CS4450

Computer Networks: Architecture and Protocols

Lecture 24
Where's the puck going?

Rachit Agarwal

Announcements

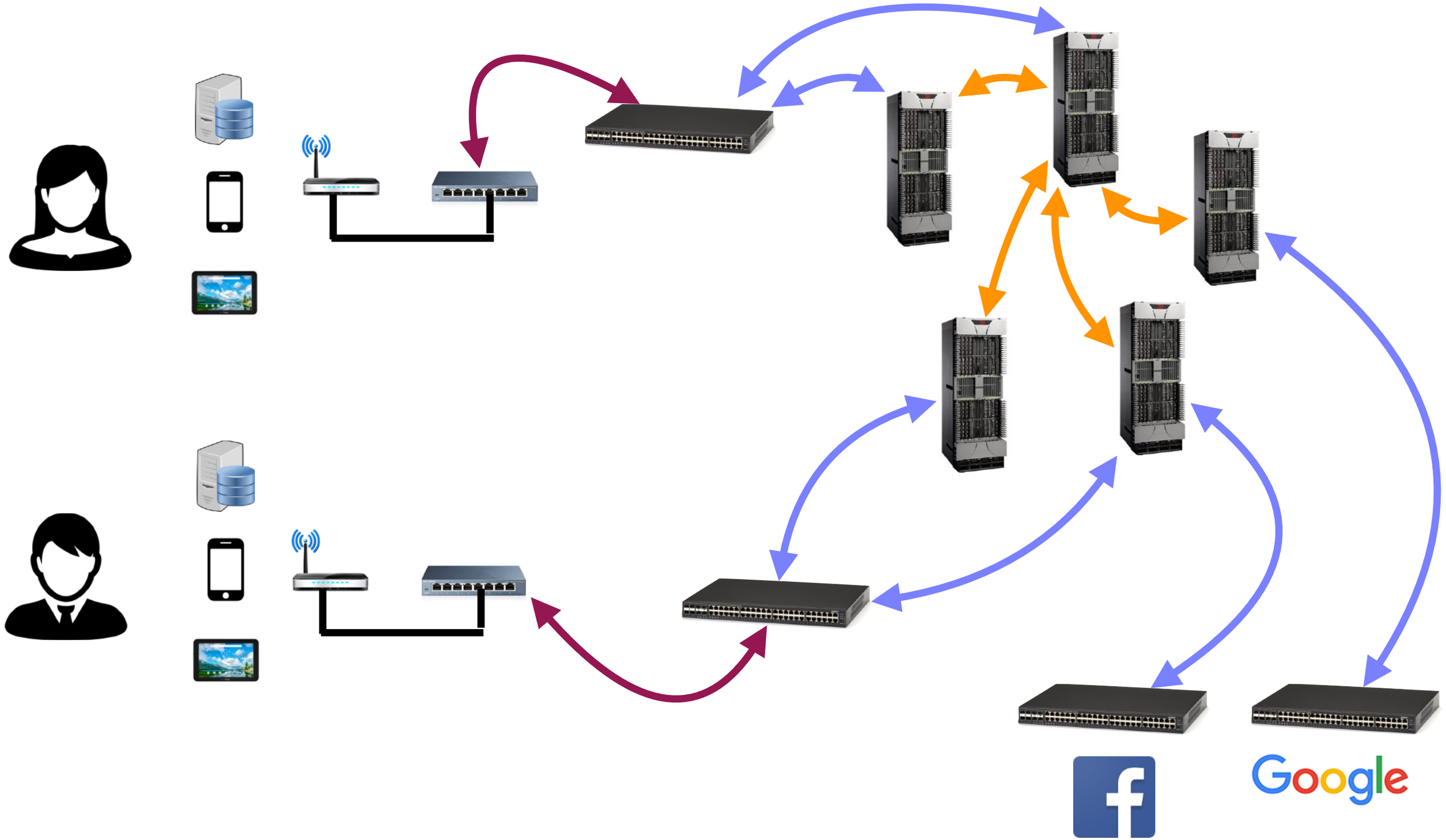
- **Final: 05/18, 05/19 — same format as prelim**

- **Please fill out the course evaluations**
  - Easy way to get 5%
  - **Please be constructive** (evaluations are for many eyes, not just me)

Taking 23 steps back!

What is a computer network?

A set of network elements connected together, that implement a set of protocols for the purpose of sharing resources at the end hosts



Sharing networks

- **Two approaches**
    - Reservation (circuit switching)
    - Statistical multiplexing (packet switching)

- **Motivation for WHY modern networks use “packets”**

- **How to implement this?**

The end-to-end story

- Application opens a **socket** that allows it to connect to the **network stack**
- Maps **name** of the web site to its **address** using **DNS**
- The network stack at the source embeds the address and **port** for both the source and the destination in **packet header**
- Each **router** constructs a **routing table** using a distributed algorithm
- Each router uses destination address in the packet header to look up the **outgoing link** in the routing table
    - And when the link is free, forwards the packet
- When a packet arrives the destination:
    - The network stack at the destination uses the port to forward the packet to the right application

Realizing end-to-end design: Three Principles

- How to break system into modules
  - **Layering**

- Where are modules implemented
  - **End-to-End Principle**

- Where is state stored?
  - **Fate-Sharing**

Five Layers (Top - Down)

- **Application:** Providing network support for apps
- **Transport (L4):** (Reliable) end-to-end delivery
- **Network (L3):** Global best-effort delivery
- **Datalink (L2):** Local best-effort delivery
- **Physical:** Bits on wire

Link Layer (L2)

- **Broadcast medium:** Ethernet and CSMA/CD

- **We studied that Broadcast Ethernet does not scale to large networks**
    - Motivation for switched Ethernet

- **Broadcast storm:** if using broadcast on switched Ethernet
    - Motivation for Spanning Tree Protocol

- **Limitations of Spanning Tree Protocol:**
    - Low bandwidth utilization, high latency, unnecessary processing
    - Does not scale to the entire Internet
    - Motivation for **routing protocols** in the Internet

Network Layer (L3)

- **Internet Protocol:**
    - Addressing, packet header as an interface, routing
- **Routing tables:**
    - Correctness and validity: Dead ends, loops
    - A collection of spanning trees, one per destination
- **Constructing valid routing tables (within an ISP)**
    - Link-state and distance-vector protocols
    - Focused a lot on learning via examples
    - Can still have loops: failures remain to be a pain
- **How to use routing tables**
    - **Packet header as an interface**
    - Learnt why packet headers look like the way they do

Network Layer (L3), Cont.

- **Internet Protocol:**
    - Addressing, packet header as an interface, routing
- **Addressing:**
    - Link layer uses “flat” addresses
    - **Does not scale to Internet**: motivation for IP addresses
    - **Scalability challenges**: Routing table sizes, #updates
    - Solution: **Hierarchical addressing**
- **Forwarding**
    - **Switch architecture**
    - **Longest Prefix matching for forwarding at line rate**
    - Scheduling using priorities

Network Layer (L3), Cont.

- **Internet Protocol:**
    - Addressing, packet header as an interface, routing
- **Limitations of link-state and distance-vector routing:**
    - Require visibility of the entire Internet
    - **ISPs do not like that:** motivation for Inter-domain routing
    - **Border Gateway Protocol**
        - A simple modification of distance-vector protocol
- **Routing with policies**
    - **Customer-provider-peer relationships**
    - **Gao-Rexford policies**
- **Completes the network layer: provides connectivity**

Details for complete picture

- **DHCP: Dynamic Host Configuration Protocol**
    - For each host to figure out its IP address, local DNS, first-hop router

- **ARP: Address Resolution Protocol**
    - For finding other servers on the same local area network (L2)
    - Mapping from IP addresses to names (MAC addresses)

- **Domain Name System**
    - **Mapping Human readable destination names to IP addresses**
    - Hierarchical structure

Transport Layer

- **Goals of reliable transport**
    - **Correctness condition**
    - Why do we need ACKs, timers, window-based design

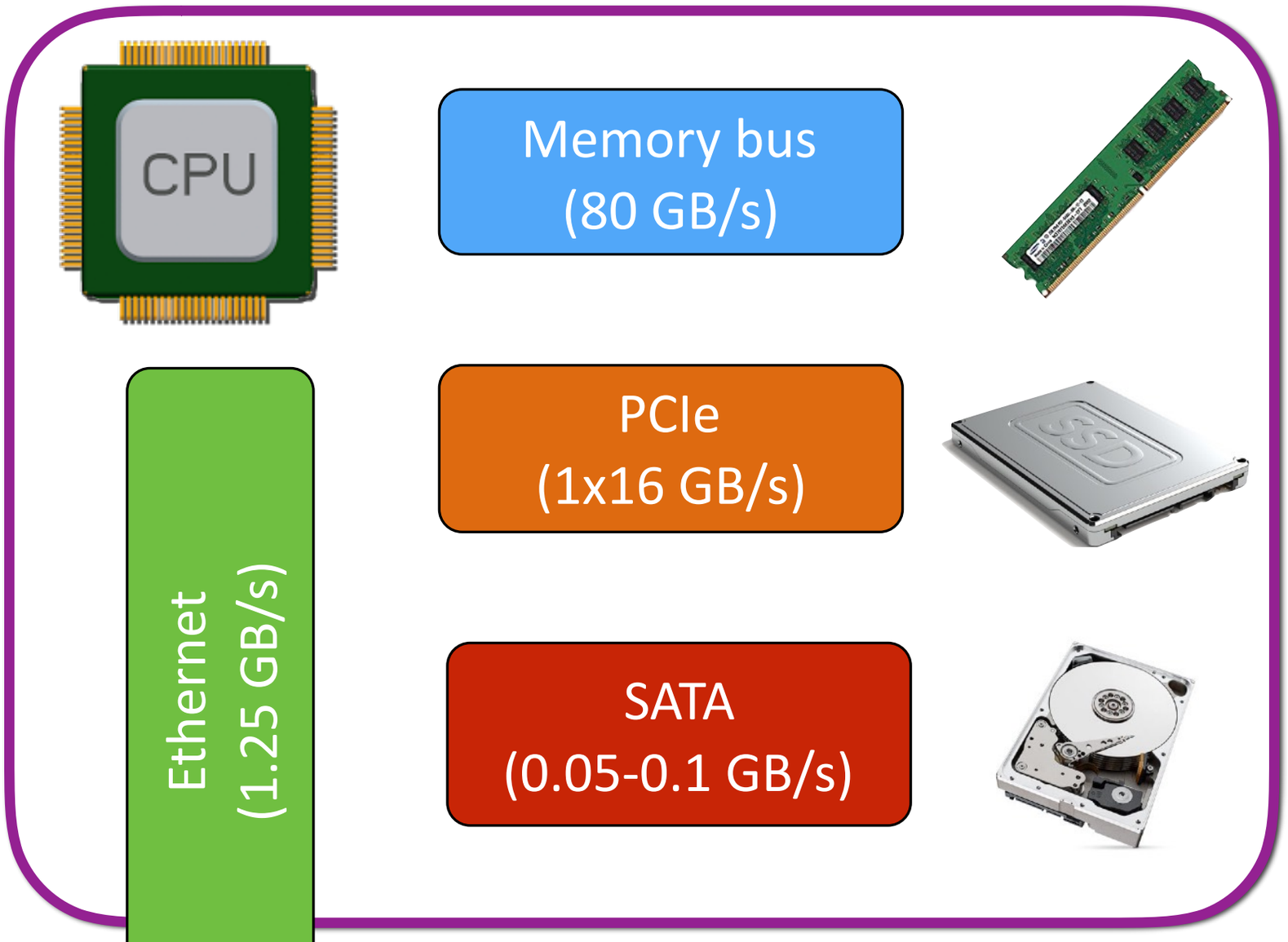
- **One realization of reliable transport: TCP**
    - Mostly implementation details following the above design
    - For **max-min fairness**, flow performance and utilization
    - **Flow control**
        - Ensuring the sender does not overwhelm the **receiver**
        - Via receiver advertised window size
    - **Congestion control**
        - Ensuring the sender does not overwhelm the **network**
        - Slow start, Additive-increase Multiplicative-decrease, timeouts

Taking 1 step forward!

Skate where the puck's going, not where it's been!

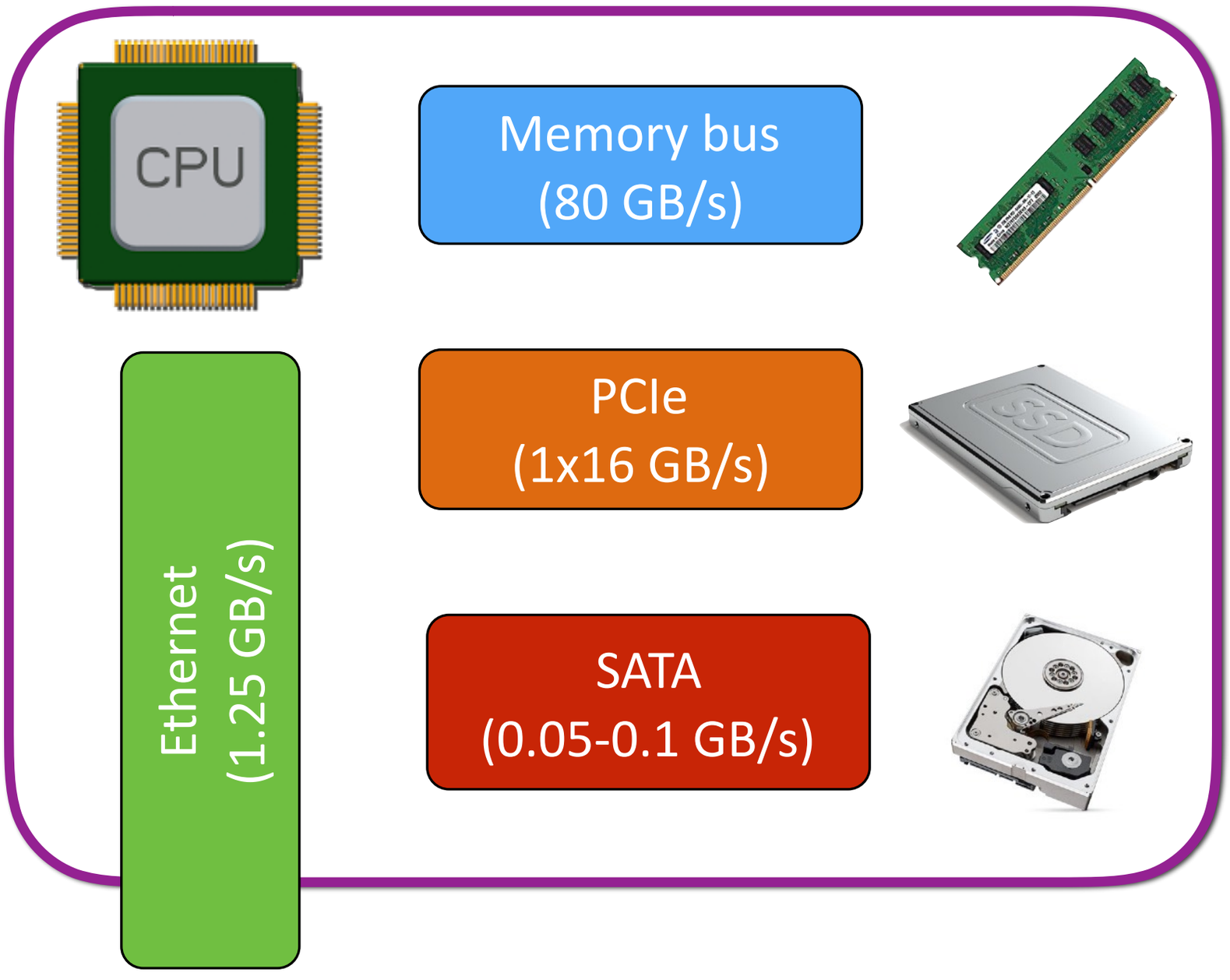
- Walter Gretzky

Where is the puck right now?

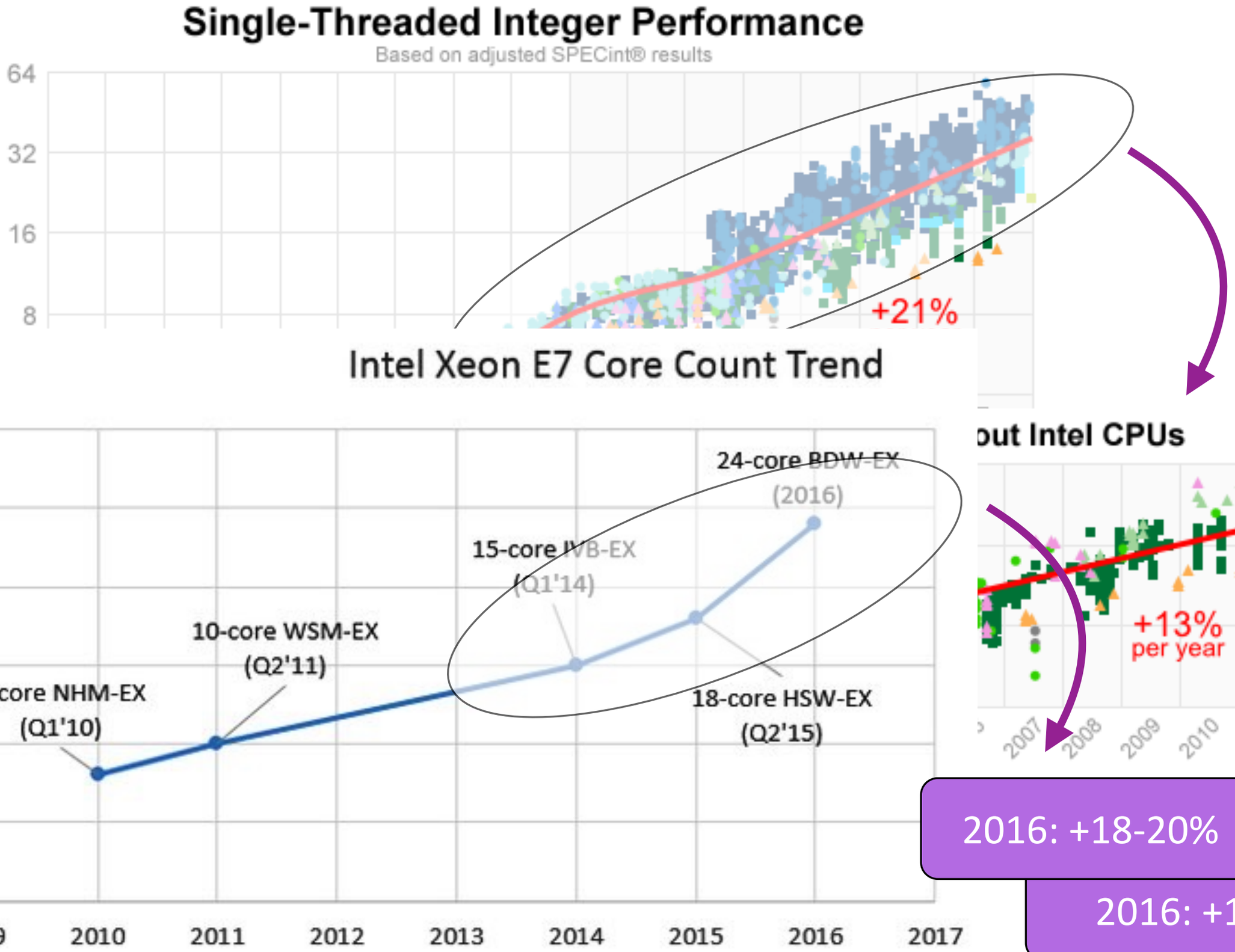


Size (TB)	Random Access (us)	Seq. Access (GB/s)
0.1	0.1	80
1	25	1x
10	4000	0.1x

Where is the puck going?

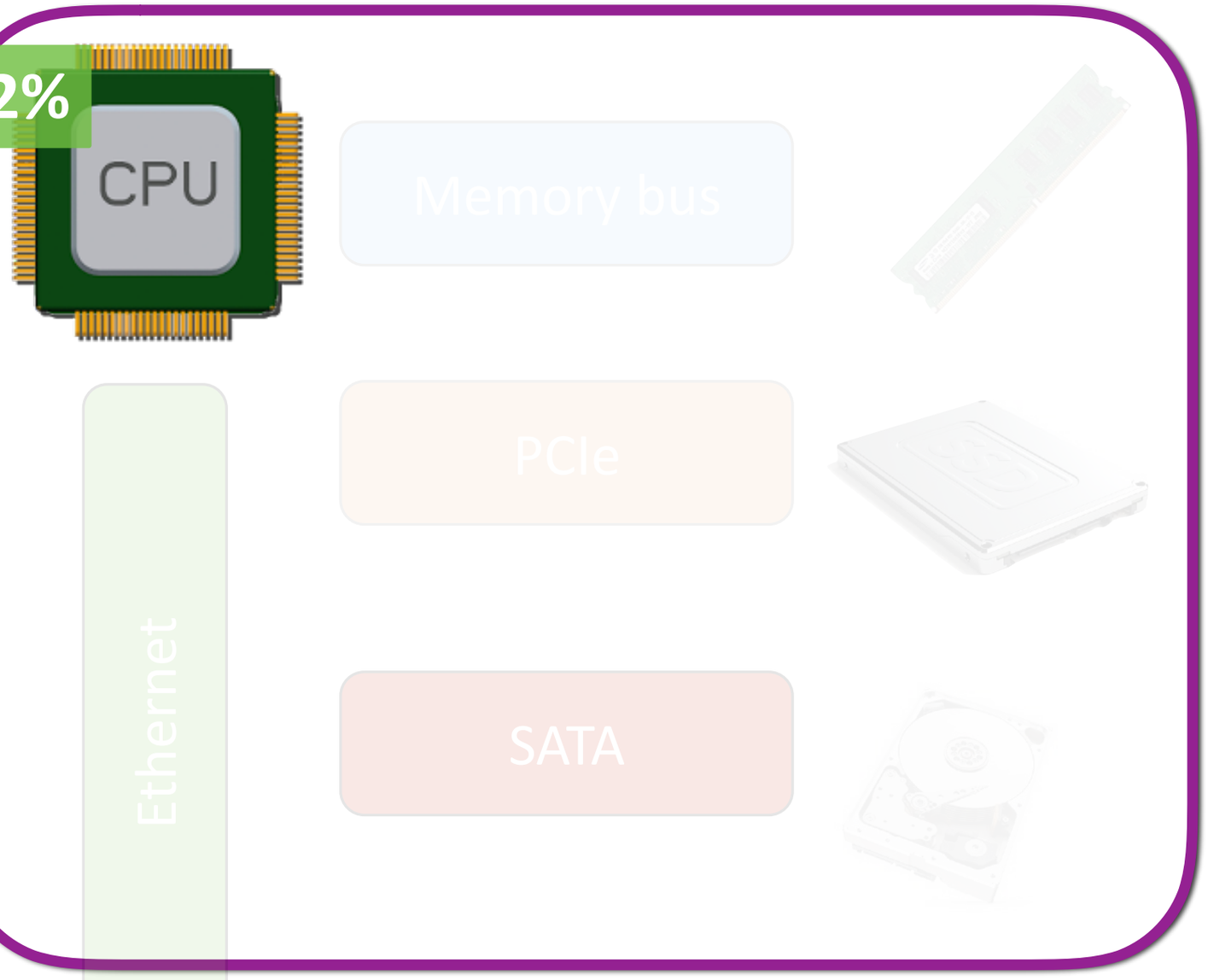


Where is the puck going? (CPU performance)



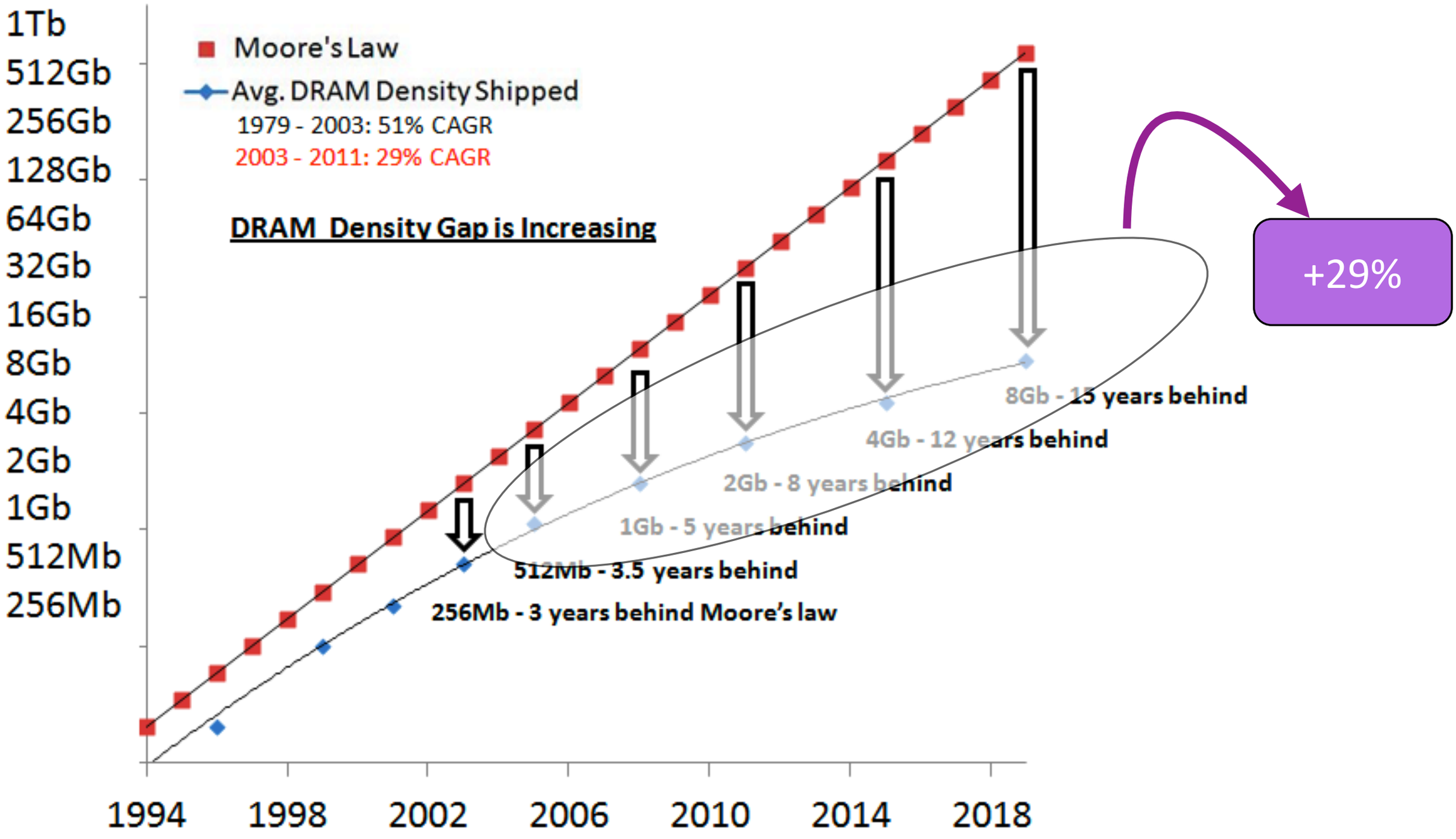
Where is the puck going?

+30-32%

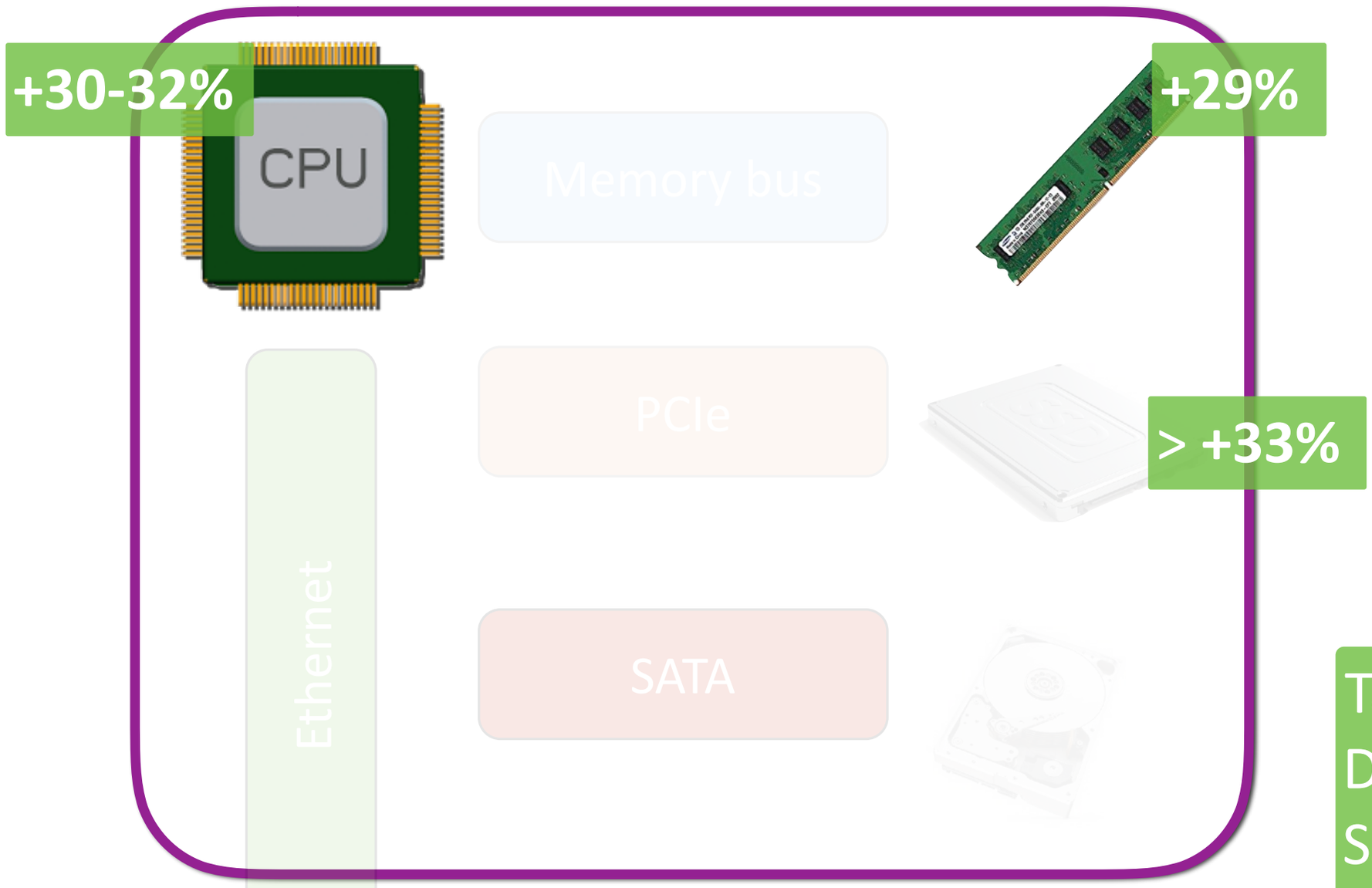


- **#Cores:** +18-20%
- **Per core:** +10%

Where is the puck going? (DRAM capacity)



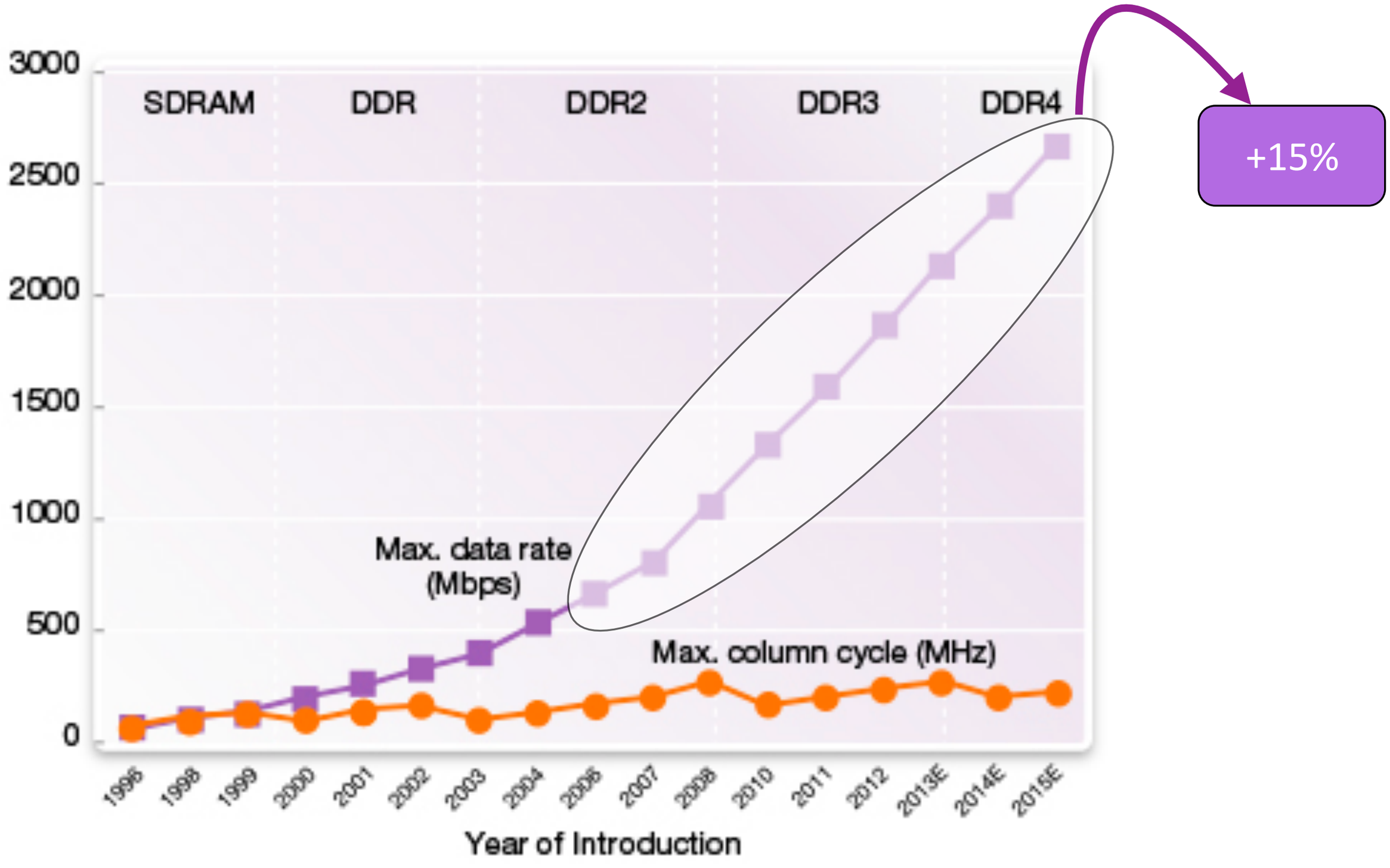
Where is the puck going?



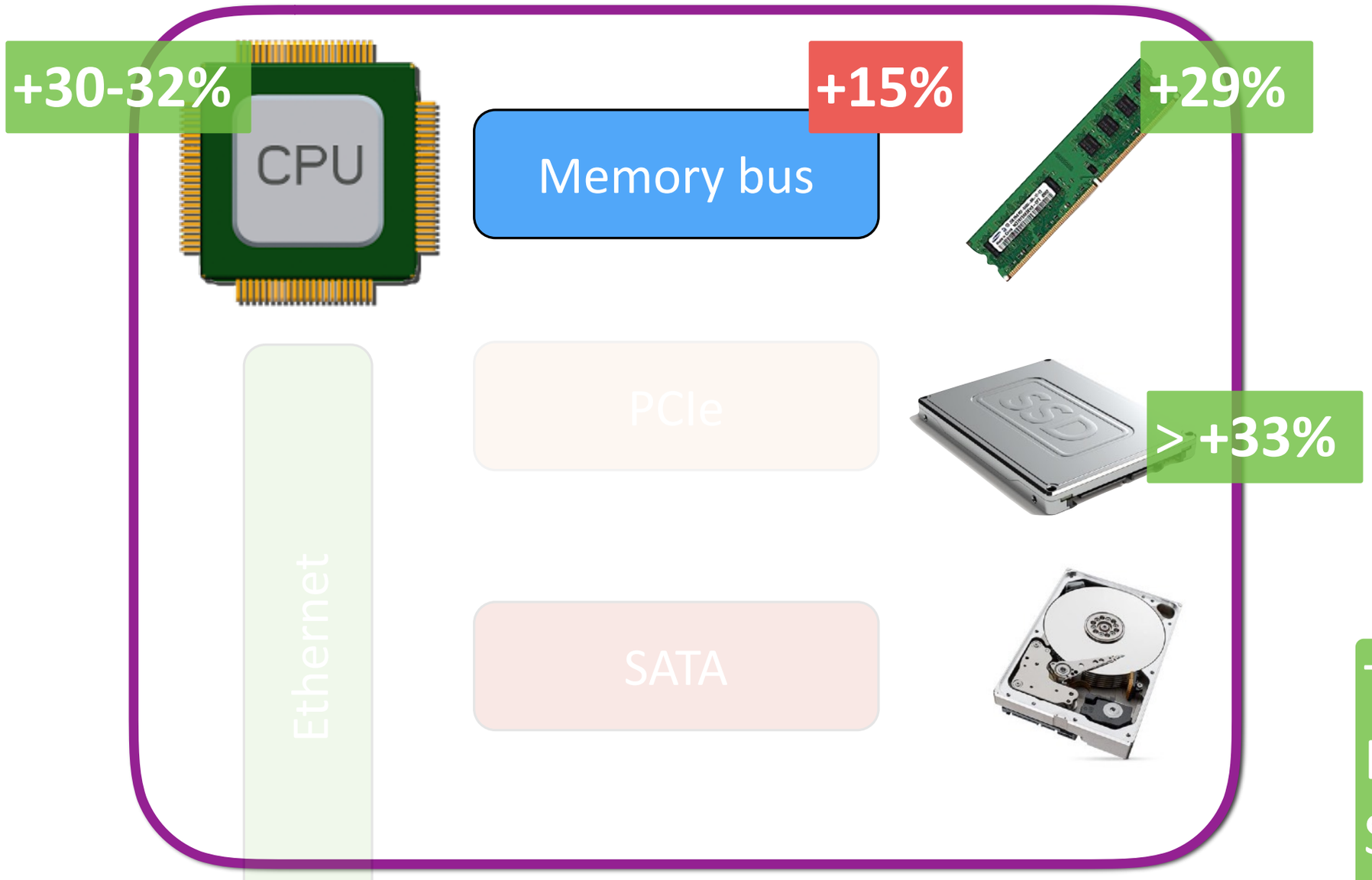
Tape is dead,
Disk is tape,
SSD is disk,
RAM is the king!

- Jim Gray

Where is the puck going? (Memory bus)



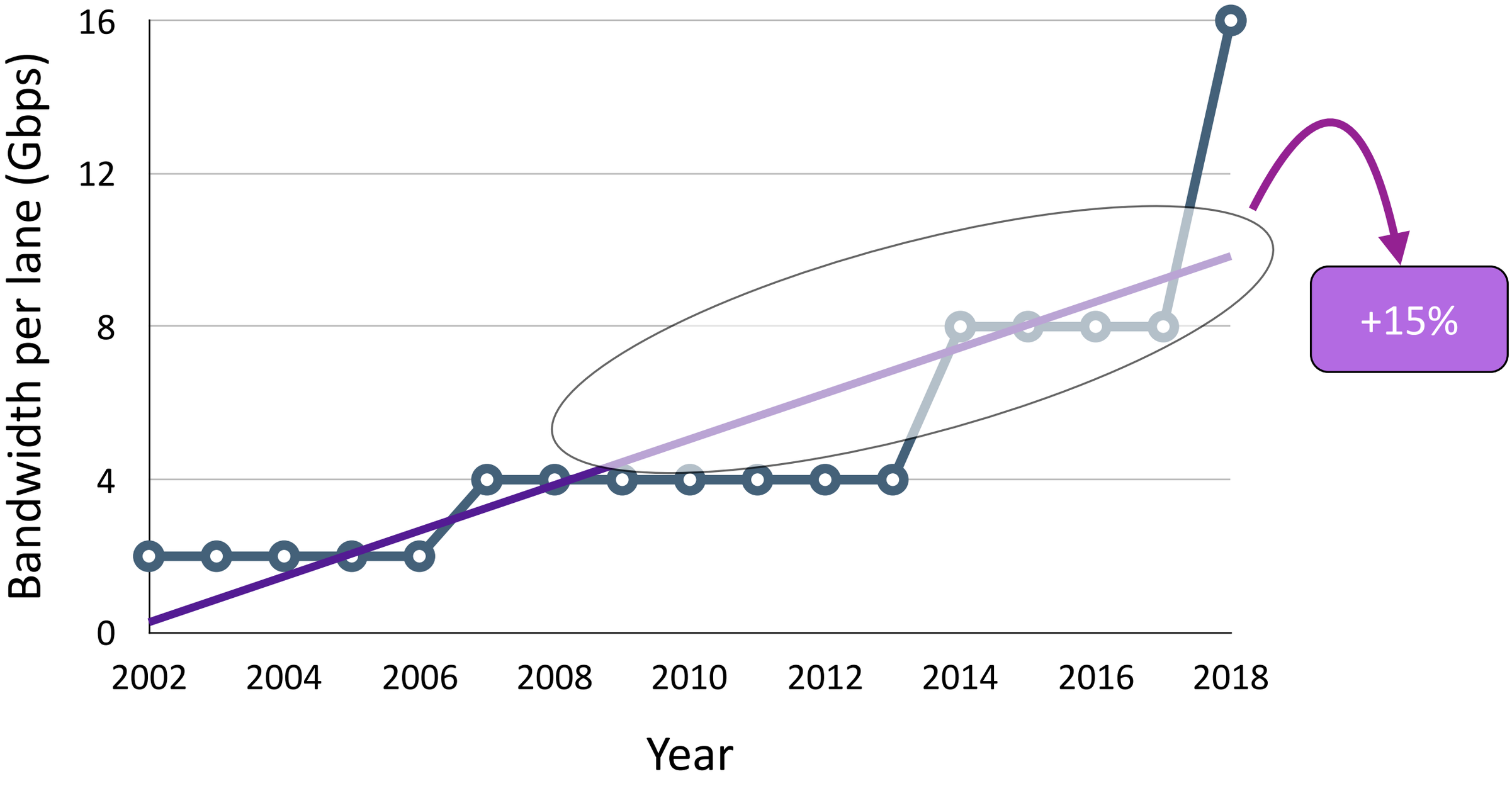
Where is the puck going?



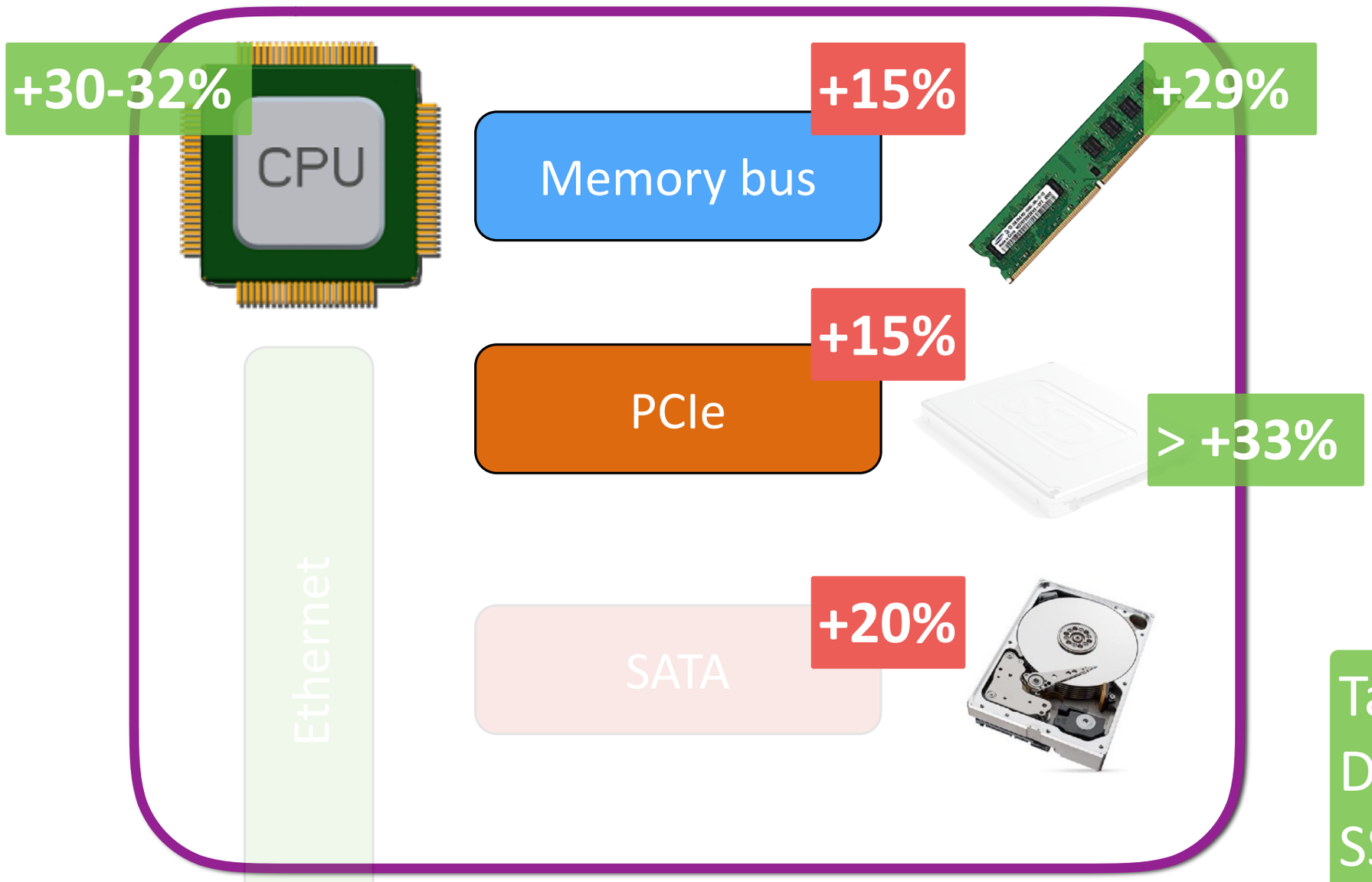
Tape is dead,
Disk is tape,
SSD is disk,
RAM is the king!

- Jim Gray

Where is the puck going? (PCIe)



Where is the puck going?

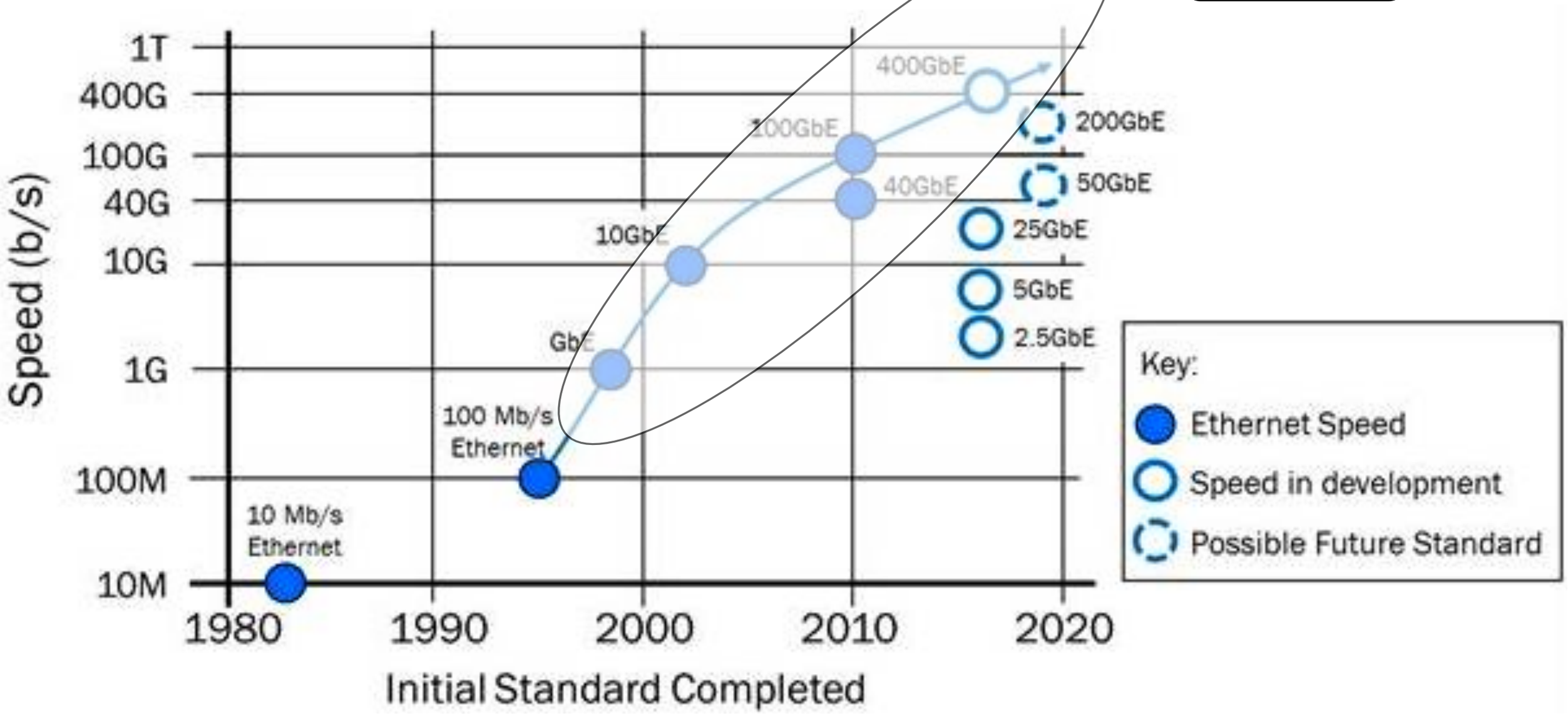


Tape is dead,
Disk is tape,
SSD is disk,
RAM is the king!

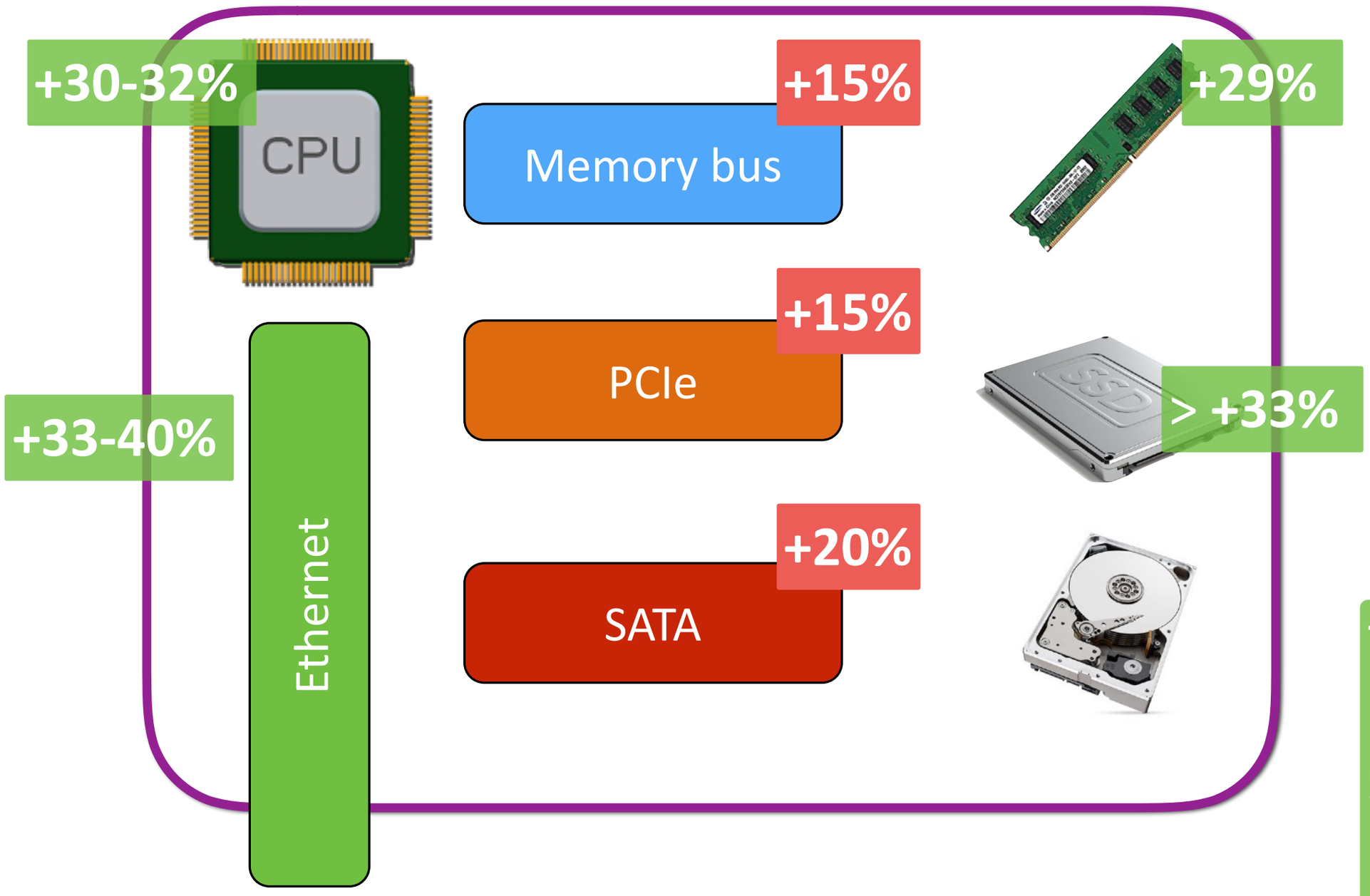
- Jim Gray

Where is the puck going? (Ethernet)

+33-40%



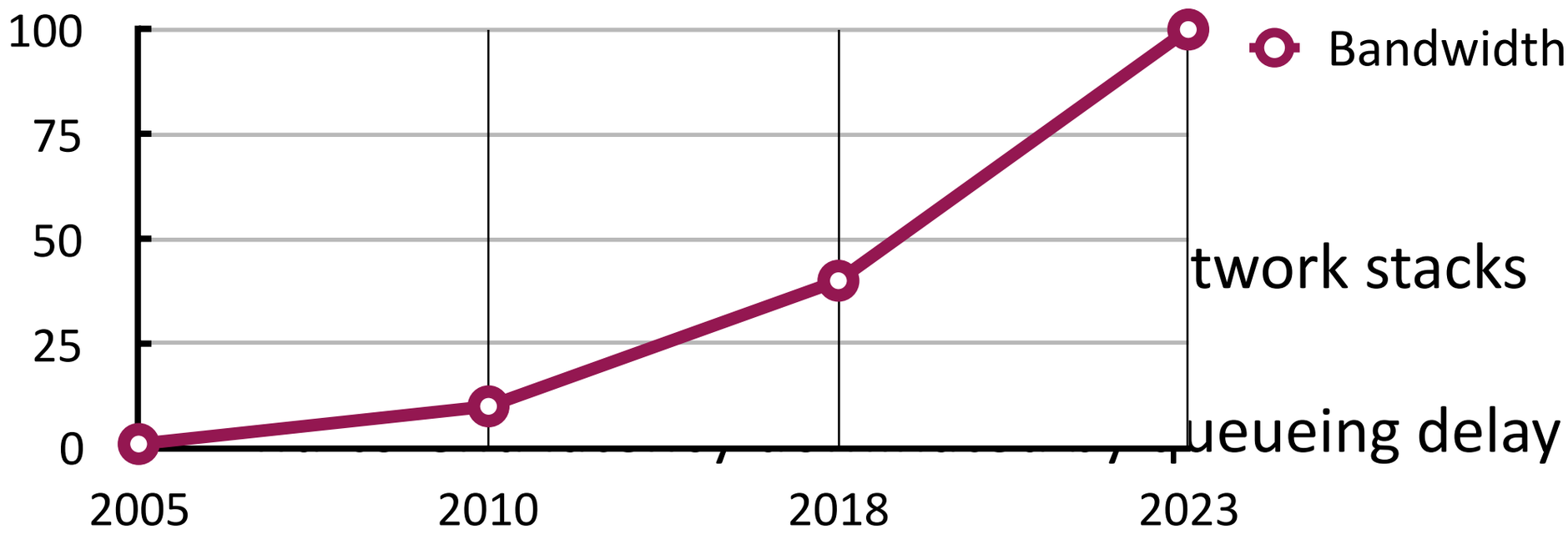
Where is the puck going?



Tape is dead,
Disk is tape,
SSD is disk,
RAM is the king!

- Jim Gray

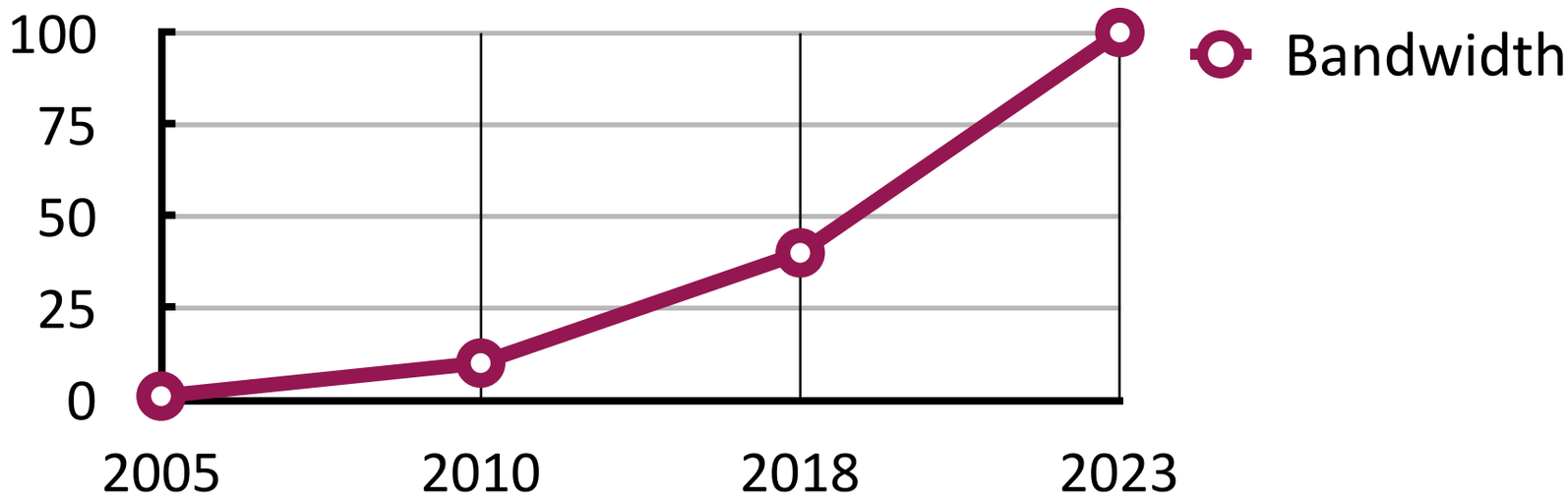
Network Technology Trends



Powerful implications

- Remote memory faster than local SSD
  - When queueing delay = 0

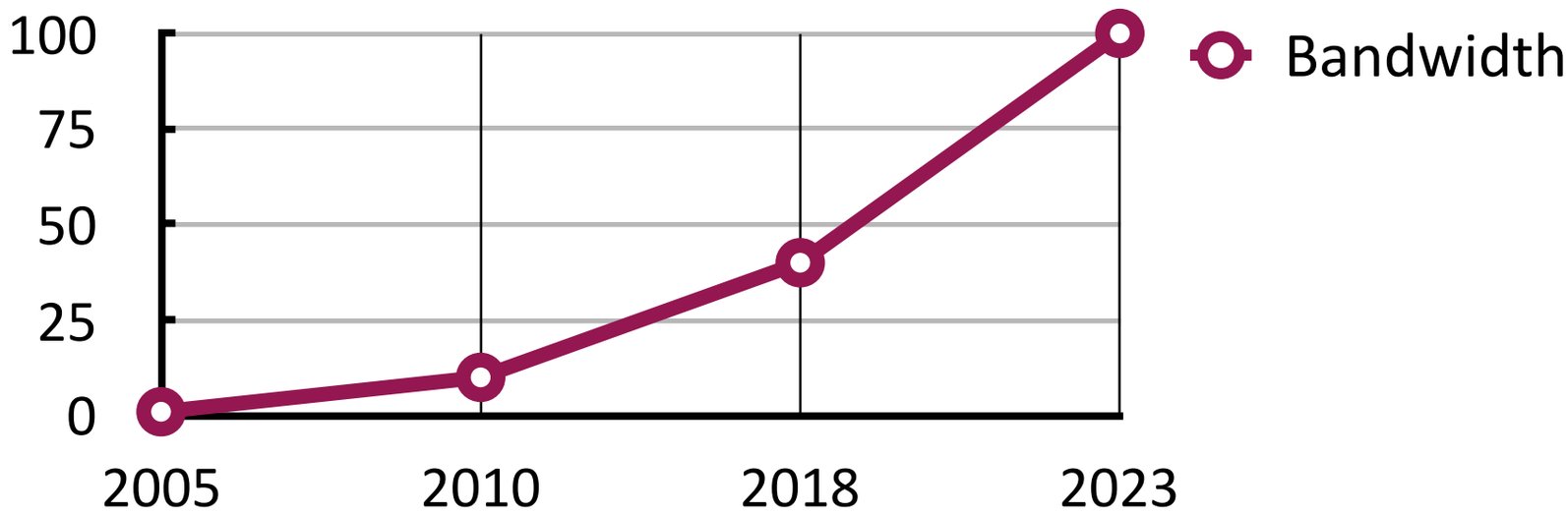
Unsustainable CPU overheads



- **Existing network stacks were designed for 1Gbps networks**
  - Known TCP problem: ~3.2Gbps per core
  - With low-level optimizations: ~9-12Gbps per core
    - 40Gbps would take >3 cores per server!
    - **100Gbps would take >8 cores per server!!**

- **Take away: unsustainable cloud economics**
  - Every core used for the stack is a core stolen from applications/ customers

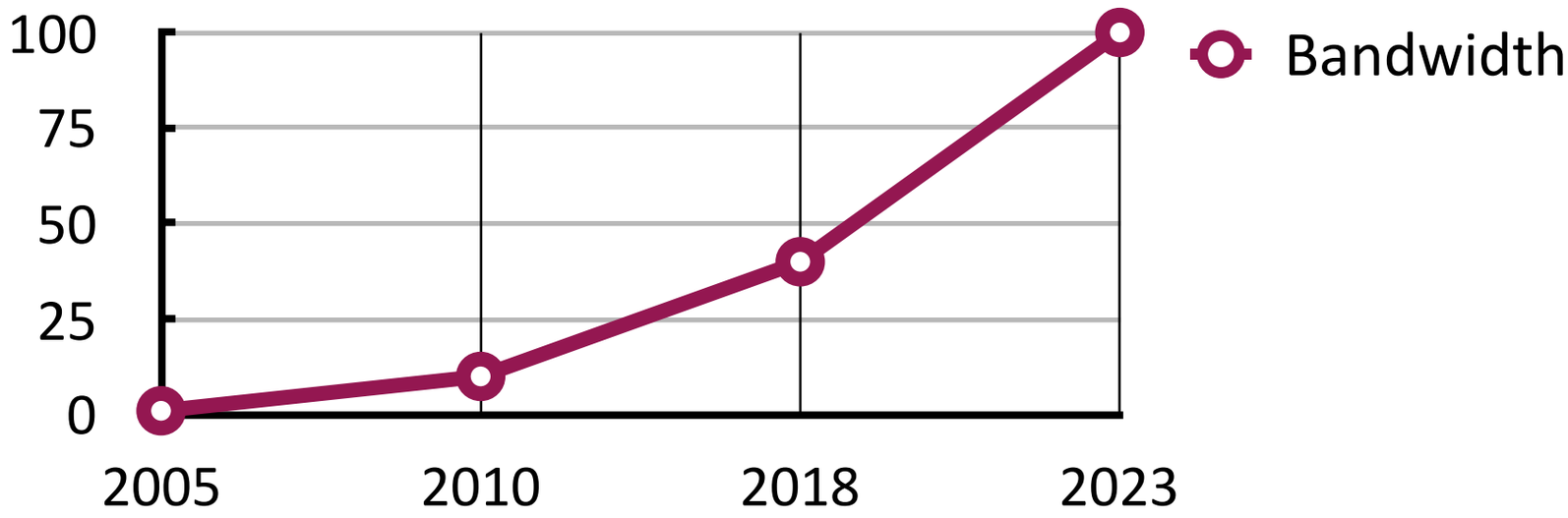
Curse of queueing delay



	2018 (40Gbps)
	Latency (us)
Latency for an uncontested network	6.30
Max Queueing delay (32MB buffers)	6400 (per congested switch/router)

- Take away: queueing delay is the core bottleneck
  - End-to-end latency bottlenecked by queueing delay

Remote Memory Faster than Local Storage



- **Under zero queueing:**
  - Remote memory access takes less than 6.3us
  - Local SSD access latency today is 25us (hardware, ignoring stack)
  - Accessing remote memory is faster than local SSD!!!

- **If we can design zero-queue networks**
  - **We can fundamentally re-architect computer systems**

Current Network Stacks are the Bottleneck!

- **Lot of research in "hardware offload"**
    - Implementing TCP (and other mechanisms) on hardware
    - Lots of interesting challenges

- **Lot of research in low-latency transport design**
    - TCP was not designed for low latency
    - New transport protocols for ultra low-latency

- **Lot of research in kernel-bypass**
    - TCP requires processing each and every packet
    - 1Gbps links: 90,000 packets per second
    - 100Gbps links: 9 million packets per second
    - Extremely high CPU requirements
    - Bypass the kernel entirely
        - Implement congestion control in user space, in hardware?

Closing Thoughts

- **These are exciting times for computer networking**
    - The first ever since the invention of the Internet
    - You are witness to the transformation!!!!

- **And, I am glad I got the chance to introduce you to this world :-)**
    - You have made me a better teacher!!!! Thank you.

Closing Thoughts

- Hard to recognize what we have achieved this semester
    - We did not give in to tough times!
    - Your efforts instrumental!
    - Thank you for working with me on remote education!
    - **BE rightfully proud** of what you have accomplished this semester!

- Wherever you end up:
    - Please remember me; say hello if you see me
    - Remember, there is nothing more important than
        - **Knowing the fundamentals!!!!**
        - **Being happy!!!!**