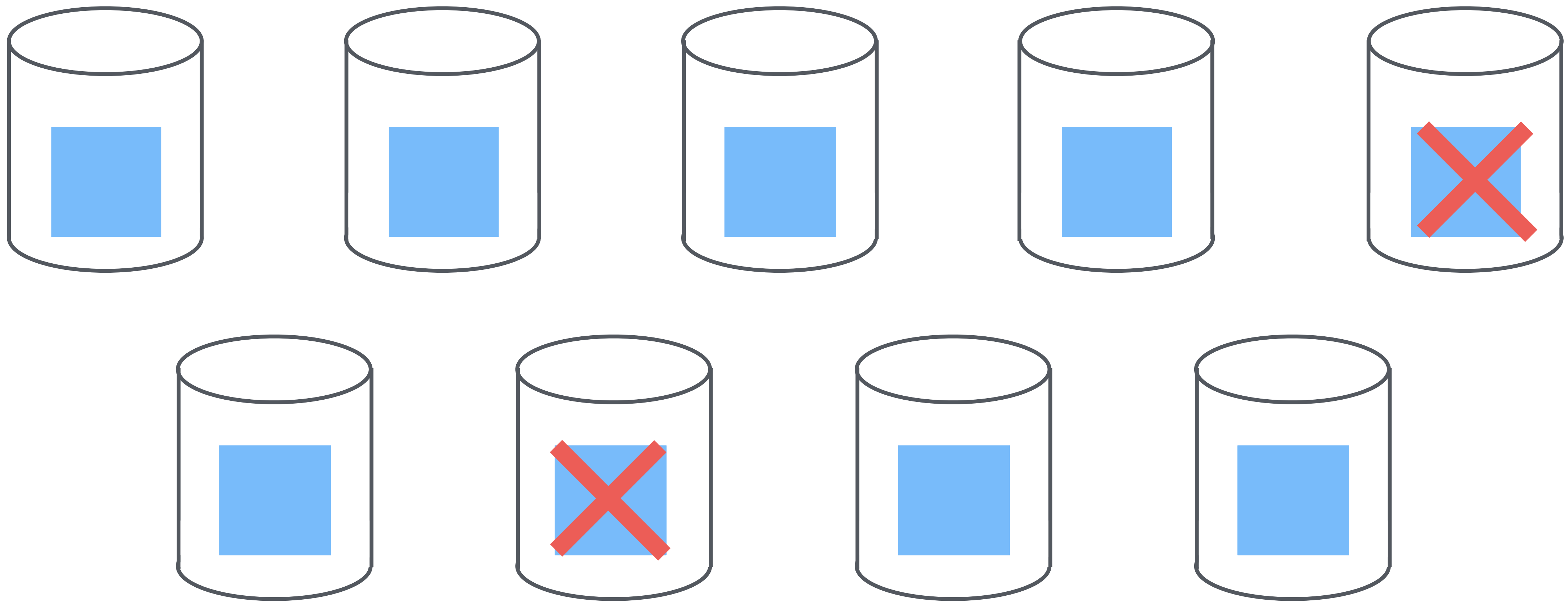
Beehive: Erasure Codes for Fixing Multiple Failures in Distributed Storage Systems

Jun Li, Baochun Li
University of Toronto

HotStorage '15

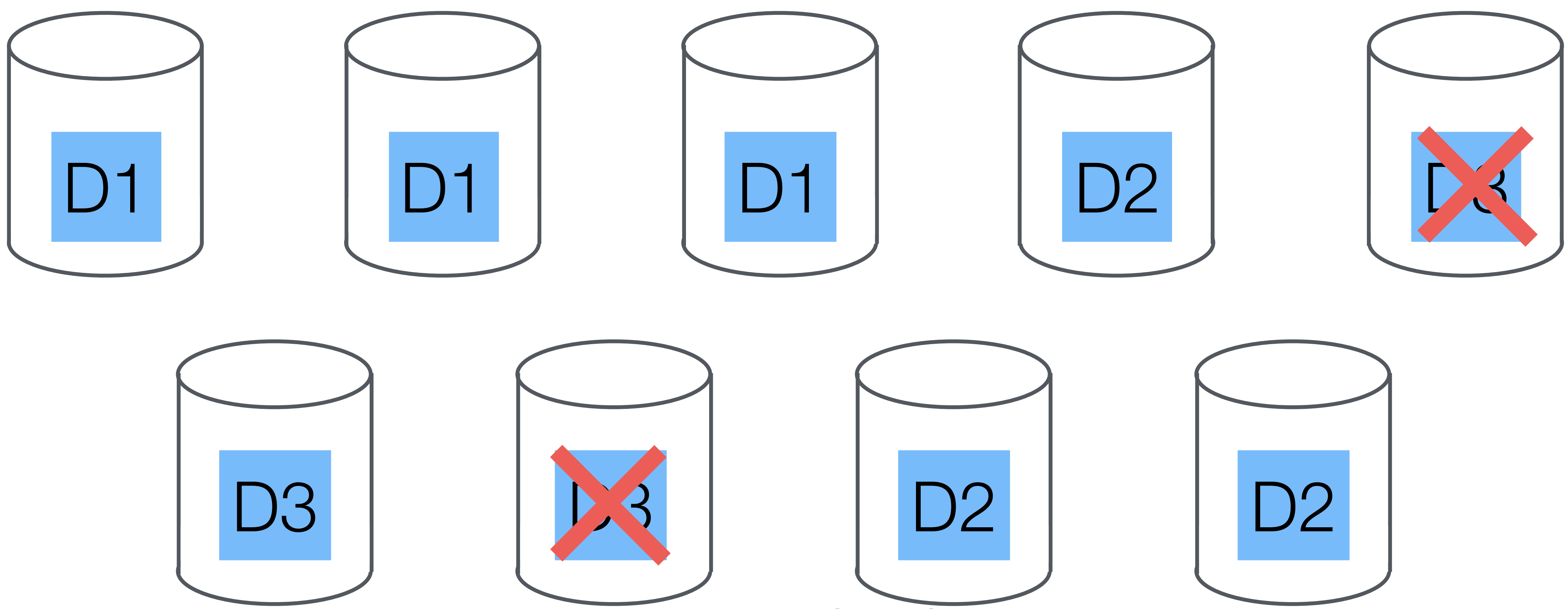
Distributed Storage

- Store a massive amount of data over a large number of commodity servers, such as HDFS
- Servers are subject to frequent failures



Distributed Storage

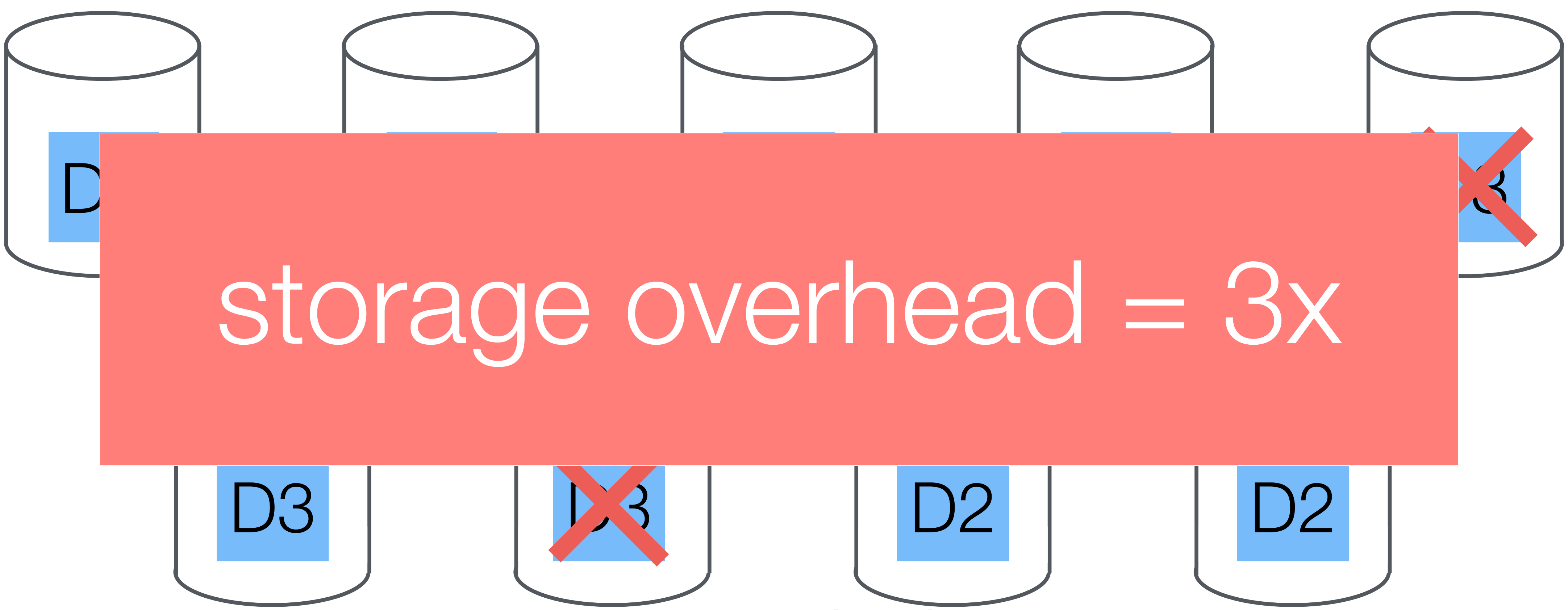
- Store redundant data to ensure data durability and availability regardless of failures
  - replication: store multiple copies on different servers



3-way replication

Distributed Storage

- Store redundant data to ensure data durability and availability regardless of failures
  - replication: store multiple copies on different servers

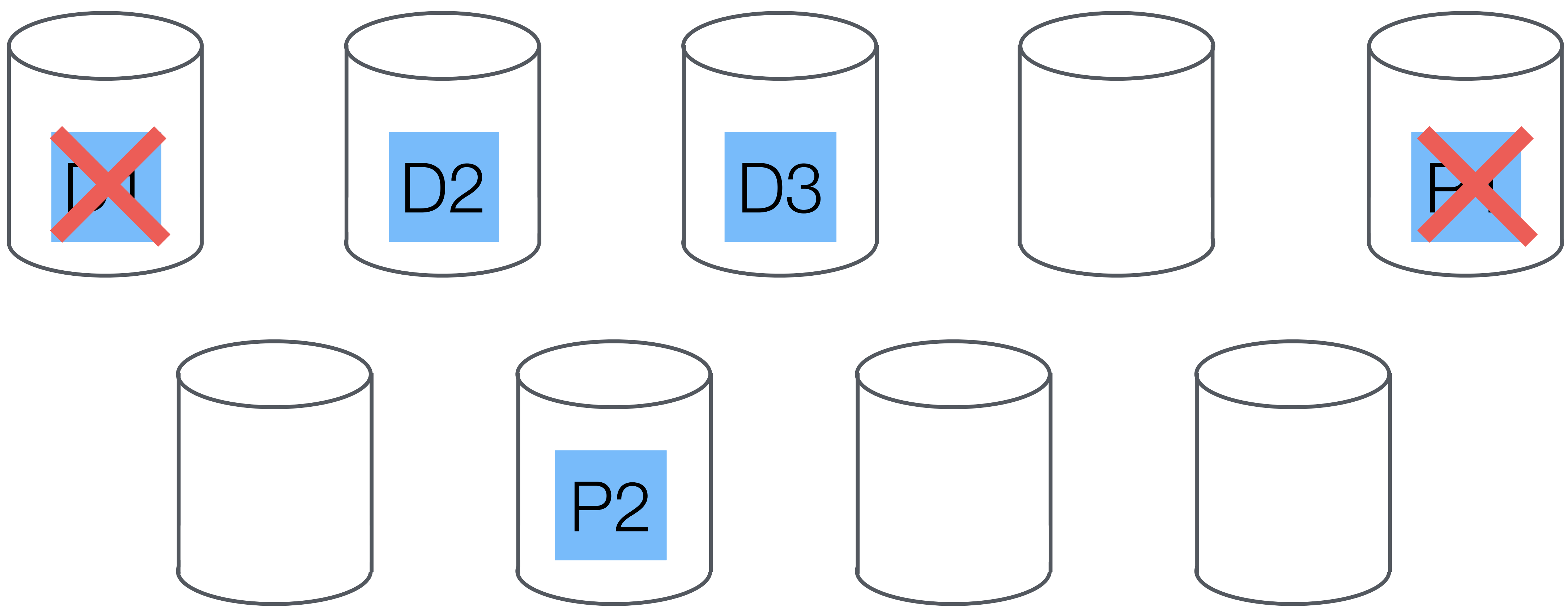


storage overhead = 3x

3-way replication

Erasure Coding

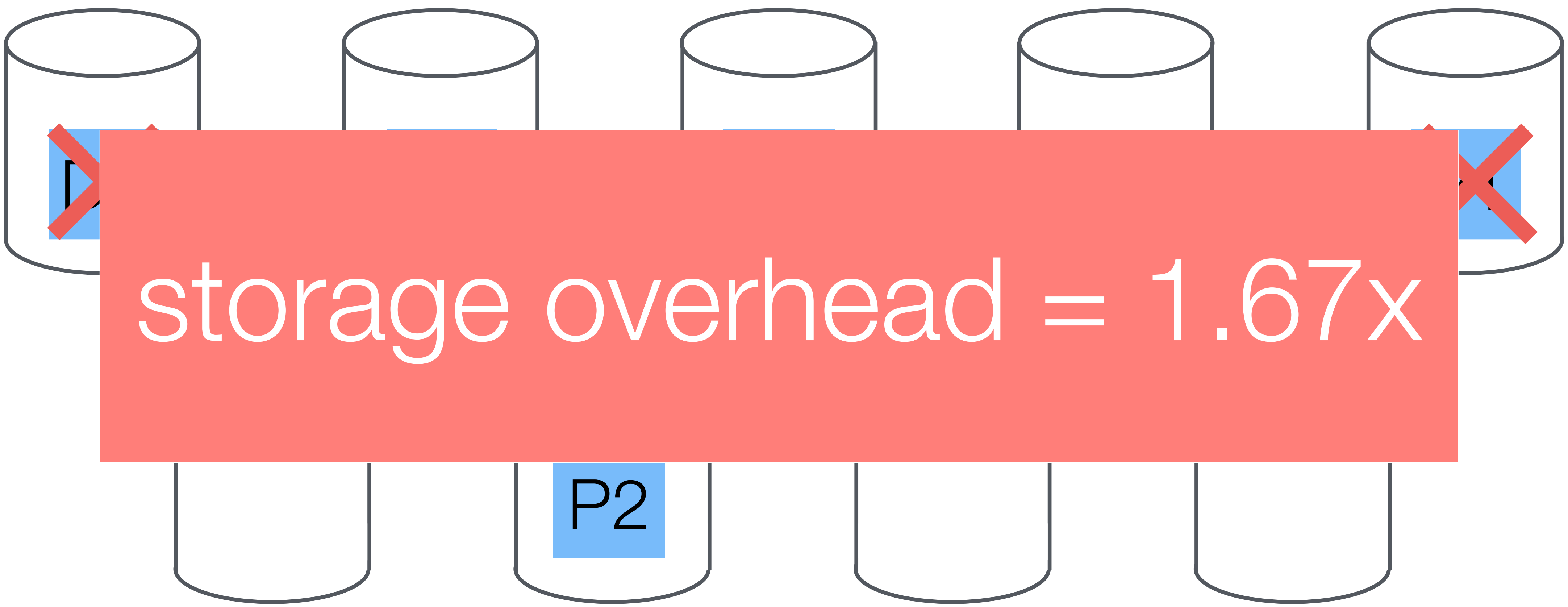
- Use less storage space to tolerate the same number of failures
- (k,r) Reed-Solomon (RS) code
  - compute r parity blocks from k data blocks



(k=3,r=2) RS code

Erasure Coding

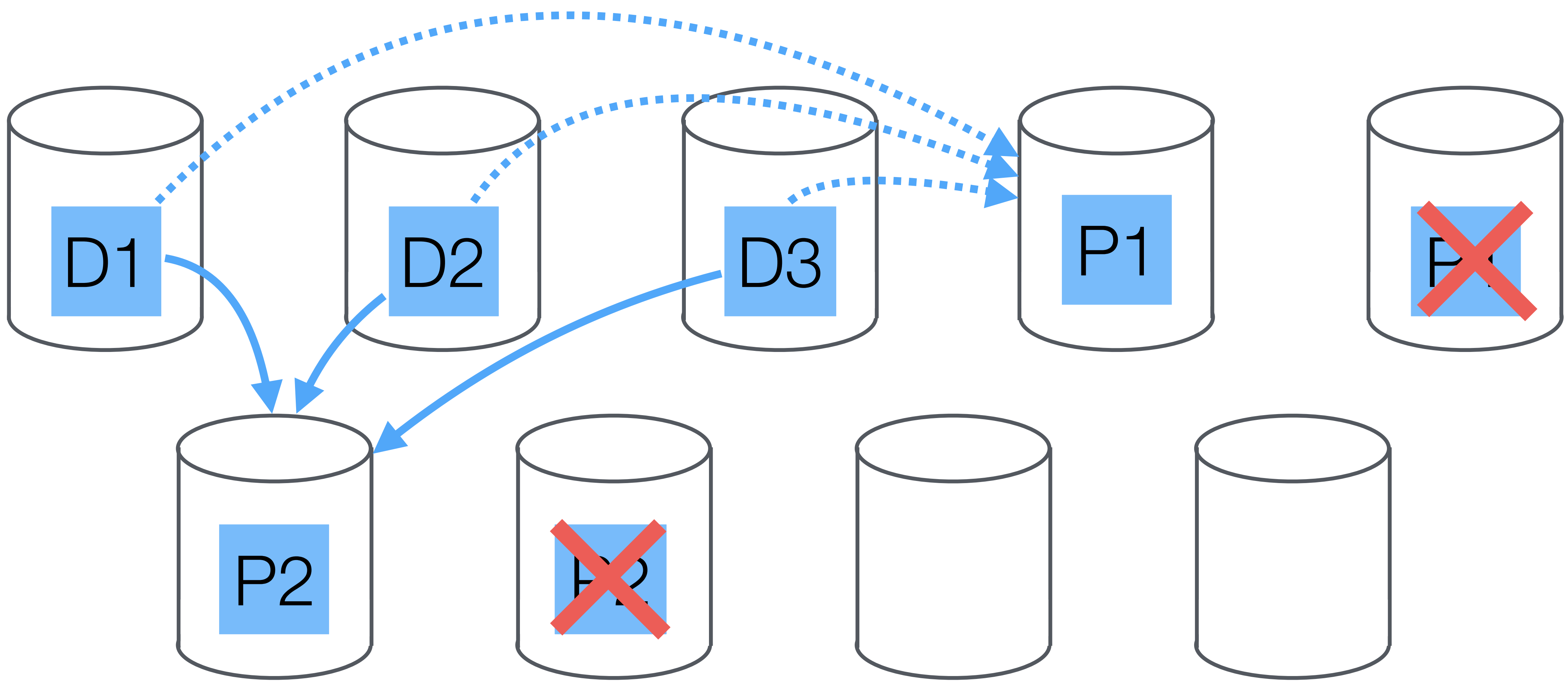
- Use less storage space to tolerate the same number of failures
- (k,r) Reed-Solomon (RS) code
  - compute r parity blocks from k data blocks



(k=3,r=2) RS code

Reed-Solomon Code

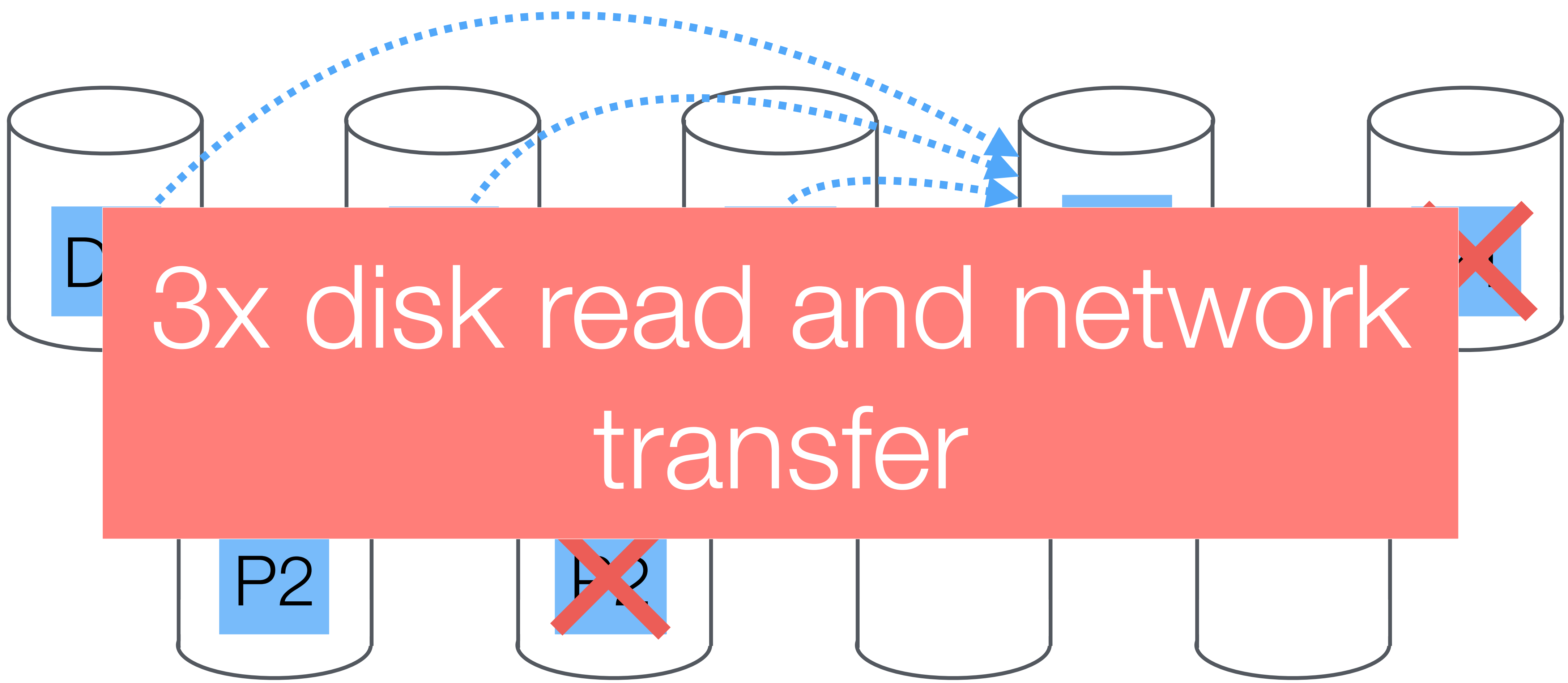
- Achieve the optimal storage overhead to tolerate the same number of failures
- Typically high cost of reconstruction
  - need to obtain k blocks to reconstruct one



(k=3,r=2) RS code

Reed-Solomon Code

- Achieve the optimal storage overhead to tolerate the same number of failures
- Typically high cost of reconstruction
  - need to obtain k blocks to reconstruct one



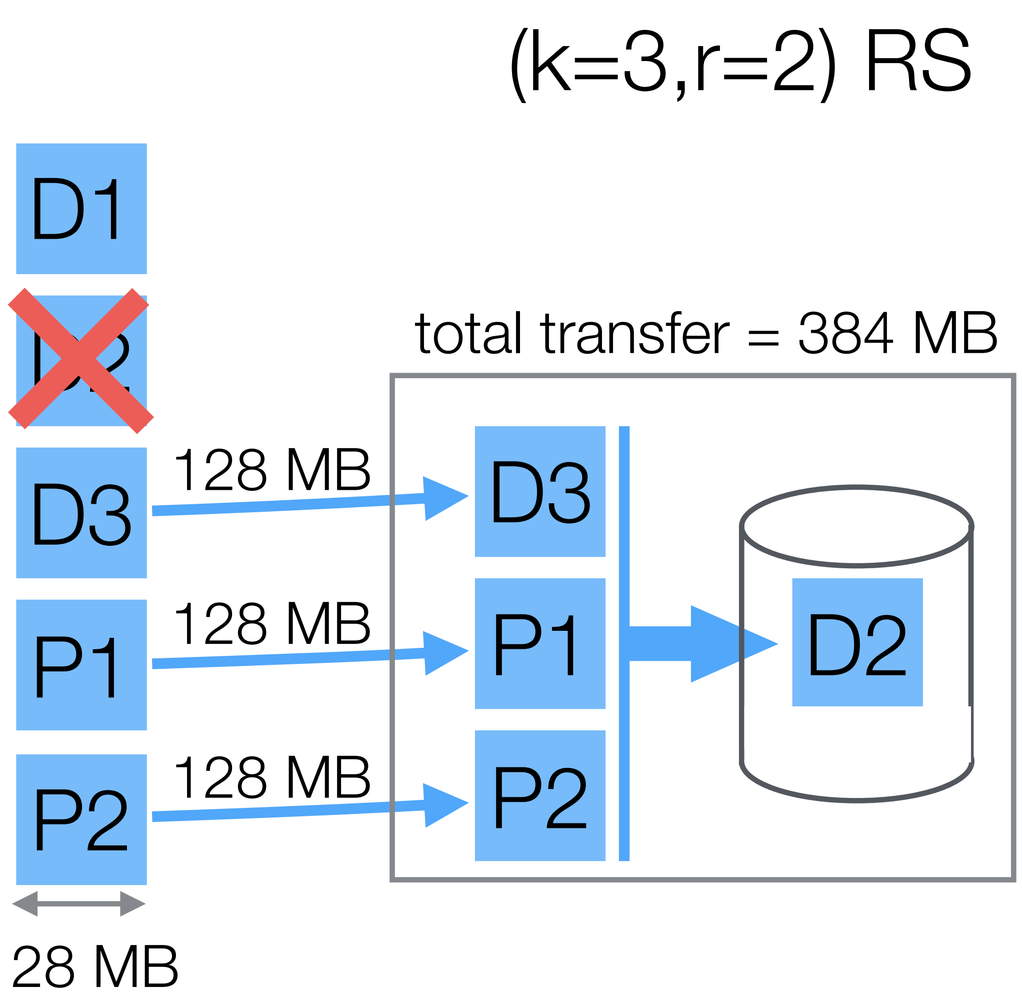
(k=3,r=2) RS code

Network Transfer

- Minimum-storage regenerating (MSR) codes [Dimakis et al, Trans. IT, 2011]
  - the optimal storage overhead like RS code
  - minimize the network transfer during reconstruction

Network Transfer

- Minimum-storage regenerating (MSR) codes [Dimakis et al, Trans. IT, 2011]
  - the optimal storage overhead like RS code
  - minimize the network transfer during reconstruction

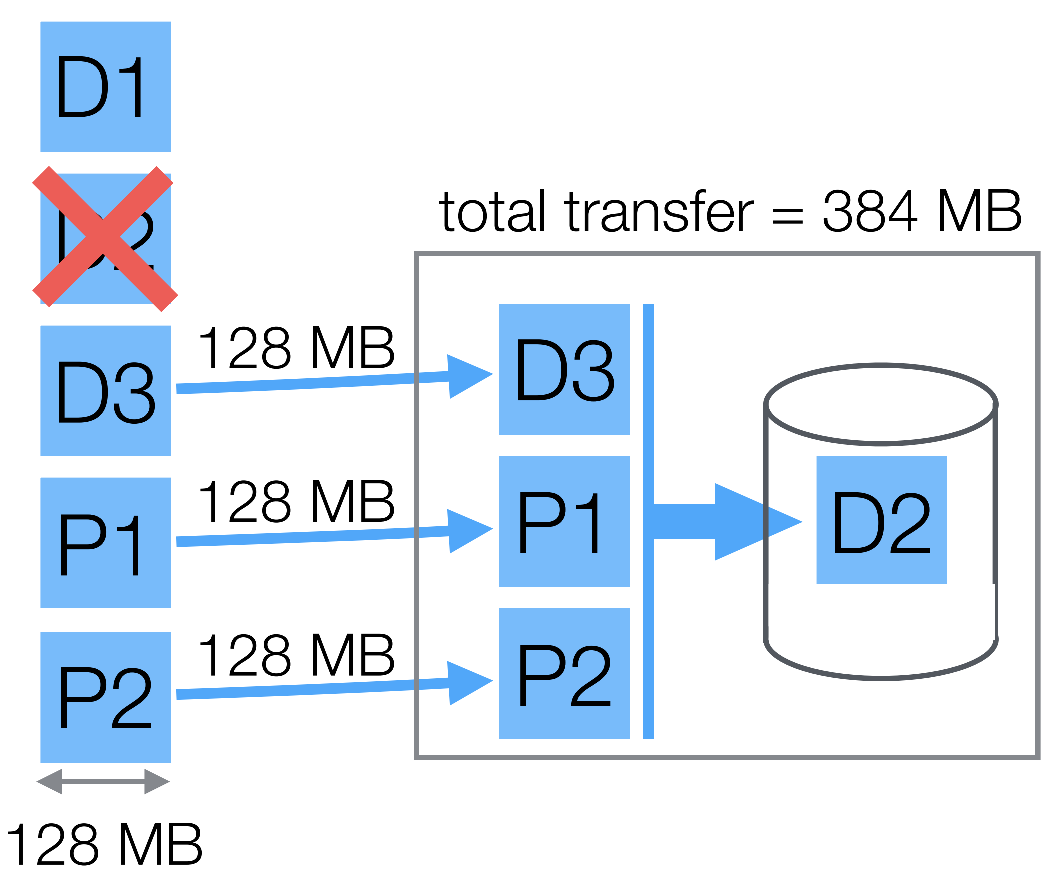


Network Transfer

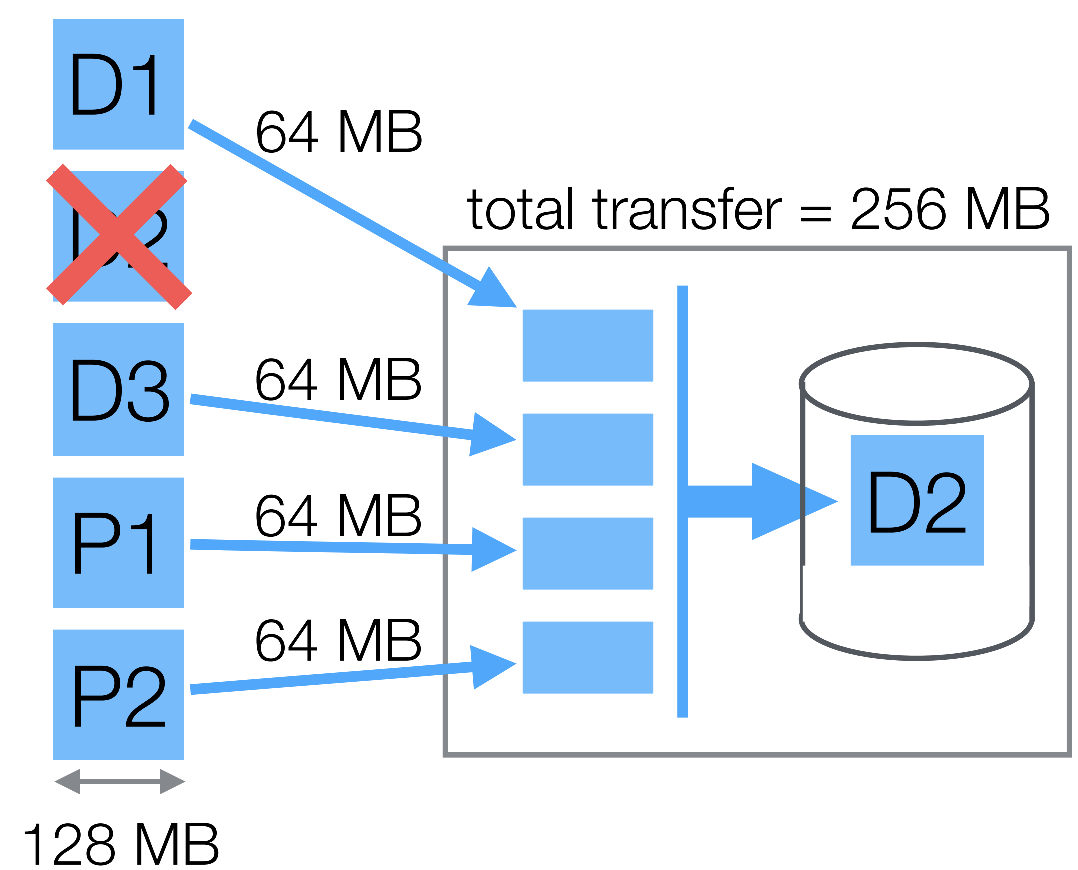
- Minimum-storage regenerating (MSR) codes [Dimakis et al, Trans. IT, 2011]
  - the optimal storage overhead like RS code
  - minimize the network transfer during reconstruction

download
a small fraction of
data from d
helpers

(k=3,r=2) RS

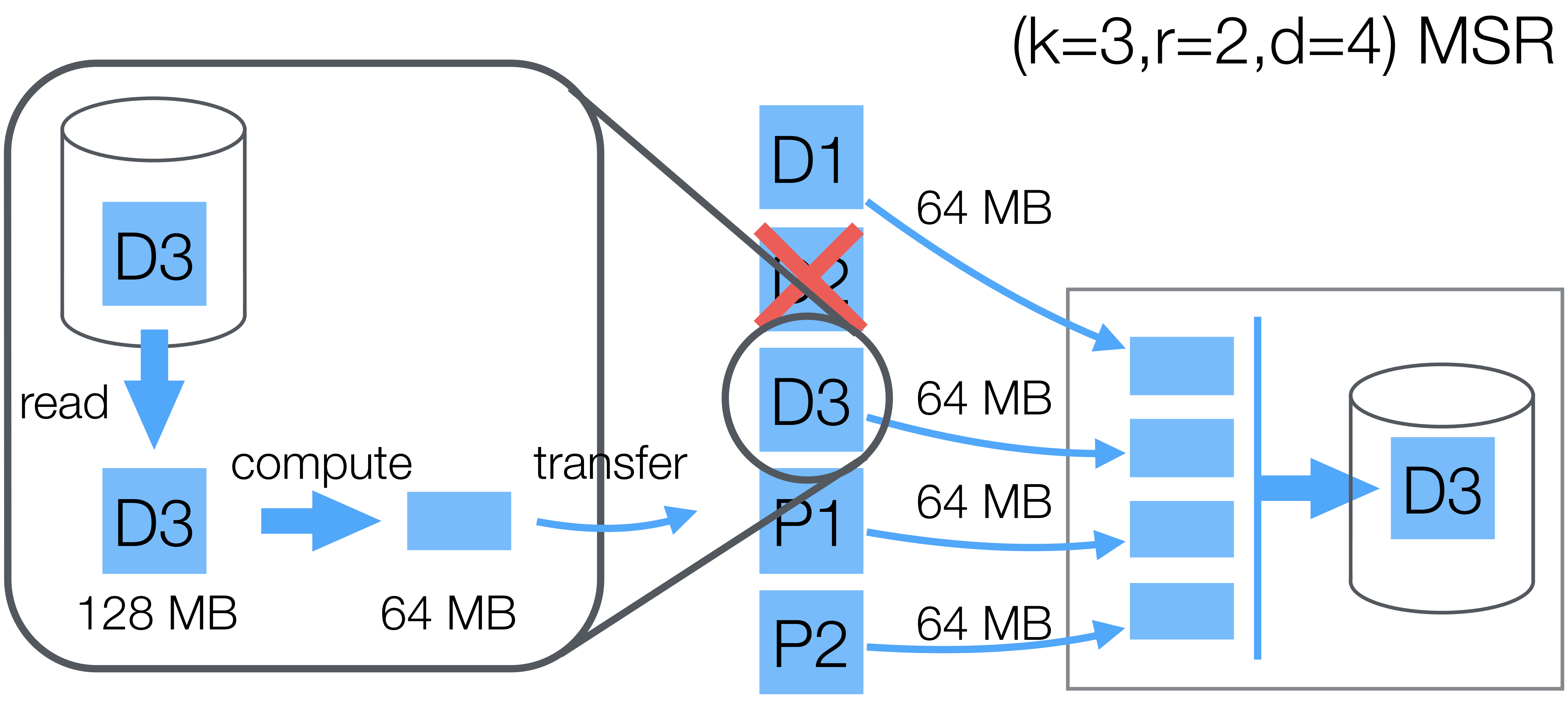


(k=3,r=2,d=4) MSR



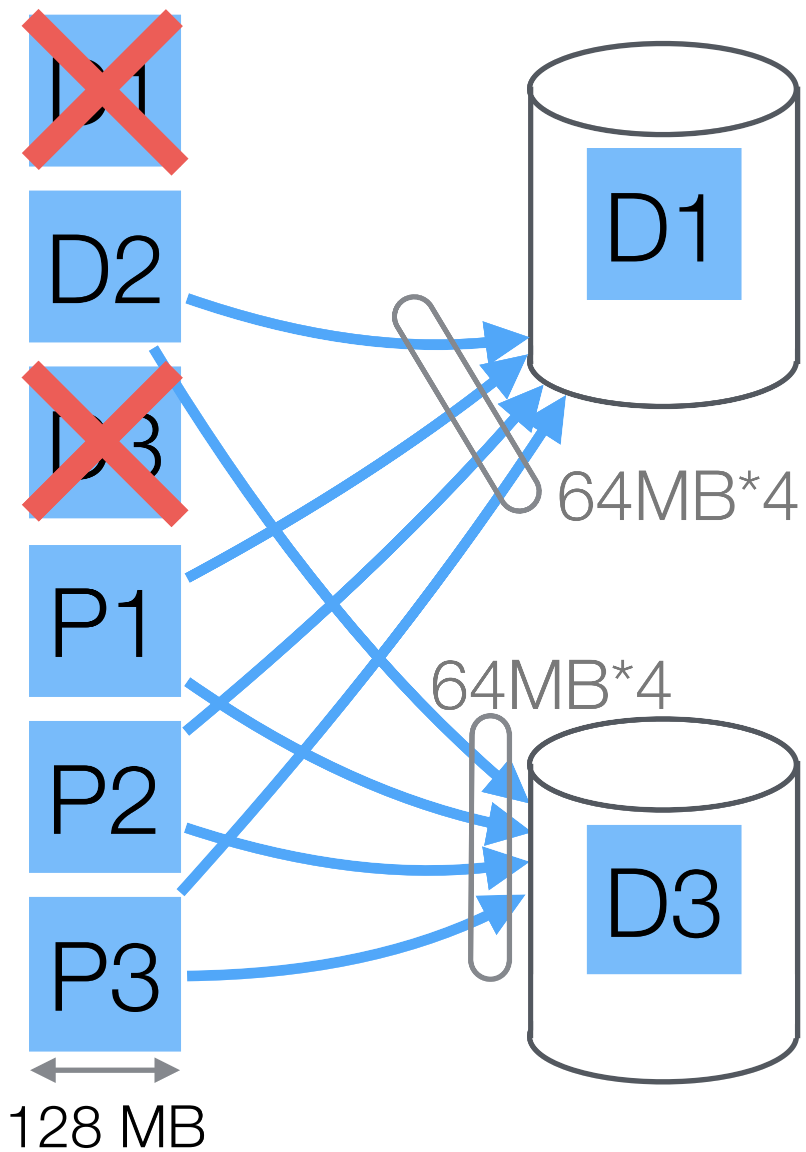
Disk I/O

- MSR codes will incur even more disk I/O than RS codes since each helper needs to read all its data to compute a small fraction sent out.



Can we have erasure codes that save both network transfer and disk I/O during reconstruction?

Multiple Failures

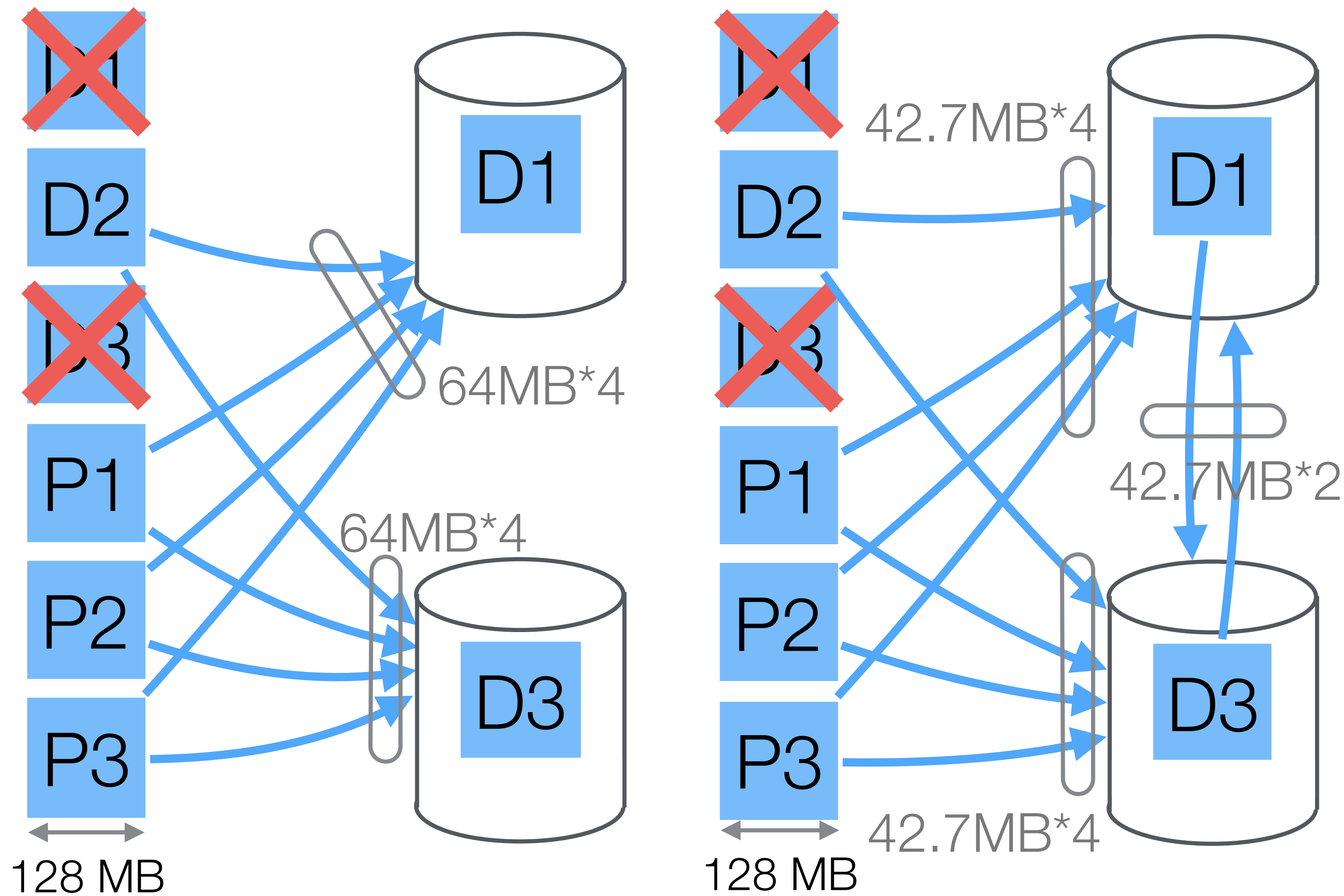


total transfer = 512 MB
disk read = 1024 MB
storage overhead = 2x

(k=3,r=3,d=4) MSR

- Opportunities of fixing multiple failures exists.
  - correlated failures (disk, switch, power)
  - periodical check of failures
  - reconstruct after a certain number of failures
- Typically, erasure codes like RS and MSR codes fix failures separately.
- Coalesce reconstructions can instantly save disk I/O

Multiple Failures



total transfer = 512 MB
disk read = 1024 MB
storage overhead = 2x

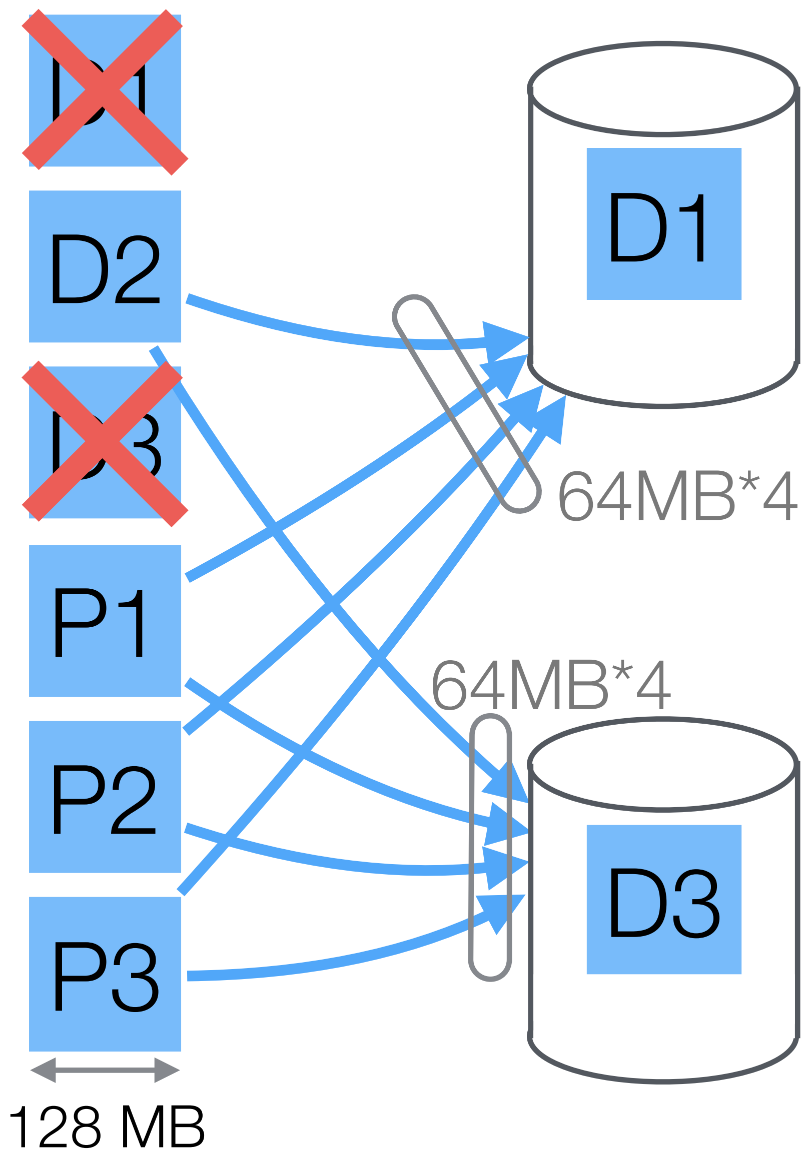
(k=3,r=3,d=4) MSR

total transfer = 427 MB
disk read = 512 MB
storage overhead = 2x

optimal network transfer

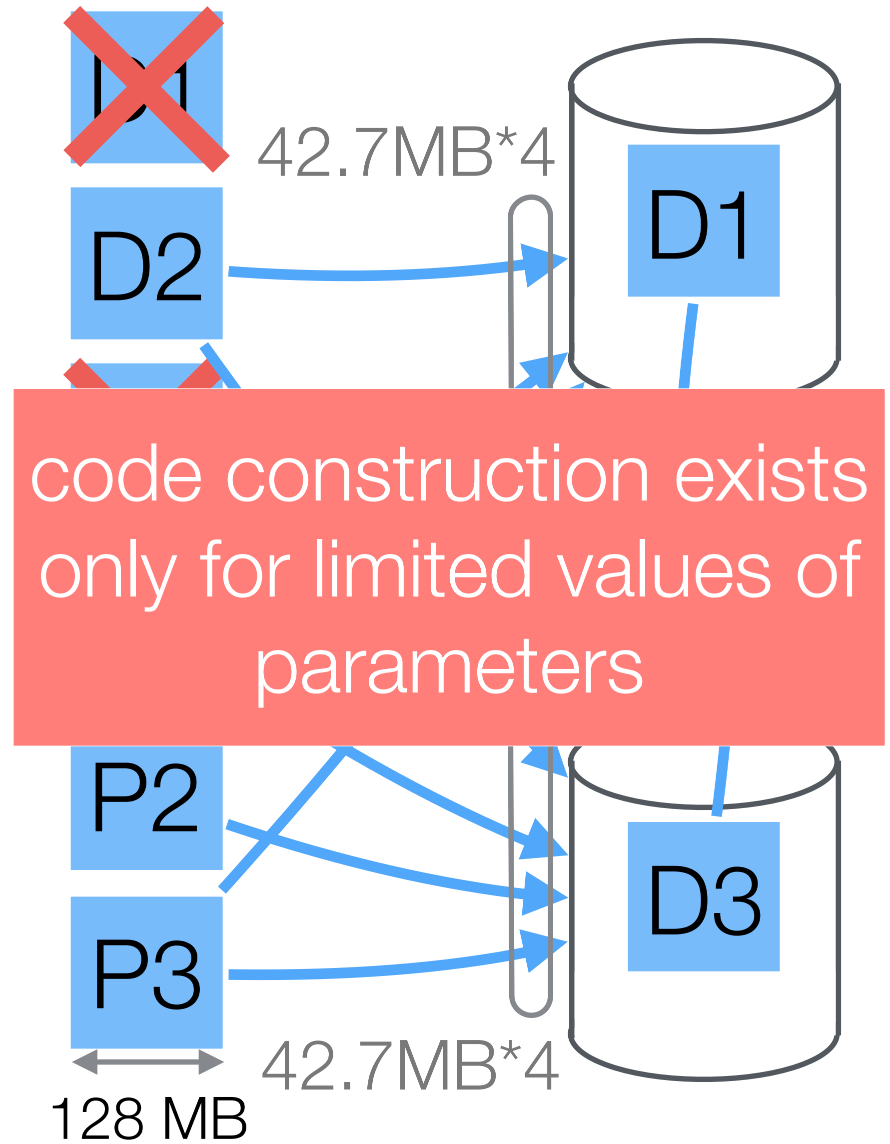
[Shum et al, Trans. IT, 2013]

Multiple Failures



total transfer = 512 MB
disk read = 1024 MB
storage overhead = 2x

(k=3,r=3,d=4) MSR

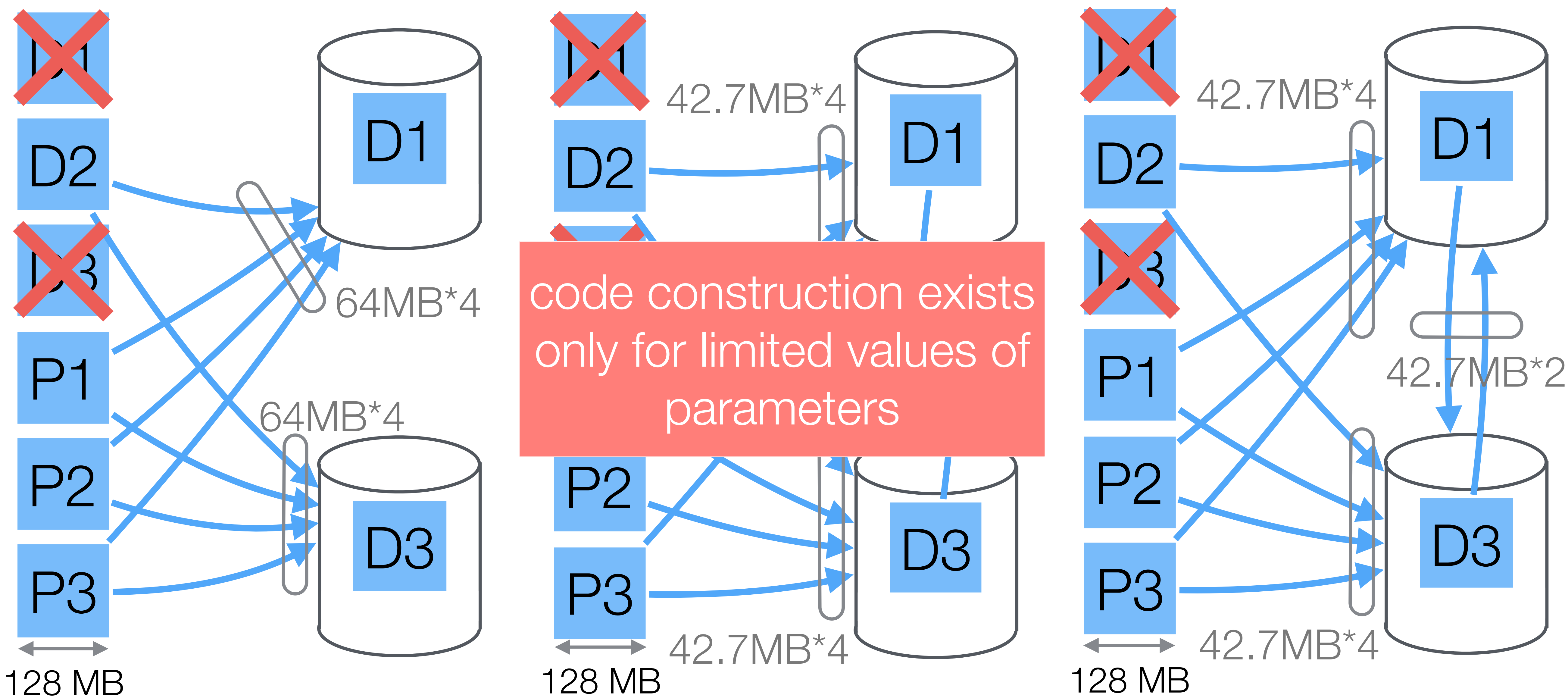


total transfer = 427 MB
disk read = 512 MB
storage overhead = 2x

optimal network transfer

[Shum et al, Trans. IT, 2013]

Multiple Failures



total transfer = 512 MB
disk read = 1024 MB
storage overhead = 2x
(k=3,r=3,d=4) MSR

total transfer = 427 MB
disk read = 512 MB
storage overhead = 2x
optimal network transfer
[Shum et al, Trans. IT, 2013]

total transfer = 427 MB
disk read = 512 MB
storage overhead = 2.25x
Beehive

Contributions

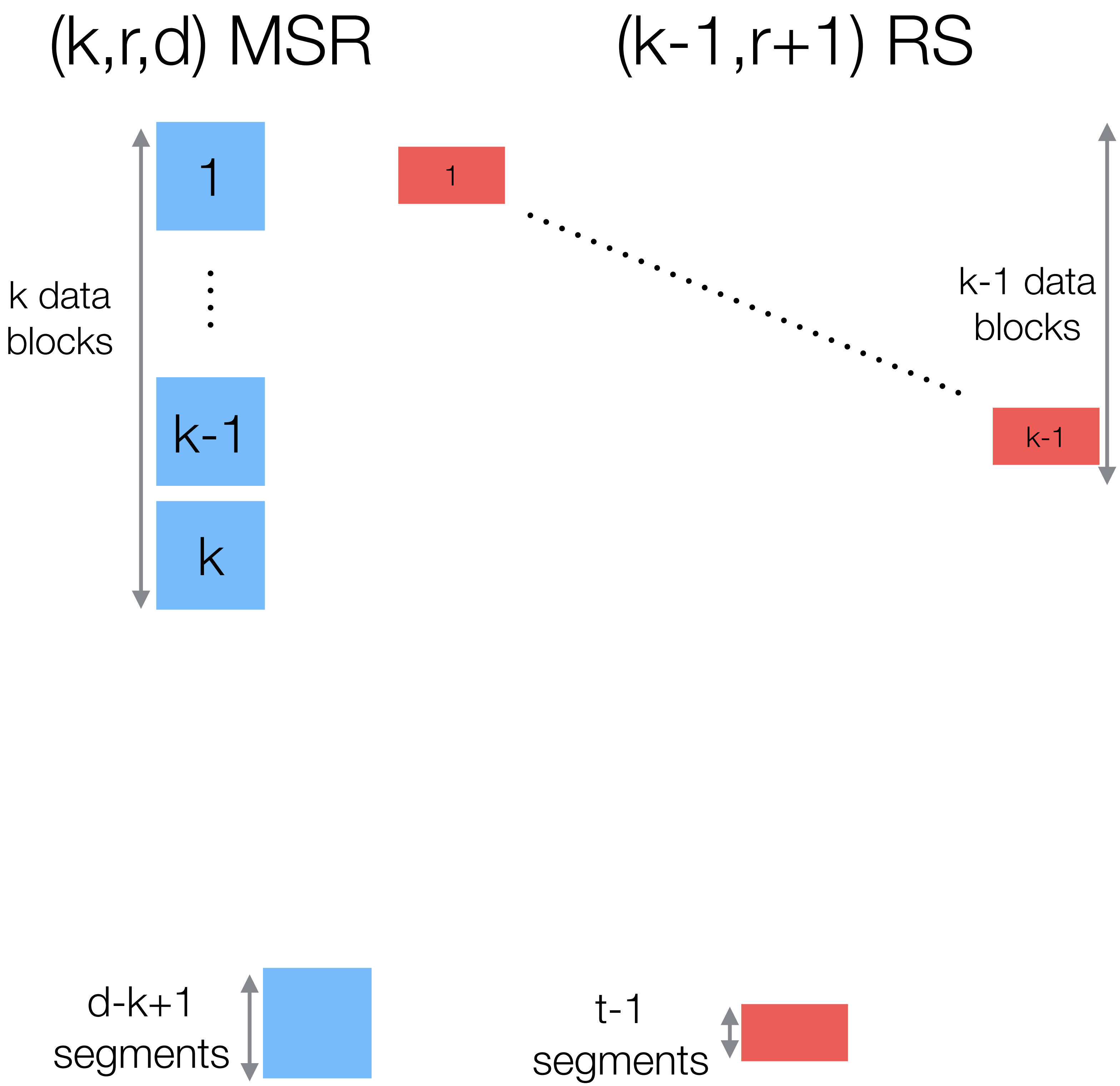
- Beehive, a new kind of erasure codes that achieve the optimal network transfer of coalesced reconstructions
  - with a wide range of system parameters
  - with marginally additional storage overhead
- C++ implementation to demonstrate the performance

System Parameters

- k: the minimum number of blocks to decode the original data
- r: the maximum number of missing blocks to tolerate without hurting data durability/availability
- t: the number of failed blocks to reconstruct
- d: the number of existing blocks to contact during reconstruction ($d \geq 2k-1$)

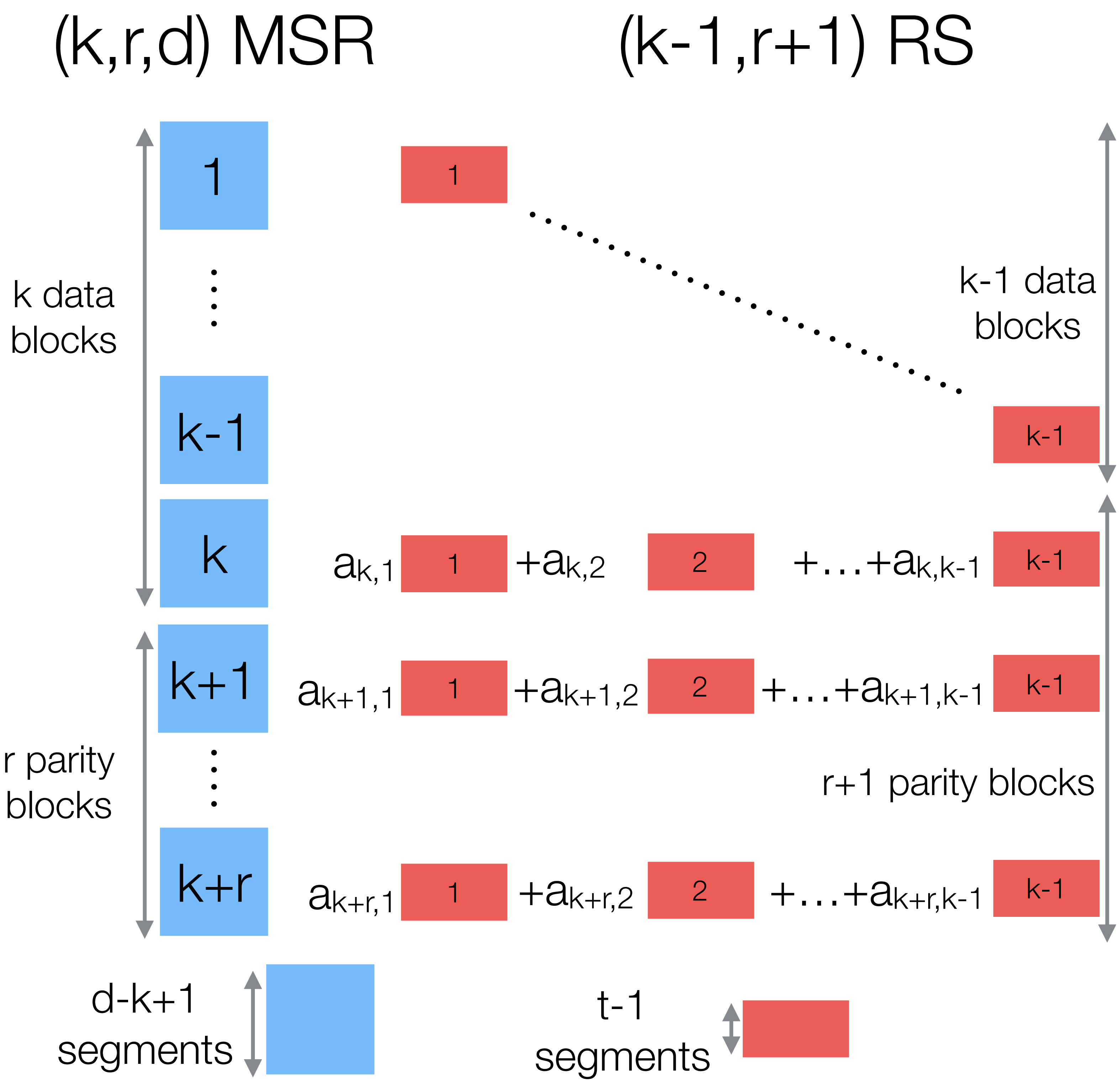
Code Construction

Code Construction



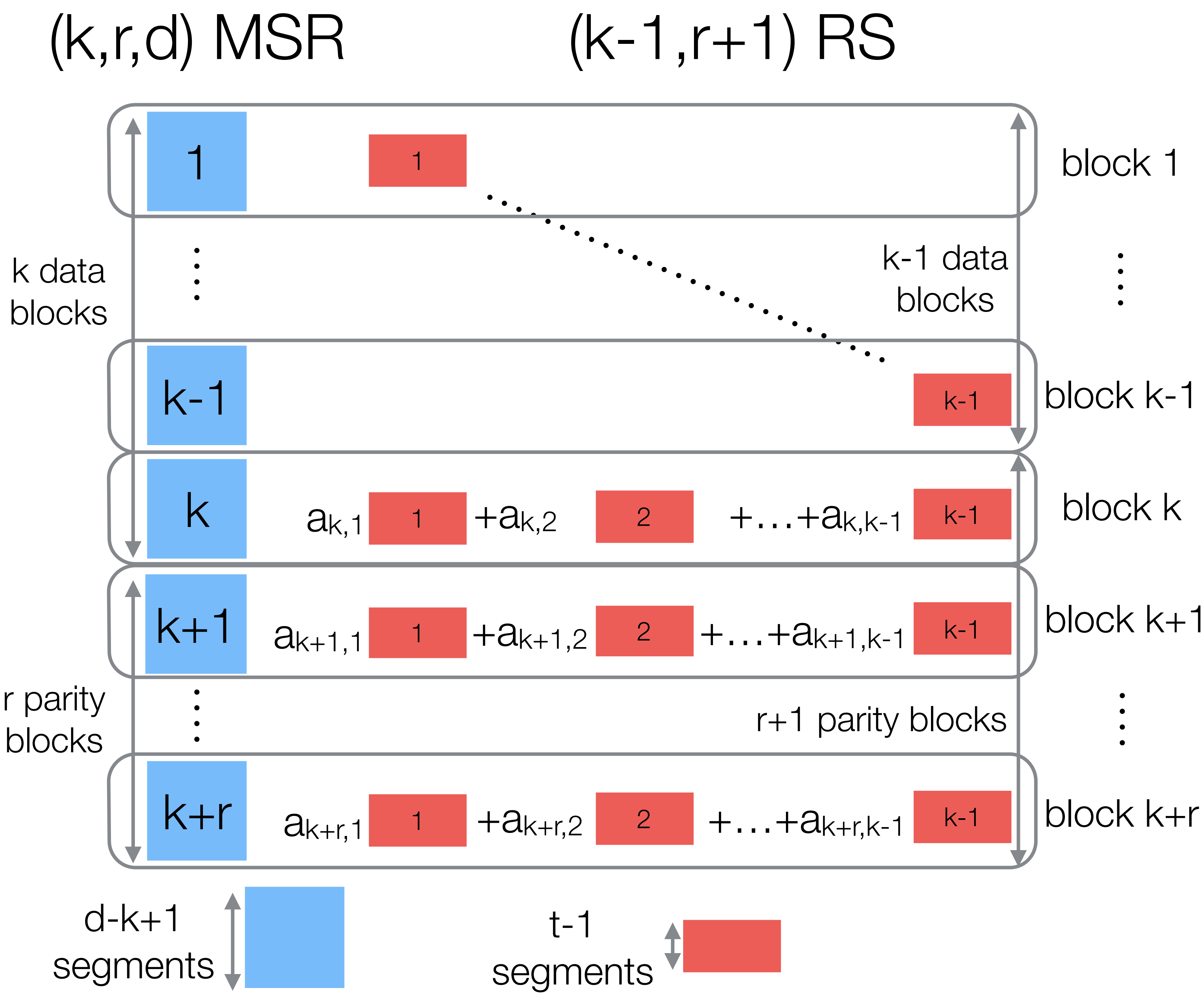
- Beehive codes are constructed by combining MSR codes and RS codes.

Code Construction



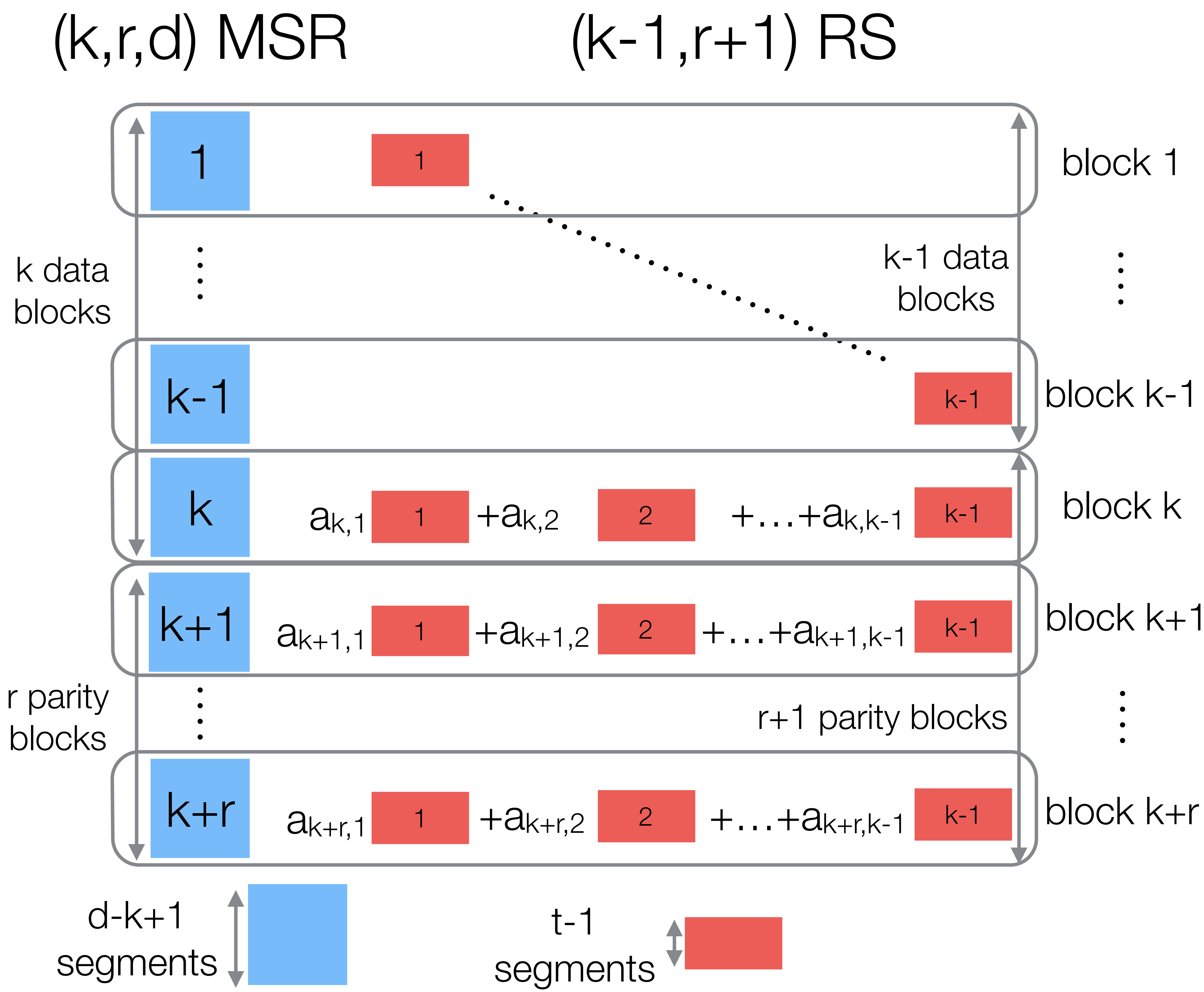
- Beehive codes are constructed by combining MSR codes and RS codes.

Code Construction



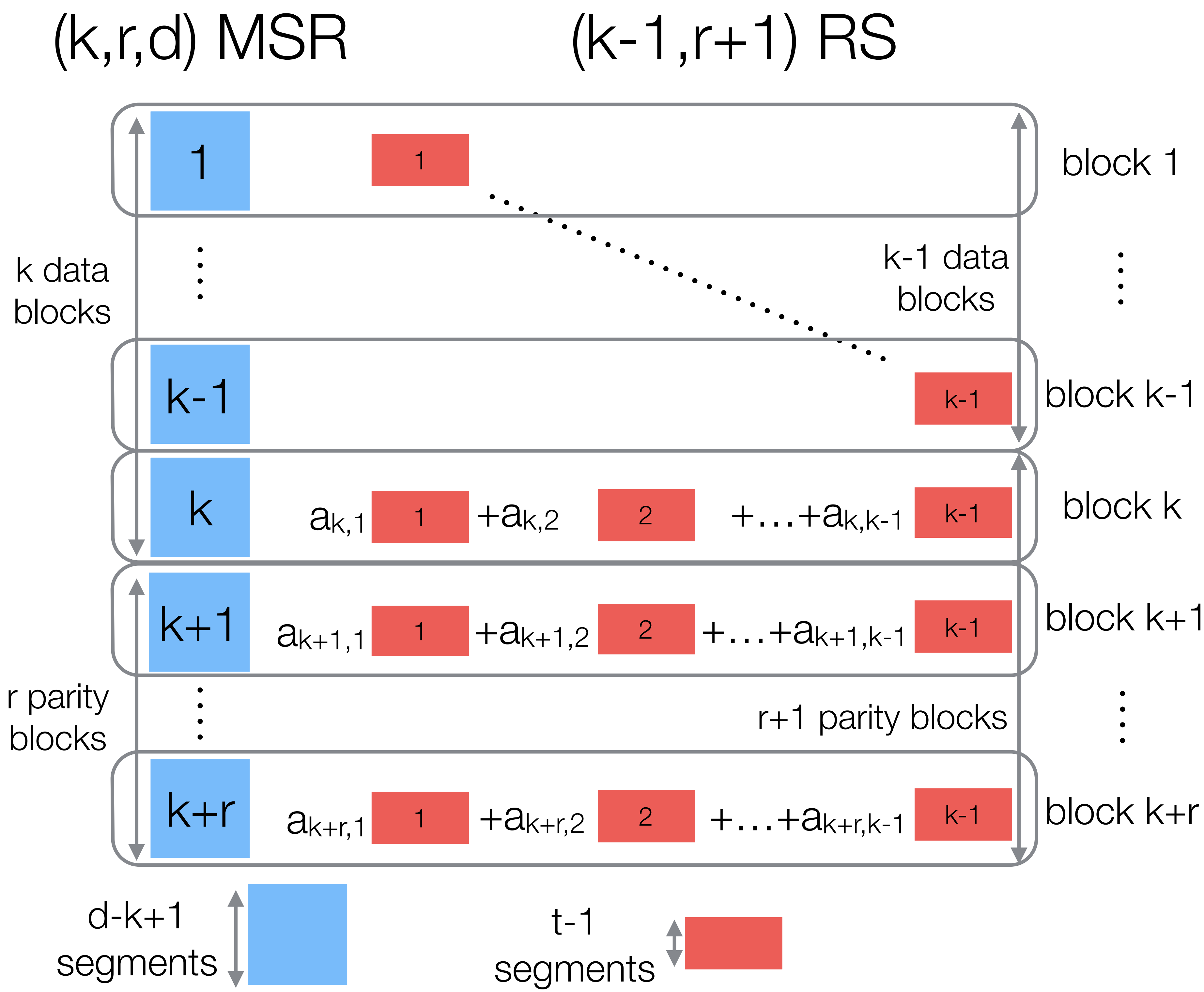
- Beehive codes are constructed by combining MSR codes and RS codes.

Code Construction



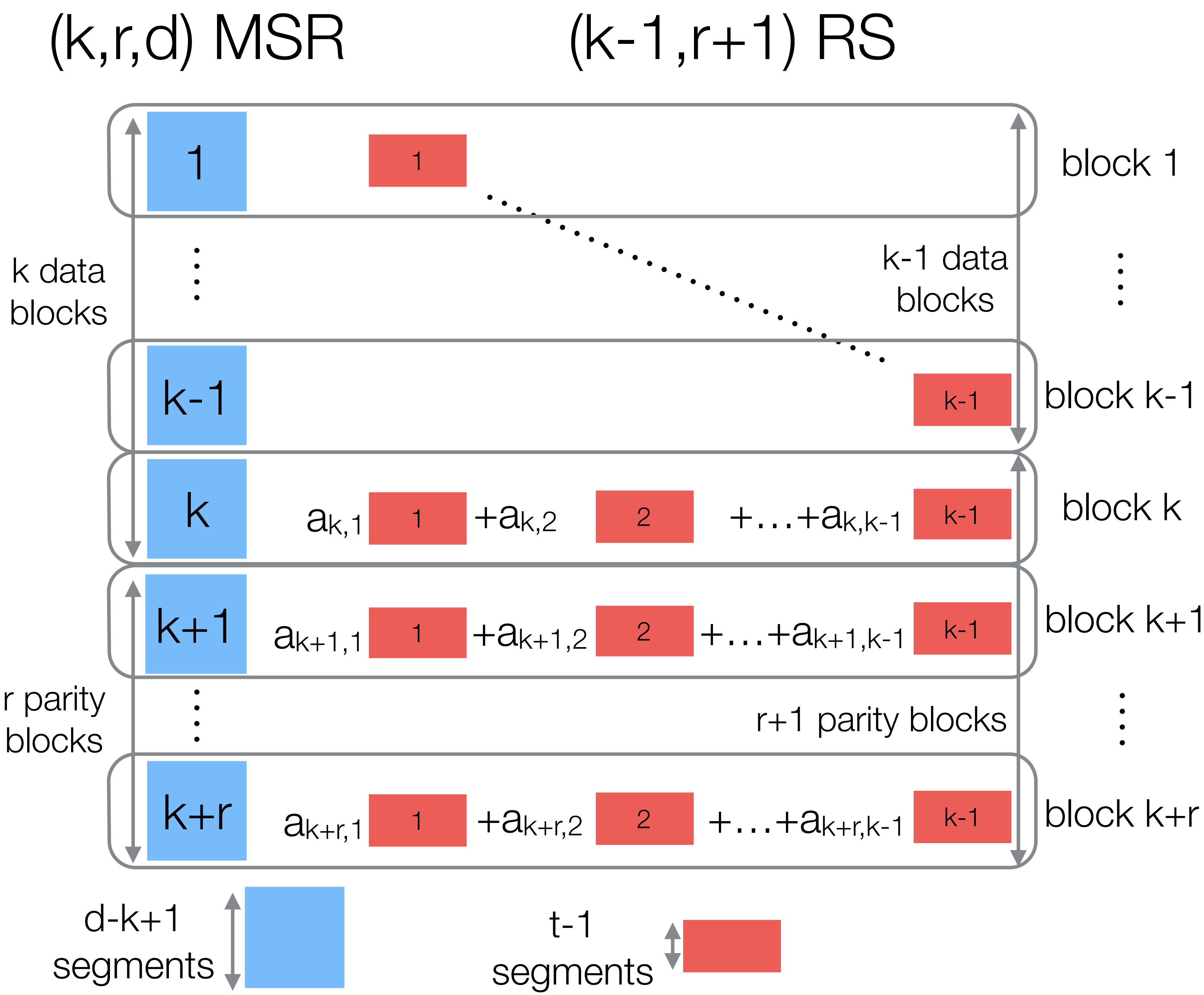
- Beehive codes are constructed by combining MSR codes and RS codes.
- Beehive codes can be decoded as long as k blocks survive

Code Construction



- Beehive codes are constructed by combining MSR codes and RS codes.
- Beehive codes can be decoded as long as k blocks survive
- With k+r blocks in total, Beehive codes store t-1 less segments than RS codes and MSR codes

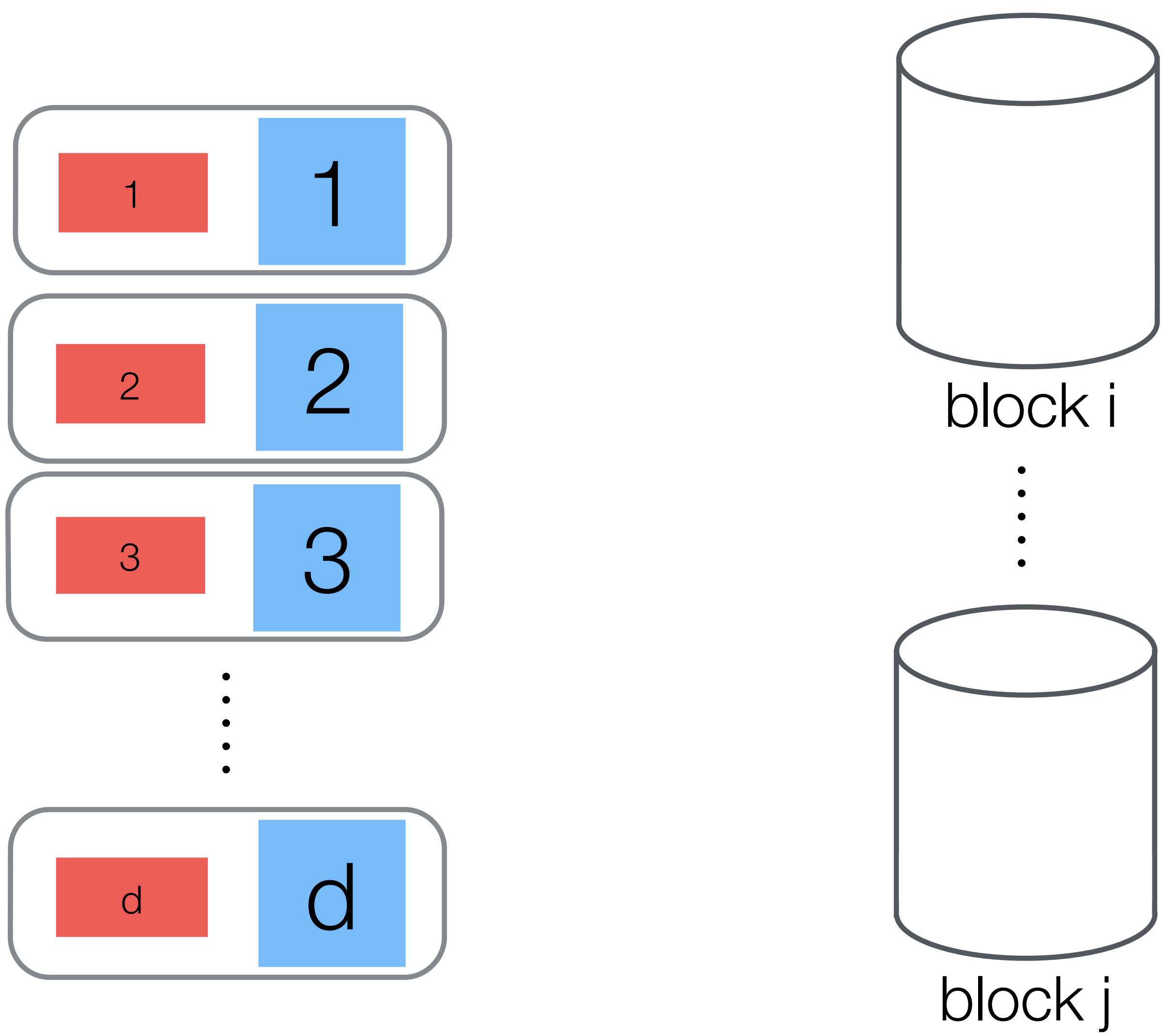
Code Construction



- Beehive codes are constructed by combining MSR codes and RS codes.
- Beehive codes can be decoded as long as k blocks survive
- With k+r blocks in total, Beehive codes store t-1 less segments than RS codes and MSR codes
  - storage overhead =

$$\frac{k+r}{k-\frac{t-1}{d-k+t}} \in \left(\frac{k+r}{k}, \frac{k+r}{k-1}\right)$$

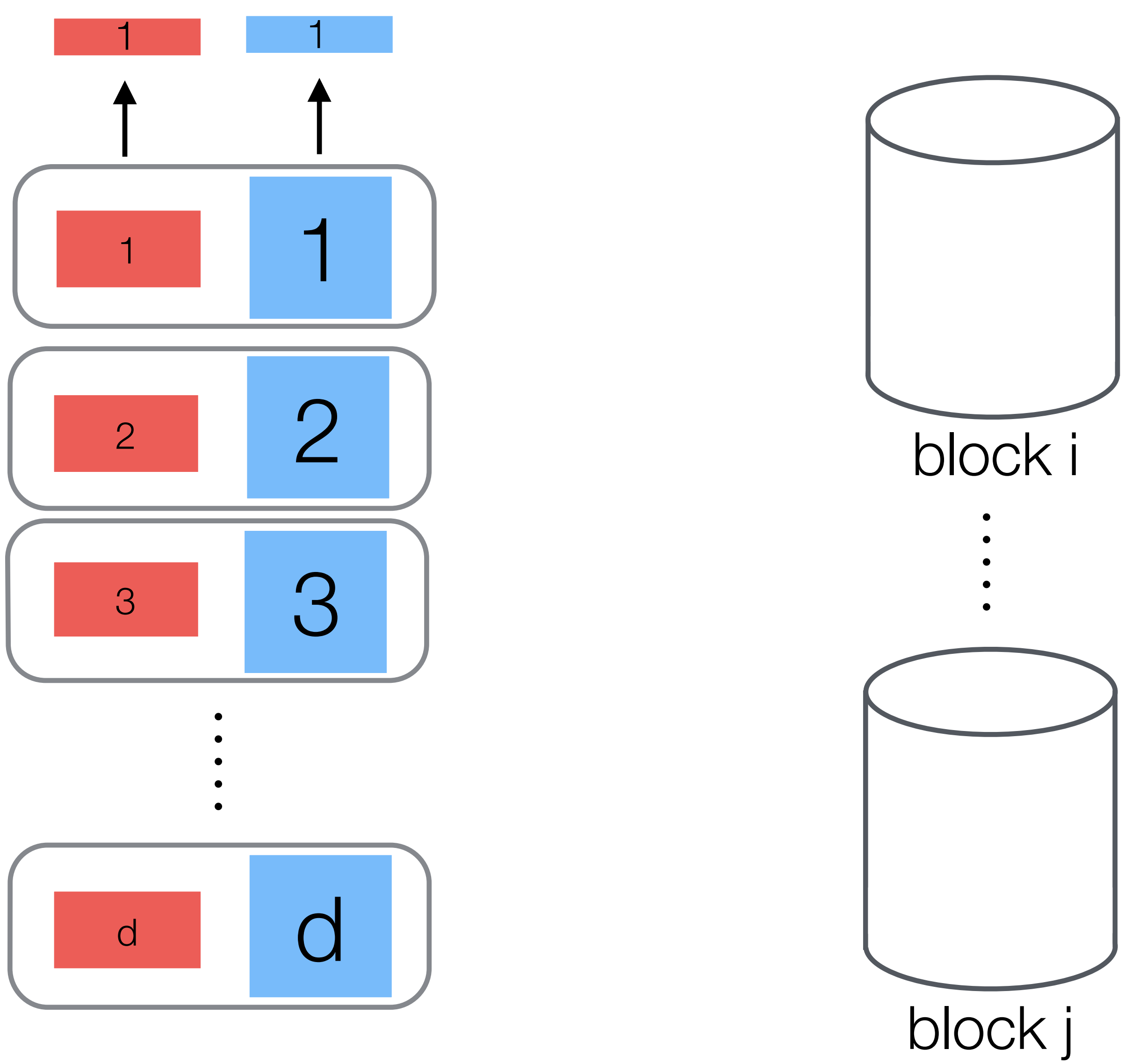
Reconstruction



d helpers

t newcomers

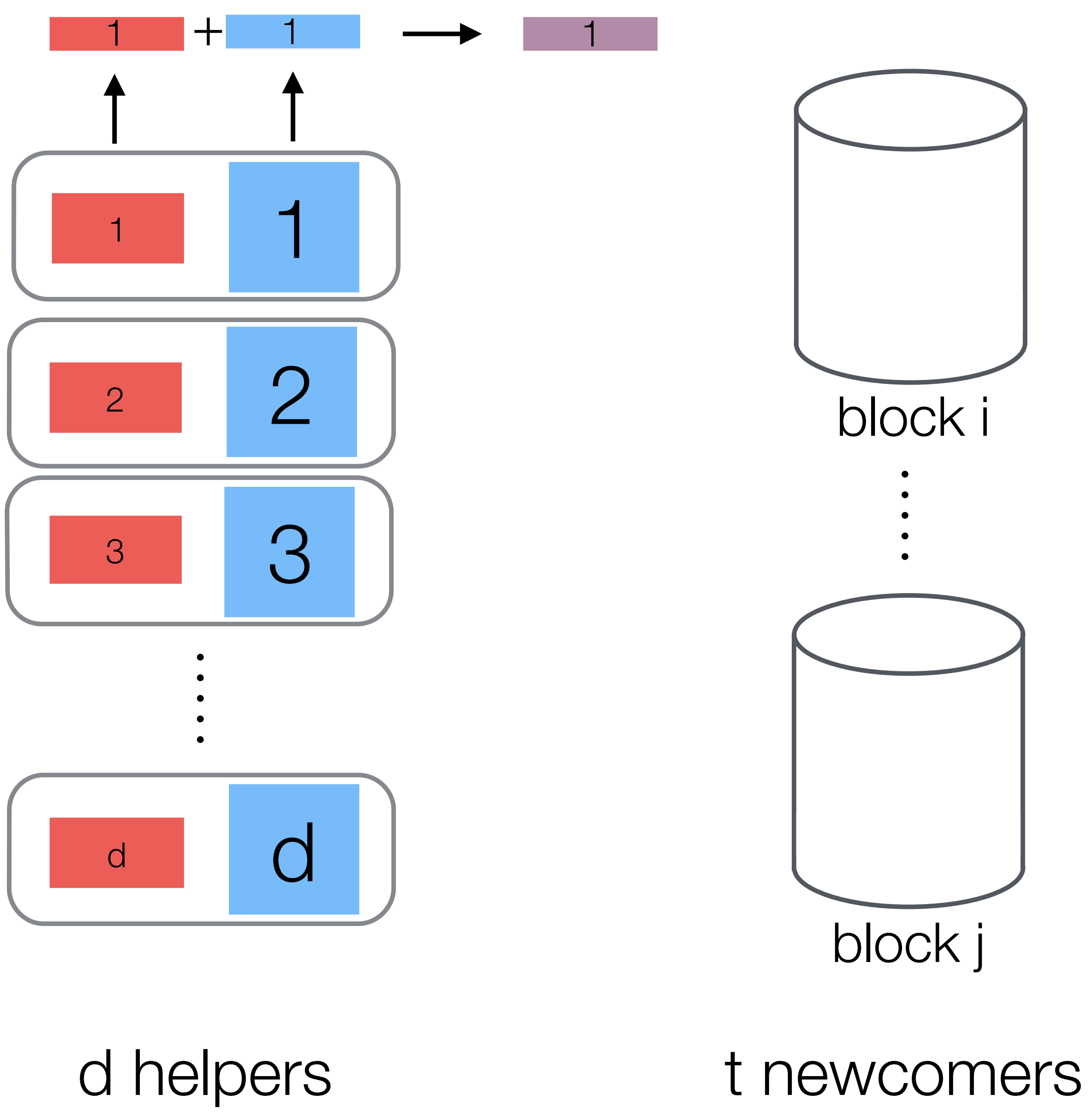
Reconstruction



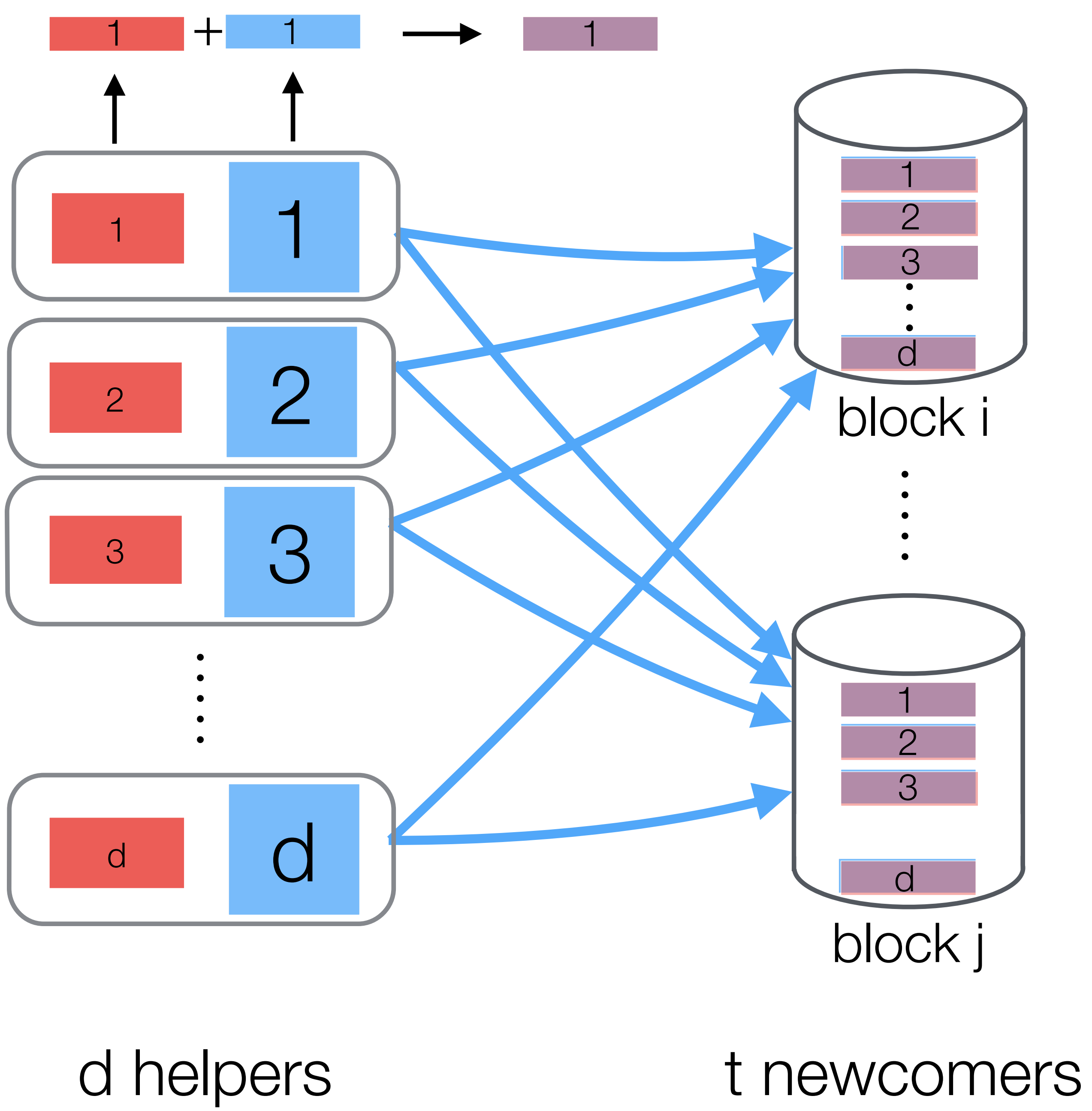
d helpers

t newcomers

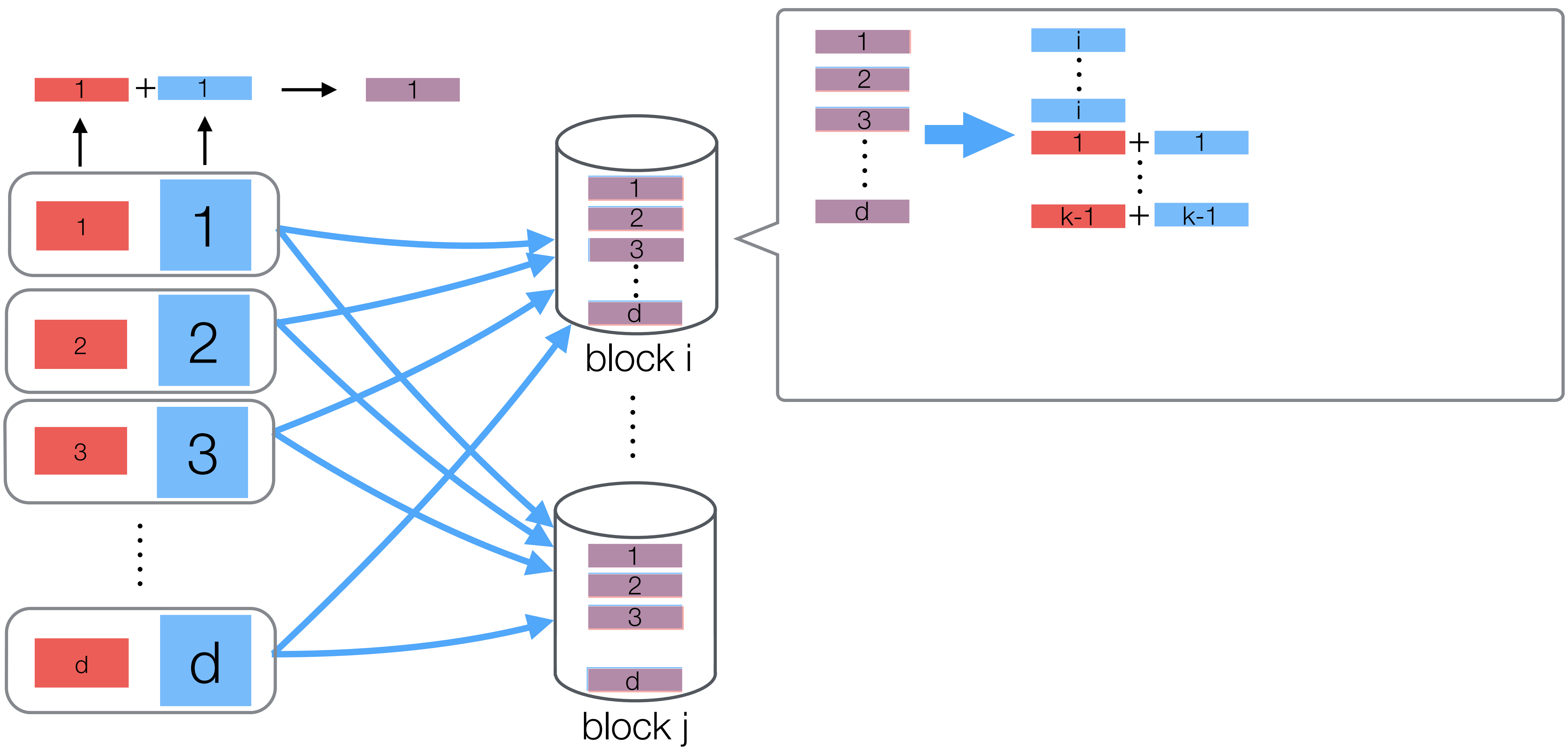
Reconstruction



Reconstruction



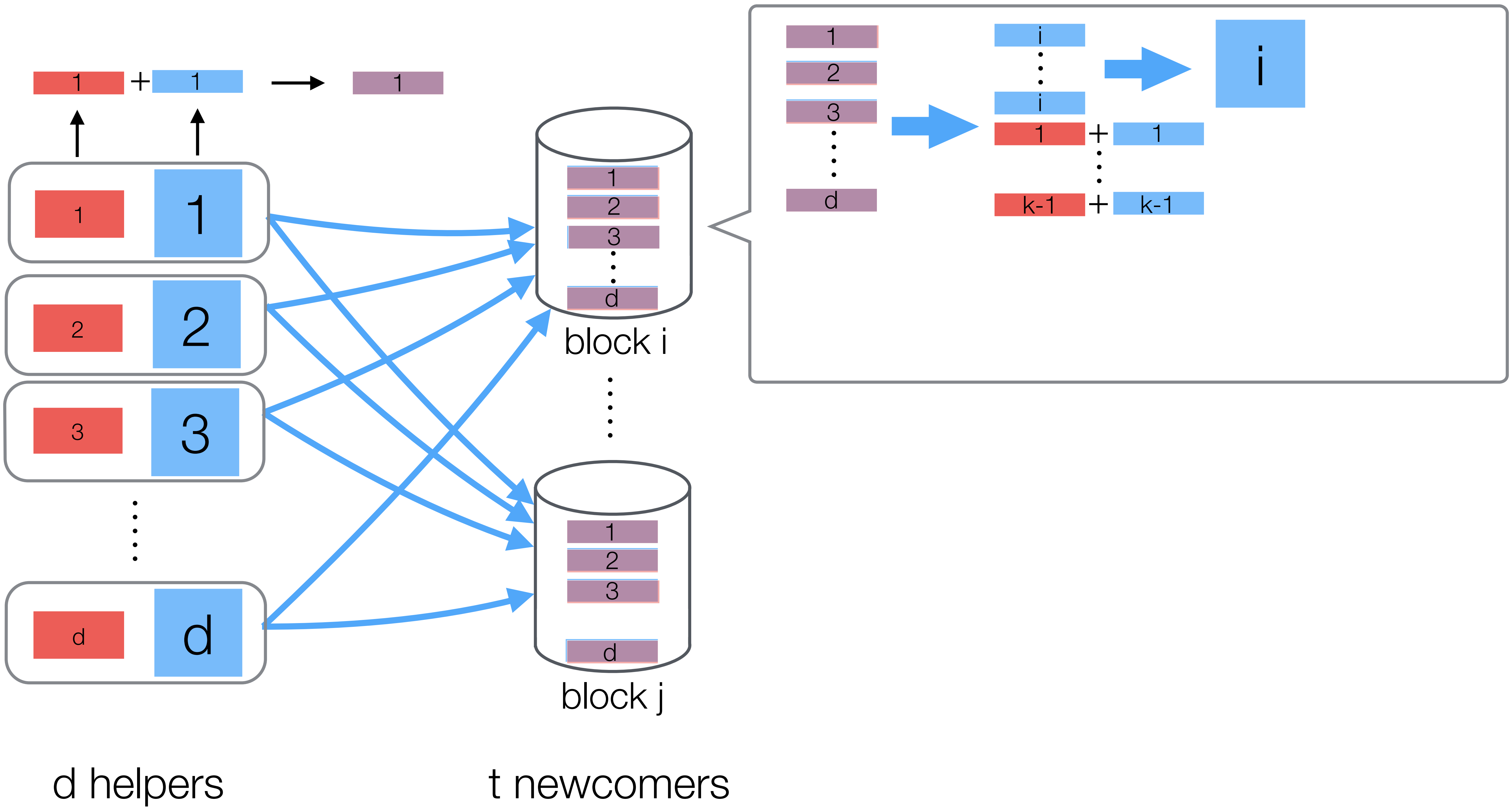
Reconstruction



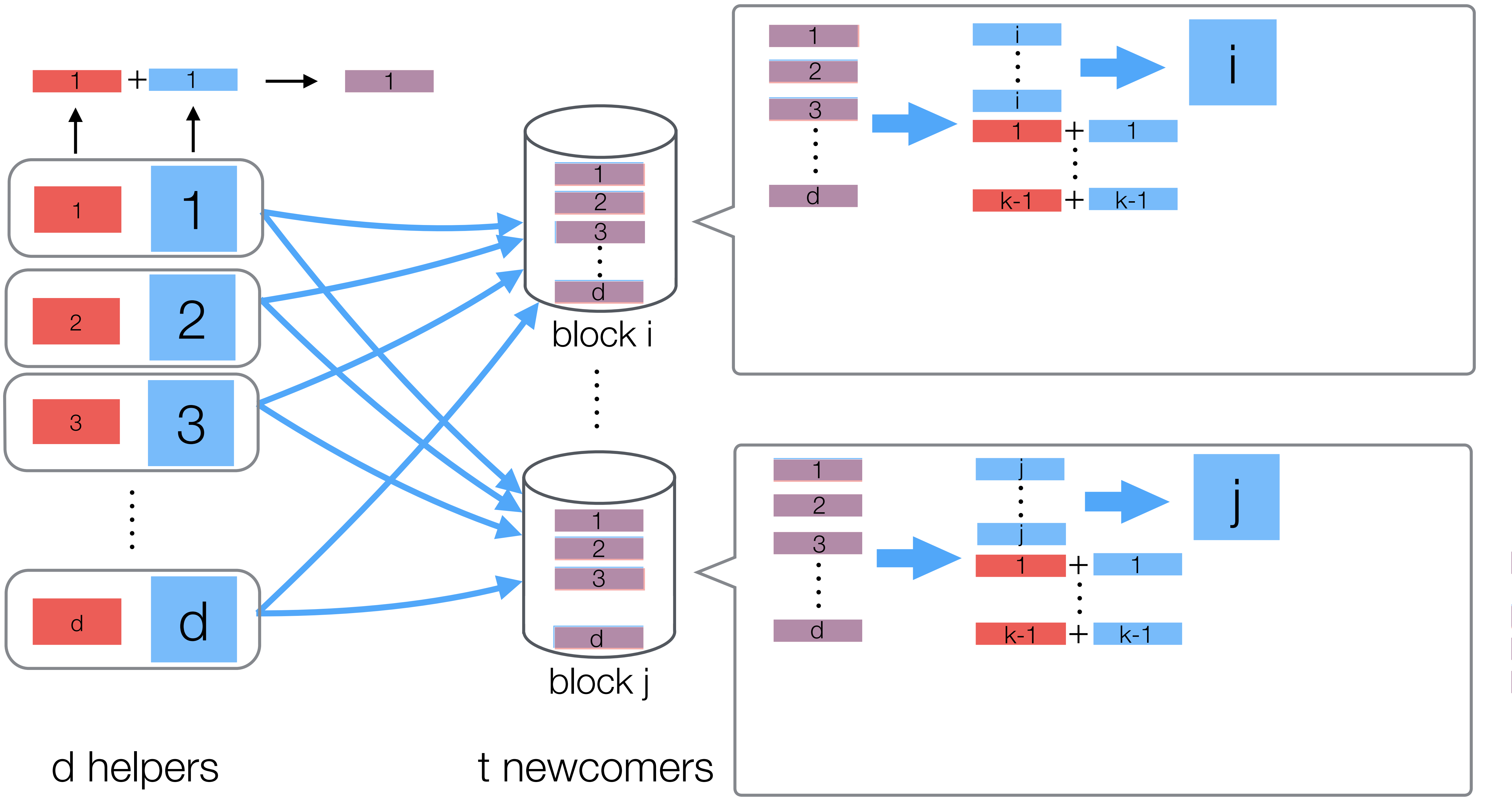
d helpers

t newcomers

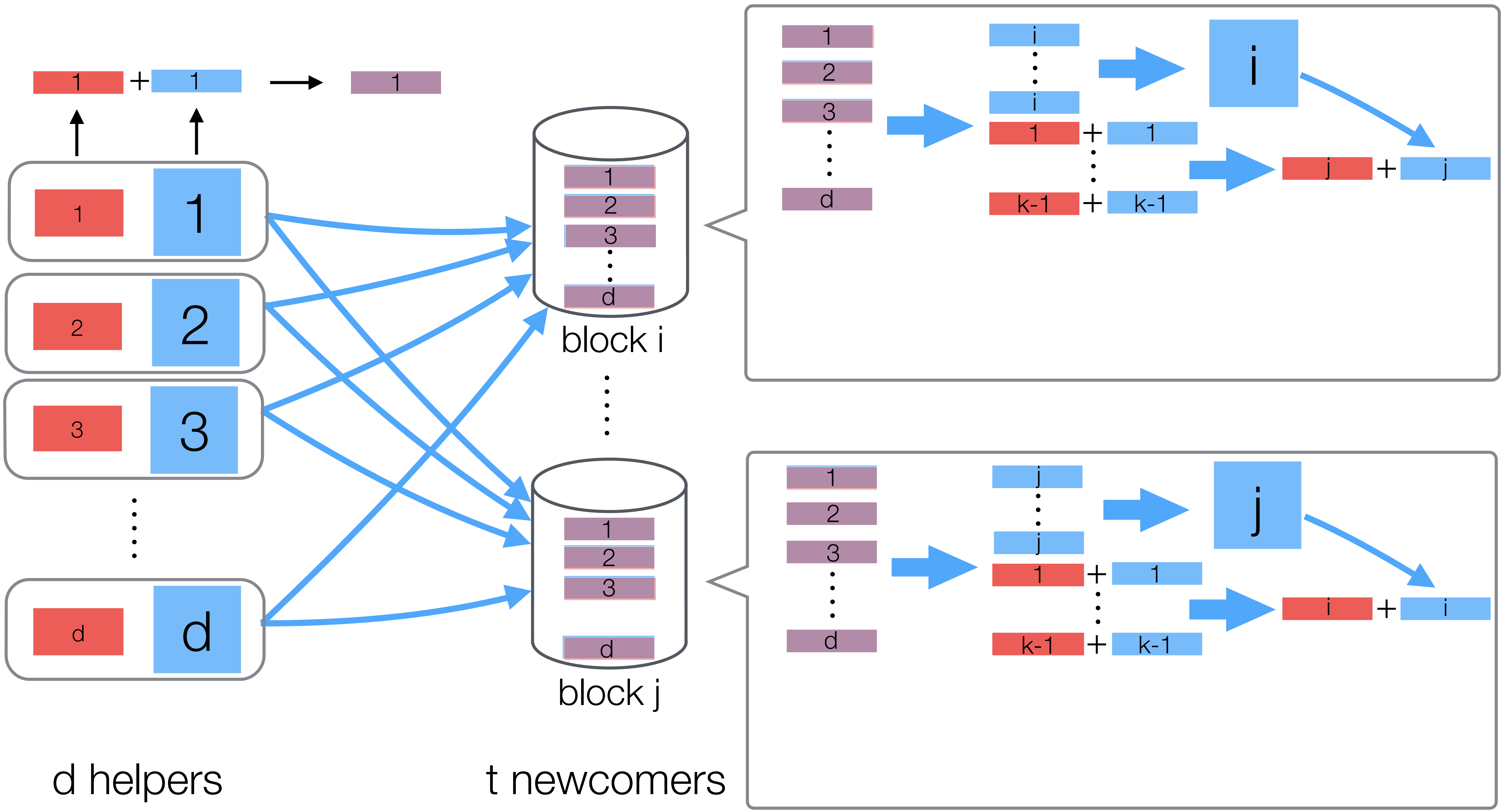
Reconstruction



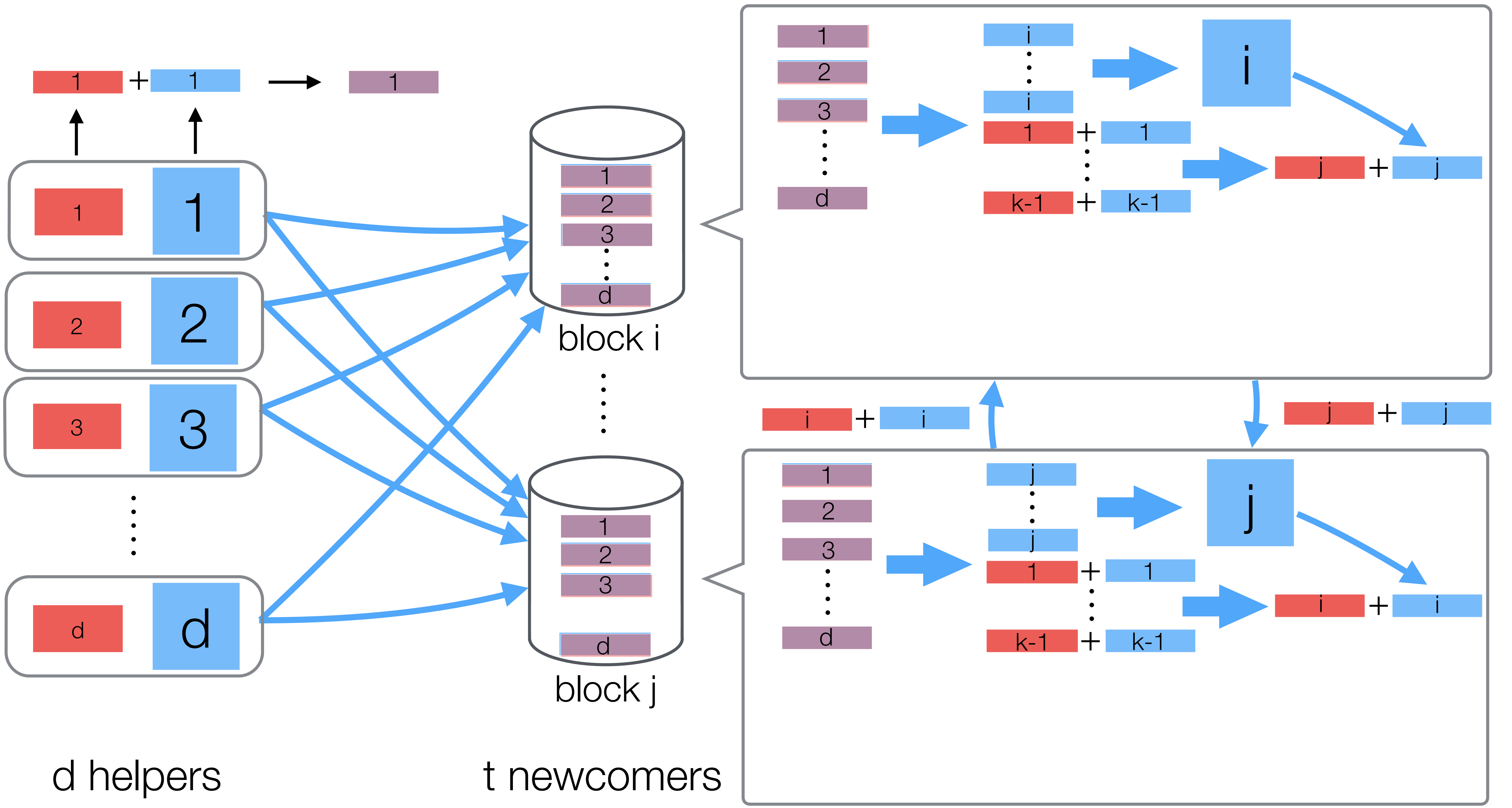
Reconstruction



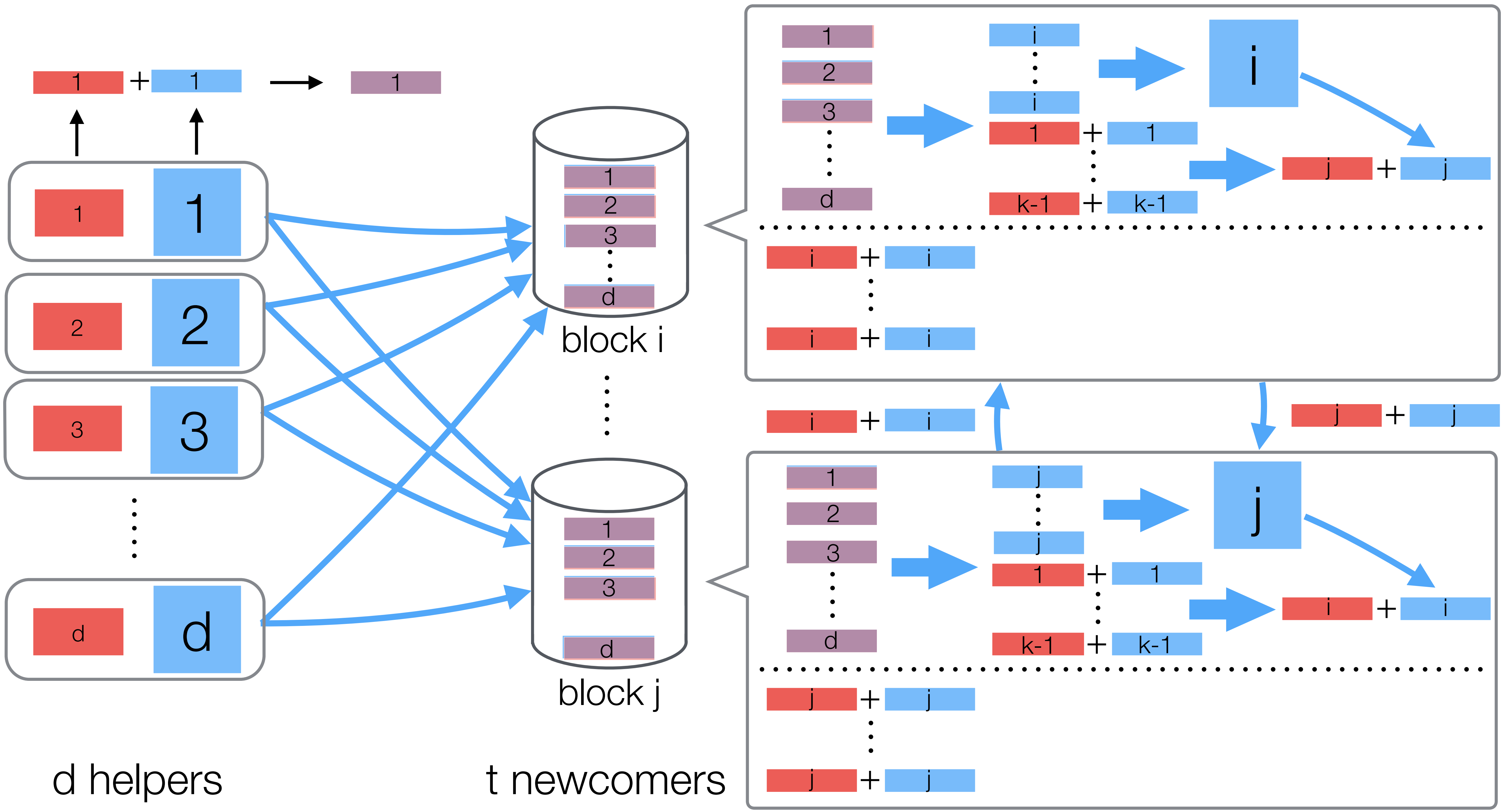
Reconstruction



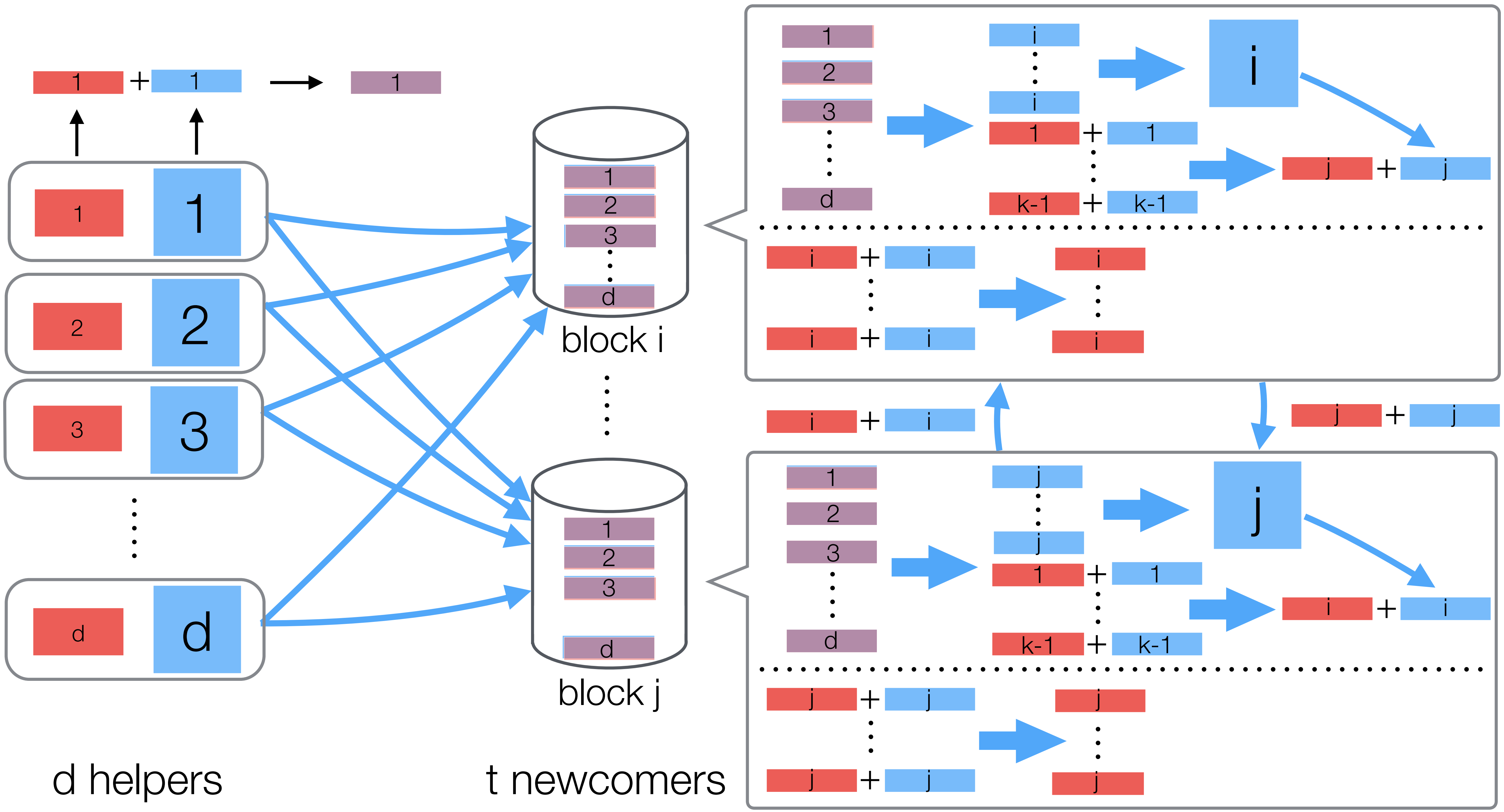
Reconstruction



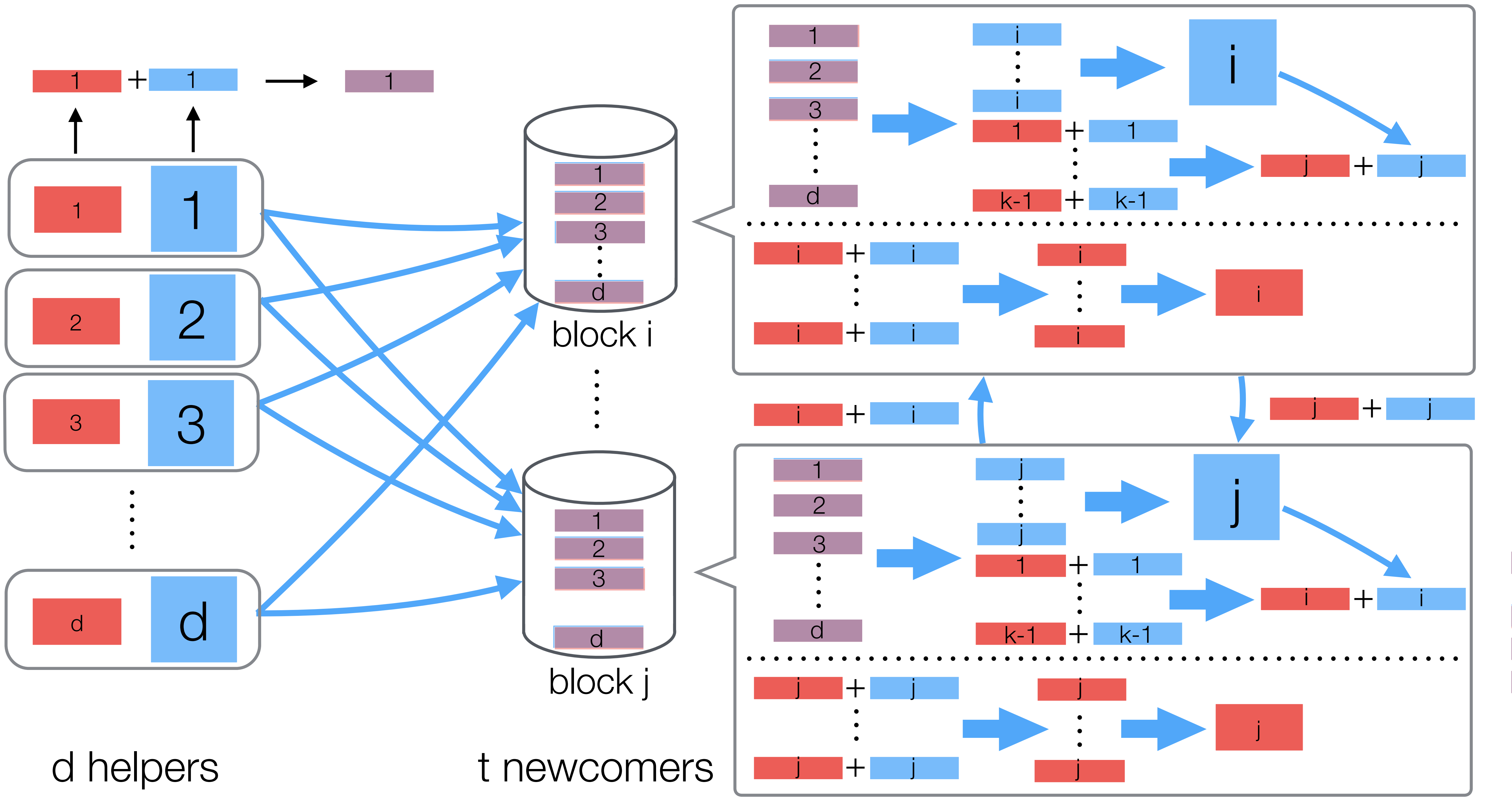
Reconstruction



Reconstruction



Reconstruction



Evaluation

- Implement Beehive in C++, as well as RS and MSR codes, with Intel storage acceleration library
- Run performance evaluation on Amazon EC2 (c4.2xlarge) instances
- Encode a file of 360 MB (RS & MSR codes) or 350 MB (Beehive codes), with k = 6, r = 6
- Compare network transfer and disk I/O

Highlights of Results

- Network Transfer
  - Beehive can save more traffic than MSR codes (up to 42.9%)
  - Network transfer per newcomer reduces with both d and t
- Disk I/O
  - Beehive codes save disk read by up to 75%
- Similar performance throughput of reconstruction
  - RS codes achieve a higher throughput of encoding and decoding due to its low complexity

Conclusions

- We present Beehive codes, erasure codes that achieve the optimal network transfer to reconstruct multiple blocks in batches
- The construction of Beehive codes can be applied with a wide range of values of system parameters
- Implemented in C++, we demonstrate that Beehive can save both disk I/O and network transfer during reconstruction

Thanks!