

Tutorial 8
Notes on Solutions

Consistency and replication

1. Explain in your own words what the main reason is for actually considering weak consistency.

Weak consistency models come from the need to replicate for performance. However, efficient replication can be done only if we can avoid global synchronisation, which, in turn, can be achieved by loosening consistency constraints.

2. Consider each of the models of distributed shared memory in (a)-(b) and explain how you would classify them according to the models of consistency and replication in the lecture. Will any have consistency problems?

- a. All processors keep a counter representing global time. Every time a write occurs, all processors have to be signalled, and stalled, until the write is complete. A copy is broadcast to all caches, along with the clock counter of the writer, and replaces any existing copy. If it is not currently cached, the new value is discarded if the processor's local clock counter is ahead of the broadcast time. If it's not, it buffers the new value until its clock has caught up, at which point the new value is cached.

Strict consistency. There will be no consistency problems but what are the performance implications of such a scheme?

- b. Before a write can occur, the memory containing the data to be written has to be flushed from memory of all other processors in the system. Any subsequent reads will result in a miss, which has to be serviced by modifying the written copy to sharable, and copying it over the network to the reader.

Entry consistency. Although we haven't mentioned locks specifically, this sort of behaviour needs something to trigger this behaviour. Without locks this isn't entry consistency so the example doesn't exactly fit the definition, but it's the closest.

3. How do you think a large free email service with a worldwide presence, such as yahoo operates? Assume that replication to a local server is a useful strategy to minimize latency, but that a fair fraction of users are highly mobile. Consider a range of potential consistency and replication models, and explain which you think is the best, and which you consider to be unsuitable. Consider possibilities of both *client-centric* and *data-centric* solutions.

This situation fits the client-centric model best, because you can think of the problem in terms of a consistent view for a mobile client. Given that a client can't move very fast, a

primary-based local write protocol is an option (the primary moves to where the client is), but practicality would depend on how big the mail was. You could discuss various options with positive and negative points.

4. You plan to build a distributed system in Java, and would like to maximize leverage from standard Java APIs. What extensions to standard Java APIs would you need to support replication?

A variety of issues could be considered, for example, RMI could be extended to locate the nearest replica, rather than a fixed remote reference. You need to think about whether the extensions are at the object level or in an underlying middleware layer.

5. How could you use read-your-writes consistency to ensure that you see the latest copy of a web page you've just updated yourself? What would you have to add to a standard web browser and cache on a personal computer?

You would need to inform the cache of updates you make to the server, so that as those changes were made, the older version of the page (or other content) was flushed from the cache.

6. A file is replicated on 6 servers.
 - a. List all the combinations of read and write quorums permitted by Gifford's scheme.

Starting from the write quorum N_W which is more constrained (must be $> N/2$), allowable values are 4, 5 and 6. For each value of N_W , we must have $N_R + N_W > N$ so to take one case, for $N_W = 4$, we can have N_R values of 3 and above.

- b. Do all of these combinations make sense?

Although the definition in the book does not put an upper bound on N_R or N_W , there is no point in having $N_R + N_W > N + 1$. This will result in unnecessary communication.

- c. What is the significance of a read quorum of 1?

You only need locate one copy to read, but writers need to communicate with all copies. This is valuable in a case where writes are relatively rare.

7. What kind of consistency would you use to implement a web-based flight booking system, with multiple airlines, accessible to users around the world? Could more than one apply to different parts of the system?

For a start, this looks like a data rather than client centric problem because we are not specifically focusing on what happens when the same client uses the system in different parts of the world. We need to have some consistency model in which ordering of writes is preserved so we can resolve races (e.g., to book a specific seat: if there is a conflict, all reads of the resulting database should see the writes in the same order). Sequential consistency fits this scenario: all writes are seen in the same order. By analysing the problem differently, you could make a case for other models.

8. For each of the following, explain why the example does not meet the specific consistency requirements:

These are good exercises in applying the definitions -- you should be able to do all of them without too much effort.

- a. Figure (b) strict consistency on slide 11.
 - b. Figure (b) sequential consistency on slide 12.
 - c. Figure (a) causal consistency on slide 15.
 - d. Figure (b) monotonic reads on slide 28.
 - e. Figure (b) monotonic writes on slide 30.
 - f. Figure (b) read-your-writes on slide 31.
9. Explain general issues in choosing between client-centric and data-centric consistency models.

The big-picture issue is that client-centric models look at what happens when the client moves, whereas data-centric models look at what happens when a different process looks at the data from a different site. You can expand the discussion to include examples to make the issues concrete.