

$b = 3$ , NextToSplit = 0,  $d = 3$

000	8	24	32	
001	17			
010	2	26		
011	43			
100				
101	13	29	45	
110	38			
111				

Einfügen mittels  $h_3(x)$ :

$$2 = 000010$$

$$8 = 001000$$

$$13 = 001101$$

$$17 = 010001$$

$$24 = 011000$$

$$29 = 011101$$

$$32 = 100000$$

$$26 = 011010$$

$$38 = 100110$$

$$43 = 101011$$

$$45 = 101101$$

$$53 = 110101 \quad ?$$



Was passiert nun, wenn Bereich überläuft?

$b = 3$ , NextToSplit = 0,  $d = 3$

000	8	24	32	
001	17			
010	2	26		
011	43			
100				
101	13	29	45	
110	38			
111				

Überlaufbereich wird durch Anhängen eines weiteren Blockes mittels einer linearen Liste gebildet.



# Linear Hashing (3)



$b = 3$ , NextToSplit = **1**,  $d = 3$

0000	32			
001	17			
010	2	26		
011	43			
100				
101	13	29	45	
110	38			
111				
1000	8	24		

Vergrößern des Primärbereichs durch Hinzufügen eines weiteren Blockes am Ende der Hashtabelle (Index NextToSplit wird um 1 erhöht)

Erfordert, daß Elemente im gesplitteten Block reorganisiert werden

53			
----	--	--	--

$b = 3$ , NextToSplit = 1,  $d = 3$

0000	32			
001	17			
010	2	26		
011	43			
100				
101	13	29	45	
110	38			
111				
1000	8	24		

$d$  wird erhöht, wenn Splitting für ursprünglichen Primärbereich einmal durchgeführt wurde, dh es gilt:

```
if (NextToSplit ==  $2^d$ )  
{  
   $d++$ ; NextToSplit = 0;  
}
```

Einfügen von Elementen mittels der Funktion  $h_d(x)$ , außer die Funktion führt auf einen Block, der bereits gesplittet wurde, dann  $h_{d+1}(x)$



53			
----	--	--	--

# Linear Hashing (5)



$b = 3$ , NextToSplit = 1,  $d = 3$

0000	32			
001	17	9		
010	2	26	18	
011	43			
100				
101	13	29	45	
110	38			
111				
1000	8	24		

Einfügen von: 9, 18, 10, 16, 7, 15, 12,  
28, 33, 14, 30, 27, 31, 11, 20, 36



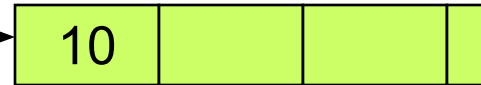
# Linear Hashing (6)



$b = 3$ , NextToSplit = **2**,  $d = 3$

0000	32			
<b>0001</b>	17			
010	2	26	18	
011	43			
100				
101	13	29	45	
110	38			
111				
1000	8	24		
<b>1001</b>	<b>9</b>			

Einfügen von: **9, 18, 10**, 16, 7, 15, 12,  
28, 33, 14, 30, 27, 31, 11, 20, 36



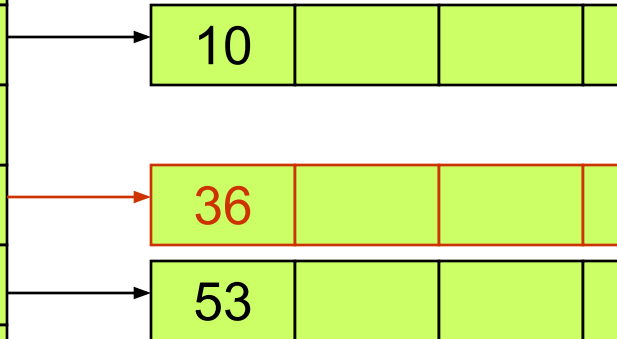
# Linear Hashing (7)



$b = 3$ , NextToSplit = 2,  $d = 3$

Einfügen von: 9, 18, 10, 16, 7, 15, 12,  
28, 33, 14, 30, 27, 31, 11, 20, 36

0000	32	16		
0001	17	33		
010	2	26	18	
011	43	27	11	
100	12	28	20	
101	13	29	45	
110	38	14	30	
111	7	15	31	
1000	8	24		
1001	9			

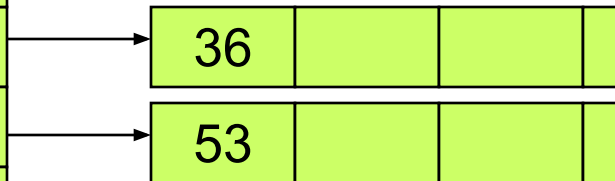


# Linear Hashing (8)



$b = 3$ , NextToSplit = **3**,  $d = 3$

0000	32	16		
0001	17	33		
0010	2	18		
011	43	27	11	
100	12	28	20	
101	13	29	45	
110	38	14	30	
111	7	15	31	
1000	8	24		
1001	9			
1010	26	10		





# Linear Hashing (9)



$b = 3$ , NextToSplit = 3,  $d = 3$

0000	32	16		
0001	17	33		
0010	2	18		
011	43	27	11	
100	12	28	20	
101	13	29	45	
110	38	14	30	
111	7	15	31	
1000	8	24		
1001	9			
1010	26	10		

Suchen von:

14 = 001110

26 = 011010

19 = 010011

36			
53			

Aktuelle  $h_d(x) = h_3(x)$ , dh

14 = 001110

# Linear Hashing (10)



$b = 3$ , NextToSplit = 3,  $d = 3$

0000	32	16		
0001	17	33		
0010	2	18		
011	43	27	11	
100	12	28	20	
101	13	29	45	
110	38	14	30	
111	7	15	31	
1000	8	24		
1001	9			
1010	26	10		

Suchen von:

14 = 001110

26 = 011010

19 = 010011

36			
53			

Aktuelle  $h_d(x) = h_{d+1}(x) = h_4(x)$ ,

(da  $h_d(x) < \text{NextToSplit}$ ) dh

26 = 011010

# Linear Hashing (11)



$b = 3$ , NextToSplit = 3,  $d = 3$

0000	32	16		
0001	17	33		
0010	2	18		
011	43	27	11	
100	12	28	20	
101	13	29	45	
110	38	14	30	
111	7	15	31	
1000	8	24		
1001	9			
1010	26	10		

Suchen von:

14 = 001110

26 = 011010

19 = 010011



36			
53			

Aktuelle  $h_d(x) = h_3(x)$ , dh

19 = 010011

**! ERFOLGLOS !**