



Einsatz und Realisierung von Datenbanksystemen

ERDB Übungsleitung

Alice Rey, Maximilian Bandle, Michael Jungmair

i3erdb@in.tum.de

Folien erstellt von Maximilian Bandle & Alexander Beischl



Organisatorisches

Disclaimer

Die Folien werden von der Übungsleitung allen Tutoren zur Verfügung gestellt.

Sollte es Unstimmigkeiten zu den Vorlesungsfolien von Prof. Kemper geben, so sind die Folien aus der Vorlesung ausschlaggebend.

Falls Ihr einen Fehler oder eine Unstimmigkeit findet, schreibt an i3erdb@in.tum.de mit Angabe der Foliennummer.



Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

- Ausgleich der Leistungsfähigkeit zwischen Lese- und Änderungstransaktionen
- ➔ Teilweise Verlagerung des Overheads von Änderungs- zu Lesetransaktionen:
 - Kopien A_i von A werden individuelle Gewichte zugeordnet
- Lesequorum $Q_r(A)$
- Schreibquorum $Q_w(A)$

Folgende Bedingungen müssen gelten:

1. $Q_r(A) + Q_w(A) > W(A)$
2. $Q_r(A) + Q_w(A) > W(A)$

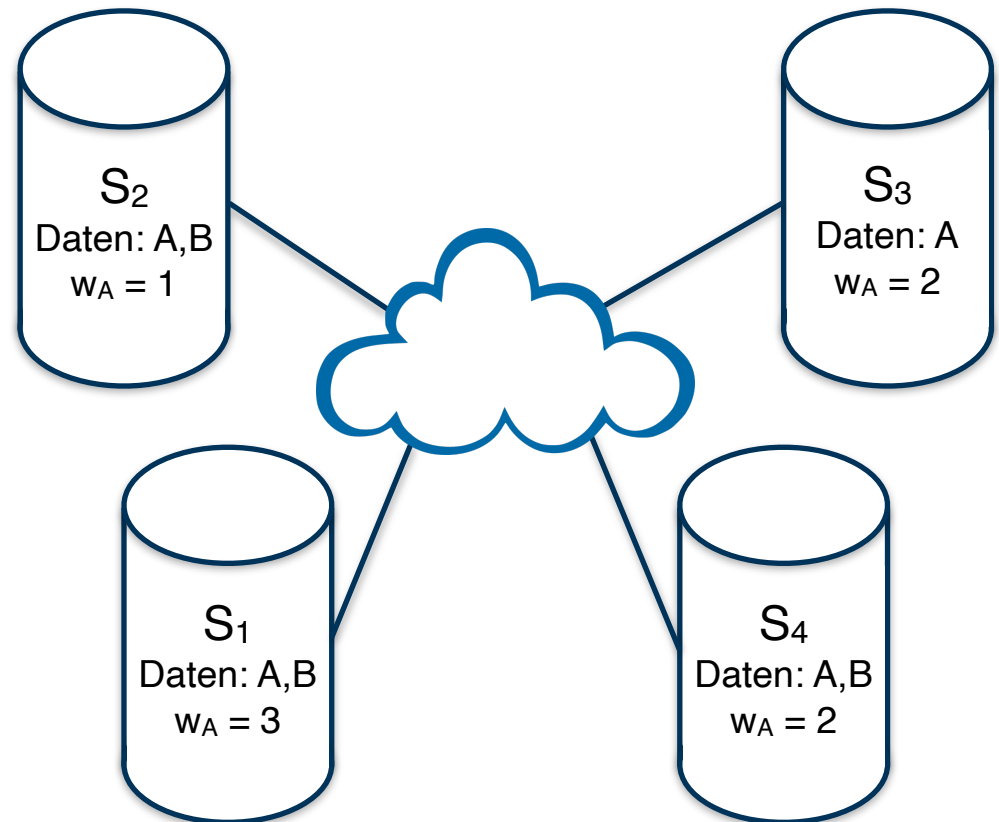
Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Berechne das Schreib- Q_w und Lesequorum Q_r für A und B.

$$W(A) = \sum_{i=1} w_i(A)$$

1. $Q_w(A) + Q_w(A) > W(A)$
2. $Q_r(A) + Q_w(A) > W(A)$



Verteilte Datenbanksysteme

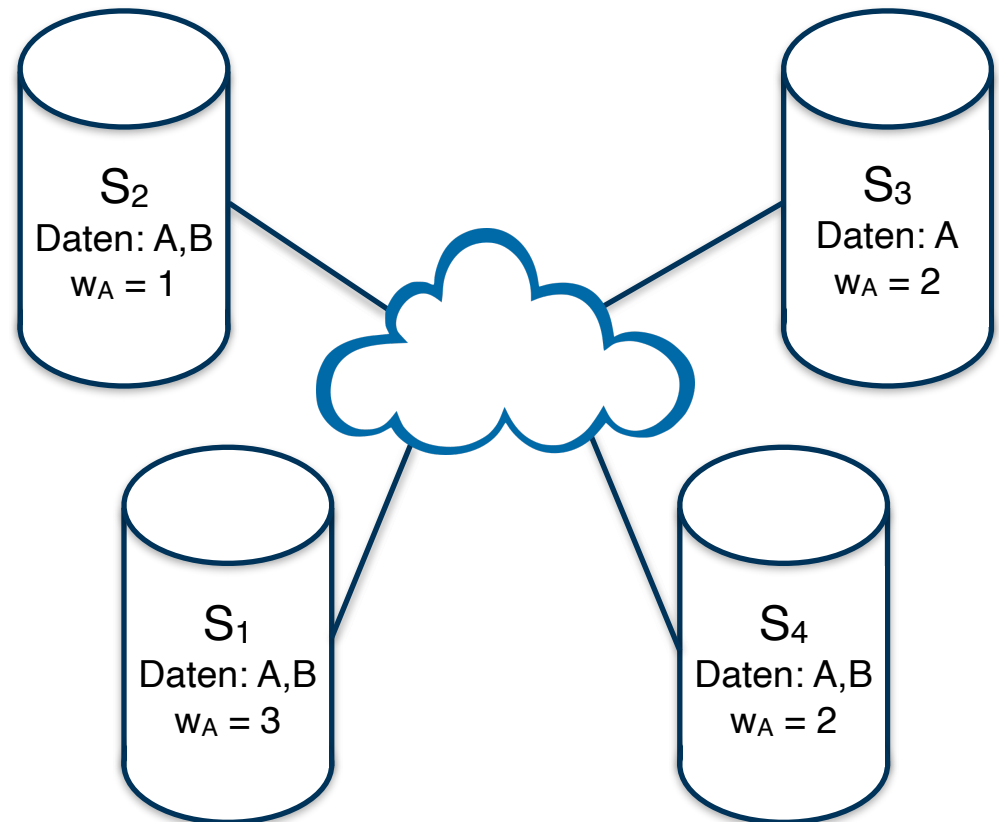
Quorum-Consensus Verfahren

Berechne das Schreib- Q_w und Lesequorum Q_r für A und B.

Für A:

1. Gesamtgewicht berechnen:

$$W(A) = 3+1+2+2 = 8$$



Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Berechne das Schreib- Q_w und Lesequorum Q_r für A und B.

Für A:

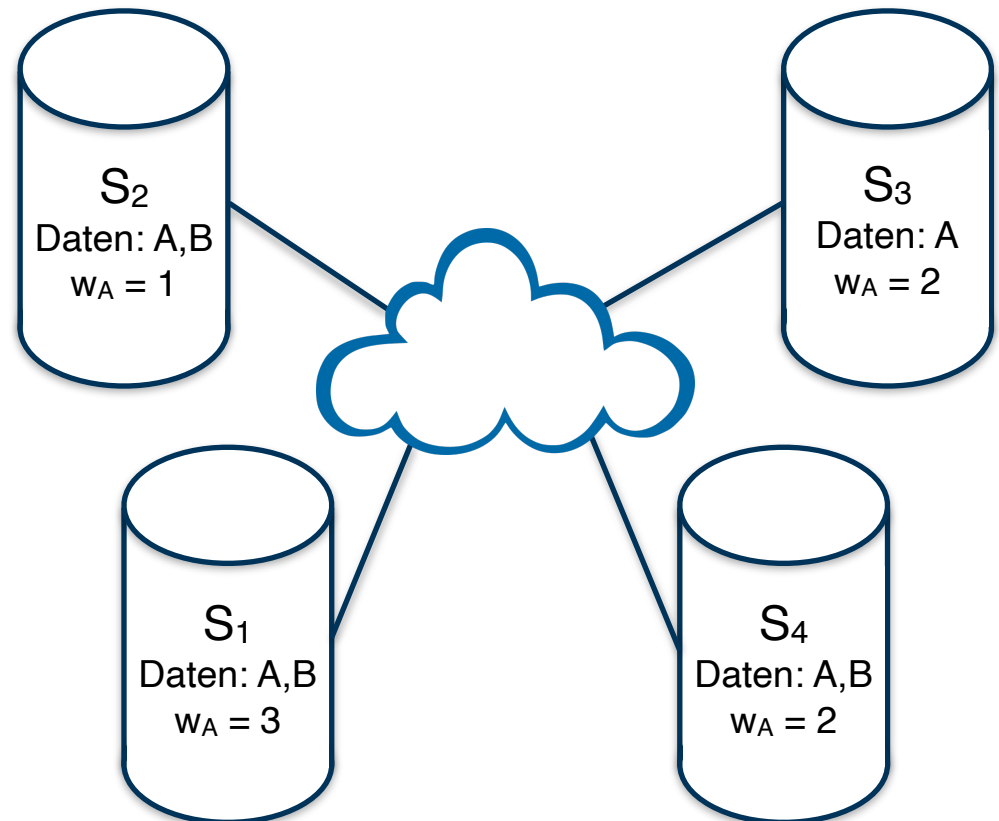
2. Schreibquorum berechnen

$$Q_w(A) + Q_w(A) > W(A)$$

$$\Rightarrow 2 * Q_w(A) > 8 \quad | :2$$

$$\Rightarrow Q_w(A) > 4$$

$$\Rightarrow Q_w(A) = 5$$



Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Berechne das Schreib- Q_w und Lesequorum Q_r für A und B.

Für A:

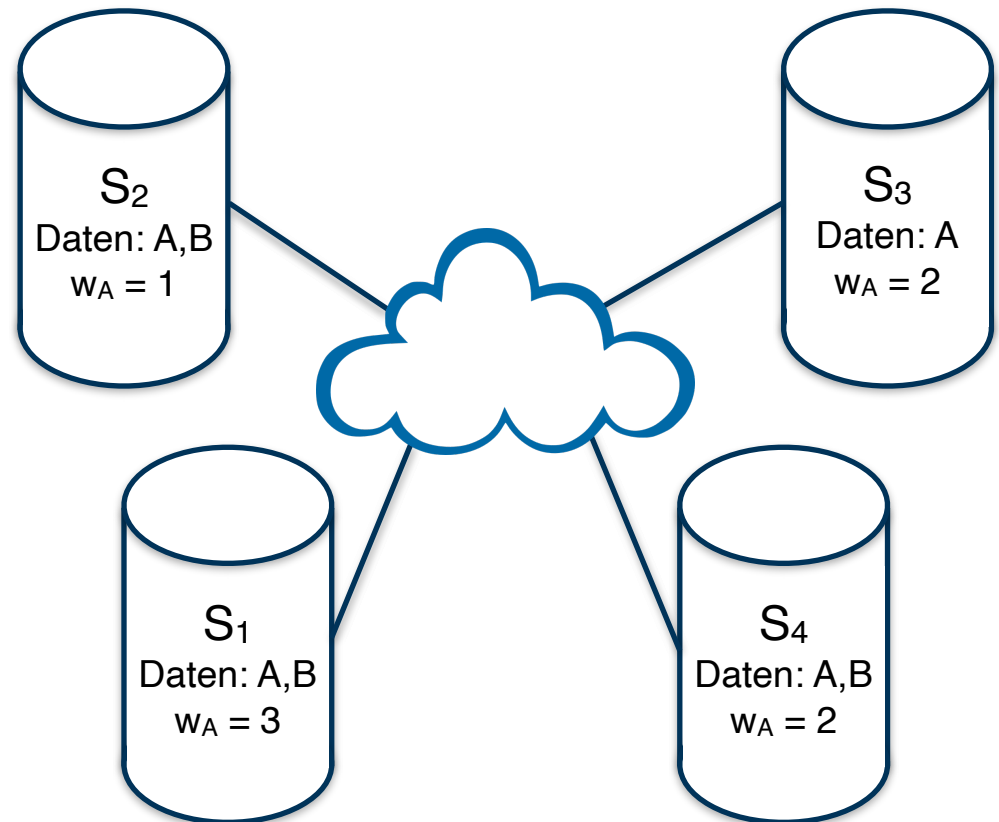
2. Lesequorum berechnen

$$Q_r(A) + Q_w(A) > W(A)$$

$$\Rightarrow Q_r(A) + 5 > 8 \quad | -5$$

$$\Rightarrow Q_r(A) > 3$$

$$\Rightarrow Q_r(A) = 4$$



Verteilte Datenbanksysteme

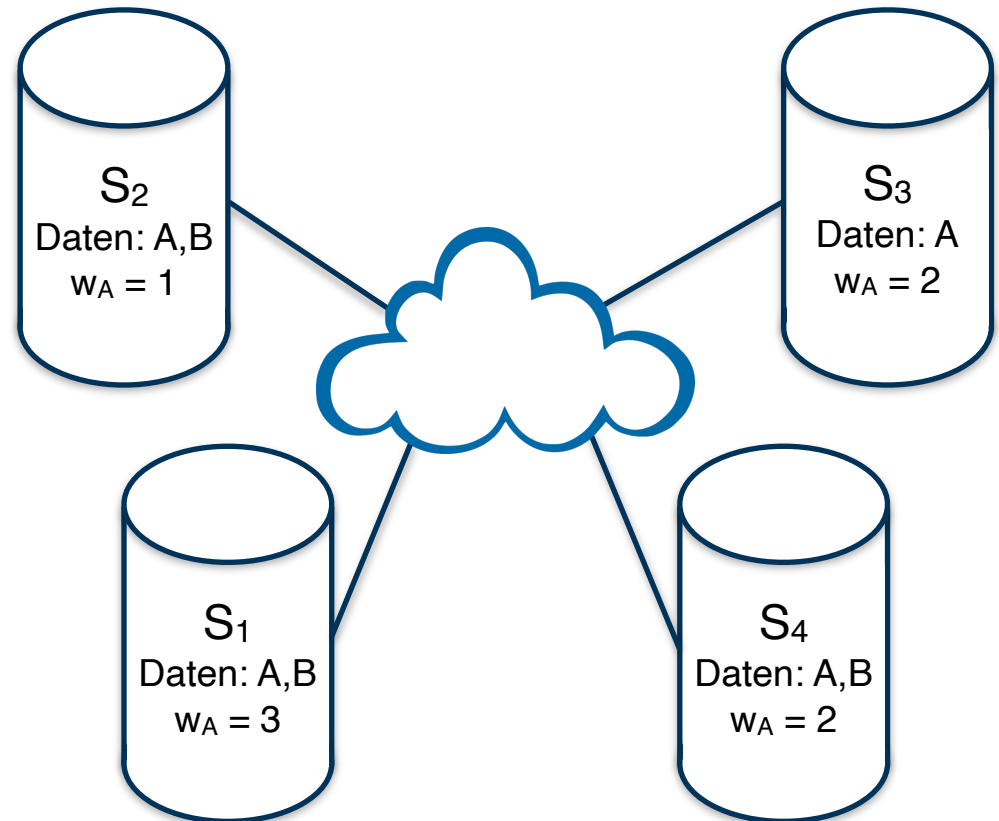
Quorum-Consensus Verfahren

Berechne das Schreib- Q_w und Lesequorum Q_r für A und B.

Für B:

1. Gesamtgewicht berechnen:

$$W(B) = 2+1+3 = 6$$



Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Berechne das Schreib- Q_w und Lesequorum Q_r für A und B.

Für B:

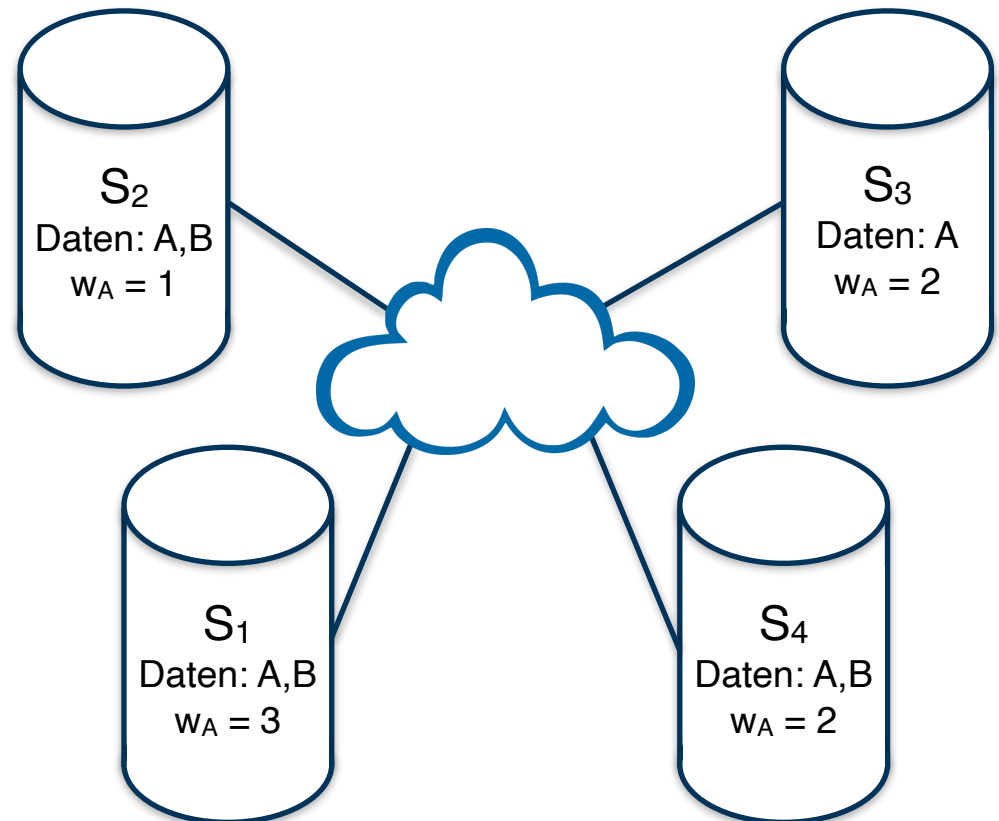
2. Schreibquorum berechnen

$$Q_w(B) + Q_w(B) > W(B)$$

$$\Rightarrow 2 * Q_w(B) > 6 \quad 1:2$$

$$\Rightarrow Q_w(B) > 3$$

$$\Rightarrow Q_w(B) = 4$$



Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Berechne das Schreib- Q_w und Lesequorum Q_r für A und B.

Für B:

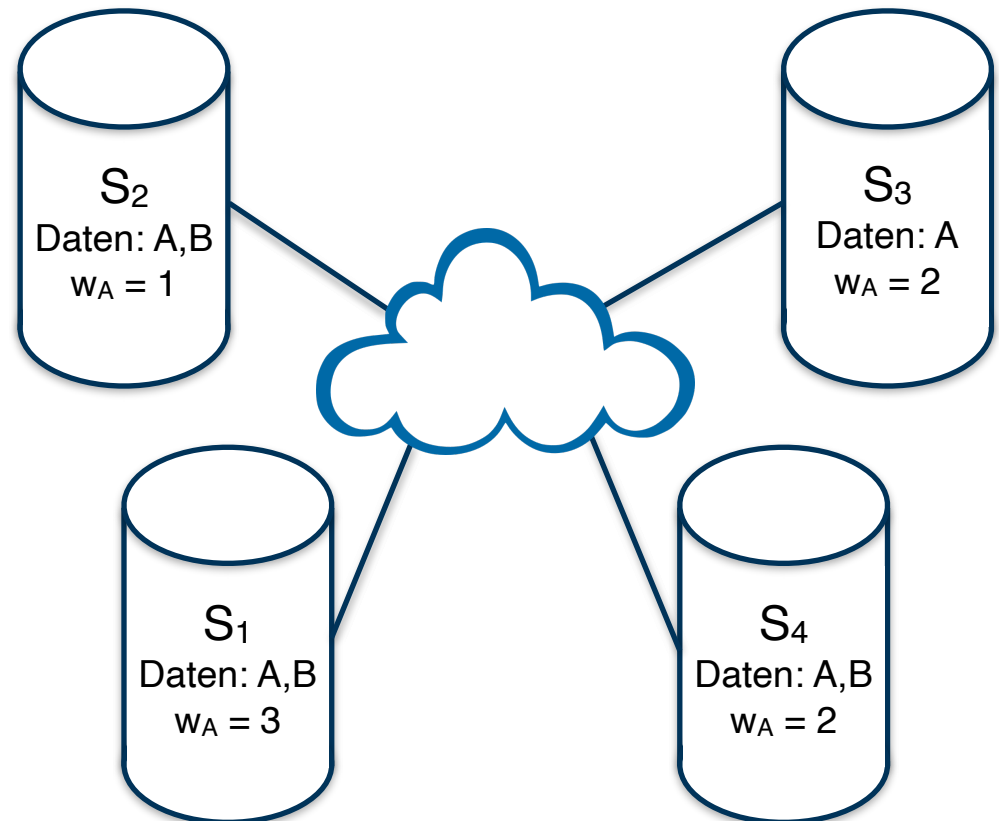
2. Lesequorum berechnen

$$Q_r(B) + Q_w(B) > W(B)$$

$$\Rightarrow Q_r(B) + 4 > 6 \quad | -4$$

$$\Rightarrow Q_r(B) > 2$$

$$\Rightarrow Q_r(B) = 3$$



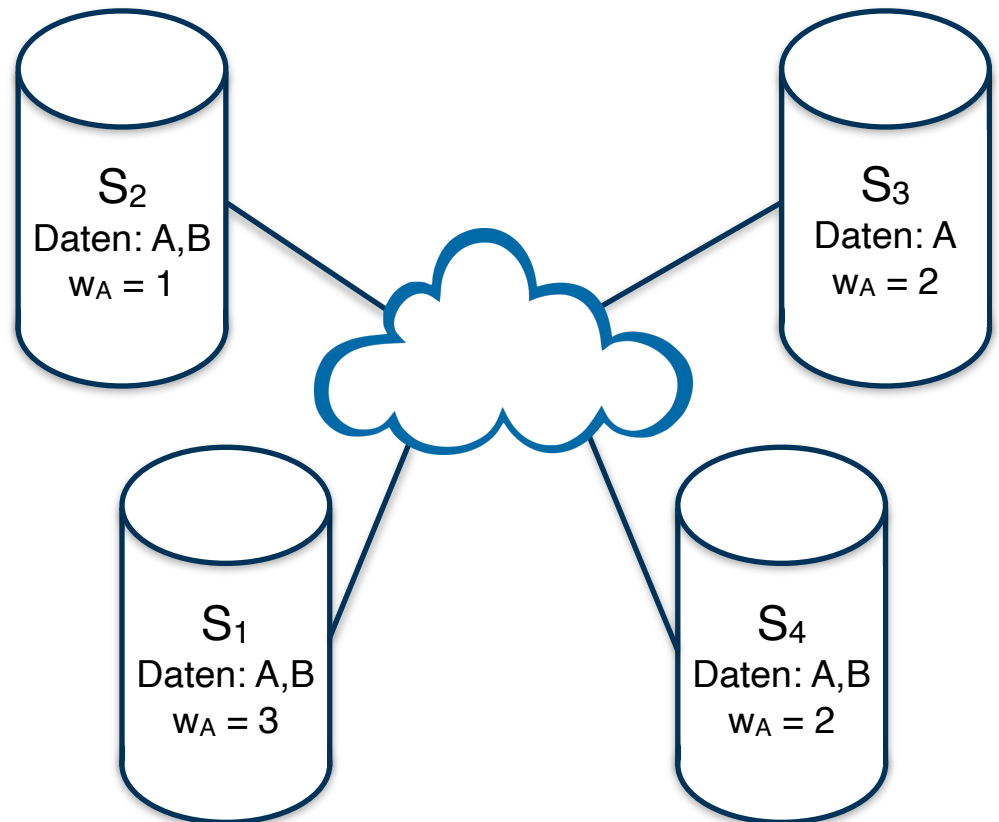
Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Schreibe:

- A- = 5
- B+ = 2

Station	Daten	Gewichte	Version
S ₁	A = 22 B = 7	w _A = 3 w _B = 2	A ₁ B ₁
S ₂	A = 22 B = 7	w _A = 1 w _B = 1	A ₁ B ₁
S ₃	A = 22	w _A = 2	A ₁
S ₄	A = 22 B = 7	w _A = 2 w _B = 3	A ₁ B ₁



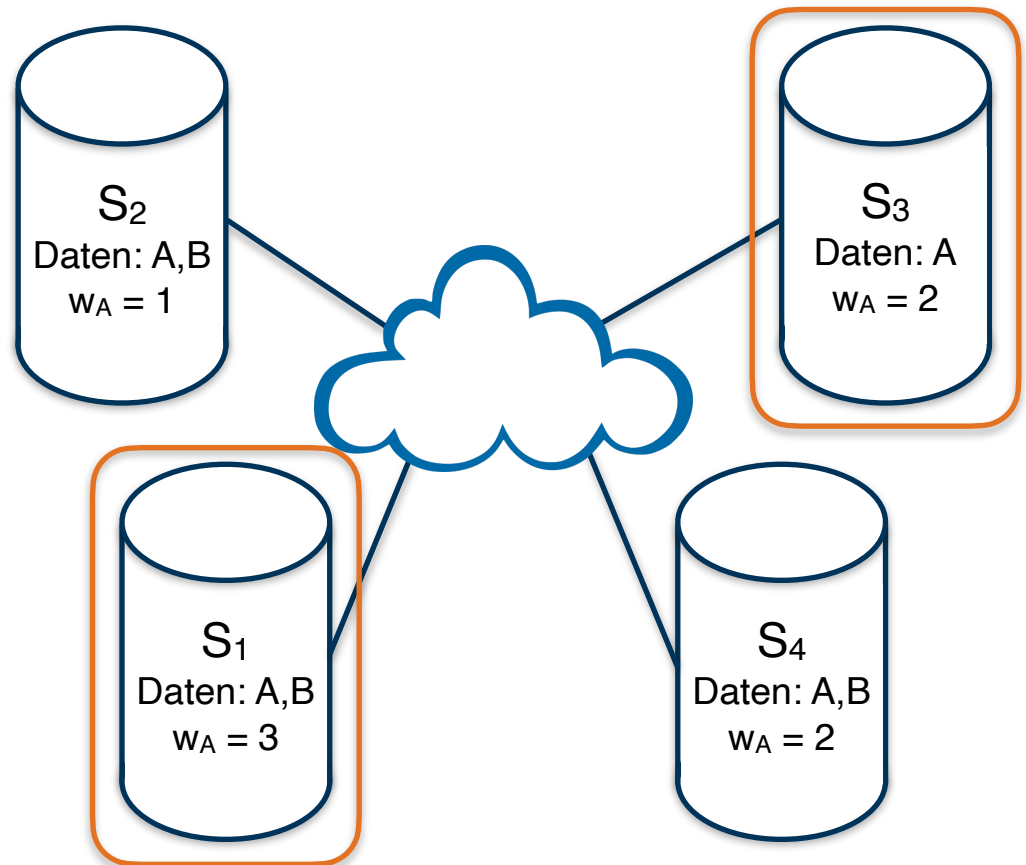
Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Schreibe:

- $A_{-}=5$
- $B_{+}=2$

Station	Daten	Gewichte	Version
S ₁	A =22 B = 7	w _A = 3 w _B = 2	A ₁ B ₁
S ₂	A =22 B = 7	w _A = 1 w _B = 1	A ₁ B ₁
S ₃	A =22	w _A = 2	A ₁
S ₄	A =22 B = 7	w _A = 2 w _B = 3	A ₁ B ₁



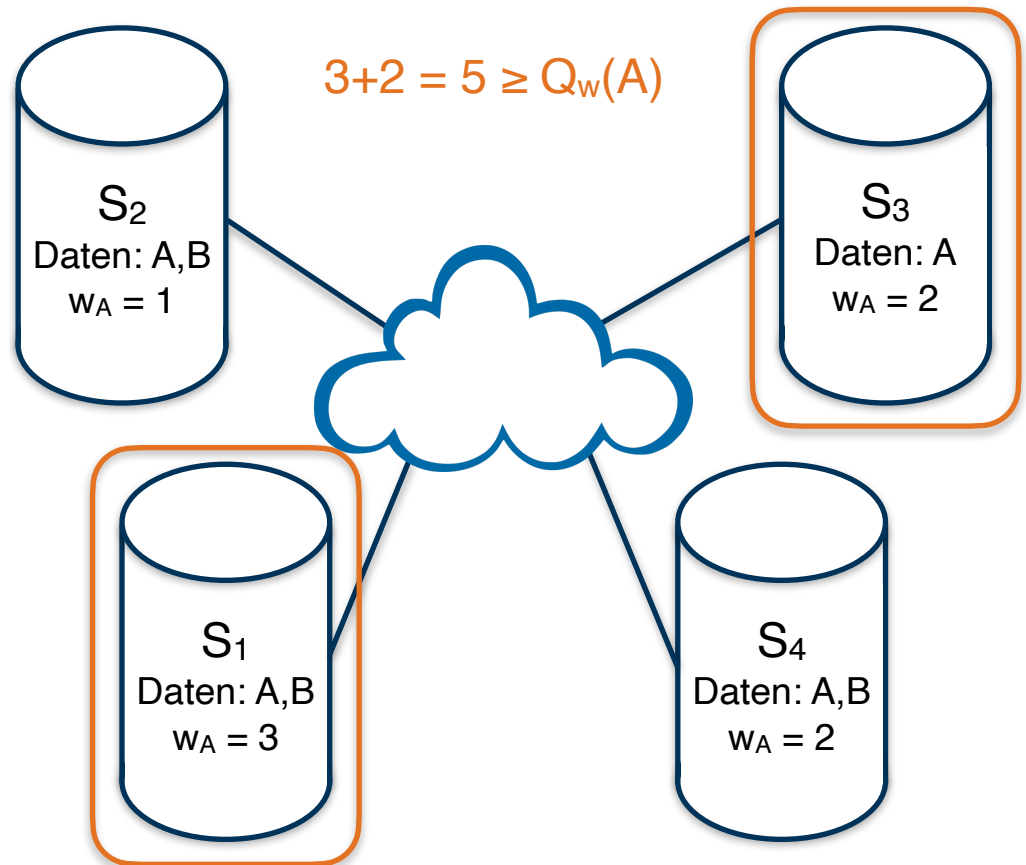
Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Schreibe:

- $A_{-}=5$
- $B_{+}=2$

Station	Daten	Gewichte	Version
S ₁	A =22 B = 7	w _A = 3 w _B = 2	A ₁ B ₁
S ₂	A =22 B = 7	w _A = 1 w _B = 1	A ₁ B ₁
S ₃	A =22	w _A = 2	A ₁
S ₄	A =22 B = 7	w _A = 2 w _B = 3	A ₁ B ₁



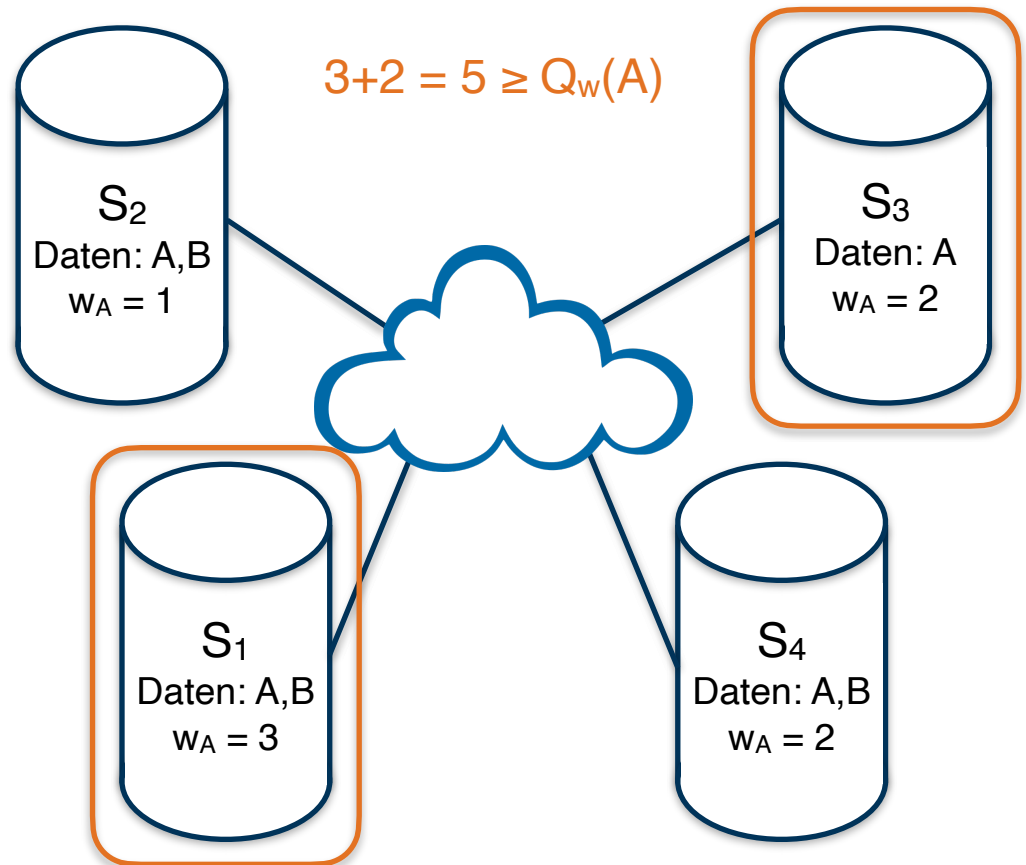
Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Schreibe:

- $A_{-}=5$
- $B_{+}=2$

Station	Daten	Gewichte	Version
S ₁	A = 17 B = 7	w _A = 3 w _B = 2	A ₂ B ₁
S ₂	A = 22 B = 7	w _A = 1 w _B = 1	A ₁ B ₁
S ₃	A = 17	w _A = 2	A ₂
S ₄	A = 22 B = 7	w _A = 2 w _B = 3	A ₁ B ₁



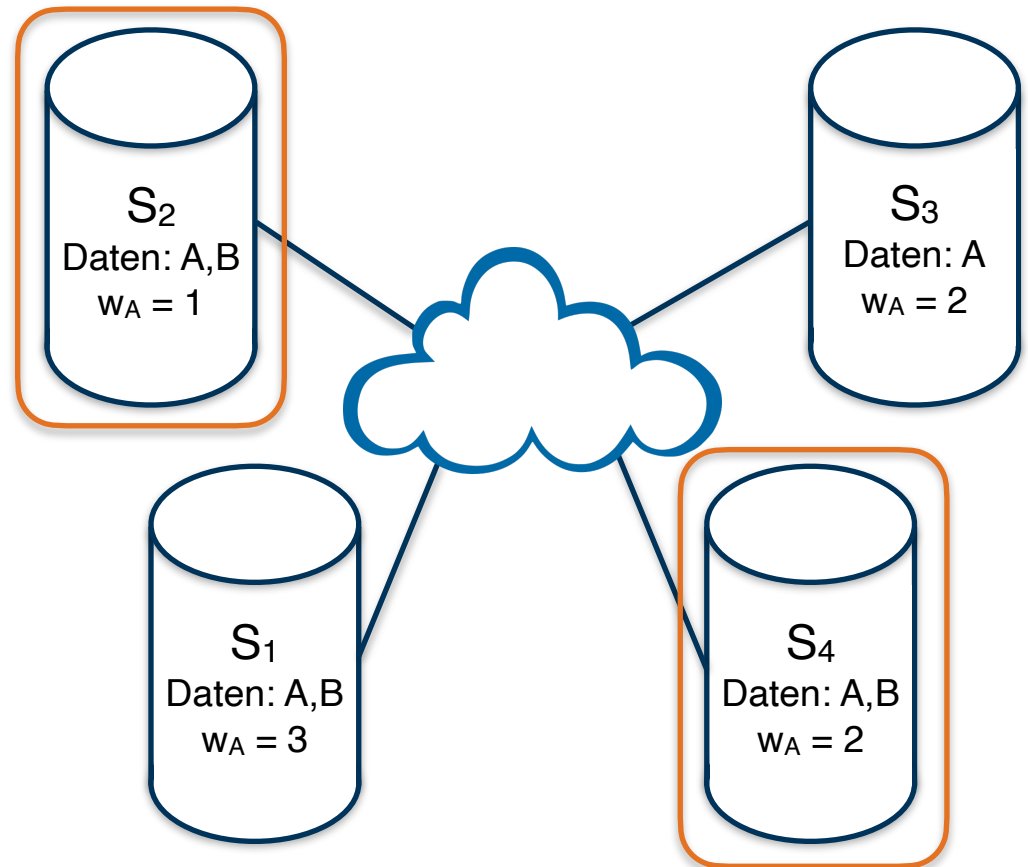
Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Schreibe:

- $A = 5$
- $B = 2$

Station	Daten	Gewichte	Version
S ₁	A = 17 B = 7	w _A = 3 w _B = 2	A ₂ B ₁
S ₂	A = 22 B = 7	w _A = 1 w _B = 1	A ₁ B ₁
S ₃	A = 17	w _A = 2	A ₂
S ₄	A = 22 B = 7	w _A = 2 w _B = 3	A ₁ B ₁



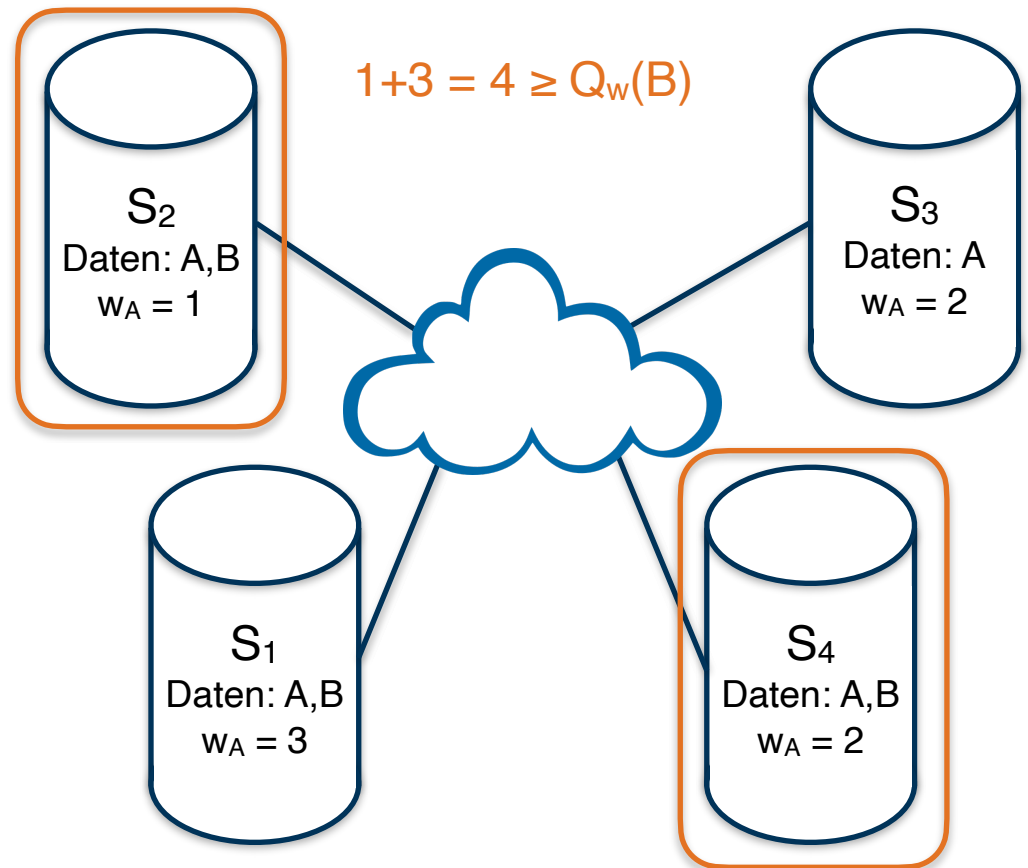
Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Schreibe:

- $A = 5$
- $B = 2$

Station	Daten	Gewichte	Version
S ₁	A = 17 B = 7	w _A = 3 w _B = 2	A ₂ B ₁
S ₂	A = 22 B = 7	w _A = 1 w _B = 1	A ₁ B ₁
S ₃	A = 17	w _A = 2	A ₂
S ₄	A = 22 B = 7	w _A = 2 w _B = 3	A ₁ B ₁



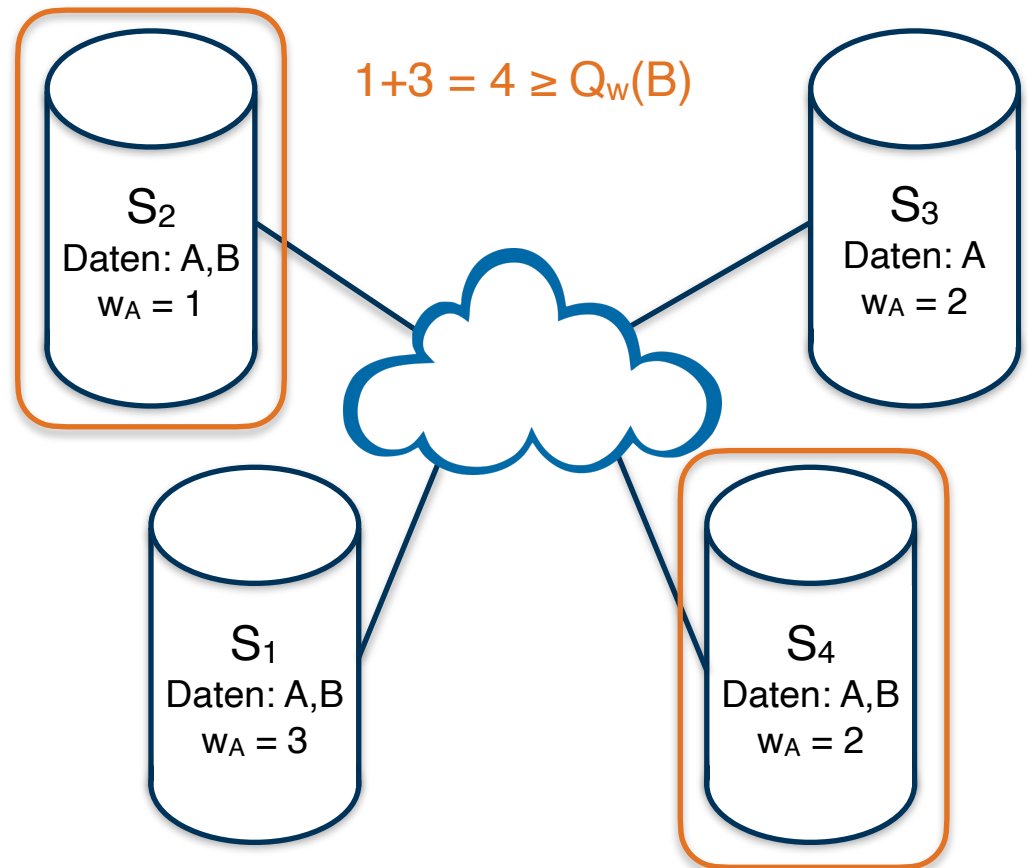
Verteilte Datenbanksysteme

Quorum-Consensus Verfahren

Schreibe:

- A = 5
- B = 2

Station	Daten	Gewichte	Version
S ₁	A = 17 B = 7	w _A = 3 w _B = 2	A ₂ B ₁
S ₂	A = 22 B = 9	w _A = 1 w _B = 1	A ₁ B ₂
S ₃	A = 17	w _A = 2	A ₂
S ₄	A = 22 B = 9	w _A = 2 w _B = 3	A ₁ B ₂





Aufgabe 1

Zeigen Sie, dass die *write-all/read-any* Methode zur Synchronisation replizierter Daten einen Spezialfall der *Quorum-Consensus*-Methode darstellt.

- Für welche Art von Workloads eignet sich dieses Verfahren besonders gut?
- Wie werden Stimmen zugeordnet um *write-all/read-any* zu simulieren?
- Wie müssen die Quoren Q_w und Q_r vergeben werden?



Aufgabe 2

Um Ausfallsicherheit zu garantieren ist ein Datenwert 'A' auf vier Rechnern verteilt. Jeder Rechner hält dabei eine vollständige Kopie von 'A'. Um Konsistenz zu garantieren wird das Quorum-Consensus-Verfahren eingesetzt. Dabei ist jedem Rechner ein Gewicht $w_i(A)$ wie folgt zugewiesen:

Rechner	Kopie	Gewicht
R_1	A_1	3
R_2	A_2	1
R_3	A_3	2
R_4	A_4	2

Das Lesequorum ist $Q_r(A) = 4$ und das Schreibquorum ist $Q_w(A) = 5$.

- Geben Sie **alle** Lesemöglichkeiten für eine Transaktion auf dem Datum 'A' nach dem Quorum-Consensus-Protokoll an.
- Geben Sie **alle** Schreibmöglichkeiten für eine Transaktion auf dem Datum 'A' nach dem Quorum-Consensus-Protokoll an.
- Zeigen Sie für dieses Beispiel, dass während eine Transaktion T_1 ein Schreibquorum auf A hält es für andere Transaktionen T_x nicht möglich ist ein Lesequorum für A zu bekommen.

Verteilte Datenbanksysteme

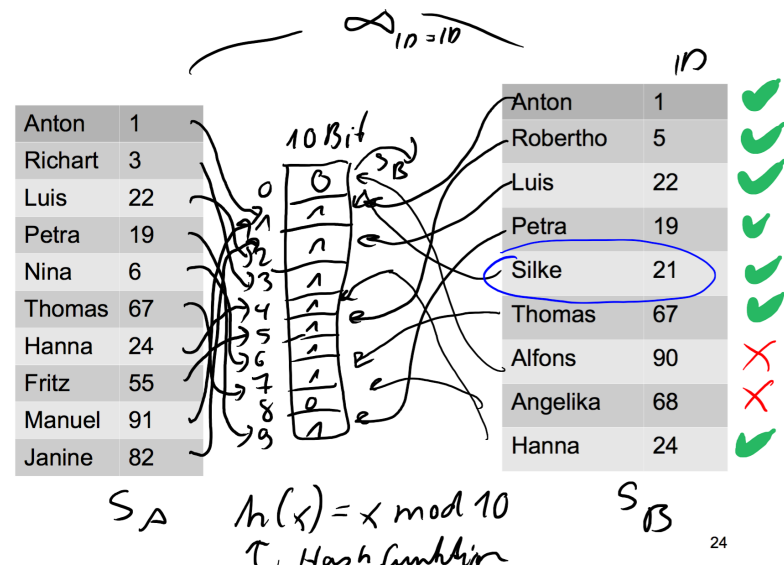
Bloom-Filter

- Einsatz bei sehr voluminösen Join-Attributen (z.B. lange Strings)

+ Verringerung der Transferkosten/
Netzwerkauslastung durch
Tupelvorauswahl mittels
Hashfunktion

+ Filter wird kompakter (Bitvektor V)

- Filterpräzision geht verloren





Verteilte Datenbanksysteme

Bloom-Filter

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1
2
3
4
5
6
7
8
9

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(x) = x \text{ mod } 10$$

Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5



V
0
1
2 1
3
4
5
6
7
8
9

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(2) = 2 \bmod 10 = 2$$

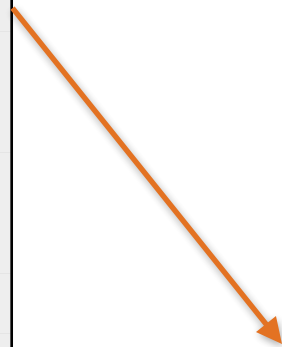
Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1
2
3
4
5
6
7
8
9



S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(7) = 7 \bmod 10 = 7$$

Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1
2 1
3
4
5
6 1
7 1
8
9

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(6) = 6 \bmod 10 = 6$$



Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1
2 1
3
4
5
6 1
7 1
8
9

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(2) = 2 \bmod 10 = 2$$

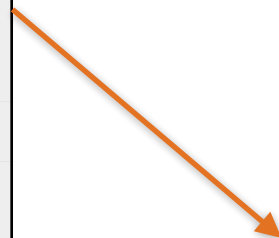
Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1
2 1
3
4
5
6 1
7 1
8 1
9



S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(8) = 8 \bmod 10 = 8$$

Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1 1
2 1
3
4
5
6 1
7 1
8 1
9

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(1) = 1 \bmod 10 = 1$$

Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1 1
2 1
3
4
5
6 1
7 1
8 1
9



S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(16) = 16 \bmod 10 = 6$$



Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1 1
2 1
3
4
5
6 1
7 1
8 1
9

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(8) = 8 \bmod 10 = 8$$



Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1 1
2 1
3
4
5
6 1
7 1
8 1
9

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(7) = 7 \bmod 10 = 7$$



Verteilte Datenbanksysteme

Bloom-Filter

1. Tabelle R mit $h(x)$ auf V mappen:

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0
1 1
2 1
3
4
5 1
6 1
7 1
8 1
9

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(5) = 5 \bmod 10 = 5$$



Verteilte Datenbanksysteme

Bloom-Filter

2. Felder in V ohne hash-Treffer mit 0 füllen

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0 0
1 1
2 1
3 0
4 0
5 1
6 1
7 1
8 1
9 0

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(x) = x \bmod 10$$



Verteilte Datenbanksysteme

Bloom-Filter

3. Bitvektor V an S schicken

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V	
0	0
1	1
2	1
3	0
4	0
5	1
6	1
7	1
8	1
9	0

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(x) = x \bmod 10$$



Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V	
0	0
1	1
2	1
3	0
4	0
5	1
6	1
7	1
8	1
9	0

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(x) = x \bmod 10$$



Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0 0
1 1
2 1
3 0
4 0
5 1
6 1
7 1
8 1
9 0



S	
Raum	Gebäude
1 ✓	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(1) = 1 \bmod 10 = 1$$



Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0 0
1 1
2 1
3 0
4 0
5 1
6 1
7 1
8 1
9 0



S	
Raum	Gebäude
1 ✓	IMETUM
2 ✓	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(2) = 2 \bmod 10 = 2$$

Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V	
0	0
1	1
2	1
3	0
4	0
5	1
6	1
7	1
8	1
9	0

S	
Raum	Gebäude
1	IMETUM
2	MI Büro
4	Physik
6	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie



$$h(4) = 4 \bmod 10 = 4$$



Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0 0
1 1
2 1
3 0
4 0
5 1
6 1
7 1
8 1
9 0

S	
Raum	Gebäude
1 ✓	IMETUM
2 ✓	MI Büro
4 ✗	Physik
6 ✓	MW
7	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(6) = 6 \bmod 10 = 6$$



Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0 0
1 1
2 1
3 0
4 0
5 1
6 1
7 1
8 1
9 0

S	
Raum	Gebäude
1 ✓	IMETUM
2 ✓	MI Büro
4 ✗	Physik
6 ✓	MW
7 ✓	MI Raum
8	ERI
9	MI Bib
10	Physik
11	Chemie

$$h(7) = 7 \bmod 10 = 7$$

Verteilte Datenbanksysteme

Bloom-Filter

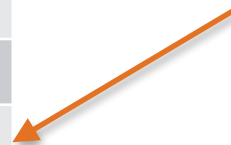
4. S überprüft mit $h(x)$ den Bitvektor V

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V	
0	0
1	1
2	1
3	0
4	0
5	1
6	1
7	1
8	1
9	0

S		
Raum		Gebäude
1	✓	IMETUM
2	✓	MI Büro
4	✗	Physik
6	✓	MW
7	✓	MI Raum
8	✓	ERI
9		MI Bib
10		Physik
11		Chemie



$$h(8) = 8 \bmod 10 = 8$$



Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0 0
1 1
2 1
3 0
4 0
5 1
6 1
7 1
8 1
9 0

S		
Raum		Gebäude
1	✓	IMETUM
2	✓	MI Büro
4	✗	Physik
6	✓	MW
7	✓	MI Raum
8	✓	ERI
9	✗	MI Bib
10		Physik
11		Chemie

$$h(9) = 9 \bmod 10 = 9$$

Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V	
0	0
1	1
2	1
3	0
4	0
5	1
6	1
7	1
8	1
9	0

S		
Raum		Gebäude
1	✓	IMETUM
2	✓	MI Büro
4	✗	Physik
6	✓	MW
7	✓	MI Raum
8	✓	ERI
9	✗	MI Bib
10	✗	Physik
11		Chemie

$$h(10) = 10 \bmod 10 = 0$$



Verteilte Datenbanksysteme

Bloom-Filter

4. S überprüft mit $h(x)$ den Bitvektor V

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V
0 0
1 1
2 1
3 0
4 0
5 1
6 1
7 1
8 1
9 0

S		
Raum		Gebäude
1	✓	IMETUM
2	✓	MI Büro
4	✗	Physik
6	✓	MW
7	✓	MI Raum
8	✓	ERI
9	✗	MI Bib
10	✗	Physik
11	✓	Chemie

False positive

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

$$h(11) = 11 \bmod 10 = 1$$



Verteilte Datenbanksysteme

Bloom-Filter

5. Übermitteln der Treffer zur Station R

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V	
0	0
1	1
2	1
3	0
4	0
5	1
6	1
7	1
8	1
9	0

S		
Raum		Gebäude
1	✓	IMETUM
2	✓	MI Büro
4	✗	Physik
6	✓	MW
7	✓	MI Raum
8	✓	ERI
9	✗	MI Bib
10	✗	Physik
11	✓	Chemie

$$h(x) = x \bmod 10$$



Verteilte Datenbanksysteme

Bloom-Filter

False positives werden übermittelt und von R beim Join verworfen.

False positive Rate
1/6

- ✓ Tupel wird zur Station mit R geschickt
- ✗ Tupel wird nicht übermittelt

R	
Pers	Raum
Max	2
Magda	7
Tom	6
Alex	2
Julius	8
Kathi	1
Anna	16
Gregor	8
Thuy	7
Domi	5

V	
0	0
1	1
2	1
3	0
4	0
5	1
6	1
7	1
8	1
9	0

S		
Raum		Gebäude
1	✓	IMETUM
2	✓	MI Büro
4	✗	Physik
6	✓	MW
7	✓	MI Raum
8	✓	ERI
9	✗	MI Bib
10	✗	Physik
11	✓	Chemie

$$h(x) = x \bmod 10$$



Aufgabe 3

Gegeben seien die Tabellen **Studenten** und **Punkte** mit Schlüssel **MatrNr**, wobei **Punkte** auf einem separaten Rechner gespeichert ist. Es soll folgende Anfrage ausgeführt werden:

```
SELECT Name, Bonus FROM Student s, Punkte p WHERE s.MatrNr = p.MatrNr;
```

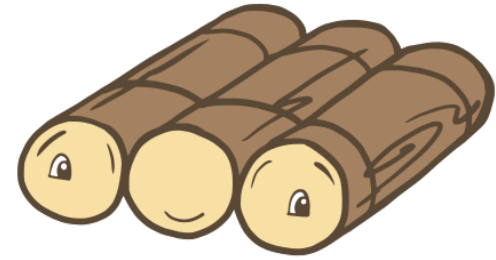
Der Datenbankadministrator entscheidet sich für einen Bloom-Filter zur Vorauswahl der Tupel. Auf **MatrNr** wird die Hash-Funktion $h(x) = x \bmod 5$ angewendet.

Studenten			Punkte		
<u>MatrNr</u>	Name	Hashwert	<u>MatrNr</u>	Bonus	Hashwert
27	Magda		27	ja	
4	Josef		16	nein	
19	Erik		25	nein	
95	Philipp		95	ja	

- Berechnen Sie die Hash-Werte und tragen Sie diese in die obige Tabelle ein.
- Füllen Sie den von **Studenten** zu übertragenden Bitvektor aus. Verwenden Sie 0 oder 1.
- Geben Sie basierend auf dem Bitvektor an, welche Tupel aus **Punkte** übertragen werden (nur **MatrNr** angeben).
- Geben Sie die Falsch-Positiv-Rate (false positive rate) an.
- Nehmen Sie an, dass jedes Tupel 8 Byte und der Bloomfilter selbst 1 Byte groß ist. Berechnen Sie zunächst die übertragenen Bytes ohne und mit Einsatz des Bloom-Filters.

Verteilte Datenbanksysteme

Raft Konsensalgorithmus

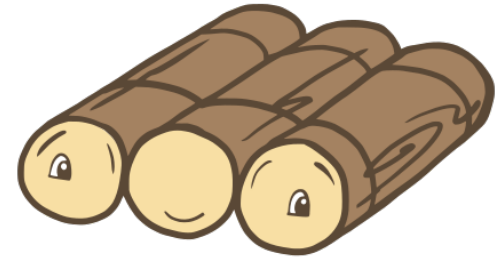


- Server innerhalb eines Clusters können **Follower**, **Candidate** oder **Leader** sein
- Zu Beginn sind alle Knoten Follower
- Es gibt immer nur ein Leader
 - Leader sendet „**Heartbeats**“ zu den Follower-Servern, damit diese wissen, dass der Leader noch existiert und nicht abgestürzt ist
 - Write & Read requests von außerhalb laufen über den Leader
- Jeder Server schreibt Log Einträge („**log replication**“)
 - Leader triggered die Log updates aller Server
- Zur Bestimmung des Leaders wird mit **randomisierten Timeouts** gearbeitet. Die Knoten die zuerst einen Timeout erreichen, werden zu Candidates. Sollte die **Mehrheit** der Server im Cluster für einen bestimmten Candidate stimmen, wird dieser der neue Leader



Verteilte Datenbanksysteme

Raft Konsensalgorithmus



- <http://thesecretlivesofdata.com/raft/>



Aufgabe 4

Um ein Gefühl für das *Raft Consensus Protokoll* zu bekommen, führen Sie folgende Aktionen mit *RaftScope* unter <https://raft.github.io/> aus. Die Simulation kann mit einem Klick auf das Uhr Symbol gestoppt und gestartet werden. Ein Rechtsklick auf einen der Server öffnet ein Menü um Aktionen auszulösen.

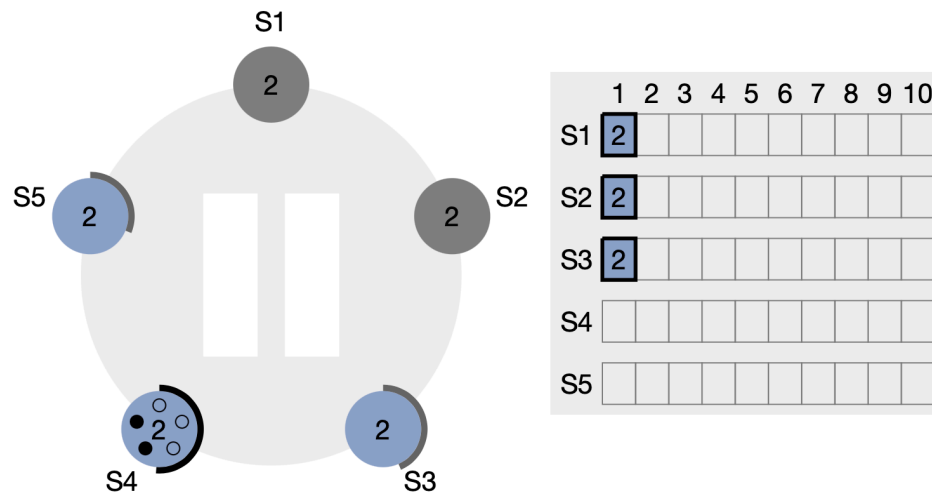
Führen Sie folgende Aktion aus und beschreiben Sie in Stichpunkten was passiert.

1. Warten sie bis die erste Leader Election abgeschlossen ist.
2. Senden Sie einen Request an den Leader. (Rechtsklick auf Leader)
3. Stoppen Sie den Leader (Server ausschalten), warten Sie bis ein neuer Leader gewählt wurde, und senden Sie einen Request an den neuen Leader.
4. Wiederholen Sie den vorigen Schritt 2x bis nur noch 2 Knoten übrig sind. Wird ein neuer Leader gewählt?
5. Starten Sie (resume) wieder einen weiteren Server. Wird ein neuer Leader gewählt?

Aufgabe 5

Beantworten Sie folgende Fragen zum RAFT Protokoll. Verwenden Sie *RaftScope* unter <https://raft.github.io/> um Ihre Vermutungen zu bestätigen.

1. Wie viele Server müssen ausfallen, dass in einem Cluster mit n Servern kein neuer Leader bestimmt werden kann?
2. Wie können Sie mit `restart` und `time-out` in RaftScope einen bestimmten Knoten als neuen Leader erzwingen? Geben Sie die Schritte an.
3. Wie verläuft das Beispiel in Abbildung 1 weiter? Wie kann sicher gestellt werden, dass der neu gewählte Leader den neuesten Logeintrag hält? Beschreiben Sie in Stichpunkten.





Fragen?