Lecture 13

Friday, 20 October 2023 08:44

consistency is only relevant for replication, not for partitioning
because there is exactly one copy of each part of the data

R_1 only one party can hold a lock on an object

R_2 the order of operations follows the original order of the program

R_3 the order of operations follows the global (real-time) ordering

actual executions that satisfy R_1 and R_2 are sequentially consistent

actual executions that satisfy R_1 and R_3 are called linearizable

data store is linearizable if all executions are linearizable

operations overlap in time → concurrent

operations do not overlap in time → serial

causal consistency : writes that are potentially causally related must be seen by all processes in the same order

concurrent writes may be seen in different order by different machines

eventual consistency : in absence of further updates, all replicas eventually have the same state

client-centric models

monotonic read consistency — if a process reads the value of a data item x, any successive read operation on x by that process will always return the same value or a more recent one

providing a timestamp with the value might help

or version number

monotonic write consistency — a write operation on a data item x is completed before any successive write operation on x by the same process

read your writes consistency — the effect of a write operation by a process on data item x will always be seen by a successive read operation on x by the same process

write follow reads consistency — a write operation by a process on a data item x following a previous read operation on x by the same process is guaranteed to take place on the same or a more recent value of x that was read

replica management

placement strategies

permanent replicas

server-initiated replicas

client-initiated replicas

statically determined number of replicas

e.g. Docker replica number changes (dynamically)

client-side caching

update protocols

pull- and push-based

client

server

protocol classification

what information is disseminated

which party takes initiative for dissemination

what communication primitives are used to disseminate

when is completion of an operation reported to the client

protocol classes

passive replication → primary-read, primary-write

active replication → write

gossip-based → local-read, local-write

all these protocols are push-based with respect to updates

passive replication

1. request phase
2. coordination phase
3. execution phase
4. agreement phase
5. response phase