

# Distributed System I

## Wintersemester2020/21

### Assignment 1

*Ciheng Zhang (3472321) zch3183505@gmail.com*  
*Chenxi Li(3502796) cli216@outlook.com*  
*Leqi Xu(3556962) st176119@stud.uni-stuttgart.de*  
*Yaosheng Zheng (3563285) zhengyaosheng312@icloud.com*  
*Team 19*

November 22, 2020

## Transparency Levels

a)

Concurrency transparency: in this question we use http protocol. HTTP allow multiple people to access at the same time.

Replication transparency: here users never know how many copies exist in the server.

Scaling transparency: The server can extend the storage of the server but dont need to change the structure.

b)

Concurrency transparency, Replication transparency, Scaling transparency. The reason is same as the question a because of http protocol.

Location transparency: In the question we use domain not IP address. we need to use DNS to get the real address of the server. So we actually dont know where its the server.

Migration transparency: we use DNS to locate. So if we change our real location of our server, there will be no affect to application.

c)

We use write all and read one methode.

Write operation: the protocol lock all copys. Then the client write the change to all copys. Then all lock release.

Read operation: At first lock all copys, and then the client read only one copy. after reading the locks release.

In this protocol the changes will be write in all the copys. And when the client is reading one copy. the server is locked no one can change the copys. so it ensure delivers the most recent copy.

## System Models

a)

In this question, The necessary assumptions are: there is no Failure in the system(No communication Failure and no Process Failure). This mean if a request is sent, after a period of time  $P_2$  always will received it.

b)

The transmission delay and excute time is always less than a  $t_{max}$  also the drift of reciver and sender should be same.

c)

Both P1 and P2 w.r.t a real time, and after the request can be reached anyway. Justify: a request is send and the transmission need some time. So  $t_{P2}^{receive}(m_{req})$  and  $t_{P1}^{send}(m_{req})$  must have a time delay. and we want to compare this two time. We need to make the tow local clock with reference with a same global time.

d)

The P2 process after process send responses in order recived, that can make  $t_{P1}^{receive}(m_{res1}) < t_{P1}^{receive}(m_{res2})$  and the P2 send responses of first msg after send second msg from P1. So  $t_{P1}^{send}(m2) - t_{P1}^{send}(m1) < t_{delay} + t_{excution} + t_{drift}$ . This mean the time between P1 send first message and second message should less than the send delay time and excution time in P2 and drift time of P2.

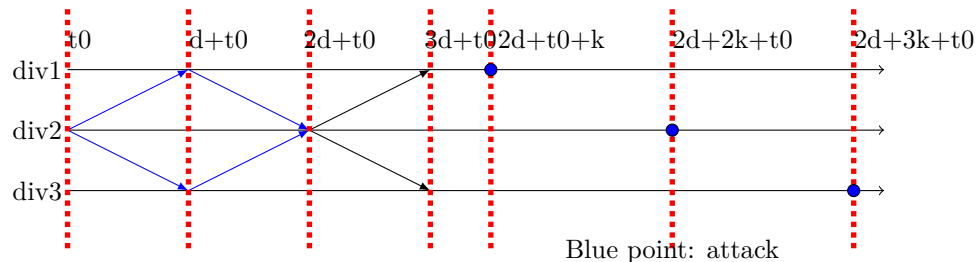
## Three-Army-Problem

a)

The assumptions is there should be no Failure in communication, this mean all the message can always send to the reciver. also the delay time between send and recive should not be too long( $d > k$ ).

Three army in the same time send the number of division. And then the largest send message to other 2 divisions about the attack time. Then at the attack time the largest division attack, after k mins the second-largest attack. And k mins later the smallest division attack.

b)



in this diagram. we assum the largest division is div1 and second-largest is div2 and smallest is div3. At time  $t_0$  div2 want to attack, and send message to div1 and div2. Then at time  $d$  div1 and div2 got the message and send back message about their size to div2. At  $2d$  time div2 know all three divisions size, and send

to div1 and div3 message about attack time. At  $2d+t_0+k$  time the biggest div1 start attack. After  $k$  time div2 and then div3.

## System Availablity

a)

$$A_X = \frac{60}{100} = 0.6$$

$$A_Y = \frac{80}{100} = 0.8$$

b)

$$A_{DS} = 1 - \bar{A}_X \bar{A}_Y = 0.92$$

c)

$$P(X = up|Y = up) = \frac{60}{80} = 0.75$$

$$P(X = up|Y = down) = 0$$

d)

accroding question c. if Y fail X must be fail:

$$P = 1 - 0.2 = 0.8$$