

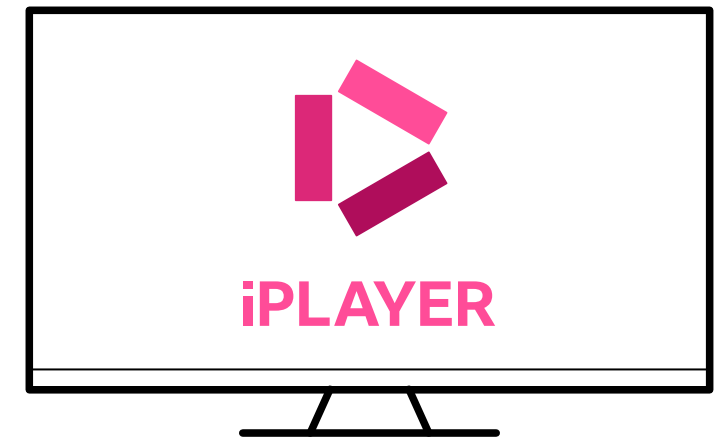
# Towards a higher throughput media CDN

Jonathan Heathcote • BBC Research & Development • NetUK 2024

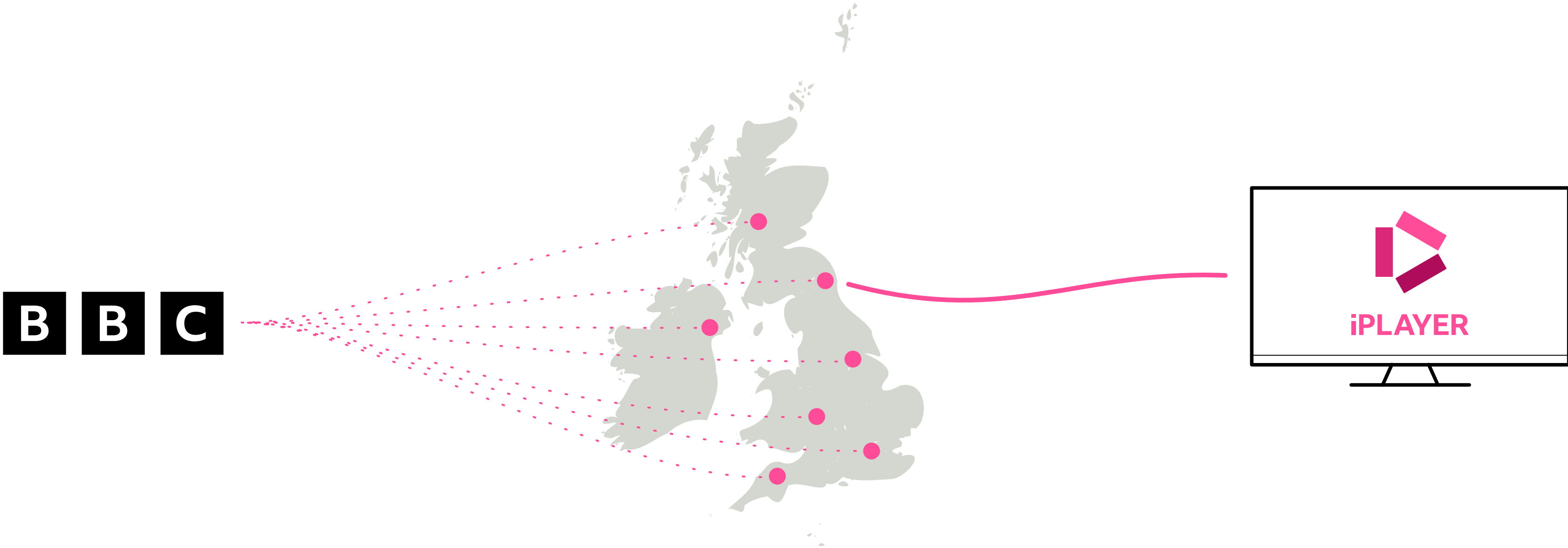


# The BBC's Media CDN

---



# The BBC's Media CDN

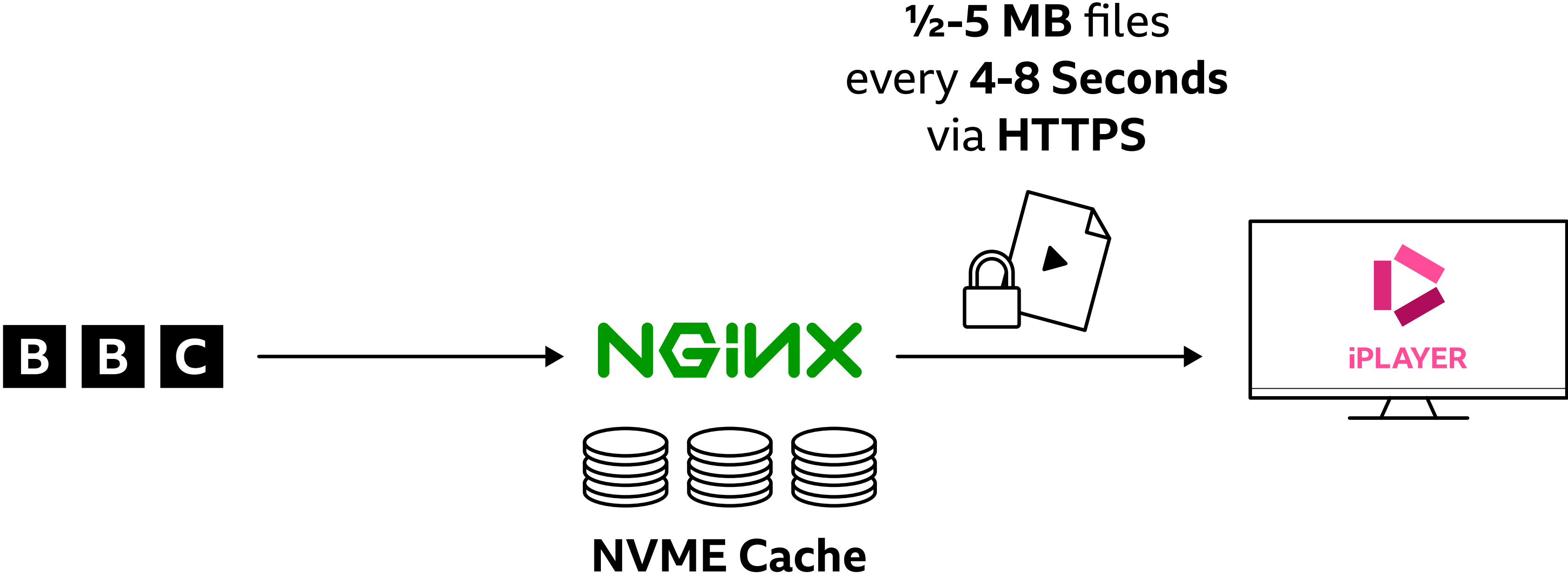


# The BBC's Media CDN





# Edge cache workload



# See also...

BBC Design + Engineering

BIDI GROWTH SO FAR

### BIDI BATTERY: HIGH AVAILABILITY

19

UKNOF virtual November 2020

with co-host BT

<https://uknof.uk/virtualNovember2020>

203.0.113.0  
VIP (single)

ECMP

Servers advertise same VIP over BGP

203.0.113.2

203.0.113.2

203.0.113.2

N+1 resilience

patrons 2020

PRINCIPAL BT LONAP

PREMIUM CenturyLink IP Market Group

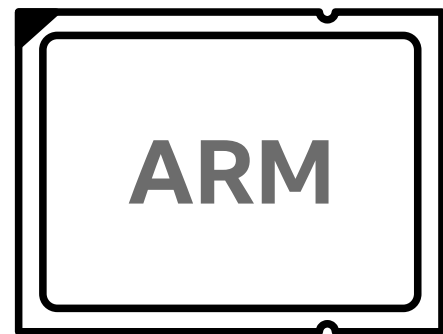
FLEXPOTIX RIPE NCC



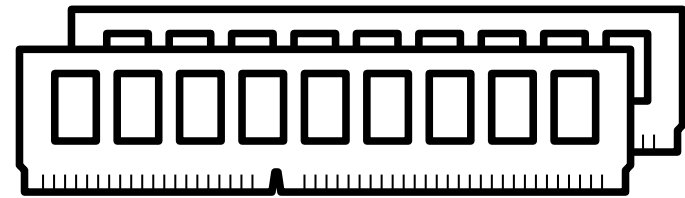
“Building the BBC's media CDN” by Alistair Wooldrige from UKNOF 2020

# This talk

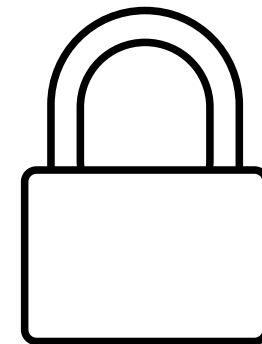
---



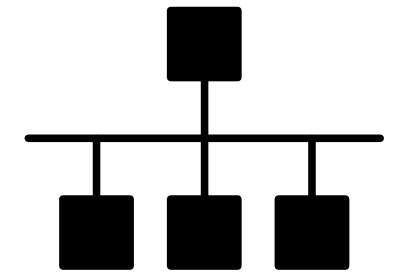
**ARM  
Servers**



**Memory  
Bandwidth**



**kTLS &  
Offload**

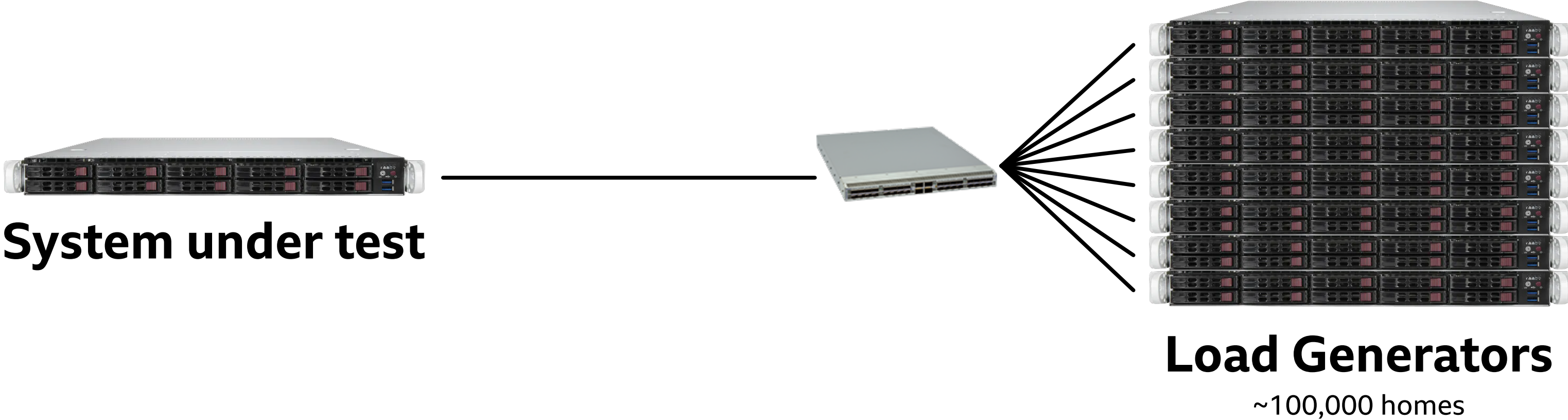


**Network  
Behaviour**

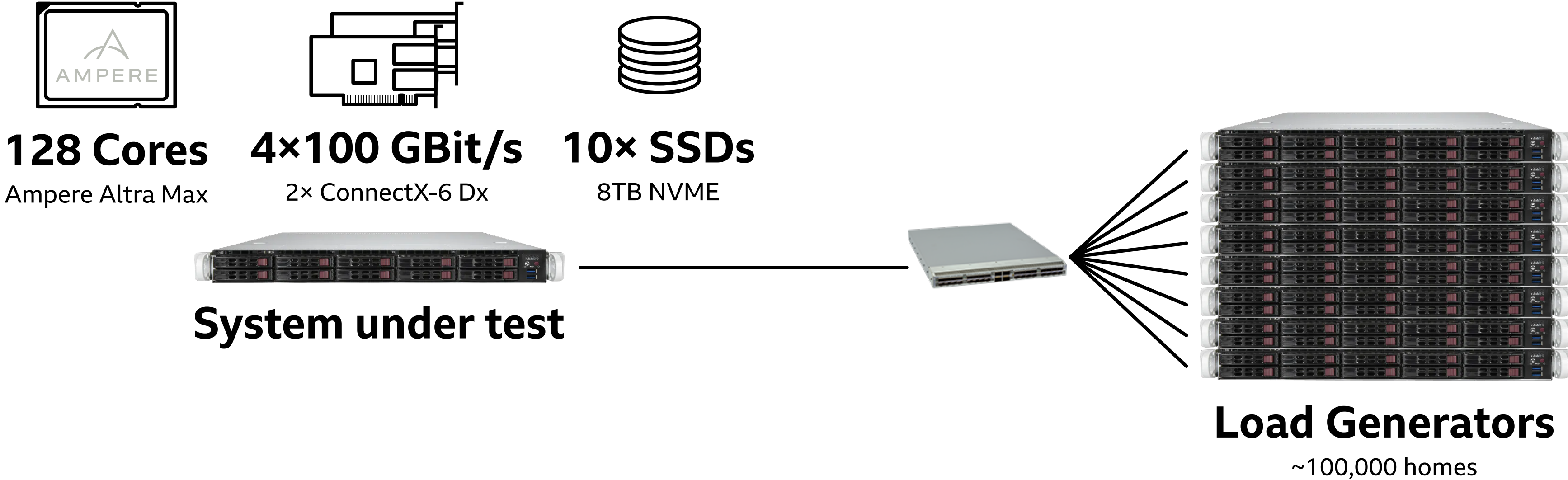
# ARM Servers

# Test lab

---




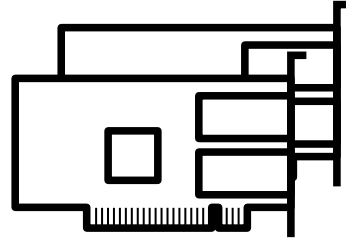
# Test lab

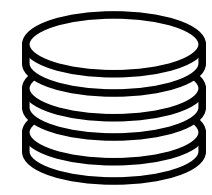




# Test lab

  
**128 Cores**  
Ampere Altra Max

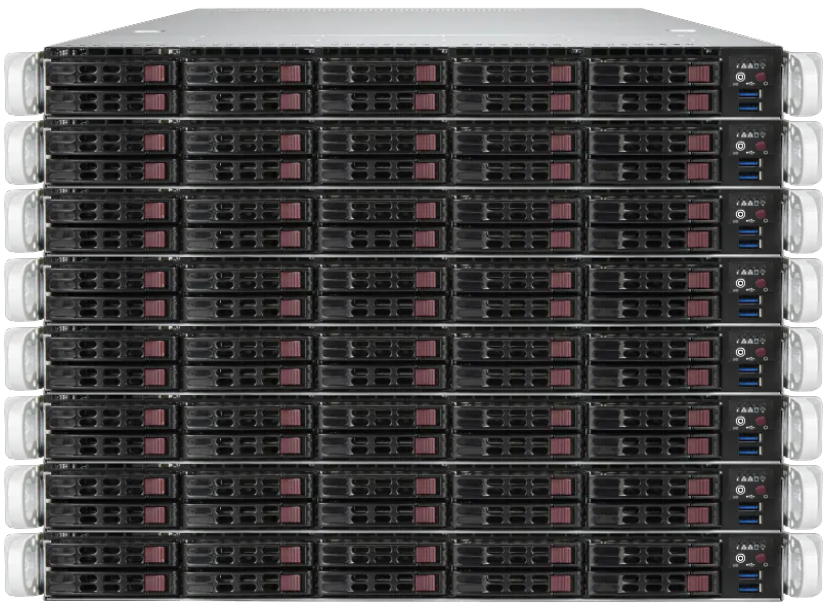
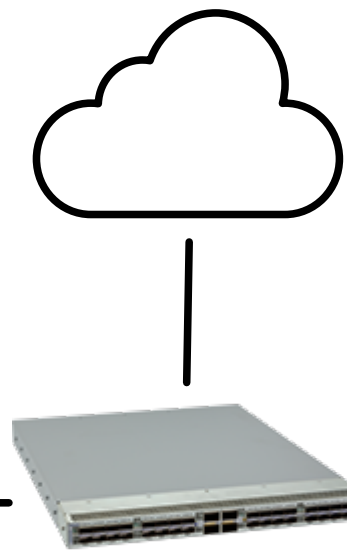
  
**4×100 GBit/s**  
2× ConnectX-6 Dx

  
**10× SSDs**  
8TB NVME



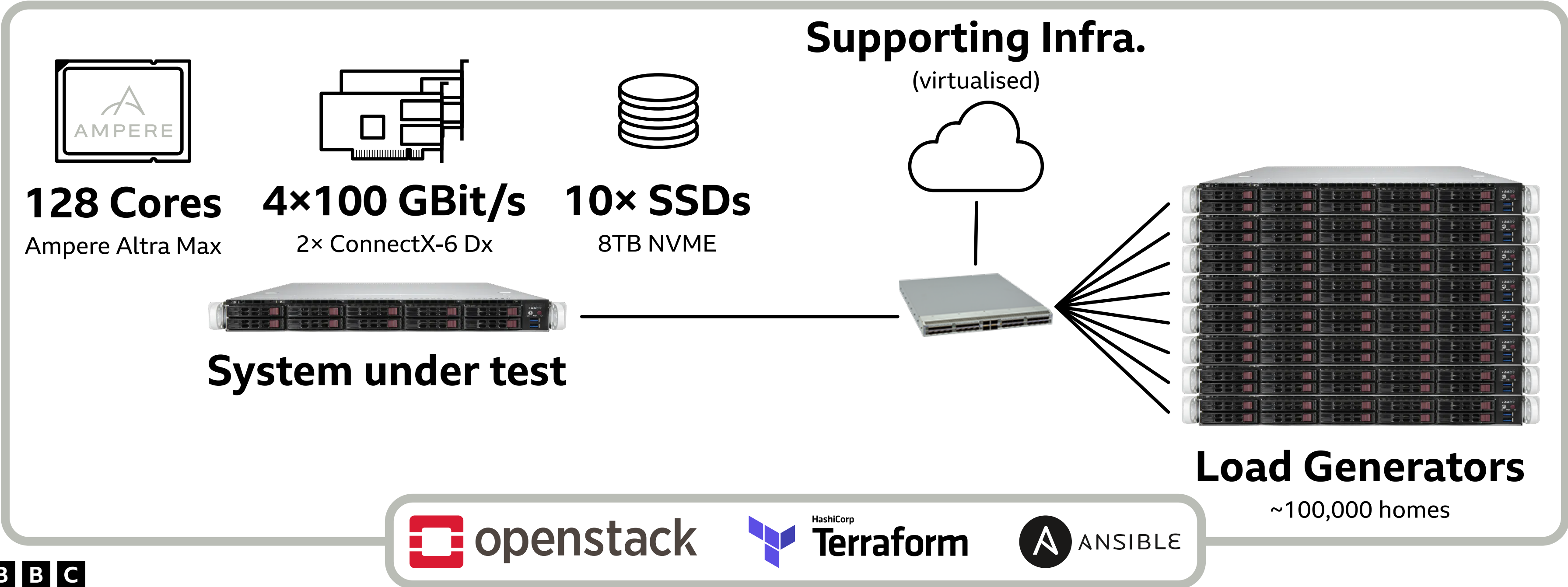
**System under test**

## Supporting Infra. (virtualised)



**Load Generators**  
~100,000 homes

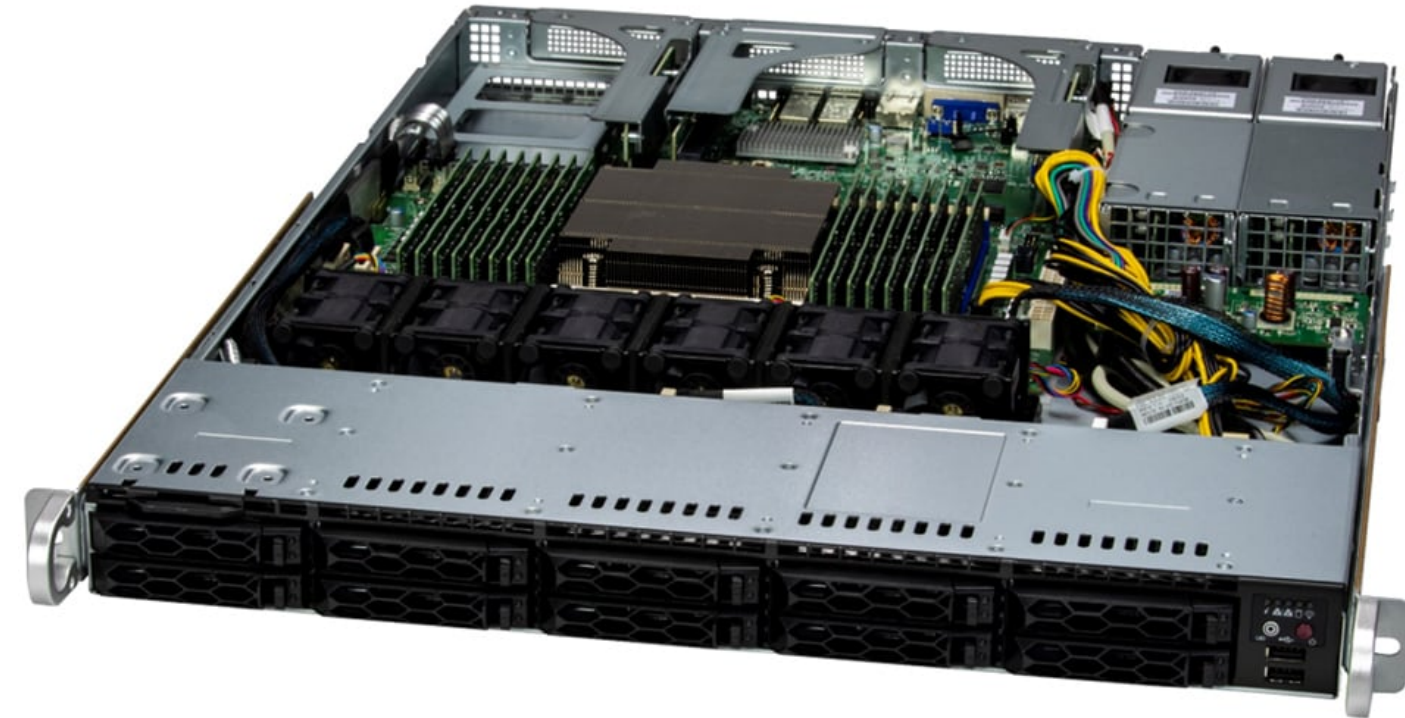
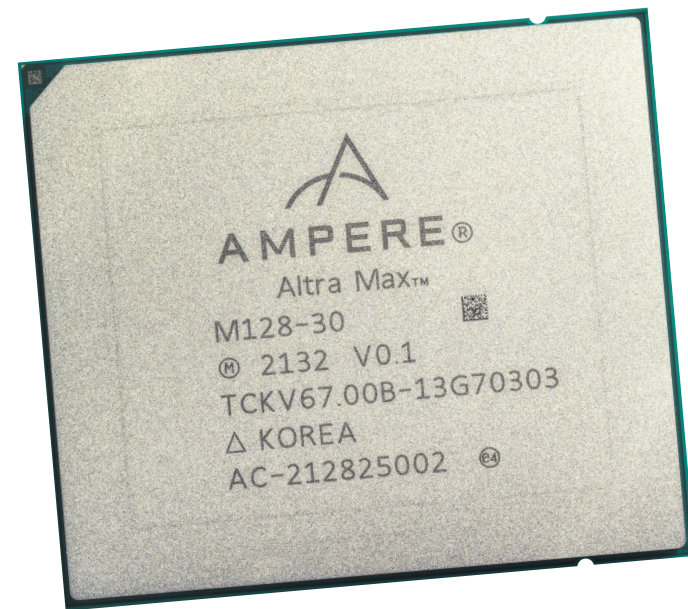
# Test lab





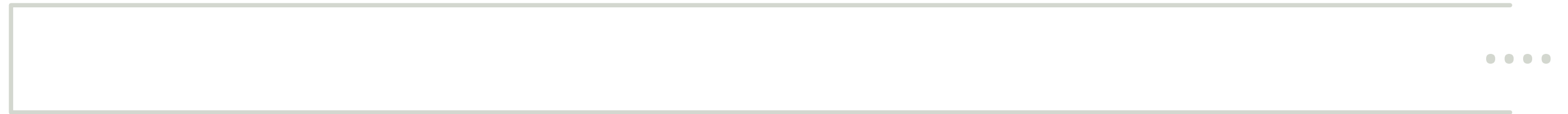
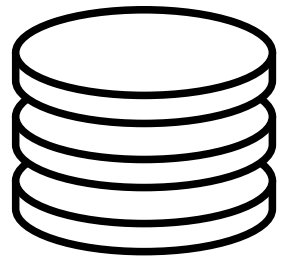
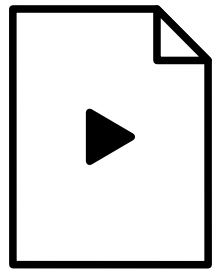
# ARM Servers

---



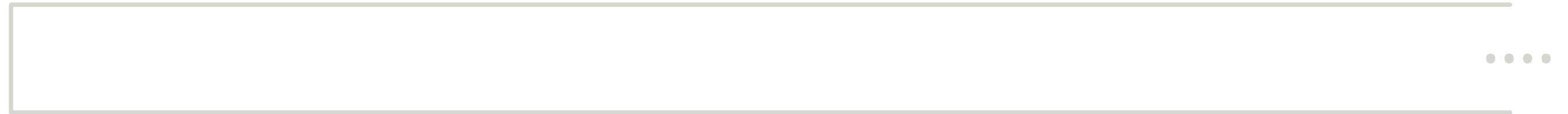
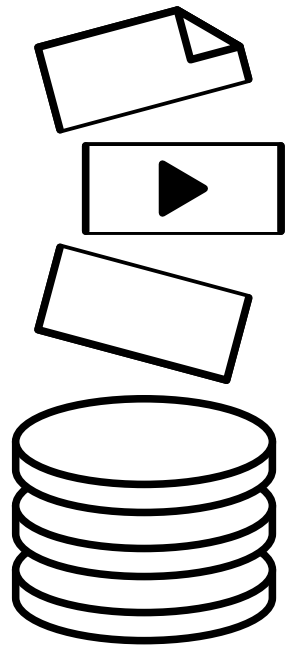
# Operating System Page Cache Primer

---



# Operating System Page Cache Primer

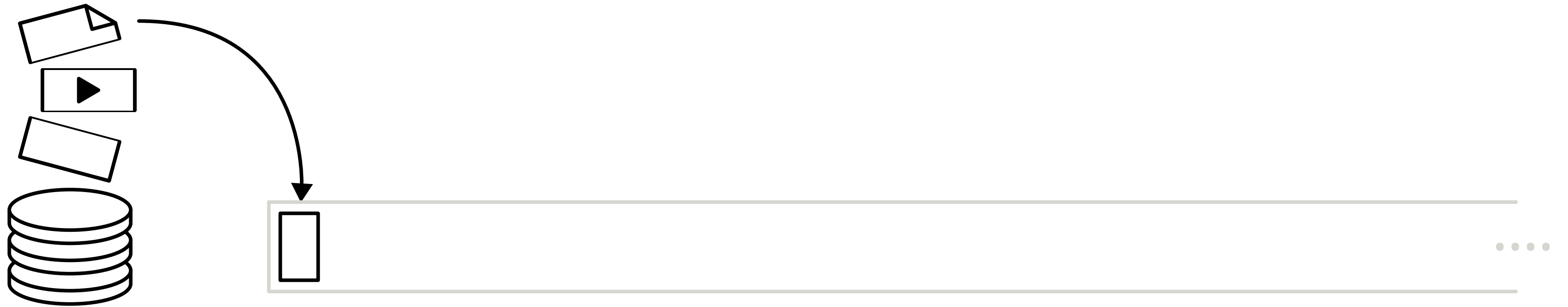
---



NGINX

# Operating System Page Cache Primer

---



NGINX

# Operating System Page Cache Primer



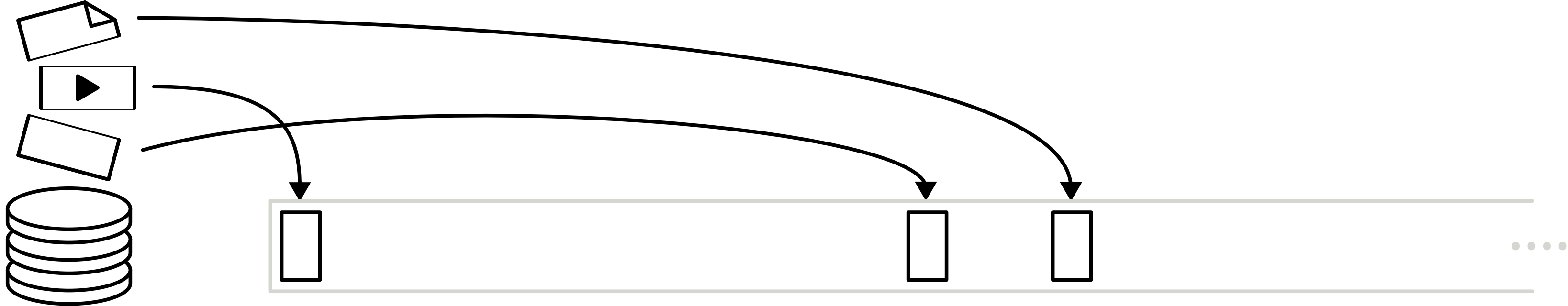
NGINX

# Operating System Page Cache Primer



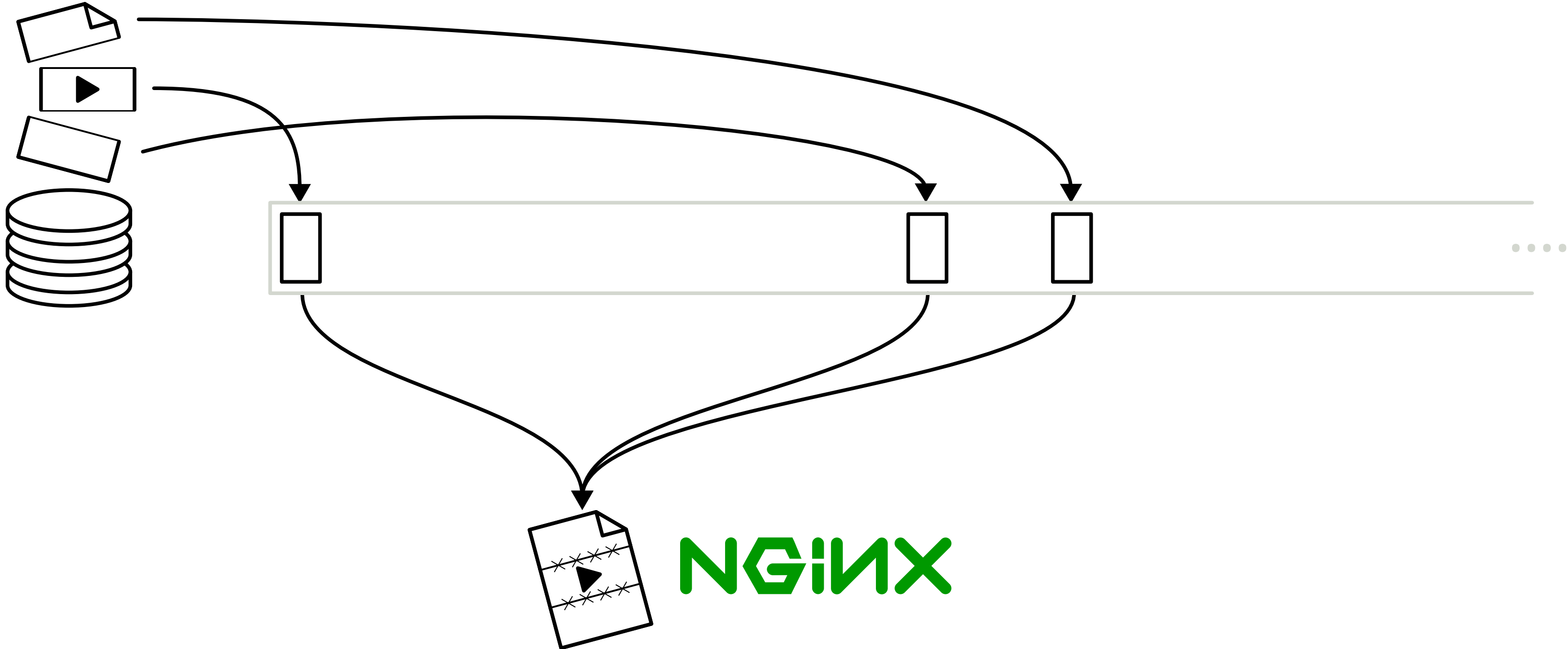
NGINX

# Operating System Page Cache Primer



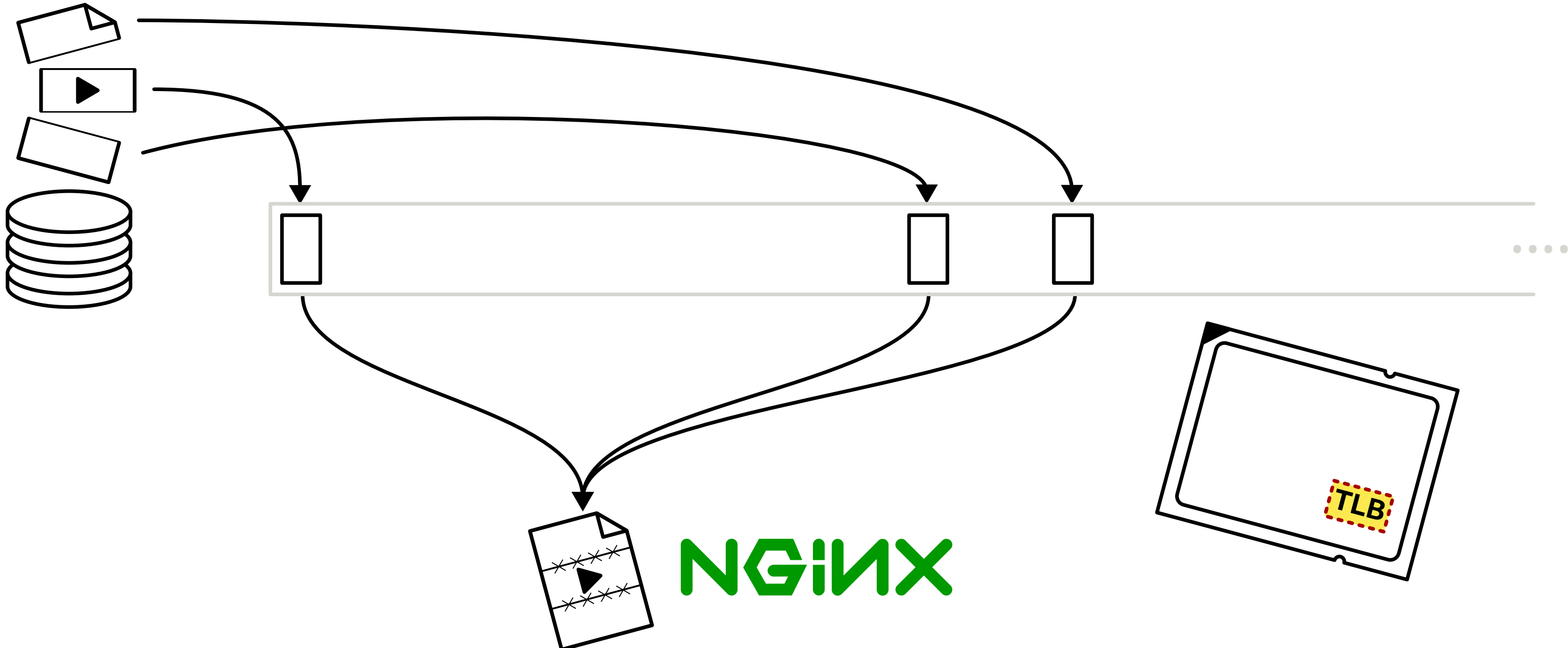
NGINX

# Operating System Page Cache Primer

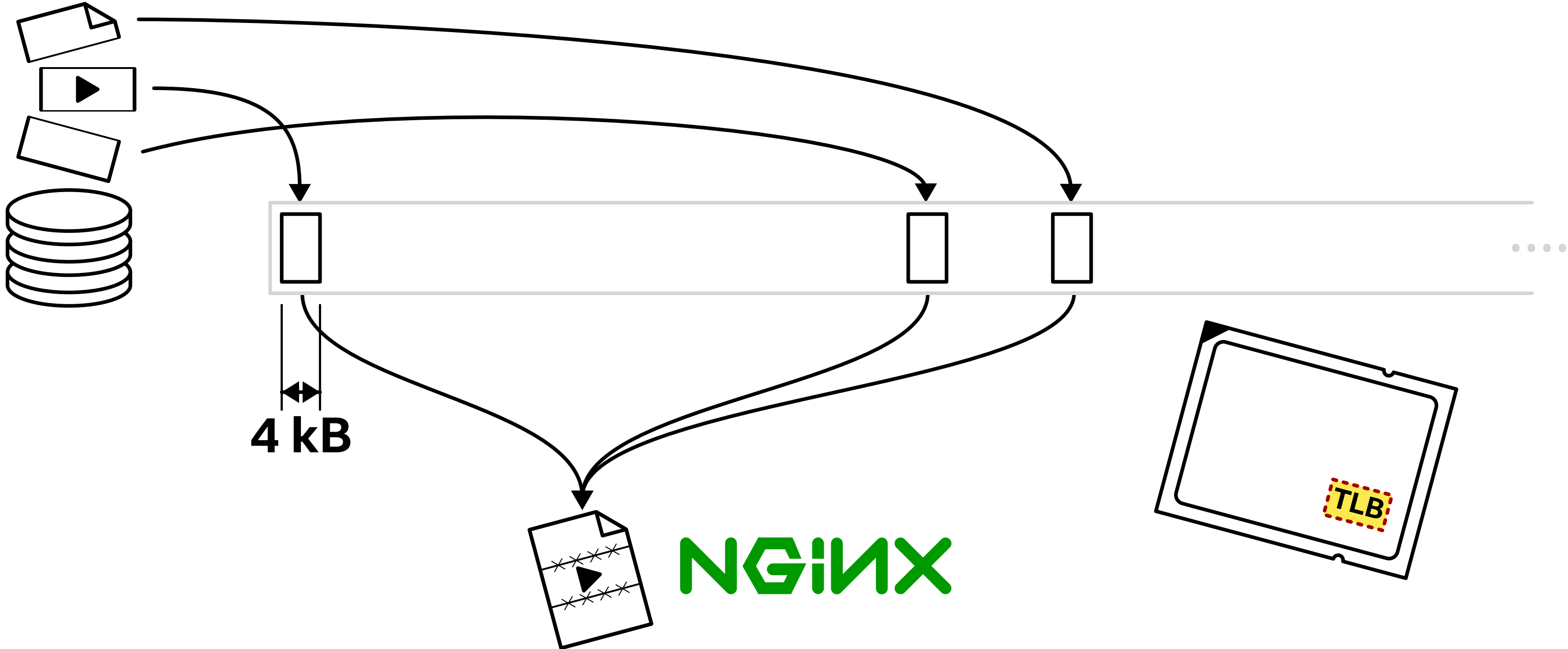




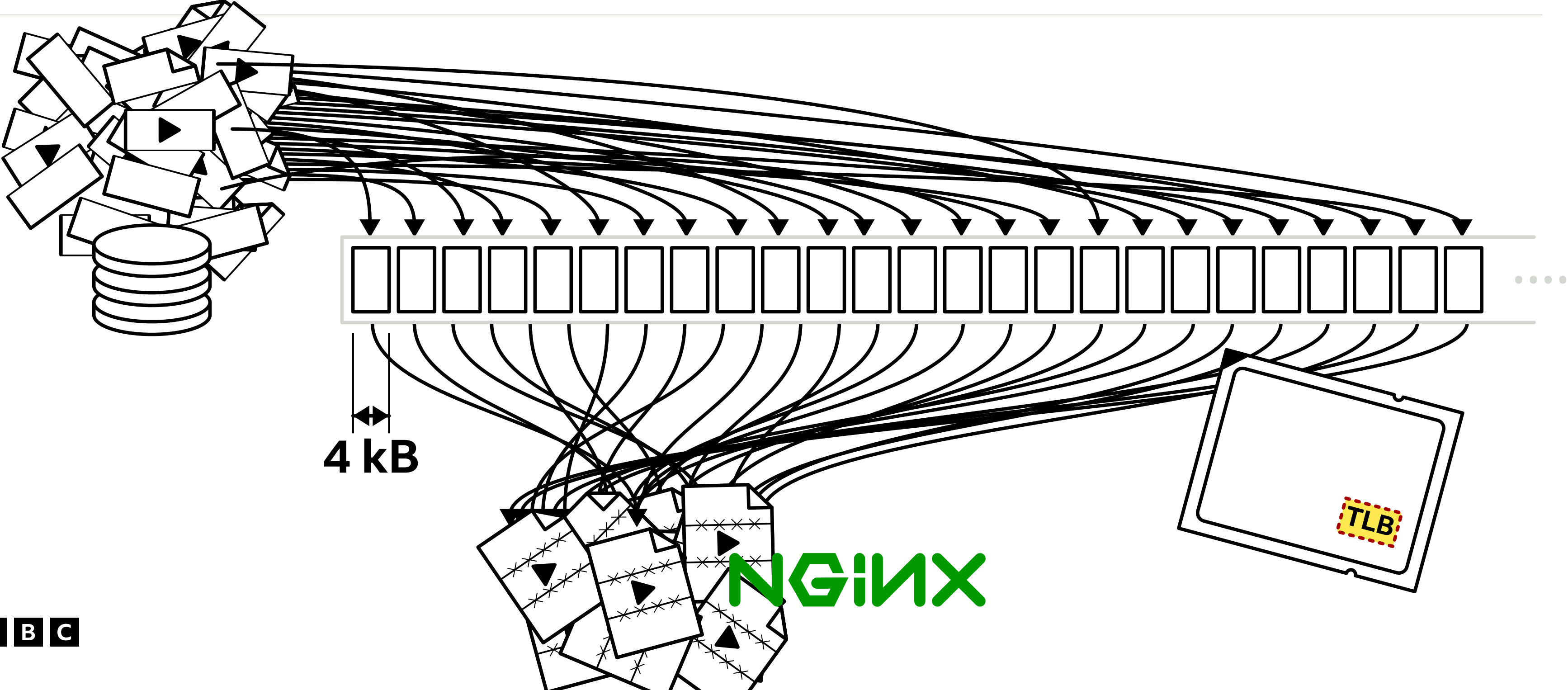
# Operating System Page Cache Primer



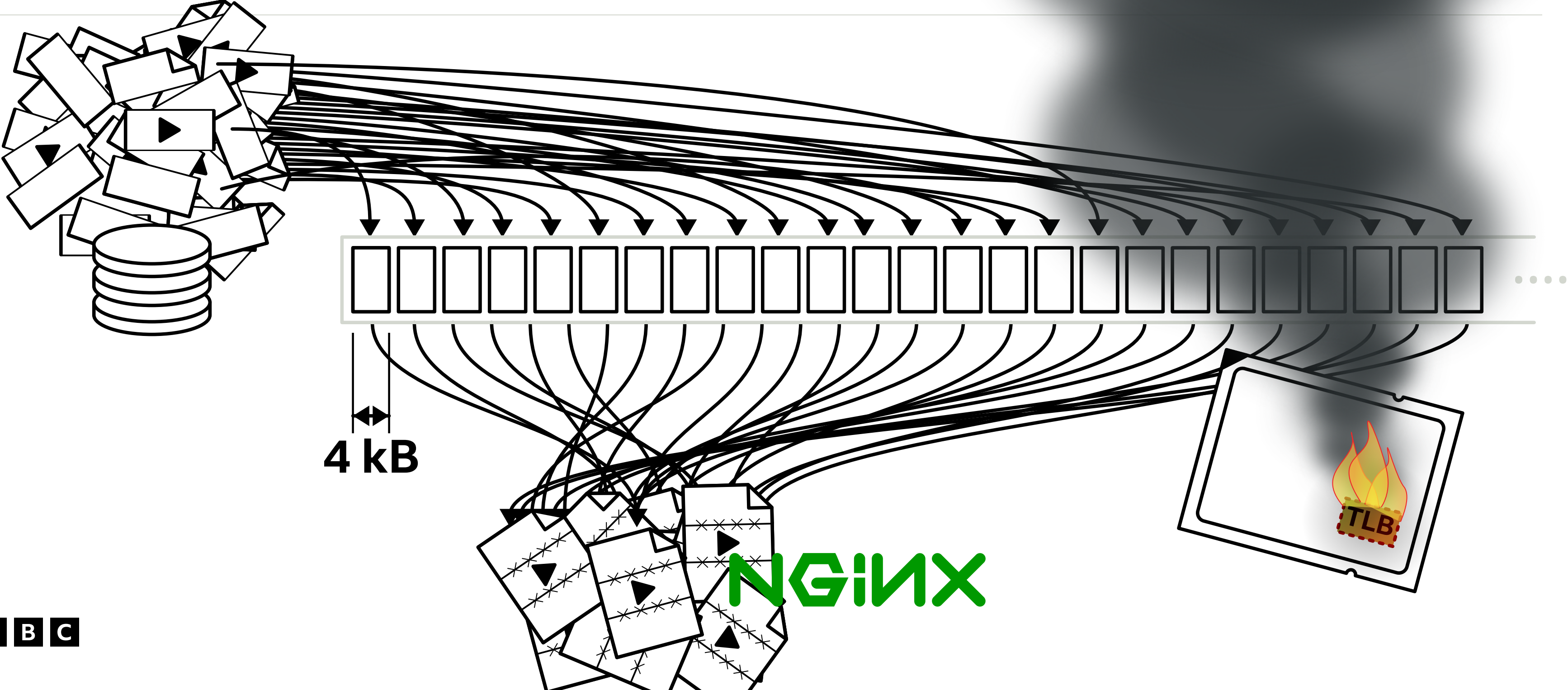
# Operating System Page Cache Primer



# Operating System Page Cache Primer



# Operating System Page Cache Primer



NGINX

# Available Page Sizes

---

4 kB

2 MB

1 GB

**x86\_64 (Intel/AMD)**

# Available Page Sizes

---

4 kB

~~2 MB~~

~~1 GB~~

**x86\_64 (Intel/AMD)**

# Available Page Sizes

---

4 kB

~~2 MB~~

~~1 GB~~

**x86\_64 (Intel/AMD)**

4 kB

16 kB

64 kB

2 MB

32 MB

512 MB

1 GB

**AARCH64 (ARM)**

# Available Page Sizes

---

4 kB

~~2 MB~~

~~1 GB~~

**x86\_64 (Intel/AMD)**

4 kB

16 kB

64 kB

~~2 MB~~

~~32 MB~~

~~512 MB~~

~~1 GB~~

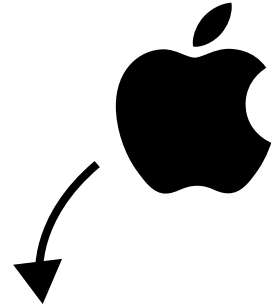
**AARCH64 (ARM)**



# Available Page Sizes

4 kB  
~~2 MB~~  
~~1 GB~~

**x86\_64 (Intel/AMD)**

  
4 kB   16 kB   64 kB  
~~2 MB~~   ~~32 MB~~   ~~512 MB~~  
~~1 GB~~

**AARCH64 (ARM)**

# Available Page Sizes

---

4 kB

~~2 MB~~

~~1 GB~~

**x86\_64 (Intel/AMD)**

4 kB

16 kB

64 kB

~~2 MB~~

~~32 MB~~

~~512 MB~~

~~1 GB~~

**AARCH64 (ARM)**

4kB Pages

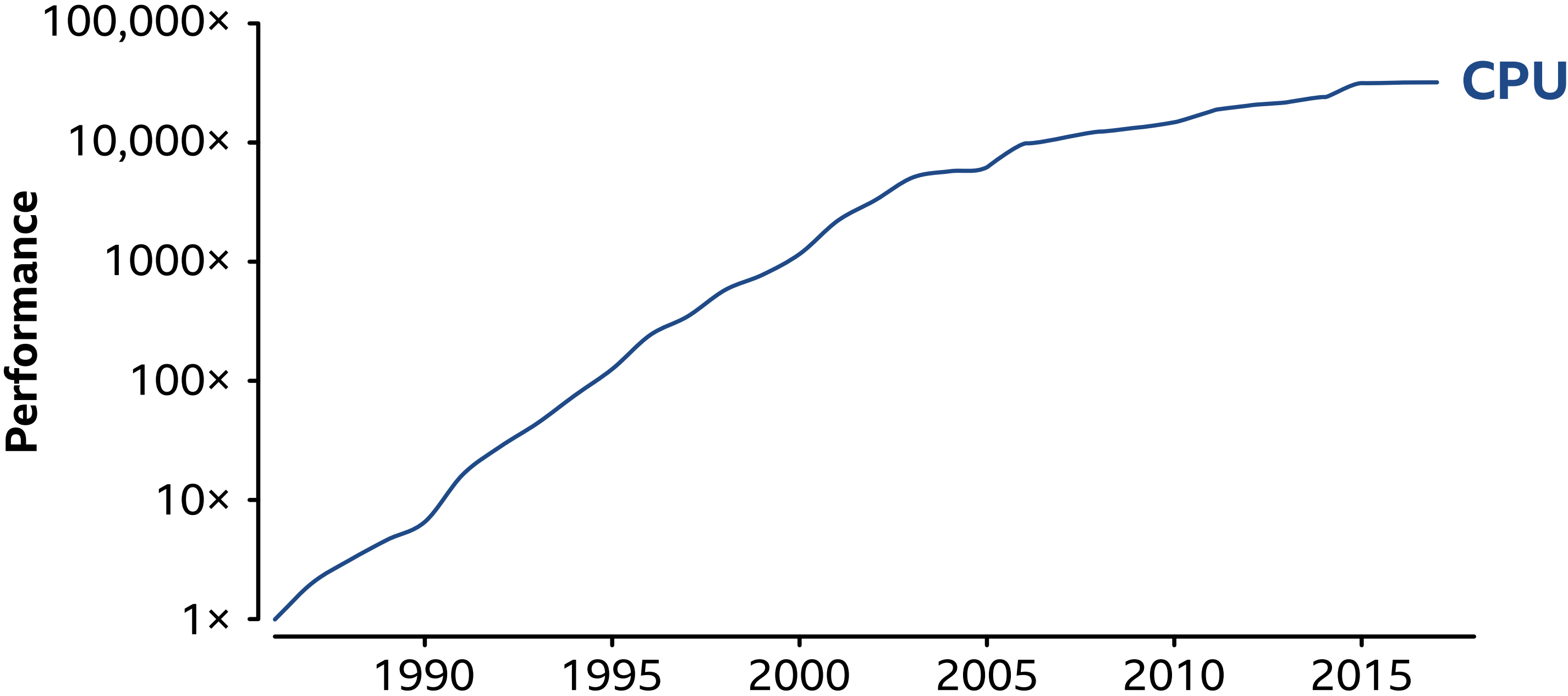
64kB Pages

**80 GBit/s**

**150 GBit/s**

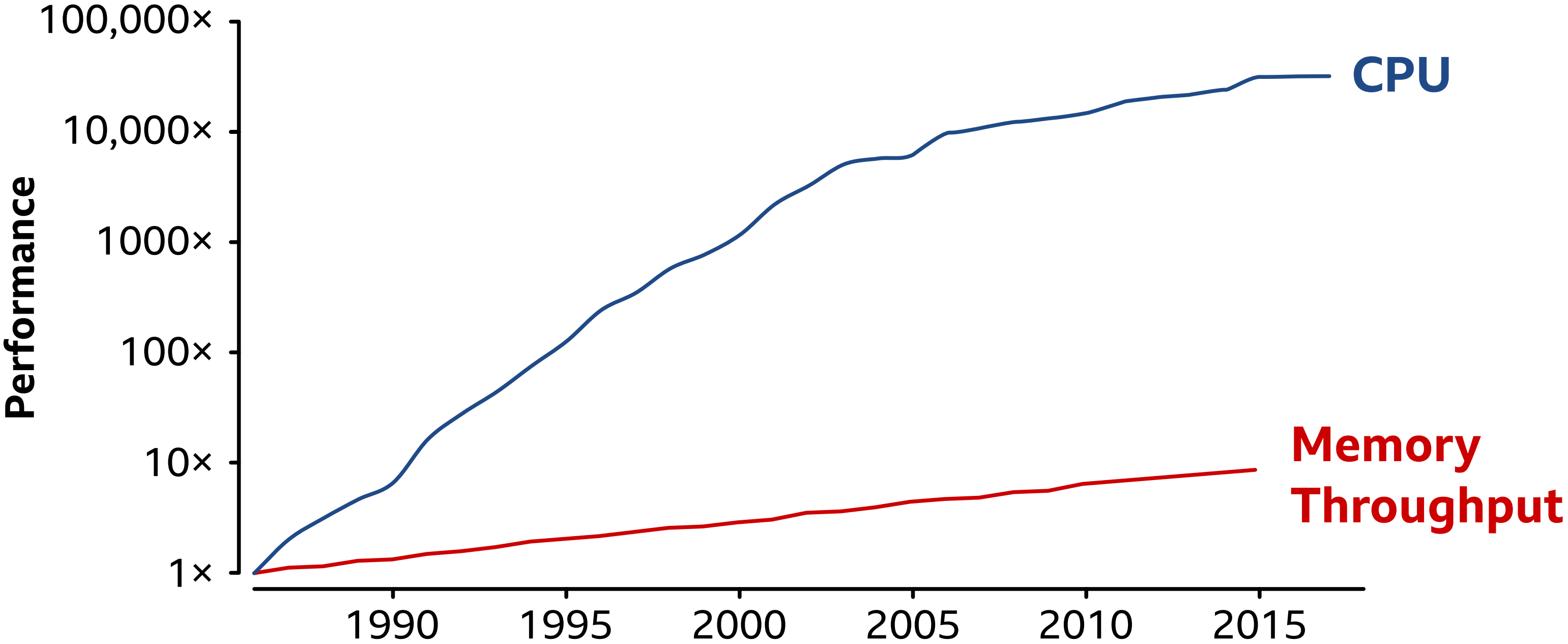
# Memory Bandwidth

# Bottleneck: Not the CPU anymore



Data from "Computer Architecture, A Quantitative Approach (Sixth Edition)"  
Hennessy and Patterson 2019, figures 1.1 (CPU) and 2.2 (Memory Throughput).

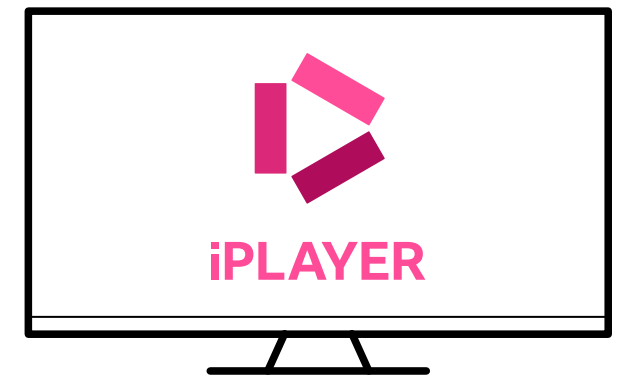
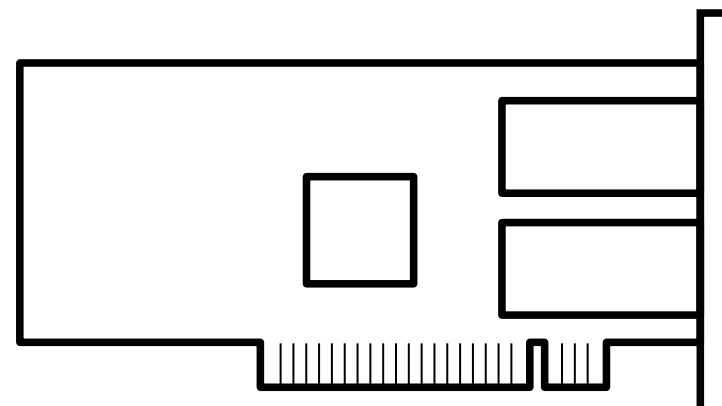
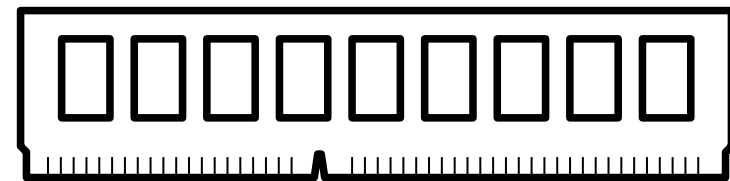
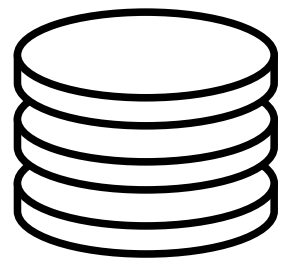
# Bottleneck: Not the CPU anymore



Data from "Computer Architecture, A Quantitative Approach (Sixth Edition)"  
Hennessy and Patterson 2019, figures 1.1 (CPU) and 2.2 (Memory Throughput).

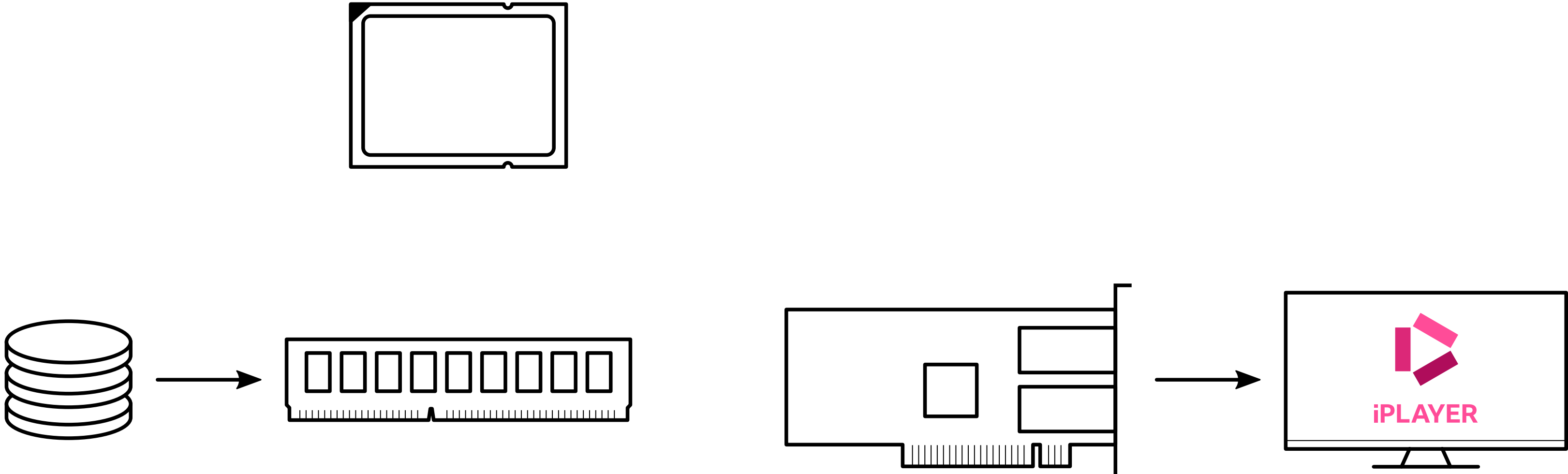
# Where is the memory bandwidth going?

---



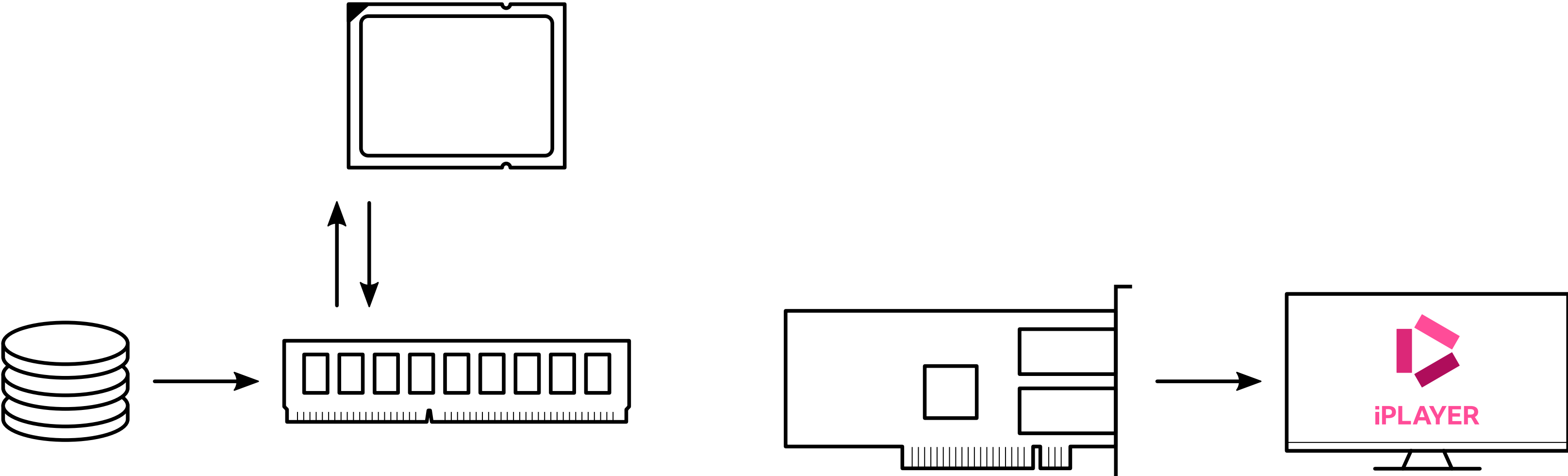
# Where is the memory bandwidth going?

---

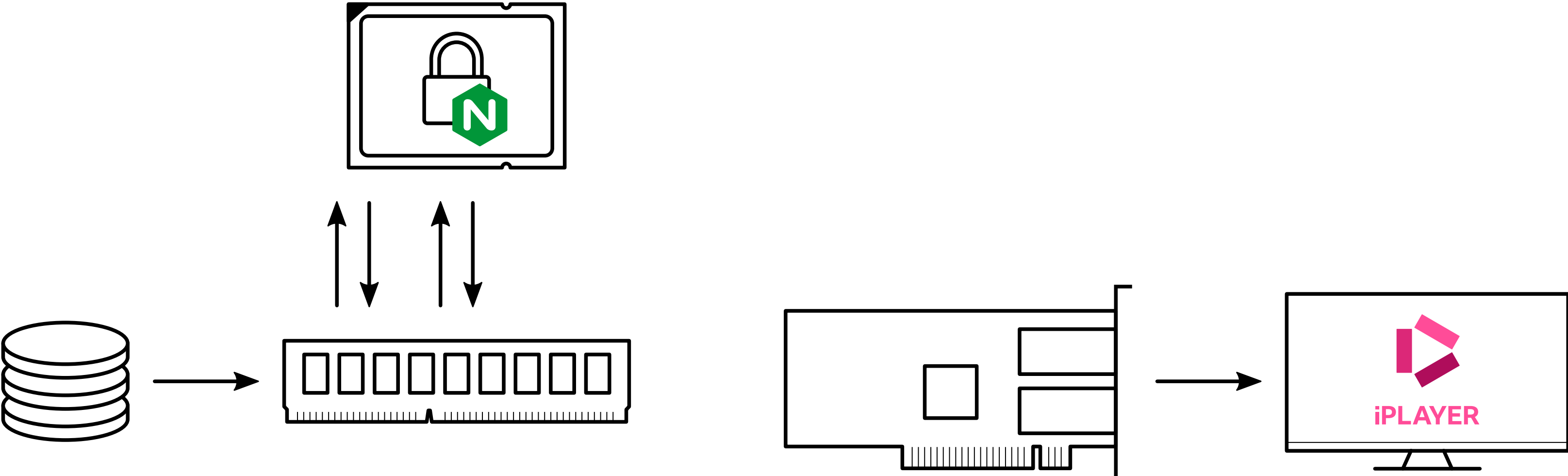




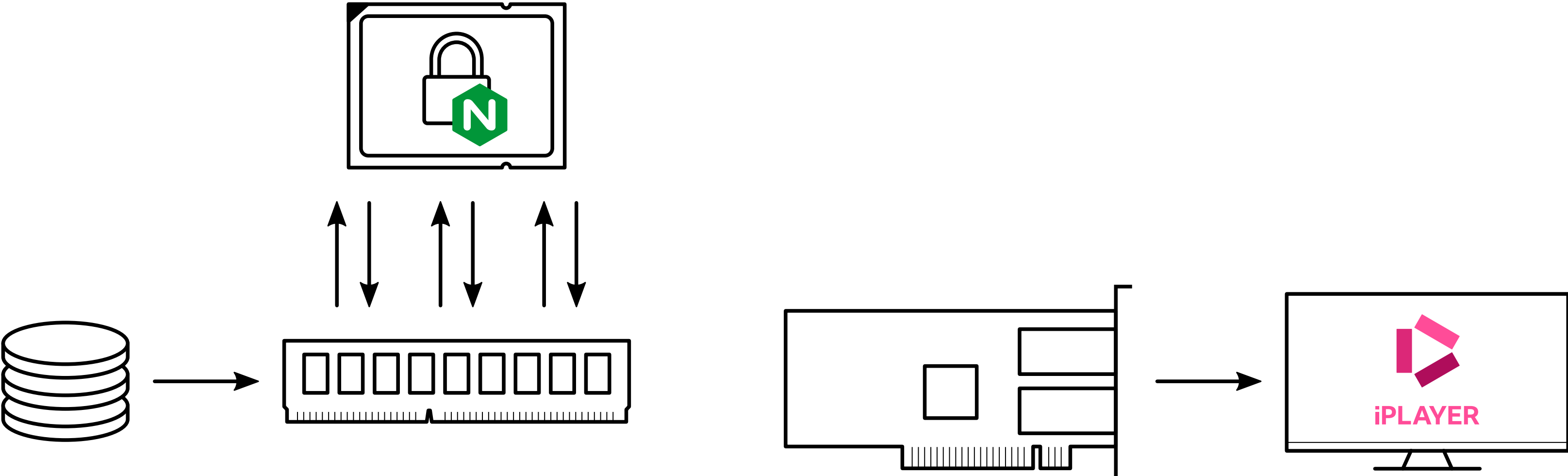
# Where is the memory bandwidth going?



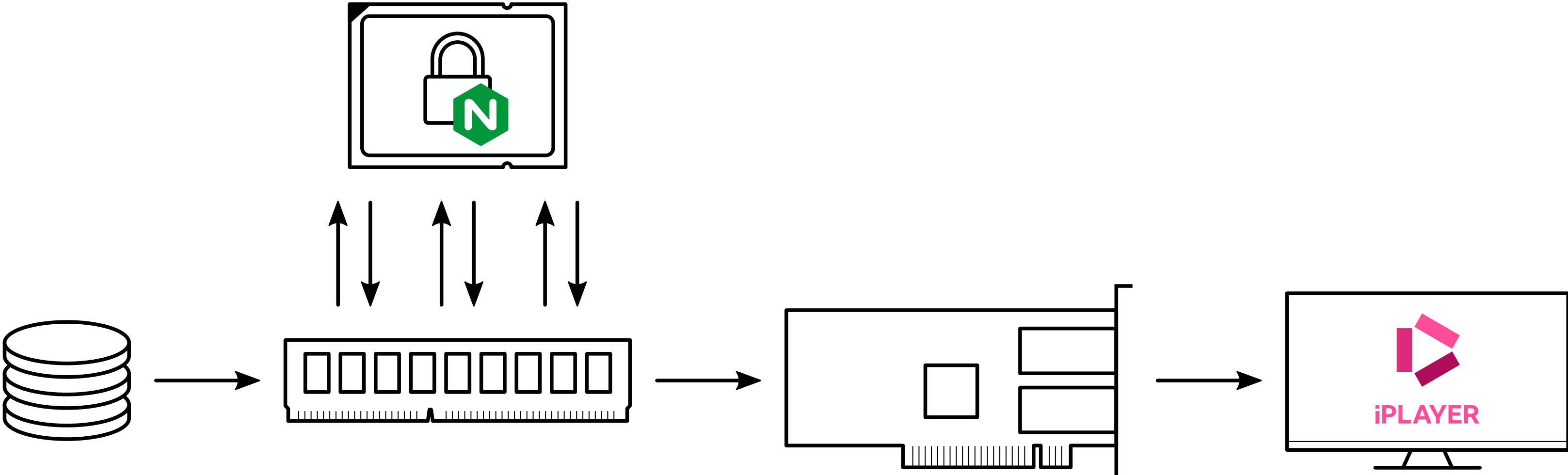
# Where is the memory bandwidth going?



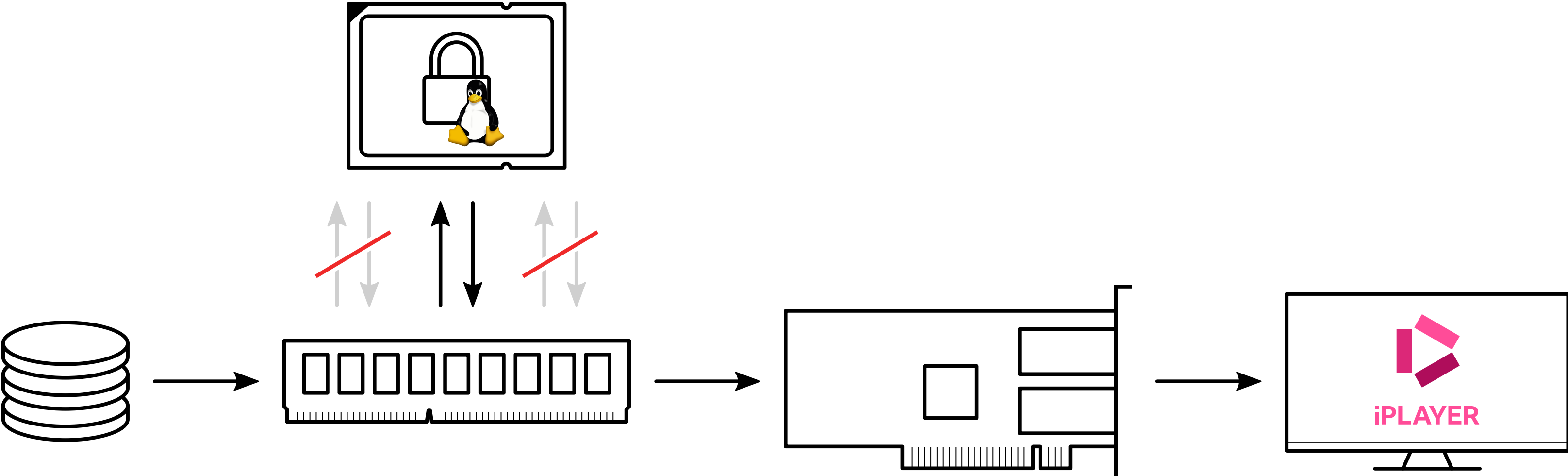
# Where is the memory bandwidth going?



# Where is the memory bandwidth going?



# Kernel TLS (kTLS)



Userspace TLS

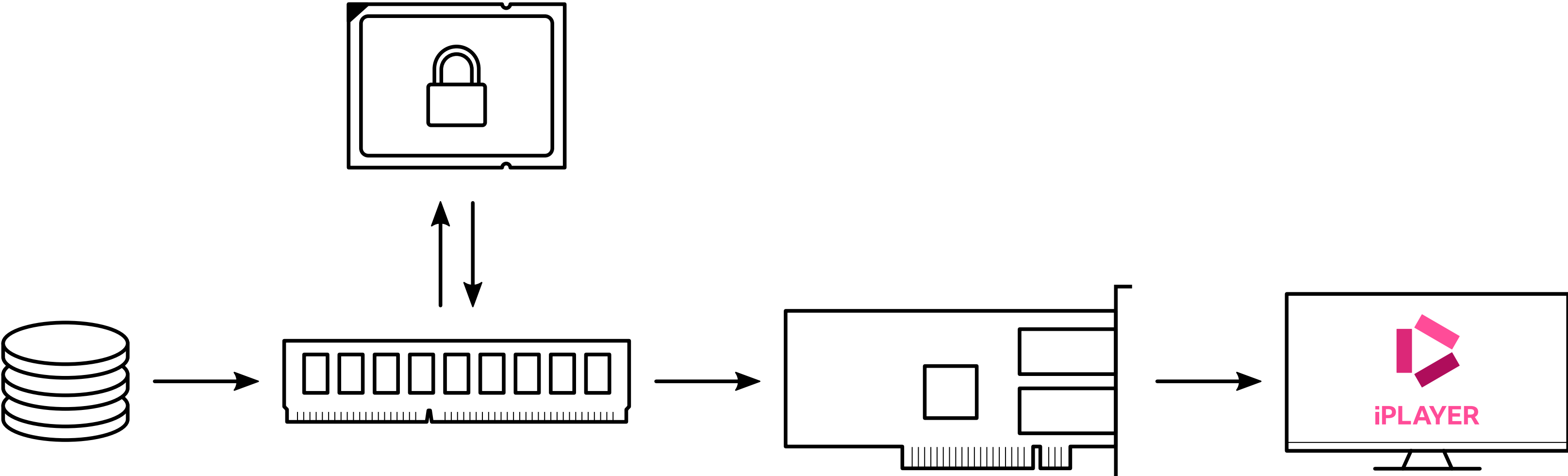
Kernel TLS

150 GBit/s

260 GBit/s

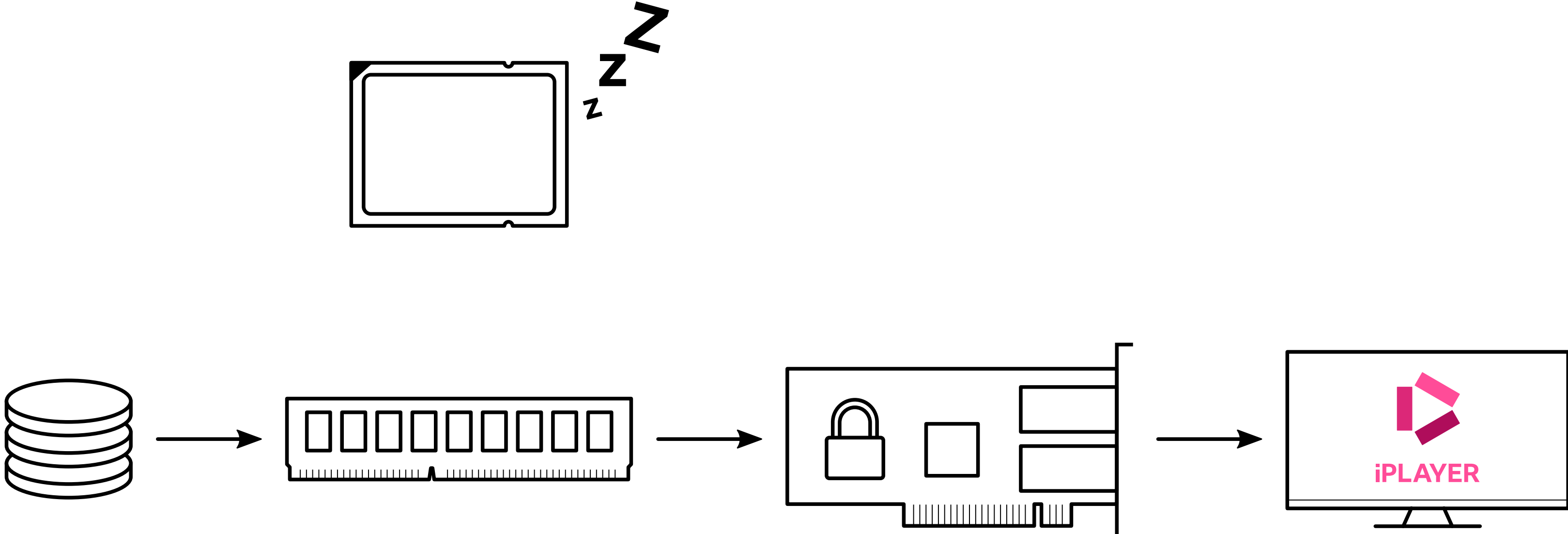
# TLS Offload

# TLS Offload: The theory

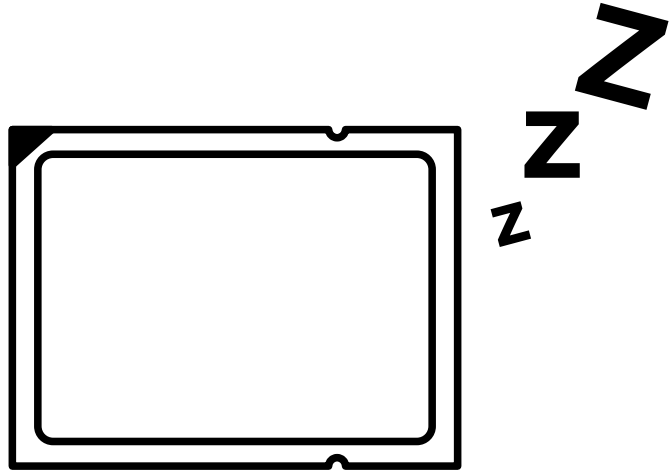




# TLS Offload: The theory



# TLS Offload: The theory



**NVIDIA**  
ConnectX-6 Dx 100G/200G Ethernet NIC

Accelerated networking for modern cloud data centers.

**Product Specifications**

Total bandwidth	200Gb/s
Supported Ethernet speeds	10/25/40/50/100/200GbE
Number of network ports	1 or 2
Network interface technologies	NRZ/PAM4
Host interface	PCIe Gen4 x16, with NVIDIA Multi-Host technology
Platform security	Hardware root of trust and secure firmware update
Form factors	PCIe HHHL, OCP3.0 SFF
Network interfaces	SFP56, QSFP56

**Advanced Networking and Security**

ConnectX-6 Dx features virtual switch (vSwitch) and virtual router (vRouter) hardware accelerations that deliver significantly higher performance than non-accelerated solutions. ConnectX-6 Dx supports a choice of single-root input/output (IO) virtualization (SR-IOV) and VirtIO in hardware, so customers can best address their application needs. By offloading cloud networking workloads, ConnectX-6 Dx frees up CPU cores for business applications while reducing total cost of ownership.

In the face of a growing cyber threat landscape, ConnectX-6 Dx provides built-in inline encryption and decryption, stateful packet filtering, and other capabilities, bringing advanced security down to every node with unprecedented performance and scalability.

Built on the solid foundation of NVIDIA's ConnectX line of NICs, ConnectX-6 Dx offers best-in-class remote direct-memory access (RDMA) over converged Ethernet (RoCE) capabilities, enabling scalable, resilient, and easy-to-deploy RoCE solutions. For data storage, ConnectX-6 Dx optimizes a suite of storage accelerations, bringing Non-Volatile Memory Express over Fabrics (NVMe-oF) target and initiator offloads.

**Features\***

- Network Interface**
  - > Dual ports of 10/25/40/50/100GbE or a single port of 200GbE
  - > Up to 200Gb/s total bandwidth
- Host Interface**
  - > PCIe Gen 4.0 compatible, 16 lanes
  - > NVIDIA Multi-Host supports connection of up to four hosts

**Chelsio Communications**  
Accelerate

100G NGINX Offload & Encryption  
Chelsio T6 vs. Mellanox ConnectX-6 Dx

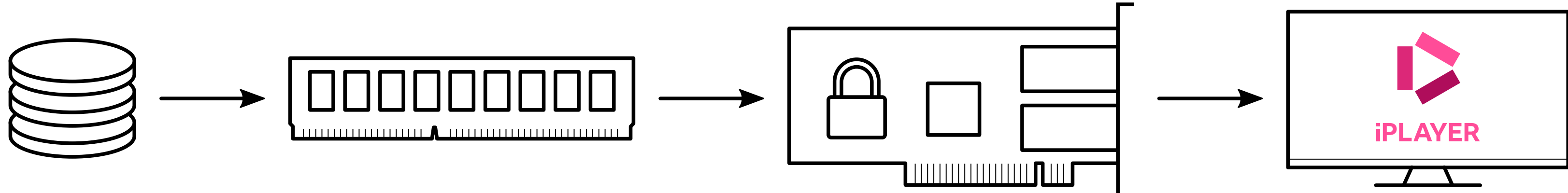
**Executive Summary**

Chelsio is the leading provider of network protocol offloading technologies, and its TCP Offload Engine (TOE) is the first and currently only engine capable of full TCP/IP offload up to 100Gbps. Additionally, the Inline TLS/SSL offload capability offered by Chelsio T6 adapters is uniquely capable of producing record breaking TLS/SSL performance. Chelsio Inline TLS/SSL offload solution supports TCP/IP and TLS/SSL processing in cut-through fashion to achieve optimal bandwidth and latency. The Chelsio adapters are designed specifically to perform computationally intensive network and cryptographic operations more efficiently than general purpose CPUs. Servers with system load comprising of network and cryptographic operations see great savings in CPU Utilization by offloading these operations to the Chelsio T6 adapter.

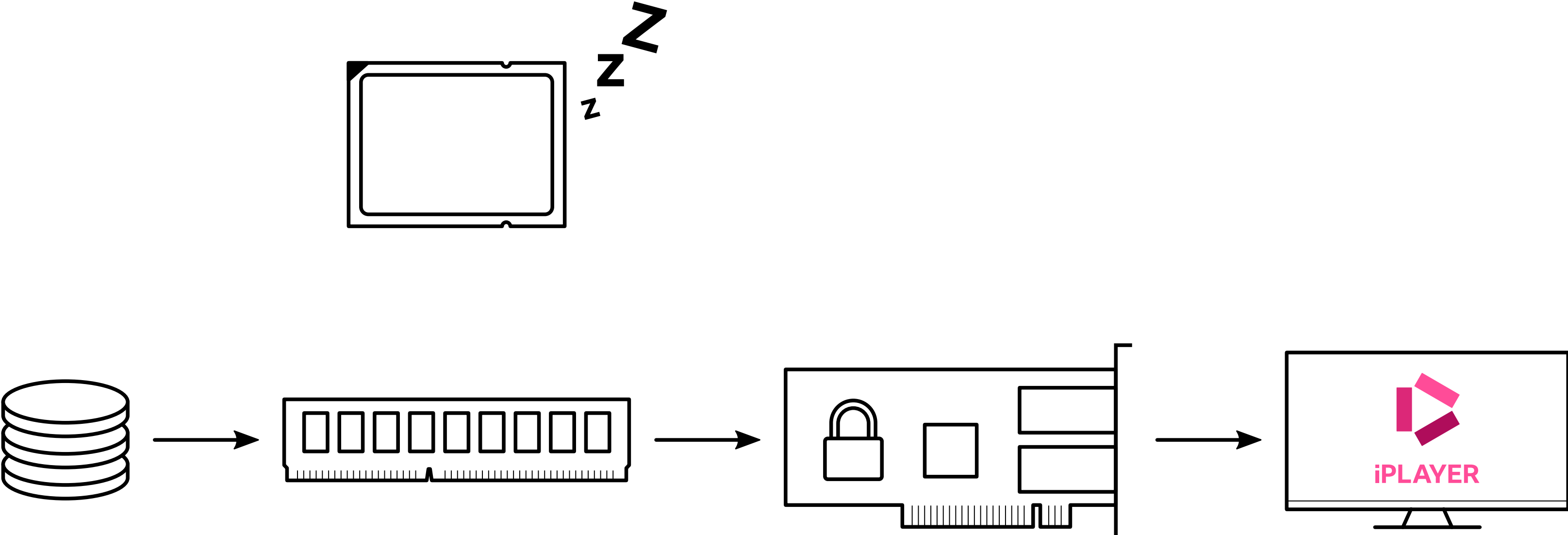
per presents a performance comparison of 100G Chelsio T6 Inline TLS/SSL offload vs. ConnectX-6 Dx kTLS offload with a real-world application, NGINX Plus web server running. It establishes the fact that using Chelsio T6 to offload both the network and cryptographic operations of the NGINX web server consumes significantly less CPU with consistent performance as to Chelsio T6 enabling radically reduced Capex for NGINX server deployments.

per presents the %CPU and throughput results of NGINX Plus web server, using TLS offload and Mellanox ConnectX-6 Dx kTLS offload. The numbers are with connections ranging from 1000 to 10000.

NVIDIA ConnectX-6 Dx Ethernet NIC | Datasheet | 1

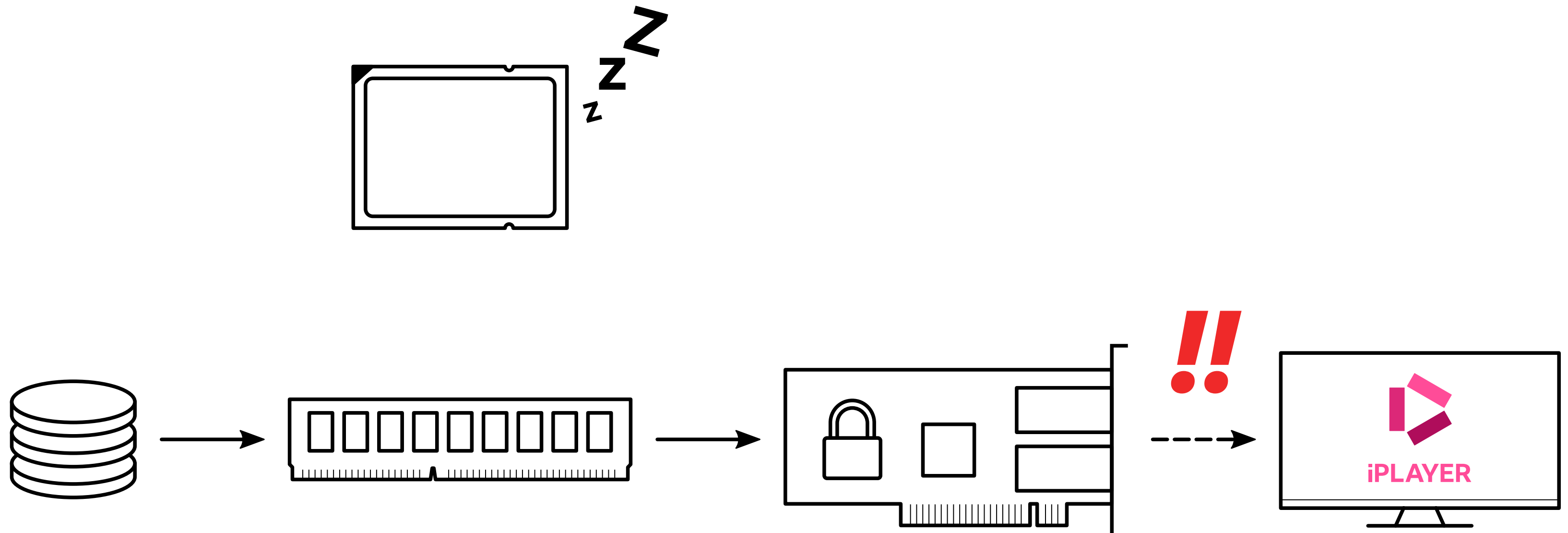


# TLS Offload: The reality



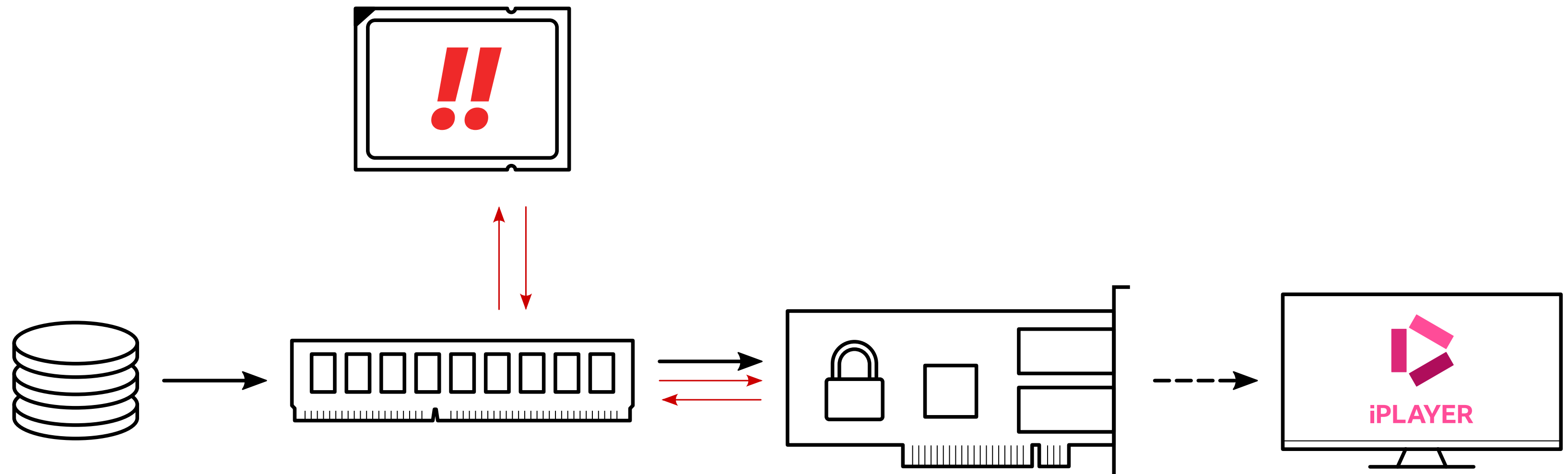
# TLS Offload: The reality

## Problem 1: Retransmissions



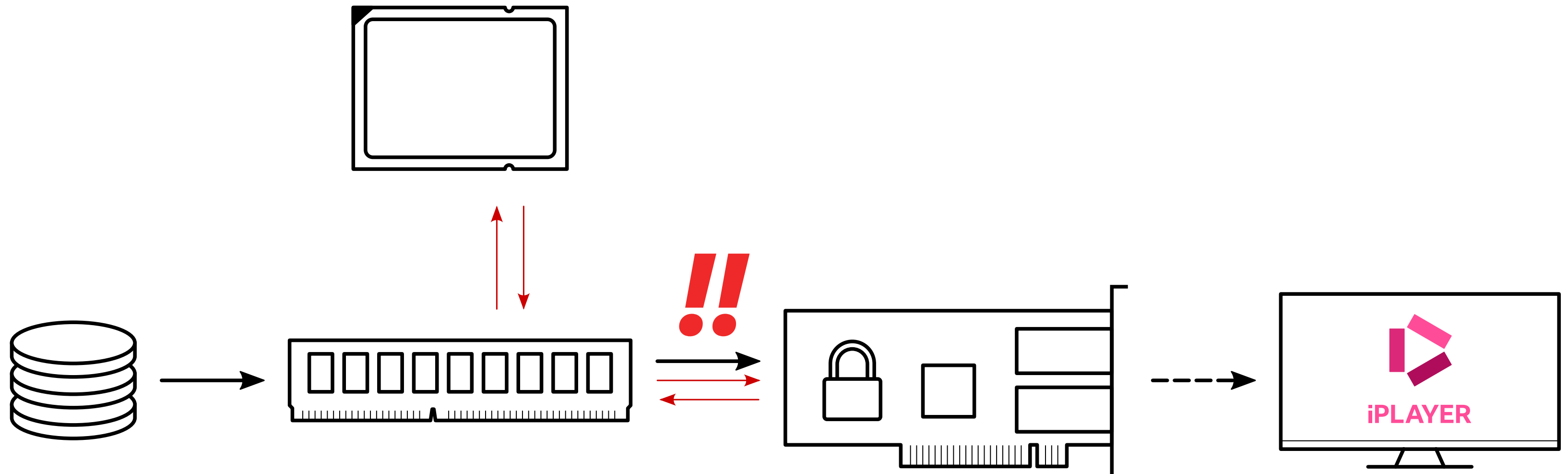
# TLS Offload: The reality

## Problem 1: Retransmissions



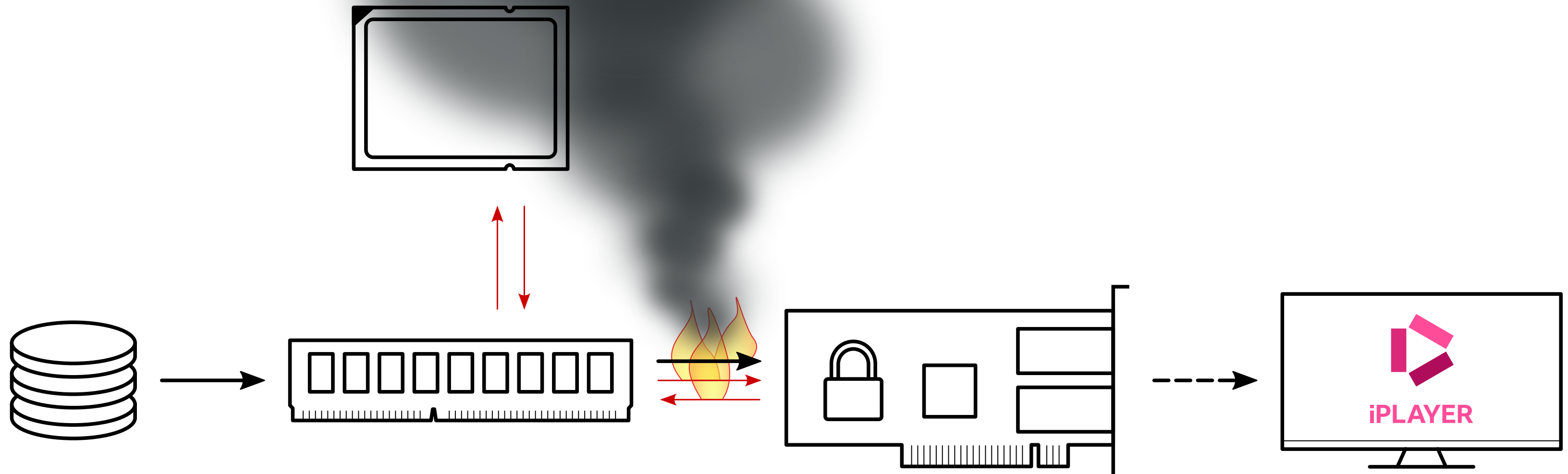
# TLS Offload: The reality

## Problem 1: Retransmissions



## Problem 1: Retransmissions

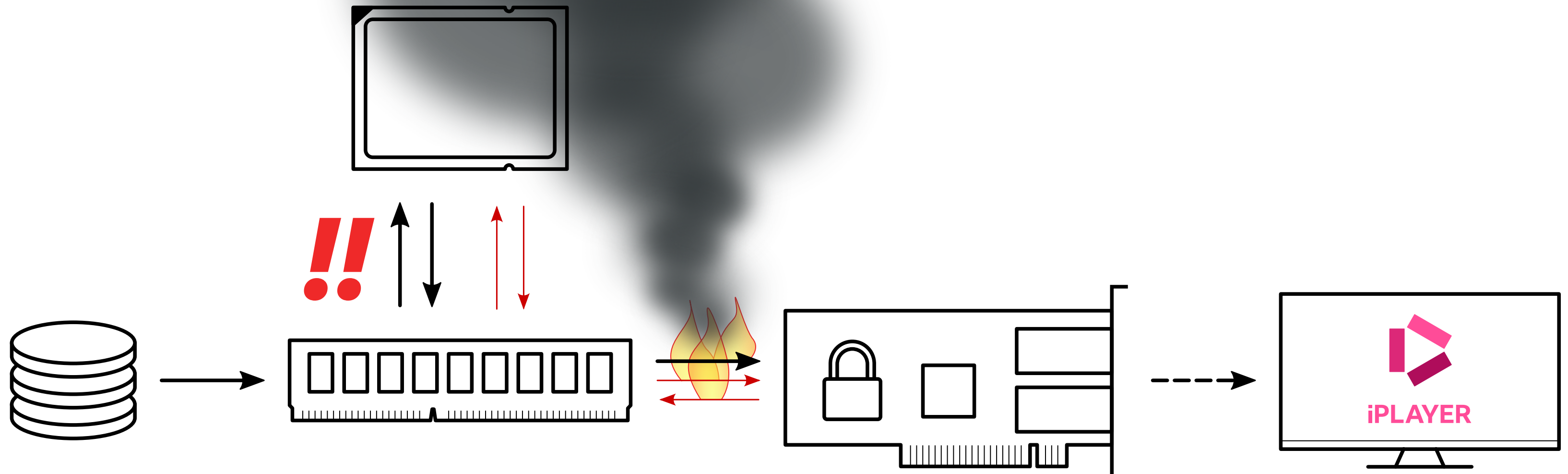
# TLS Offload: The reality



# TLS Offload: The reality

Problem 1: Retransmissions

Problem 2: Linux implementation

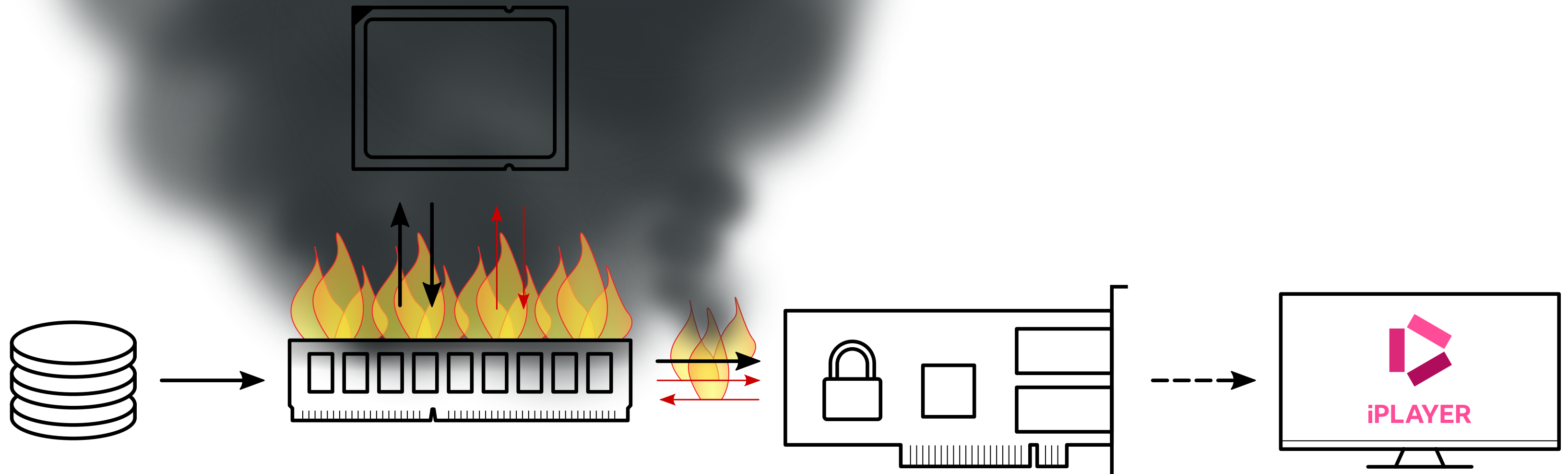




# TLS Offload: The reality

Problem 1: Retransmissions

Problem 2: Linux implementation

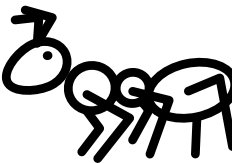
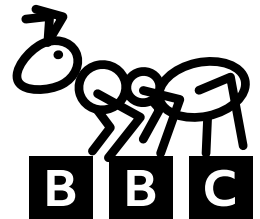
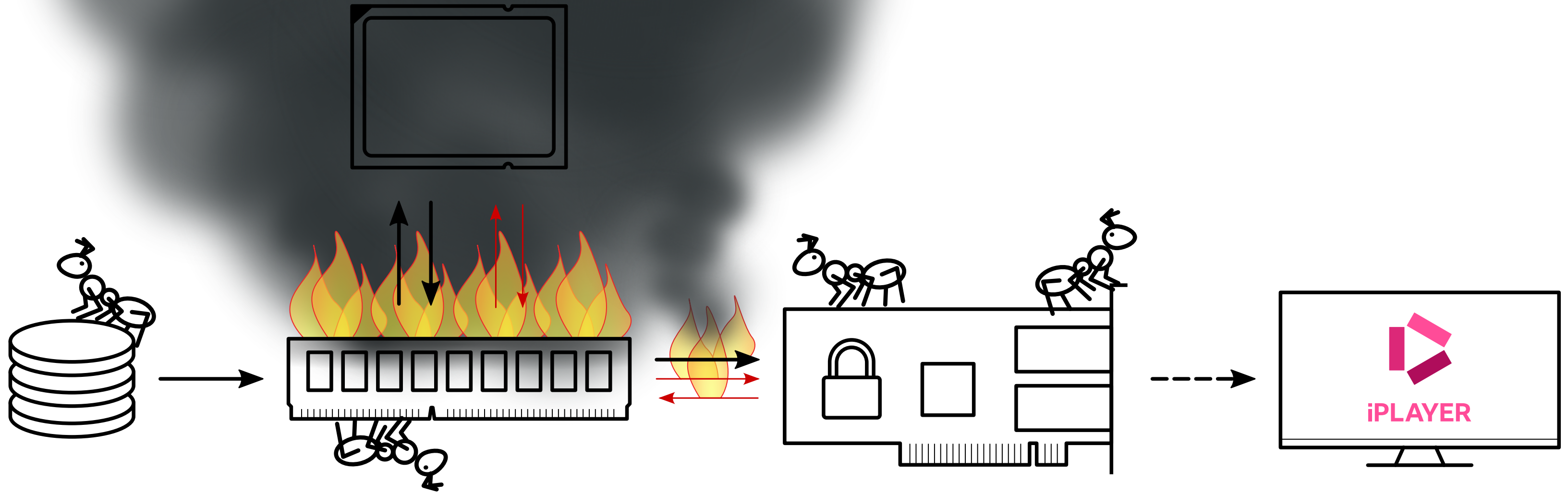
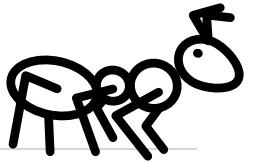


# TLS Offload: The reality

Problem 1: Retransmissions

Problem 2: Linux implementation

Problem 3: Stability





# TLS Offload: The reality

Problem 1: Retransmissions

Problem 2: Linux implementation

Problem 3: Stability



BBC

BBC

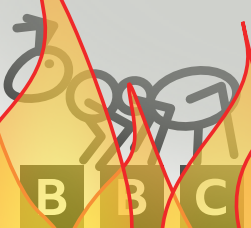
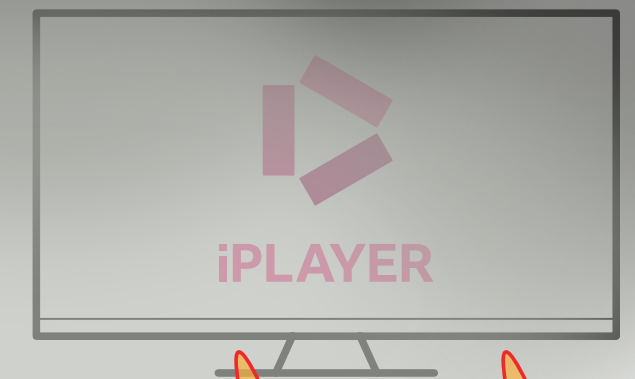
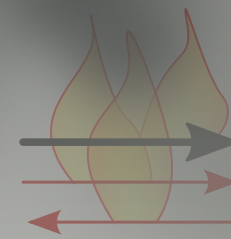
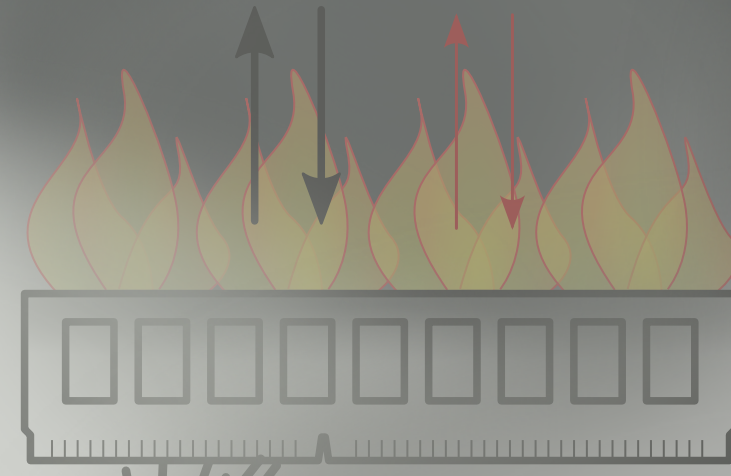


# TLS Offload: The reality

Problem 1: Retransmissions

Problem 2: Linux implementation

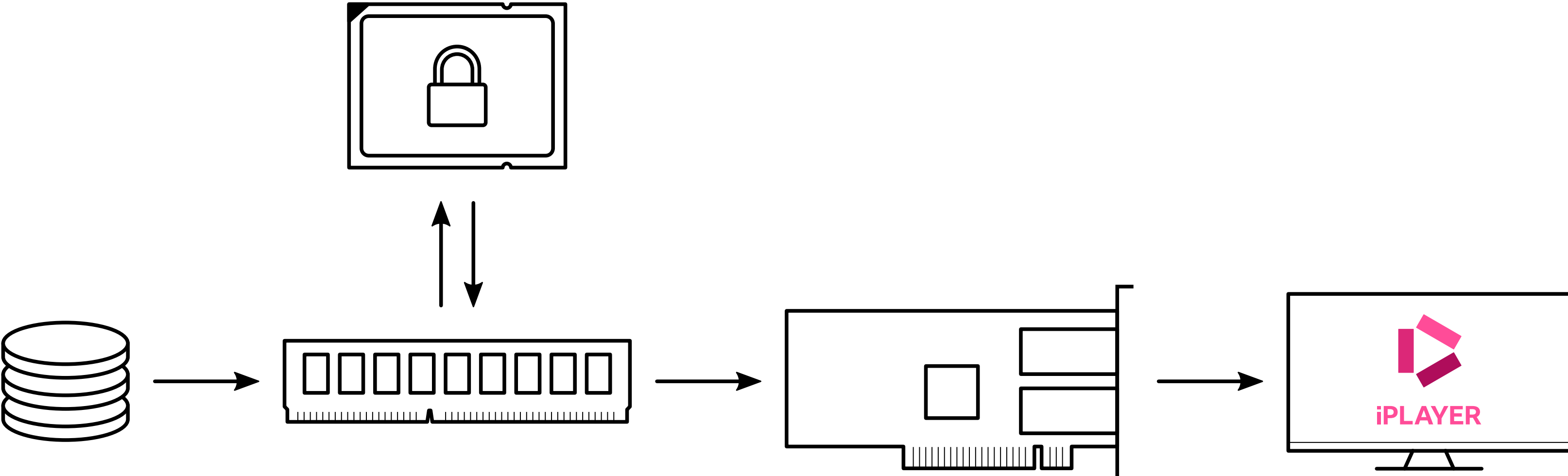
Problem 3: Stability



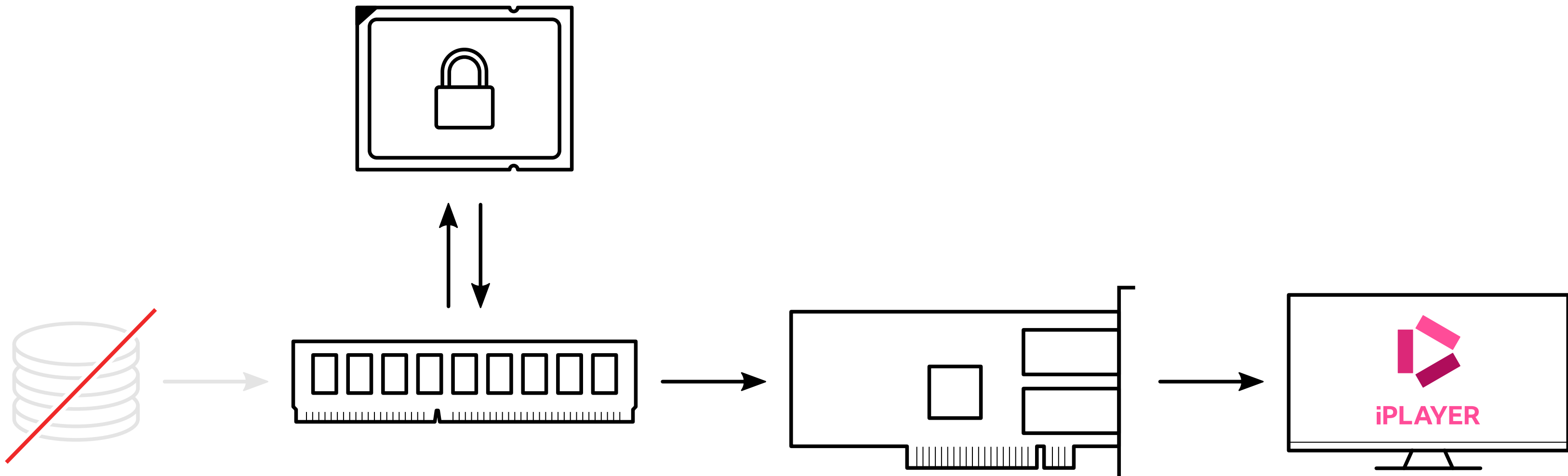


# Network Behaviour

# Request pattern effects



# Request pattern effects



Served From Disk

Served from RAM

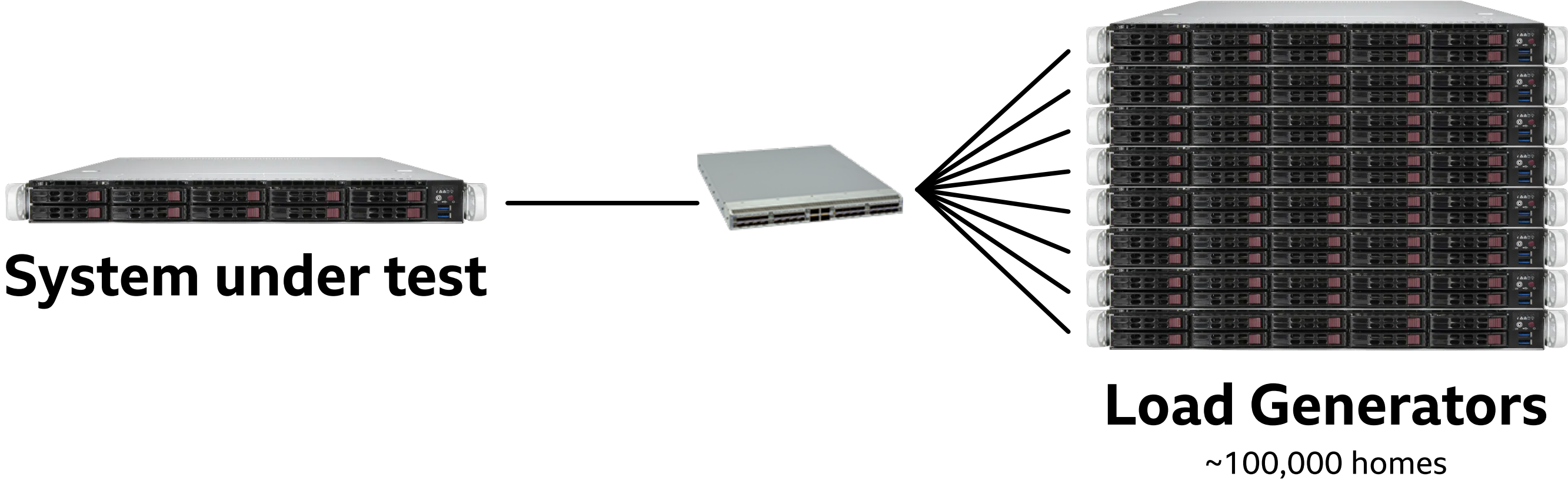
260 GBit/s

340 GBit/s

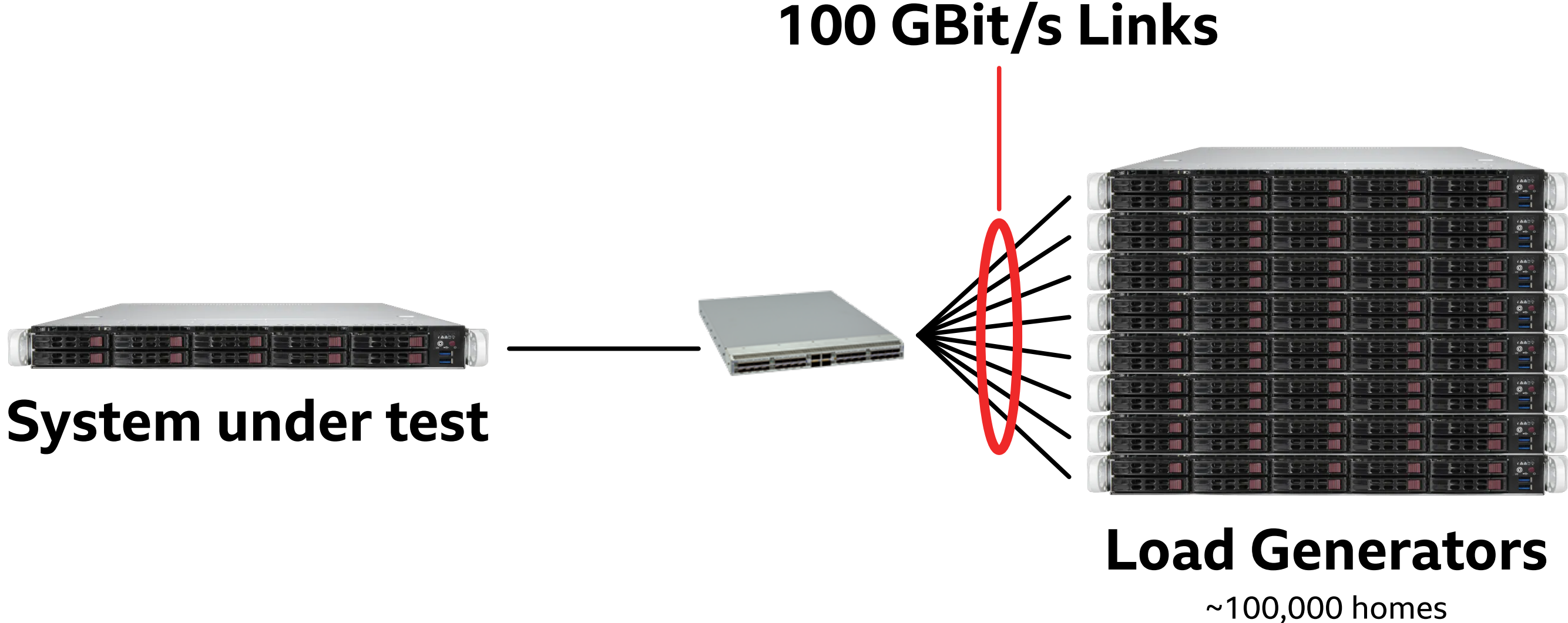


# Test lab realism

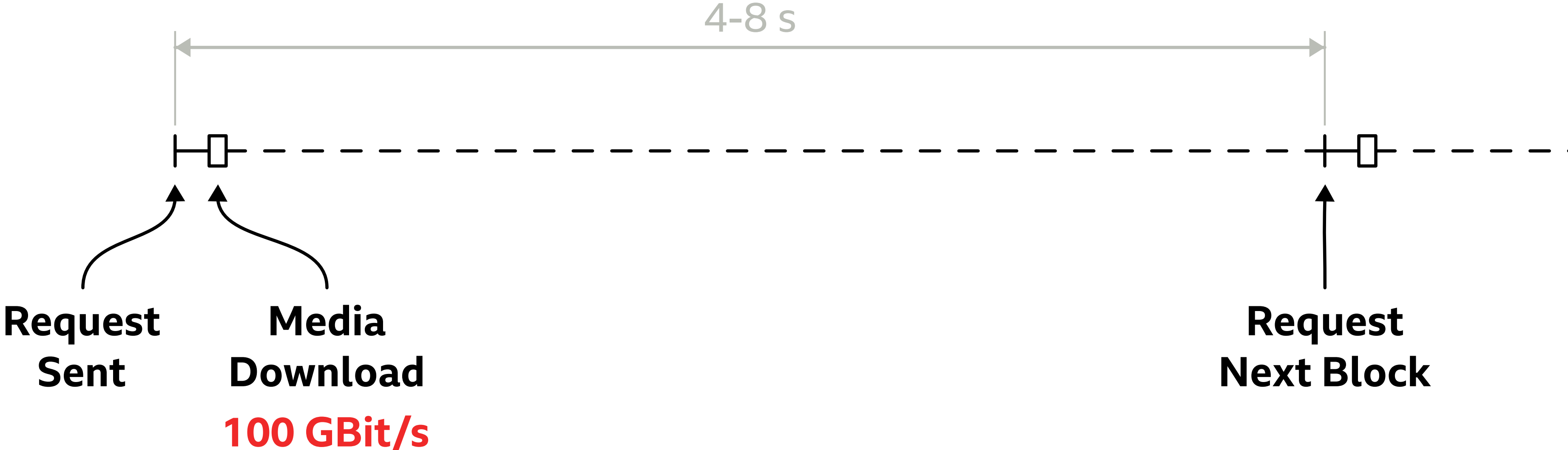
---



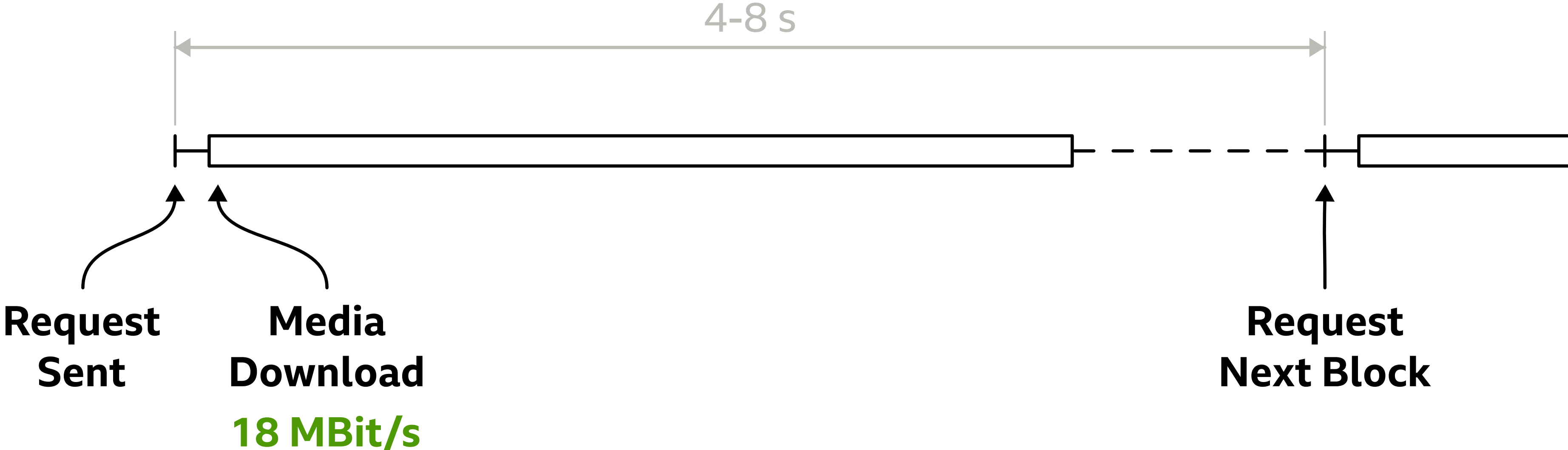
# Test lab realism



# Client behaviour

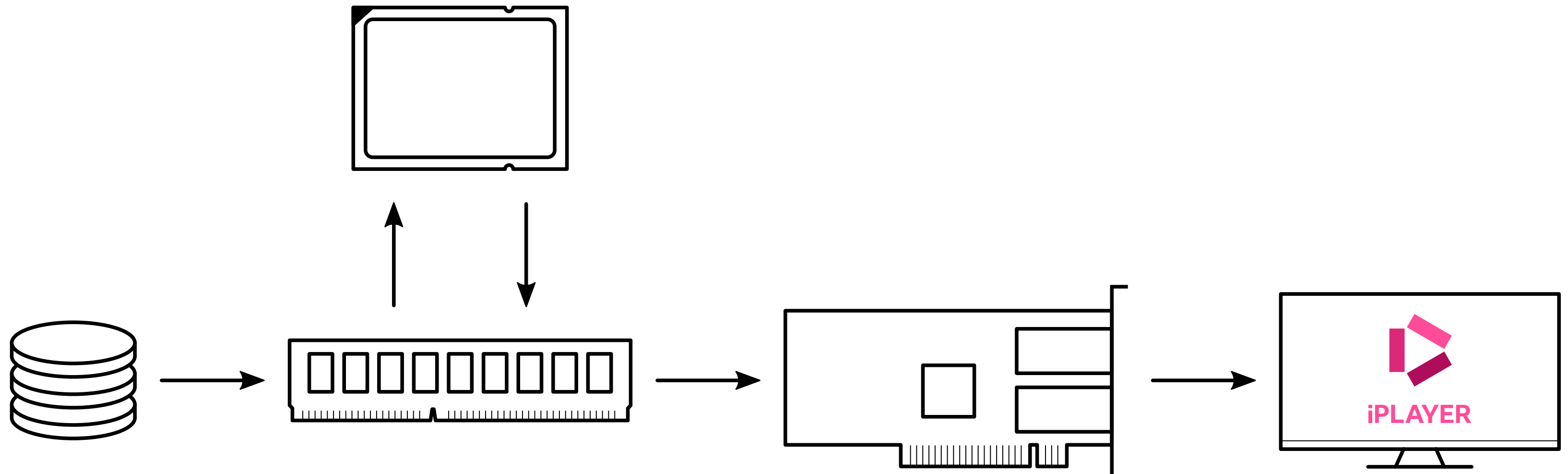


# Client behaviour

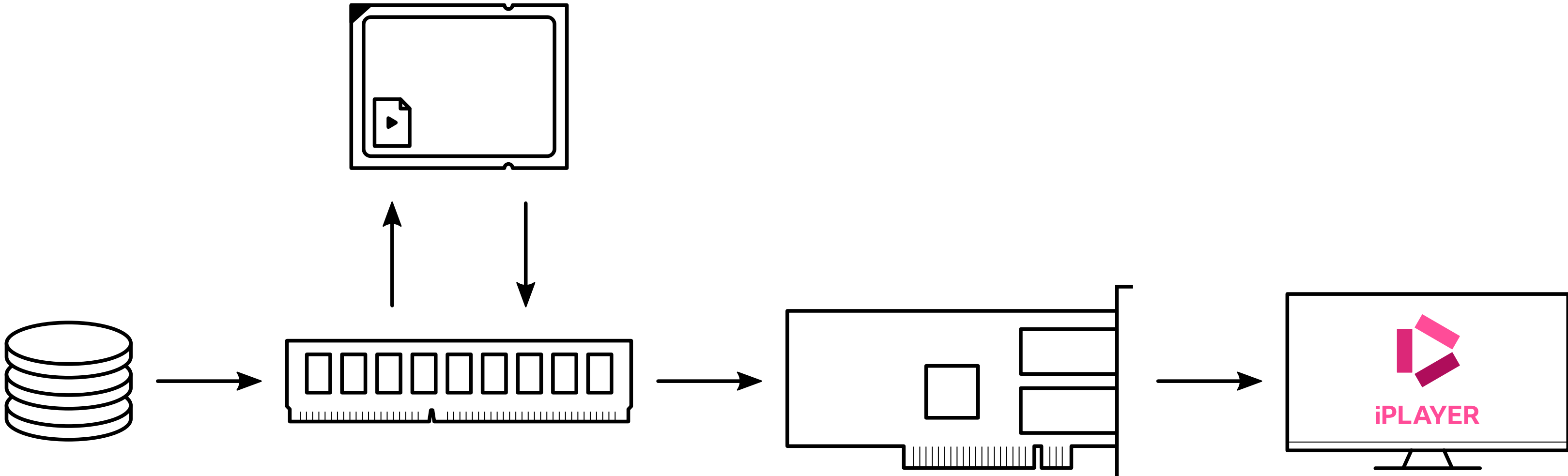


# Slow client effects

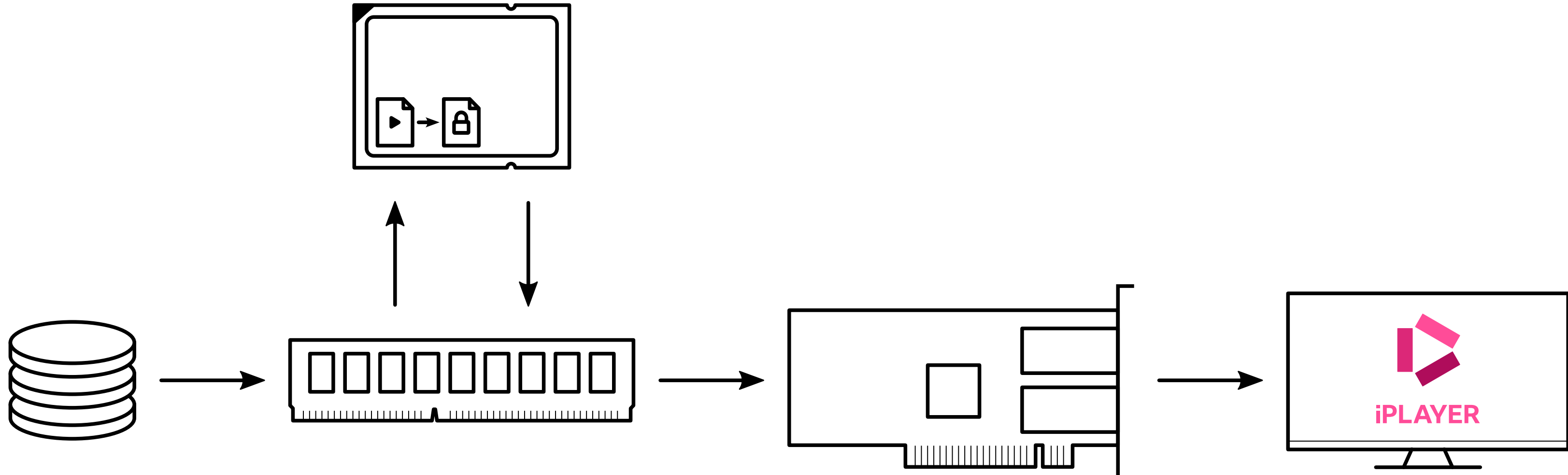
---



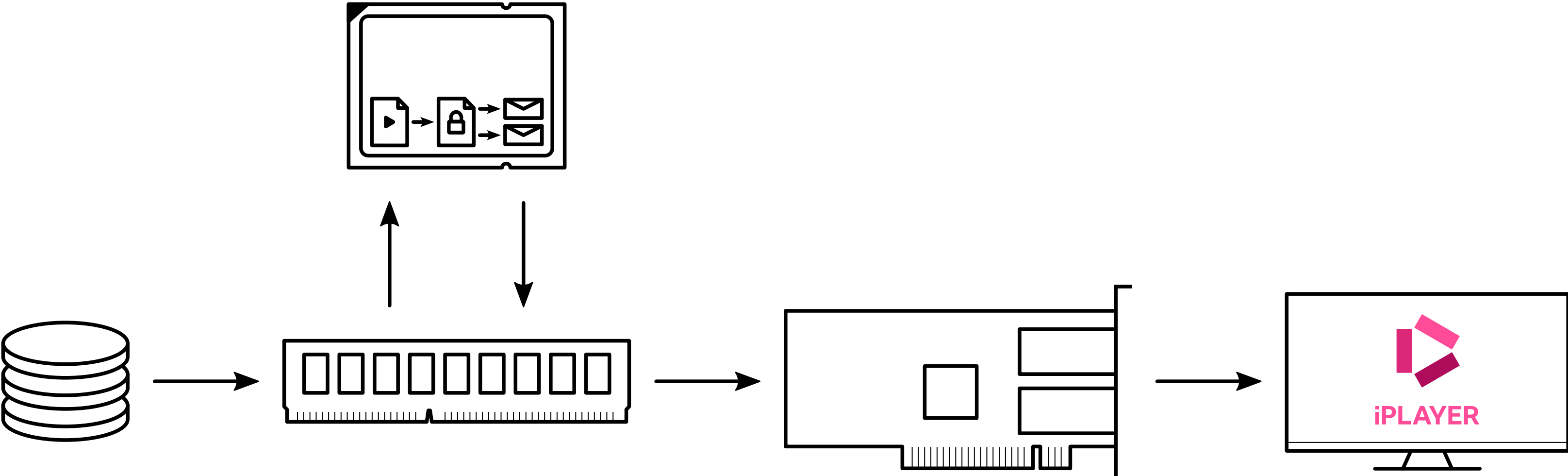
# Slow client effects



# Slow client effects

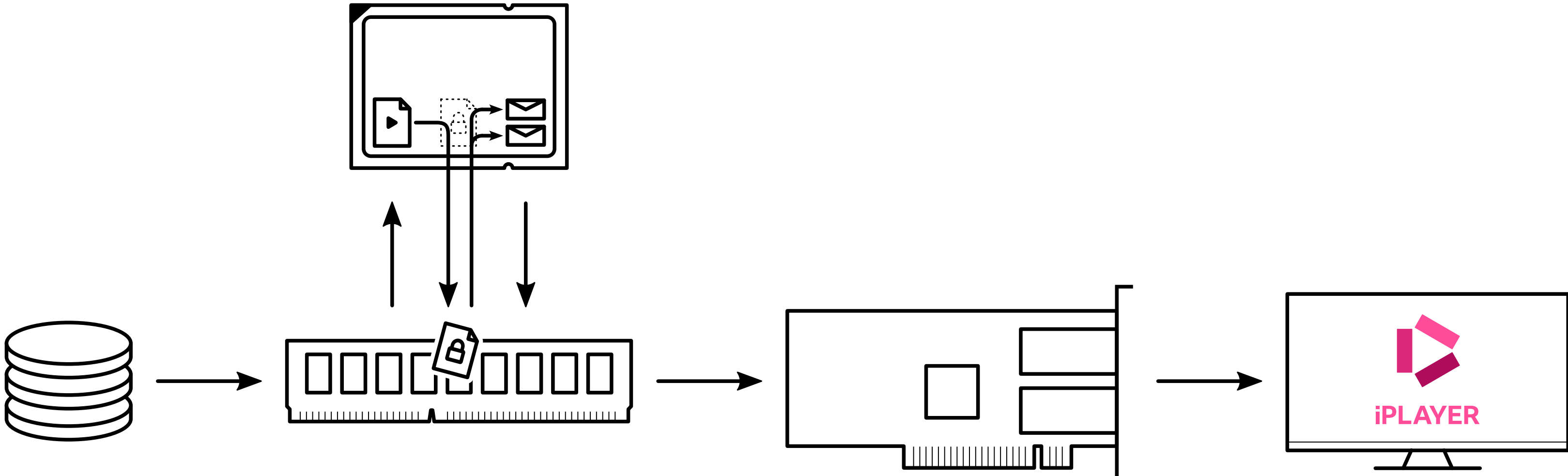


# Slow client effects

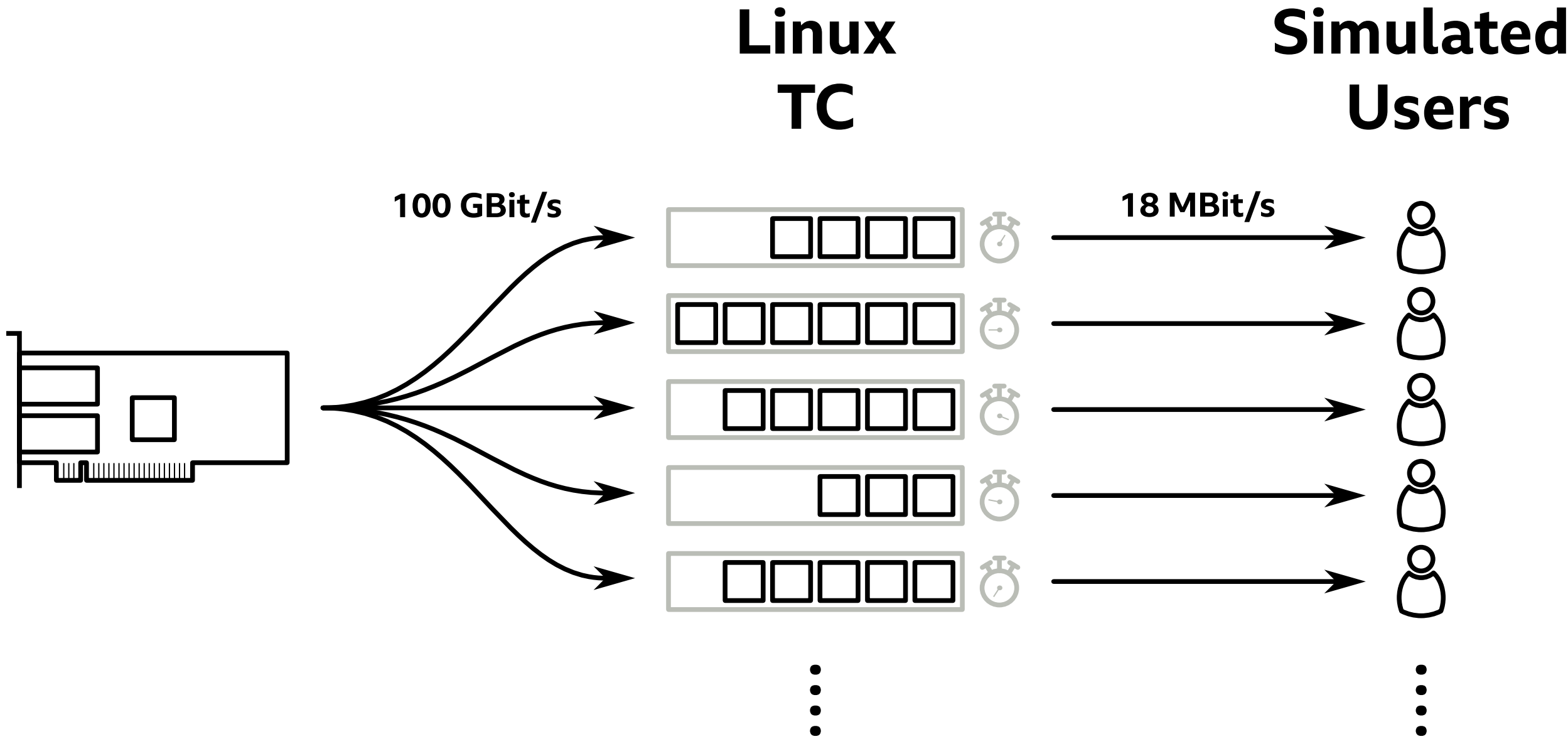




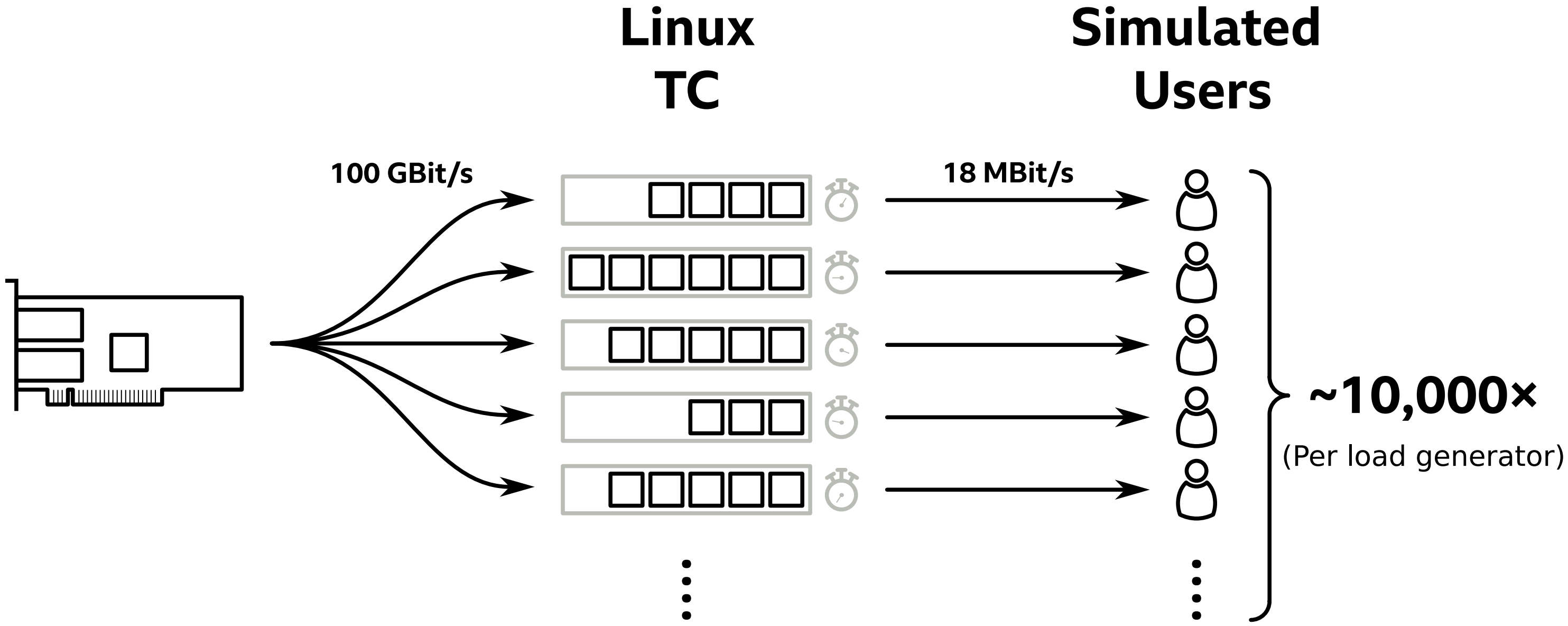
# Slow client effects



# Modelling real internet connections: Linux TC



# Modelling real internet connections: Linux TC

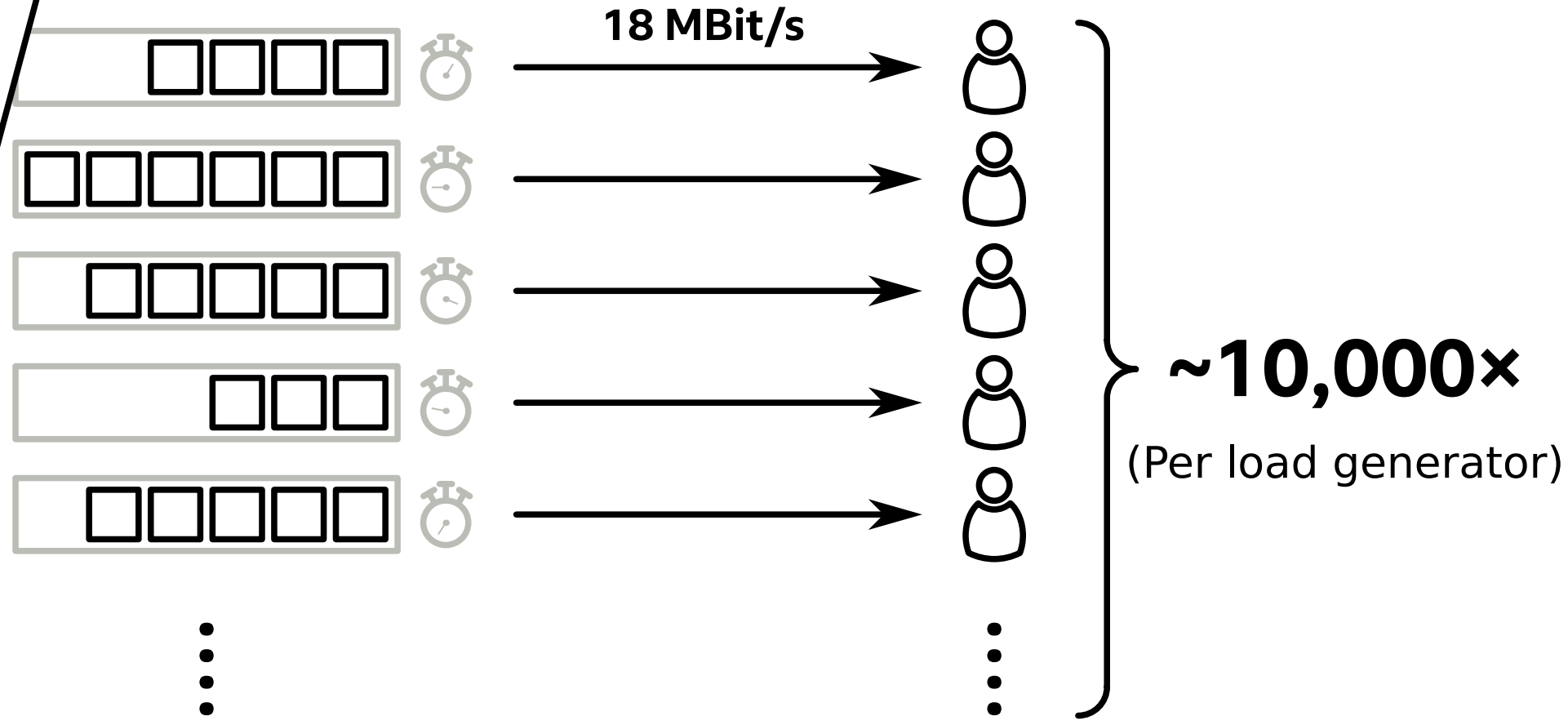


# Modelling real internet connections: Linux TC



Linux TC

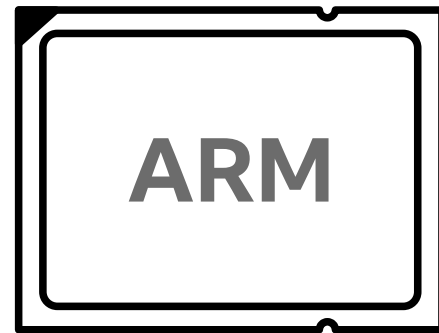
Simulated Users



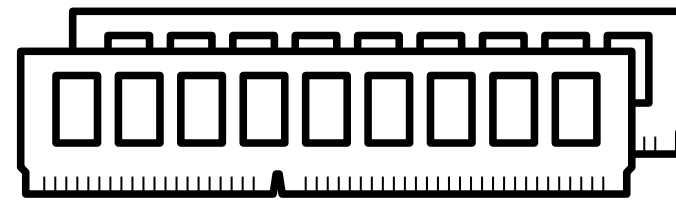
<https://medium.com/criteo-engineering/demystification-of-tc-de3dfe4067c2>

# Conclusions

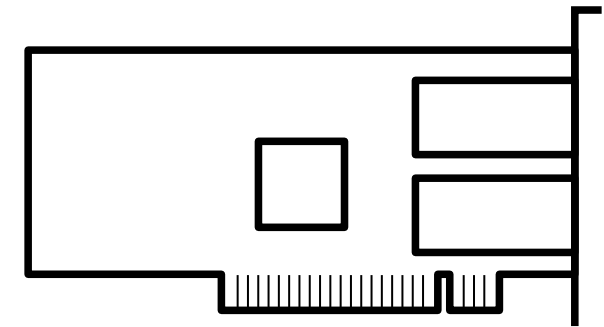
---



**ARM Servers  
Work Well**



**Memory  
Bandwidth now  
the Bottleneck**



**Realising NIC  
Performance  
is Challenging**

# Thank you!

Jonathan Heathcote

**BBC Research & Development**

[jonathan.heathcote@bbc.co.uk](mailto:jonathan.heathcote@bbc.co.uk)

