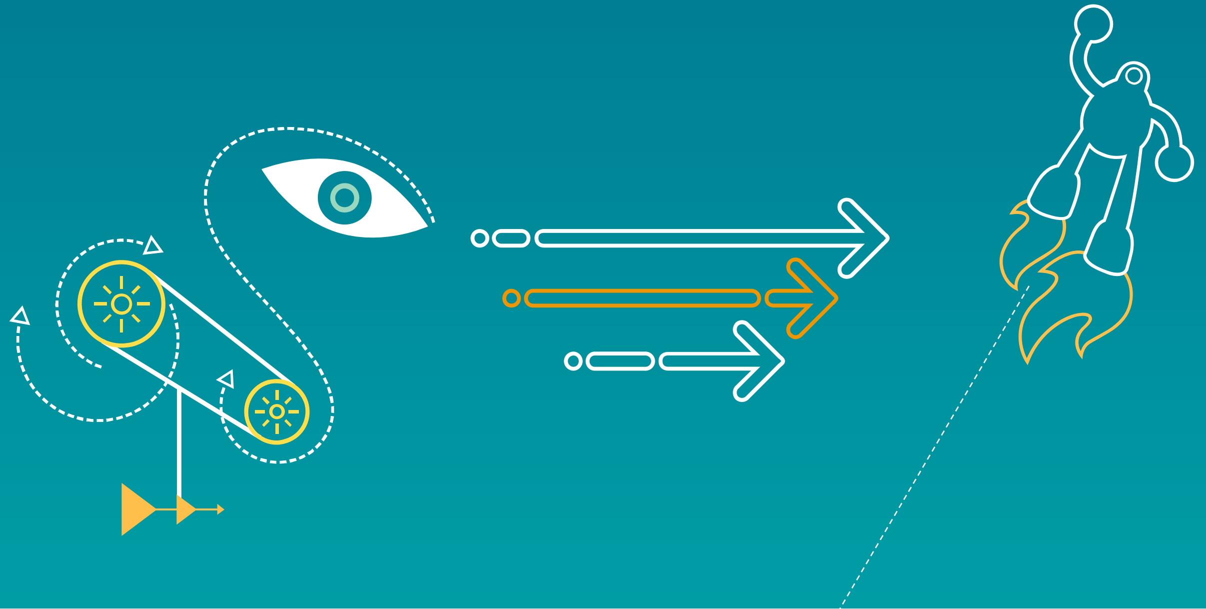
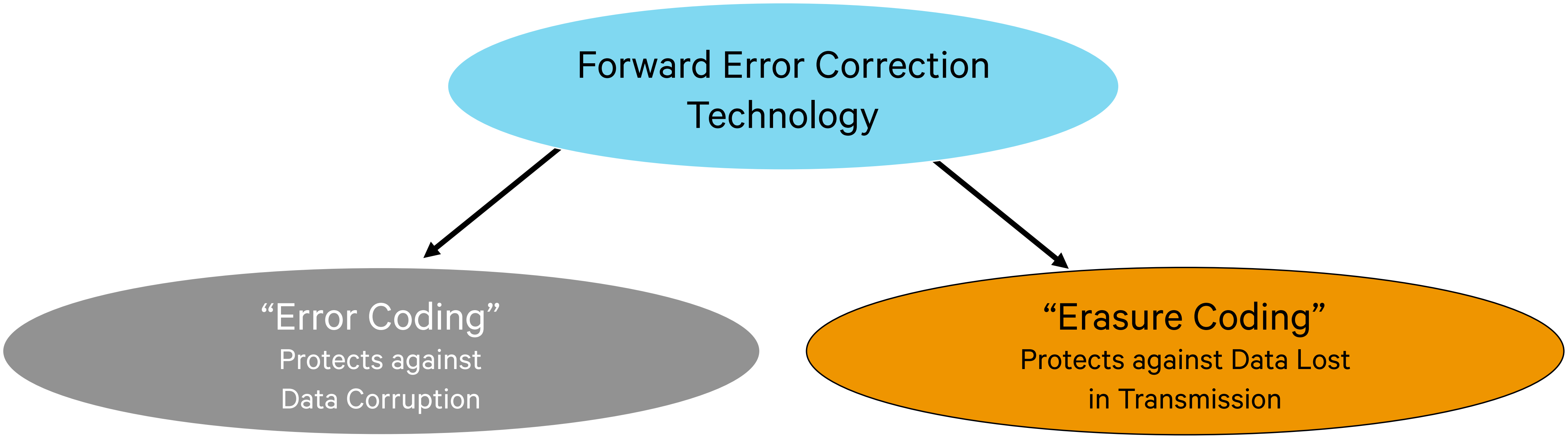
Coding theory for scalable media delivery

Michael Luby



RaptorQ is a product of Qualcomm Technologies, Inc.

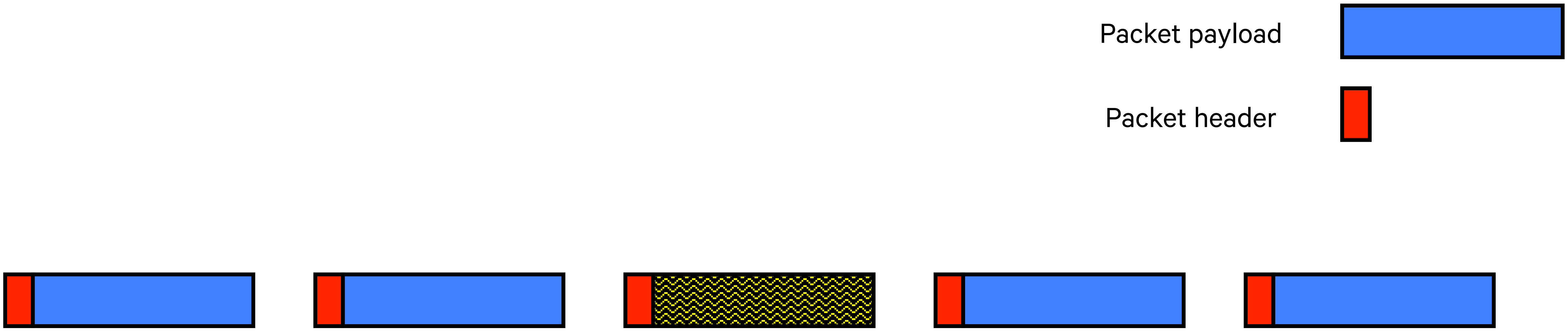
Application layer erasure coding complements traditional error coding



- Vast majority of current use of FEC
- Probably what you’re familiar with
- Typically applied at layers 1 or 2
- Usually performed in hardware
- PHY-FEC (physical layer FEC)

- Commercial application relatively new
- Applied above layer 2
- Complement to Error Coding
- Typically performed in software
- AL-FEC (application layer FEC)

Packet transmission

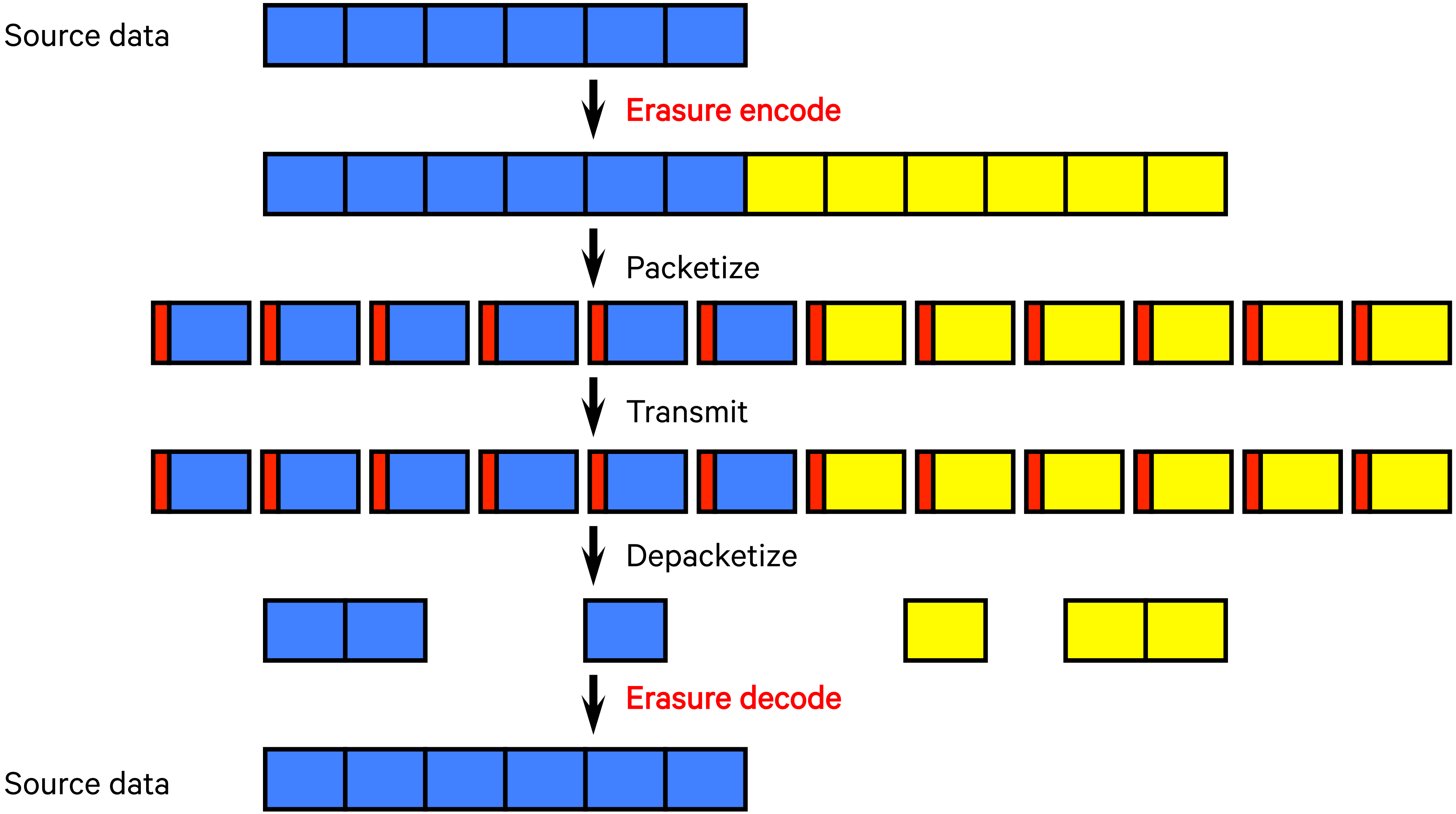


Stream of packets

Received corrupted packet is discarded

Can identify received packet payloads from packet headers

Application Layer erasure codes



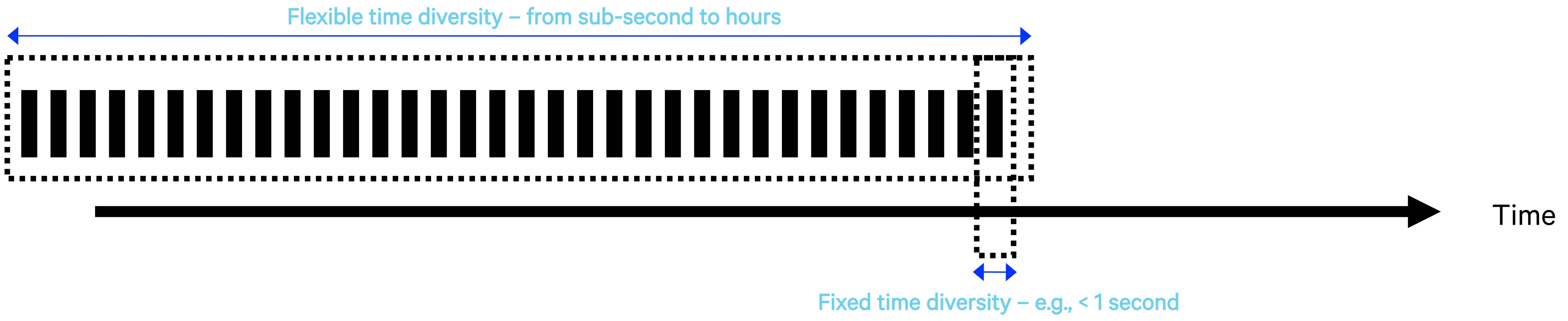
AL-FEC and PHY-FEC are complementary

AL-FEC

Packet loss protection over small to large block

Flexible time diversity

Flexible amount of protection



PHY-FEC

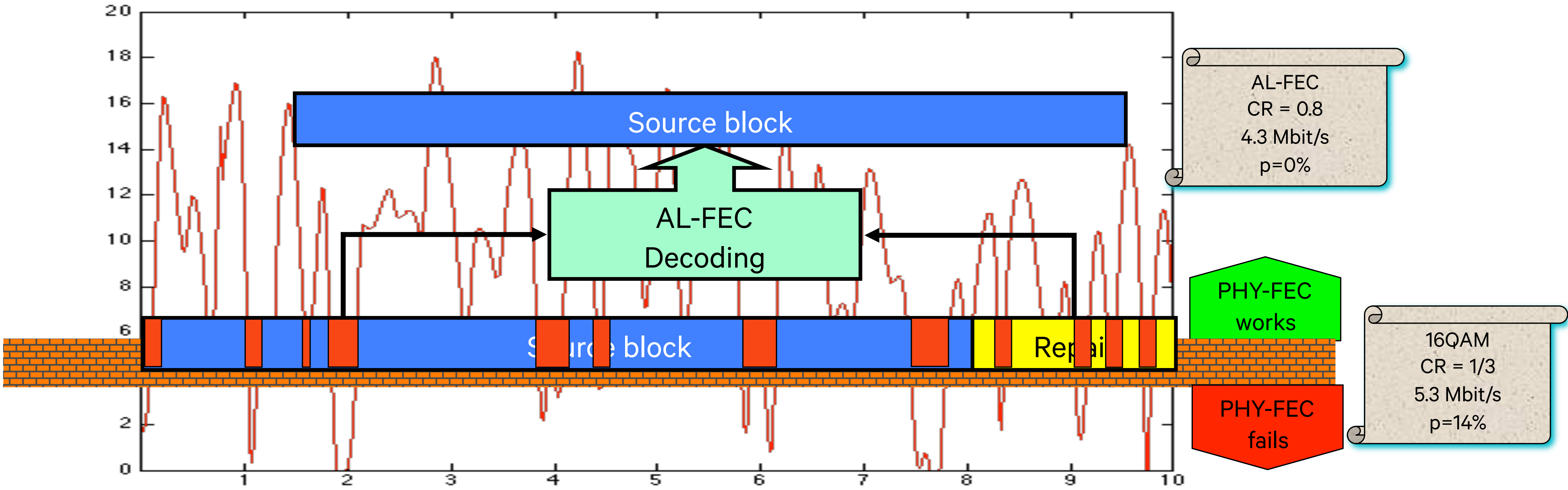
Correct or discard corrupted packet data over small block

Fixed time diversity

Fixed amount of protection

AL-FEC and PHY-FEC working together

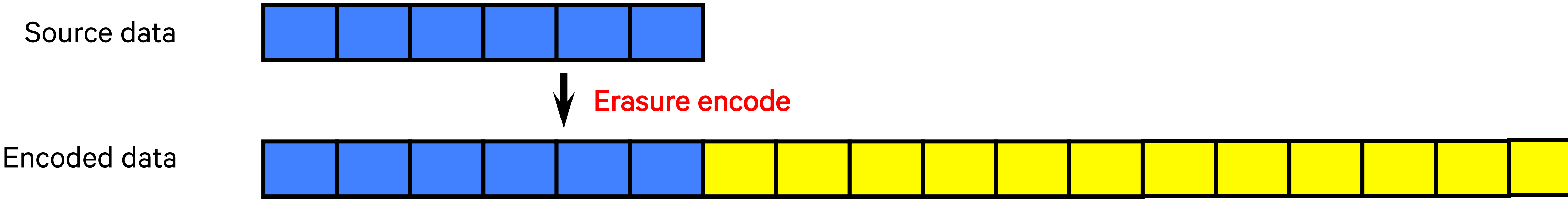
- PHY-FEC corrects noise and interference
- AL-FEC "interleaves" and corrects erasures
  - Longer block length ("interleavers") ➔ better performance



What is a fountain code?

- Generate as much encoding as desired
- Recover source from the minimal possible encoding
  - It doesn't matter what is received or lost
  - It only matters that enough is received

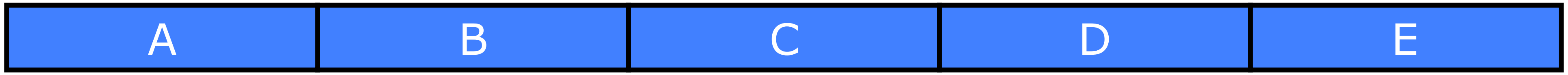
Fountain codes – erasure codes without a rate



- Fountain codes have no predetermined rate

- For fountain codes, for a fixed source data size
  - Erasure code design is extendable to provide any code rate
  - All code rates use the same extendable erasure code design
  - Particular encoded symbols are generated independently of one another
  - Number of encoded symbols that can be generated on the fly is unconstrained

LT encoding

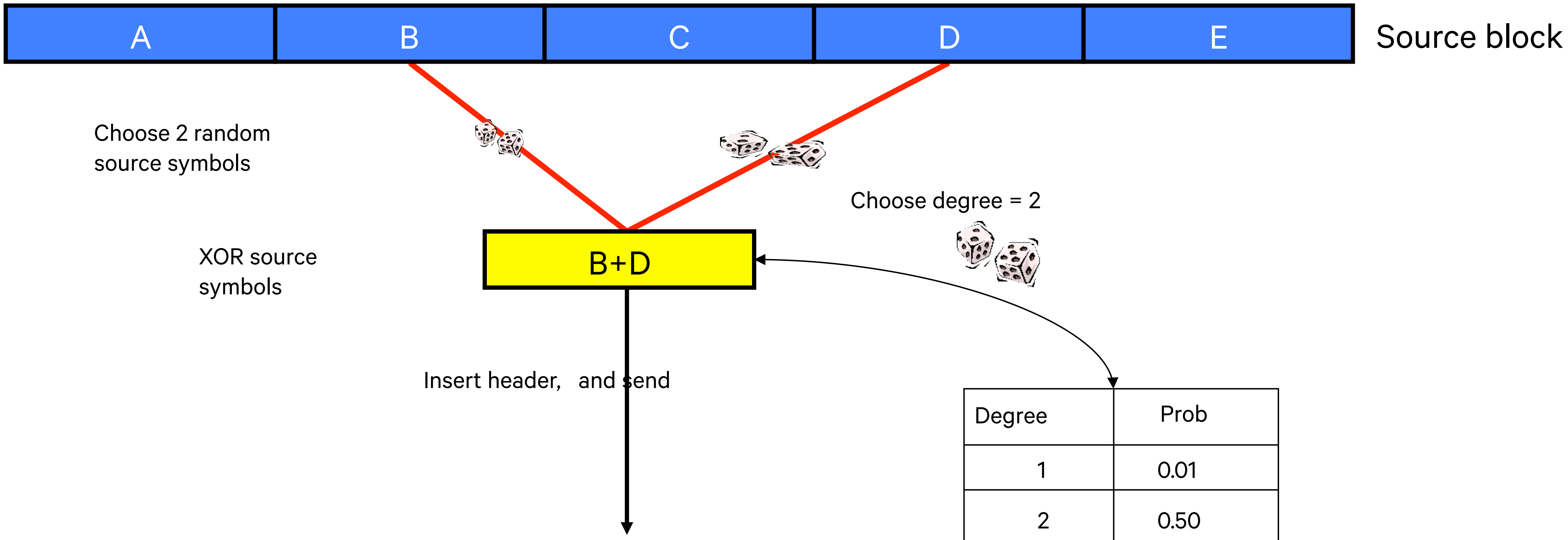


Source block

Degree	Prob
1	0.01
2	0.50
3	0.17
4	0.08
⋮	⋮

Degree Distribution

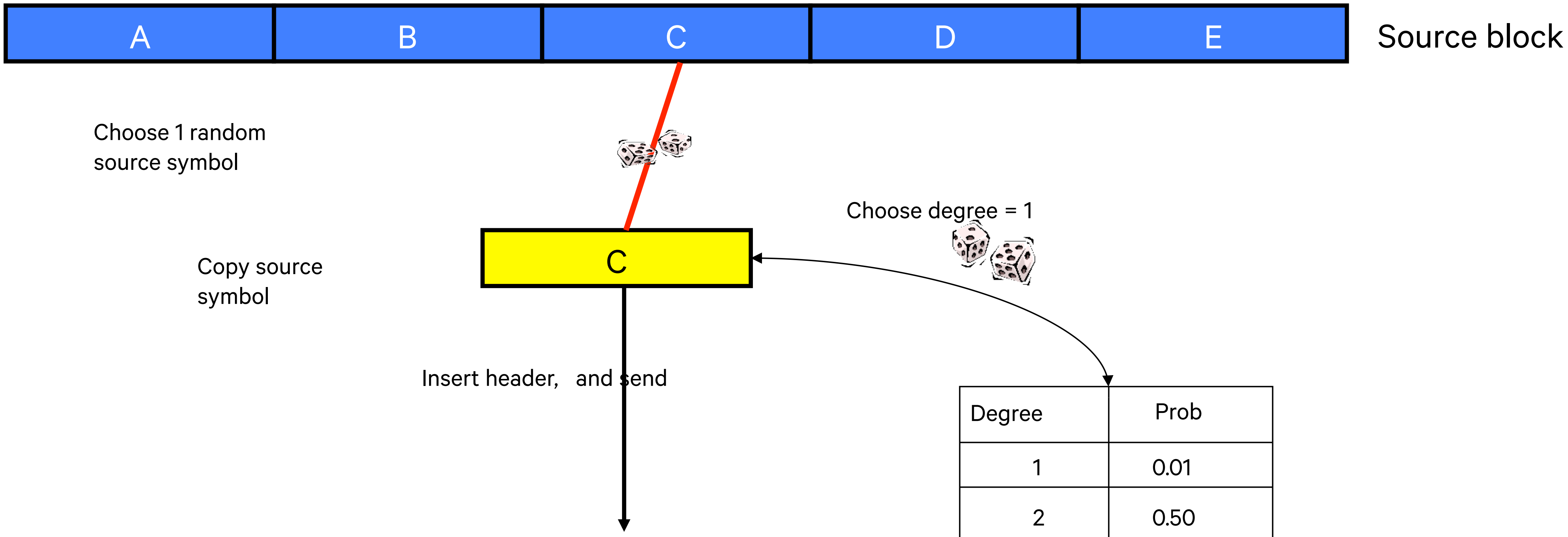
LT encoding



Degree	Prob
1	0.01
2	0.50
3	0.17
4	0.08
⋮	⋮

Degree Distribution

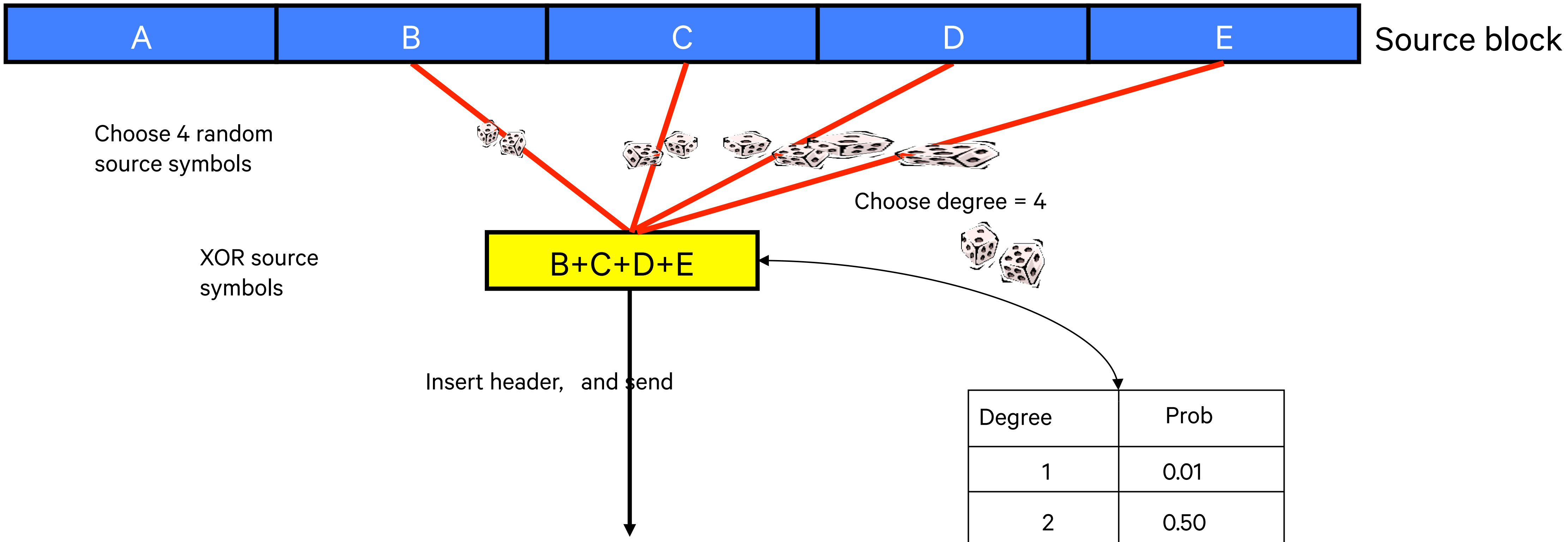
LT encoding



Degree	Prob
1	0.01
2	0.50
3	0.17
4	0.08
⋮	⋮

Degree Distribution

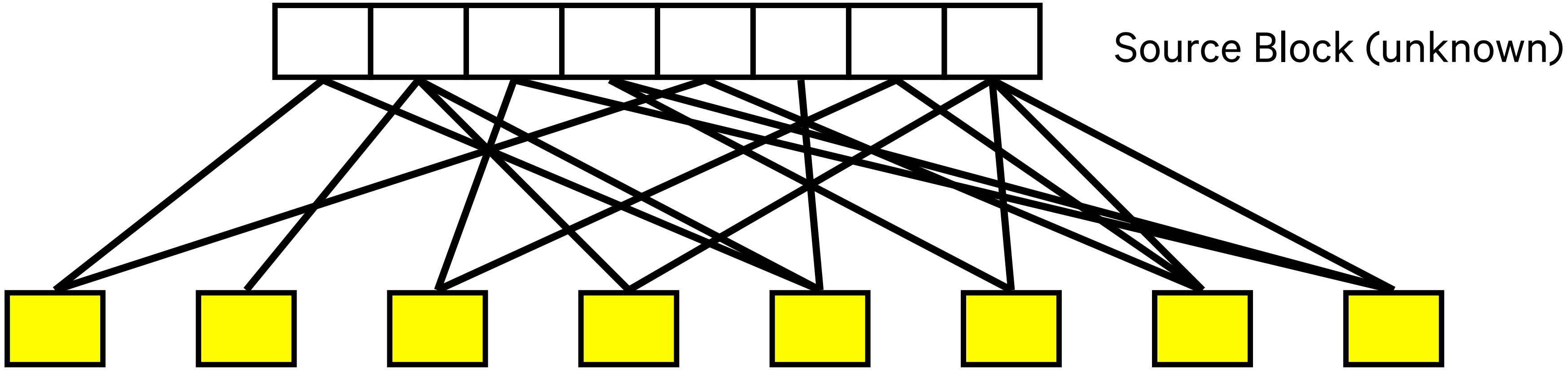
LT encoding



Degree	Prob
1	0.01
2	0.50
3	0.17
4	0.08
⋮	⋮

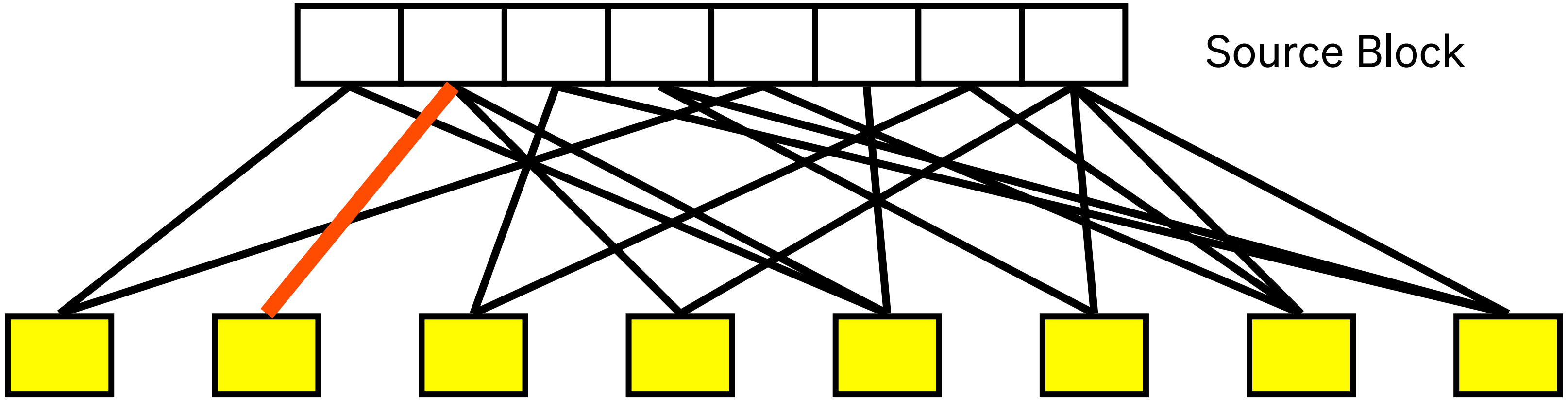
Degree Distribution

Belief propagation decoding



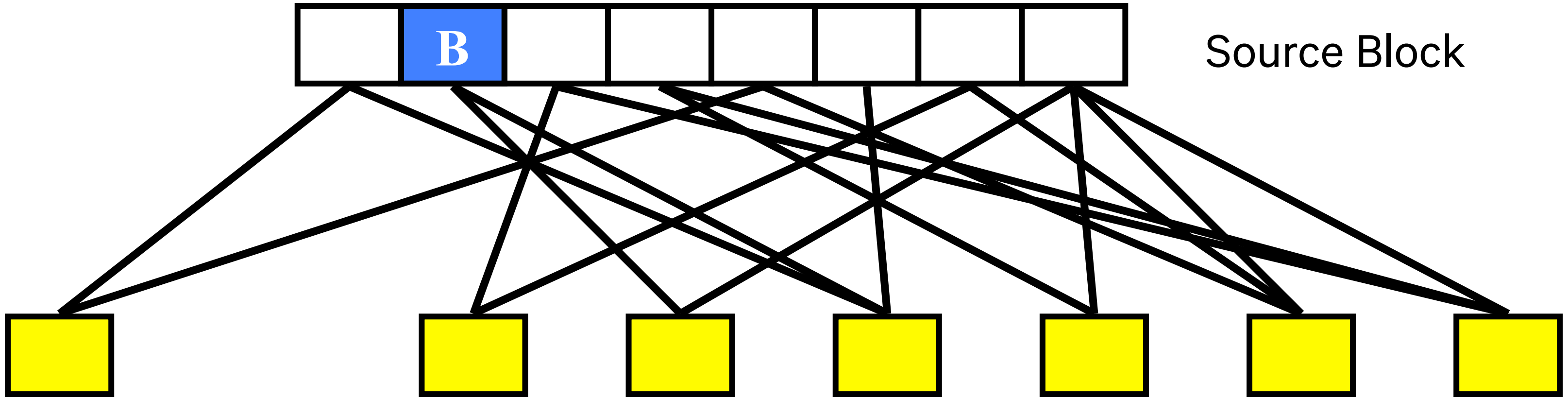
Collect enough encoded symbols and set up graph
between encoded symbols and source symbols to be decoded

Belief propagation decoding



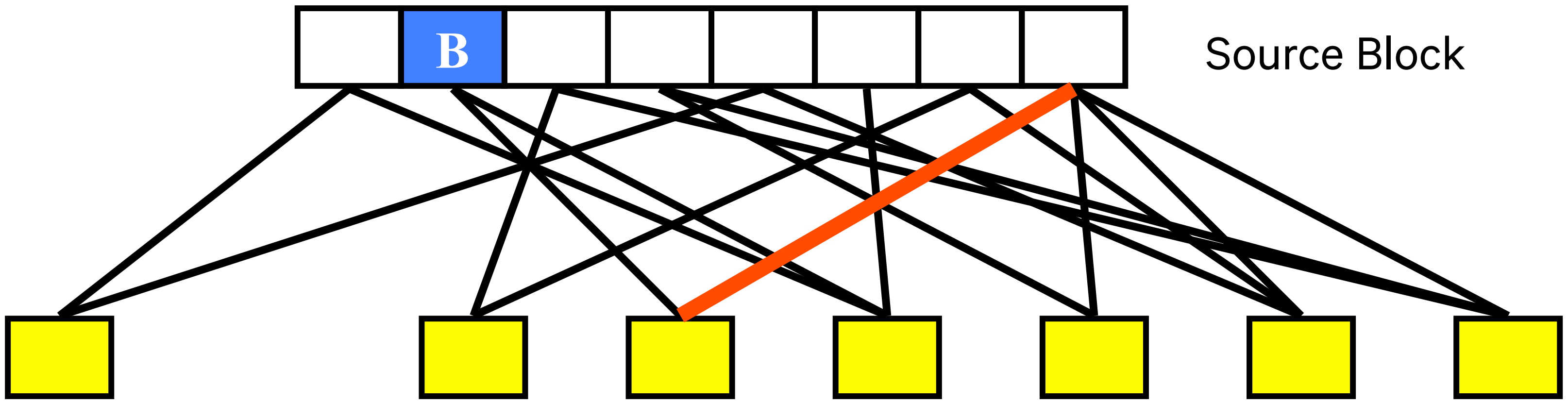
Identify encoded symbol with one unrecovered neighbor
STOP if none exists

Belief propagation decoding



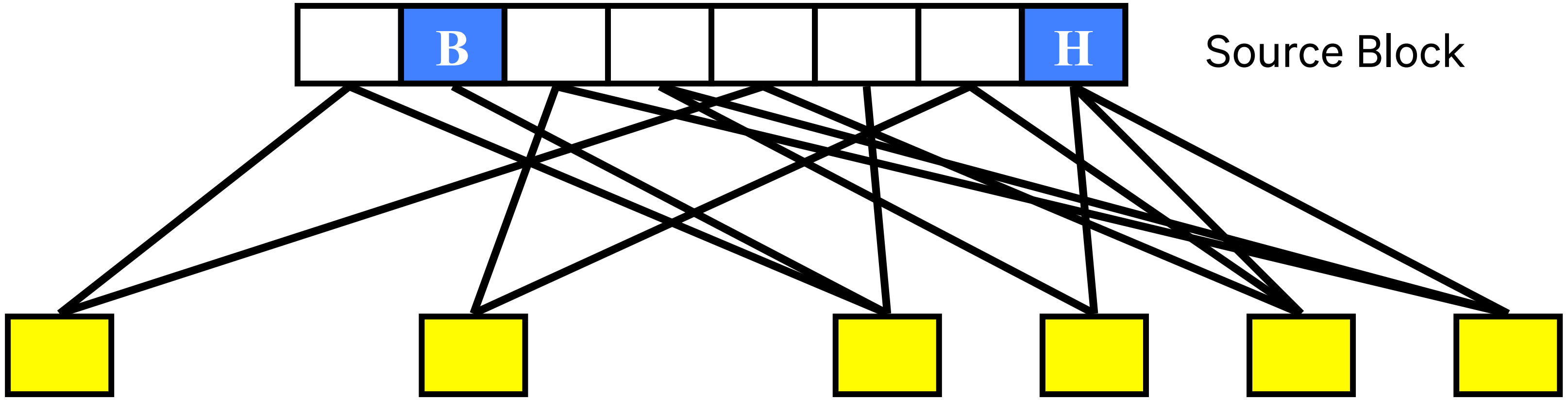
Unrecovered source symbol value is the value of all recovered neighbors XORed into the encoded symbol value

Belief propagation decoding



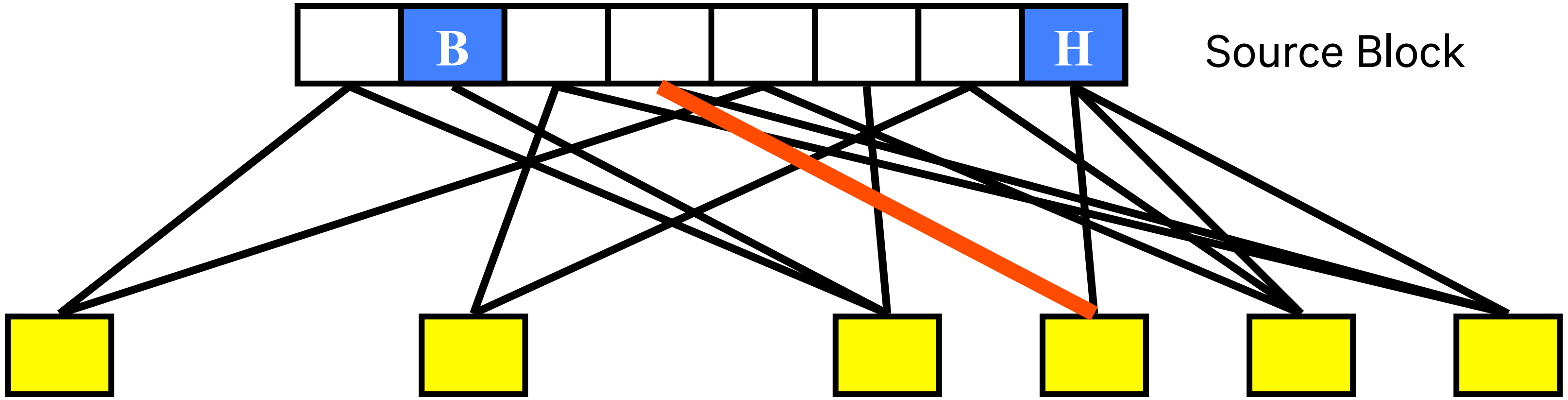
Identify encoded symbol with one unrecovered neighbor
STOP if none exists

Belief propagation decoding



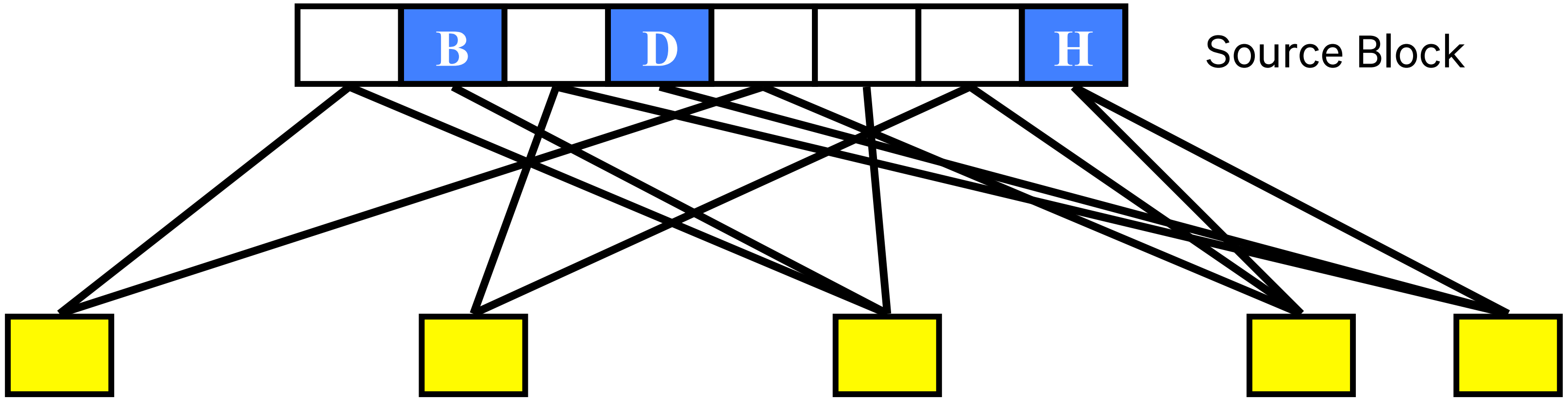
Unrecovered source symbol value is the value of all recovered neighbors XORed into the encoded symbol value

Belief propagation decoding



Identify encoded symbol with one unrecovered neighbor
STOP if none exists

Belief propagation decoding



Unrecovered source symbol value is the value of all recovered neighbors XORed into the encoded symbol value

Belief propagation decoding

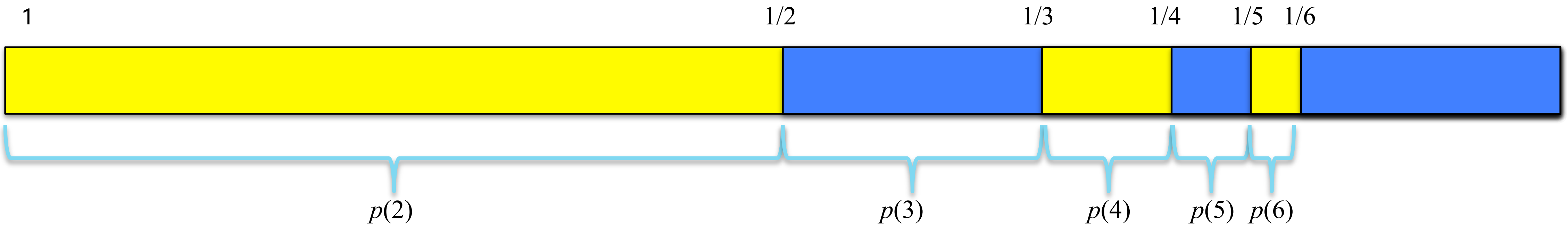
A	B	C	D	E	F	G	H

Source Block (recovered)

Intuition for Soliton degree distribution

- Consider a symbol of degree d
  - Releases when exactly 1 of its d neighbors remains unrecovered
  - Degree d releases when 1/d fraction of the symbols remain to be decoded
- A probability distribution on degrees so release distribution is uniform
  - p(d) "covers" the interval 1/(d-1) to 1/d of the uniform distribution
  - Length of interval 1/(d-1) to 1/d is $\frac{1}{d \cdot (d-1)}$
  - For d = 2, 3, …

$$p(d) = \frac{1}{d \cdot (d-1)}$$



Choosing from Soliton distribution

$$y \in_R [0,1]$$

$$d = \begin{cases} \left\lceil \dfrac{1}{y} \right\rceil & \text{if y} \geq \dfrac{1}{k} \\ 1 & \text{if y} < \dfrac{1}{k} \end{cases}$$

Raptor codes in standards

- Raptor codes (IETF RFC 5053, 3GPP, DVB, ITU, ATIS)
  - Systematic fountain codes
  - Linear time encoding and decoding
  - Standardized – 3GPP MBMS, DVB-H IPDC
  - Good recovery properties – like a random code over GF(2)
  - Good flexibility
    - Up to 8,192 source symbols
    - Up to 65,384 source + repair symbols

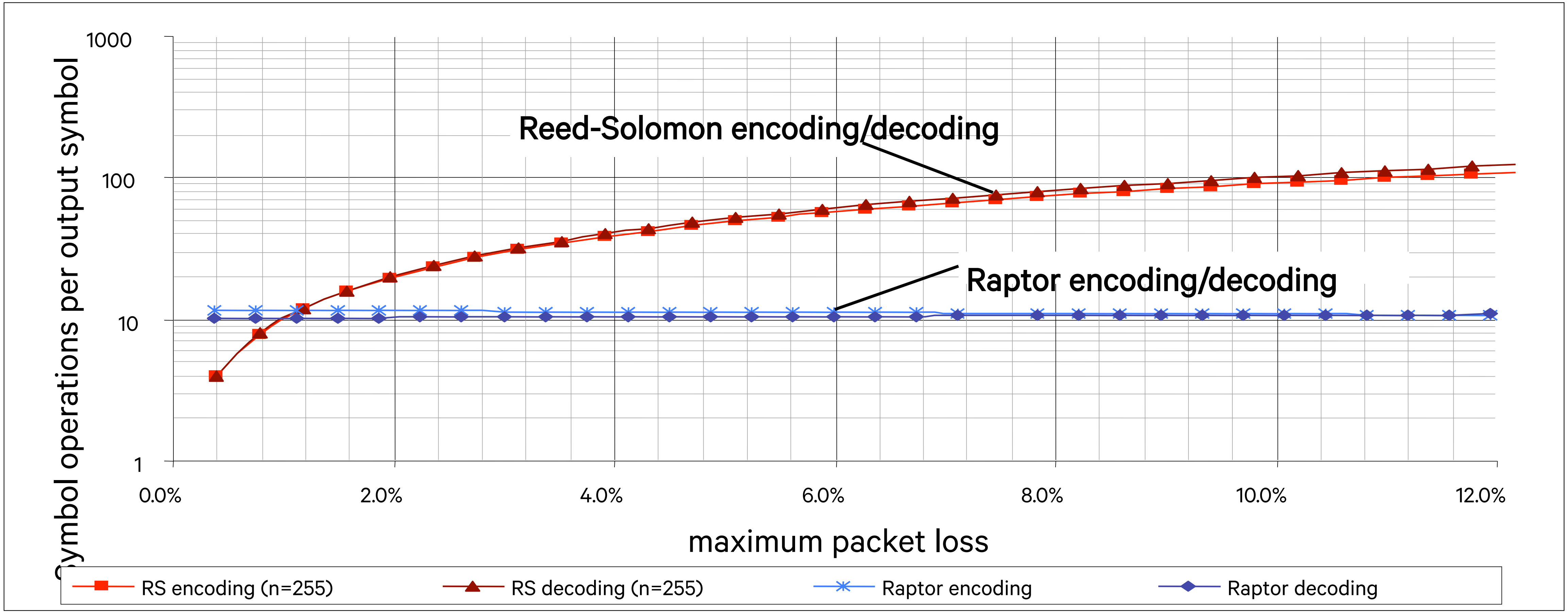
- RaptorQ codes (IETF RFC 6330)
  - Systematic fountain codes
  - Linear time encoding and decoding
  - Great recovery properties – like a random code over GF(256)
  - Great flexibility
    - Up to 56,403 source symbols
    - Up to 16,777,216 source + repair symbols (essentially unlimited)

Major **technical** features (first appearance)

- LT code (Raptor RFC 5053)
  - fountain property
- Pre-coding (Raptor RFC 5053)
  - linear time
- Inactivation decoding (Raptor RFC 5053)
  - linear time
- Systematic construction (Raptor RFC 5053)
  - encoding includes original source
- Larger finite fields (RaptorQ RFC 6330)
  - reduced reception overhead
- Permanent inactivations (RaptorQ RFC 6330)
  - reduced reception overhead

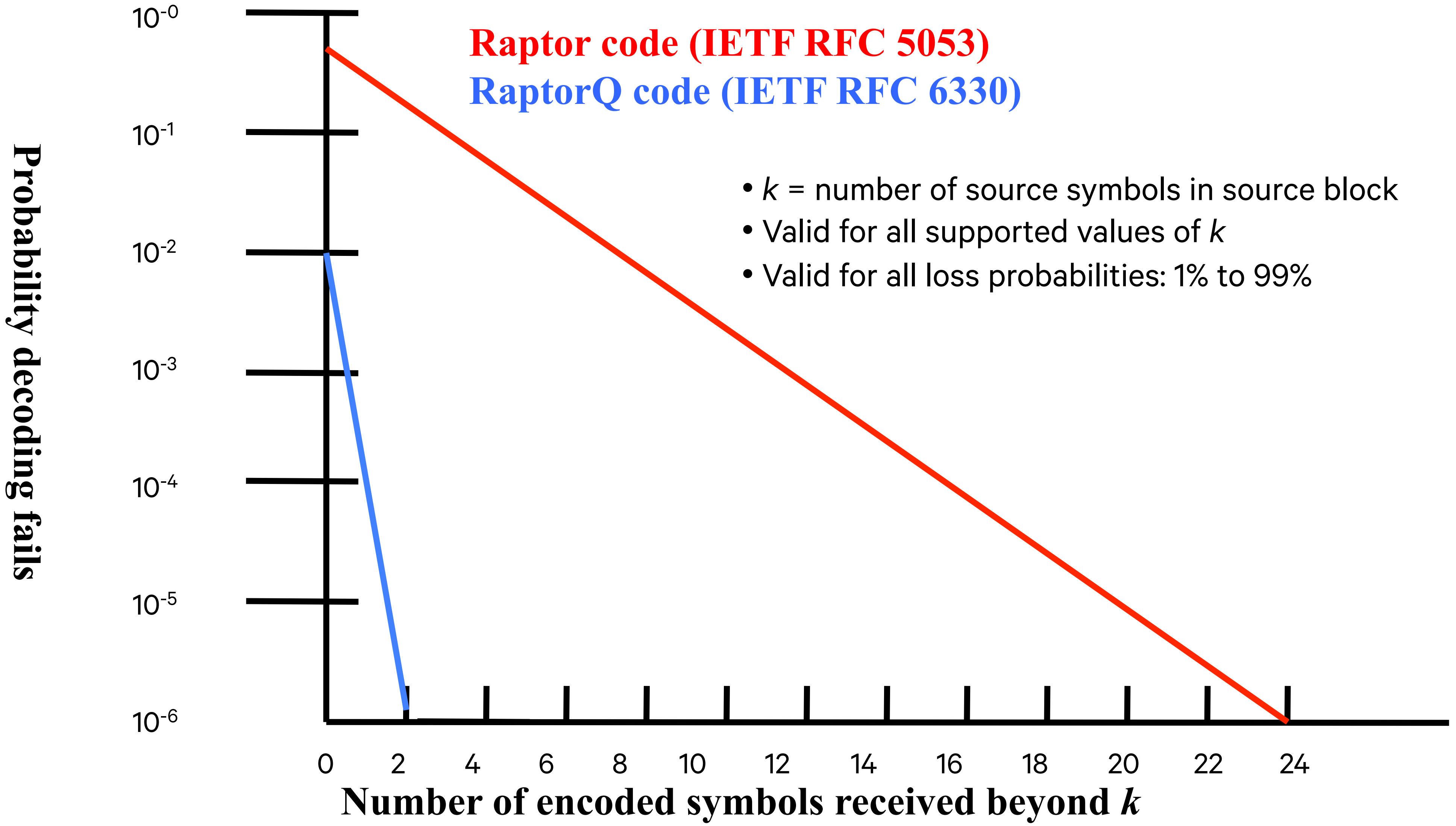
Raptor Codes. Foundations and Trends in Communications and Information Theory
A. Shokrollahi, M. Luby
2011, Vol. 6: No 3-4, pp 213-322.

Raptor codes computational complexity



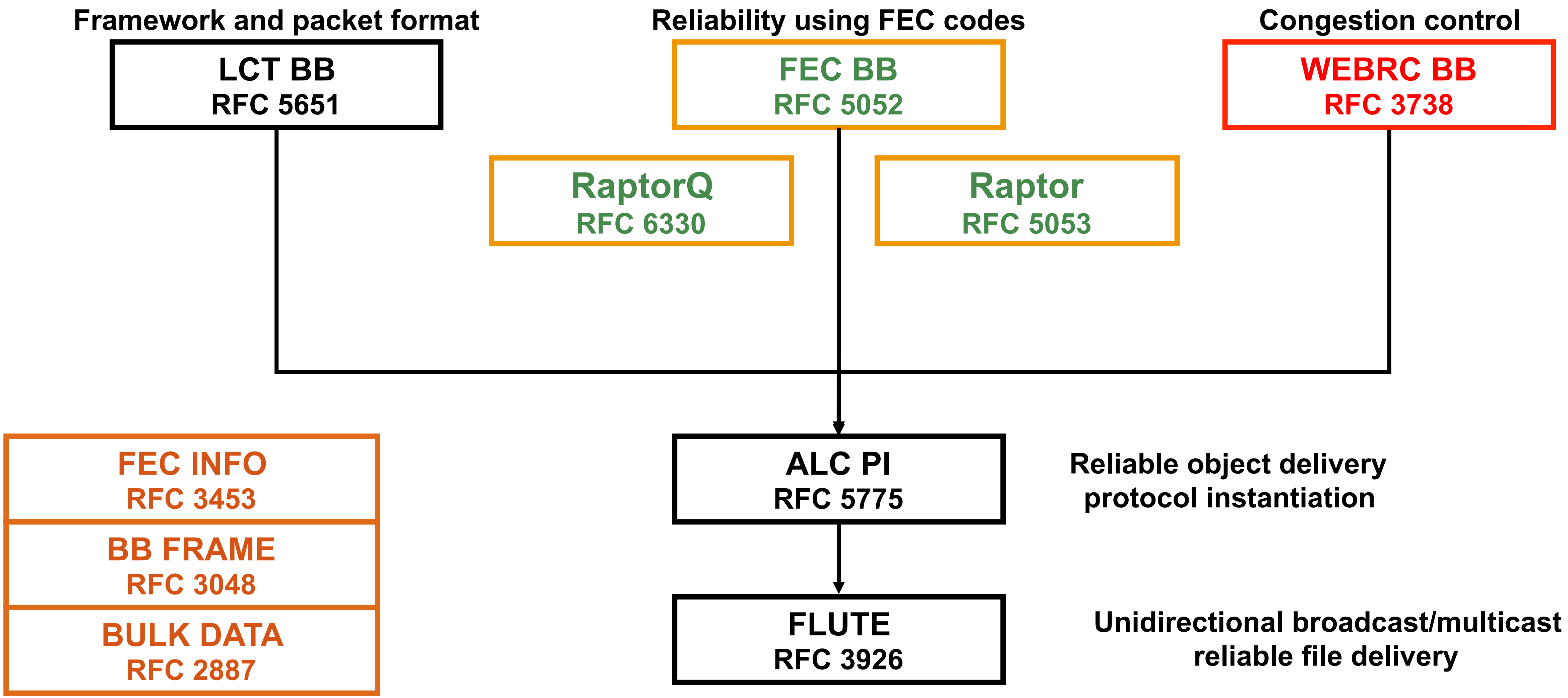
In comparison to other typical alternative FEC technologies, Raptor codes are an order of magnitude or more less complex

Raptor codes overhead

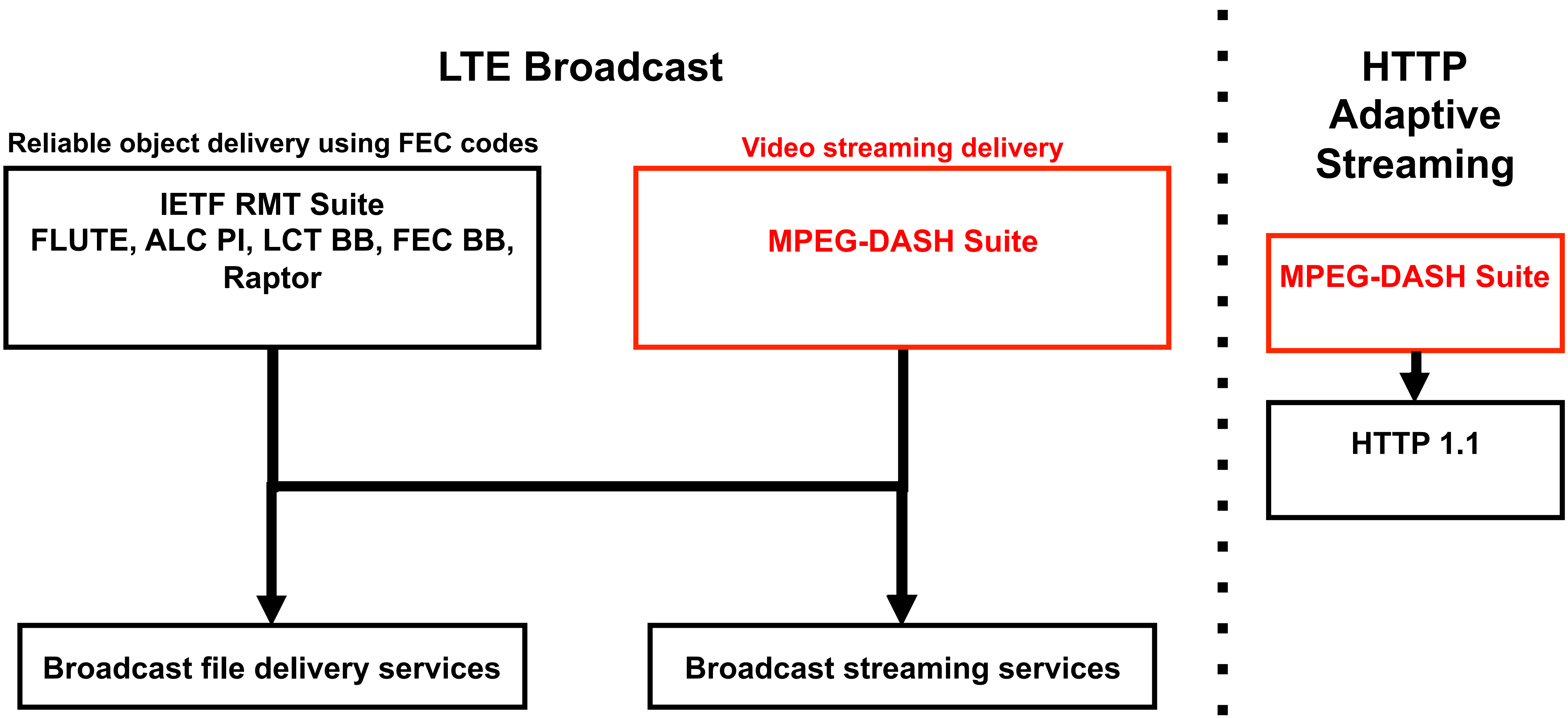


- k = number of source symbols in source block
- Valid for all supported values of k
- Valid for all loss probabilities: 1% to 99%

IETF RMT Broadcast/Multicast Object Delivery Suite



3GPP LTE Broadcast (eMBMS) Service Layer



Why Fountain codes?
Some applications that you may (not) have thought about

Mobile File Delivery Services Over Cellular Network

Challenge: reliable file delivery to mobile devices

Applications:

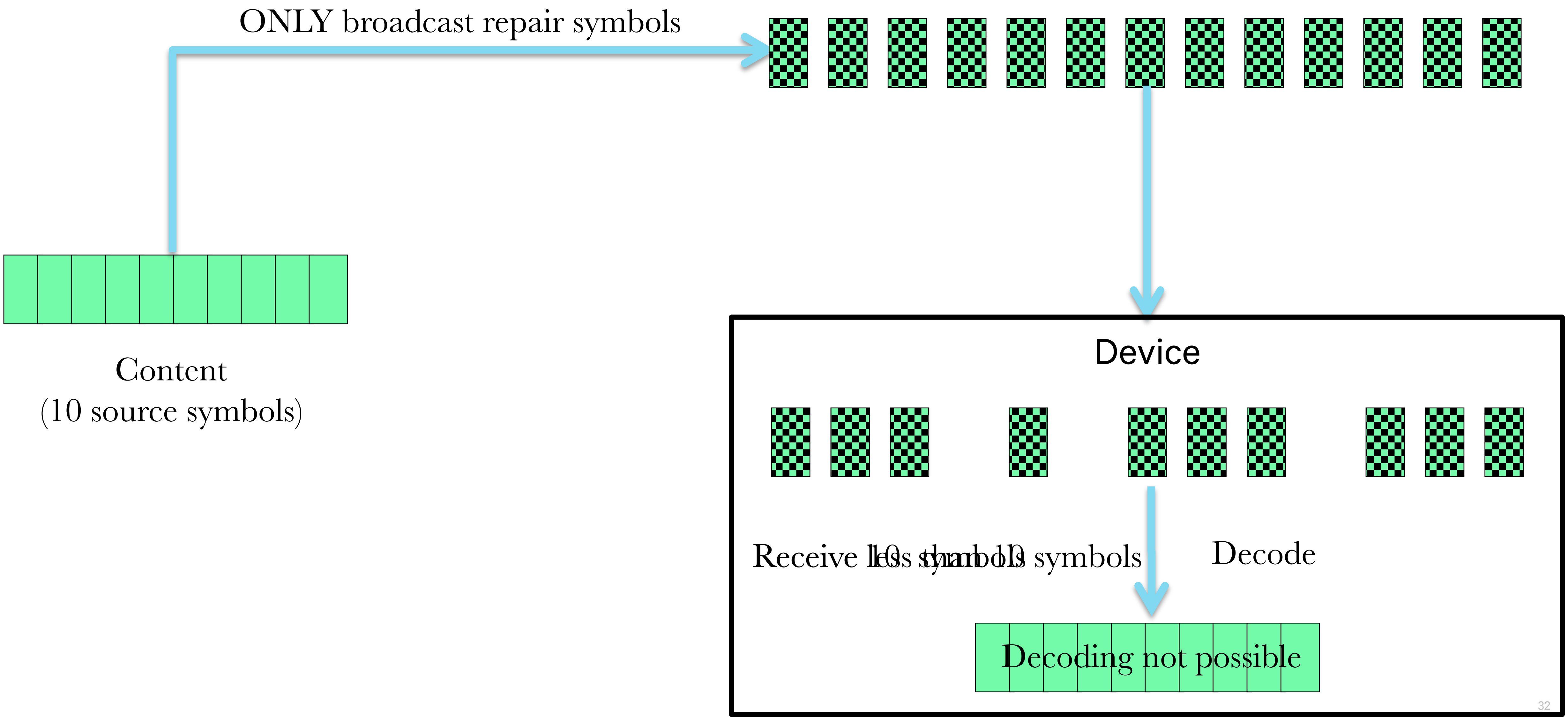
- Streaming
- Delivery of popular content
  - Media
  - Games

LTE broadcast (eMBMS)

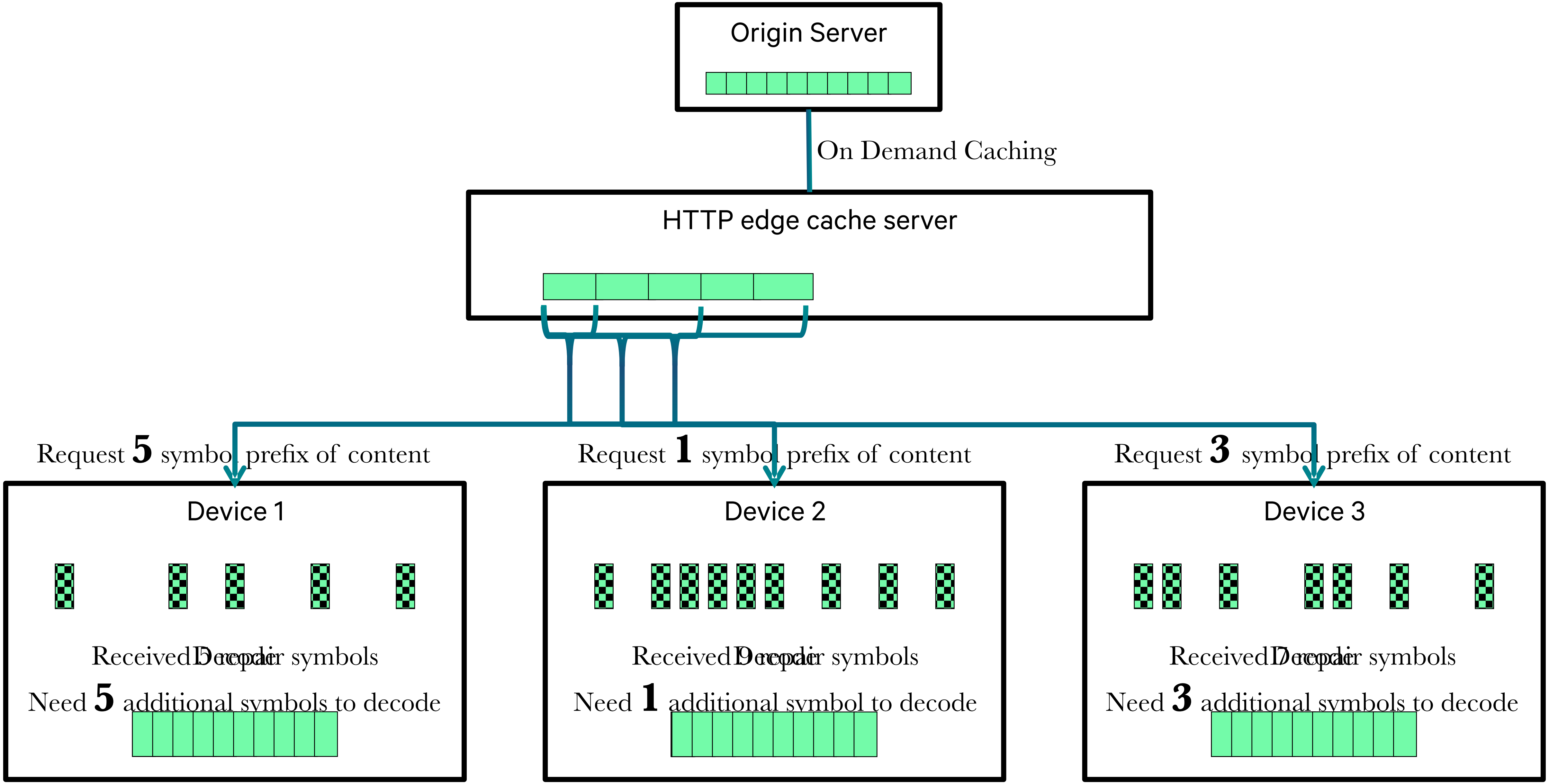
Receivers

- Millions of mobile devices

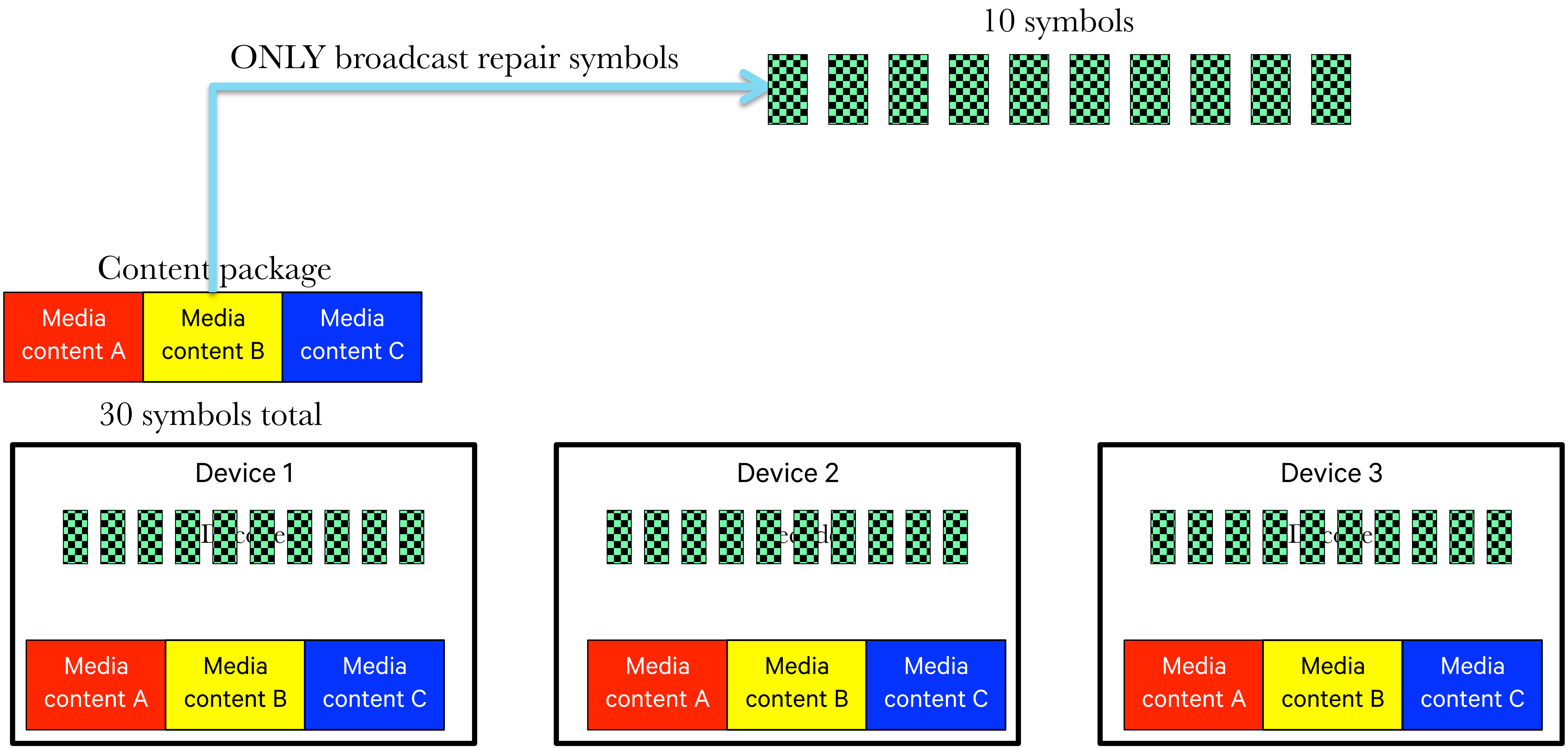
LTE broadcast offload service for HTTP



File download completion using HTTP 1.1 byte range requests



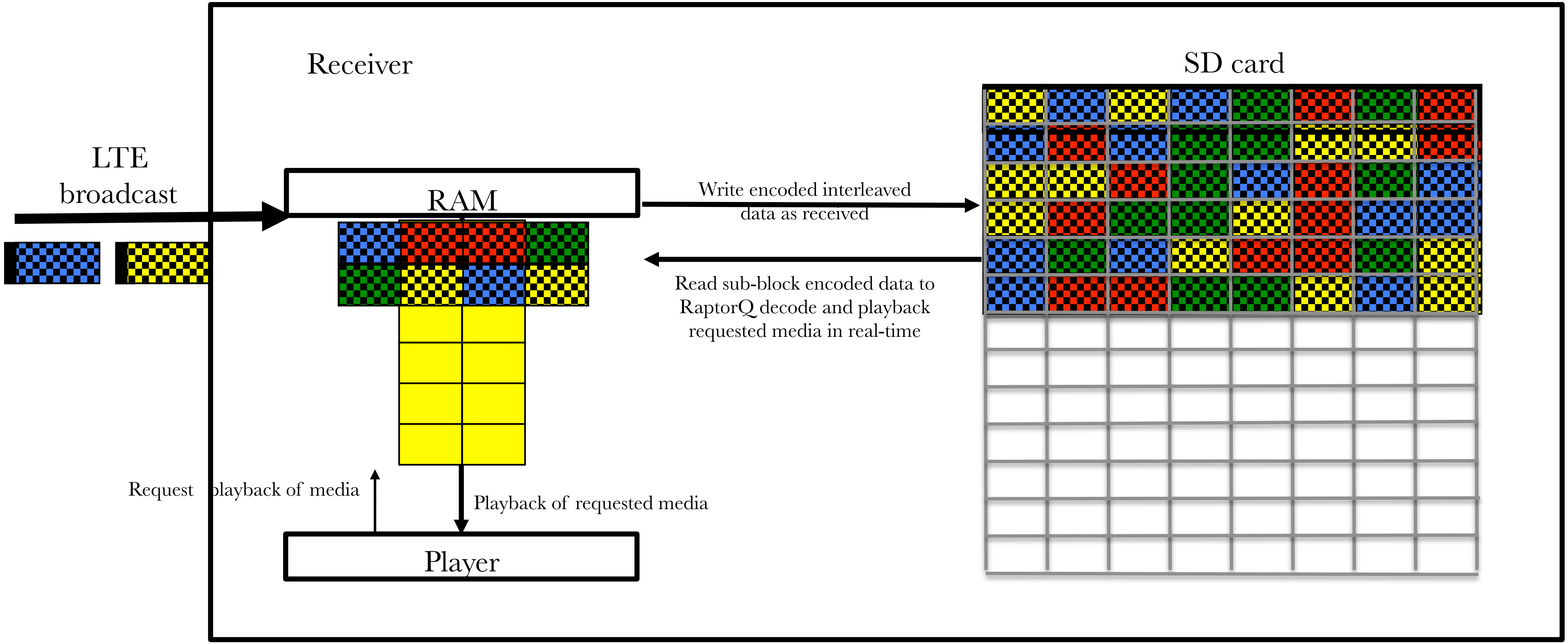
LTE broadcast offload service for HTTP



Just-in-Time recovery

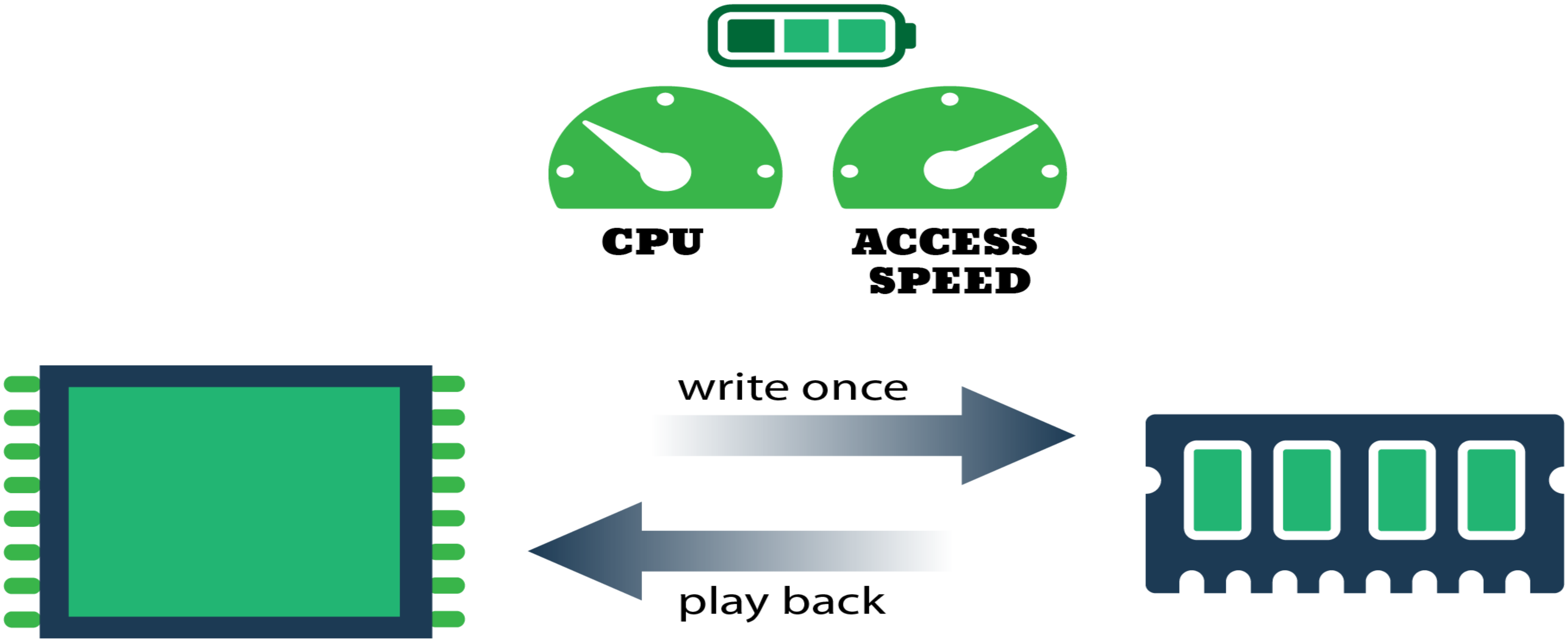
- LTE broadcast data stored directly on the SD card
  - Original multimedia data is never stored on the SD card
  - SD card stores one copy of file – not two!
- Multimedia content available immediately after reception
  - Avoids FEC decode post-processing of file after reception
- Just-in-Time recovery
  - Based on user actions – player requests data to playback multimedia content
  - Relevant data read from SD card, FEC decoded, provided directly to player
  - Trick play response time
    - Playback starts after one sub-block of data read from SD card and FEC decoded
    - Size of sub-block and decode speed determines the response time

Just-in-Time recovery



Just-in-Time recovery avoids costly post-processing
Portions of multimedia never played back are never processed
Storage usage is minimized – avoids double the writes to SD card

Just-in-Time recovery advantages



Good user experience – media available immediately after reception
Minimizes UE CPU, I/O, and storage resources

Just-in-Time recovery demo

- Demonstrates
  - Just-in-Time recovery (importance of sub-blocking to support this)
  - Only repair symbols sent in the original broadcast session
    - Provides ability to efficient combine with HTTP-based repair service
    - Provides ability to provide broadcast/HTTP hybrid services
- Demo parameters
  - Elephant's Dream – 91.3 MB file
  - Partitioned into 10 source blocks (each of size 9.13 MB)
    - Provides reasonable network efficiency
  - Each source block is partitioned into 41 sub-blocks (each of size 223 KB, Symsize = 36 bytes, K ~ 6300)
    - Read encoded data from SD card and decode sub-block when requested by app
  - Original broadcast session
    - Transmit repair symbols only
    - 20% packet loss according to Markov model applied before reception

For more information on Raptor products or to request your free RaptorQ™ Evaluation Kit please go to www.qualcomm.com/raptor

For questions please email <raptorsupport@qti.qualcomm.com>

Thank you

Follow us on:

For more information on Qualcomm, visit us at:
www.qualcomm.com & www.qualcomm.com/blog