



Übersicht

Zeit in verteilten Systemen

- Problem der Synchronisierung
 - Verfahren von Cristian
 - Verfahren von Berkeley
 - NTP
- Logische Uhren
 - Lamport-Zeit
 - Vektor-Zeit
- Globale Zustände
 - Chandy/Lamport – Algorithmus (Snapshot)



Problem der Zeitsynchronisierung

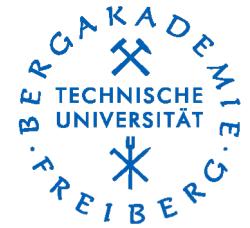
Warum ist eine Zeitsynchronisierung in verteilten Systemen überhaupt notwendig?



Problem der Zeitsynchronisierung

Warum ist eine Zeitsynchronisierung in verteilten Systemen überhaupt notwendig?

- Reihenfolge von Arbeitsschritten



Problem der Zeitsynchronisierung

Warum ist eine Zeitsynchronisierung in verteilten Systemen überhaupt notwendig?

- Reihenfolge von Arbeitsschritten
- Konsistenz von Daten



Problem der Zeitsynchronisierung

Warum ist eine Zeitsynchronisierung in verteilten Systemen überhaupt notwendig?

- Reihenfolge von Arbeitsschritten
- Konsistenz von Daten
- Sicherheit



Problem der Zeitsynchronisierung

Warum ist eine Zeitsynchronisierung in verteilten Systemen überhaupt notwendig?

- Reihenfolge von Arbeitsschritten
- Konsistenz von Daten
- Sicherheit
- Ortung



Problem der Zeitsynchronisierung

Warum ist eine Zeitsynchronisierung in verteilten Systemen überhaupt notwendig?

- Reihenfolge von Arbeitsschritten
- Konsistenz von Daten
- Sicherheit
- Ortung
- Nutzer möchte Uhrzeit evtl. wissen



Problem der Zeitsynchronisierung

Warum kann die Zeit in gleich initialisierten Systemen abweichen?



Problem der Zeitsynchronisierung

Warum kann die Zeit in gleich initialisierten Systemen abweichen?

- Clock Drift (Uhrenfehler)
 - Standfehler (fixed offset)
 - Gangfehler



Problem der Zeitsynchronisierung

Warum kann die Zeit in gleich initialisierten Systemen abweichen?

- Clock Drift (Uhrenfehler)
 - Standfehler (fixed offset)
 - Gangfehler

Ursachen:

- Temperatur, Luftdruck, Geschwindigkeit (Relativitätstheorie)



Problem der Zeitsynchronisierung

Warum kann die Zeit in gleich initialisierten Systemen abweichen?

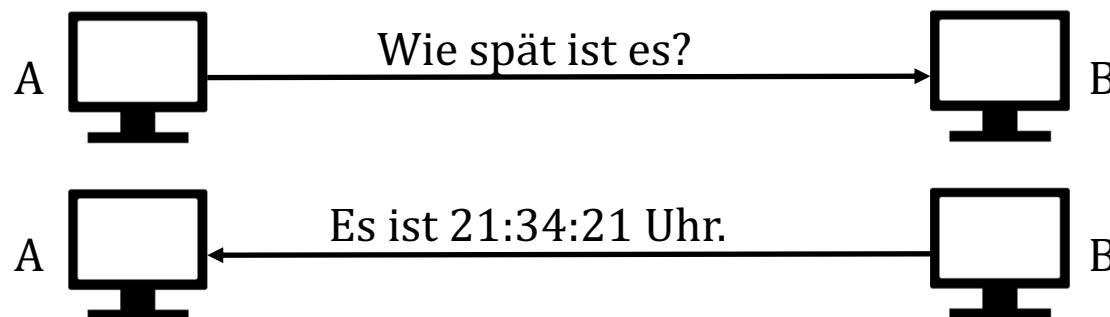
- Clock Drift (Uhrenfehler)
 - Standfehler (fixed offset)
 - Gangfehler

Ursachen:

- Temperatur, Luftdruck, Geschwindigkeit (Relativitätstheorie)
- Clock Skew (Taktversatz)

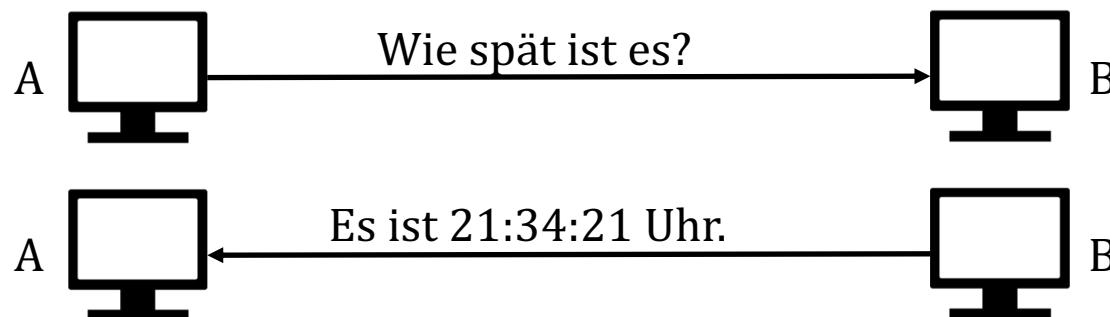
Problem der Zeitsynchronisierung

Welches grundsätzliche Problem muss von Zeitsynchronisierungsmechanismen in verteilten Systemen beachtet werden?



Problem der Zeitsynchronisierung

Welches grundsätzliche Problem muss von Zeitsynchronisierungsmechanismen in verteilten Systemen beachtet werden?



Verzögerung durch:

- Signallaufzeit im Medium (Luft, Kupfer, Licht...)
- Kollisionen
- Verarbeitungsverzögerung (Routing, Buffer, Prozessorscheduling)

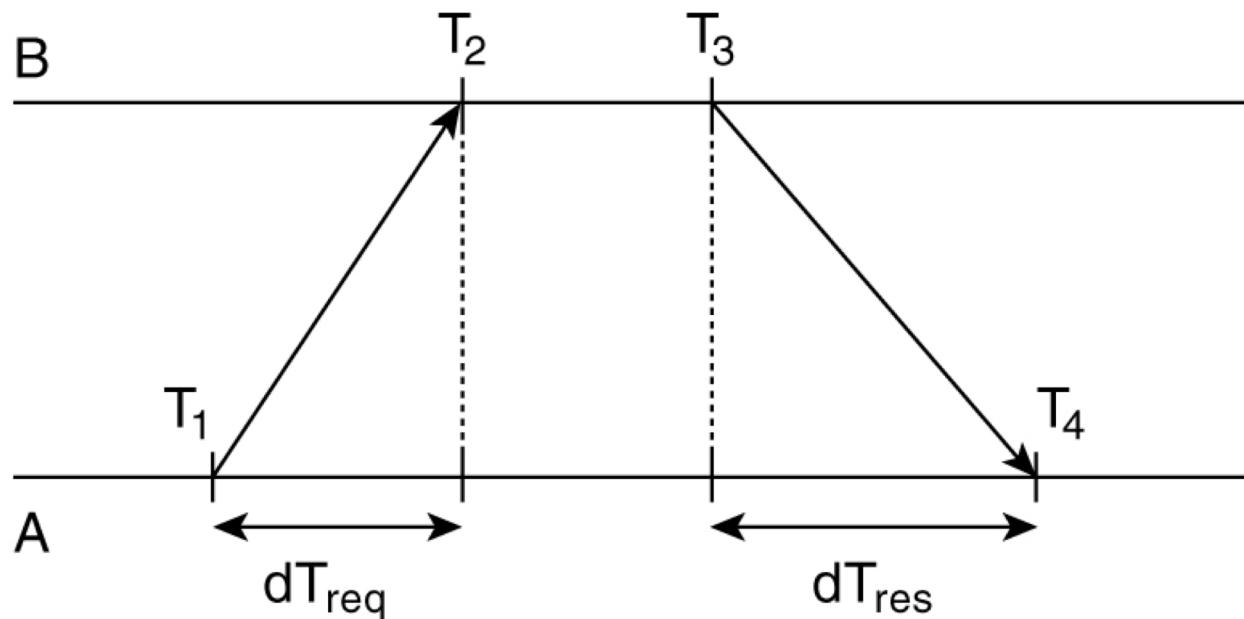


Problem der Zeitsynchronisierung

Wie wirkt das Verfahren von Cristian der Verzögerung bei der Zeitsynchronisation entgegen?

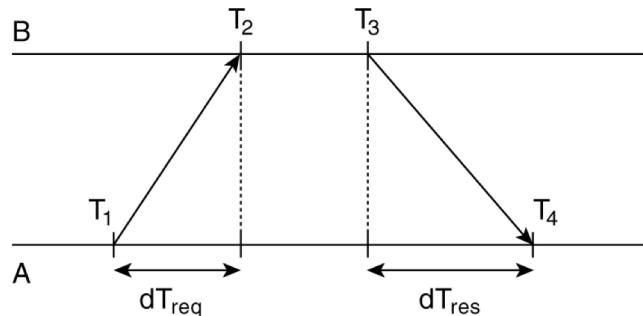
Problem der Zeitsynchronisierung

Wie wirkt das Verfahren von Cristian der Verzögerung bei der Zeitsynchronisation entgegen?



Problem der Zeitsynchronisierung

Wie wirkt das Verfahren von Cristian der Verzögerung bei der Zeitsynchronisation entgegen?

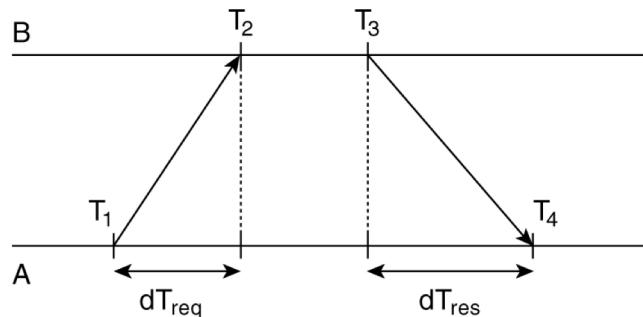


Einfache Abschätzung der Signallaufzeit:

$$S = \frac{T_4 - T_1}{2}$$

Problem der Zeitsynchronisierung

Wie wirkt das Verfahren von Cristian der Verzögerung bei der Zeitsynchronisation entgegen?



Einfache Abschätzung der Signallaufzeit:

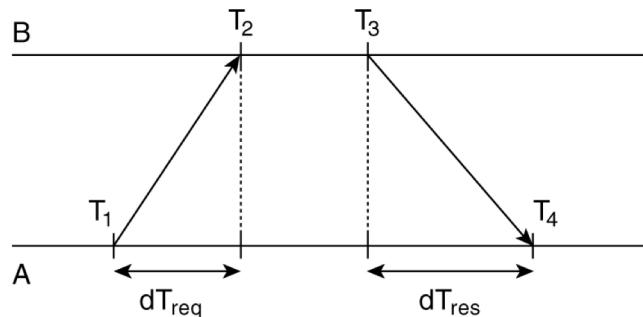
$$S = \frac{T_4 - T_1}{2}$$

Bessere Abschätzung, wenn bekannt:

$$S = \frac{T_2 - T_1 + T_4 - T_3}{2}$$

Problem der Zeitsynchronisierung

Wie wirkt das Verfahren von Cristian der Verzögerung bei der Zeitsynchronisation entgegen?



Einfache Abschätzung der Signallaufzeit:

$$S = \frac{T_4 - T_1}{2}$$

Bessere Abschätzung, wenn bekannt:

$$S = \frac{T_2 - T_1 + T_4 - T_3}{2}$$

RTT < 2 * gewünschte Genauigkeit

Neue Zeit:

von B gelieferte Zeit + S



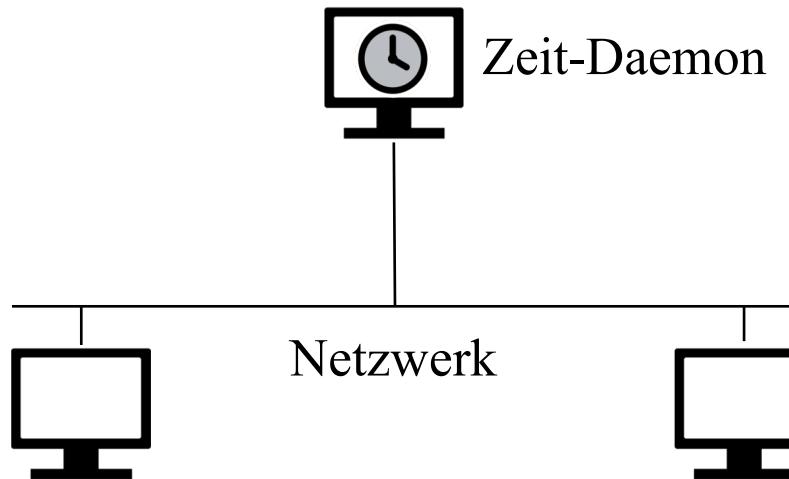
Problem der Zeitsynchronisierung

Wie funktioniert der Berkeley-Algorithmus?



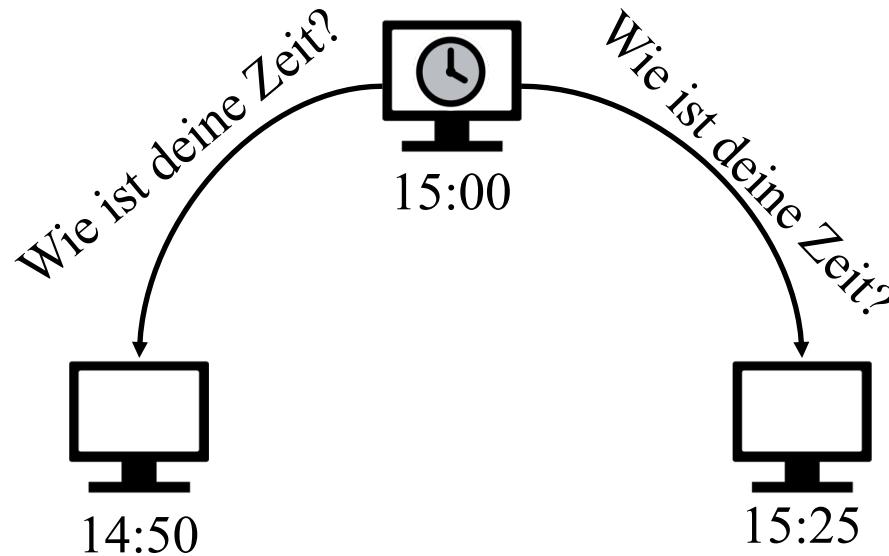
Problem der Zeitsynchronisierung

Wie funktioniert der Berkeley-Algorithmus?



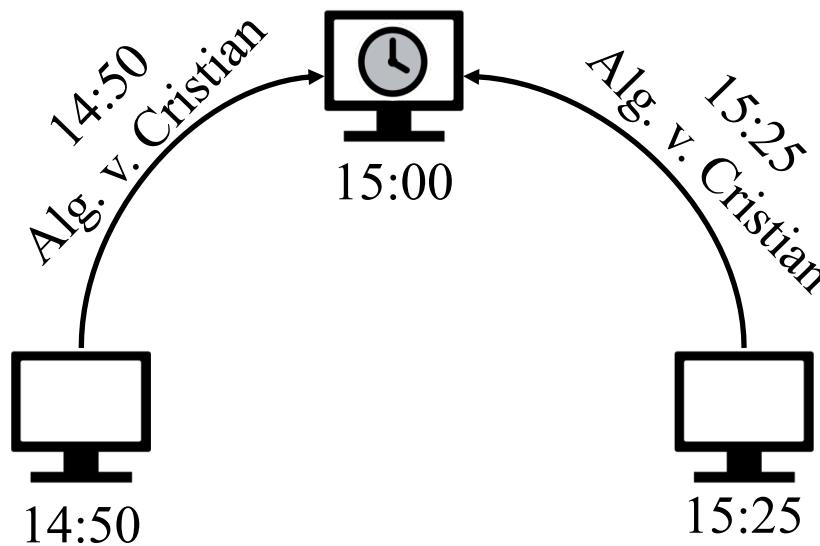
Problem der Zeitsynchronisierung

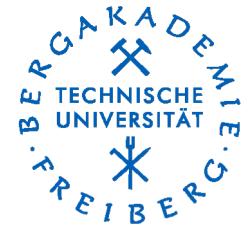
Wie funktioniert der Berkeley-Algorithmus?



Problem der Zeitsynchronisierung

Wie funktioniert der Berkeley-Algorithmus?





Problem der Zeitsynchronisierung

Wie funktioniert der Berkeley-Algorithmus?



15:00
14:50
15:25

Berechne „Durchschnittszeit“
(ohne Ausreißer!)



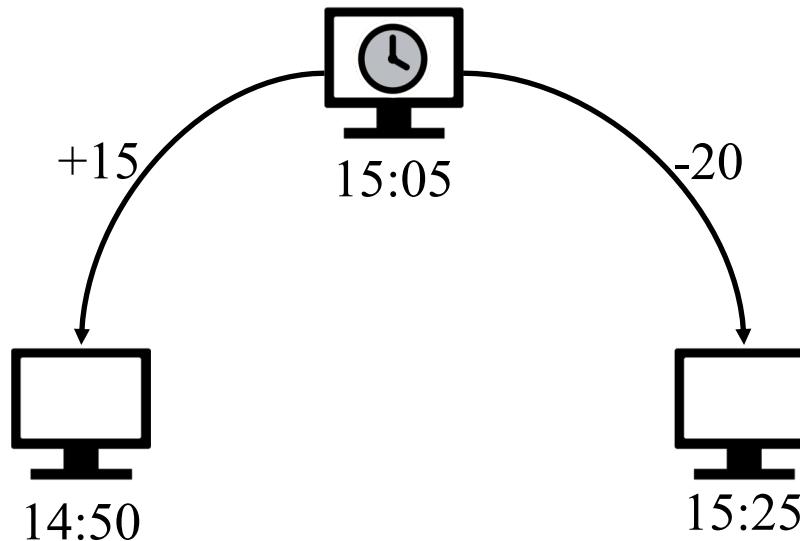
14:50



15:25

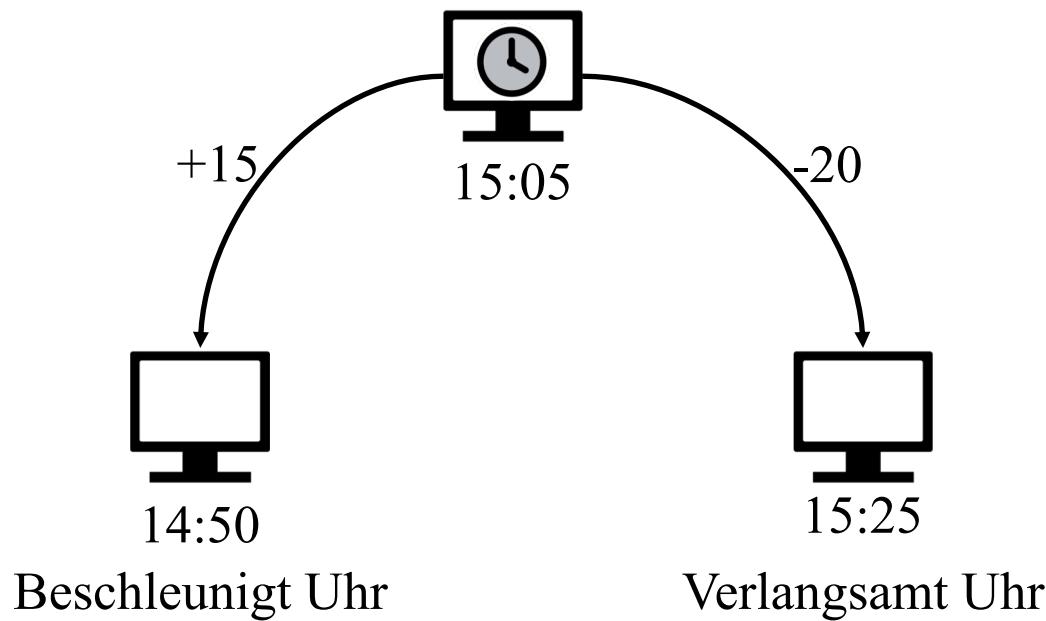
Problem der Zeitsynchronisierung

Wie funktioniert der Berkeley-Algorithmus?



Problem der Zeitsynchronisierung

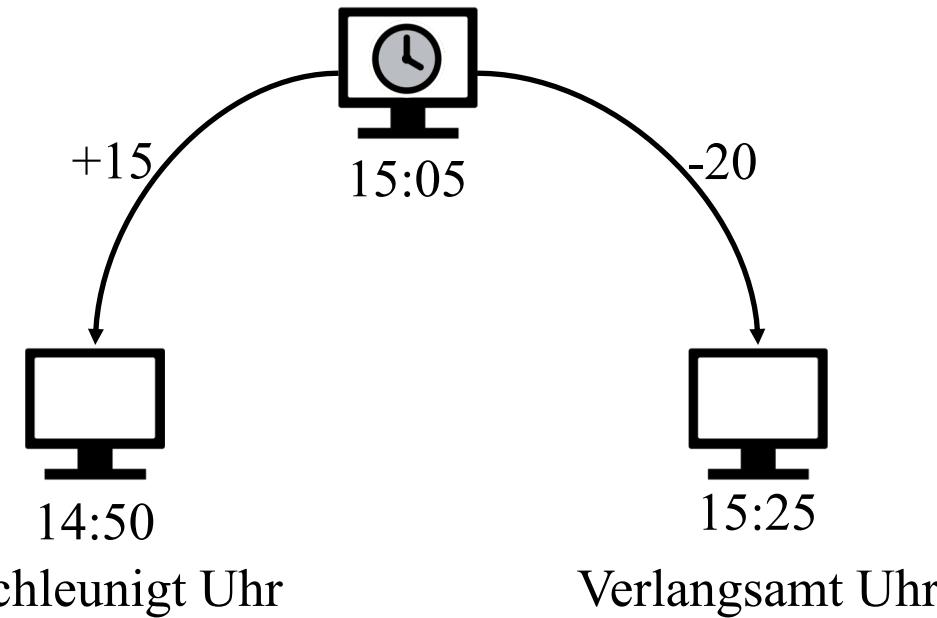
Wie funktioniert der Berkeley-Algorithmus?



Problem der Zeitsynchronisierung

Wie funktioniert der Berkeley-Algorithmus?

Warum keine
direkte Anpassung
der Zeit?





Problem der Zeitsynchronisierung

Wie funktioniert der Berkeley-Algorithmus?



15:10



15:10



15:10



Problem der Zeitsynchronisierung

Wie arbeitet grundsätzlich das NTP und was ist der Unterschied zum Berkeley-Algorithmus?



Problem der Zeitsynchronisierung

Wie arbeitet grundsätzlich das NTP und was ist der Unterschied zum Berkeley-Algorithmus?

- Network Time Protocol nutzt UDP auf Port 123
- 1985 veröffentlicht ([RFC 958](#))
- nativ auf allen UNIXoiden BS und Windows



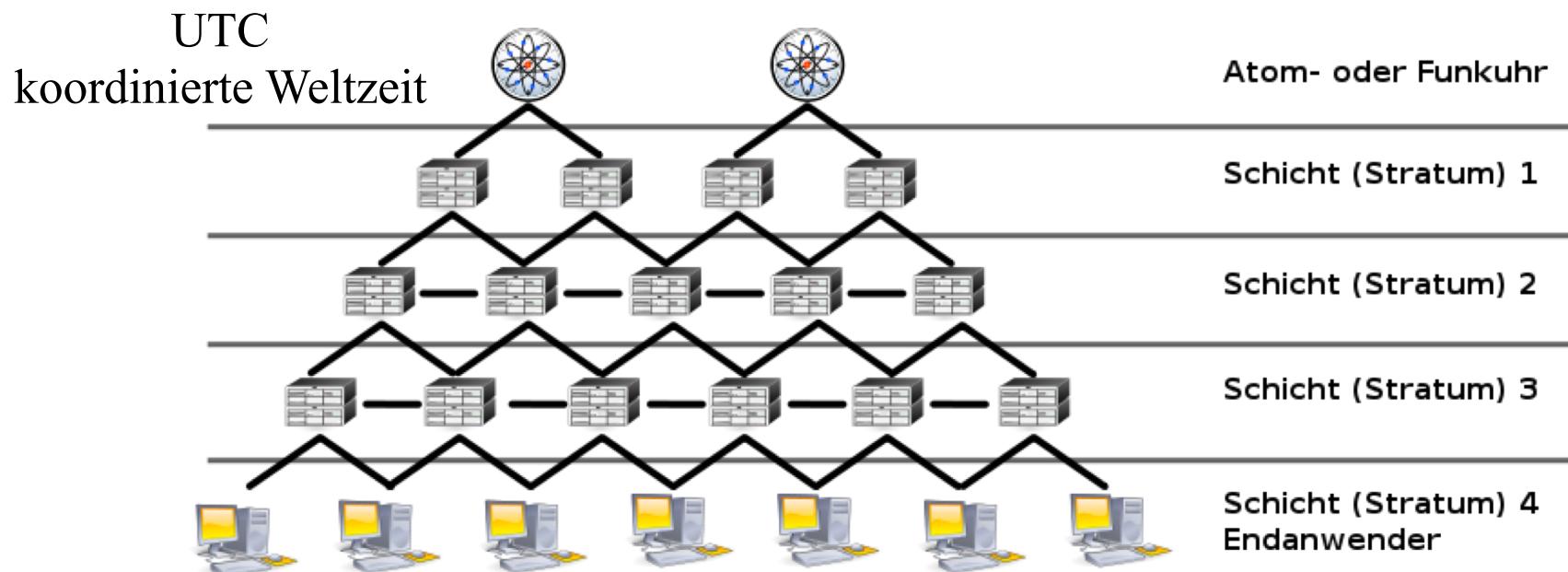
Problem der Zeitsynchronisierung

Wie arbeitet grundsätzlich das NTP und was ist der Unterschied zum Berkeley-Algorithmus?

- Network Time Protocol nutzt UDP auf Port 123
- 1985 veröffentlicht ([RFC 958](#))
- nativ auf allen UNIXoiden BS und Windows
- Zeitstempel 64 Bit lang
 - 32 Bit für Sekunden seit 1. Januar 1900 00:00:00 Uhr
 - 32 Bit für Sekundenbruchteile

Problem der Zeitsynchronisierung

Wie arbeitet grundsätzlich das NTP und was ist der Unterschied zum Berkeley-Algorithmus?

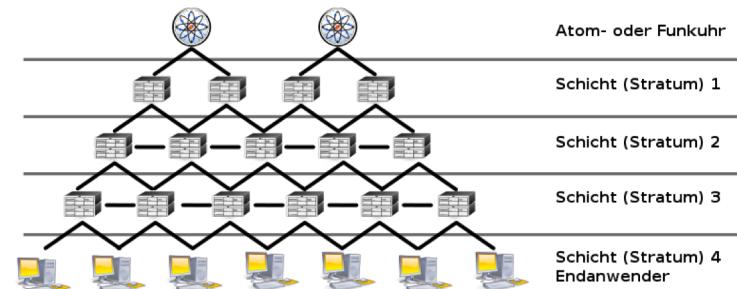




Problem der Zeitsynchronisierung

Wie arbeitet grundsätzlich das NTP und was ist der Unterschied zum Berkeley-Algorithmus?

- Server von Stratum n synchronisieren sich mit Servern von Stratum n und n-1
- Jeder Client/Server nutzt Bellman-Ford-Algorithmus, um RTT zum Stratum 1 zu minimieren



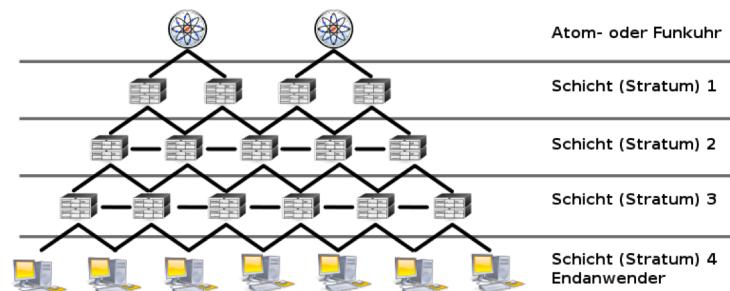


Problem der Zeitsynchronisierung

Wie arbeitet grundsätzlich das NTP und was ist der Unterschied zum Berkeley-Algorithmus?

Unterschiede zum Berkeley-Algorithmus:

- UTC Server / Atomuhren
- Synchronisieren auf „globaler“ Ebene
- Kein zentraler Zeitdaemon





Logische Uhren

Was ist der Unterschied zwischen einer logischen und einer „normalen“ Uhr?



Logische Uhren

Was ist der Unterschied zwischen einer logischen und einer „normalen“ Uhr?

- Zeitstempel \neq „reale“ Zeit
- Zeitstempel lassen nur Kausalordnung zu
- Zeitstempel wird eventbasiert inkrementiert



Logische Uhren

Welche Eigenschaften hat eine logische Uhr?



Logische Uhren

Welche Eigenschaften hat eine logische Uhr?

- (schwaches) Konsistenzkriterium für Uhren (happen-before)
 - e1 ist Ursache von e2 → Zeitstempel(e1) ≤ Zeitstempel(e2)



Logische Uhren

Welche Eigenschaften hat eine logische Uhr?

- (schwaches) Konsistenzkriterium für Uhren (happen-before)
 - e_1 ist Ursache von $e_2 \rightarrow \text{Zeitstempel}(e_1) \leq \text{Zeitstempel}(e_2)$
- Starkes Konsistenzkriterium für Uhren
 - $\text{Zeitstempel}(e_1) \leq \text{Zeitstempel}(e_2) \rightarrow e_1$ ist Ursache von e_2



Logische Uhren

Welche Eigenschaften hat eine logische Uhr?

- (schwaches) Konsistenzkriterium für Uhren (happen-before)
 - e_1 ist Ursache von $e_2 \rightarrow \text{Zeitstempel}(e_1) \leq \text{Zeitstempel}(e_2)$
- Starkes Konsistenzkriterium für Uhren
 - $\text{Zeitstempel}(e_1) \leq \text{Zeitstempel}(e_2) \rightarrow e_1$ ist Ursache von e_2
- Nebenläufigkeit
 - e_1 nicht Ursache von e_2 und umgekehrt $\rightarrow e_1 \parallel e_2$ bzw. $e_2 \parallel e_1$



Logische Uhren

Wie funktioniert eine Lamport-Uhr?



Logische Uhren

Wie funktioniert eine Lamport-Uhr?

- Jeder Prozess hat eigene Uhr/Zähler



Logische Uhren

Wie funktioniert eine Lamport-Uhr?

- Jeder Prozess hat eigene Uhr/Zähler
- Inkrementiere Zähler vor jedem Ereignis



Logische Uhren

Wie funktioniert eine Lamport-Uhr?

- Jeder Prozess hat eigene Uhr/Zähler
- Inkrementiere Zähler vor jedem Ereignis
- Senden: aktuelle Zeit wird an Nachrichten gehängt



Logische Uhren

Wie funktioniert eine Lamport-Uhr?

- Jeder Prozess hat eigene Uhr/Zähler
- Inkrementiere Zähler vor jedem Ereignis
- Senden: aktuelle Zeit wird an Nachrichten gehängt
- Empfangen: Maximum der eigenen und gesendeten Zeit wird inkrementiert

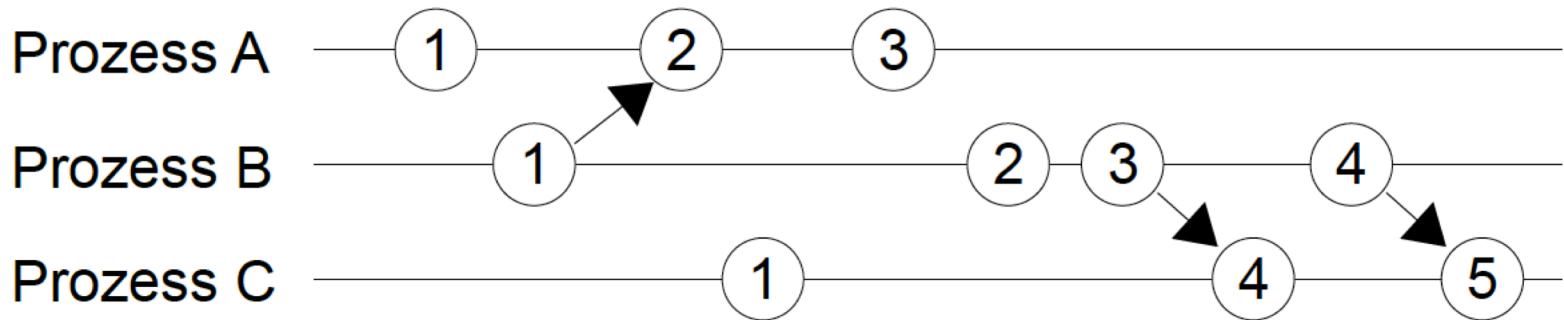


Logische Uhren

Zeichne für folgende Ereignisse ein Zeitdiagramm für die Prozesse A, B und C inkl. Pfeile für Sende- und Empfangsereignisse und trage die Lamportzeit für alle Ereignisse ein: A, B → A, C, A, B, B → C, B → C

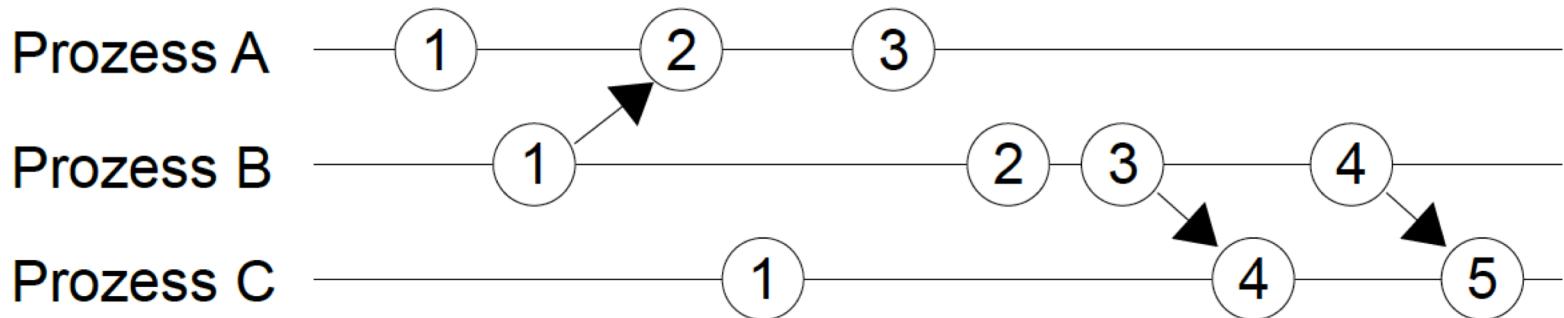
Logische Uhren

Zeichne für folgende Ereignisse ein Zeitdiagramm für die Prozesse A, B und C inkl. Pfeile für Sende- und Empfangsereignisse und trage die Lamportzeit für alle Ereignisse ein: A, B → A, C, A, B, B → C, B → C



Logische Uhren

Zeichne für folgende Ereignisse ein Zeitdiagramm für die Prozesse A, B und C inkl. Pfeile für Sende- und Empfangsereignisse und trage die Lamportzeit für alle Ereignisse ein: A, B → A, C, A, B, B → C, B → C



A1, B1, C1, A2, B2, A3, B3, C4, B4, C5



Logische Uhren

Bei der Vektorzeit halten alle Prozesse (wenn bekannt) den Wert der anderen Prozesszähler. Welchen Vorteil biete dies gegenüber der Lamportzeit? A, B→A, C, A, B, B→C, B→C



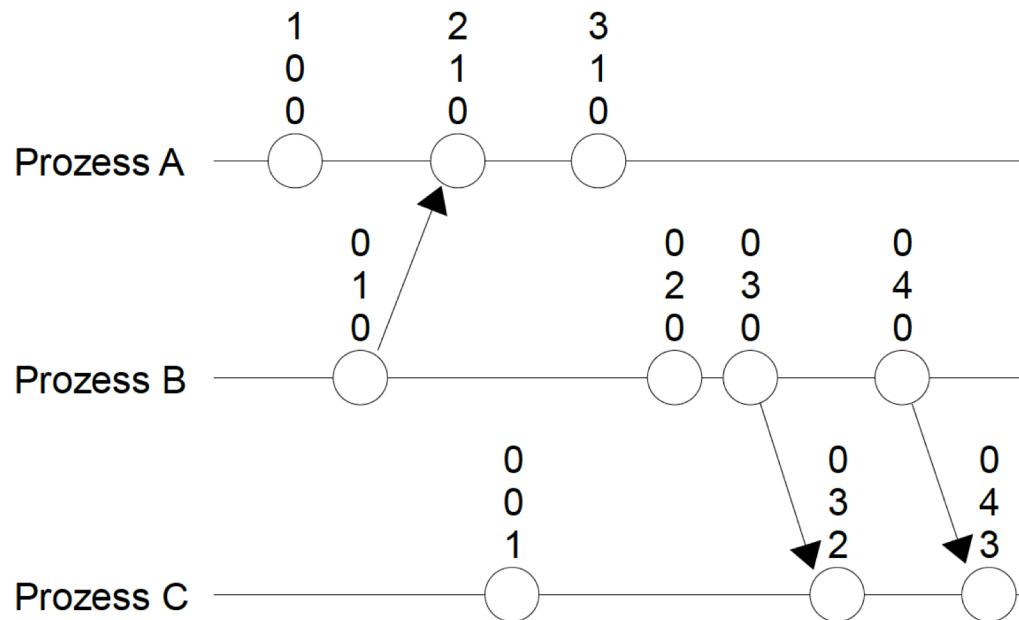
Logische Uhren

Bei der Vektorzeit halten alle Prozesse (wenn bekannt) den Wert der anderen Prozesszähler. Welchen Vorteil biete dies gegenüber der Lamportzeit? A, B → A, C, A, B, B → C, B → C

- Mit Hilfe der Vektorzeit lässt sich Nebenläufigkeit erkennen
- Lamportzeit Gegenbsp: C1 und B3

Logische Uhren

Bei der Vektorzeit halten alle Prozesse (wenn bekannt) den Wert der anderen Prozesszähler. Welchen Vorteil biete dies gegenüber der Lamportzeit? A, B→A, C, A, B, B→C, B→C



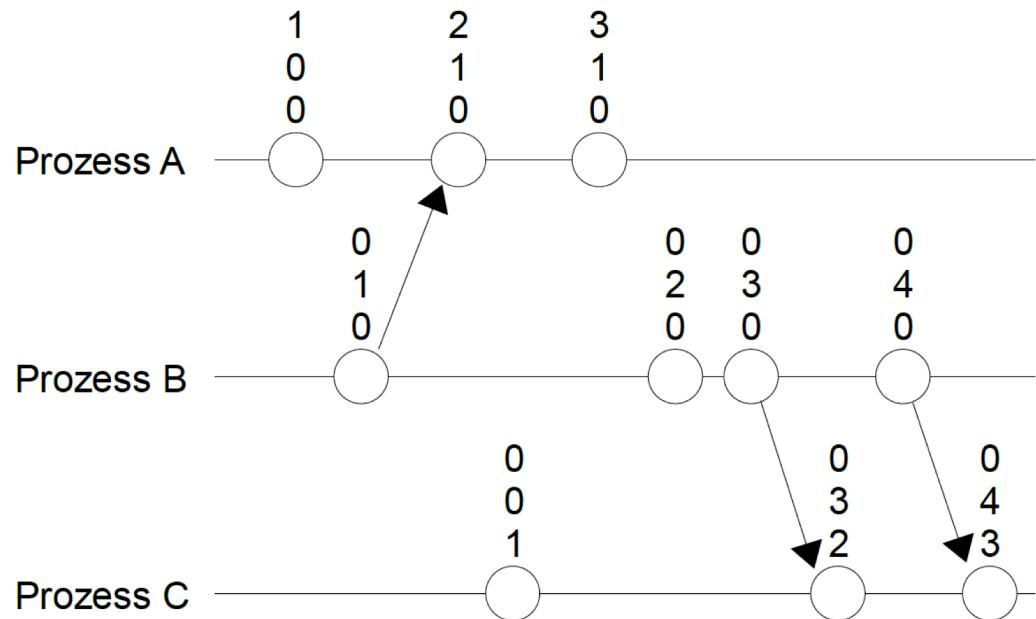
Logische Uhren

Bei der Vektorzeit halten alle Prozesse (wenn bekannt) den Wert der anderen Prozesszähler. Welchen Vorteil biete dies gegenüber der Lamportzeit? A, B → A, C, A, B, B → C, B → C

A(1,0,0)A(2,1,0)A(3,1,0)

B(0,1,0)B(0,2,0)B(0,3,0)B(0,4,0)

C(0,0,1)C(0,3,2)C(0,4,3)





Logische Uhren

Bei der Vektorzeit halten alle Prozesse (wenn bekannt) den Wert der anderen Prozesszähler. Welchen Vorteil biete dies gegenüber der Lamportzeit? A, B → A, C, A, B, B → C, B → C

Bsp:

A(1,0,0)A(2,1,0)A(3,1,0)

B(0,1,0) ≤ A(2,1,0) – B(0,1,0) → A(2,1,0)

B(0,1,0)B(0,2,0)B(0,3,0)B(0,4,0)

!(C(0,0,1) ≤ A(1,0,0)) – C(0,0,1) || A(1,0,0)

C(0,0,1)C(0,3,2)C(0,4,3)

B(0,1,0) ≤ C(0,4,3) – B(0,1,0) → C(0,4,3)

!(A(3,1,0) ≤ C(0,3,2)) – A(3,1,0) || C(0,3,2)

!(C(0,0,1) ≤ B(0,3,0)) – C(0,0,1) || B(0,3,0)



Globale Zustände

Warum ist es nützlich den globalen Zustand in einem verteilten System bestimmen zu können?



Globale Zustände

Warum ist es nützlich den globalen Zustand in einem verteilten System bestimmen zu können?

- Gemeinsamer Zugriff auf Daten (wird Datei gelesen/geschrieben?)



Globale Zustände

Warum ist es nützlich den globalen Zustand in einem verteilten System bestimmen zu können?

- Gemeinsamer Zugriff auf Daten (wird Datei gelesen/geschrieben?)
- Terminierungsbedingung erreicht?



Globale Zustände

Warum ist es nützlich den globalen Zustand in einem verteilten System bestimmen zu können?

- Gemeinsamer Zugriff auf Daten (wird Datei gelesen/geschrieben?)
- Terminierungsbedingung erreicht?
- Deadlocks erkennen



Globale Zustände

Warum ist es nützlich den globalen Zustand in einem verteilten System bestimmen zu können?

- Gemeinsamer Zugriff auf Daten (wird Datei gelesen/geschrieben?)
- Terminierungsbedingung erreicht?
- Deadlocks erkennen
- Konsistenzerkennung



Globale Zustände

Warum ist es nützlich den globalen Zustand in einem verteilten System bestimmen zu können?

- Gemeinsamer Zugriff auf Daten (wird Datei gelesen/geschrieben?)
- Terminierungsbedingung erreicht?
- Deadlocks erkennen
- Konsistenzerkennung
- Verteiltes Debugging

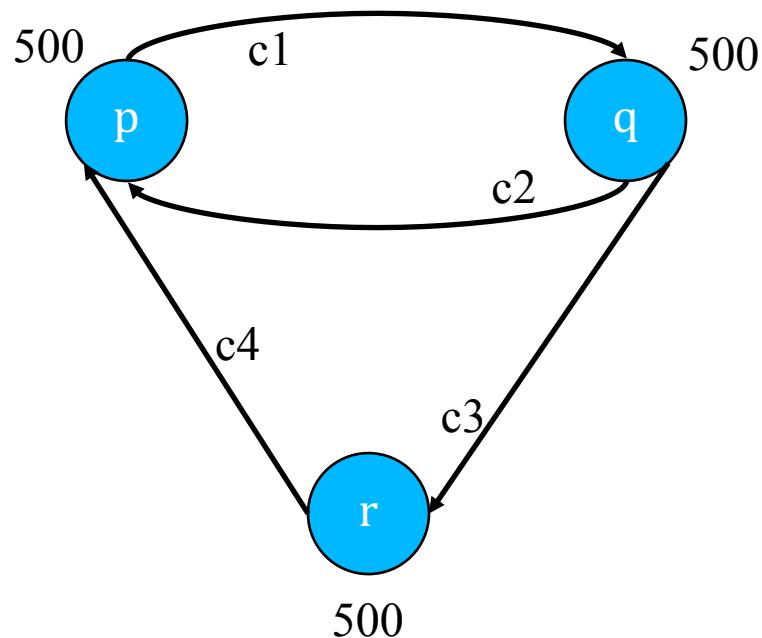


Globale Zustände

Wie kann man den globalen Zustand eines verteilten Systems mit Hilfe des Chandy-Lamport Algorithmus ermitteln?

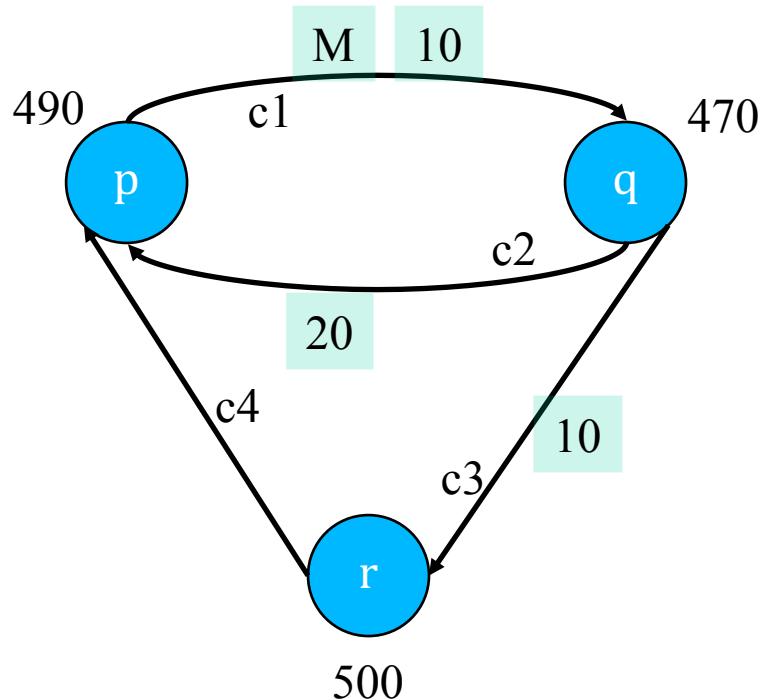
Globale Zustände

Wie kann man den globalen Zustand eines verteilten Systems mit Hilfe des Chandy-Lamport Algorithmus ermitteln?



Globale Zustände

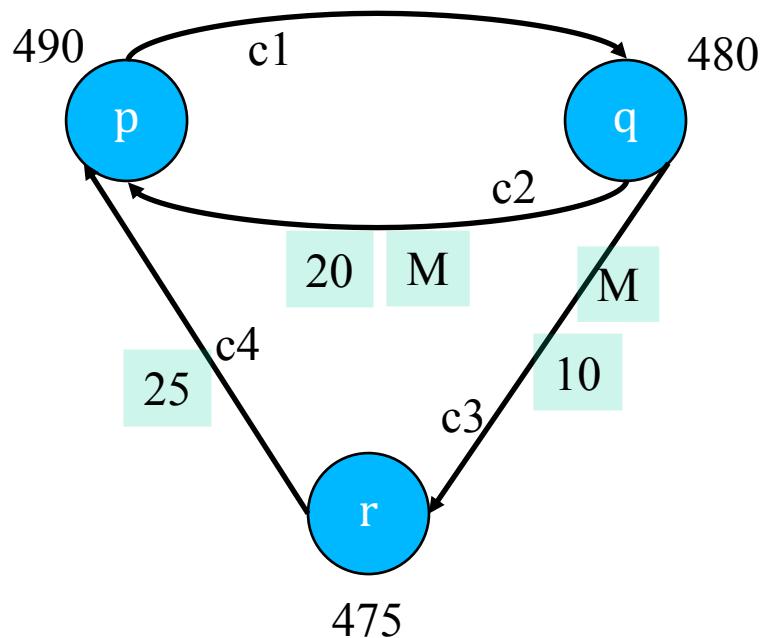
Wie kann man den globalen Zustand eines verteilten Systems mit Hilfe des Chandy-Lamport Algorithmus ermitteln?



	state	c1	c2	c3	c4
p	490		{}		{}
q		{}			
r				{}	

Globale Zustände

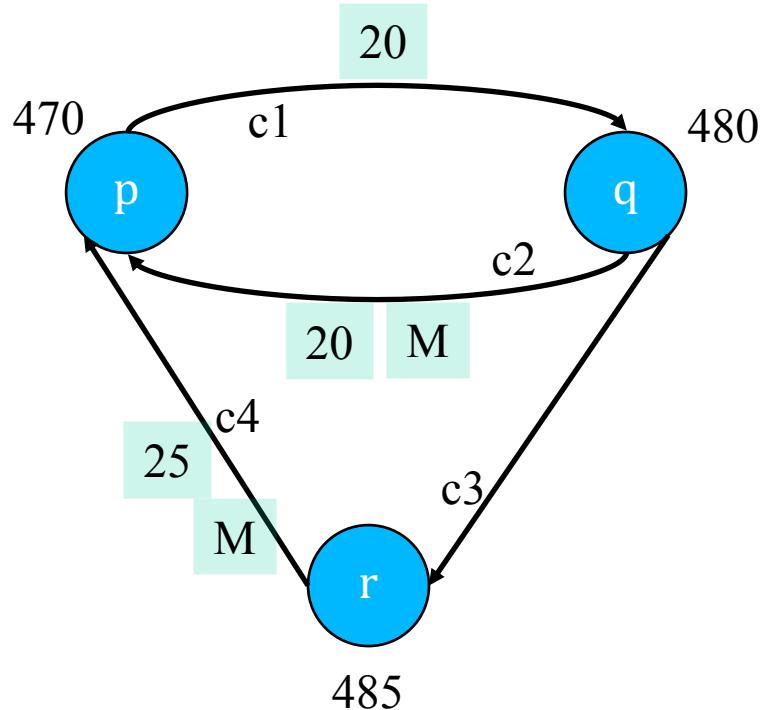
Wie kann man den globalen Zustand eines verteilten Systems mit Hilfe des Chandy-Lamport Algorithmus ermitteln?



	state	c1	c2	c3	c4
p	490		{}		{}
q	480	{}			
r				{}	

Globale Zustände

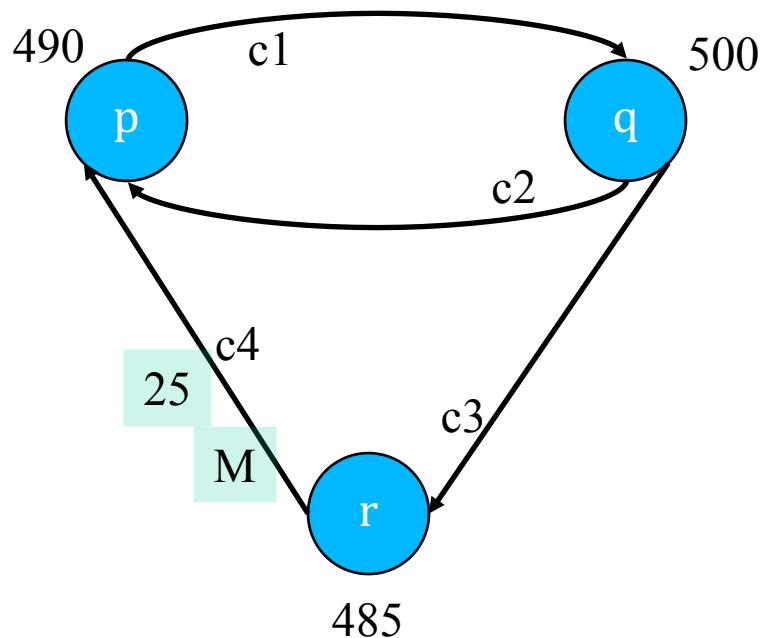
Wie kann man den globalen Zustand eines verteilten Systems mit Hilfe des Chandy-Lamport Algorithmus ermitteln?



	state	c1	c2	c3	c4
p	490		{}		{}
q	480	{}			
r	485			{}	

Globale Zustände

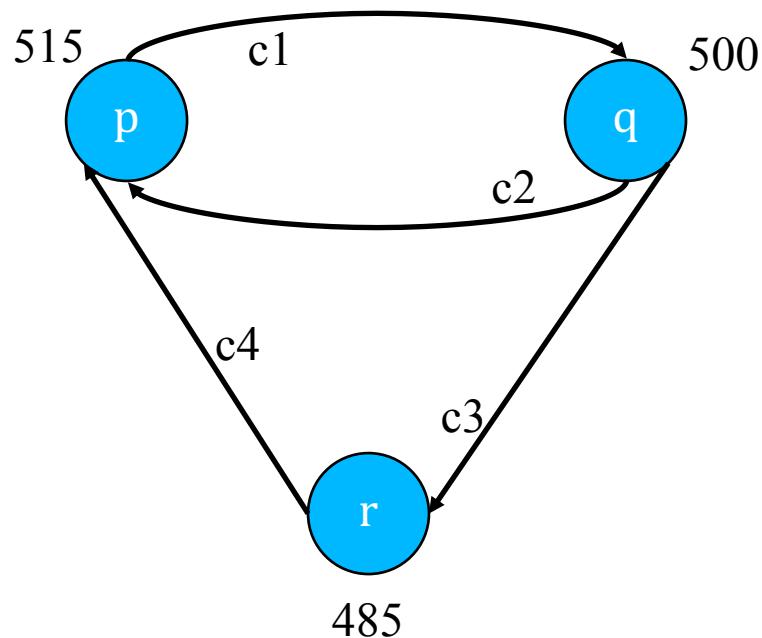
Wie kann man den globalen Zustand eines verteilten Systems mit Hilfe des Chandy-Lamport Algorithmus ermitteln?



	state	c1	c2	c3	c4
p	490		{20}		{}
q	480	{}			
r	485			{}	

Globale Zustände

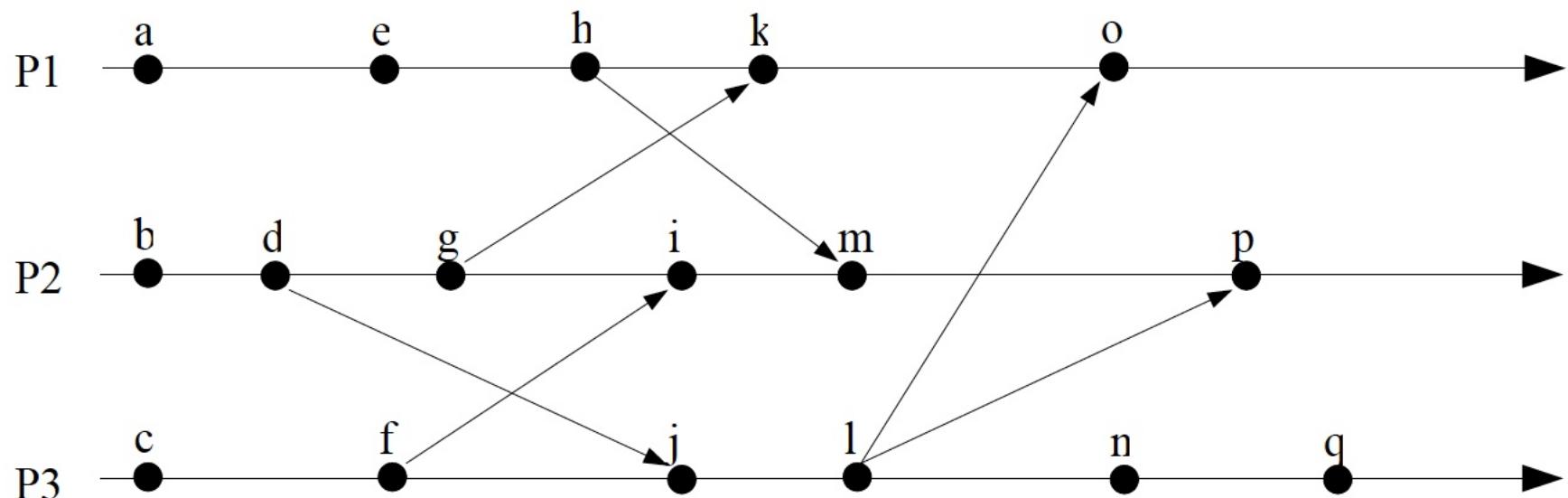
Wie kann man den globalen Zustand eines verteilten Systems mit Hilfe des Chandy-Lamport Algorithmus ermitteln?



	state	c1	c2	c3	c4
p	490		{20}		{25}
q	480	{}			
r	485			{}	

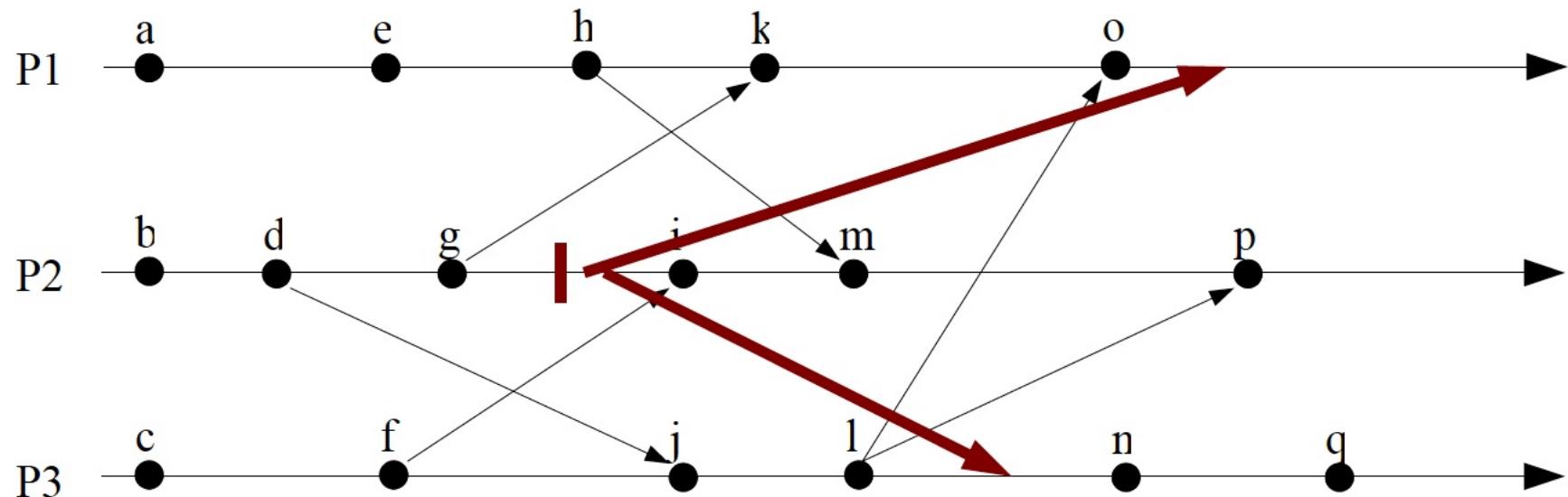
Globale Zustände

P2 löst zwischen den Ereignissen g und i den Chandy-Lamport Algorithmus zur Erstellung eines Schnappschusses aus. Ist es möglich, dass die erste Marke bei P1 nach o und bei P3 zwischen l und n eintrifft?



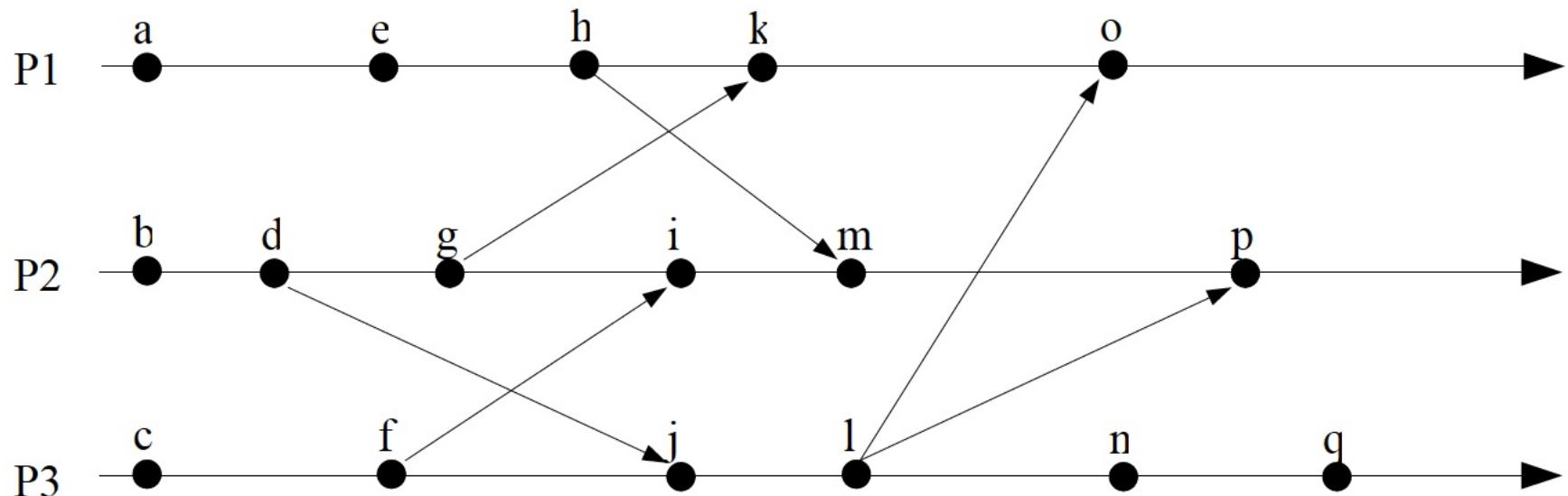
Globale Zustände

P2 löst zwischen den Ereignissen g und i den Chandy-Lamport Algorithmus zur Erstellung eines Schnappschusses aus. Ist es möglich, dass die erste Marke bei P1 nach o und bei P3 zwischen l und n eintrifft?



Globale Zustände

P2 löst zwischen den Ereignissen g und i den Chandy-Lamport Algorithmus zur Erstellung eines Schnappschusses aus. Ist es möglich, dass die erste Marke bei P1 nach o und bei P3 zwischen j und l eintrifft?



Globale Zustände

P2 löst zwischen den Ereignissen g und i den Chandy-Lamport Algorithmus zur Erstellung eines Schnappschusses aus. Ist es möglich, dass die erste Marke bei P1 nach o und bei P3 zwischen j und l eintrifft?

