Distributed Systems

2015 Exam 1 Review

Paul Krzyzanowski

Rutgers University

Fall 2016

© 2015 Paul Krzyzanowski

Question 1

Why did the use of reference counting for remote objects prove to be impractical? Explain.

It's not fault tolerant.

If a client process dies or exits without properly decrementing reference counts, the object would not get deleted.

Bad answers:

- *Requires more network usage (or extra unnecessary requests issued by client)*
  - *That may be true only in some cases (e.g., a lot of object referencing activity on the client) but it does not make the solution impractical*
- *Problems with lost messages*
  - *That could be a problem but is a problem with any protocol, including leasing. You need to use reliable messaging (e.g., acknowledgements & retransmissions).*

© 2015 Paul Krzyzanowski

Question 2

(a) What is the advantage of vector clocks over Lamport clocks?

Vector clocks allow you to tell whether a set of events are causally related or concurrent by comparing their timestamps. They let you identify the most recent version of a message or whether there are conflicts.

(b) What is a disadvantage?

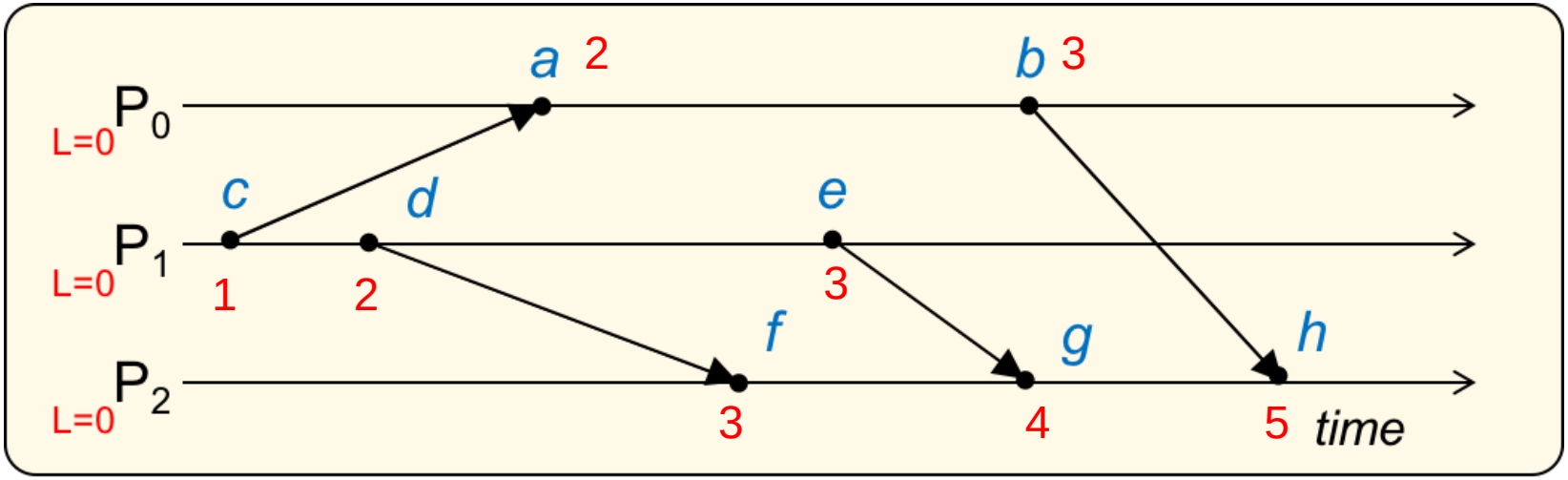
1. Vector timestamps use more space because you have a vector (one element for each process) rather than one integer.
2. Comparing them takes more time since you need to do an element-by-element comparison.

Bad answer: "more expensive", "slower"
Answers such as these are too vague to show that you understand the material.

© 2015 Paul Krzyzanowski

Question 3

Attach Lamport time stamps to each of the following events.
Assume that the initial clock value at each processor (before any event takes place) is 0.



a would be *1* but it's the receipt of (*c*=*1*), so *a* gets *c*+*1* = *2*.

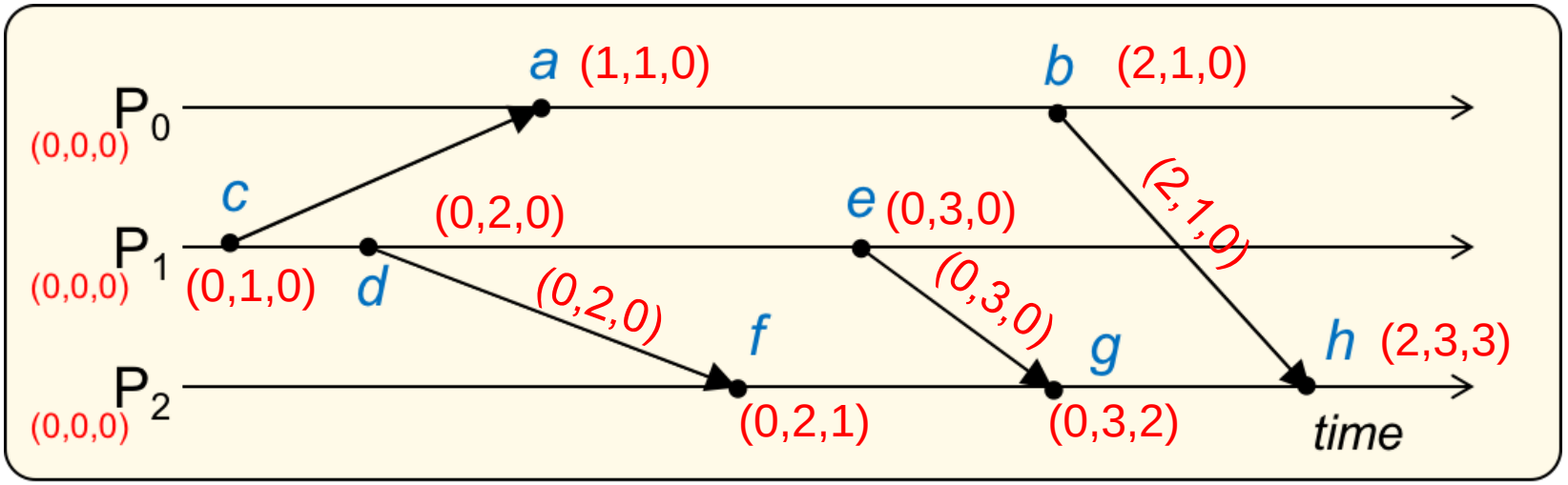
f would be 1 but it's the receipt of (*d*=2), so *f* gets *d*+*1* = *3*.

a	2	e	3
b	3	f	3
c	1	g	4
d	2	h	5

© 2015 Paul Krzyzanowski

Question 4a

Attach vector time stamps to each of the following events. Use a vector ordering of <P_0, P_1, P_2> and assume that the initial clock value at each processor (before any event takes place) is 0.



a would be *(1,0,0)* but it's the receipt of *c=(0,1,0)*, so *a* gets the max values: *(1,1,0)*.

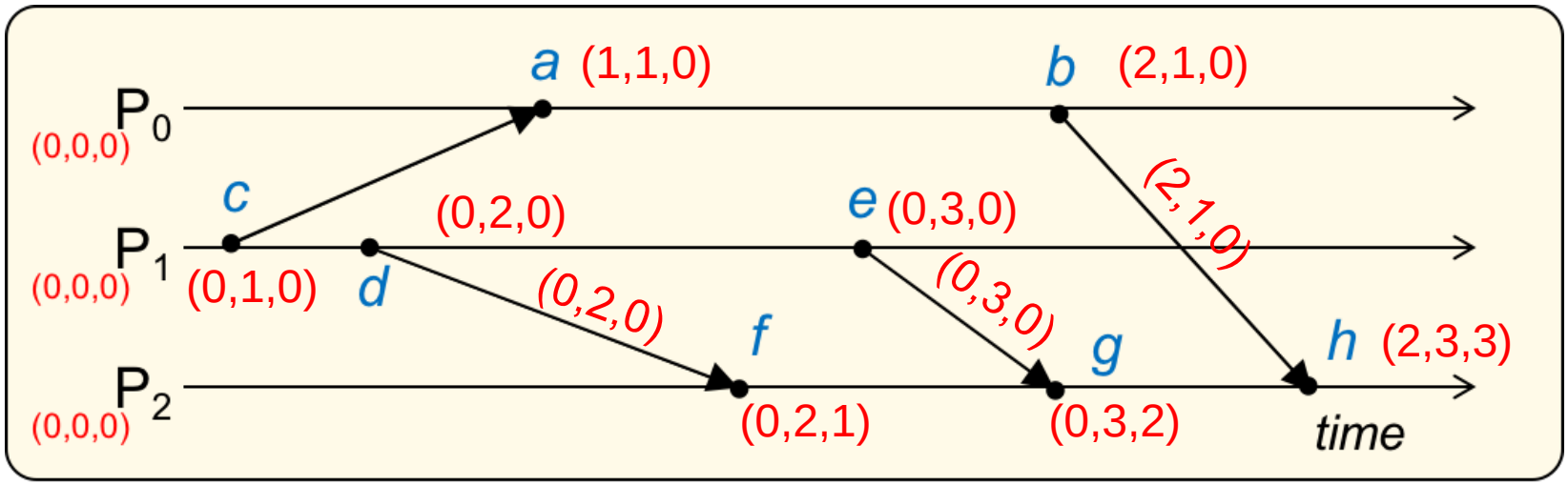
f would be (0,0,1) but it's the receipt of *d*=(0,2,0), so *f* gets the max values: *(0,2,1)*.

a	(1,1,0)	e	(0,3,0)
b	(2,1,0)	f	(0,2,1)
c	(0,1,0)	g	(0,3,2)
d	(0,2,0)	h	(2,3,3)

© 2015 Paul Krzyzanowski

Question 4b

Identify which event(s) is/are concurrent with event *e*.



V and V' are concurrent if $V \not\leq V'$ and $V \not\geq V'$

e=(0,3,0) > c=(0,1,0)

e=(0,3,0) > d=(0,2,0)

e=(0,3,0) < g=(0,3,2)

e=(0,3,0) < h=(2,3,3)

e is concurrent with

e=(0,3,0) : a=(1,1,0)

e=(0,3,0) : b=(2,1,0)

e=(0,3,0) : f=(0,2,1)

© 2015 Paul Krzyzanowski

Question 5

Multiprocessor systems differ from a network of computers in that they:

a) Are designed for high performance rather than high throughput.
b) Must have a data communications network.
c) Run specialized operating systems.
d) Must have shared memory.

- By definition

© 2015 Paul Krzyzanowski

Question 6

A Non-Uniform Memory Access (NUMA) system is called that because

a) A region of memory is local to each processor and can be accessed more quickly by that processor.
b) Some memory may be located on remote computers and requires sending a request over a data network.
c) Frequently-used memory is stored in a cache within the processor and can be accessed more quickly.
d) Memory access time may vary depending on how much contention there is from other processors.

(b) No remote computers or data networks in NUMA – just multiprocessors with shared memory

(c) No. That's just a cache

(d) Maybe, but that applies to bus-based access too. It's not a definition of NUMA

© 2015 Paul Krzyzanowski

Question 7

With home snoop caching:

a) Each processor's cache always listens for memory operations on the bus and invalidates its cache if it sees writes for addresses it has cached.
b) A CPU contacts the processor that is responsible for a specific memory address. That processor, in turn, forwards the request to the processor that has the latest cached value for that memory.
c) Only one processor, called the "home processor" snoops on memory operations from all other processors.
d) A CPU broadcasts a snoop request to all other processors, asking for the latest version of a memory address.

(a) No. That's bus-based snooping.

(c) No.

(d) No. That's source snooping.

© 2015 Paul Krzyzanowski

Question 8

The end-to-end design principle states that:

a) Network protocols should be implemented in layers to simplify programming and enable changes in protocols.
b) Whenever possible, application-specific functions should be implemented at the end hosts of the network.
c) The network should be smart enough to figure out how to deliver data from one endpoint to another.
d) Every data packet should contain both the source address and destination address.

© 2015 Paul Krzyzanowski

Question 9

ONC RPC (the default RPC system on Linux, OS X, and BSD) does not support the following:

a) Selection of transport protocol at run-time.
b) Marshaling parameters to create a network message.
c) Remote object references.
d) A name server for service location.

- ONC RPC has no support for remote objects; it's strictly functional.

© 2015 Paul Krzyzanowski

Question 10

Multi-canonical network data representation:

a) Uses a textual representation for data to ensure maximum interoperability.
b) Allows two processes to choose from one of several formats for representing data.
c) Encodes the same data in both binary and text formats to simplify debugging and inspection.
d) Encodes descriptive information about the data, such as names and types, in addition to the data.

- Goal: try to avoid systems having to convert data into an intermediate form by supporting several standard formats. Ideally, at least one system will be able to handle one of the formats natively.
- The standard supports multiple data representation formats.
- Endpoints negotiate which one they want to use.

© 2015 Paul Krzyzanowski

Question 11

Microsoft introduced the concept of a surrogate process to:

a) Enable the client to locate the server.
b) Create a proxy object that the client loads.
c) Act as a secure gateway that validates all remote requests.
d) Be the process that loads COM objects at the server based on client requests.

- A surrogate process lives on the server and dynamically loads objects as needed by clients.

© 2015 Paul Krzyzanowski

Question 12

An RPC name server is used to:

a) Obtain a unique name for a set of functions.
b) Find the port number for a set of functions.
c) Convert a function name to a remote address.
d) Store distributed objects.

(a) No. You query the name server with the name (#).

(c) No. What's a remote address?

(d) No. The RPC name server (e.g., portmapper on Linux) is:

- Used by the server to register (program #, port #)

- Used by the client to look up the port # given the program #

© 2015 Paul Krzyzanowski

Question 13

A key distinction between SOAP and REST interfaces is that:

a) REST sends and receives documents while SOAP uses remote procedures.
b) REST must use JSON to encode its data.
c) REST is architecture independent.
d) Operations are encoded in the document with SOAP but in the URL in REST.

- The design principle of REST is to take advantage of the HTTP protocol and URLs
  - You should be able to get an idea of the API simply by looking at the URLs
- SOAP encodes operations within the XML document

(a) Both REST & SOAP are document based; both can invoke remote procedures

(b) REST doesn't specify what the document encoding is. You can use XML.

(c) So is SOAP.

© 2015 Paul Krzyzanowski

Question 14

Google Protocol Buffers are:

a) A library for converting between different network protocols.
b) An efficient binary method for serializing data.
c) A library for buffering requests and responses to account for slow networks.
d) A fast way to convert arbitrary data into portable text-based XML documents.

© 2015 Paul Krzyzanowski

Question 15

Java manages the lifetime of remote objects via:

a) Tracking client connectivity.
b) Leases.
c) Remote reference counting.
d) A distributed dependency graph.

- Java uses LOCAL reference counting to keep track of the lifetime of objects.
- For remote objects, it sends a *dirty* message on the first reference to the object and a *clean* message when the object is no longer needed.
- A client has to send *dirty* messages periodically to keep the lease active.

© 2015 Paul Krzyzanowski

Question 16

Microsoft's Leasing Distributed Garbage Collector:

a) Renews an object's expiration timer whenever an object is accessed.
b) Deletes replicas of an object when they are no longer referenced.
c) Follows a chain of dependencies to clean up unused objects across multiple servers.
d) Deletes an object only if its reference count is 0 and the object is not renewed.

The LDGC uses leasing and avoids unneeded pings by renewing the lease whenever the object is referenced.

(b) Replicas of an object? There's no replication!

(c) Chain of dependencies? None of that either; I just made it up.

(d) No reference counting!

© 2015 Paul Krzyzanowski

Question 17

Cristian's algorithm improves simply setting the clock to the value returned by the server by:

a) Adjusting for drift.
b) Adjusting for skew.
c) Factoring in latency.
d) All of the above.

- "Simply setting the clock" adjusts for skew.
- You need to keep track of the skew and frequency of setting to establish drift. Cristian's algorithm does not do that.
- Cristian's algorithm factors in the latency between sending a request and getting a response from the server.

© 2015 Paul Krzyzanowski

Question 18

A Berkeley clock master has a time of 1:20:00. Two slave systems, A and B, synchronize with the master. Currently, A has a time of 1:20:04 and B has a time of 1:20:05. After synchronizing, to what value does A set its clock?

a) 1:20:00
b) 1:20:02
c) 1:20:03
d) 1:20:04

The Berkeley algorithm simply computes the average

(1:20:00 + 1:20:04 + 1:20:05) ÷ 3 = 1:20:[09÷3] = 1:20:03

© 2015 Paul Krzyzanowski

Question 19

A stratum 2 NTP time server:

a) Syncs from two servers for increased accuracy.
b) Requires two network hops to connect to a client.
c) Syncs its clock from a stratum 1 server.
d) Is a backup for a stratum 1 time server.

Your position in the synchronization subnet (your stratum) is defined by the stratum of the server you synchronize from.

© 2015 Paul Krzyzanowski

Question 20

PTP, the Precision Time Protocol:

a) Assumes symmetric uplink & downlink delays.
b) Requires the client to initiate the sync with the server.
c) Measures, and accounts for, the differences between uplink and downlink latency.
d) Is an extension to NTP with 128-bit precision for time values.

(b) The server initiates the sync

(c) No.

(d) No. It's a completely different protocol from NTP
(and uses an 80-bit timestamp)

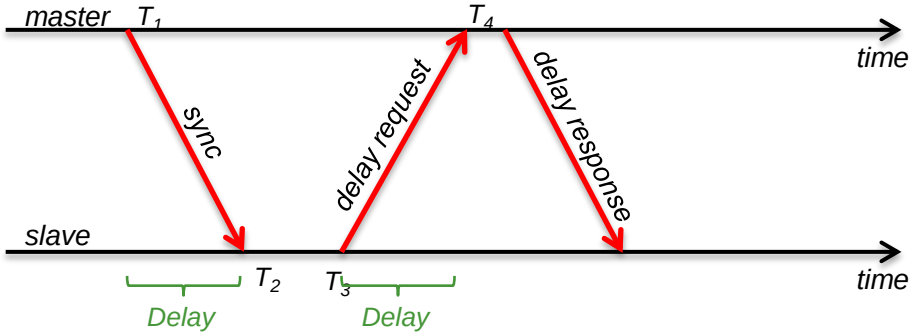
(a) PTP is designed for Ethernet LANs and assumes a symmetric delay

$T_2 - T_1 = delay + offset$

$T_4 - T_3 = delay - offset$

$T_2 - T_1 + T_4 - T_3 = 2\ (offset)$

$offset = (T_2 - T_1 + T_4 - T_3) / 2$



© 2015 Paul Krzyzanowski

Question 21

What can you tell by comparing two Lamport timestamps L_a and L_b?
If $L_a < L_b$ then:

a) Event a must have happened before event b.
b) Event a must have happened after event b.
c) Events a and b are concurrent.
d) None of the above.

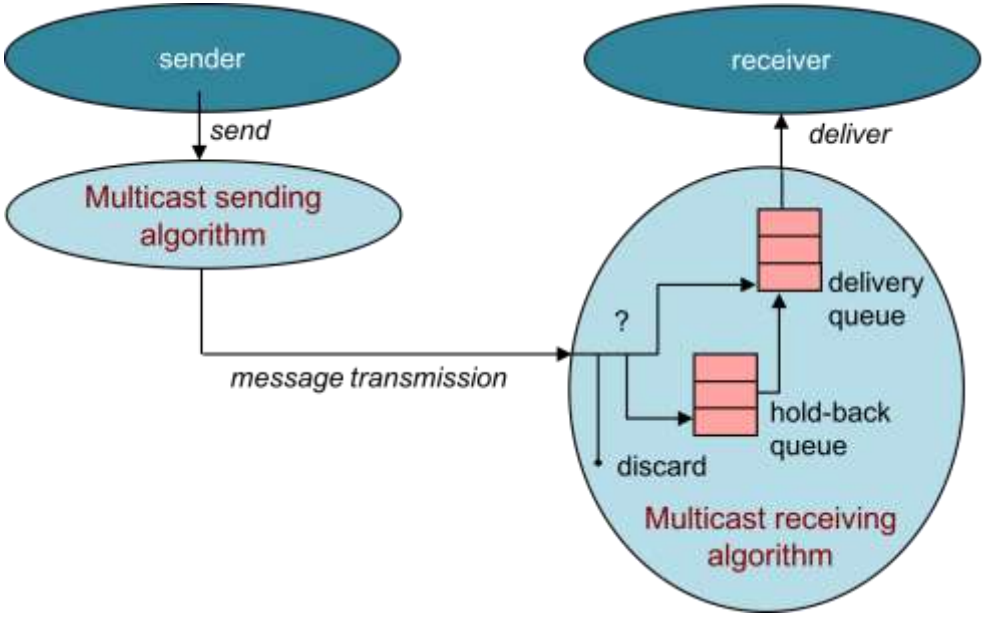
- You cannot tell anything by looking at the timestamps.
- If $a \rightarrow b$ then $L_a < L_b$
- But if $L_a < L_b$ then it is not necessarily the case that $a \rightarrow b$

© 2015 Paul Krzyzanowski

Question 22

A hold-back queue:

a) Buffers received messages to deliver them at a constant rate to streaming media applications.
b) Holds and re-sequences messages so they can be delivered in the correct order to applications.
c) Holds messages at the receiver until an application is ready to consume them.
d) Holds messages at the sender in case they need to be retransmitted due to network data loss.



© 2015 Paul Krzyzanowski

Question 23

A precedence vector enables:

a) Global time ordering of messages.
b) Total ordering of messages.
c) Causal ordering of messages.
d) Sync ordering of messages.

It allows a receiver to see whether it has missed any messages from other processes that have been received & processed by the sender.

© 2015 Paul Krzyzanowski

Question 25

Which mutual exclusion algorithm ensures that a process can be granted access to a resource in just two network messages assuming reliable message delivery and that no other processes are using the resource?

a) Centralized.
b) Token ring.
c) Lamport's.
d) Ricart & Agrawala.

(a) One *request* message, one *grant* message.

(b) Requires sending messages from process to process around a ring.

(c) Requires sending messages to the entire group.

(d) Requires sending messages to the entire group.

© 2015 Paul Krzyzanowski

Question 26

Which election algorithm does not always require contacting all group members?

a) Bully algorithm.
b) Ring algorithm.
c) Chang and Roberts ring algorithm.
d) All of the above require contacting all group members.

(b) Ring circulates an *election* message, trying to contact all live members and then choosing the winner.

(c) Chang & Roberts optimizes the ring but the same messaging takes place.

(a) Bully requires contacting *only* the group members with PID > yours

I will also accept (d) on the argument that once a winner has been decide, all group members need to be informed.

© 2015 Paul Krzyzanowski

Question 27

True or False?

IP routers ensure that traffic is delivered reliably through the Internet.

FALSE

- IP does not promise reliable delivery.

© 2015 Paul Krzyzanowski

Question 28

True or False?

TCP implements reliable communication on top of UDP, which is unreliable.

FALSE

- TCP is a transport layer protocol on top of IP, a network layer protocol.
- UDP is a different transport layer protocol.

© 2015 Paul Krzyzanowski

Question 29

True or False?

Sockets are not needed for UDP communication since UDP is stateless.

FALSE

- UDP is stateless but still needs a socket as a queueing endpoint for a process to send and receive network data.

© 2015 Paul Krzyzanowski

Question 30

True or False?

A problem with passing parameters by reference to a remote function is the lack of shared memory between the processes.

TRUE

- Pass by reference implies passing the address of the parameter. This makes no sense in a different process's address space.

© 2015 Paul Krzyzanowski

Question 31

True or False?

Physical clocks must be synchronized on systems for logical clocks to work properly.

FALSE

- Logical clocks have no relation to physical clocks.
- They are just sequence numbers.

© 2015 Paul Krzyzanowski

Question 32

True or False?

Ricart & Agrawala's mutual exclusion algorithm is more efficient than Lamport's because it never requires comparing time stamps.

FALSE

- It's a bit more efficient but not because it doesn't require comparing time stamps.
- The way you decide who gets the resource if multiple processes are contending for it is by comparing time stamps.

© 2015 Paul Krzyzanowski

Question 33

True or False?

If a network is partitioned (segmented), an election algorithm may elect multiple leaders.

TRUE

- Yes. Some processes will assume other processes died because they cannot communicate with them

© 2015 Paul Krzyzanowski

The End

© 2015 Paul Krzyzanowski