Einsatz und Realisierung von Datenbanksystemen

ERDB Übungsleitung

Maximilian {Bandle, Schüle}, Josef Schmeißer

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.

Big Data

Big Data
Term Frequency - Inverse Document Frequency

Anwendung im *Information Retrieval*
- Finde zu einer Suchanfrage die relevantesten Dokumente
- Große Herausforderung aufgrund der Menge an Web-Dokumenten

Term Frequency - Inverse Document Frequency (TF-IDF)
- Dokument-Ranking basierend auf Begriffshäufigkeiten
- Vollautomatische Analyse
- Meist wird nur ein *Vokabular* berücksichtigt, nicht alle Worte

Big Data
Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$TF_{ij} = \frac{f_{ij}}{\sum_{i=1\ldots|V|} f_{ij}}$$

#Wort i im Dokument j

#Worte im Dokument j

- Gewicht eines Begriffs in einem kurzen Dokument höher als in einem langen Dokument
- Normalisierung

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$TF_{ij} = \frac{f_{ij}}{\sum_{i=1\ldots|V|} f_{ij}}$$

#Wort i im Dokument j

#Worte im Dokument j

	TF		
Wort i	D_1	D_2	D_3
ERDB			
Klausur			
Erfolg			

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$TF_{ij} = \frac{f_{ij}}{\sum_{i=1\ldots|V|} f_{ij}}$$

#Wort i im Dokument j

#Worte im Dokument j

	TF		
Wort i	D_1	D_2	D_3
ERDB	1/5		
Klausur	0/5		
Erfolg	0/5		

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$TF_{ij} = \frac{f_{ij}}{\sum_{i=1\ldots|V|} f_{ij}}$$

#Wort i im Dokument j

#Worte im Dokument j

	TF		
Wort i	D_1	D_2	D_3
ERDB	1/5	0/5	
Klausur	0/5	1/5	
Erfolg	0/5	0/5	

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$TF_{ij} = \frac{f_{ij}}{\sum_{i=1\ldots|V|} f_{ij}}$$

#Wort i im Dokument j

#Worte im Dokument j

	TF		
Wort i	D_1	D_2	D_3
ERDB	1/5	0/5	1/10
Klausur	0/5	1/5	1/10
Erfolg	0/5	0/5	1/10

Big Data
Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$IDF_i = log(N/n_i)$$

#Dokumente

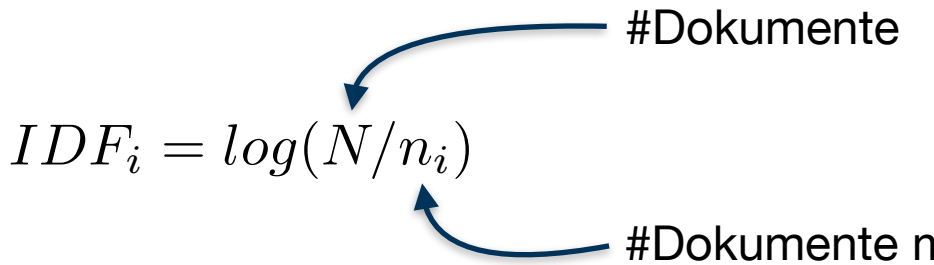
#Dokumente mit Wort i

- Gewichtung für jeden Begriff
- Seltene Begriffe bekommen eine höhere Gewichtung als Häufige

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur



	IDF			
Wort i	**N**	**n_i**	**N/n_i**	**$log(N/n_i)$**
ERDB				
Klausur				
Erfolg				

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$IDF_i = log(N/n_i)$$

#Dokumente

#Dokumente mit Wort i

	IDF			
Wort i	**N**	**n_i**	**N/n_i**	**$log(N/n_i)$**
ERDB	3			
Klausur	3			
Erfolg	3			

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$IDF_i = log(N/n_i)$$

#Dokumente

#Dokumente mit Wort i

	IDF			
Wort i	**N**	**n_i**	**N/n_i**	**$log(N/n_i)$**
ERDB	3	2		
Klausur	3	2		
Erfolg	3	1		

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

#Dokumente

$$IDF_i = log(N/n_i)$$

#Dokumente mit Wort i

	IDF			
Wort i	**N**	**n_i**	**N/n_i**	**$log(N/n_i)$**
ERDB	3	2	3/2	
Klausur	3	2	3/2	
Erfolg	3	1	3/1	

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

$$IDF_i = log(N/n_i)$$

#Dokumente

#Dokumente mit Wort i

	IDF			
Wort i	**N**	**n_i**	**N/n_i**	**$log(N/n_i)$**
ERDB	3	2	3/2	0,176
Klausur	3	2	3/2	0,176
Erfolg	3	1	3/1	0,477

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

Relevanz von Dokument j

$$rel(D_j, Q) = \sum_{i \in Q} TF_{ij} \times IDF_i$$

Anfrage Q

Für alle Wörter von Q

Big Data
Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

Relevanz von Dokument j

$$rel(D_j, Q) = \sum_{i \in Q} TF_{ij} \times IDF_i$$

Anfrage Q

Für alle Wörter von Q

rel(D_1,{ERDB,Klausur,Erfolg}) =

$TF_{ERDB,1}$ * IDF_{ERDB} + $TF_{Klausur,1}$ * $IDF_{Klausur}$ + $TF_{Erfolg,1}$ * IDF_{Erfolg}

Big Data

Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

Relevanz von Dokument j

$$rel(D_j, Q) = \sum_{i \in Q} TF_{ij} \times IDF_i$$

Anfrage Q

Für alle Wörter von Q

rel(D_1,{ERDB,Klausur,Erfolg}) =

$TF_{ERDB,1}$ * IDF_{ERDB} + $TF_{Klausur,1}$ * $IDF_{Klausur}$ + $TF_{Erfolg,1}$ * IDF_{Erfolg}

rel(D_1,{ERDB,Klausur,Erfolg}) =

1/5 * 0,176 + 0 * 0,176 + 0 * 0,477

Big Data
Term Frequency - Inverse Document Frequency

D_1	D_2	D_3
ERDB macht echt viel Spaß	Die Klausur ist sicher machbar	Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

Relevanz von Dokument j

$$rel(D_j, Q) = \sum_{i \in Q} TF_{ij} \times IDF_i$$

Anfrage Q

Für alle Wörter von Q

rel(D_1,{ERDB,Klausur,Erfolg}) =

$TF_{ERDB,1}$ * IDF_{ERDB} + $TF_{Klausur,1}$ * $IDF_{Klausur}$ + $TF_{Erfolg,1}$ * IDF_{Erfolg}

rel(D_1,{ERDB,Klausur,Erfolg}) =

1/5 * 0,176 + 0 * 0,176 + 0 * 0,477 = **0,0352**

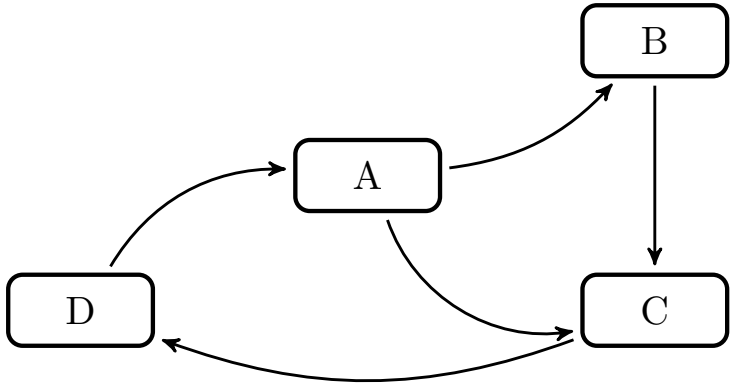
Aufgabe 1

Berechnen Sie für folgende drei Dokumente die TF-IDF-Werte:

1. „Beim Fußball dauert ein Spiel neunzig Minuten – und am Ende gewinnen die Deutschen“
2. „Beim Fußball muss das Runde (der Ball) in das Eckige (das Tor)“
3. „Nie war ein Tor so wertvoll wie jetzt“

Welches Ranking ergibt sich gemäß der Relevanzwerte für die Anfrage: „Fußball“ $\wedge$ „Tor“. Zur Ermittlung des TF Wertes gehen sie davon aus, dass alle Wörter eines Dokuments *interessant* sind?

Aufgabe 2



In Abbildung 1 gezeigte Netzwerk von Web-Seiten wird ein kleines Beispiel für einen Webgraphen gezeigt. Lösen sie folgende Aufgaben.

1. Berechnen Sie, für das in Abbildung, den PageRank, sowie die HITS-Werte nach 2 Iterationen. Nutzen Sie $1/|V|$ als Anfangswert für den PageRank und 1 für HITS. $a = 0.1$
2. Formulieren sie eine Iteration des Pagerank Algorithmus in SQL. Der Graph ist dabei in der Tabelle *edges(From, To)* gespeichert, die aktuelle PageRank Gewichtung in der Tabelle *pagerank(Vertex, Weight.* Sie können die Anzahl der Knoten als Konstante annehmen, z.B. 1000.

HITS Algorithmus

Hypertext **I**nduced **T**opic **S**election

Automatische Relevanz-Beurteilung für Websites

Vernetzung als Kriterium

Zwei Rollen:

- Hub (Knotenpunkt)
- Autorität (Website mit Inhalt)

➡ Alle Seiten werden in beiden Rollen beurteilt

Hub

- Wertvoller auf je mehr höherwertige Autoritäten er verweist (ausgehende Kanten)

Autorität

- Wertvoller je mehr höherwertige Hubs auf sie verweisen (eingehende Kanten)

HITS Algorithmus

Hypertext **I**nduced **T**opic **S**election

Iteration:

1. Berechne alle Hub-Werte

Summe der Gewichte der Knoten aller ausgehenden Kanten

$$h_i = \sum_{j=1\ldots N} A_{ij} a_j$$

2. Berechne alle Autoritäts-Werte

Summe der Gewichte der Knoten aller eingehenden Kanten

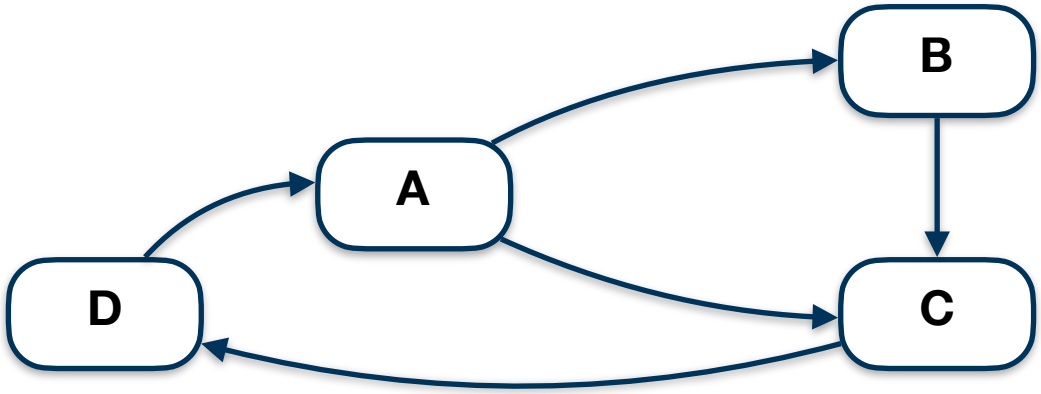
$$a_i = \sum_{j=1\ldots N} A_{ji} h_j$$

3. Normalisiere die Autoritäts-Werte mit

$$\lambda = \frac{1}{max(a)}$$

Aufgabe 2

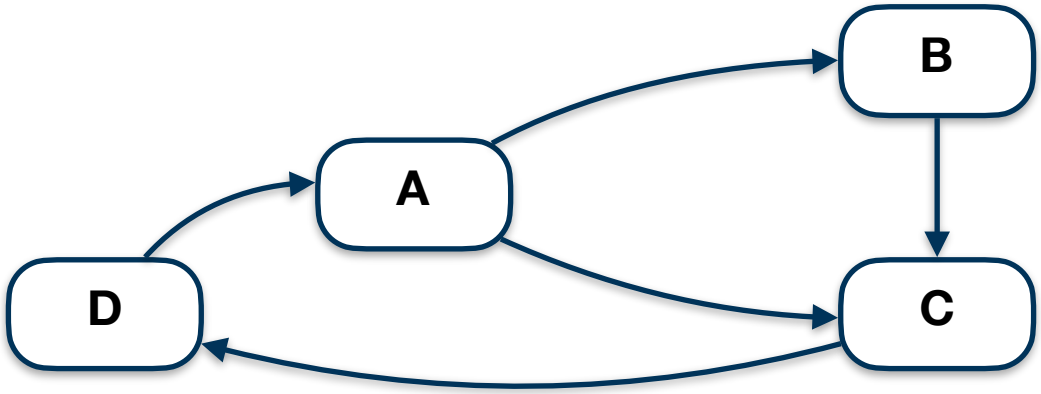
HITS Algorithmus



Berechne für den obigen Graphen die HITS-Werte nach 2 Iterationen.
Nutze 1 als Startwert für HITS.

Aufgabe 2

HITS Algorithmus



1. Iteration

	A	B	C	D
Hubs	2	1	1	1
Vorläufige Autoritäten				
Normalisierte Autoritäten				

$h_A = a_B + a_C = 1 + 1$

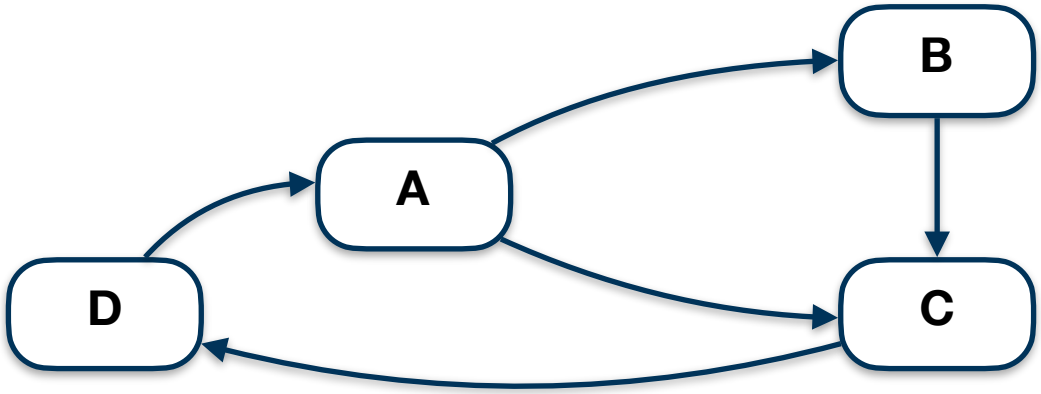
$h_B = a_C = 1$

$h_C = a_D = 1$

$h_D = a_A = 1$

Aufgabe 2

HITS Algorithmus



1. Iteration

	A	B	C	D
Hubs	2	1	1	1
Vorläufige Autoritäten	1	2	3	1
Normalisierte Autoritäten				

$a_A = h_D = 1$

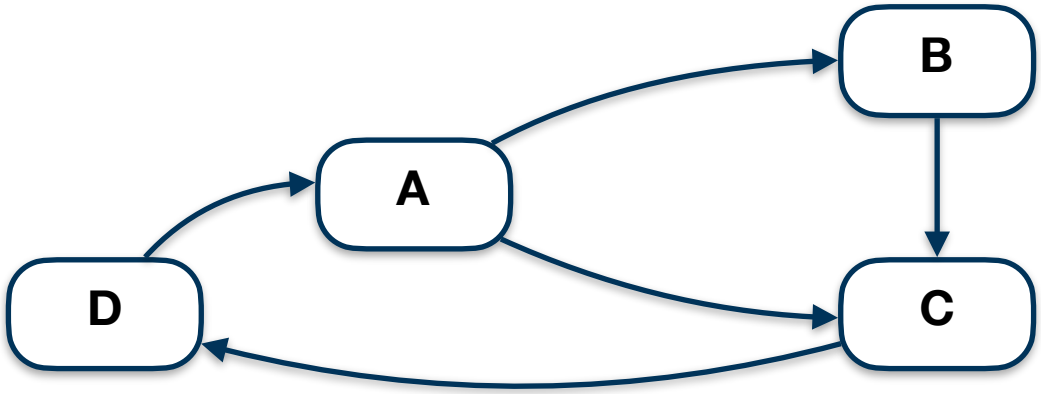
$a_B = h_A = 2$

$a_C = h_A + h_B = 2 + 1$

$a_D = h_C = 1$

Aufgabe 2

HITS Algorithmus



1. Iteration

	A	B	C	D
Hubs	2	1	1	1
Vorläufige Autoritäten	1	2	3	1
Normalisierte Autoritäten	1/3	2/3	1	1/3

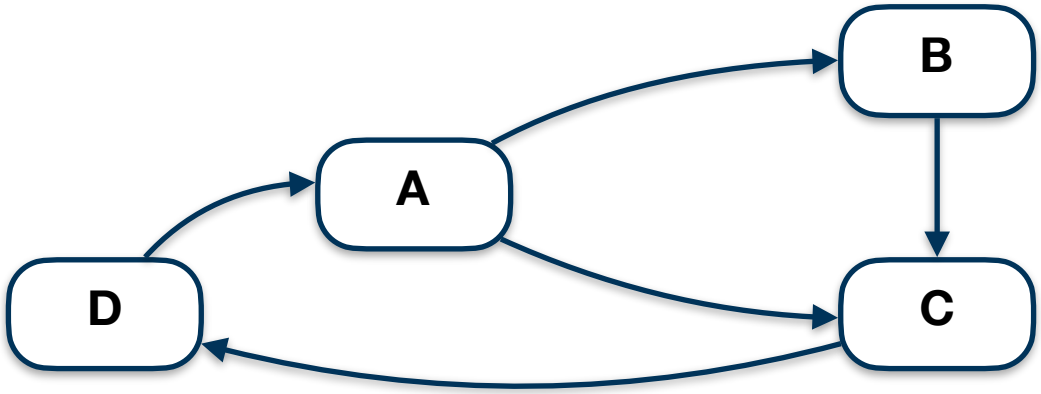
Normalisieren:

max(a) = 3

=> a_i*1/3

Aufgabe 2

HITS Algorithmus



1. Iteration

	A	B	C	D
Hubs	2	1	1	1
Vorläufige Autoritäten	1	2	3	1
Normalisierte Autoritäten	1/3	2/3	1	1/3

2. Iteration

	A	B	C	D
Hubs	5/3	1	1/3	1/3
Vorläufige Autoritäten				
Normalisierte Autoritäten				

$h_A = a_B + a_C = 2/3 + 1$

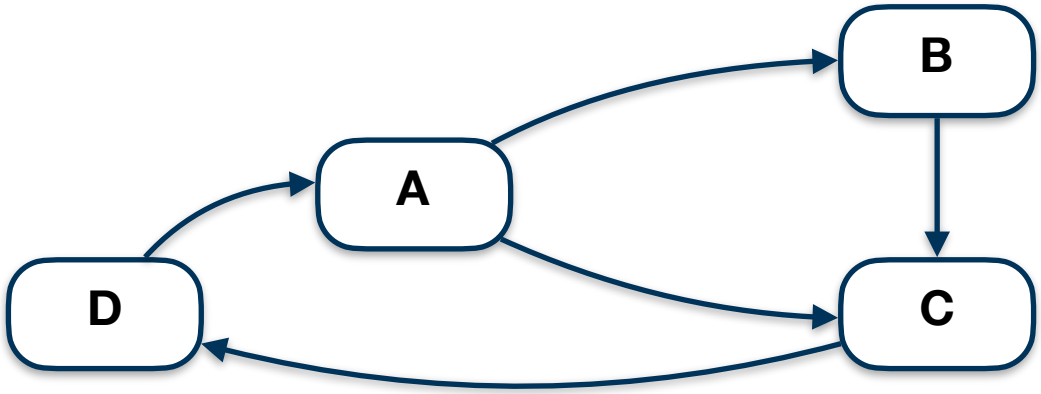
$h_B = a_C = 1$

$h_C = a_D = 1/3$

$h_D = a_A = 1/3$

Aufgabe 2

HITS Algorithmus



1. Iteration

	A	B	C	D
Hubs	2	1	1	1
Vorläufige Autoritäten	1	2	3	1
Normalisierte Autoritäten	1/3	2/3	1	1/3

2. Iteration

	A	B	C	D
Hubs	5/3	1	1/3	1/3
Vorläufige Autoritäten	1/3	5/3	8/3	1/3
Normalisierte Autoritäten				

$a_A = h_D = 1/3$

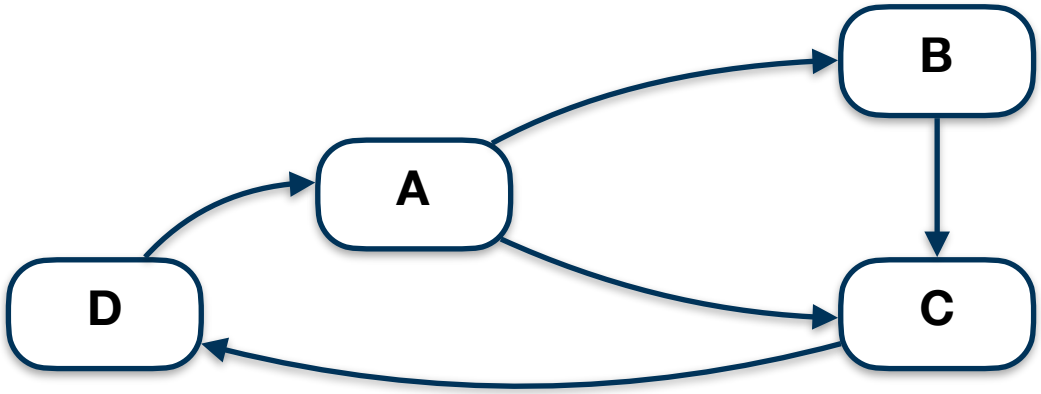
$a_B = h_A = 5/3$

$a_C = h_A + h_B = 5/3 + 1$

$a_D = h_C = 1/3$

Aufgabe 2

HITS Algorithmus



1. Iteration

	A	B	C	D
Hubs	2	1	1	1
Vorläufige Autoritäten	1	2	3	1
Normalisierte Autoritäten	1/3	2/3	1	1/3

2. Iteration

	A	B	C	D
Hubs	5/3	1	1/3	1/3
Vorläufige Autoritäten	1/3	5/3	8/3	1/3
Normalisierte Autoritäten	1/8	5/8	1	1/8

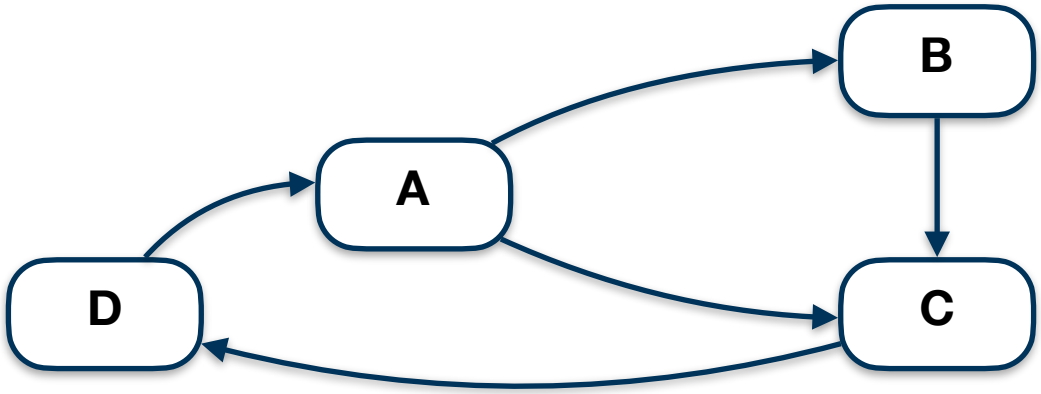
Normalisieren:

max(a) = 8/3

=> a_i*3/8

Aufgabe 2

HITS Algorithmus



1. Iteration

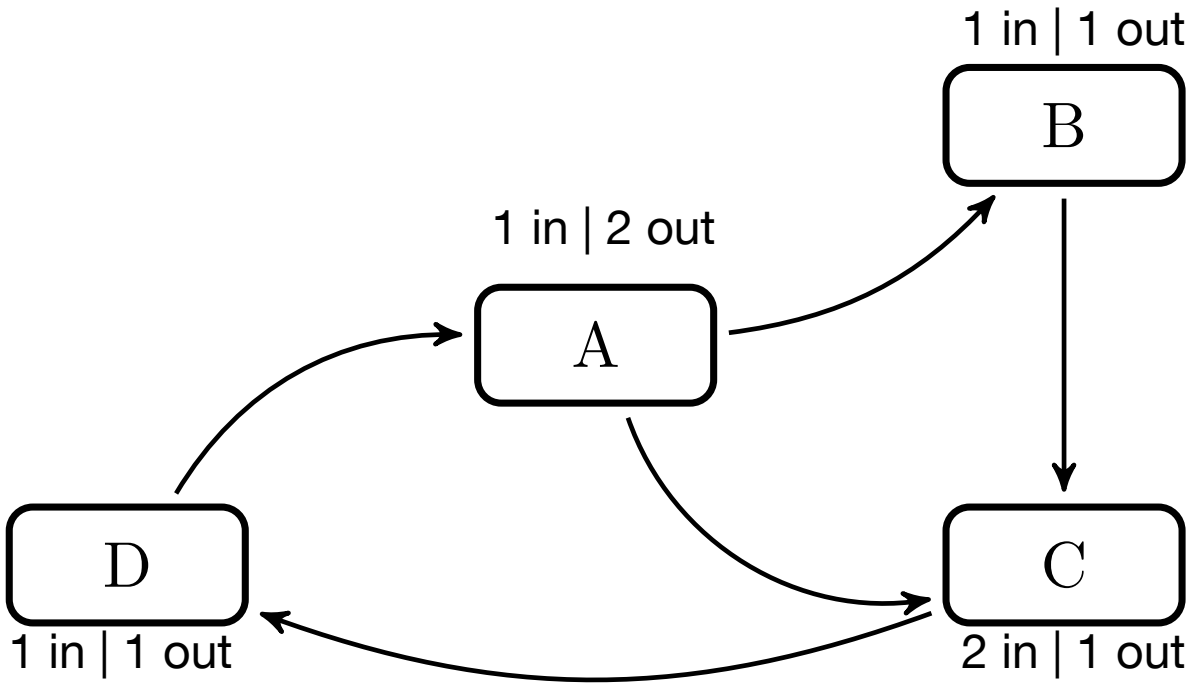
	A	B	C	D
Hubs	2	1	1	1
Vorläufige Autoritäten	1	2	3	1
Normalisierte Autoritäten	1/3	2/3	1	1/3

2. Iteration

	A	B	C	D
Hubs	5/3	1	1/3	1/3
Vorläufige Autoritäten	1/3	5/3	8/3	1/3
Normalisierte Autoritäten	1/8	5/8	1	1/8

Aufgabe 2

Pagerank - Berechnung am Graphen



Aufgabe 2
Pagerank

```
2. select VTo, 0.1/(CAST((select count(*) from pagerank)AS FLOAT))
      +0.9*sum(Beitrag)
   from(
     select e.VTo, p.Weight/
        (select count(*) from edges x where x.VFrom=e.VFrom) as Beitrag
     from edges e , pagerank p
     where e.VFrom=p.Vertex
   ) i
   group by VTo
```

Zentralitätsmaße

- aus den Sozialwissenschaften
- charakterisiert ganze Graphen oder Teilstrukturen
- Sinnhaftigkeit ist manchmal zweifelhaft

Zentralitätsmaße
Verbindungszentralität

Verbindungszentralität

v* = Knoten mit max. Grad

$$C_D(G) = \sum_{i=1}^{|V|}[C_D(v^*) - C_D(v_i)]$$

- aus den

Für Sternförmige Graphen, wird zur Normierung genutzt

- charakte
- Sinnhaft

$$C_D(G^*) = \sum_{i=1}^{|V|}[C_D(v*) - C_D(v_i)] = (|V| - 2)(|V| - 1)$$

Normierte Verbindungszentralität

$$C'_D(G) = C_D(G)/[(|V| - 2)(|V| - 1)]$$

Zentralitätsmaße
Nähe-Zentralität

- aus den Sozialwissenschaften
- charakterisiert ganz

$$H(v) = \sum_{v \neq y \in V} 1/d(y, v)$$

Hierbei definiert man $1/\infty$ als 0.

- Sinnhaftigkeit ist manchmal zweifelhaft

Zentralitätsmaße
Pfad-Zentralität

Für einen Knoten v im Graph $G = (V, E)$ wird also dieser Wert wie folgt bestimmt:

1. Für jedes Knotenpaar (s, t) berechne deren kürzeste Pfade im Graphen.
2. Bestimme die Anzahl der kürzesten Pfade von s nach t als σ_{st}.
3. Für jedes Knotenpaar (s, t) bestimme die Anzahl der kürzesten Pfade, die durch den betrachteten Knoten v verlaufen. Dieser Wert sei als $\sigma_{st}(v)$ bezeichnet.

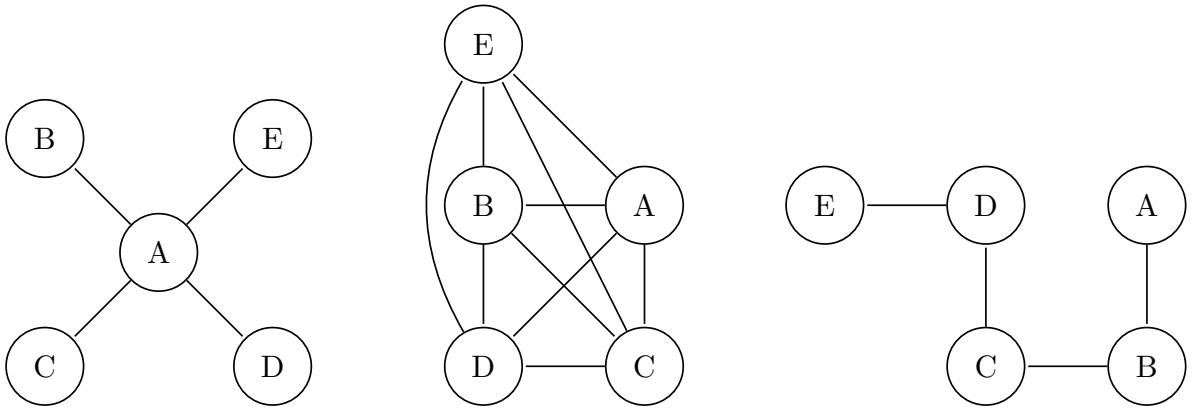
- a
- c
- s

$$C_B(v) = \sum_{s \neq v \neq t \in V} (\sigma_{st}(v)/\sigma_{st})$$

Aufgabe 3

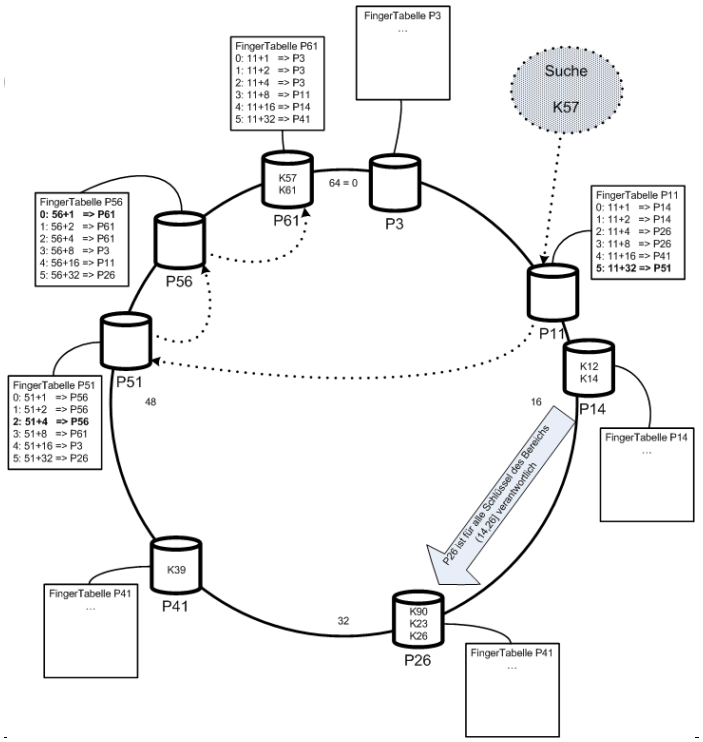
In Abbildung 2 sind drei Graphen gegeben, ein sternförmiger, eine Clique und ein linear angeordneter.

1. Berechnen Sie den Grad der Knoten für jeden der Graphen.
2. Berechnen Sie die Verbindungszentralität $C_D(G)$ der drei Graphen, sowie deren normierte Verbindungszentralität $C'_D(G)$.
3. Berechnen Sie die Nähe-Zentralität $H_G(v)$ für einen Knoten der drei Graphen.
4. Berechnen Sie die Pfad-Zentralität $H_G(v)$ für einen Knoten der drei Graphen.



Verteilte Datenbanksysteme

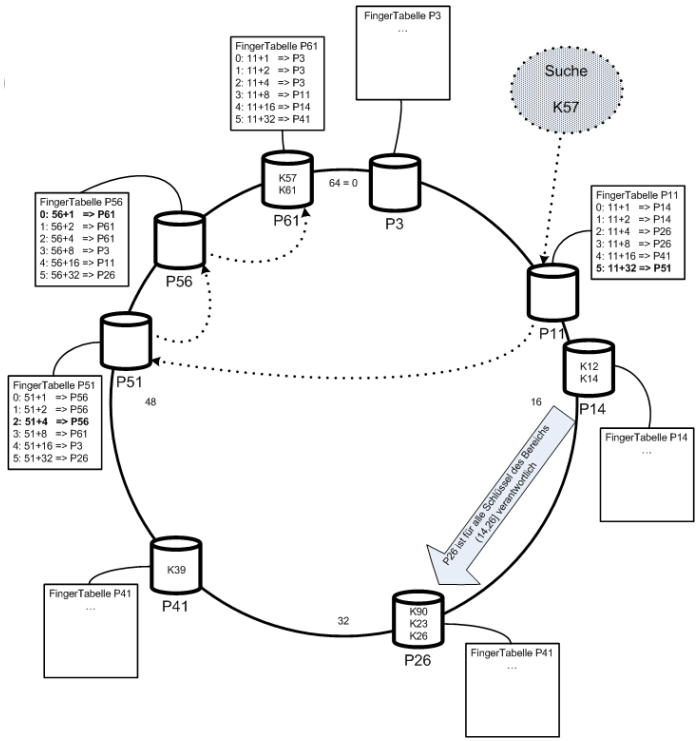
Chord-Overlaynetzwerk



- Dezentral organisiert
- Alle Peers haben die gleiche Funktionalität
- Anfragen werden zielgerichtet geroutet
- Peers werden mittels Hashfunktion auf Zahlenring $[0\ldots2^n)$ platziert (IP)
- Datenobjekte werden auf den Zahlenring gehasht (durch Suchschlüssel)

Verteilte Datenbanksysteme

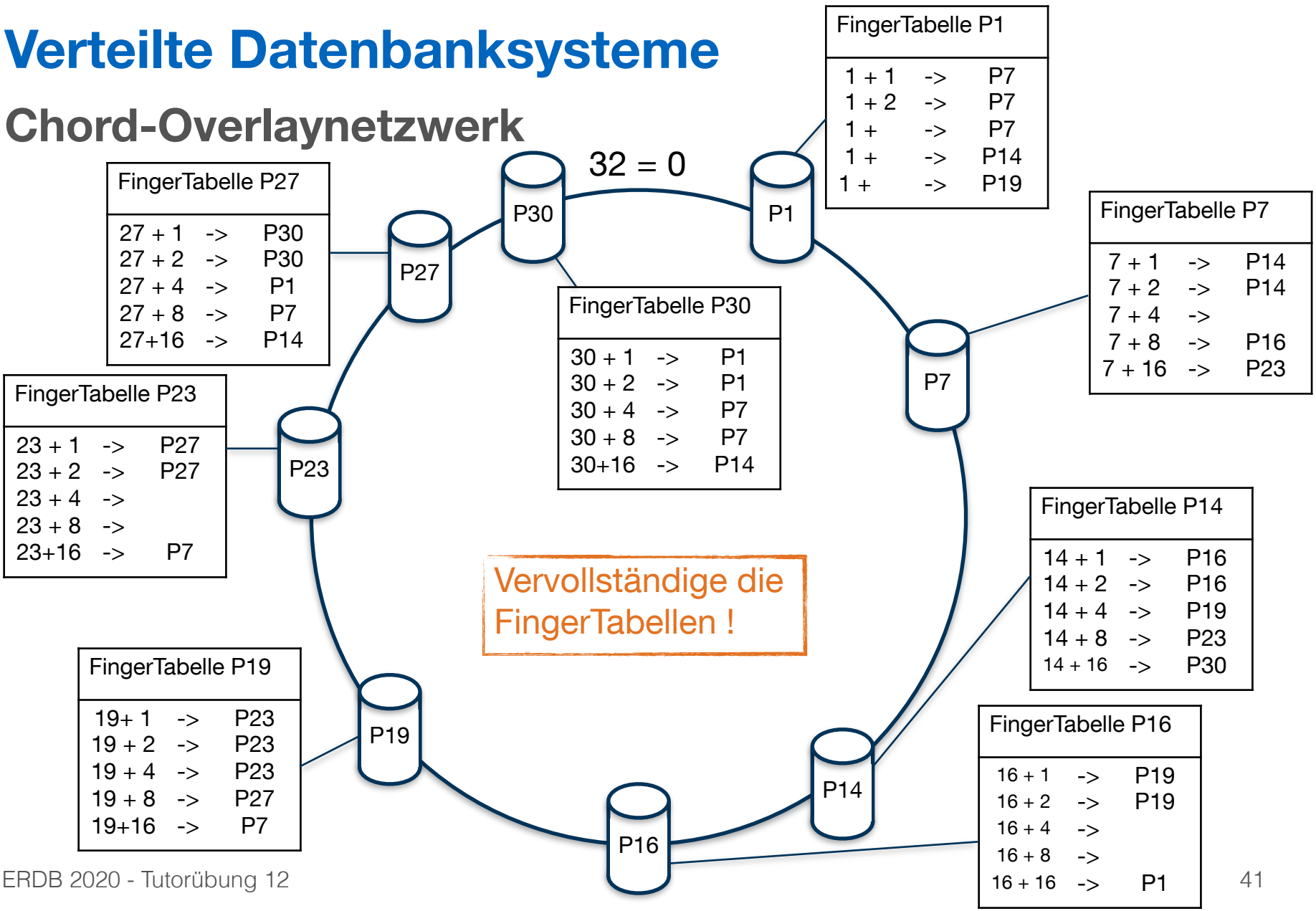
Chord-Overlaynetzwerk



- Peer ist für Daten zwischen Vorgänger und seiner Hashzahl verantwortlich
- Fällt ein Peer aus, wird der Bereich dem Nachfolger zugeordnet
- Jeder Peer hat eine FingerTabelle:
  - Enthält IP-Adressen von log(#Peers) Peers

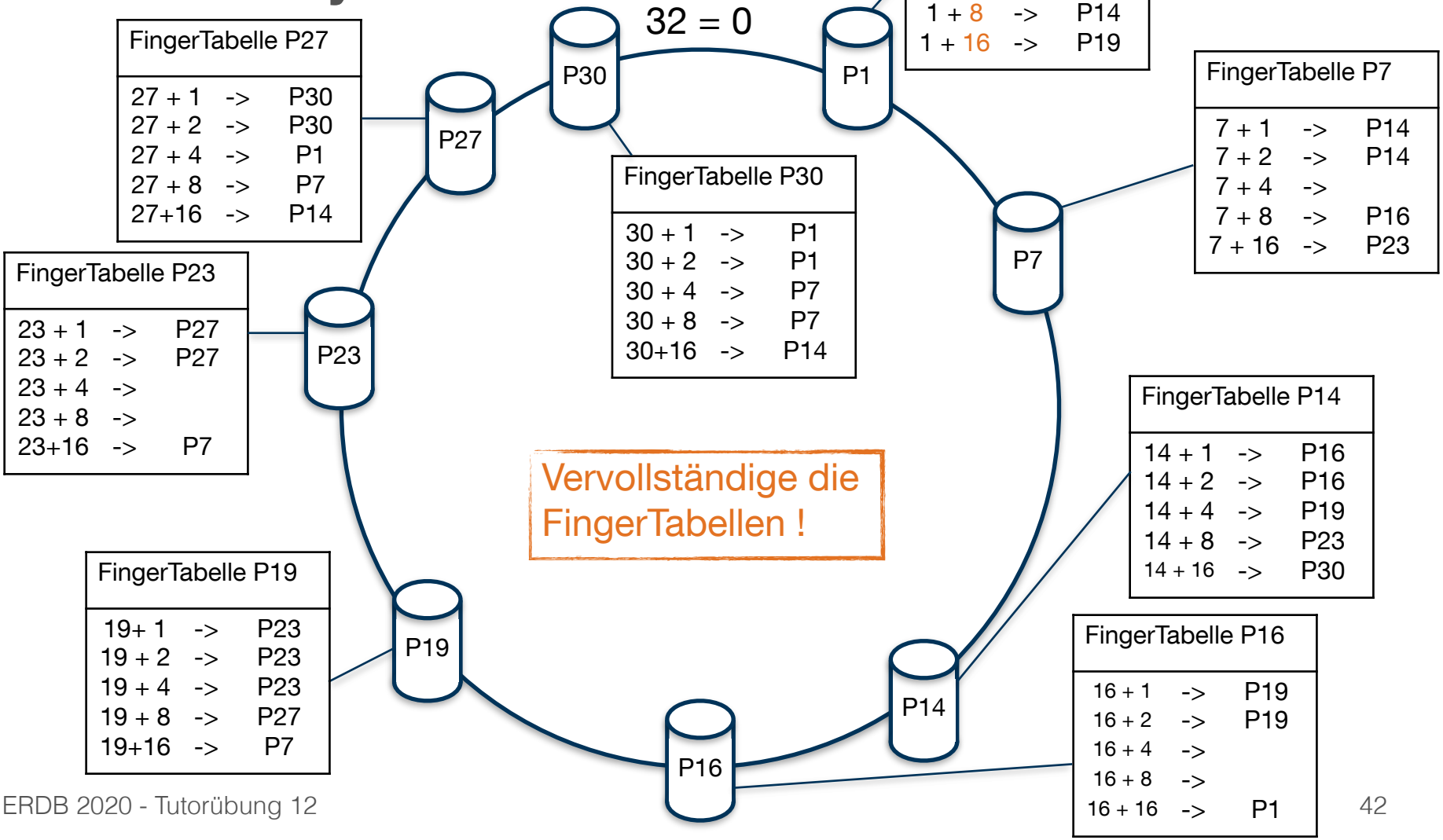
Verteilte Datenbanksysteme

Chord-Overlaynetzwerk



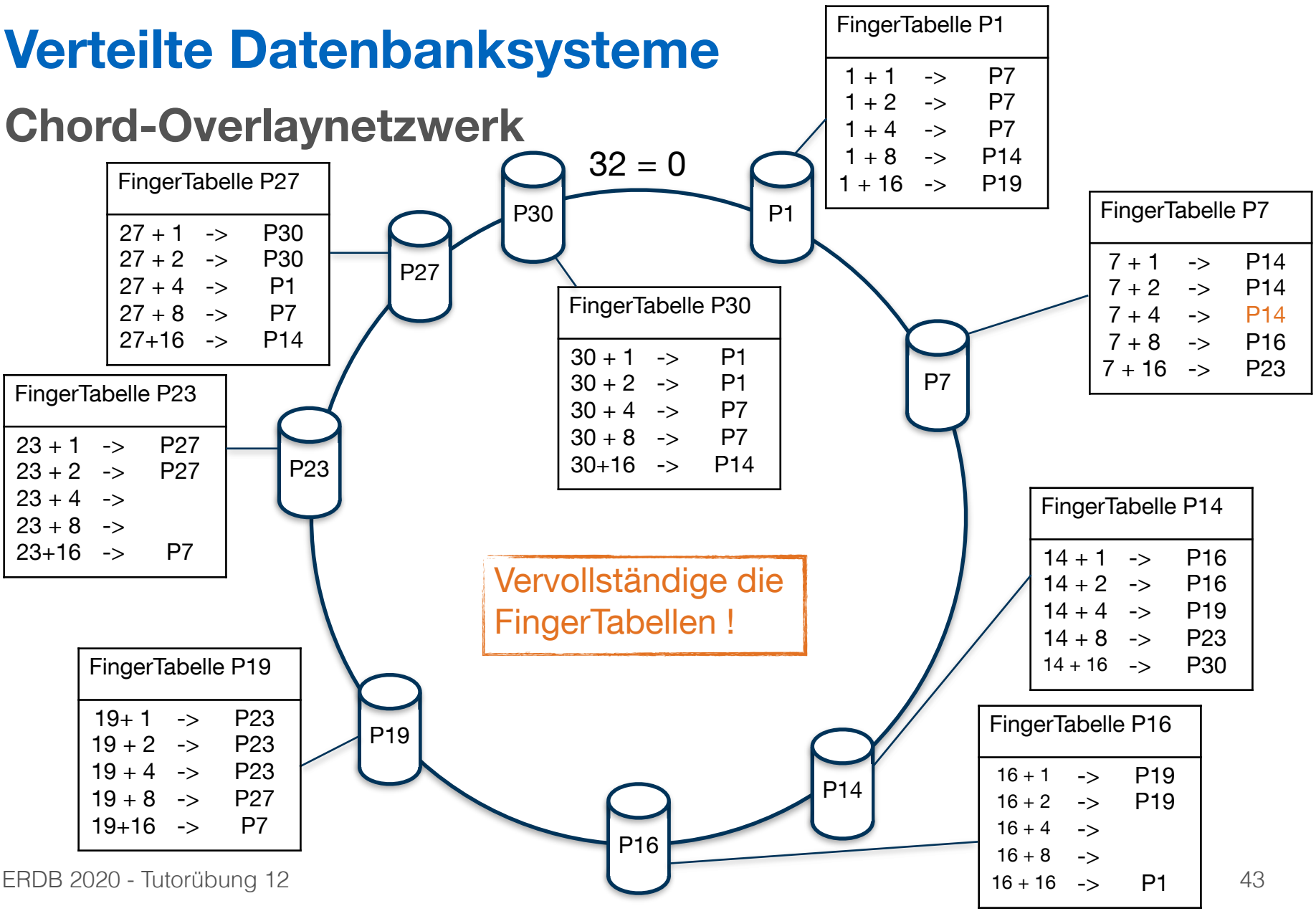
Verteilte Datenbanksysteme

Chord-Overlaynetzwerk



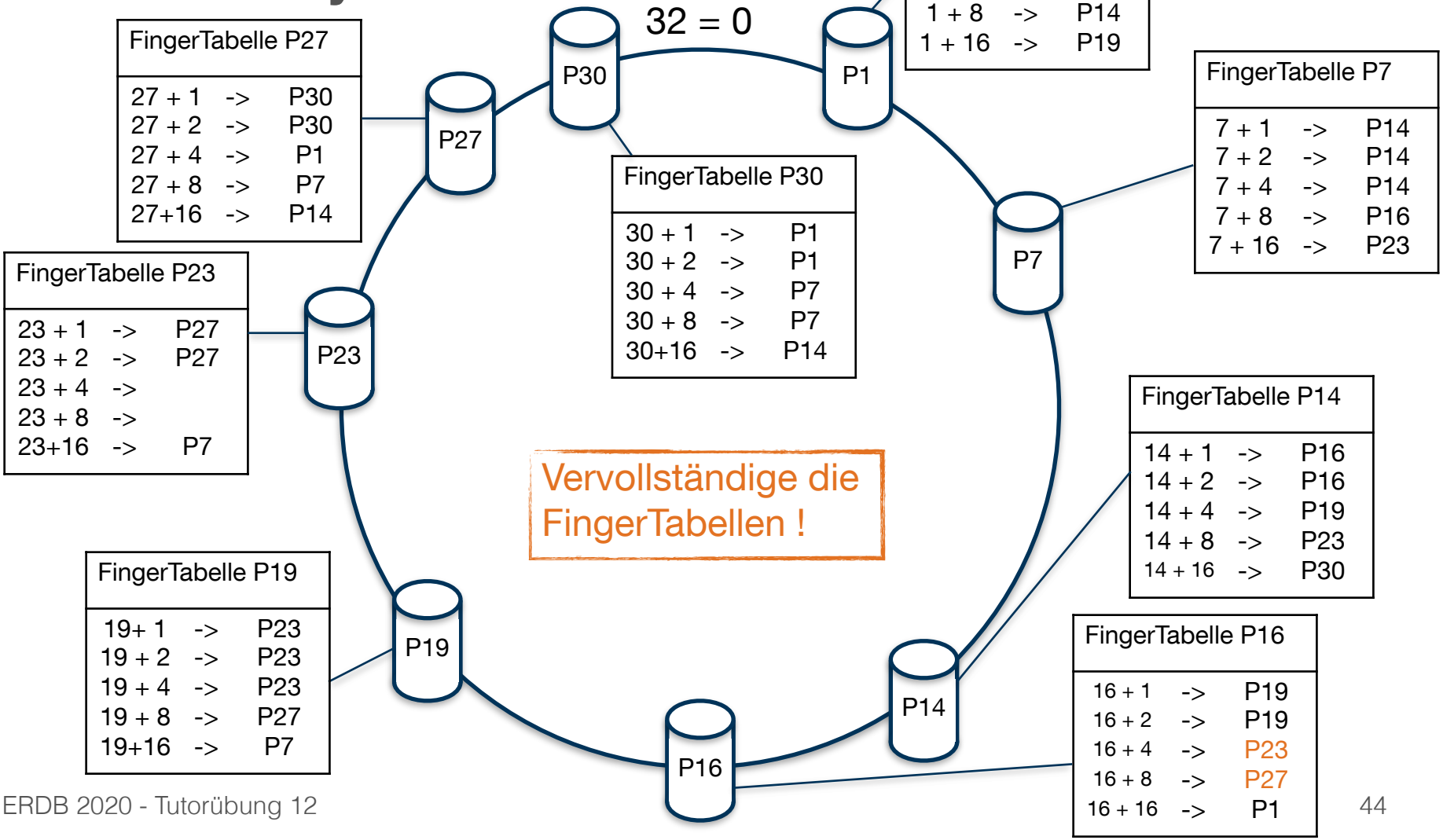
Verteilte Datenbanksysteme

Chord-Overlaynetzwerk



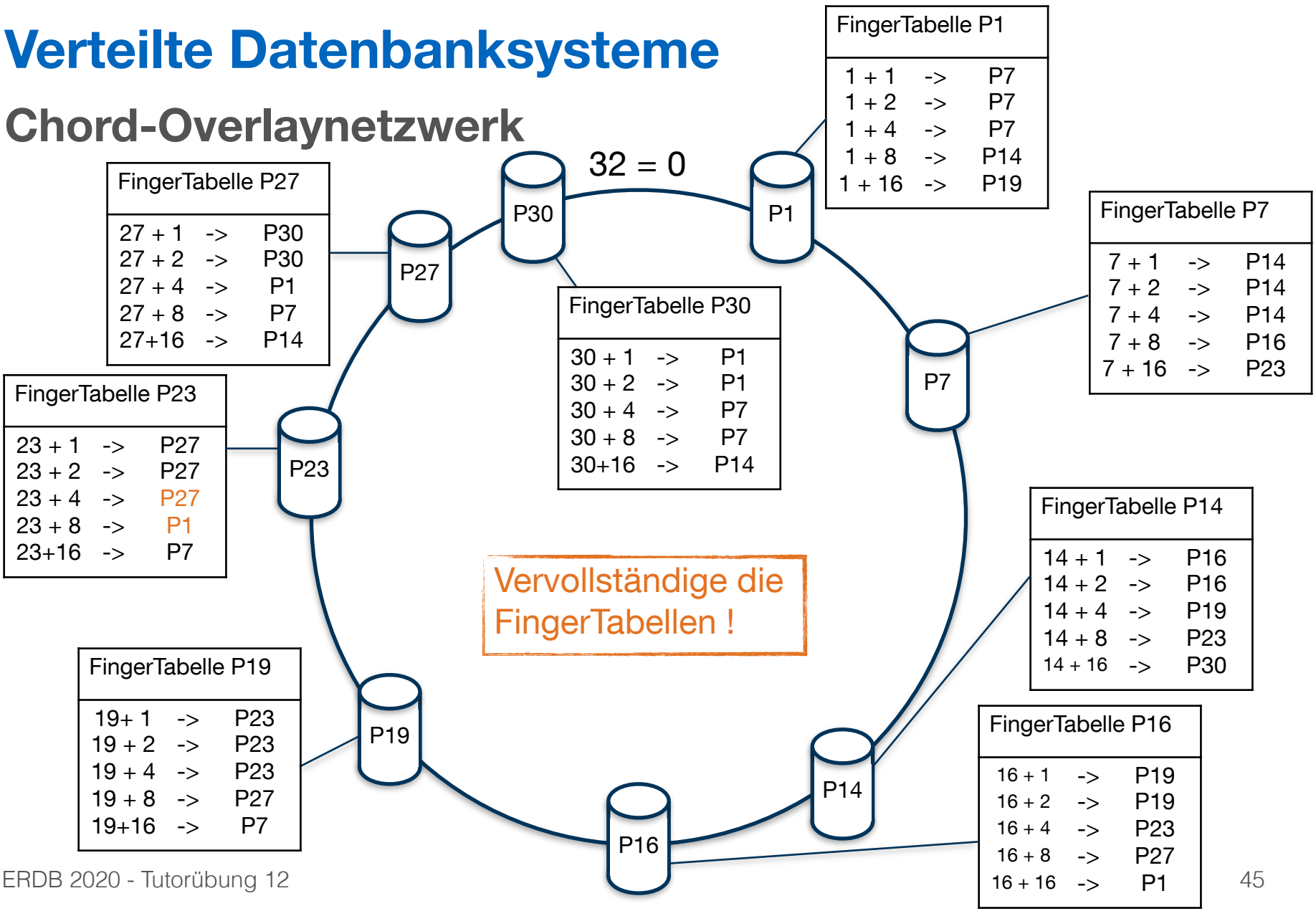
Verteilte Datenbanksysteme

Chord-Overlaynetzwerk



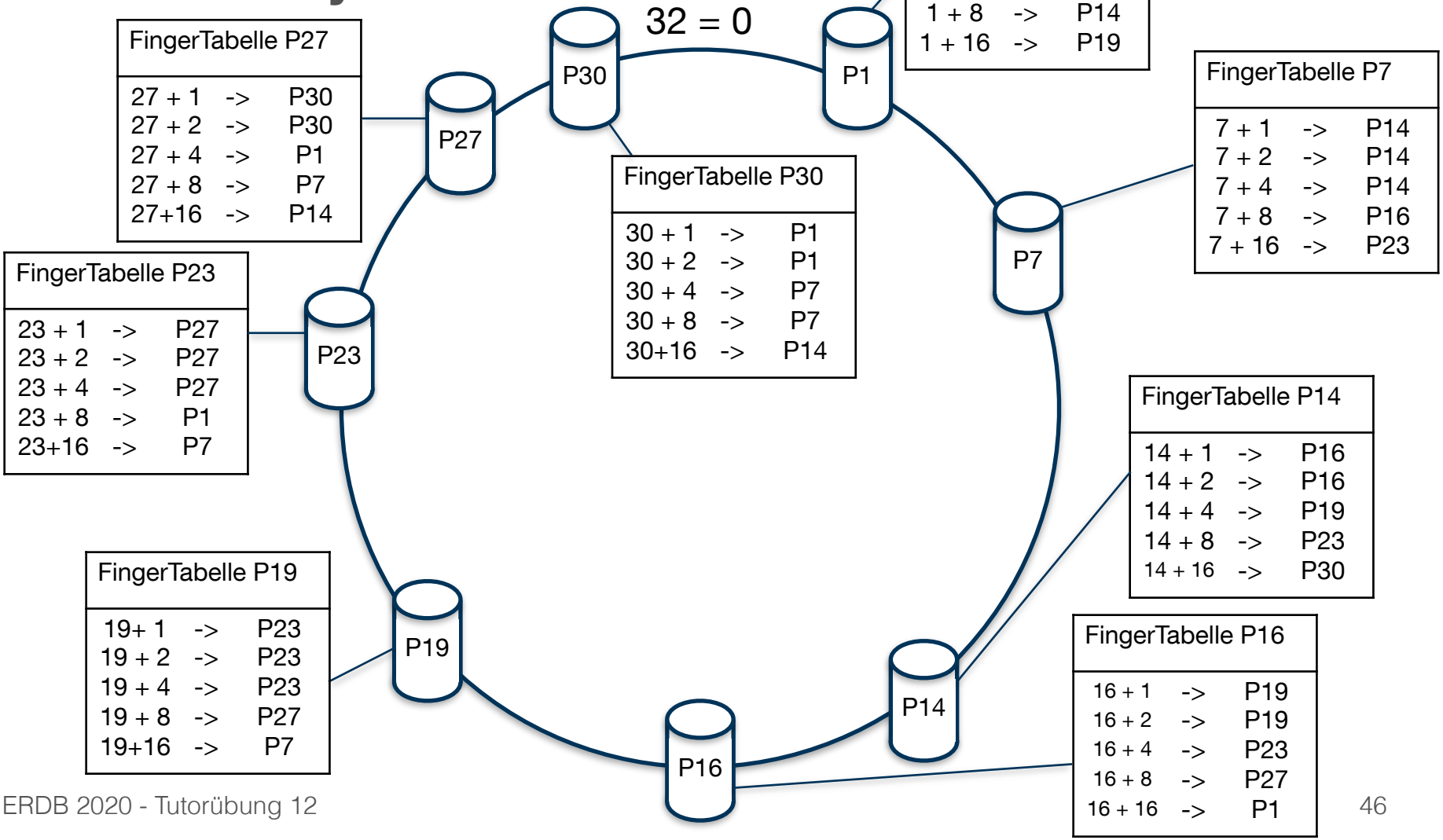
Verteilte Datenbanksysteme

Chord-Overlaynetzwerk



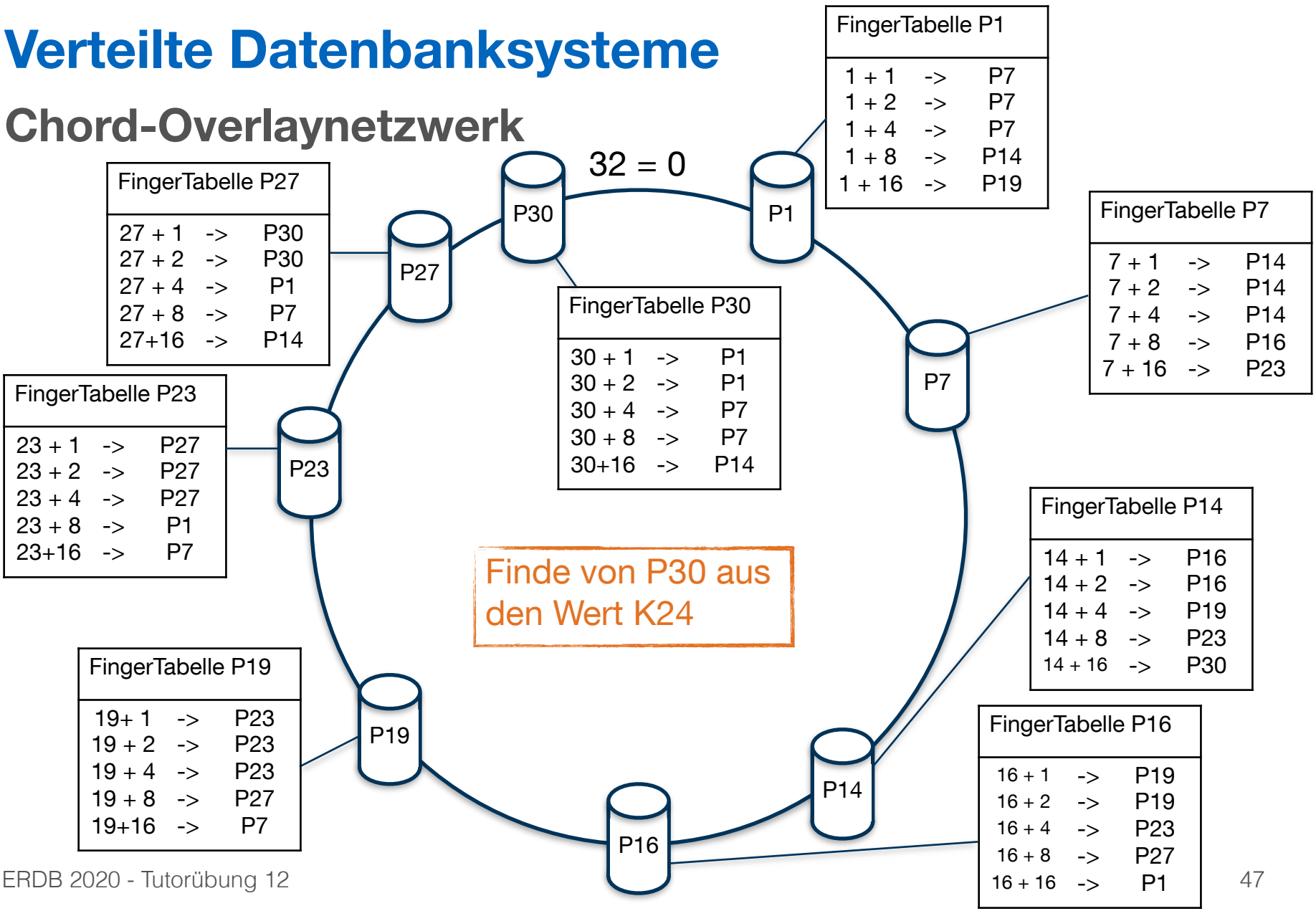
Verteilte Datenbanksysteme

Chord-Overlaynetzwerk



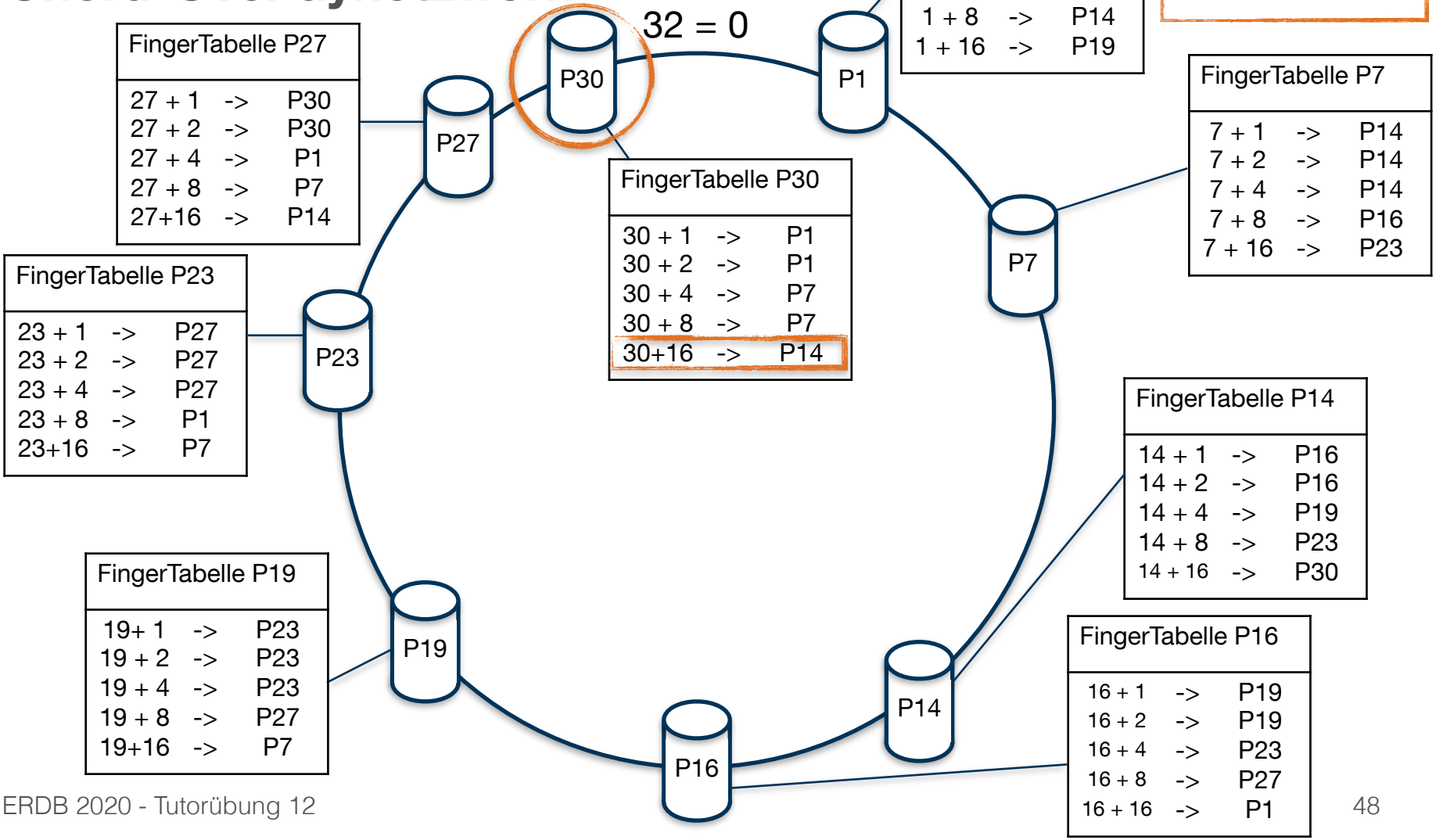
Verteilte Datenbanksysteme

Chord-Overlaynetzwerk



Verteilte Datenbanksysteme

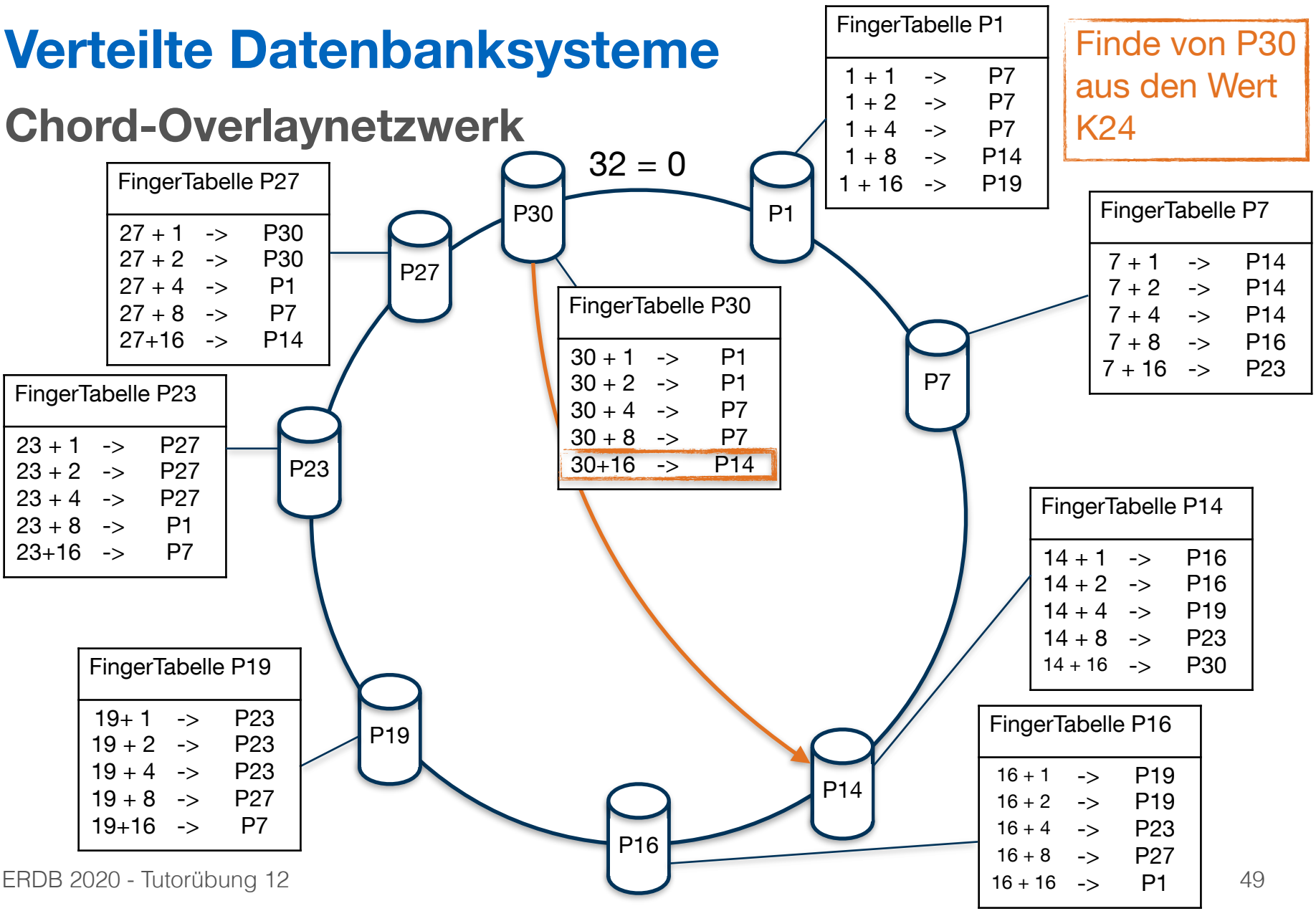
Chord-Overlaynetzwerk



Finde von P30 aus den Wert K24

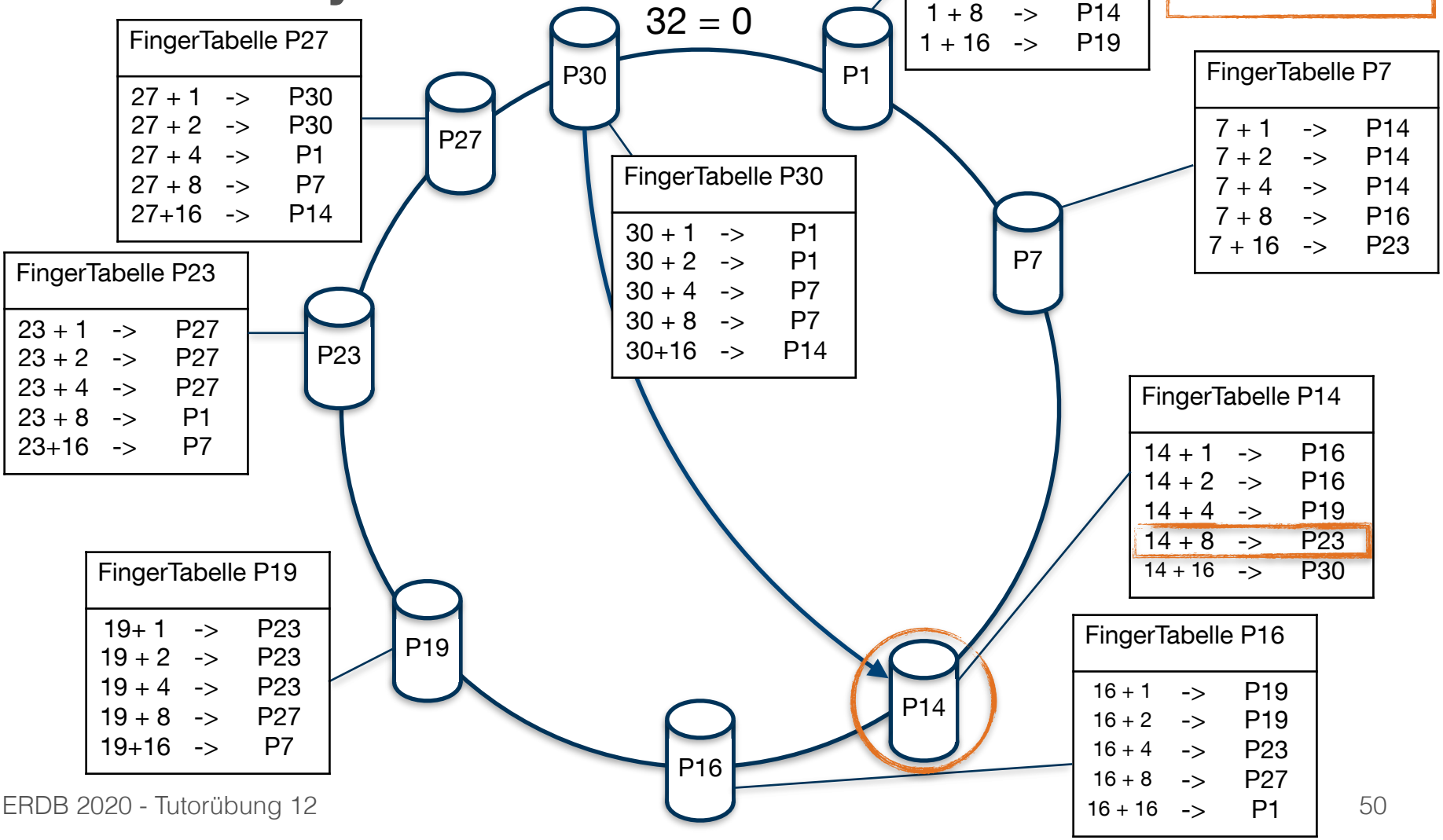
Verteilte Datenbanksysteme

Chord-Overlaynetzwerk



Verteilte Datenbanksysteme

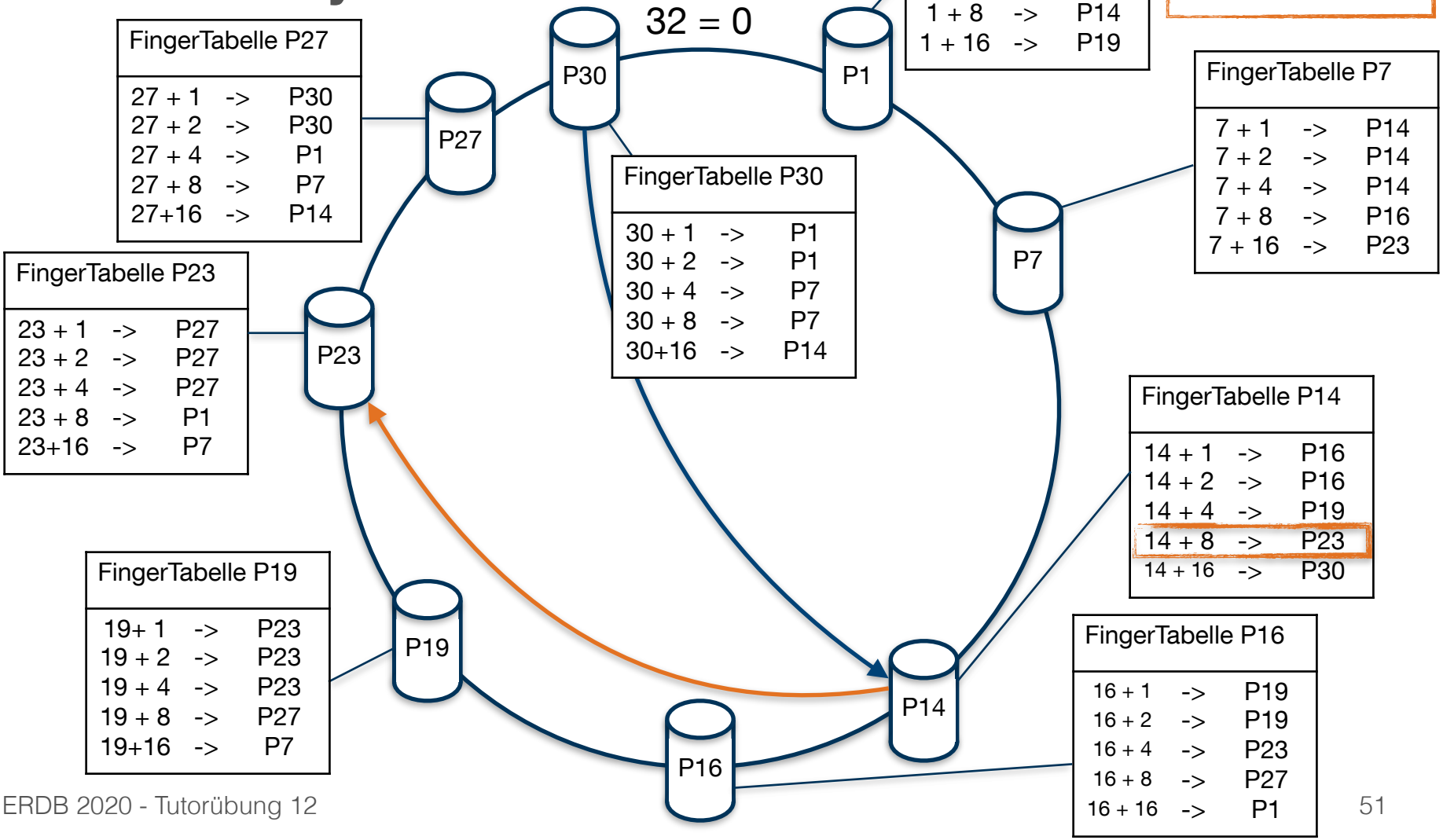
Chord-Overlaynetzwerk



Finde von P30 aus den Wert K24

Verteilte Datenbanksysteme

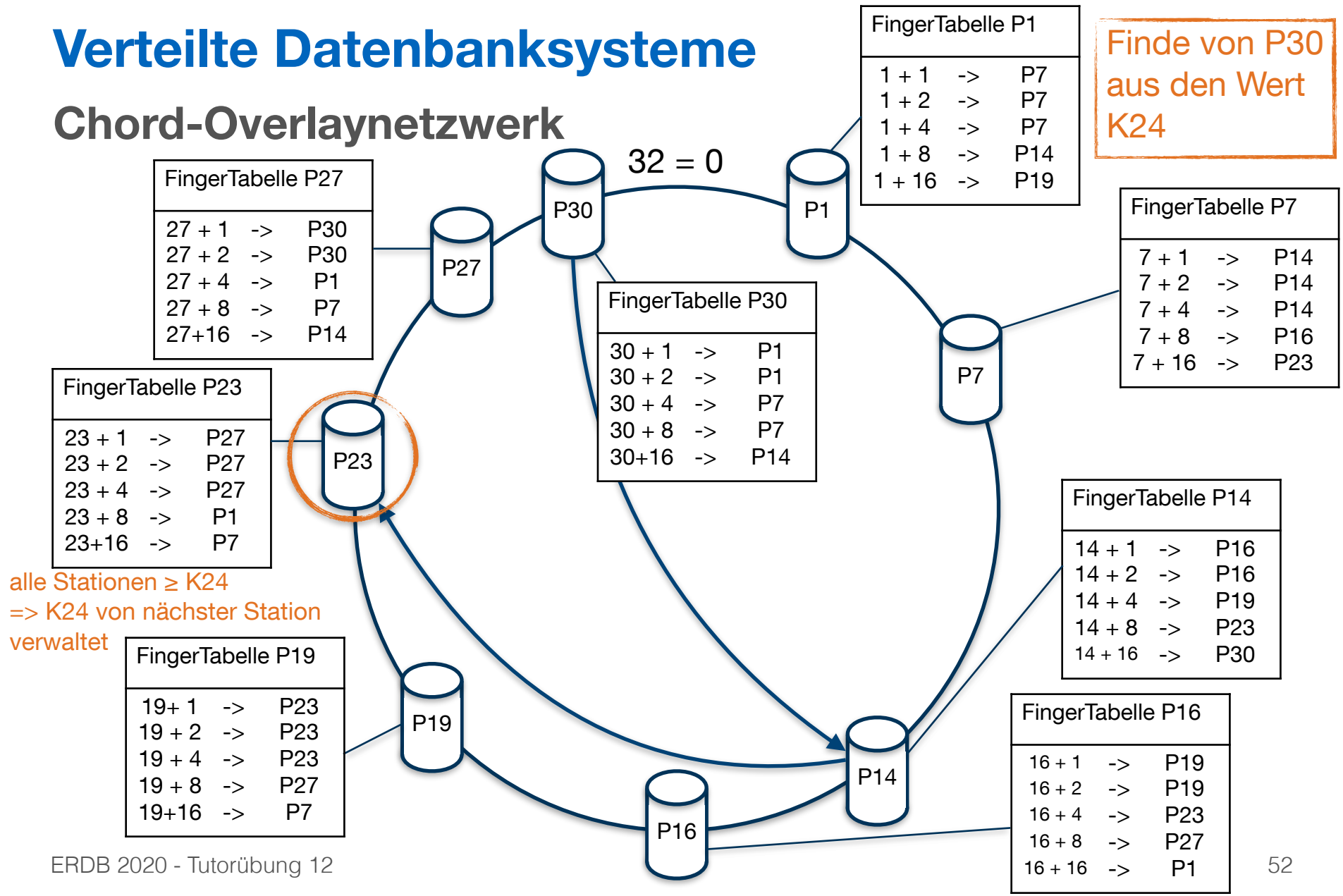
Chord-Overlaynetzwerk



Finde von P30 aus den Wert K24

Verteilte Datenbanksysteme

Chord-Overlaynetzwerk

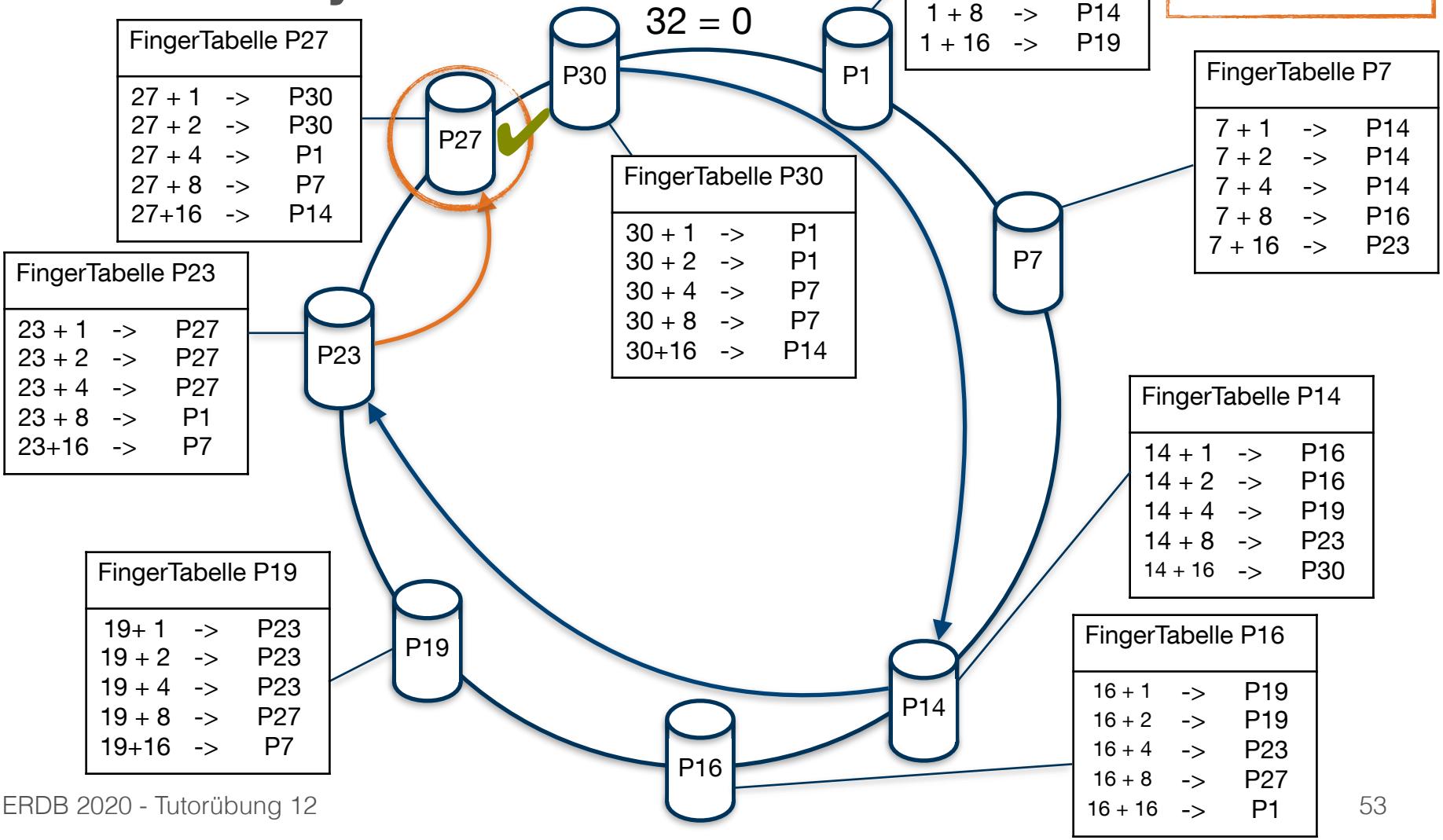


Finde von P30 aus den Wert K24

alle Stationen ≥ K24
=> K24 von nächster Station verwaltet

Verteilte Datenbanksysteme

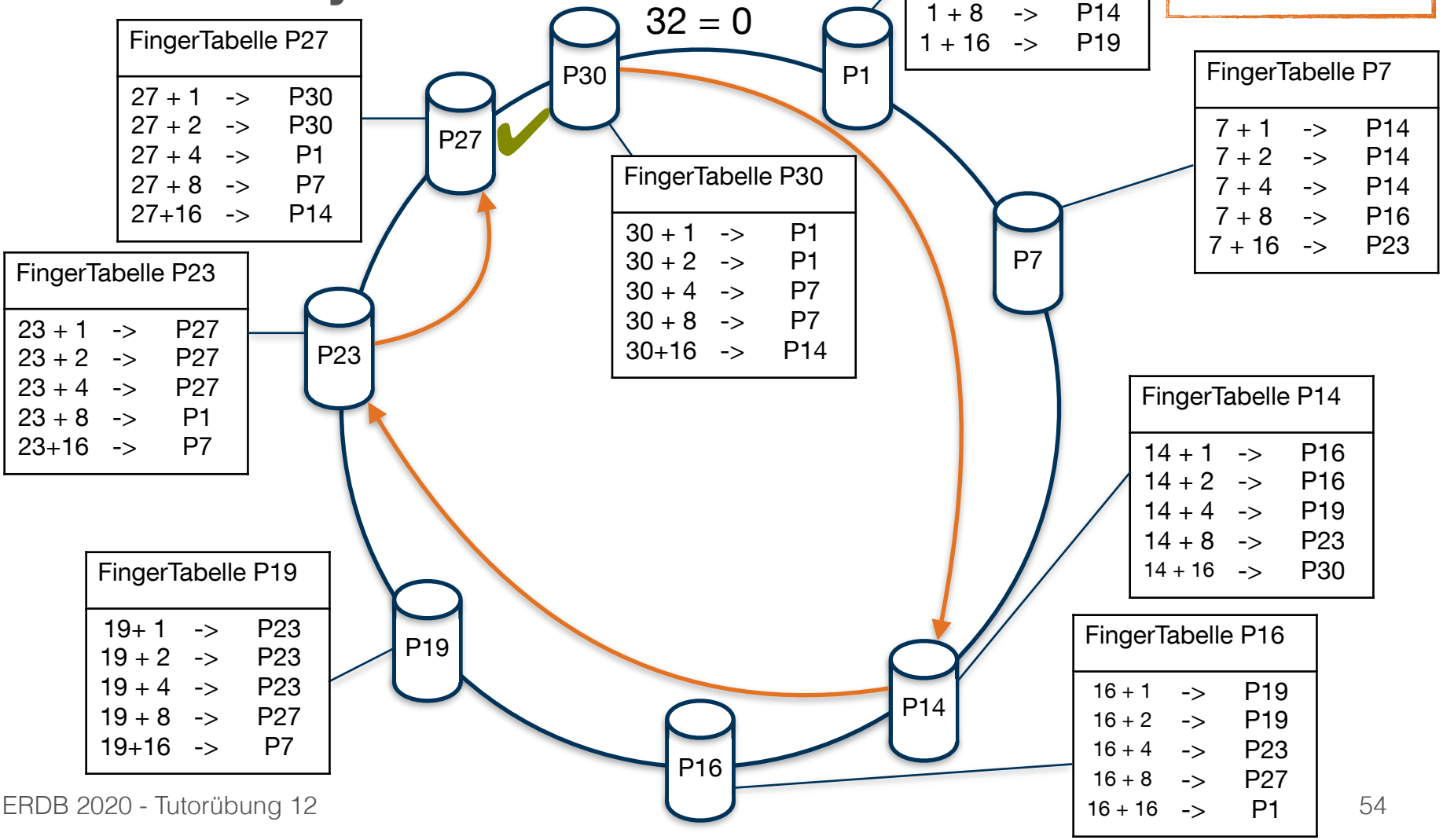
Chord-Overlaynetzwerk



Finde von P30 aus den Wert K24

Verteilte Datenbanksysteme

Chord-Overlaynetzwerk

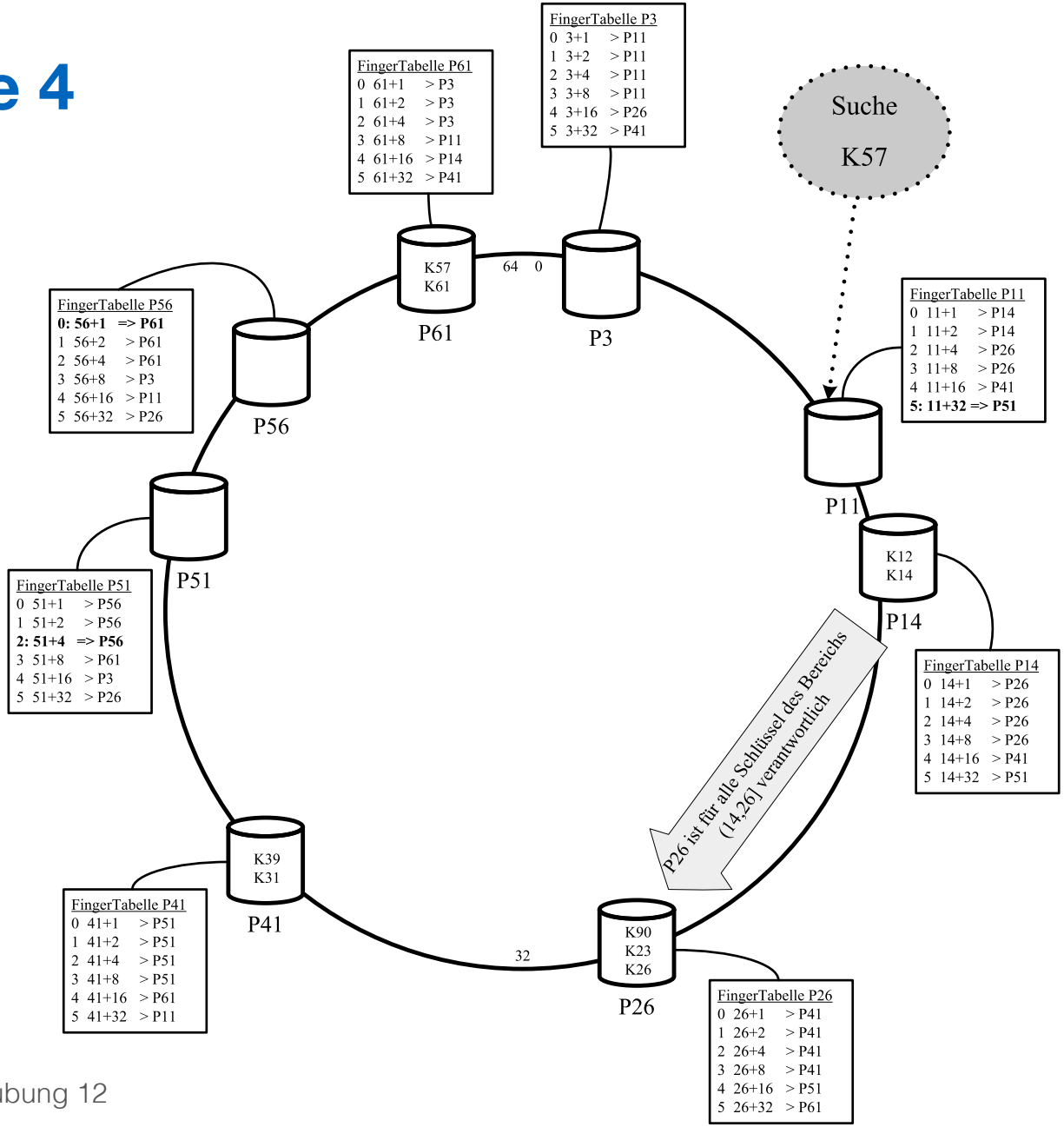


Finde von P30 aus den Wert K24

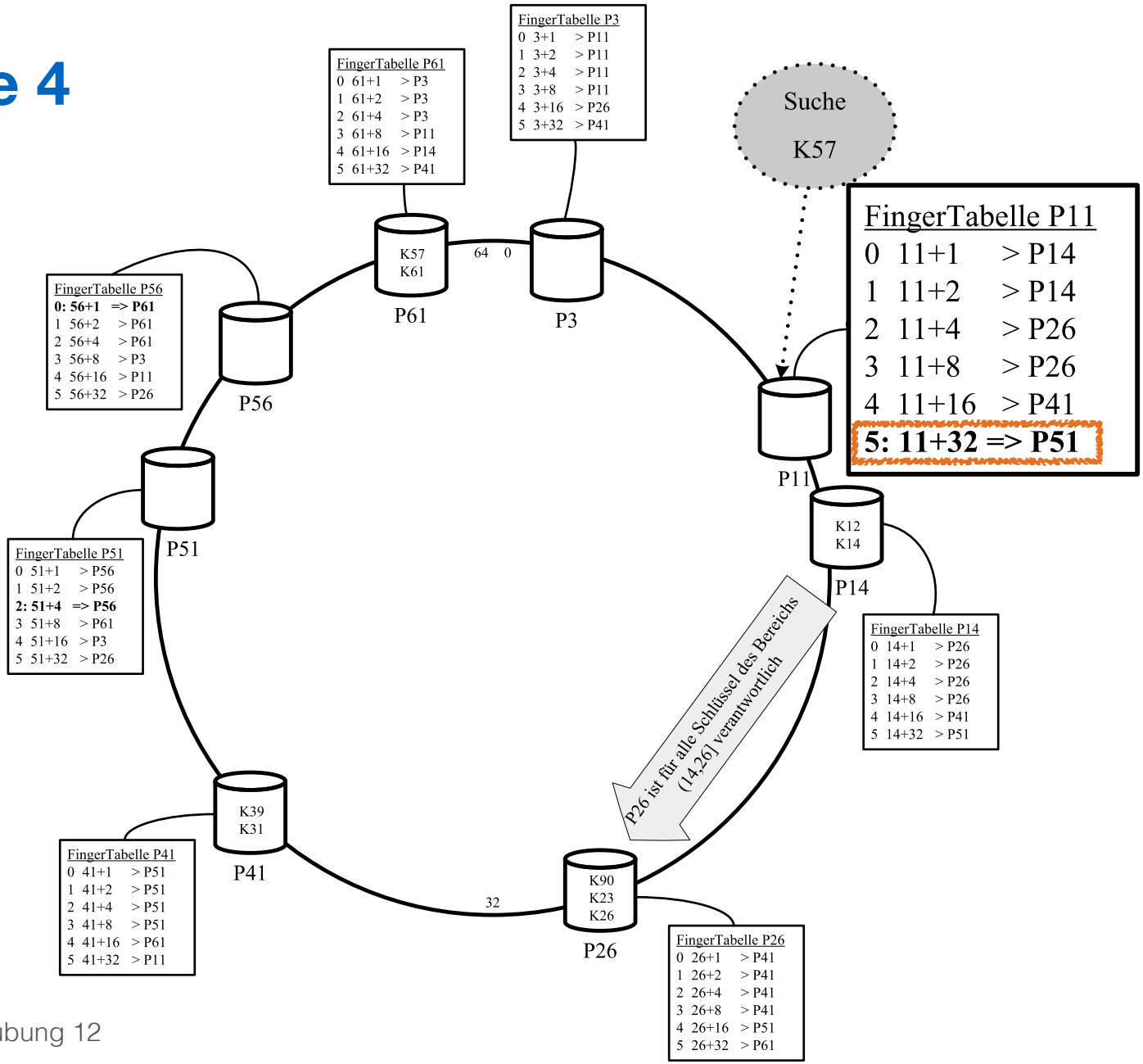
Aufgabe 4

Zeigen Sie, dass die Suche in einem Chord-Overlaynetzwerk durch die Nutzung der FingerTabellen in maximal logarithmisch vielen Schritten zur Größe des Zahlenrings (bzw. der Anzahl der Stationen) durchgeführt werden kann. Verwenden Sie die Suche nach K57 beginnend an Station P11 (siehe Abbildung 2) zur Illustration.

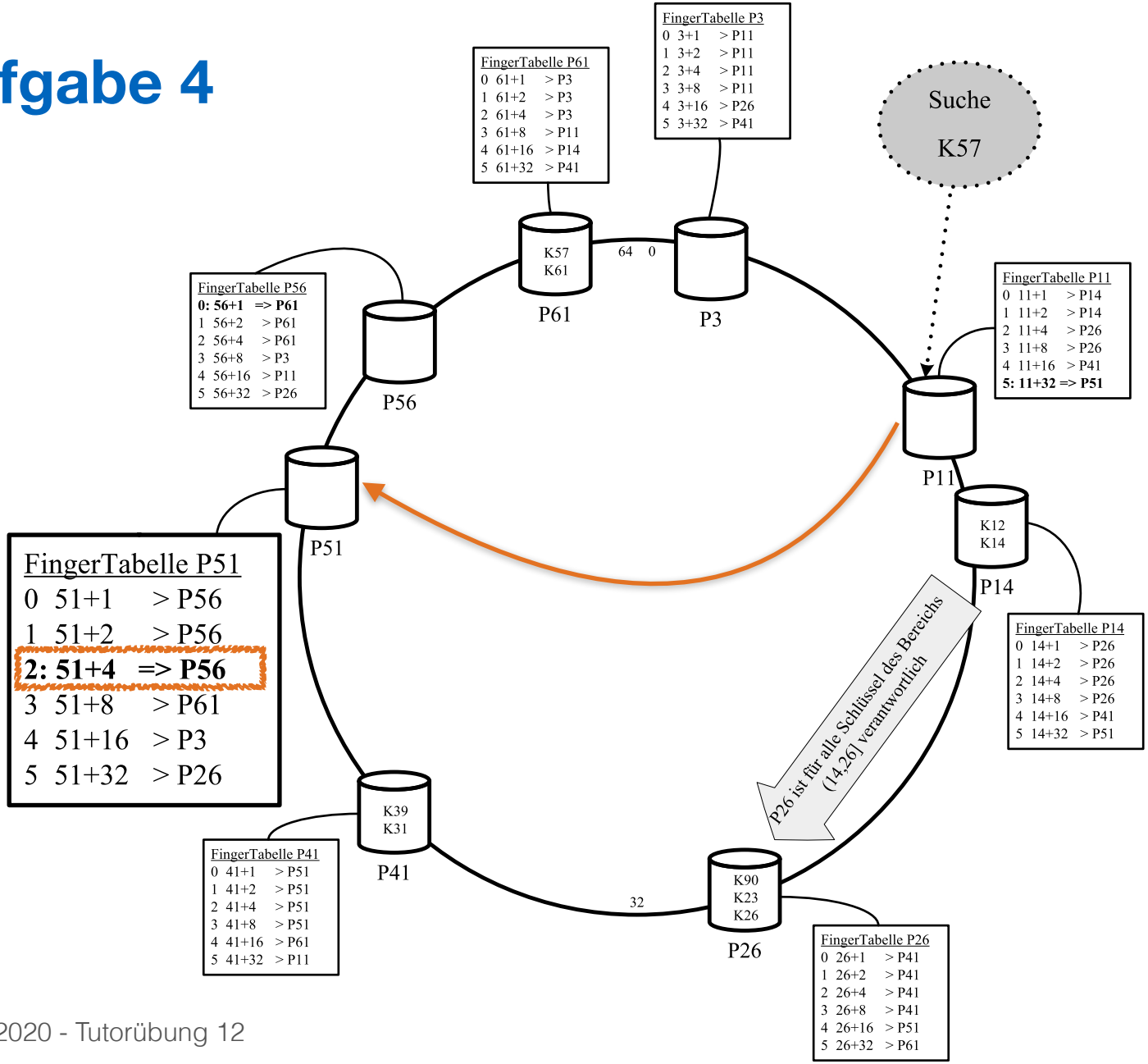
Aufgabe 4


FingerTabelle P3
0 3+1 > P11
1 3+2 > P11
2 3+4 > P11
3 3+8 > P11
4 3+16 > P26
5 3+32 > P41
FingerTabelle P61
0 61+1 > P3
1 61+2 > P3
2 61+4 > P3
3 61+8 > P11
4 61+16 > P14
5 61+32 > P41
Suche
K57
K57
K61
64
0
P61
P3
FingerTabelle P56
0: 56+1 => P61
1 56+2 > P61
2 56+4 > P61
3 56+8 > P3
4 56+16 > P11
5 56+32 > P26
P56
FingerTabelle P11
0 11+1 > P14
1 11+2 > P14
2 11+4 > P26
3 11+8 > P26
4 11+16 > P41
5: 11+32 => P51
P11
K12
K14
P14
FingerTabelle P51
0 51+1 > P56
1 51+2 > P56
2: 51+4 => P56
3 51+8 > P61
4 51+16 > P3
5 51+32 > P26
P51
FingerTabelle P14
0 14+1 > P26
1 14+2 > P26
2 14+4 > P26
3 14+8 > P26
4 14+16 > P41
5 14+32 > P51
P26 ist für alle Schlüssel des Bereichs (14,26] verantwortlich
K39
K31
P41
FingerTabelle P41
0 41+1 > P51
1 41+2 > P51
2 41+4 > P51
3 41+8 > P51
4 41+16 > P61
5 41+32 > P11
32
K90
K23
K26
P26
FingerTabelle P26
0 26+1 > P41
1 26+2 > P41
2 26+4 > P41
3 26+8 > P41
4 26+16 > P51
5 26+32 > P61

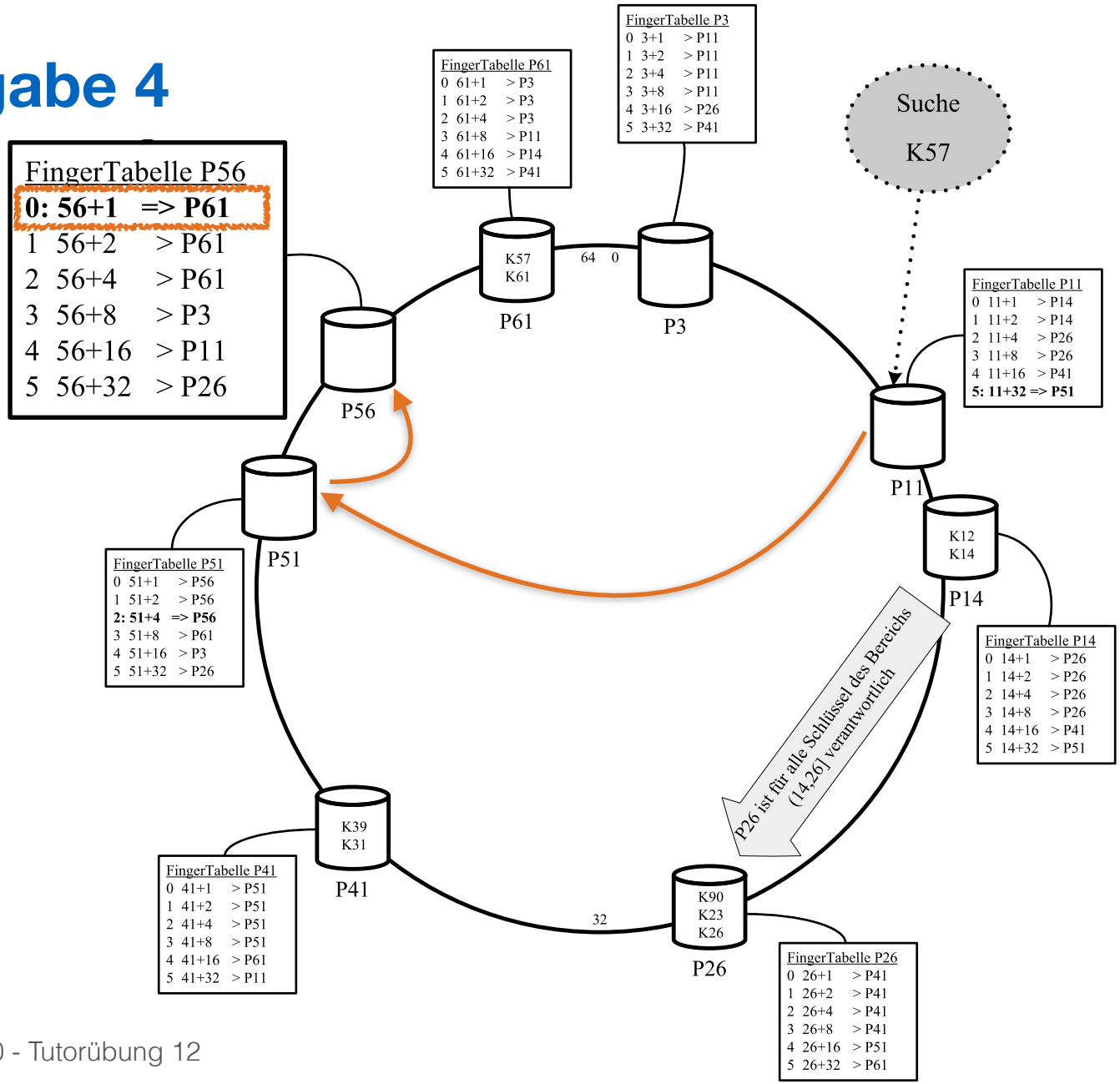
Aufgabe 4


FingerTabelle P3
0 3+1 > P11
1 3+2 > P11
2 3+4 > P11
3 3+8 > P11
4 3+16 > P26
5 3+32 > P41
FingerTabelle P61
0 61+1 > P3
1 61+2 > P3
2 61+4 > P3
3 61+8 > P11
4 61+16 > P14
5 61+32 > P41
Suche
K57
FingerTabelle P11
0 11+1 > P14
1 11+2 > P14
2 11+4 > P26
3 11+8 > P26
4 11+16 > P41
5: 11+32 => P51
K57
K61
64
0
P61
P3
P11
FingerTabelle P56
0: 56+1 => P61
1 56+2 > P61
2 56+4 > P61
3 56+8 > P3
4 56+16 > P11
5 56+32 > P26
P56
P51
FingerTabelle P51
0 51+1 > P56
1 51+2 > P56
2: 51+4 => P56
3 51+8 > P61
4 51+16 > P3
5 51+32 > P26
K12
K14
P14
FingerTabelle P14
0 14+1 > P26
1 14+2 > P26
2 14+4 > P26
3 14+8 > P26
4 14+16 > P41
5 14+32 > P51
P26 ist für alle Schlüssel des Bereichs (14,26] verantwortlich
K39
K31
P41
FingerTabelle P41
0 41+1 > P51
1 41+2 > P51
2 41+4 > P51
3 41+8 > P51
4 41+16 > P61
5 41+32 > P11
32
K90
K23
K26
P26
FingerTabelle P26
0 26+1 > P41
1 26+2 > P41
2 26+4 > P41
3 26+8 > P41
4 26+16 > P51
5 26+32 > P61

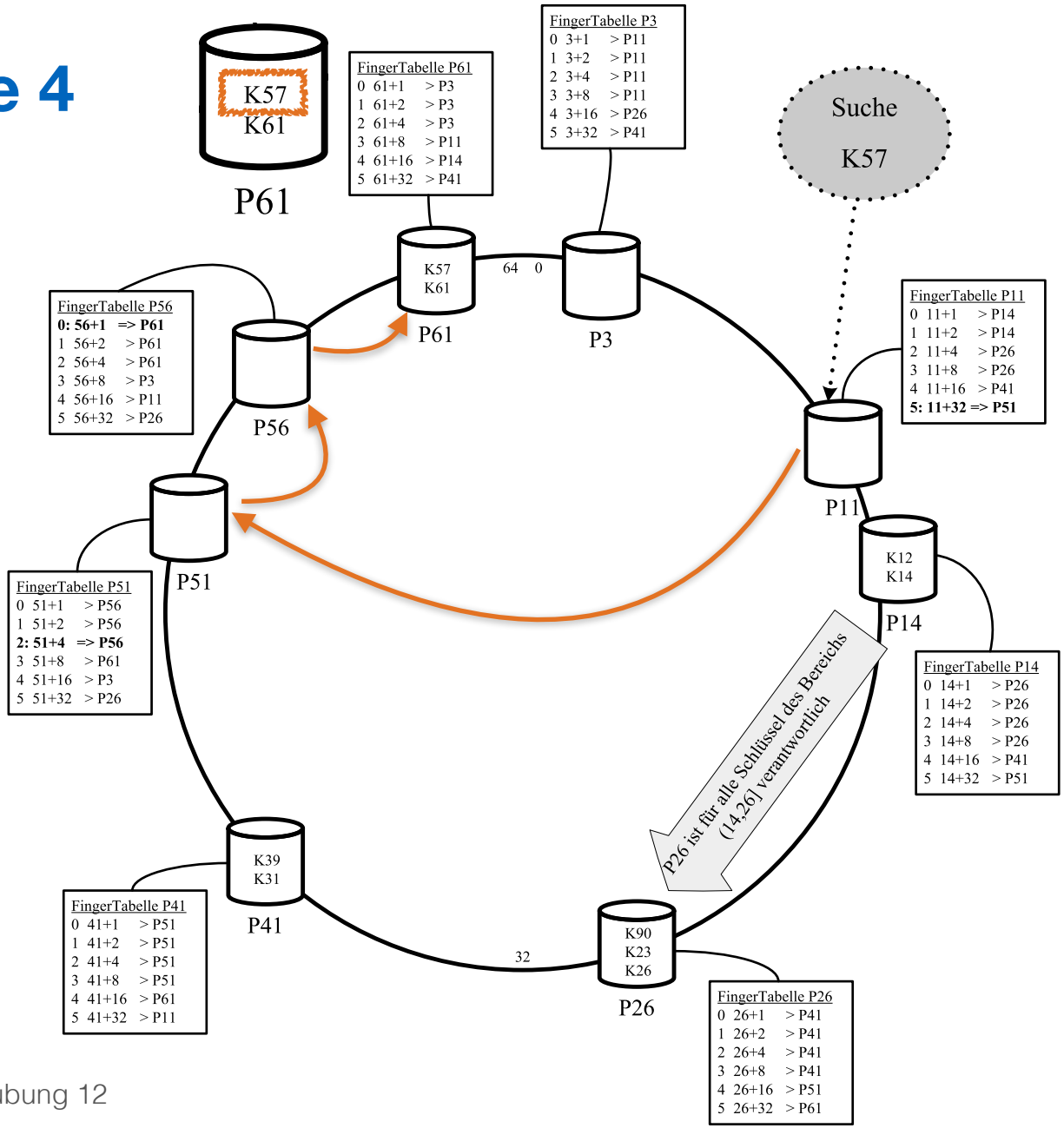
Aufgabe 4


FingerTabelle P3
0 3+1 > P11
1 3+2 > P11
2 3+4 > P11
3 3+8 > P11
4 3+16 > P26
5 3+32 > P41
FingerTabelle P61
0 61+1 > P3
1 61+2 > P3
2 61+4 > P3
3 61+8 > P11
4 61+16 > P14
5 61+32 > P41
Suche
K57
K57
K61
64
0
P61
P3
FingerTabelle P56
0: 56+1 => P61
1 56+2 > P61
2 56+4 > P61
3 56+8 > P3
4 56+16 > P11
5 56+32 > P26
P56
FingerTabelle P11
0 11+1 > P14
1 11+2 > P14
2 11+4 > P26
3 11+8 > P26
4 11+16 > P41
5: 11+32 => P51
P11
K12
K14
P14
FingerTabelle P14
0 14+1 > P26
1 14+2 > P26
2 14+4 > P26
3 14+8 > P26
4 14+16 > P41
5 14+32 > P51
P51
FingerTabelle P51
0 51+1 > P56
1 51+2 > P56
2: 51+4 => P56
3 51+8 > P61
4 51+16 > P3
5 51+32 > P26
P26 ist für alle Schlüssel des Bereichs (14,26] verantwortlich
K39
K31
P41
FingerTabelle P41
0 41+1 > P51
1 41+2 > P51
2 41+4 > P51
3 41+8 > P51
4 41+16 > P61
5 41+32 > P11
32
K90
K23
K26
P26
FingerTabelle P26
0 26+1 > P41
1 26+2 > P41
2 26+4 > P41
3 26+8 > P41
4 26+16 > P51
5 26+32 > P61

Aufgabe 4


FingerTabelle P56
0: 56+1 => P61
1 56+2 > P61
2 56+4 > P61
3 56+8 > P3
4 56+16 > P11
5 56+32 > P26
FingerTabelle P61
0 61+1 > P3
1 61+2 > P3
2 61+4 > P3
3 61+8 > P11
4 61+16 > P14
5 61+32 > P41
FingerTabelle P3
0 3+1 > P11
1 3+2 > P11
2 3+4 > P11
3 3+8 > P11
4 3+16 > P26
5 3+32 > P41
Suche
K57
K57
K61
P61
64
0
P3
P56
P11
FingerTabelle P11
0 11+1 > P14
1 11+2 > P14
2 11+4 > P26
3 11+8 > P26
4 11+16 > P41
5: 11+32 => P51
K12
K14
P14
FingerTabelle P14
0 14+1 > P26
1 14+2 > P26
2 14+4 > P26
3 14+8 > P26
4 14+16 > P41
5 14+32 > P51
P51
FingerTabelle P51
0 51+1 > P56
1 51+2 > P56
2: 51+4 => P56
3 51+8 > P61
4 51+16 > P3
5 51+32 > P26
P26 ist für alle Schlüssel des Bereichs (14,26] verantwortlich
K39
K31
P41
FingerTabelle P41
0 41+1 > P51
1 41+2 > P51
2 41+4 > P51
3 41+8 > P51
4 41+16 > P61
5 41+32 > P11
32
K90
K23
K26
P26
FingerTabelle P26
0 26+1 > P41
1 26+2 > P41
2 26+4 > P41
3 26+8 > P41
4 26+16 > P51
5 26+32 > P61

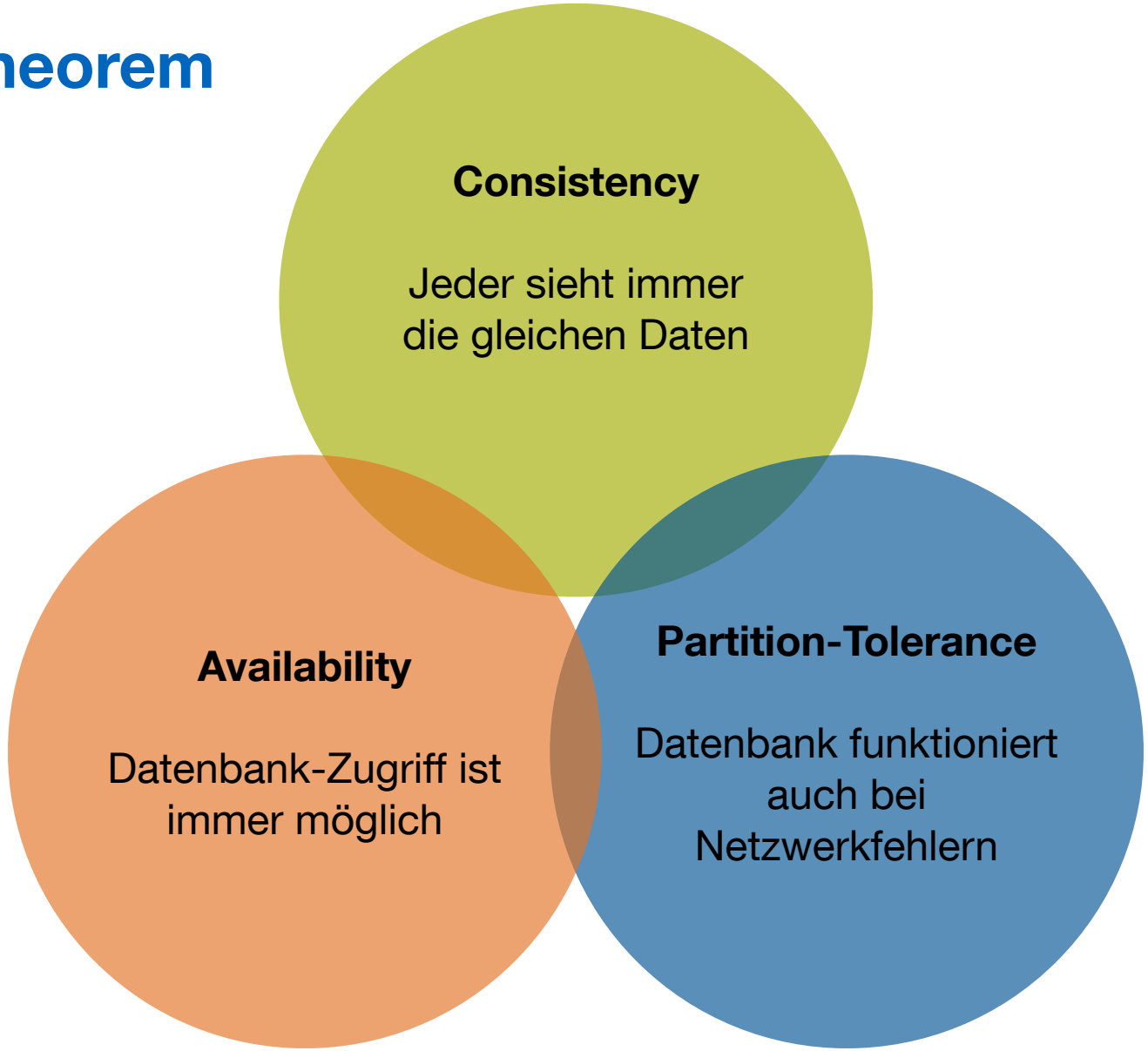
Aufgabe 4


K57
K61
P61
FingerTabelle P61
0 61+1 > P3
1 61+2 > P3
2 61+4 > P3
3 61+8 > P11
4 61+16 > P14
5 61+32 > P41
FingerTabelle P3
0 3+1 > P11
1 3+2 > P11
2 3+4 > P11
3 3+8 > P11
4 3+16 > P26
5 3+32 > P41
Suche
K57
K57
K61
P61
64
0
P3
FingerTabelle P56
0: 56+1 => P61
1 56+2 > P61
2 56+4 > P61
3 56+8 > P3
4 56+16 > P11
5 56+32 > P26
P56
FingerTabelle P11
0 11+1 > P14
1 11+2 > P14
2 11+4 > P26
3 11+8 > P26
4 11+16 > P41
5: 11+32 => P51
P11
P51
FingerTabelle P51
0 51+1 > P56
1 51+2 > P56
2: 51+4 => P56
3 51+8 > P61
4 51+16 > P3
5 51+32 > P26
K12
K14
P14
FingerTabelle P14
0 14+1 > P26
1 14+2 > P26
2 14+4 > P26
3 14+8 > P26
4 14+16 > P41
5 14+32 > P51
P26 ist für alle Schlüssel des Bereichs (14,26] verantwortlich
K39
K31
P41
FingerTabelle P41
0 41+1 > P51
1 41+2 > P51
2 41+4 > P51
3 41+8 > P51
4 41+16 > P61
5 41+32 > P11
32
K90
K23
K26
P26
FingerTabelle P26
0 26+1 > P41
1 26+2 > P41
2 26+4 > P41
3 26+8 > P41
4 26+16 > P51
5 26+32 > P61

Aufgabe 5

Skizzieren Sie die Vorgehensweise beim Hinzufügen eines neuen Peers im Chord Netzwerk. Als Beispiel nehmen Sie die Hinzunahme eines Peers P33 in dem Beispiel-Netzwerk aus Abbildung 2.

CAP Theorem


Consistency
Jeder sieht immer die gleichen Daten
Availability
Datenbank-Zugriff ist immer möglich
Partition-Tolerance
Datenbank funktioniert auch bei Netzwerkfehlern

Aufgabe 6

Zum CAP-Theorem hieß es in der Vorlesung, dass in verteilten Systemen nur zwei der drei "Wünsche" (Konsistenz, Verfügbarkeit und Partitionstoleranz) gleichzeitig erfüllbar sind.

Welche der drei Kombinationen CA, CP, und AP sind jedoch sehr ähnlich?

Fragen?