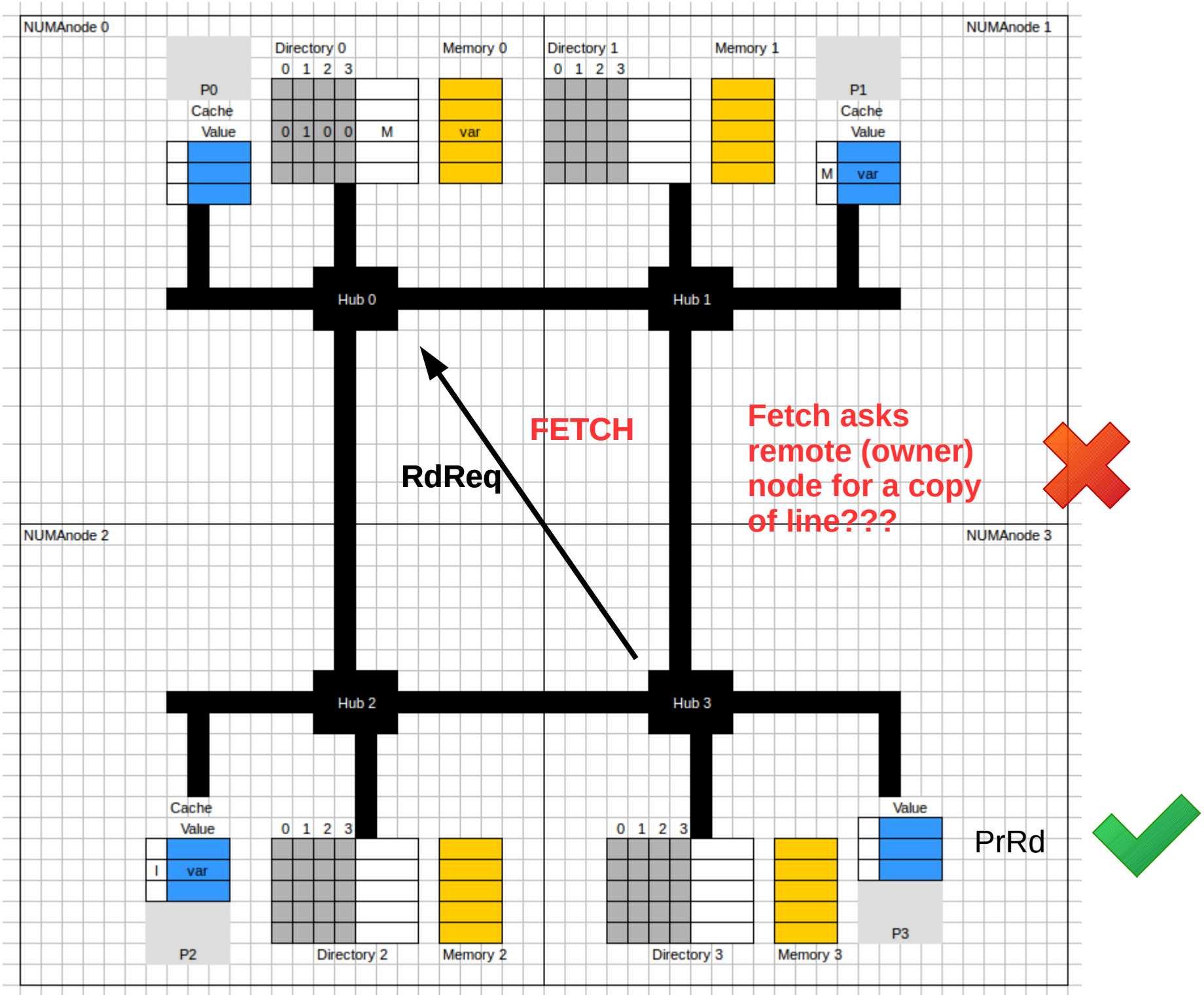
- [ ] The processor in the local node (NUMAnode1) issues PrWr.
- [ ] The local node sends a UpgrReq command to the home node (NUMAnode0).
- [ ] Since there is another copy of the line in the cache of a remote node (NUMAnode2), the home node sends an Invalidate command to it.
- [ ] Since there is another copy of the line in the cache of a remote node (NUMAnode2), the local node sends an Invalidate command to it.

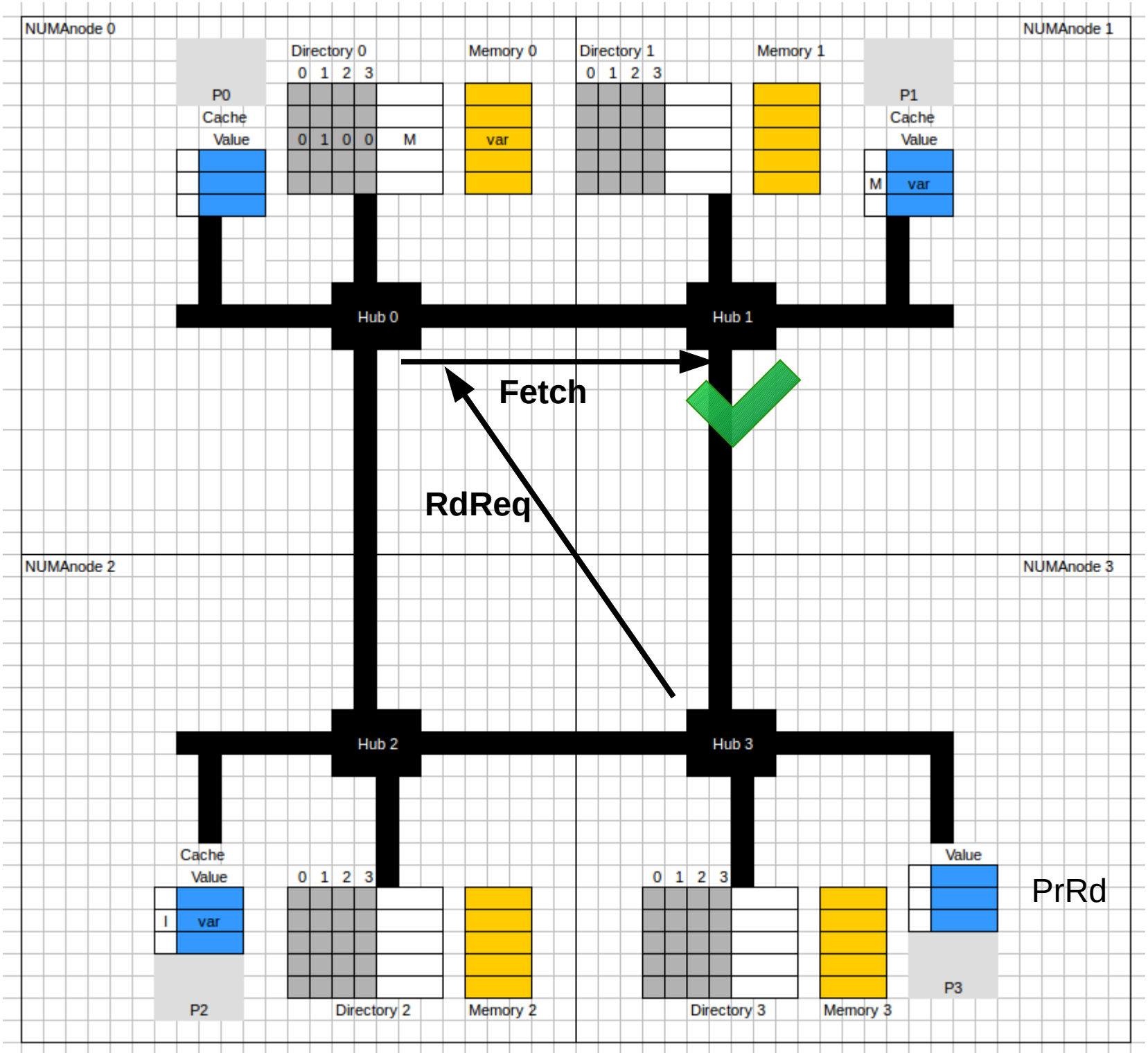
After the previous access, the processor in NUMAnode3 reads the same variable *var*. Which of the following coherence actions do NOT occur?

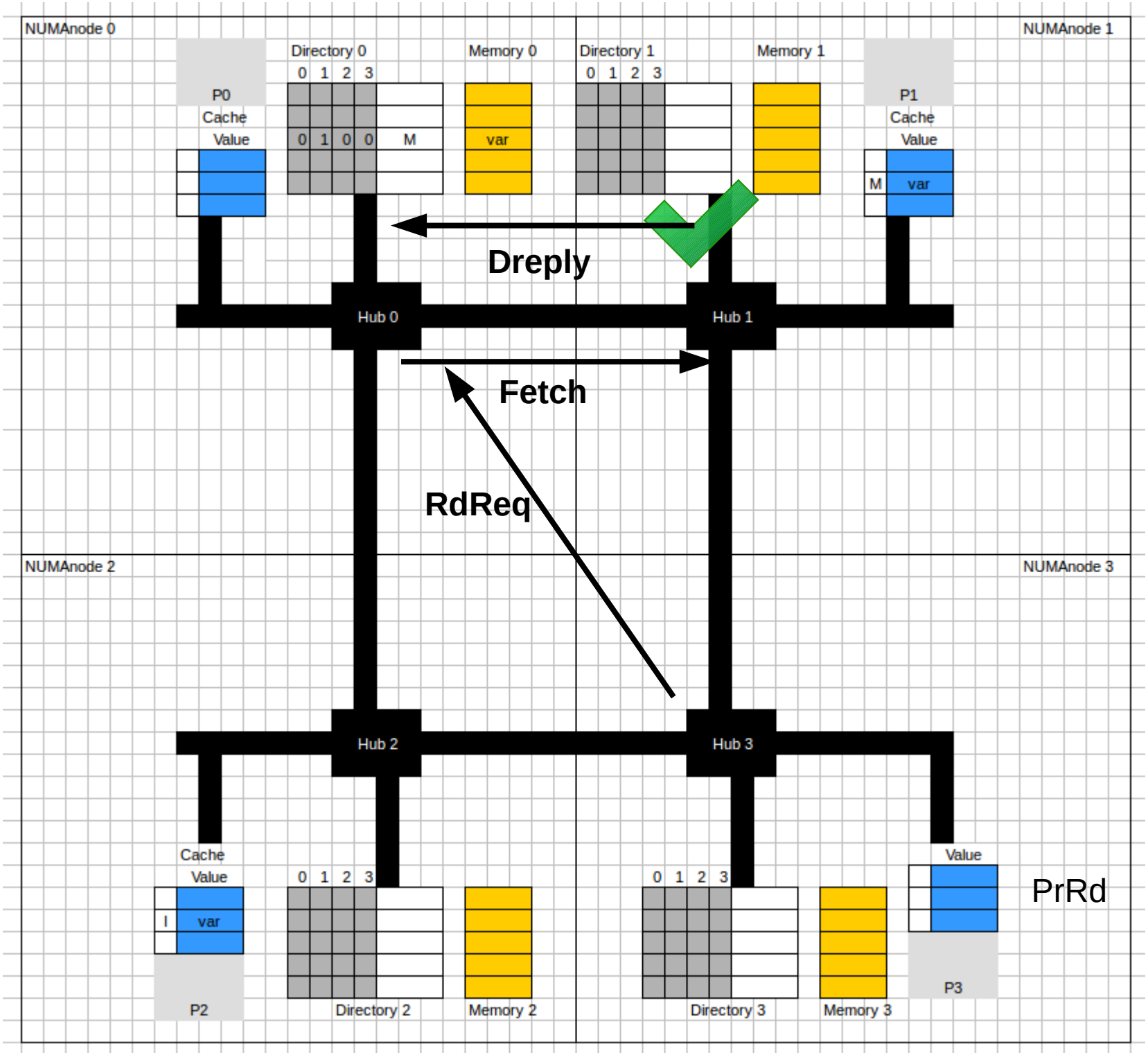
Trieu-ne una o més:

- [ ] The processor in the local node (NUMAnode3) issues PrRd.
- [ ] The local node issues a Fetch command to the home node (NUMAnode0) to obtain the line containing *var*.
- [ ] The home node then issues a Fetch command to the remote NUMAnode1 in order to get the most up-to-date copy of the requested memory line.
- [ ] The remote node issues a Dreply command with the only valid copy of the line in the whole system.


NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 0 0 M
var
P1
M var
Hub 0
Hub 1
RdReq
Hub 2
Hub 3
I var
P2
Directory 2
Memory 2
Directory 3
Memory 3
P3
PrRd


NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 0 0 M
var
P1
M var
Hub 0
Hub 1
Hub 2
Hub 3
FETCH
RdReq
Fetch asks remote (owner) node for a copy of line???
I var
P2
Directory 2
Memory 2
Directory 3
Memory 3
P3
PrRd


NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 0 0 M
var
P1
M var
Hub 0
Hub 1
Fetch
RdReq
Hub 2
Hub 3
I var
P2
Directory 2
Memory 2
Directory 3
Memory 3
P3
PrRd


NUMAnode 0
NUMAnode 1
NUMAnode 2
NUMAnode 3
Directory 0
Memory 0
Directory 1
Memory 1
0 1 2 3
P0
Cache
Value
0 1 0 0 M
var
P1
M var
Dreply
Hub 0
Hub 1
Fetch
RdReq
Hub 2
Hub 3
Cache
Value
I var
P2
Directory 2
Memory 2
Directory 3
Memory 3
Value
P3
PrRd

After the previous access, the processor in NUMAnode3 reads the same variable *var*. Which of the following coherence actions do NOT occur?

Trieu-ne una o més:

- [ ] The processor in the local node (NUMAnode3) issues PrRd.
- [ ] The local node issues a Fetch command to the home node (NUMAnode0) to obtain the line containing *var*.
- [ ] The home node then issues a Fetch command to the remote NUMAnode1 in order to get the most up-to-date copy of the requested memory line.
- [ ] The remote node issues a Dreply command with the only valid copy of the line in the whole system.