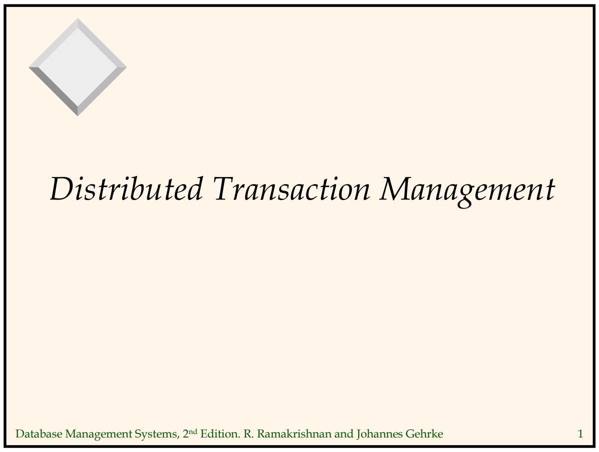
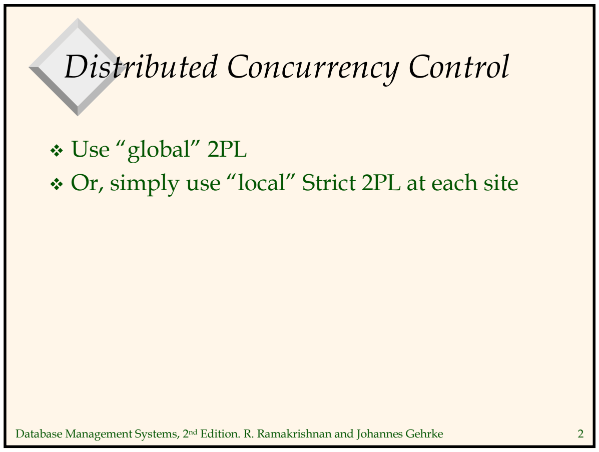
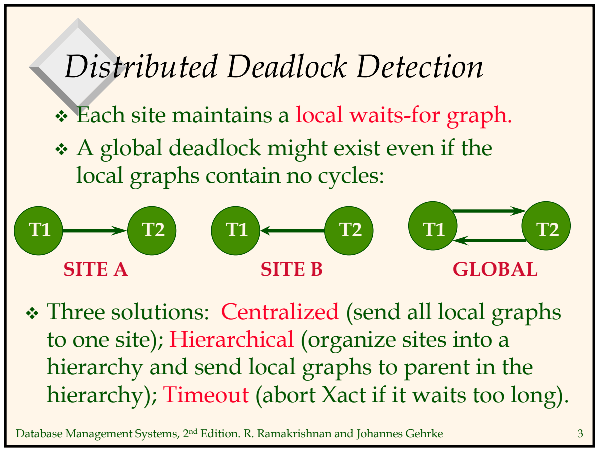


Distributed Transaction Management



Distributed Concurrency Control

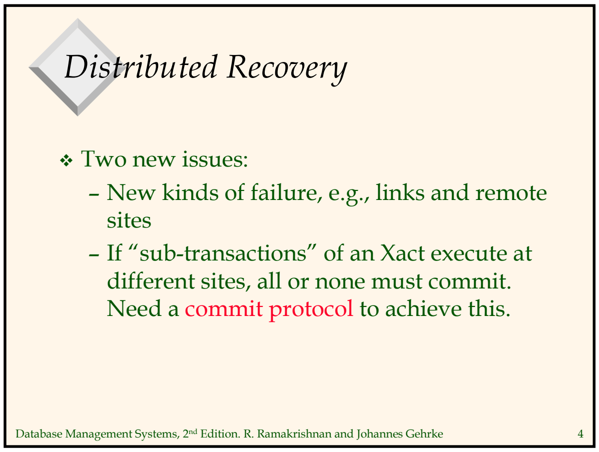
- Use “global” 2PL
- Or, simply use “local” Strict 2PL at each site



Distributed Deadlock Detection

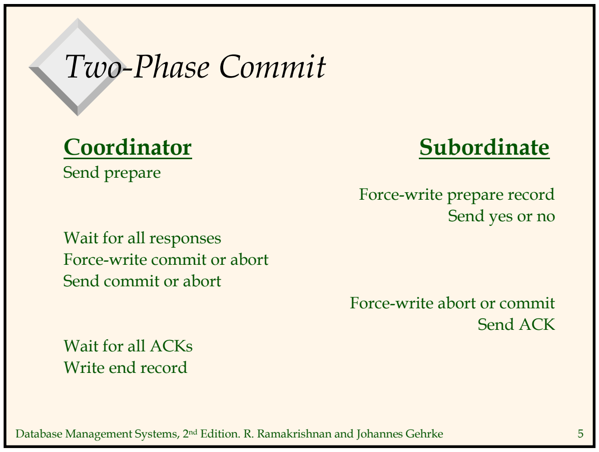
- Each site maintains a local waits-for graph.
- A global deadlock might exist even if the local graphs contain no cycles:

- Three solutions: Centralized (send all local graphs to one site); Hierarchical (organize sites into a hierarchy and send local graphs to parent in the hierarchy); Timeout (abort Xact if it waits too long).



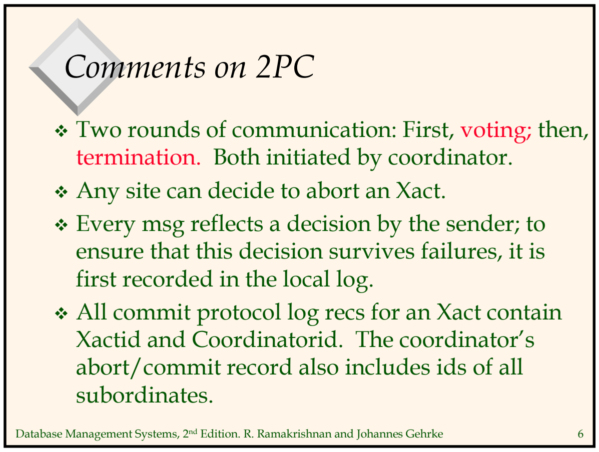
Distributed Recovery

- Two new issues:
  - New kinds of failure, e.g., links and remote sites
  - If "sub-transactions" of an Xact execute at different sites, all or none must commit. Need a commit protocol to achieve this.



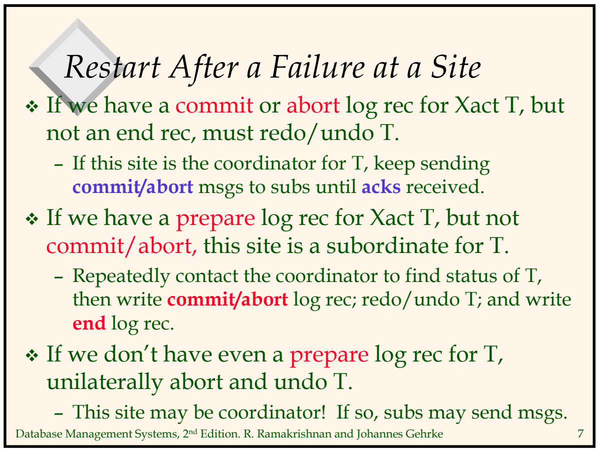
Two-Phase Commit

Coordinator	**Subordinate**
Send prepare	
	Force-write prepare record Send yes or no
Wait for all responses Force-write commit or abort Send commit or abort	
	Force-write abort or commit Send ACK
Wait for all ACKs Write end record	



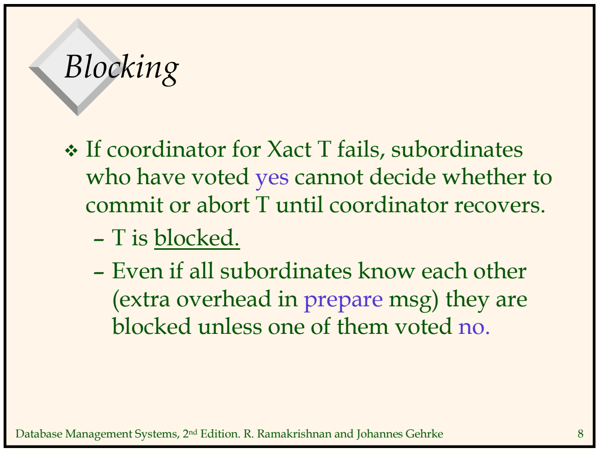
Comments on 2PC

- Two rounds of communication: First, voting; then, termination. Both initiated by coordinator.
- Any site can decide to abort an Xact.
- Every msg reflects a decision by the sender; to ensure that this decision survives failures, it is first recorded in the local log.
- All commit protocol log recs for an Xact contain Xactid and Coordinatorid. The coordinator's abort/commit record also includes ids of all subordinates.



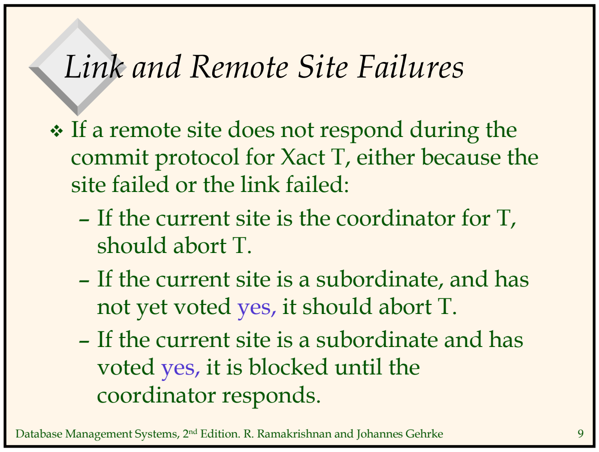
Restart After a Failure at a Site

- If we have a commit or abort log rec for Xact T, but not an end rec, must redo/undo T.
  - If this site is the coordinator for T, keep sending **commit/abort** msgs to subs until **acks** received.
- If we have a prepare log rec for Xact T, but not commit/abort, this site is a subordinate for T.
  - Repeatedly contact the coordinator to find status of T, then write **commit/abort** log rec; redo/undo T; and write **end** log rec.
- If we don't have even a prepare log rec for T, unilaterally abort and undo T.
  - This site may be coordinator! If so, subs may send msgs.



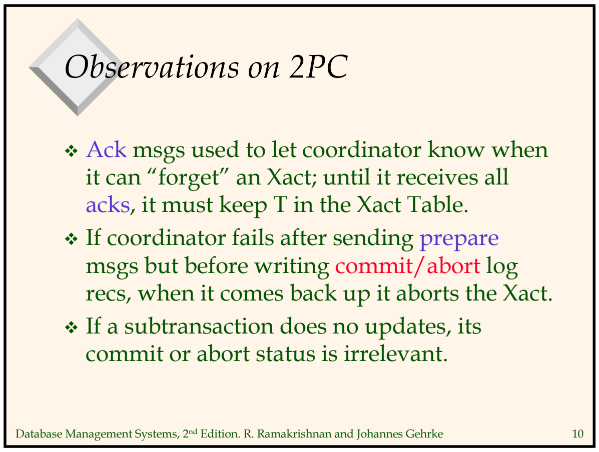
Blocking

- If coordinator for Xact T fails, subordinates who have voted yes cannot decide whether to commit or abort T until coordinator recovers.
  - T is blocked.
  - Even if all subordinates know each other (extra overhead in prepare msg) they are blocked unless one of them voted no.



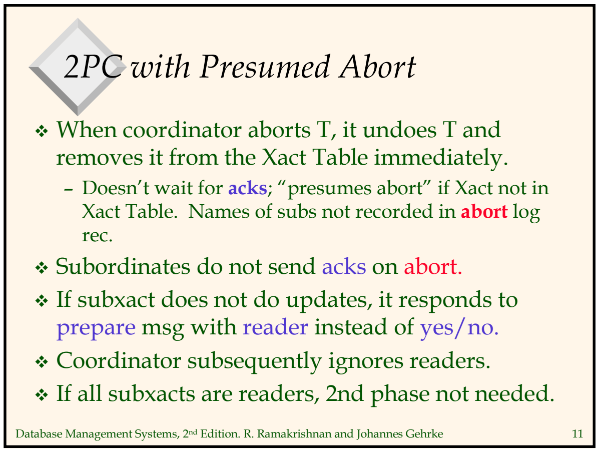
Link and Remote Site Failures

- If a remote site does not respond during the commit protocol for Xact T, either because the site failed or the link failed:
  - If the current site is the coordinator for T, should abort T.
  - If the current site is a subordinate, and has not yet voted yes, it should abort T.
  - If the current site is a subordinate and has voted yes, it is blocked until the coordinator responds.



Observations on 2PC

- Ack msgs used to let coordinator know when it can “forget” an Xact; until it receives all acks, it must keep T in the Xact Table.
- If coordinator fails after sending prepare msgs but before writing commit/abort log recs, when it comes back up it aborts the Xact.
- If a subtransaction does no updates, its commit or abort status is irrelevant.



2PC with Presumed Abort

- When coordinator aborts T, it undoes T and removes it from the Xact Table immediately.
  - Doesn’t wait for **acks**; “presumes abort” if Xact not in Xact Table. Names of subs not recorded in **abort** log rec.
- Subordinates do not send acks on abort.
- If subxact does not do updates, it responds to prepare msg with reader instead of yes/no.
- Coordinator subsequently ignores readers.
- If all subxacts are readers, 2nd phase not needed.