CSE138 Lecture 10 this time: 1 - exam review -recap: Strong consistency
-other (weaker) Consistency policies: Causal consistency
 FIFO consistency
 "read-your-writes" consistency - mechanisms for implementing - primary-backup - chain replication - lateray and twoughput RZ PLIT(X,2) violation of totally-ordered message delivery leads message delivery leads to.
violation of strong consistence informally: clients rout tell the data is replicated. C. * PUT(x, 5) ?? RYW (read your writes) violation BR2 SO bank 1 R, \$0 ۶c, employee deposit 50 \$50 4-ack +50 withdraw ud 140 \$10 FIFO violation! of FIFO consistency violation FIFO consistency:

writes issued by a single process

one seem by all processes

in the order they were issued. (A message would need to be delivered to be "seen") RZ thick client B Rz G R BR, \$0 dep 100 balan ce? \$ 100 wd \$70 Problem: the withdrawal causally depends on the deposit, but Rz only saw the withdrawal! violation of causal consistency. Causal consistency: parentially & happens-before writes that are (causally related) must be seen by all processes in the same (causal) order. $A \rightarrow B$ send (m,) -> send (mz) $m, \rightarrow m_z$ "potentially causally related" means related by happens-before [linearizability Stronger "Strong" consistency
sequential consistency Cousal consistency what you're impumenting (nopofully!) FIFO consistency consistency weaker RYW

Mechanisms for implementing strong consistency. primary backup replication PUT(x, 5) backup Z GET(x) commit point (the moment at which the write becomes Pros and Cons: Strong consistency! (pro) - Primary is a bottleneck! (con) Replication should ideally give you I fault tolerance dividing up the work ("scalability") X data locality primary-backup replication is only 1 for 3.