LTAT.06.007 Distributed Systems

Seminar 11 - Fault tolerance

Mohan Liyanage, PhD

Tartu, Estonia May 1, 2020
Quiz 11

Questions are based on the Fault tolerance lecture

Short answers

Link to quiz – https://tinyurl.com/y9qpusep

Quiz will be available until April-27-2020:23.59
1. Name the requirements of dependable systems

• Availability
  • Probability that the system is operating correctly at any given moment and is available to perform its functions on behalf of its users

• Reliability
  • Probability that the system can run continuously without a failure for an interval of time

• Safety
  • When a system temporarily fails to operate correctly, nothing catastrophic should happen

• Maintainability
  • Refers to how easily a failed system can be repaired
2. What do you mean by the intermittent fault?

An intermittent fault occurs, then vanishes of its own accord, then reappears, and so on. A loose contact on a connector will often cause an intermittent fault. Intermittent faults cause a great deal of aggravation because they are difficult to diagnose. Typically, when the fault doctor shows up, the system works fine.
3. The key technique for masking faults is to use redundancy. Name there possible kinds of redundancy

- Information redundancy
  - Extra bits are added to allow recovery from garbled bits. For example, a Hamming code can be added to transmitted data to recover from noise on the transmission line.

- Time redundancy
  - An action is performed, and then, if need be, it is performed again. Transactions use this approach. If a transaction aborts, it can be redone with no harm. Another well-known example is retransmitting a request to a server when lacking an expected response. Time redundancy is especially helpful when the faults are transient or intermittent.

- Physical redundancy
  - Extra equipment or processes are added to make it possible for the system as a whole to tolerate the loss or malfunctioning of some components. Physical redundancy can thus be done either in hardware or in software. For example, extra processes can be added to the system so that if a small number of them crash, the system can still function correctly. In other words, by replicating processes, a high degree of fault tolerance may be achieved.
4. The key approach to tolerating a faulty process is to organize several identical processes into a group. Name two types of such groups that reflect their internal structure

- **Flat groups**
  - All processes are equal
  - There is no distinctive leader and all decisions are made collectively
  - Symmetrical and has no single point of failure
  - If one of the processes crashes, the group simply becomes smaller, but can otherwise continue
  - Decision making is more complicated

- **Hierarchical groups**
  - One process is the coordinator and all the others are workers
  - When a request for work is generated it is sent to the coordinator and then decides which worker is best suited to carry it out
  - Loss of the coordinator brings the entire group to a grinding halt
5. What do you mean by the degree of fault tolerance in a fault tolerant group?

When a group can mask any k concurrent member failures then k is called as the degree of fault tolerance.
6. Paxos is a widely adopted consensus algorithm for many practical distributed systems. Name three realistic assumptions that the Paxos algorithm based on

- The distributed system is partially synchronous (in fact, it may even be asynchronous)
- Communication between processes may be unreliable, meaning that messages may be lost, duplicated, or reordered
- Messages that are corrupted can be detected as such (and thus subsequently ignored)
- All operations are deterministic: once an execution is started, it is known exactly what it will do
- Processes may exhibit crash failures, but not arbitrary failures, nor do processes collude
7. What is the CAP Theorem?

- CAP Theorem: Any networked system providing shared data can provide only two of the following three properties:
  - C: consistency, by which a shared and replicated data item appears as a single, up-to-date copy
  - A: availability, by which updates will always be eventually executed
  - P: Tolerant to the partitioning of process group (e.g., because of a failing network)
8. Name at least three different classes of failures that can occur in RPC systems

1. The client is unable to locate the server
   • Server may be down or binding fails due to interface mismatch
   • Error raises an exception (error signal)
   • Not all languages support exceptions and error signals
   • Compromises transparency

2. The request message from the client to the server is lost
   • Start a timer when message is sent, resend if timer expires

3. The server crashes after receiving a request

4. The reply message from the server to the client is lost
   • Client assigns each request a sequence number, server filters requests that have already been carried out

5. The client crashes after sending a request
9. When considering the situation of server recovery, there are three events that can happen at the server. i. Send the completion message (M) ii. Complete the processing of the document (P) iii. Crash (C). What are the different orderings of events if the server based on the strategy P -> M?

P -> M -> C
P -> C -> M
C -> P -> M
10. This figure shows the finite state machine for a participant in the two-phase commit protocol. Name a, b and c accordingly.
11. The whole idea of error recovery is to replace an erroneous state with an error-free state. According to that, what do you mean by the backward error recovery?

- Backward recovery
  - Bring the system from its present erroneous state back into a previously correct state
  - It will be necessary to record the system’s state from time to time, and to restore such a recorded state when things go wrong
12. In a distributed snapshot, if a process P has recorded the receipt of a message, then there should also be a process Q that has recorded the sending of that message. To recover after a process or system failure requires that we construct a consistent global state from local states as saved by each process. This figure shows a collection of checkpoints and possible recovery lines. Please explain the recovery line 2 is not suitable compared to the recovery line 1.

Recovery line 1 has the consistence checkpoints from both P1 and P2.
Thank You !!