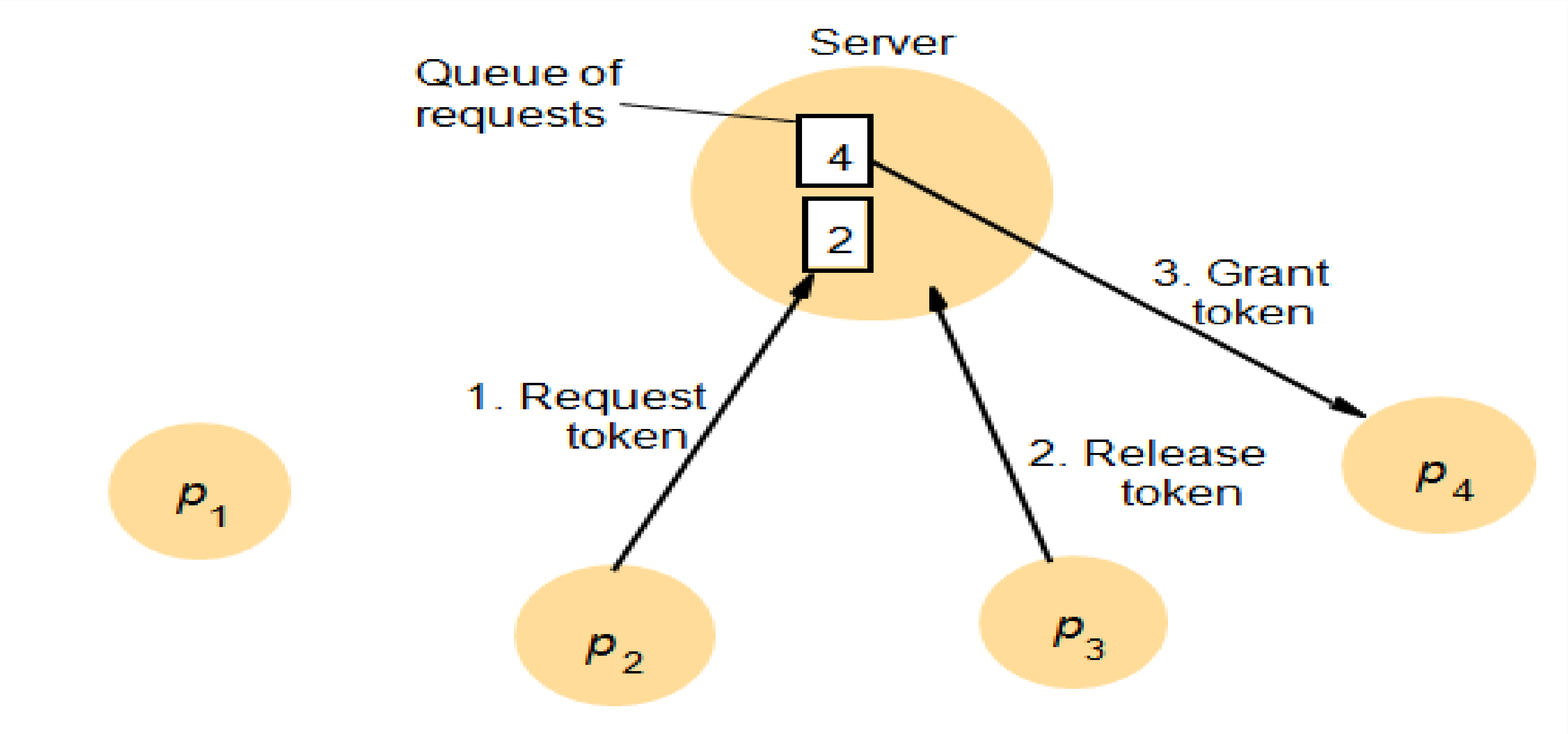
Centralized & Ring-based Mutex Algorithms

Reference: 1. Mukesh Singhal & N.G. Shivaratri, Advanced Concepts in Operating Systems,

2. George Coulouris, Jean Dollimore and Tim Kindberg, "Distributed Systems Concepts and Design", Fifth Edition, Pearson Education, 2012

Centralized Mutex Algorithms



Centralized Mutex Algorithms

Central server receives access requests

- – if no process in critical section, request will be granted
- – if process in critical section, request will be queued

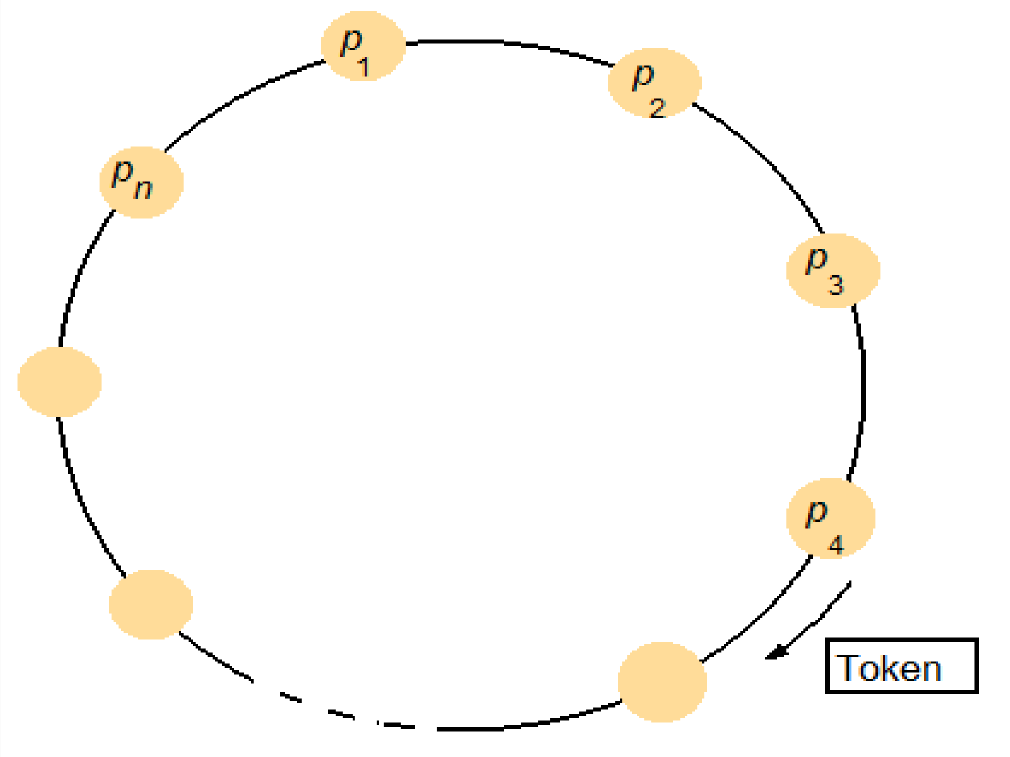
Process leaving critical section

- – grant access to next process in queue, or wait for new requests if queue is empty

Centralized Mutex Algorithms

- Satisfies Safety and Liveness property
- But Fairness is not satisfied as there may be delay in network which will not order request as they are sent. Absence of timestamp.

Ring-based Mutex Algorithms



Ring-based Mutex Algorithms

- Every process pi has connection to process p(i+1) mod N
- Links between processes may be Logical, not necessarily physical link
- Token passes in one direction through the ring

Token arrival does the following

- – only process in possession of token may access critical region
- – if no request upon arrival of token, or when exiting critical region, pass token on to neighbor.

Ring-based Mutex Algorithms

- This ensure Safety property.
- Liveness: Process has to wait for critical section until it gets a chance to receive a token
- Fairness is satisfied as every process will get token as token moves along ring.
- This is not suitable in Distributed environment as request will be received from multiple sites at same time and other sites need to be aware of who is currently holding a token.