What is claimed :

1. A computing device, comprising:

a plurality of processing units;

a plurality of N-level caches;

an (N+i)-level cache that is associated with two or more of the plurality of N-level caches, and that is configured as a backing store for the plurality of N-level caches; and

control logic that configures the computing device to perform at least the following:

detect an influx to a first N-level cache of plurality of N-level caches, the influx comprising data stored at a memory location; and

based on detecting the influx to the first N-level cache, check the (N+i)-level cache to determine if the data for the memory location has been previously logged on behalf of a second processing unit, and performing one of:

when the data for the memory location has been previously logged on behalf of the second processing unit, cause the data for the memory location to be logged on behalf of the first processing unit by reference to log data that was previously logged on behalf of the second processing unit; or

when the data for the memory location has not been previously logged on behalf of the second processing unit, cause the data for the memory location to be logged by value on behalf of the first processing unit.

2. The computing device of claim 1, wherein checking the (N+i)-level cache to determine if the data for the memory location has been previously logged on behalf of the second processing unit comprises one or more of:

determining if a cache line in the (N+i)-level cache that corresponds to the memory location has one or more accounting bits set;

determining if a cache line in the (N+i)-level cache that corresponds to the memory location is stored in a way that corresponds to a logged processing unit; or

determining if a cache line in the (N+i)-level cache that corresponds to the memory location has associated cache coherence protocol (CCP) state data that is usable to determine that the cache line has been logged.

3. The computing device of claim 1, wherein i equals 1, such that the (N+i)-level cache comprises an (N+l)-level cache.

4. The computing device of claim 1, wherein i equals 2, such that the (N+i)-level cache comprises an (N+2)-level cache.

5. The computing device of claim 1, wherein:

i equals 1, such that the (N+i)-level cache comprises an (N+l)-level cache;

the computing device also comprises an (N+2)-level cache that is configured as a backing store for the (N+l)-level cache; and

checking the (N+l)-level cache to determine if the data for the memory location has been previously logged on behalf of the second processing unit comprises:

determining that no cache line in the (N+l)-level cache corresponds to the memory location; and

checking the (N+2)-level cache to determine if the data for the memory location has been previously logged on behalf of the second processing unit.

6. The computing device of claim 1, wherein causing the data for the memory location to be logged on behalf of the first processing unit by reference to log data that was previously logged on behalf of the second processing unit comprises one or more of:

logging an address of the memory location;

logging an address of the memory location and ordering data;

logging a reference to a cache way;

logging a reference to a cache way and ordering data;

logging the second processing unit as a previous owner of a cache line corresponding to the memory location; or

logging cache coherence protocol (CCP) data referencing the second processing unit.

7. The computing device of claim 1, wherein causing the data for the memory location to be logged on behalf of the first processing unit comprises delaying the logging based on availability of one or both of processor or memory resources.

8. The computing device of claim 1, wherein delaying the logging comprises invalidating a cache line to retain the data for the memory location for the delayed logging.

9. The computing device of claim 1, wherein the control logic also configures the computing device to perform at least the following:

evict a first cache line in the first N-level cache that corresponds to the memory location, while retaining a second cache line in the (N+i)-level cache that corresponds to the memory location; and

based on detecting a subsequent influx to the first N-level cache, the subsequent influx also comprising the data stored at the memory location, cause the subsequent influx to be logged by reference based on presence of the second cache line.

10. The computing device of claim 1, wherein the control logic also configures

the computing device to perform at least the following:

evict a first cache line in the first N-level cache that corresponds to the memory location, while retaining a second cache line in the (N+i)-level cache that also corresponds to the memory location;

detect a subsequent influx to the first N-level cache based on additional code execution at the first processing unit, the subsequent influx also comprising the data stored at the memory location; and

based at least on detecting the subsequent influx to the first N-level cache, and based at least on presence of the second cache line, determine that that the subsequent influx need not be logged.

11. A method for trace recording based on recording an influx to a lower-level cache by reference to prior log data based on knowledge of one or more upper-level caches, the method being implemented at a computing device that includes (i) a plurality of processing units, (ii) a plurality of N-level caches, and (iii) an (N+i)-level cache that is associated with two or more of the plurality of N-level caches, and that is configured as a backing store for the plurality of N-level caches, the method comprising:

detecting an influx to a first N-level cache of plurality of N-level caches, the influx comprising data stored at a memory location; and

based on detecting the influx to the first N-level cache, checking the (N+i)-level cache to determine if the data for the memory location has been previously logged on behalf of a second processing unit, and performing one of:

when the data for the memory location has been previously logged on behalf of the second processing unit, causing the data for the memory location to be logged on behalf of the first processing unit by reference to log data that was previously logged on behalf of the second processing unit; or

when the data for the memory location has not been previously logged on behalf of the second processing unit, causing the data for the memory location to be logged by value on behalf of the first processing unit.

12. The method of claim 11, wherein checking the (N+i)-level cache to determine if the data for the memory location has been previously logged on behalf of the second processing unit comprises one or more of:

determining if a cache line in the (N+i)-level cache that corresponds to the memory location has one or more accounting bits set;

determining if a cache line in the (N+i)-level cache that corresponds to the memory

location is stored in a way that corresponds to a logged processing unit; or

determining if a cache line in the (N+i)-level cache that corresponds to the memory location has associated cache coherence protocol (CCP) state data that is usable to determine that the cache line has been logged.

13. The method of claim 11, wherein i equals 1, such that the (N+i)-level cache comprises an (N+l)-level cache.

14. The method of claim 11, wherein i equals 2, such that the (N+i)-level cache comprises an (N+2)-level cache.

15. The method of claim 11, wherein :

i equals 1, such that the (N+i)- level cache comprises an (N+1)-level cache;

the computing device also comprises an (N+2)-level cache that is configured as a backing store for the (N+1)-level cache; and

checking the (N+1)-level cache to determine if the data for the memory location has been previously logged on behalf of the second processing unit comprises :

determining that no cache line in the (N+1)-level cache corresponds to the memory location; and

checking the (N+2)-level cache to determine if the data for the memory location has been previously logged on behalf of the second processing unit.

16. The method of claim11, wherein causing the data for the memory location to be logged on behalf of the first processing unit by reference to log data that was previously logged on behalf of the second processing unit comprises one or more of :

logging an address of the memory location;

logging an address of the memory location and ordering data;

logging a reference to a cache way;

logging a reference to a cache way and ordering data;

logging the second processing unit as a previous owner of a cache line corresponding to the memory location; or

logging cache coherence protocol (CCP) data referencing the second processing unit.

17. The method of claim 11, wherein causing the data for the memory location to be logged on behalf of the first processing unit comprises delaying the logging, including invalidating a cache line to retain the data for the memory location for the delayed logging.

18. The method of claim 11, further comprising :

evicting a first cache line in the first N-level cache that corresponds to the memory location, while retaining a second cache line in the (N+i)-level cache that corresponds to the memory location; and

based on detecting a subsequent influx to the first N-level cache, the subsequent influx also comprising the data stored at the memory location, causing the subsequent influx to be logged by reference based on presence of the second cache line.

19. The method of claim 11, further comprising :

evicting a first cache line in the first N-level cache that corresponds to the memory location, while retaining a second cache line in the (N+i)-level cache that also corresponds to the memory location;

detecting a subsequent influx to the first N-level cache based on additional code execution at the first processing unit, the subsequent influx also comprising the data stored at the memory location; and

based at least on detecting the subsequent influx to the first N-level cache, and based at least on presence of the second cache line, determining that that the subsequent influx need not be logged.

20. A computer program product comprising one or more hardware storage devices having stored thereon computer-executable instructions that configure a computing device that includes (i) a plurality of processing units, (ii) a plurality of N-level caches, and (iii) an (N+i)-level cache that is associated with two or more of the plurality of N-level caches, and that is configured as a backing store for the plurality of N-level caches, to perform at least the following:

detect an influx to a first N-level cache of plurality of N-level caches, the influx comprising data stored at a memory location; and

based on detecting the influx to the first N-level cache, check the (N+i)-level cache to determine if the data for the memory location has been previously logged on behalf of a second processing unit, and performing one of:

when the data for the memory location has been previously logged on behalf of the second processing unit, cause the data for the memory location to be logged on behalf of the first processing unit by reference to log data that was previously logged on behalf of the second processing unit; or

when the data for the memory location has not been previously logged on behalf of the second processing unit, cause the data for the memory location to be logged by value on behalf of the first processing unit.