CSC0056 Fall 2020 Solution

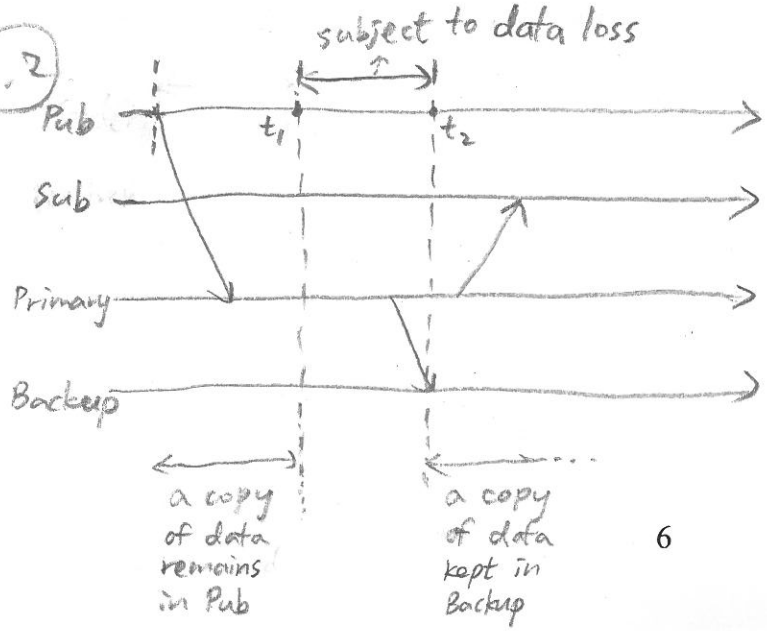
Name: _____ Student ID: _____ Homework 6

(1.1) Absolute time consistence specifies whether or not an event is "temporally valid". See Section 3.2 in reference [8] (↑ on course website) for detail. The notion is useful because often an application should only process events that are still "fresh" in time. (Consider an application that tracks the distance between cars, for example.). Shedding inconsistent events give benefits because it could save CPU%.

(1.2) In this way, the workers will not be delayed much by the movers of the same priority level, and this may reduce the response time of event processing. Further, the mover will not be delayed much by the workers of some lower priority level, and this may reduce the undesirable chain-effect of delay added to the workers of the same priority level (consider, for example, that typically one uses mutex to protect critical section to prevent racing).

(2.1) Passive replication may save CPU% because it does not process data in fault-free operation.

(2.2)



With a longer interpublish time, the publisher may keep longer a copy of date it sent, and this may reduce the likelihood of data loss.
Essentially, t_1 would move further to the right on the timeline and thus the time gap between t_1 and t_2 are reduced.

Name: Solution Student ID: Homework 6

(3.1) File-transfer date flow:

low demand on latency, high demand on data loss

Video-streaming data flow:

high demand on latency, low demand on data loss

(3.2) No. The data rate permitted is $\frac{W}{d}$ with W being the window size and d the round-trip delay. The round-trip delay is often out of control of a service provider.

Literature Reading

① There are several differences. Here I just list some:

(1) In FRAME, selective replication is an offline decision (pre-determined); in ARREC, adaptive replication is an online decision.

(2) FRAME does not replicate in batch; ARREC does.

(3) FRAME addresses cloud/edge messaging differentiation; ARREC focuses on edge computing.

② So that a system may skip unnecessary replication in Step 4.

③ This is an interesting phenomenon. Having frequent "attempts" of replication will cause fewer "executions" of replication, because a longer deadline for replication (see Lemma 2) would allow more pending data to be skipped.