

Praktikum Rechnernetze und Verteilte Systeme

Block 6

— Zeitsynchronisation —

Termin: 5.-7.1.2015 & 12.-14.1.2015

1 Vertiefungsaufgaben

Diese Aufgaben sind zu Ihrer eigenen Vertiefung in Hinblick auf die Klausurvorbereitung gedacht:

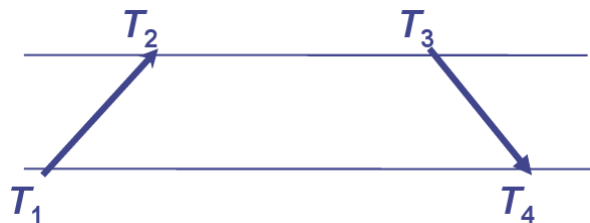
Musterlösung zu Aufgabe 7:

- Offset $\theta = \frac{(T_2 - T_1) + (T_3 - T_4)}{2}$
- Delay $\delta = \frac{(T_2 - T_1) + (T_4 - T_3)}{2}$
- Erklärung:
 - $T_1..T_4$ sind jeweils die Zeitstempel gemäß der lokalen Uhren
 - Offsetberechnung: Der Offset lässt sich herleiten, indem man den Delay der beiden Nachrichten als gleich annimmt, die Zeitstempel durcheinander ausdrückt und die beiden Gleichungen voneinander abzieht (Bitte in der Klausur dann auch ausrechnen):

$$T_2 = T_1 + \theta + \delta$$

$$T_4 = T_3 - \theta + \delta$$

- Delayberechnung: Durchschnitt der beiden Differenzen, Offset fällt dabei raus.

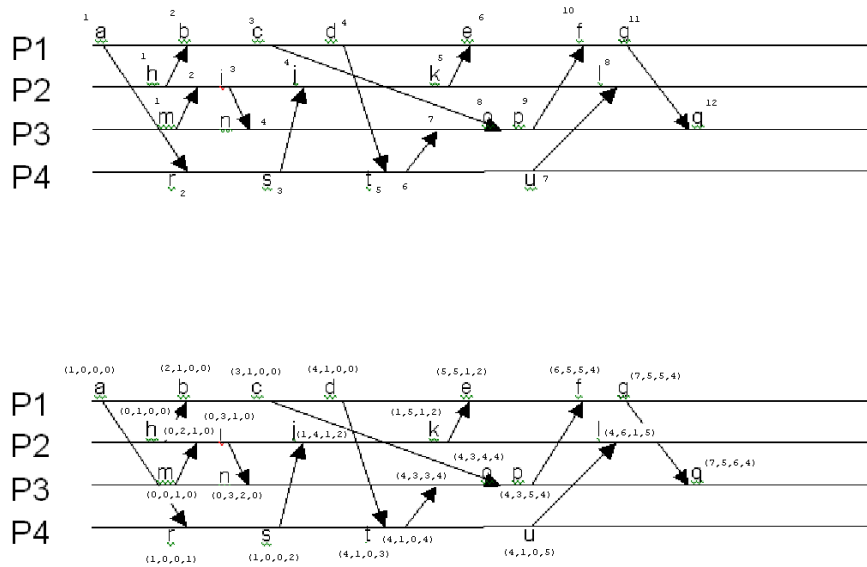


Musterlösung zu Aufgabe 8:

- Die Zeitstempel werden auf der Applikationsschicht erstellt und erfahren noch eine ungewisse Verzögerung, bis sie tatsächlich auch auf der Bitübertragungs-Schicht abgeschickt werden.

- b) Bei PTP werden die Zeitstempel wenn möglich direkt in der Hardware gesetzt, wenn das Paket auf Bitübertragungs-Schicht abgeschickt wird bzw. die Zeitstempel durch Interrupts an die Applikation gegeben. Es wird zudem ein zusätzliches Paket eingeführt, welches im Nachhinein den sehr genau gemessenen Zeitstempel enthält.
- c) gar nicht

Musterlösung zu Aufgabe 9:



Musterlösung zu Aufgabe 10:

- a)
 - Alle Knoten treffen die selbe Entscheidung
 - Die Knoten dürfen nach einer Bestätigung einer Entscheidung diese nicht mehr ändern
 - Die Entscheidung, eine Transaktion zu committen ist nur genau dann erlaubt, wenn alle Knoten committen können
- b)
 1. Koordinator sendet *VOTE_REQUEST* an alle Teilnehmer
 2. Jeder Teilnehmer, der dies erhält antwortet mit *VOTE_COMMIT* oder *VOTE_ABORT*
 3. Wenn alle Teilnehmer zugestimmt haben, committed der Koordinator selbst und sendet *GLOBAL_COMMIT* an alle. Andernfalls sendet er *GLOBAL_ABORT*.
 4. Jeder Teilnehmer welcher zuvor zum Commit zugestimmt hat, wartet auf das *GLOBAL_COMMIT* vom Koordinator. Sobald er es empfängt committed er lokal. Empfängt er ein *GLOBAL_ABORT*, bricht auch er lokal ab.
- c) Bei 2PC kann es dazu kommen, dass Knoten lange im “uncertain” Zustand verweilen wenn der Koordinator versagt und getDecision()-Nachrichten nicht wiederholt werden. (2PC blockiert dabei bis der Koordinator neu gestartet wurde.)