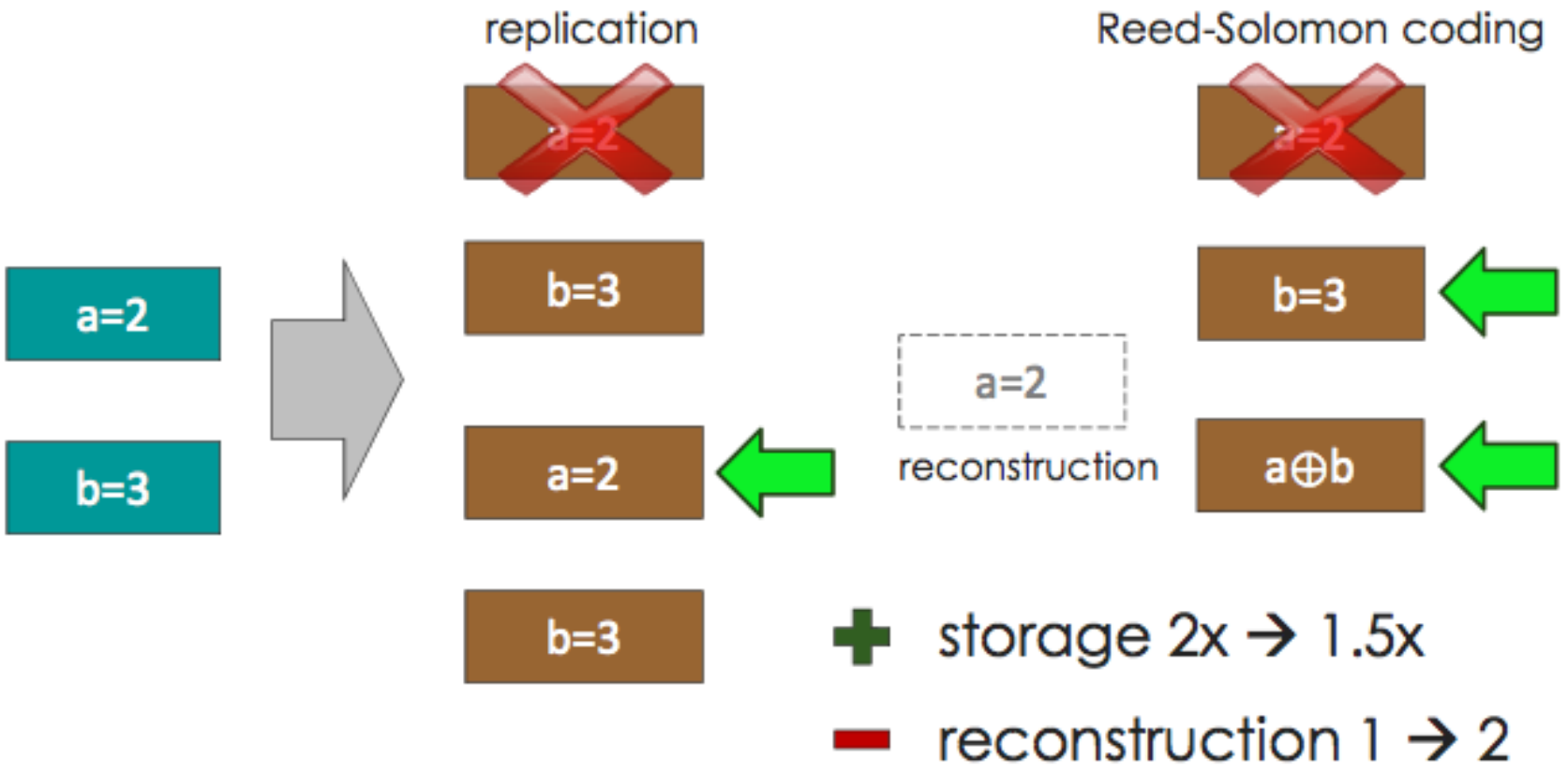
Cloud Computing: Storage as a Service

Vijay Dialani, PhD
Boise State University
vijaydialani@boisestate.edu

Recap

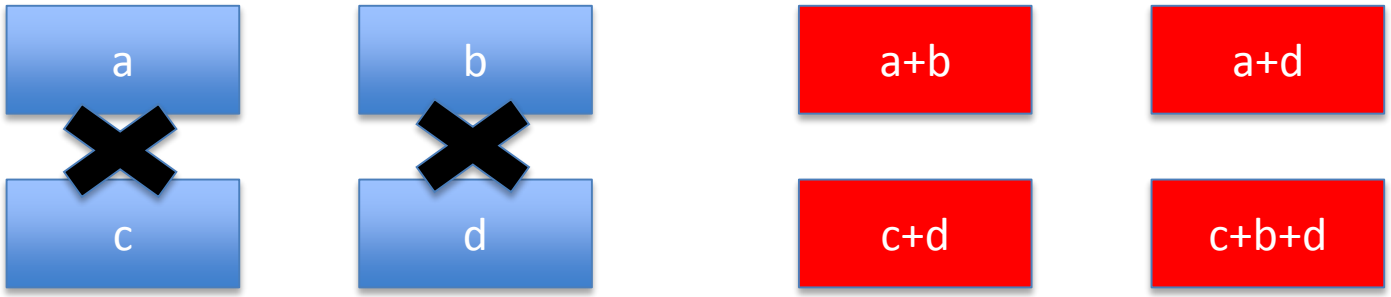
1. Parity codes allow for error detection and error correction

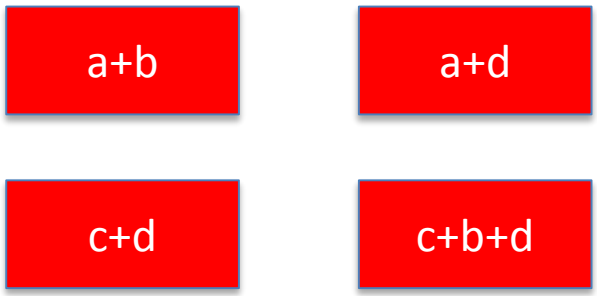


Recap

1. Recovering from multiple failures require exchanges of messages and bounded reconstruction cost is preferred.

r redundancies → correct **r** erasures





1st rebuilt block = (c+b+d)-(c+d) = b
2nd rebuilt block = (a+b)-b = a
3rd rebuilt block = (a+d) – a = d
4th rebuilt block = (c+d)-d =c

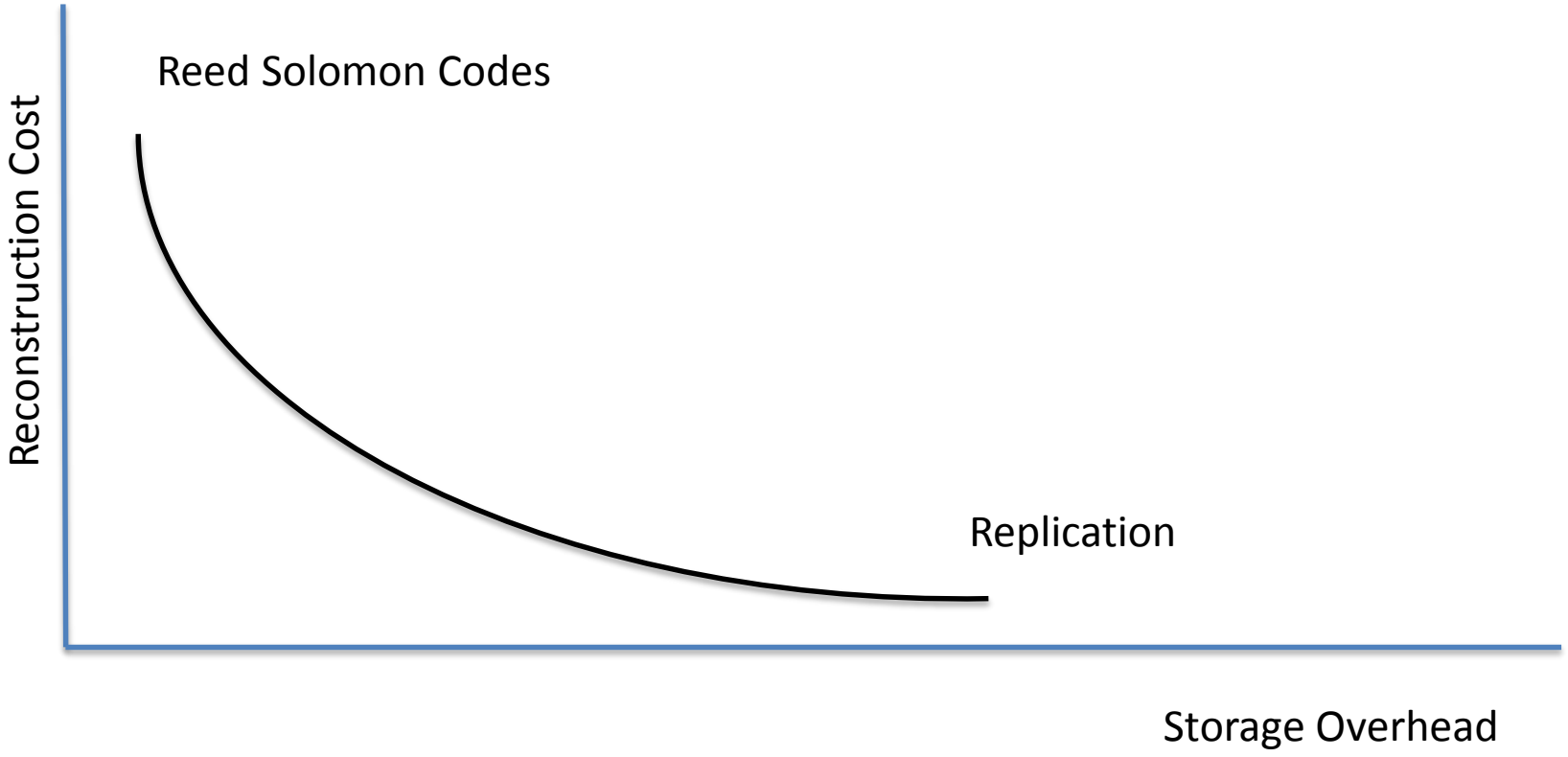
ZigZag Codes

	Node-0	Node-1	Node-2	Row sum	ZigZag sum
0	m	a	w	m+a+w	2m+c+2z
1	n	b	x	n+b+x	2n+d+y
2	p	c	y	p+c+y	p+a+x
3	q	d	z	q+d+z	q+b+2w

Access v/s Bandwidth
For M rows

	Optimal Access	Optimal Bandwidth
Optimal Update	M+1	M+1
Non-optimal update	$3m<k<2^{2m}$ **(More)**	2M **(Less)**

Storage


Reed Solomon Codes
Reconstruction Cost
Replication
Storage Overhead

Reed Solomon Codes

For an indepth mathematical treatment follow the articles:

1. http://tools.ietf.org/html/rfc5510#page-17

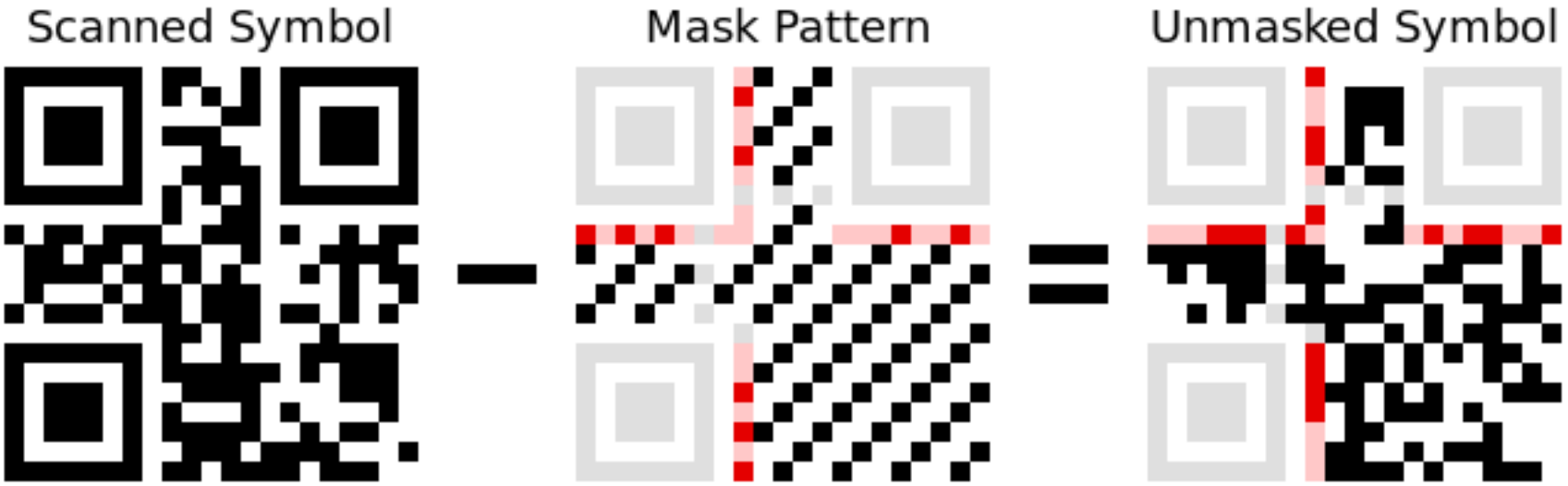
1. http://www.mth.msu.edu/~jhall/classes/codenotes/grs.pdf

Lets look at analogous QR codes

QR codes

1. Designed to reduce the number of errors in reading of digital barcodes
2. Contains both data and error correction information
3. It is possible to detect and recover from error using QR codes
4. Ability to recover from multiple failures

What is underneath the pattern?



Input	101101101001011
Mask	^ 101010000010010
Output	000111101011001

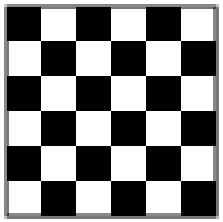
There are two identical copies of the format information, so that the symbol can still be decoded even if it is damaged. The second copy is broken in two pieces and placed around the other two locators, and is also read in a counter-clockwise direction (upwards in the lower-left corner, then left-to-right in the upper-right corner).

QR symbol

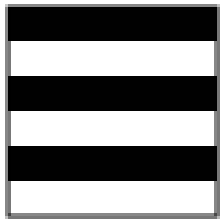
The first two bits of format information give the error correction level used for the message data. A QR symbol this size contains 26 bytes of information. Some of these are used to store the message and some are used for error correction, as shown in the table below. The left-hand column is simply a name given to that level.

Error Correction Level	Level Indicator	Error Correction Bytes	Message Data Bytes
L	01	7	19
M	00	10	16
Q	11	13	13
H	10	17	9

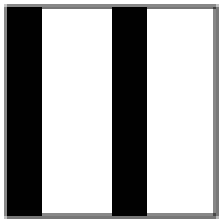
QR Codes



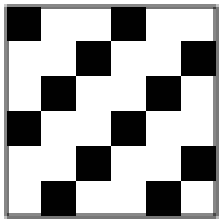
Mask 000
(i + j)%2 = 0



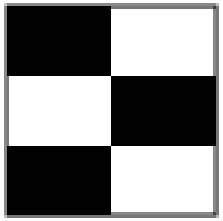
Mask 001
i % 2 = 0



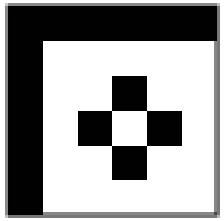
Mask 010
j % 3 = 0



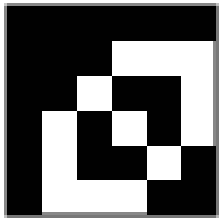
Mask 011
(i + j)%3 = 0



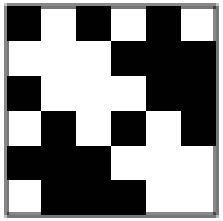
Mask 100
(i/2 + j/3)%2=0



Mask 101
(i*j)%2+(i*j)%3=0



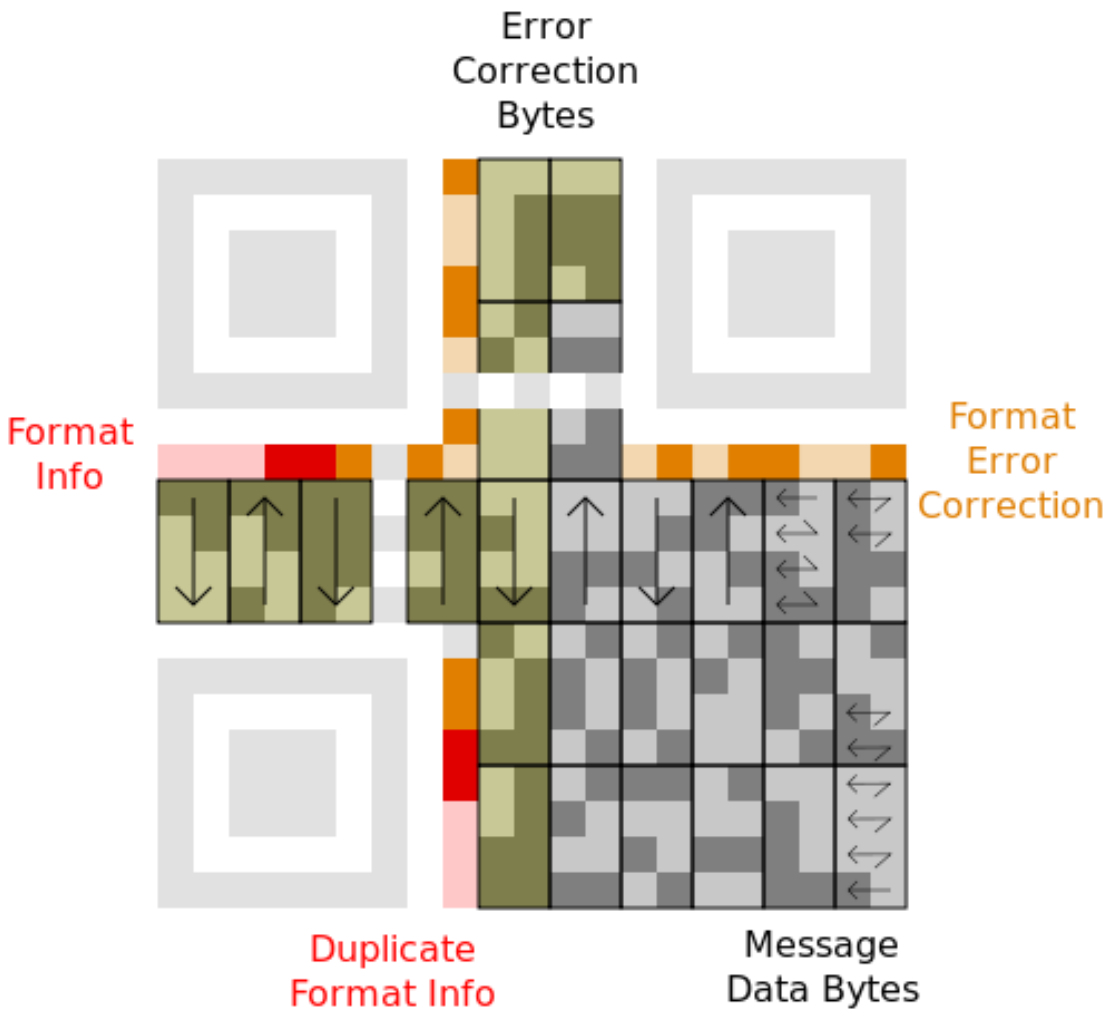
Mask 110
((i*j)%3+i*j)%2=0



Mask 111
((i*j)%3+i+j)%2=0

Each mask is well suited for detection of multiple errors for a range of encoded data

Regions in a QR code



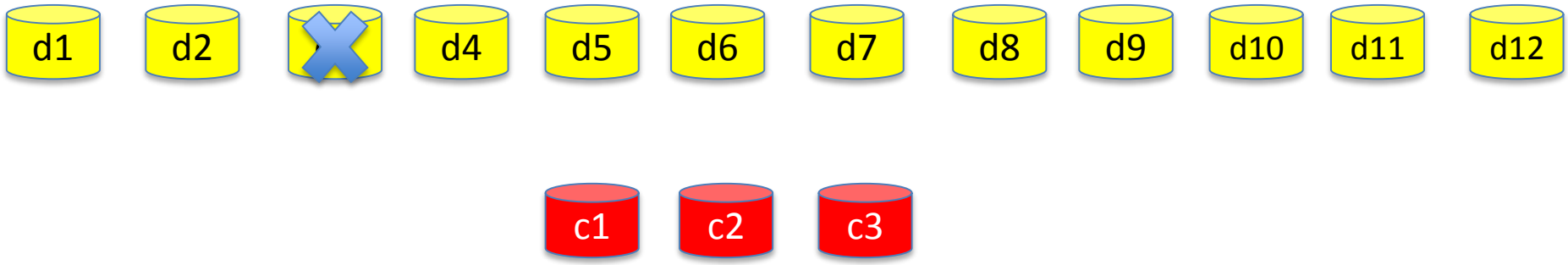
Data bits are read starting from the lower-right corner and moving up the two right-hand columns in a zig-zag pattern.

The first three bytes are 01000000 11010010 01110101. The next two columns are read in a downward direction, so the next byte is 01000111.

Upon reaching the bottom, the two columns after that are read upward. Proceed in this up-and-down fashion all the way to the left side of the symbol

Message data bytes: 40 d2 75 47 76 17 32 06 27 26 96 c6 c6 96 70 ec
Error correction bytes: bc 2a 90 13 6b af ef fd 4b e0

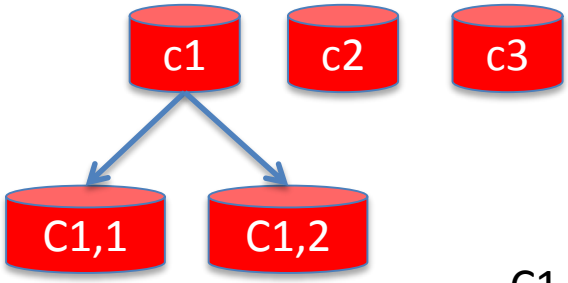
Reed Solomon Codes



Reed Solomon 12 + 3

Failure Recovery requires: 11 data access from data node + 1 parity = 12

Pyramid Codes

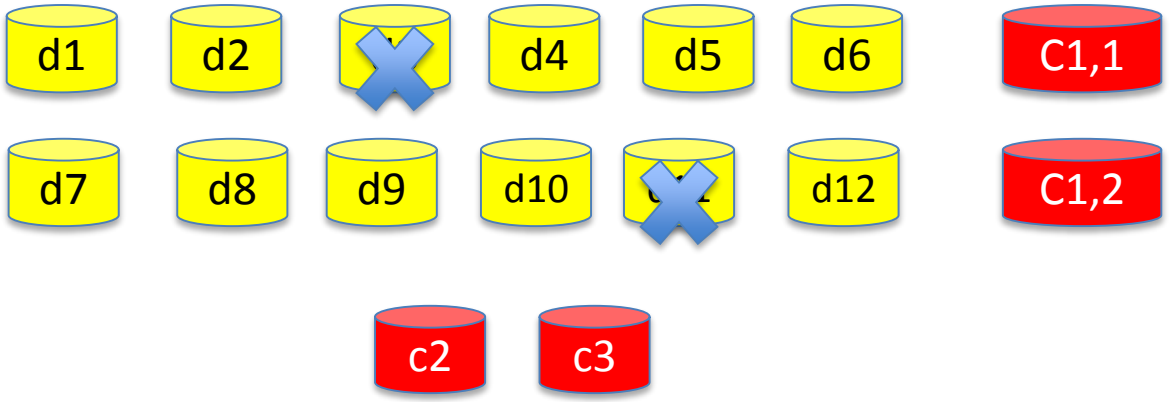


C1,1 : {d1,d2,….,d6}
C1,2 : {d7,d8,….,d12}

Parity construction costs remain the same, so storage performance is unaffected
The recovery access reduces to half (6+1).

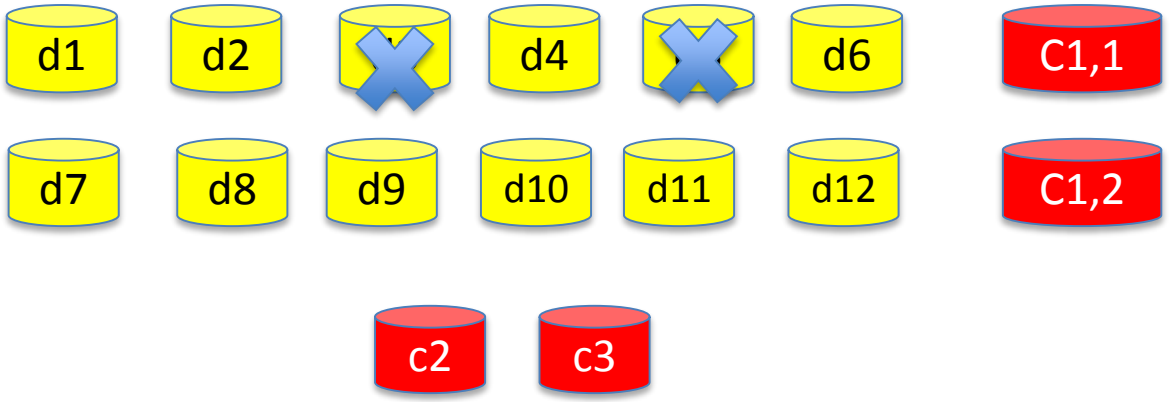
Storage overhead is the same (16/12) = 1.33, greater than 15/12 = 1.25 in previous case

Pyramid Code one failure example



Reconstruction cost for a single failure in each row: 6 messages, compared to 12 in previous non-pyramid case

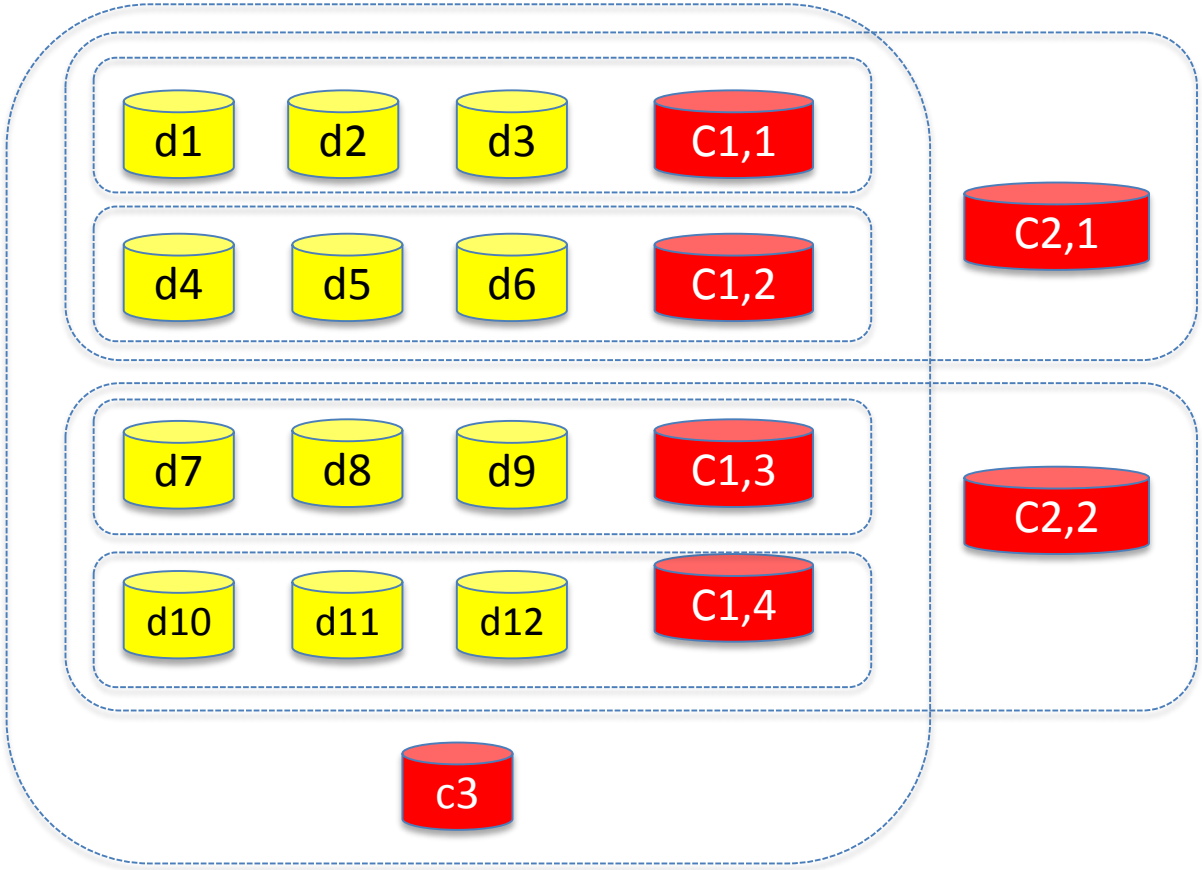
Pyramid Code two failures example



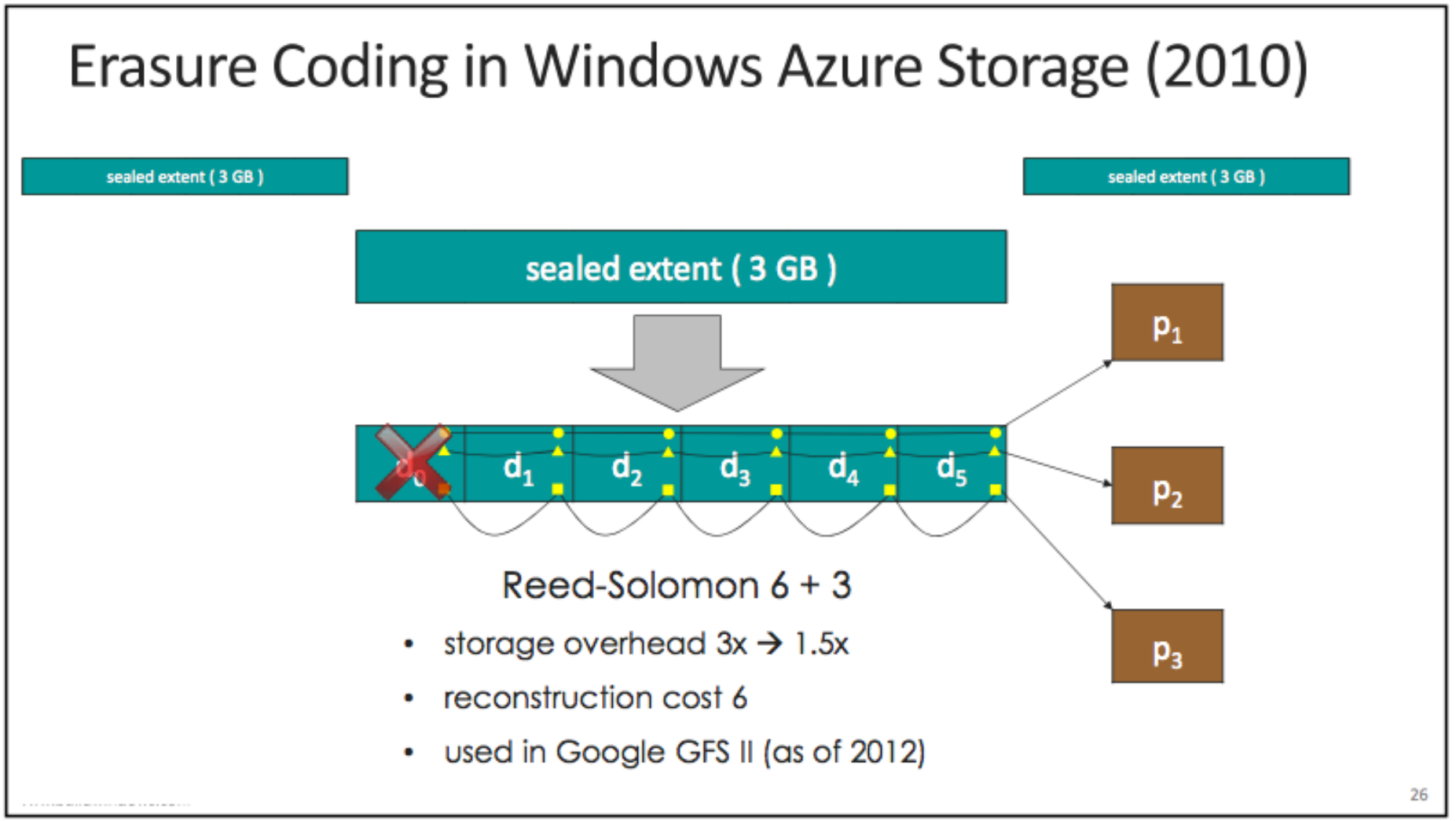
Recover d3 and d5 from c2 and c3 or reconstruct c1 to recover from multiple failures

Number of messages: 6

Pyramid Codes multiple hierarchy



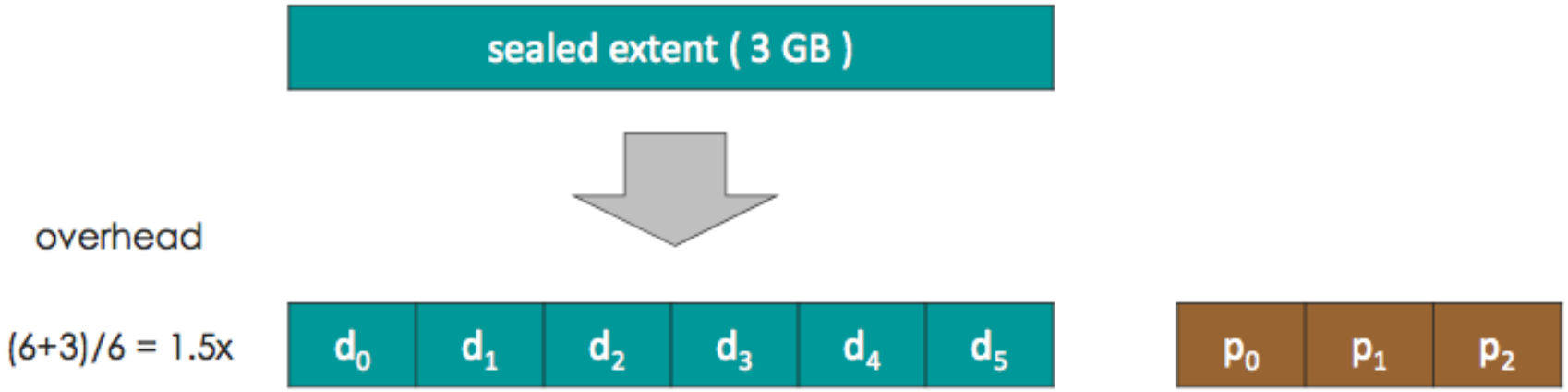
Storage in Azure



Thanks to Cheng Huang, Microsoft Research

Storage in Azure

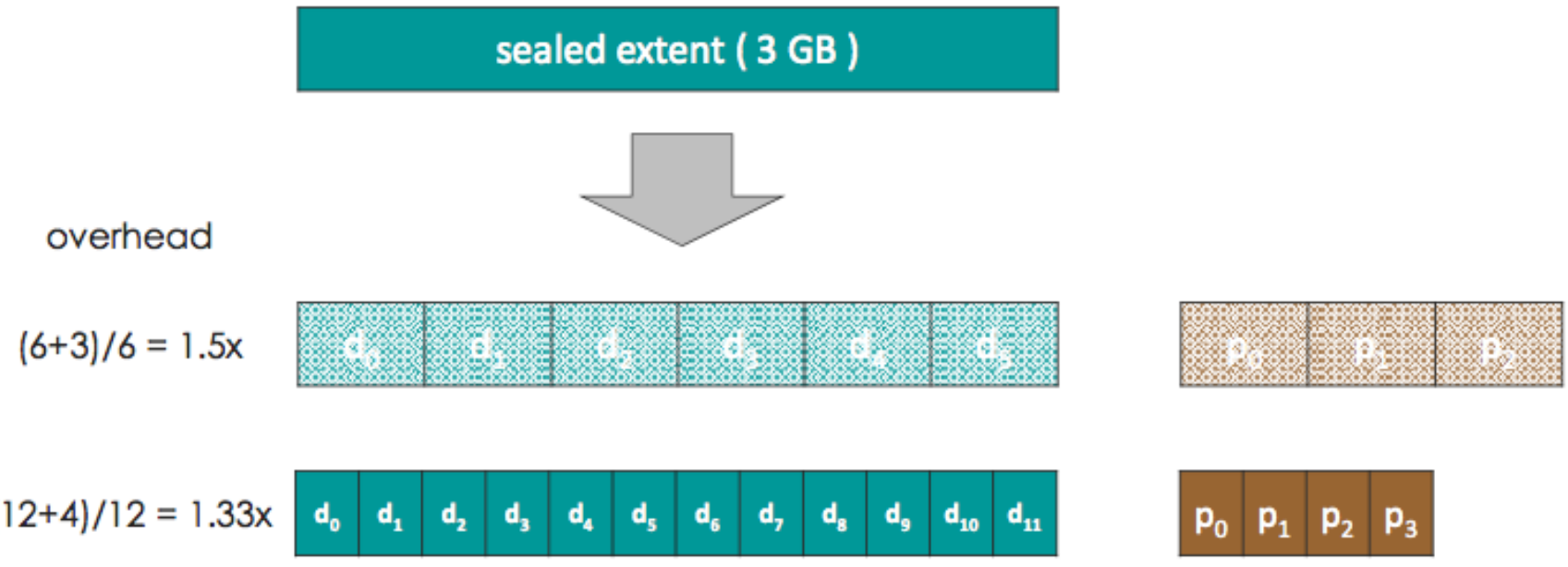
How to Further Reduce Storage Cost?



27

Thanks to Cheng Huang, Microsoft Research

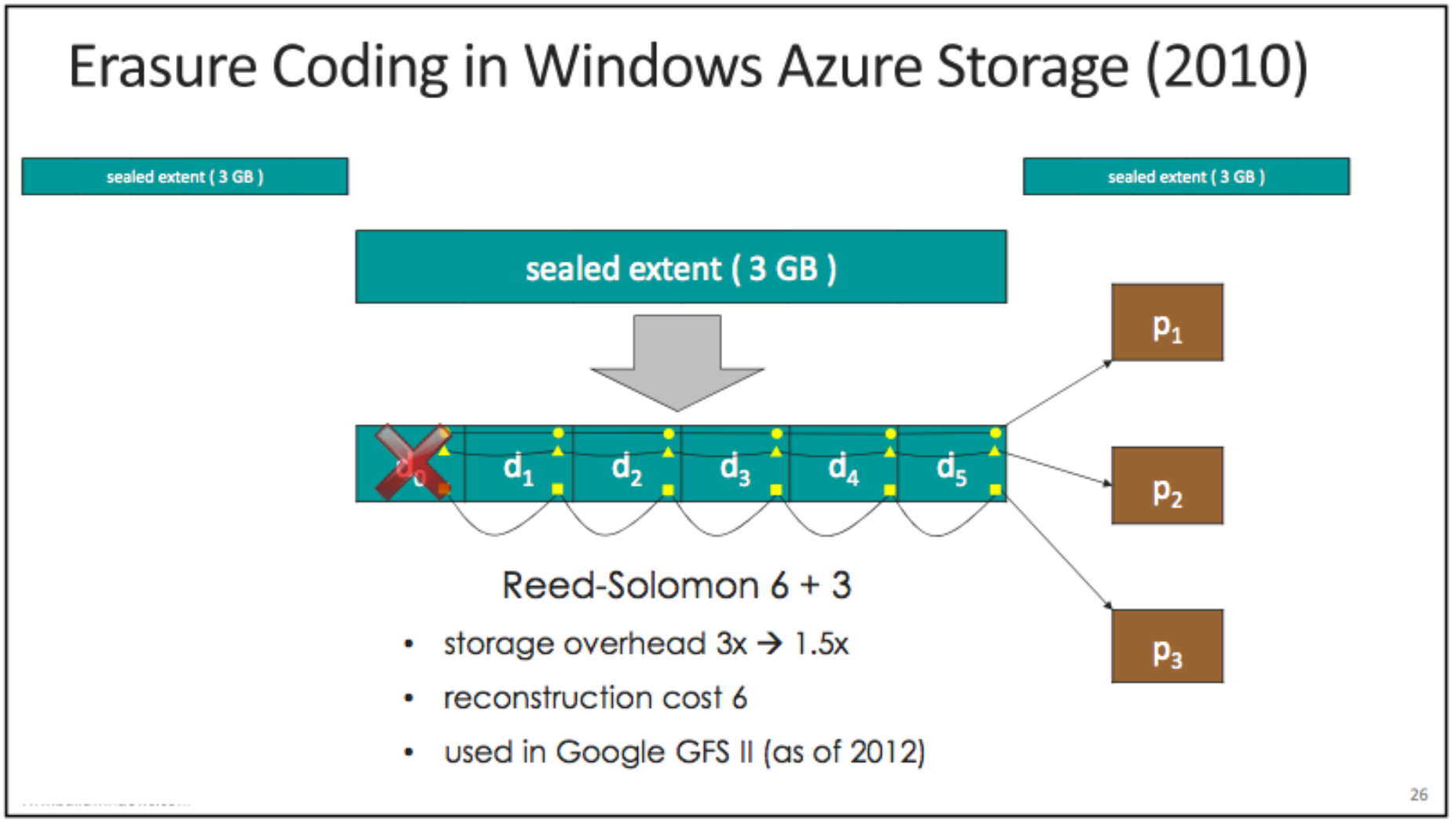
How to Further Reduce Storage Cost?



28

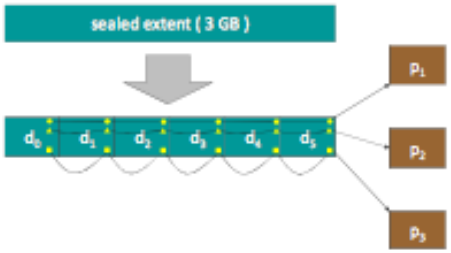
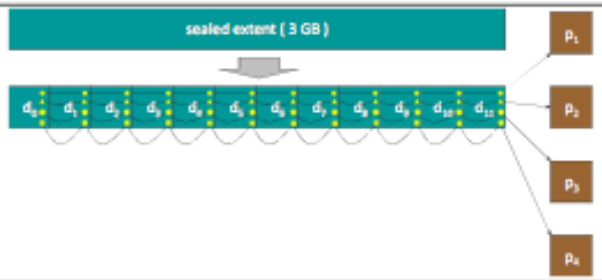
Thanks to Cheng Huang, Microsoft Research

Storage in Azure



Thanks to Cheng Huang, Microsoft Research

Best of Both Worlds?

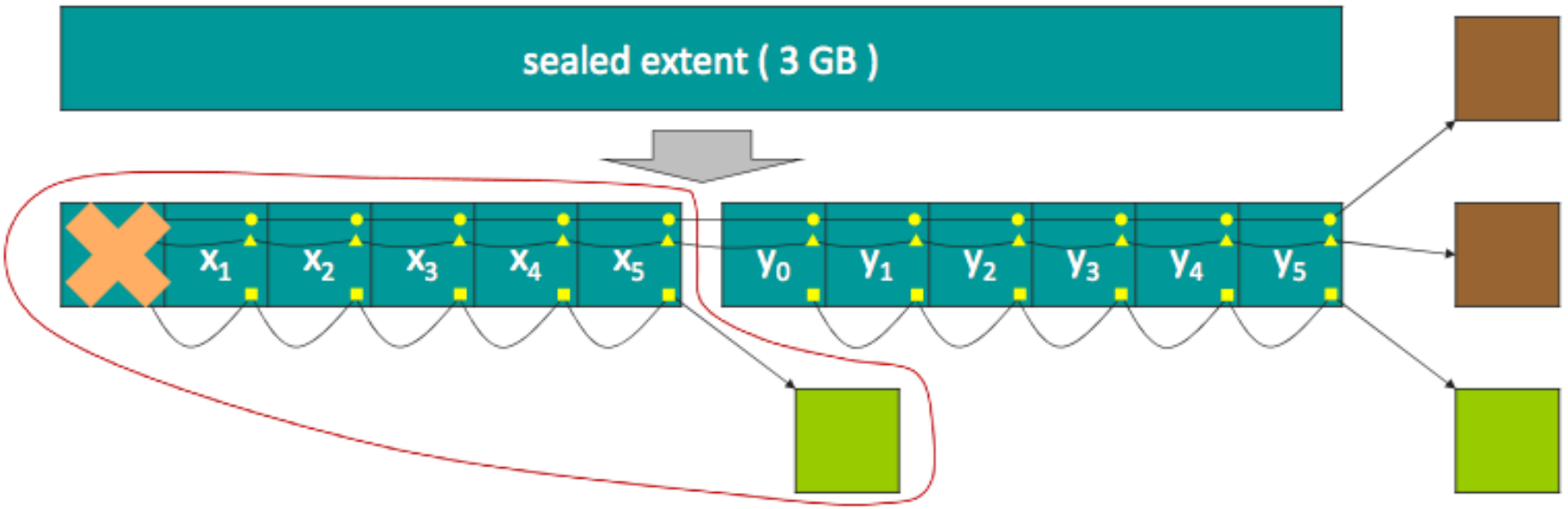
Conventional Reed-Solomon Coding	Storage Overhead	Reconstruction Cost
	1.5x	6 reads
	1.33x	12 reads

LRC

Thanks to Cheng Huang, Microsoft Research

Storage in Azure

Local Reconstruction Code

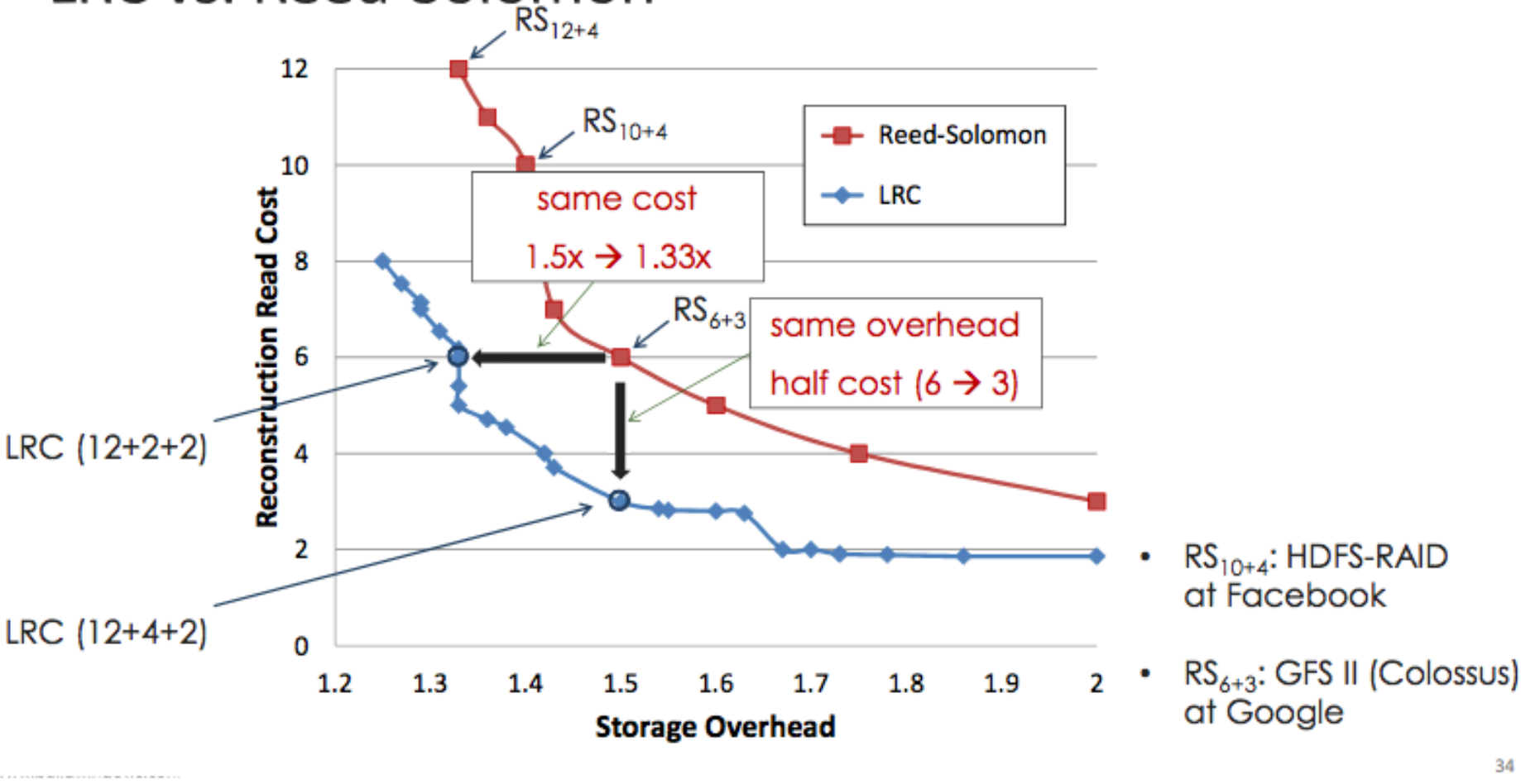


- LRC_{12+2+2}: **12** data fragments, **2** local parities and **2** global parities
  - storage overhead: (12 + 2 + 2) / 12 = 1.33x
- Local parity: reconstruction requires only 6 fragments

31

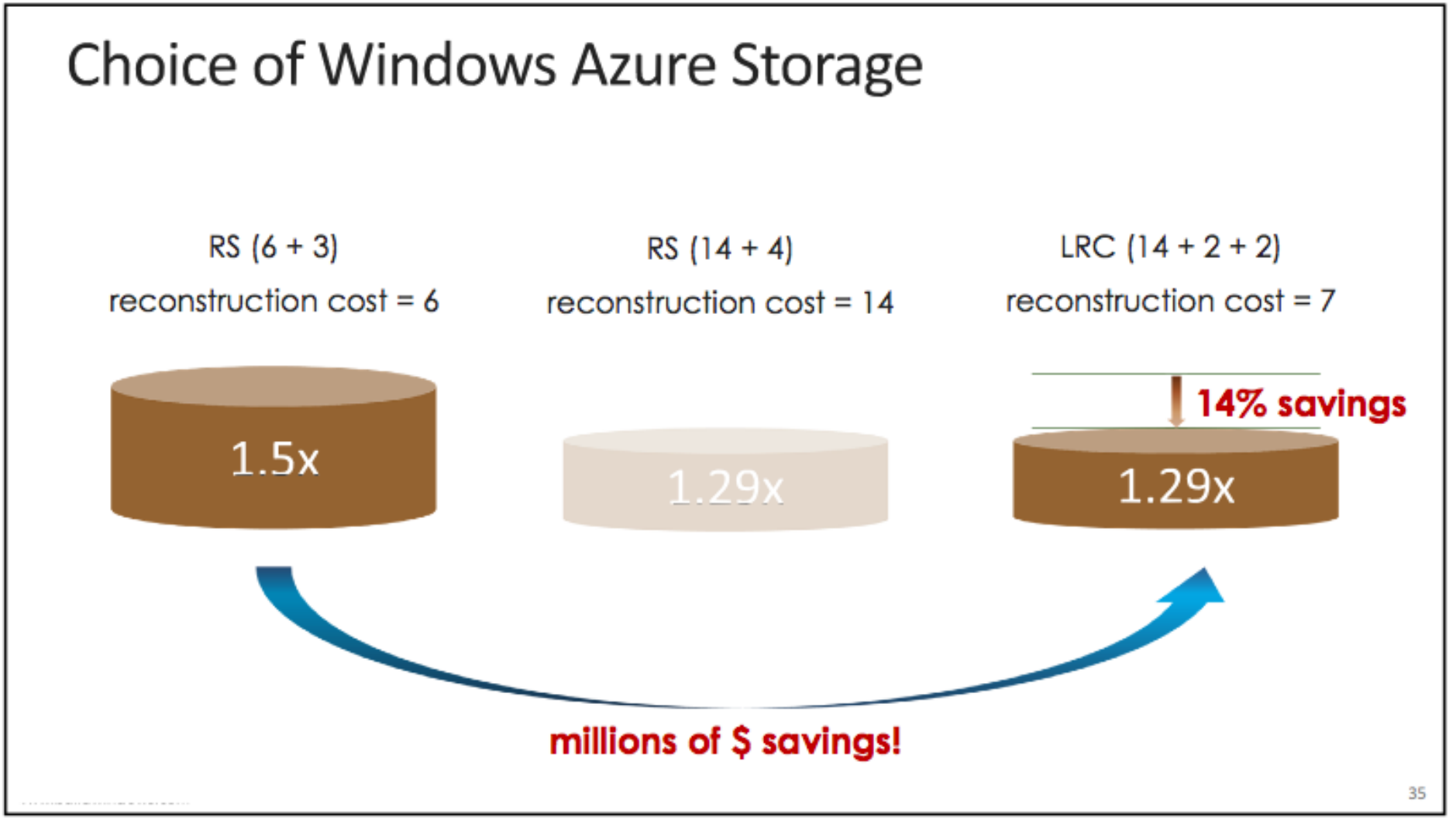
Thanks to Cheng Huang, Microsoft Research

Storage in Azure



Thanks to Cheng Huang, Microsoft Research

Storage in Azure



Thanks to Cheng Huang, Microsoft Research

References

[PDF] **Erasure Coding** in **Windows Azure** Storage.
C Huang, H Simitci, Y Xu, A Ogus, B Calder… - USENIX Annual …, 2012 - usenix.org
Huseyin Simitci and Cheng Huang presented this paper together, with Huseyin starting. In **Windows Azure** Storage, the large scale means failures are the norm rather than the exception. In the context of storage, one question is whether to use replication or **erasure ...**
Cited by 129 Related articles All 24 versions Import into EndNote Save More

[PDF] from usenix.org