

Figure 14.1 A basic architectural model for the management of replicated data

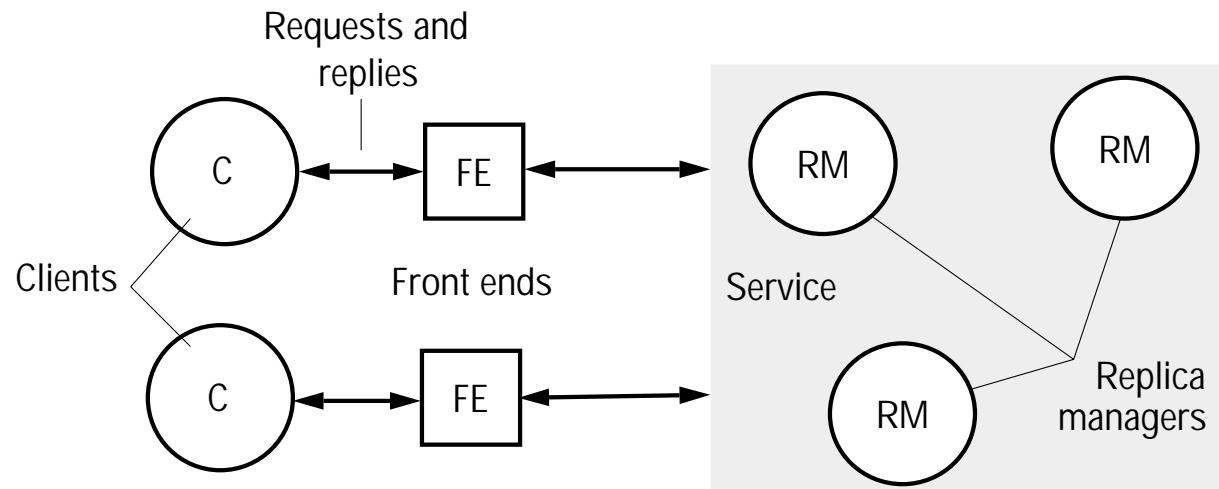


Figure 14.2 Services provided for process groups

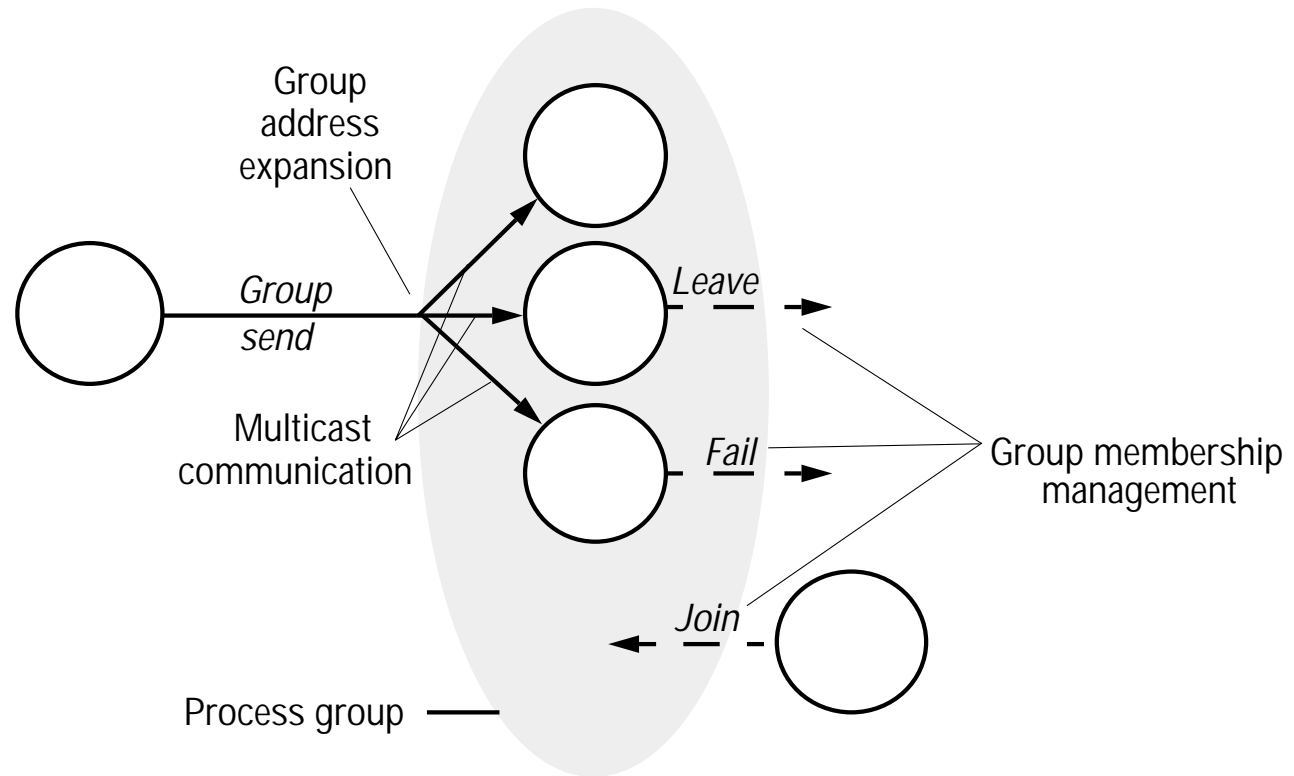
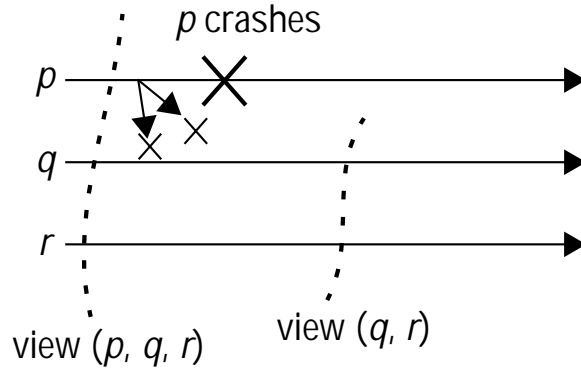
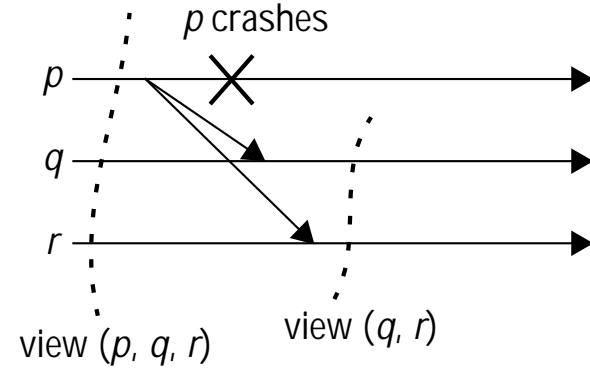


Figure 14.3 View-synchronous group communication

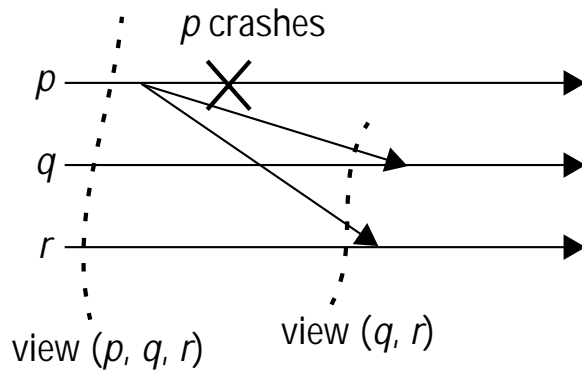
a (allowed).



b (allowed).



c (disallowed).



d (disallowed).

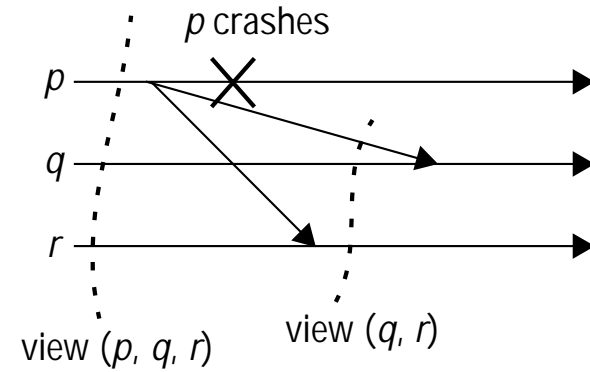


Figure 14.4 The passive (primary-backup) model for fault tolerance

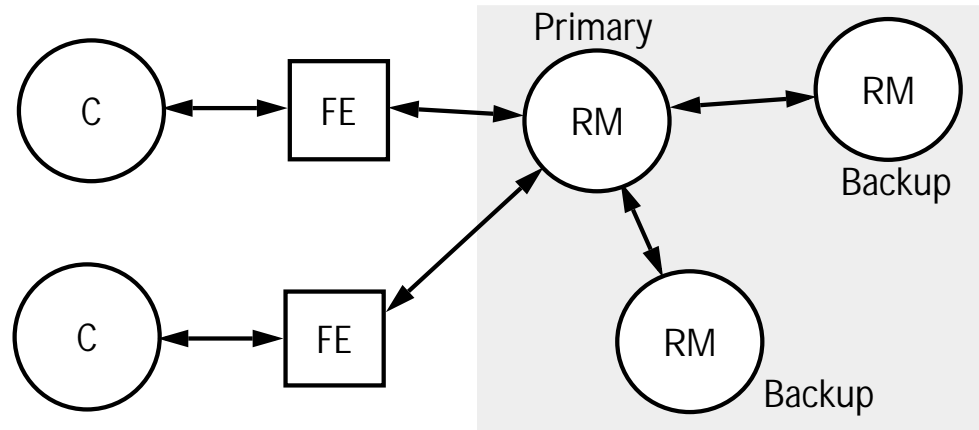


Figure 14.5 Active replication

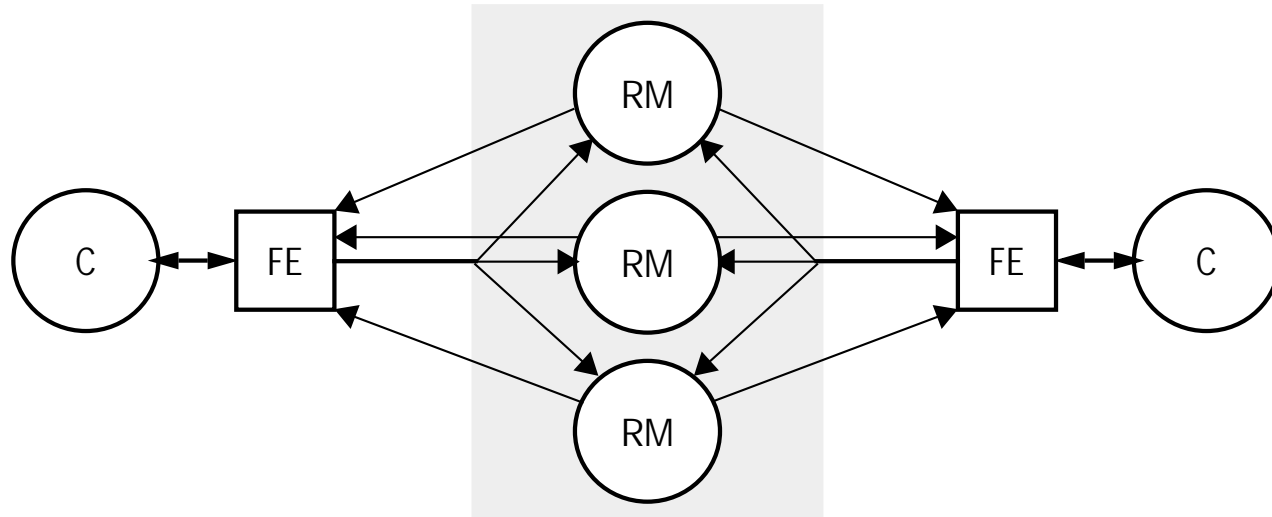


Figure 14.6 Query and update operations in a gossip service

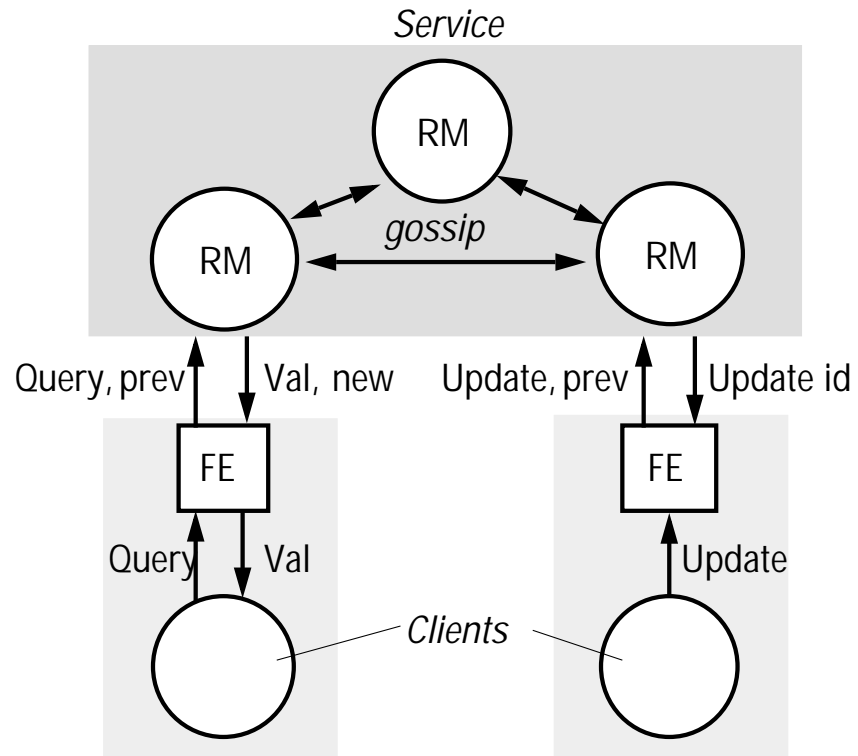


Figure 14.7 Front ends propagate their timestamps whenever clients communicate directly

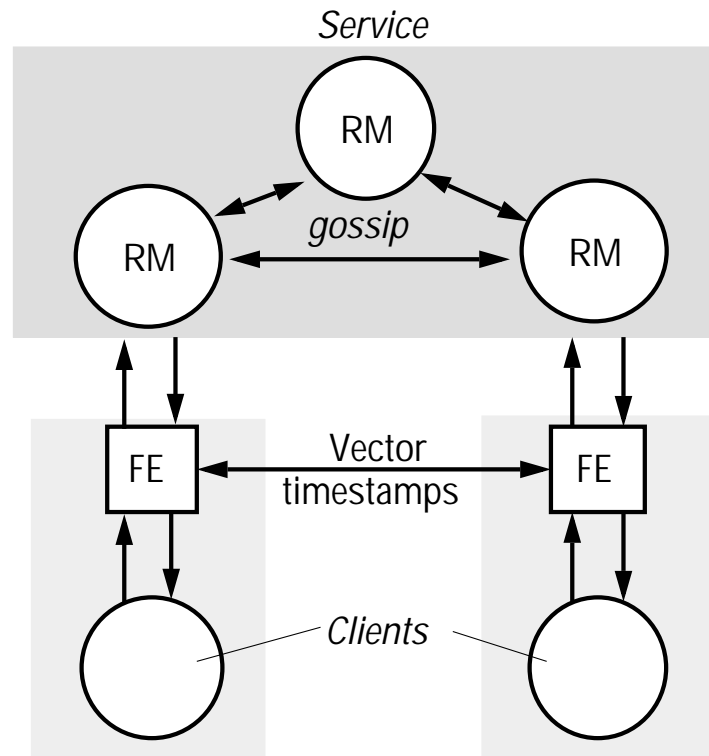


Figure 14.8 A gossip replica manager, showing its main state components

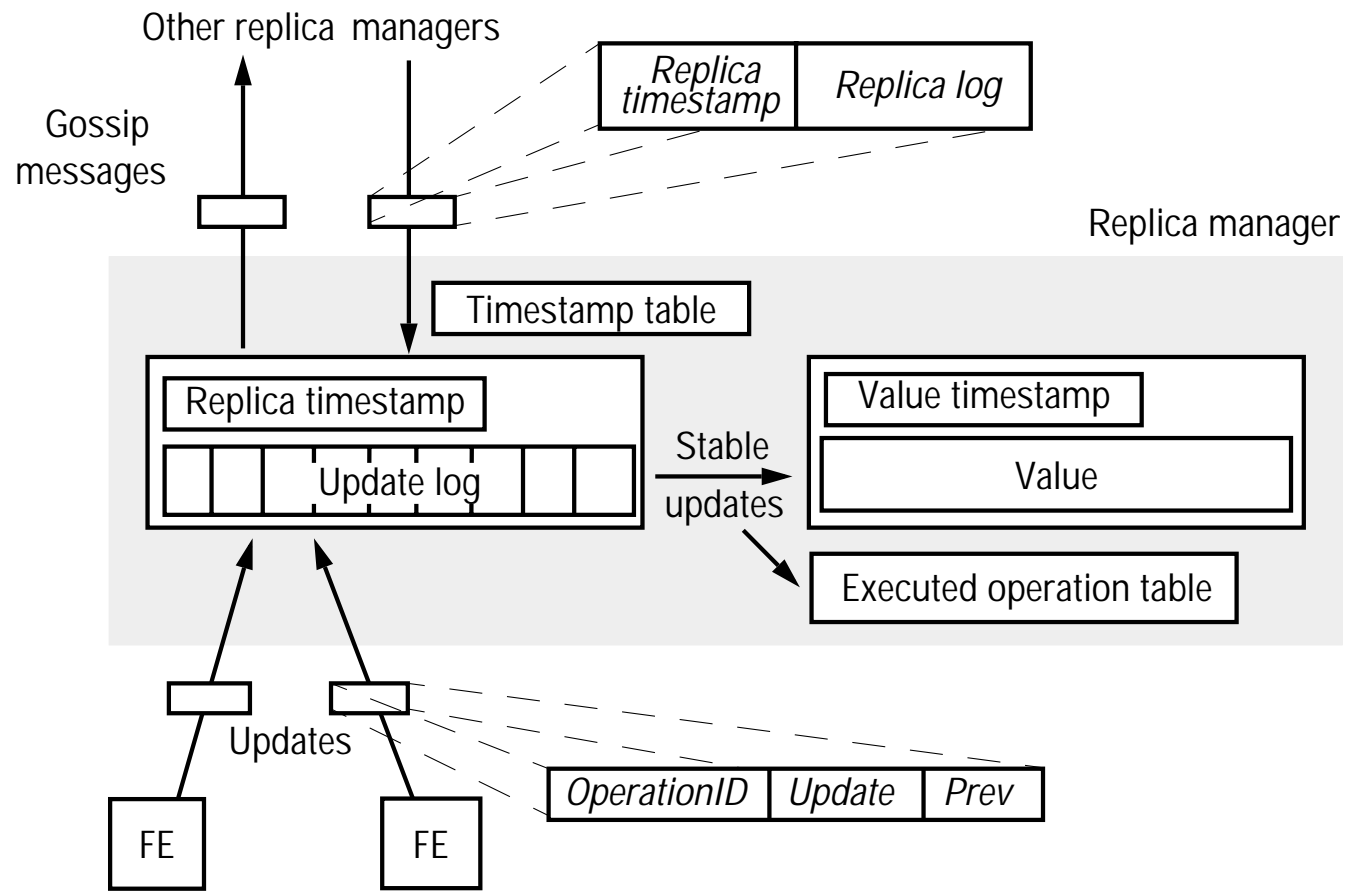
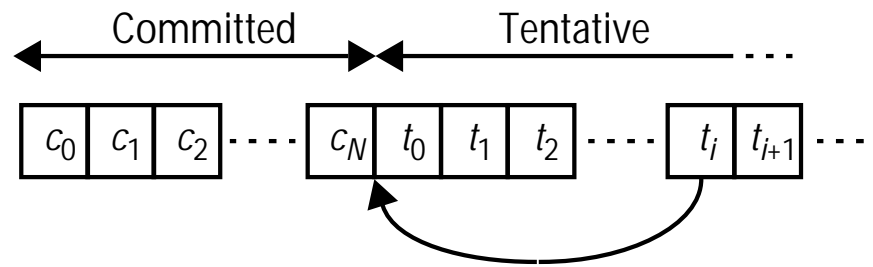




Figure 14.9 Committed and tentative updates in Bayou



Tentative update  $t_i$  becomes the next committed update and is inserted after the last committed update  $c_N$ .

Figure 14.10 Transactions on replicated data

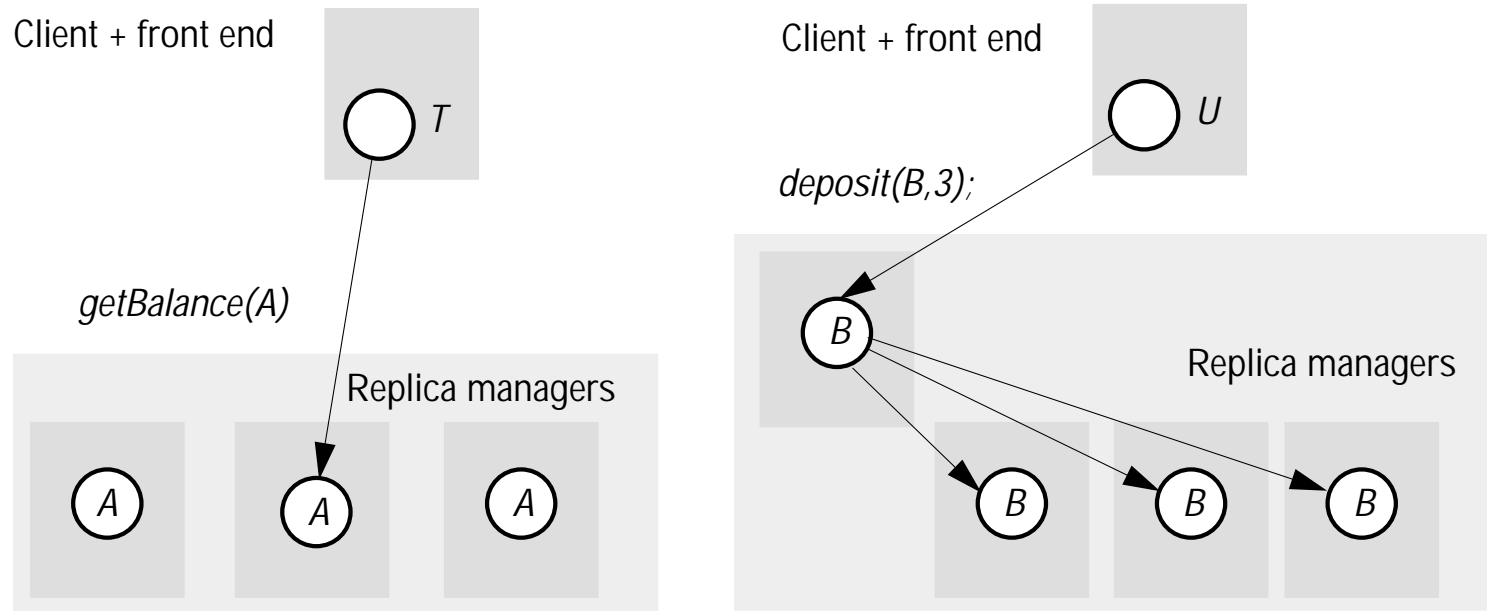


Figure 14.11 Available copies

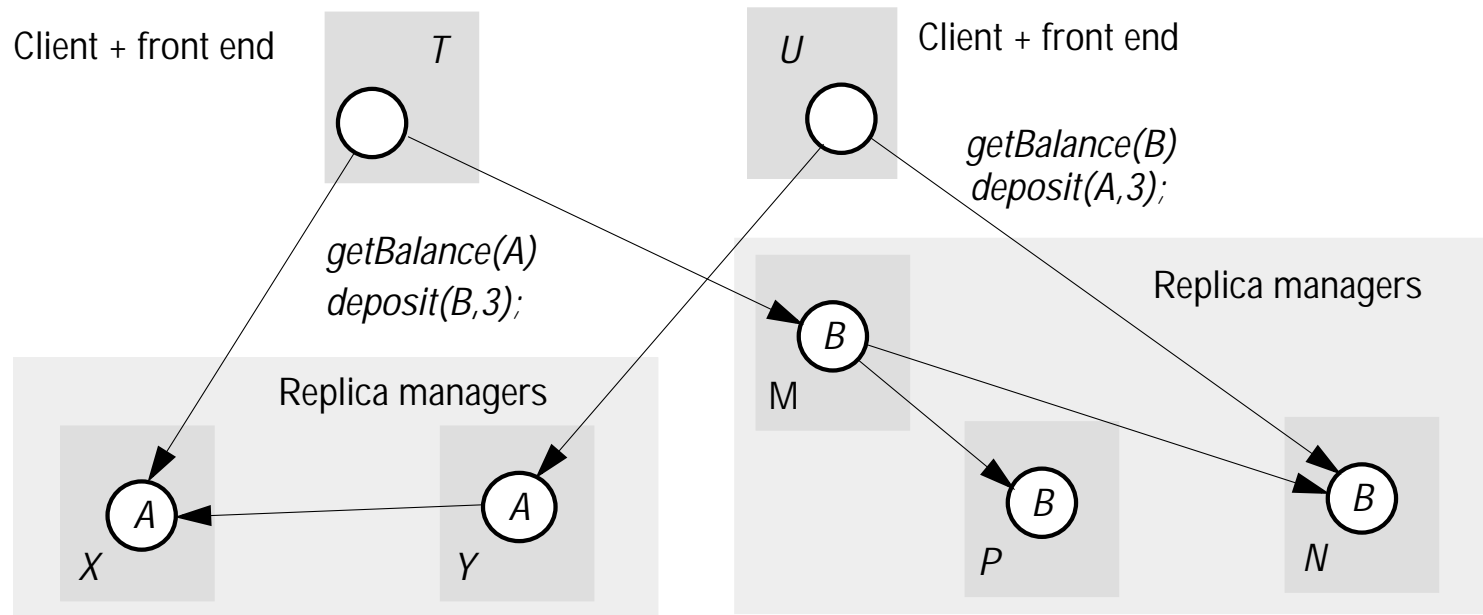
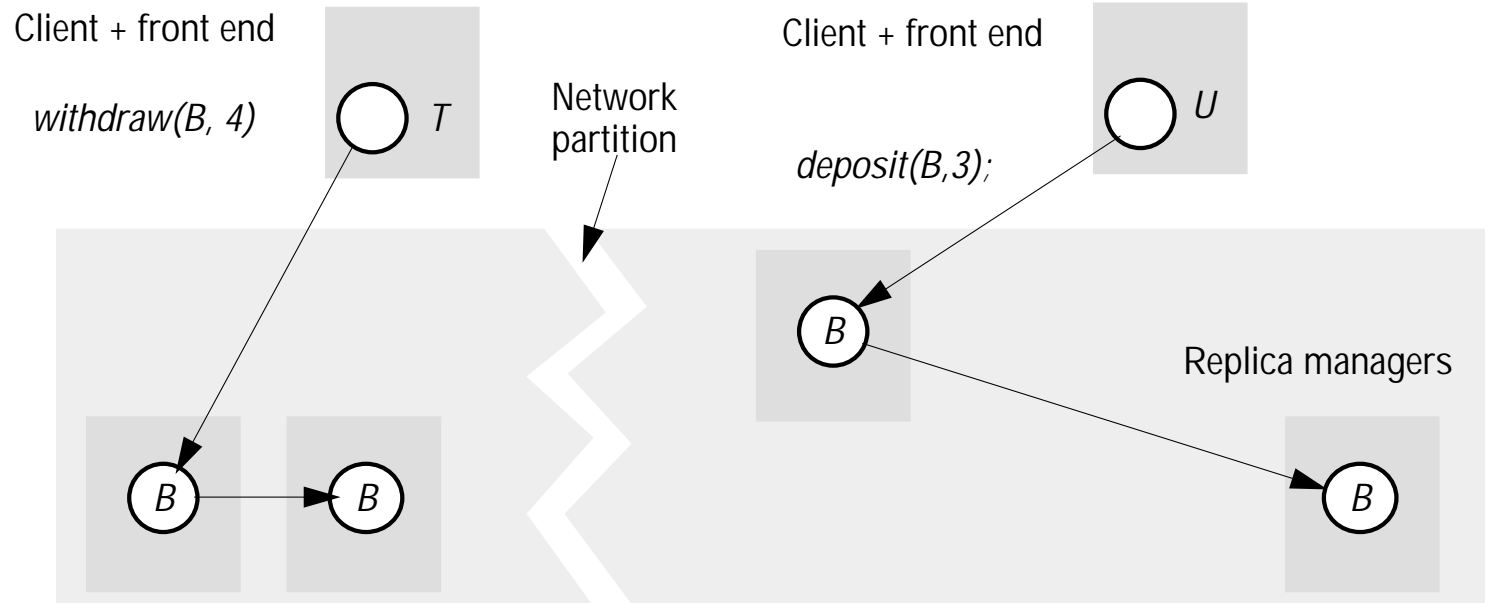


Figure 14.12 Network partition



## Gifford's quorum concensus examples

		<i>Example 1</i>	<i>Example 2</i>	<i>Example 3</i>
<i>Latency</i> <i>(milliseconds)</i>	Replica 1	75	75	75
	Replica 2	65	100	750
	Replica 3	65	750	750
<i>Voting</i> <i>configuration</i>	Replica 1	1	2	1
	Replica 2	0	1	1
	Replica 3	0	1	1
<i>Quorum</i> <i>sizes</i>	<i>R</i>	1	2	1
	<i>W</i>	1	3	3
<i>Derived performance of file suite:</i>				
<i>Read</i>	Latency	65	75	75
	Blocking probability	0.01	0.0002	0.000001
<i>Write</i>	Latency	75	100	750
	Blocking probability	0.01	0.0101	0.03

Figure 14.13 Two network partitions

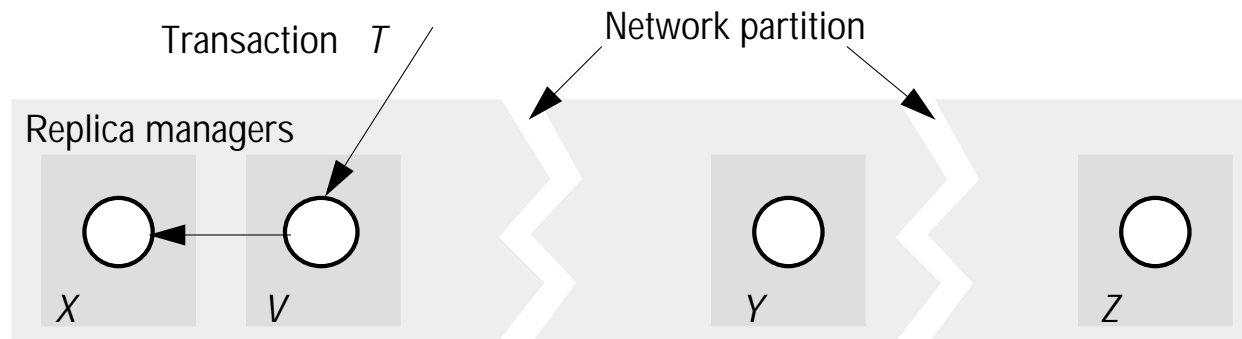


Figure 14.14 Virtual partition

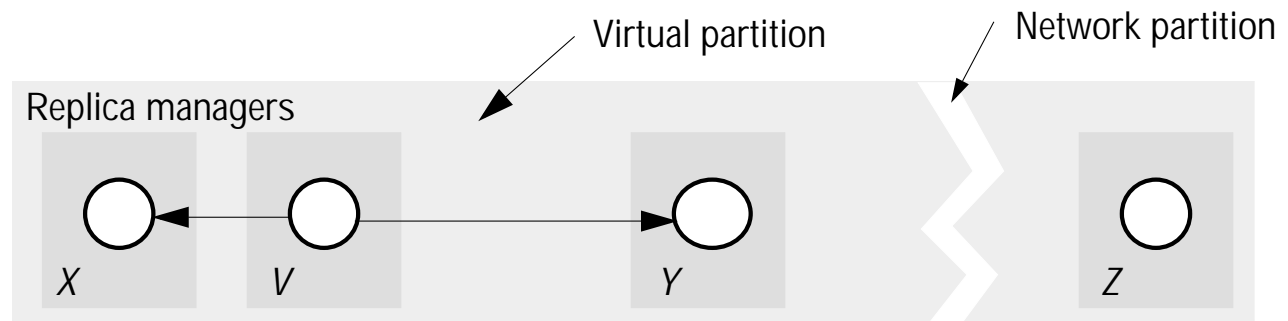
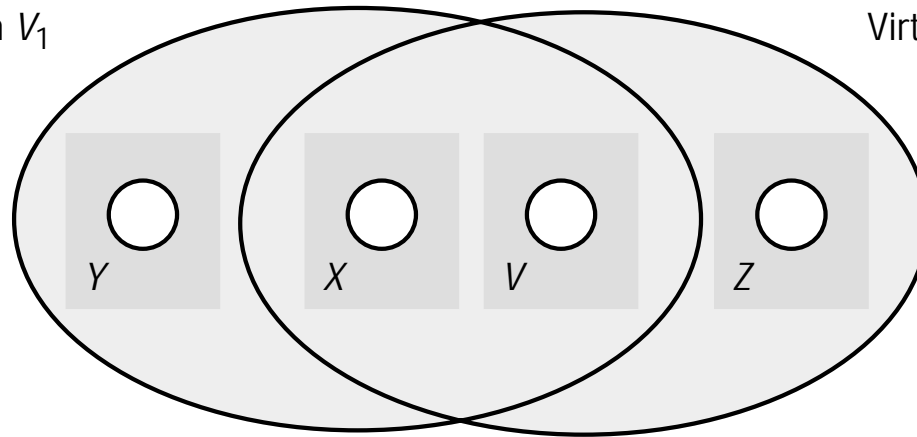


Figure 14.15 Two overlapping virtual partitions

Virtual partition  $V_1$

Virtual partition  $V_2$





## Figure 14.16 Creating a virtual partition

### Phase 1:

- The initiator sends a *Join* request to each potential member. The argument of *Join* is a proposed logical timestamp for the new virtual partition.
- When a replica manager receives a *Join* request, it compares the proposed logical timestamp with that of its current virtual partition.
  - If the proposed logical timestamp is greater it agrees to join and replies *Yes*;
  - If it is less, it refuses to join and replies *No*.

### Phase 2:

- If the initiator has received sufficient *Yes* replies to have *read* and *write* quora, it may complete the creation of the new virtual partition by sending a *Confirmation* message to the sites that agreed to join. The creation timestamp and list of actual members are sent as arguments.
- Replica managers receiving the *Confirmation* message join the new virtual partition and record its creation timestamp and list of actual members.