Exam in Distributed Systems

Justin Pearson

2004-03-17 - P.B. 14-19

Cover Sheet

This sheet should be handed in together with the exam.

Each problem must be solved on a separate sheet. Write your name on each sheet. Indicate below which questions you have answered.

Problem no.	Solution provided	Max	Your points
1		12	
2		6	
3		4	
4		8	
5		4	
6		4	
7		6	
8		7	
9		15	
	Total:	42/64	

Name : .....

Pers.no. : .....

Are you TF(1TT835) or MN1(1DT641) : .....

Obs

I will not be able to come to the exam. If you are in any doubt when answering a question please state any assumptions that you have to make.

Exam Rubic

A mark of 50% is required for a pass, a 4 and 5 are distributed evenly. Answer can be in English or Swedish.

Students who are registered on the TF (3 point version of the course) must only do question 1–7. Any other questions answered will be ignored. Students registered on the full (5 point) version of the course must do all questions.

Each full answer should be started on a separate sheet. Please write you name and personal number on each sheet.

You must indicate on the front sheet which version of the course you are registered on.

Have you read the exam rubric?? If not go back and do it now!!!!!

1. General Questions on Distributed Systems

   (a) With **examples** describe *Access*, *Location* and *Migration* transparency in a distributed system. (**4 points)**

   (b) With **examples** describe what Middleware is. (**2 points)**

   (c) With **examples**[1] describe the client server model? (**2 points)**

   (d) What is redundancy and why is it used in a distributed system. (**2 points)**

2. Communication.

   (a) What are Client and Server Stubs and how are they used in remote procedure calls? (**2 points)**

   (b) When passing data values between different machines with different operating systems what problems have to be solved? (**2 points)**

   (c) When calling a remote procedure or method where a reference parameter is passed, what problems have to be solved and how are they solved? (**2 points)**

3. Naming

   (a) With **examples** explain the difference between iterative name resolution and recursive name resolution. (**2 points)**.

   (b) Explain with examples the difference between a name server and a directory server. (**2 points)**.

4. Logical Clocks

   (a) Suppose there are three processes A,B and C. All clock runs at the same rate but initially A's clock reads 10, B's clock reads 0 and C's clock reads 5. At time 10 by A's clock, A sends a message to B, this message takes 4 units of time to reach B. B then waits one unit of time and then sends a message onto C which takes 2 units of time to reach C. Assuming that the system implements Lamport's timestamps draw a picture illustrating the timestamps

[1] By now you should have got the message, I want examples where possible.

for the messages and explain how the timestamps are obtained. (**6 points)**

(b) What does it mean for two events to be concurrent and what is the relation of the Lamport timestamps of the two events. (**2 points)**.

5. Clock Synchronisation

   (a) What is a leap second and why is it used? (**2 points)**

   (b) Explain Cristian's algorithm for Clock synchronisation. (**2 points)**

6. Explain in detail the Bully algorithm for electing a leader. (**4 points)**.

7. Data-Centric Consistency.

   (a) What is strict consistency and why is not possible to achieve in a distributed system? (**3 points)**

   (b) Define sequential-consistency. You must give examples. (**2 points)**

   (c) Is the following data store sequentially consistent? Explain your answer. (**1 points)**

A	W(x)a		W(x)b		
B		R(x)a		R(x)b	
C				R(x)b	R(x)a

   (d) Explain with examples what release consistency is. (**2 points)**

Stop here if you are a TF student

8. Replication.

   (a) What is atomic multi-cast? (**1 points)**

   (b) Why is atomic multi-cast needed when replication is used? (**2 points)**

   (c) With examples illustrate the difference between passive and active replication. (**4 points)**

9. Transactions.

   (a) With examples define what a transaction is. (**4 points)**

   (b) What is a nested transaction. (**2 points)**

(c) Explain what problems can happen there is no concurrency control where multiple transactions are being executed at the same time. (**2 points)**

(d) Define what it means for an interleaving of two transactions to be serially equivalent? (**2 points)**

(e) What is the two-phase commit algorithm? Explain it in some detail. (**3 points)**

(f) Why is the two-phase commit algorithm used in distributed transactions? (**2 points)**