Live Migration of VMs

1. What constitutes "liveness" during live migration of virtual machines?

2. What are the key states of a VM transferred during live migration?

3. What are the metrics of performance in live migration of VMs?

4. What are the tradeoffs between different performance metrics when using each migration technique?

5. How do different VM migration techniques work? How do they differ?
   1. Stop-and-copy
   2. Pure demand paging
   3. Post-copy
   4. Pre-copy
   5. Hybrid pre/post copy

6. What are some optimization techniques you can use to speed up pre-copy? Post-copy? Both?

7. Under what situation would you use each migration technique?

8. What is dirty-page tracking? Why is it needed? How does it work? What are its overheads?

9. How do pre-copy and post-copy live VM migration mechanisms compare against each other in terms of (a) downtime, (b) total migration time, and (c) performance of applications in the VM during migration?   Explain your answers.

10. Consider pre-copy live VM migration
    A. Under what situation would it have excessively long total migration time and downtime?
    B. To solve this problem, describe two optimization techniques (i.e. improvements to pre-copy — NOT post-copy, record-replay or other completely different techniques).

11. [Open-ended question] Using techniques learnt in live VM migration topic, provide the design of a system for maintaining a "hot-standby" replica of a running virtual machine (VM). In other words, if the VM fails, then its replica (on a different machine) should "instantly" take over the execution from exactly the point at which the original VM failed, all without losing correctness. In your design, also explain (a) how you would maintain a consistent replica before failure and (b) switch over to the replica upon failure.

12. [Open-ended question] Suppose you are asked to develop a system for

simultaneous live migration of multiple VMs from one physical machine to another within a LAN. You need to reduce the total migration time (from start of migration of first VM to end of migration of last VM) and the network traffic due to migration, while not unnecessarily increasing the downtime. All VMs run the same guest OS (say Linux), but not necessarily the same applications. Develop a design for your live migration system in as much detail as you can. Justify the design choices you make.