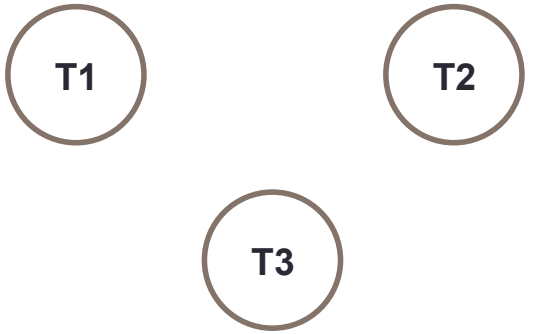
DISTRIBUTED TRANSACTIONS

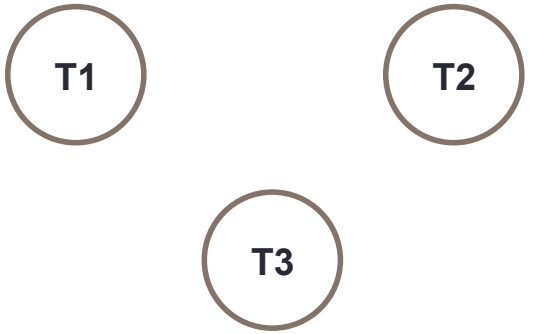
Principles of Computer System Design, Block 2

Exercise 1: Distributed transactions

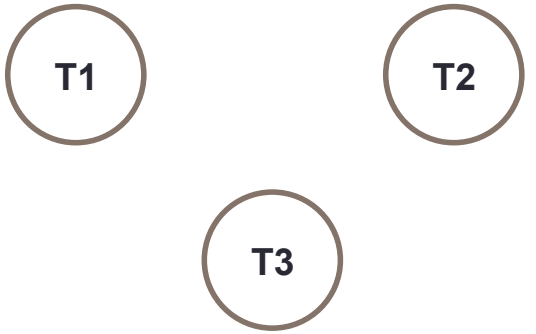
NODE 1	T1	R(X)					C?
	T2		R(E)				
	T3					W(E)	



NODE 2	T1		R(Y)		W(Y)		C?
	T2			R(F)	W(F)		
	T3	R(G)				W(G)	



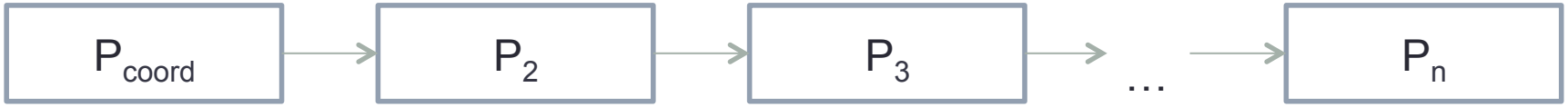
NODE 3	T1			R(Z)		W(Z)	C?
	T2				R(H)	W(H)	
	T3		R(H)				



Exercise 2: True or False

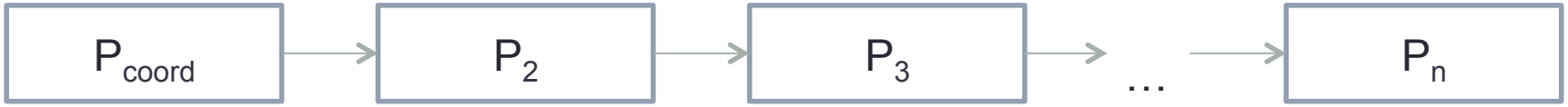
a) A fault is an unhandled error that causes an interface to break its contract.
b) A fail-fast component is one that immediately stops when an error occurs in order to prevent error propagation.
c) Read any, write all synchronous replication can be seen as a special case of state machine replication.
d) Asynchronous replication is preferred to synchronous replication, because it allows operations to return faster while offering the same level of consistency.

Exercise 3: Linear 2PC (1)



- A process P waits for a message from its left neighbor. If P receives YES and its own vote is YES, then P forwards YES to its right neighbor. Otherwise – P forwards NO.
- The rightmost process P_n will have all the information it needs to make a decision. If it receives a YES and its own vote is YES, then the decision is COMMIT, otherwise the decision is ABORT. Then it forwards the decision to the left neighbor.
- Each process that receives the decision message decides accordingly and then forwards that message to its left neighbor until the coordinator receives the message.

Exercise 3: Linear 2PC (2)



a) If there are n > 2 participants, how many messages are sent when performing centralized 2PC and linear 2PC?

b) Assume a communication system with unlimited bandwidth. Assume further, that the processor time necessary to process a 2PC-algorithm is zero, both for the coordinator and the participants. If message transmission time is *t*, how much time at least will it take when performing centralized 2PC and linear 2PC?

c) What should happen if one process fails in the linear 2PC?