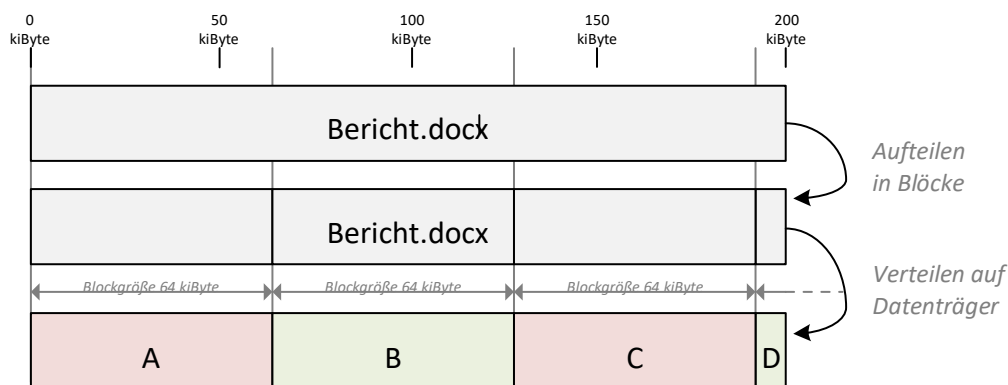


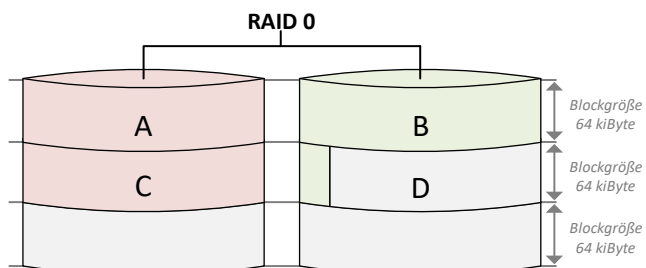
WAS IST EIGENTLICH DIE BLOCKGRÖßE?

Eine wichtige Kenngröße in einem RAID-Verbund ist die *block size / Blockgröße*, die manchmal auch als *stripe size* bezeichnet wird. Obwohl sie selten explizit auftaucht, ist sie ein wichtiger Parameter für die Performance, die der RAID-Verbund erreichen kann. Die Blockgröße beschreibt eine zusammenhängende Größe an Daten, die jeweils *gemeinsam* verarbeitet werden. Bei RAID-Systemen wird meistens eine Blockgröße im Bereich von 64 KiByte bis 512 KiByte verwendet, je nach Hersteller. Die Größe kann manchmal auch vor dem Erstellen des RAID-Verbundes konfiguriert werden.

Was bedeutet das nun konkret? In folgendem Beispiel soll eine Datei *Bericht.docx* in einem RAID0Verbund mit zwei Datenträgern abgelegt werden. Die ist 200 KiByte groß, und die Blockgröße beträgt 64 KiByte. Anhand der Blockgröße wird nun die Aufteilung der Daten bestimmt und die Datei in die vier Blöcke A, B, C und D geteilt, die jeweils eine Größe von 64 KiByte haben:



Auf den beiden Datenträgern werden die Daten dann folgendermaßen abgelegt:



Es werden also immer Abschnitte in der Größe von 64 KiByte gemeinsam als Block behandelt und zwischen den Datenträgern verteilt. Diese Daten werden auch immer gemeinsam geschrieben und gelesen. Dabei können sich auch mehrere Dateien in einem Block befinden, und umgekehrt kann auch eine kleinere Datei zufällig in zwei Blöcken landen.

In unserem Beispiel wurden in Block D erst 8 KiByte verwendet, es sind also 56 KiByte frei. Wird nun eine weitere Datei in der Größe von 60 KiByte gespeichert, landen die ersten 56 KiByte ebenfalls in Block D, und die verbleibenden 4 KiByte in einem neuen Block E.

Die Blockgröße beeinflusst also, ab welcher Datenmenge die Daten auf mehrere Festplatten gespeichert werden.



RAID-SIMULATION

In dieser Übung werden Sie die verschiedenen RAID-Modi simulieren, in dem Sie sich anschauen, wie genau die Daten auf den einzelnen Datenträgern verteilt werden und wie die Daten bei einer defekten Festplatte wieder gelesen werden können.

AUFGABE 1 – RAID-VERBÜNDE OHNE PARITÄT

In Ihrem Analyse-Labor finden Sie verschiedene RAID-Verbünde vor, die analysiert werden müssen. Die Blockgröße beträgt in unserer Simulation immer zwei Bytes. Dabei entspricht ein Buchstabe immer genau einem Byte, das Symbol `_` steht für ein Leerzeichen.

1. **Bestimmen Sie** jeweils den RAID-Modus, der verwendet wurden
2. **Ergänzen Sie** die weitere Verteilung der Daten auf die Datenträger.
3. Einige Datenträger sind leider defekt, sie sind mit `markiert`. **Geben Sie** an, wie die Daten ausgelesen aussehen, **setzen Sie** für verlorene Daten das Symbol `?` ein.

Inhalt a): [Ha|ll|o_|We|lt|!_|Ic|h_|bi|n_|ei|ne_|D|at|ei|!]

RAID: 0

	Datenträger 1	Datenträger 2	Datenträger 3	Datenträger 4
Block 1	Ha	ll	o_	We
Block 2	lt	!_	Ic	h_
Block 3	bi	n_	ei	ne
Block 4	_ D	at	ei	!

Inhalt b): [Ei|ne|_w|ic|ht|ig|e_|In|fo|r|m|at|i|o|n|.]

RAID: 5

	1	2	3	4	5	6
Block 1	Ei	Ei	ne	ne	_w	_w
Block 2	ic	ic	ht	ht	ig	ig
Block 3	e_	e_	In	In	fo	fo



Name:

Datum:

Klasse:

Blatt Nr.: 3/7

Lfd. Nr.:

Block 4	rm	rm	at	at	io	io
Block 5	n.	n.				

Inhalt c): [Hi|er|_s|te|ht|_w|as.]

RAID: 1

	Datenträger 1	Datenträger 2	Datenträger 3	Datenträger 4
Block 1	Hi	er	Hi	er
Block 2	_s	te	_s	te
Block 3	ht	_w	ht	_w
Block 4	as	.	as	.

Inhalt d): [Gu|te|n_|Ta|g.]

RAID:

	Datenträger 1	Datenträger 2	Datenträger 3	Datenträger 4
Block 1	Gu	Gu	Gu	Gu
Block 2	te	te	te	te
Block 3	n_	n_	n_	n_
Block 4	Ta	Ta	Ta	Ta
Block 5	g.	g.	g.	g.

Inhalt e): [Ha|ll|o_|We|lt|!_|Ic|h_|bi|n_|ei|ne_|D|at|ei|!]

RAID:

	Datenträger 1	Datenträger 2	Datenträger 3	Datenträger 4
Block 1	Ha	!_	ei	!



Block 2	ll	lt	n_	ei
Block 3	o_	We	bi	at
Block 4		Ic	h_	
Block 5				

AUFGABE 2 – RAID-VERBÜNDE MIT PARITÄT

Für die Berechnung der Prüfsumme müssen Sie die ASCII-Zeichen in Binärzahlen umrechnen. Die Daten sind im ISO-8859-1-Format kodiert worden, eine entsprechende ASCII-Tabelle finden Sie in Ihrem ITHandbuch oder im Anhang zu dieser Aufgabe.

Bei **RAID 5** werden für die Paritätsdaten jeweils die anderen $n - 1$ Datenblöcke mit der XOR-Funktion verknüpft. Um die Parität zu berechnen, wandeln Sie also zuerst die ASCII-Zeichen entsprechend der ASCII-Tabelle in Binärschreibweise um und verknüpfen die Binärzahlen dann bitweise mit der XOR-Funktion. Um einen fehlenden Datenblock zu berechnen, verknüpfen Sie die verbliebenen Datenblöcke und die Parität ebenfalls mit der XOR-Funktion – dann ergibt sich der fehlende Datenblock.

Bei **RAID 6** werden die Paritätsdaten p genauso wie bei RAID 5 berechnet. Die Paritätsdaten q werden mit einem Reed-Solomon-Code berechnet. Reed-Solomon-Codes werden zum Beispiel auch bei CD's und DVD's zur Fehlerkorrektur verwendet, z.B. um einfache Kratzer „auszugleichen“. Die Berechnung ist mathematisch aufwändiger, und zum Beispiel im Tutorial „A Tutorial on Reed-Solomon Coding for Fault-Tolerance in RAID-like Systems“ [<http://web.eecs.utk.edu/~jplank/plank/papers/CS-96-332.pdf>] beschrieben.

1. **Bestimmen Sie** jeweils den RAID-Modus, der verwendet wurde.
2. **Ergänzen Sie** die weitere Verteilung der Daten auf die Datenträger und **berechnen Sie** die fehlenden Paritäten.

Inhalt 2a):

[Hi|er|_|s|te|ht|_|e|tw|as|.]

RAID: 5

	Datenträger 1	Datenträger 2	Datenträger 3	Datenträger 4
Block 1	Hi	er	_ s	[Parität]
Hex:	0x4869	0x6572	0x2073	0x0D68



Name:

Datum:

Klasse:

Blatt Nr.: 5/7

Lfd. Nr.:

Binär	0100 1000 0110 1001	0110 0101 0111 0010	0010 0000 0111 0011	0000 1101 0110 0000
Block 2	11		[Parität]	u
Hex:	0x6C6C	0x0020	0x5376	0x0D75
Binär	01101100 01101100	00100000 00000000	01010011 01110110	00001101 01110101
Block 3		[Parität]		
Hex:				
Binär				

Inhalt 2b):

[Ha|11|o_|Du]

RAID: 6

	Datenträger 1	Datenträger 2	Datenträger 3	Datenträger 4
Block 1	Ha	11	[Parität p]	[Parität q]
Hex:	0x4861	0x6C6C	0x240D	[Reed-Solomon]
Binär	0100 1000 0110 0001	0110 1100 0110 1100	0010 0100 0000 0000	[Reed-Solomon]
Block 2	Du	[Parität p]	[Parität q]	
Hex:	0x6F20		[Reed-Solomon]	0x4475
Binär	0110 1111 0010 0000	01010011 01110110	[Reed-Solomon]	0100 0100 0111 0101

3. Einige Datenträger sind leider defekt, sie sind mit **?** markiert. **Stellen** Sie die Daten (wenn möglich) **wieder her**.

Setzen Sie für verlorene Daten das Symbol **[?]** ein.

Inhalt 2c):

[| | | |]

RAID: 6



Name:

Datum:

Klasse:

Blatt Nr.: 6/7

Lfd. Nr.:

	Datenträger 1	Datenträger 2	Datenträger 3	Datenträger 4
Block 1				[Parität]
Hex:	0x2020202 0	0x2020202 0	0x2020202 0	
Binär	0101 0011 0111 0101	0110 0101 0110 1001	0111 0010 0110 0111	0101 0001 0111 0000
Block 2			[Parität]	
Hex:	0x2020202 0	0x2020202 0	0110 1101 0010 0000	0x2020202 0
Binär	0110 0101 0110 1001	0110 0101 0110 1001		0110 1101 00100001



Name:

Datum:

Klasse:

Blatt Nr.: 7/7

Lfd. Nr.:

AUSZUG AUS DER ASCII-TABELLE / ISO-8859-1

Dez	Hex
...	...
32	20
33	21
...	...
46	2E
...	...
48	30
49	31
50	32
51	33
52	34
53	35
54	36
55	37
56	38
57	39
58	3A
...	...
63	3F
...	...

Dez	Hex	
...
65	41	A
66	42	B
67	43	C
68	44	D
69	45	E
70	46	F
71	47	G
72	48	H
73	49	I
74	4A	J
75	4B	K
76	4C	L
77	4D	M
78	4E	N
79	4F	O
80	50	P
81	51	Q
82	52	R
83	53	S
84	54	T
85	55	U
86	56	V
87	57	W
88	58	X
89	59	Y
90	5A	Z

Dez	Hex	
...
196	C4	Ä
...
214	D6	Ö
...
220	DC	Ü
...
228	E4	ä
...
246	F6	ö
...
252	FC	ü
...

