

Review Questions

Instructions: You are not required to turn in answers to these review questions, but you should do them on your own after you read this section's papers to test your knowledge. Use Piazza or the TAs if you have questions or get stuck on any of these.

Global Memory Management

- 1) What is the goal of the global memory management? How does the memory management policy achieve this goal?
- 2) Assume 4 nodes (N1 through N4) each with 16 physical page frames. Initially none of the nodes have any valid pages in their physical memory. N1 and N2 each make a sequence of page accesses to 32 distinct pages all from the disk. N3 and N4 are idle. What is the state of the cluster (i.e. local and global caches at each node) at the end of the above set of accesses?
- 3) How is the minimum age of a page (cluster wide) determined? Explain the algorithm to determine the MinAge.
- 4) Why is node failure not an issue in global memory management?
- 5) Explain the functionality of each of the data structures (PFD, GCD, POD) in the global memory management scheme.
- 6) In the actual implementation, how does each node collect the age information for its pages?
- 7) With reference to Table 1, explain why GCD processing time is considerably less when the page is non-shared.
- 8) Summarize the conditions under which GMS can hope to significantly improve the performance for file accesses compared to vanilla NFS.

TreadMarks

- 1) Distinguish between memory coherence and memory consistency model.
- 2) Explain the following terms with code samples to illustrate the point:
 - a) data race
 - b) false sharing
- 3) Explain the difference between sequential consistency and release consistency with code samples.
- 4) Explain the difference between eager and lazy release consistency with code samples. What are the pros and cons of each approach?
- 5) How does the TreadMarks DSM solve the false sharing problem? Explain the protocol details.

- 6) Given that TreadMarks is a user level DSM package, What support is expected from the Operating System for implementing it?
- 7) Develop the data structures and pseudo code for the TreadMarks DSM. Take into account the distributed lock manager and the protocol actions needed commensurate with the memory consistency model and the coherence mechanism of TreadMarks.
- 8) What are the sources of overhead in the implementation of TreadMarks?
- 9) What is the need for garbage collection in the TreadMarks implementation? Explain.
- 10) Consider a TreadMarks DSM system with 3 pages A, B, and C, and lock L. The following accesses happen in the order shown. Assume initially the lock L is available for N1 and that the pages A, B, and C are with nodes N1, N2, and N3, respectively.

Node N1:

```
lock(L);
  modify A;
unlock(L);
```

Node N2:

```
lock(L);
  modify B;
unlock(L);
```

Node N3:

```
lock(L);
  read A;
  read B;
  modify C;
unlock(L);
```

Show the "when" and "what" actions take place in the different nodes in terms of (a) messages exchanged between the nodes, (b) twin and diff creations at the nodes, (c) invalidations of pages at the nodes, and (d) propagation of diffs between the nodes.

Serverless Network File System

- 1) What are the technological enablers justifying a serverless NFS?
- 2) xFS builds on three prior technologies: RAID, Zebra LFS, and multiprocessor cache consistency. Discuss the limitations in each of these technologies with respect to realizing a scalable serverless NFS.
- 3) In a centralized file system, the server performs the functions of managing the data blocks, metadata for the files, server-side file cache, and consistency of datablocks of files cached by

multiple clients. Discuss how these functions are carried out in xFS.

- 4) What is the advantage of the stripe group concept used in xFS?
- 5) Explain the idea of log cleaning and why it is needed. Discuss the principle and details of the distributed log cleaners employed in xFS.
- 6) The paper claims: "For instance, in a 32 node xFS system with 32 clients, each client receives nearly as much read or write bandwidth as it would see if it were the only active client."
Explain this statement.

Coda file system

-
- 1) "Coda emulates Unix semantics": Explain what this means.
 - 2) Explain "availability" and how does Coda provide availability. What guarantees does the user of Coda get with respect to availability.
 - 3) Explain the mechanism in Coda for availability and scalability.
 - 4) Explain the "callback" mechanism in Coda.
 - 5) Explain the "optimisitic replication" mechanism of Coda. How does it affect the semantics of file usage by the client of the file system?