Revolutionizing Datacenter Networks via Reconfigurable Topologies

Stefan Schmid (TU Berlin)

"We cannot direct the wind,
but we can adjust the sails."

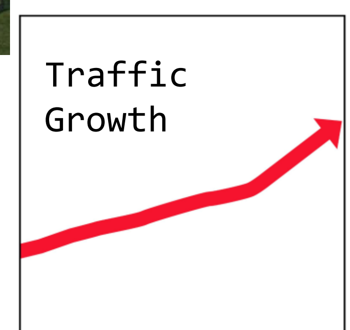
(Folklore)

Acknowledgements:

Trend

Data-Centric Applications

Datacenters (“hyper-scale”)



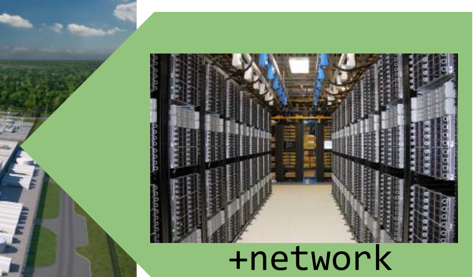
Source: Facebook

Interconnecting networks:
a **critical infrastructure**
of our digital society.

Trend

Data-Centric Applications

Datacenters ("hyper-scale")



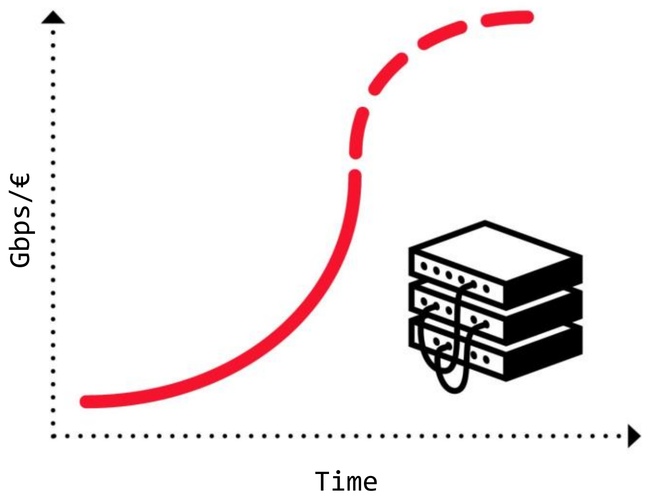
Interconnecting networks:
a **critical infrastructure**
of our digital society.

Credits: Marco Chiesa

The Problem

Huge Infrastructure, Inefficient Use

- Network equipment reaching capacity limits
  - Transistor density rates stalling
  - “End of **Moore‘s Law** in networking”
- Hence: more equipment, larger networks
- Resource intensive and: **inefficient**



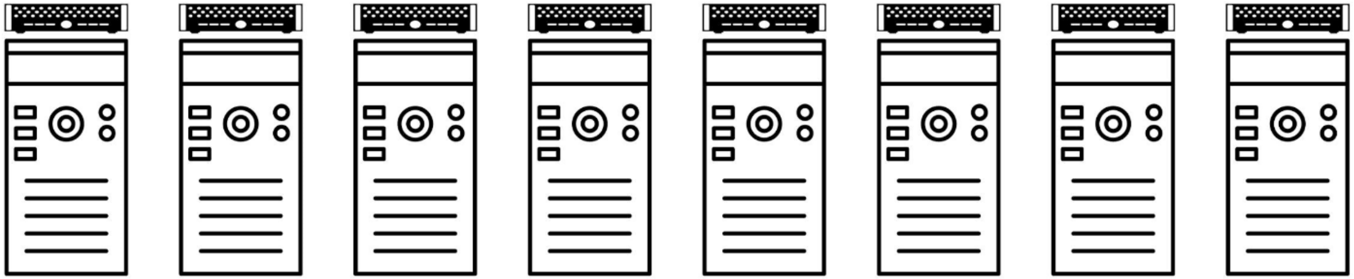
[1] Source: Microsoft, 2019

Annoying for companies, **opportunity** for researchers!

Root Cause

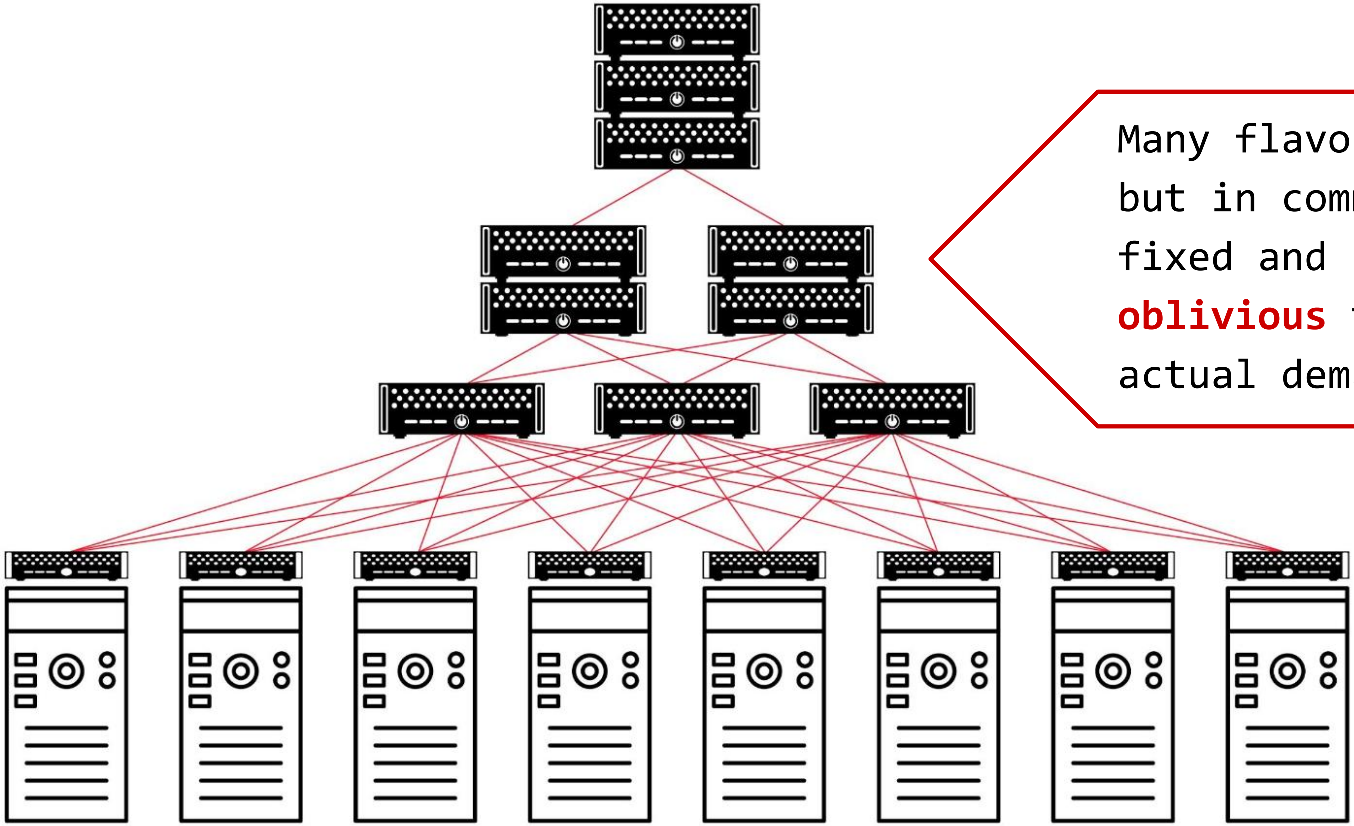
Fixed and Demand-Oblivious Topology

How to interconnect?



Root Cause

Fixed and Demand-Oblivious Topology

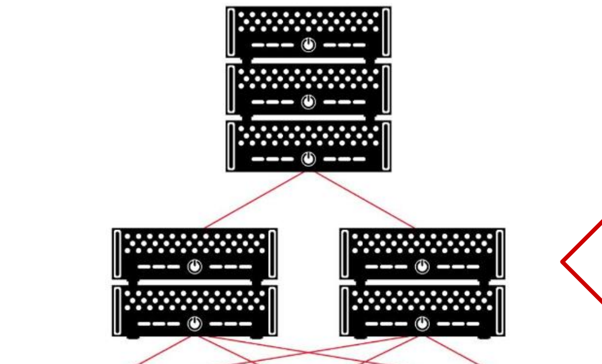


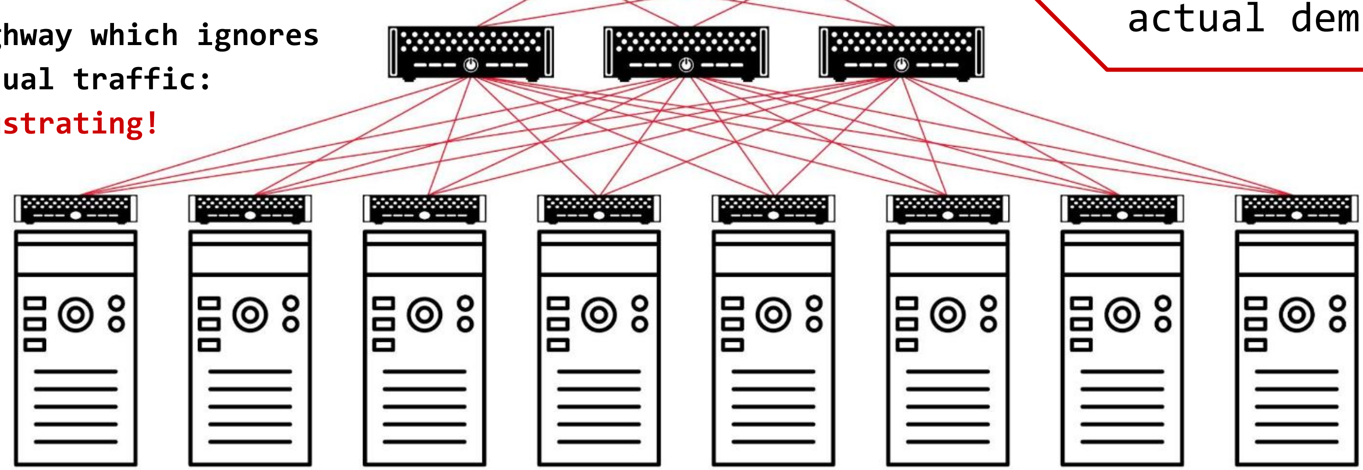
Many flavors,
but in common:
fixed and
oblivious to
actual demand.

Root Cause

Fixed and Demand-Oblivious Topology

Highway which ignores actual traffic: **frustrating!**

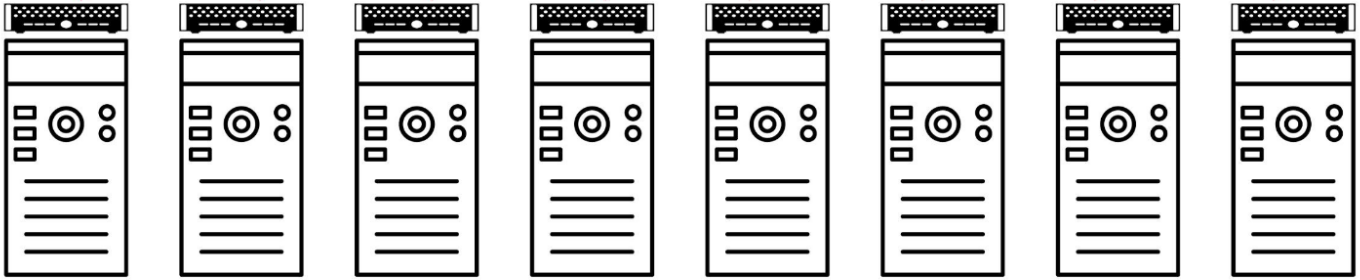




Many flavors, but in common: fixed and **oblivious** to actual demand.

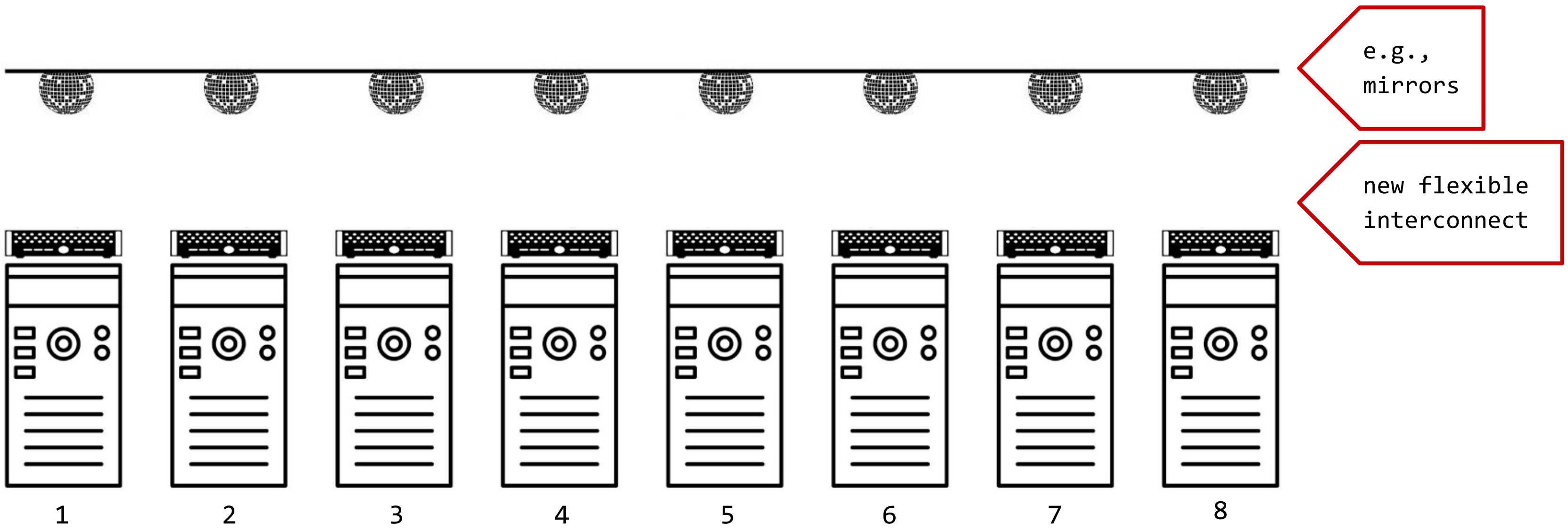
A Vision

Flexible and Demand-Aware Topologies



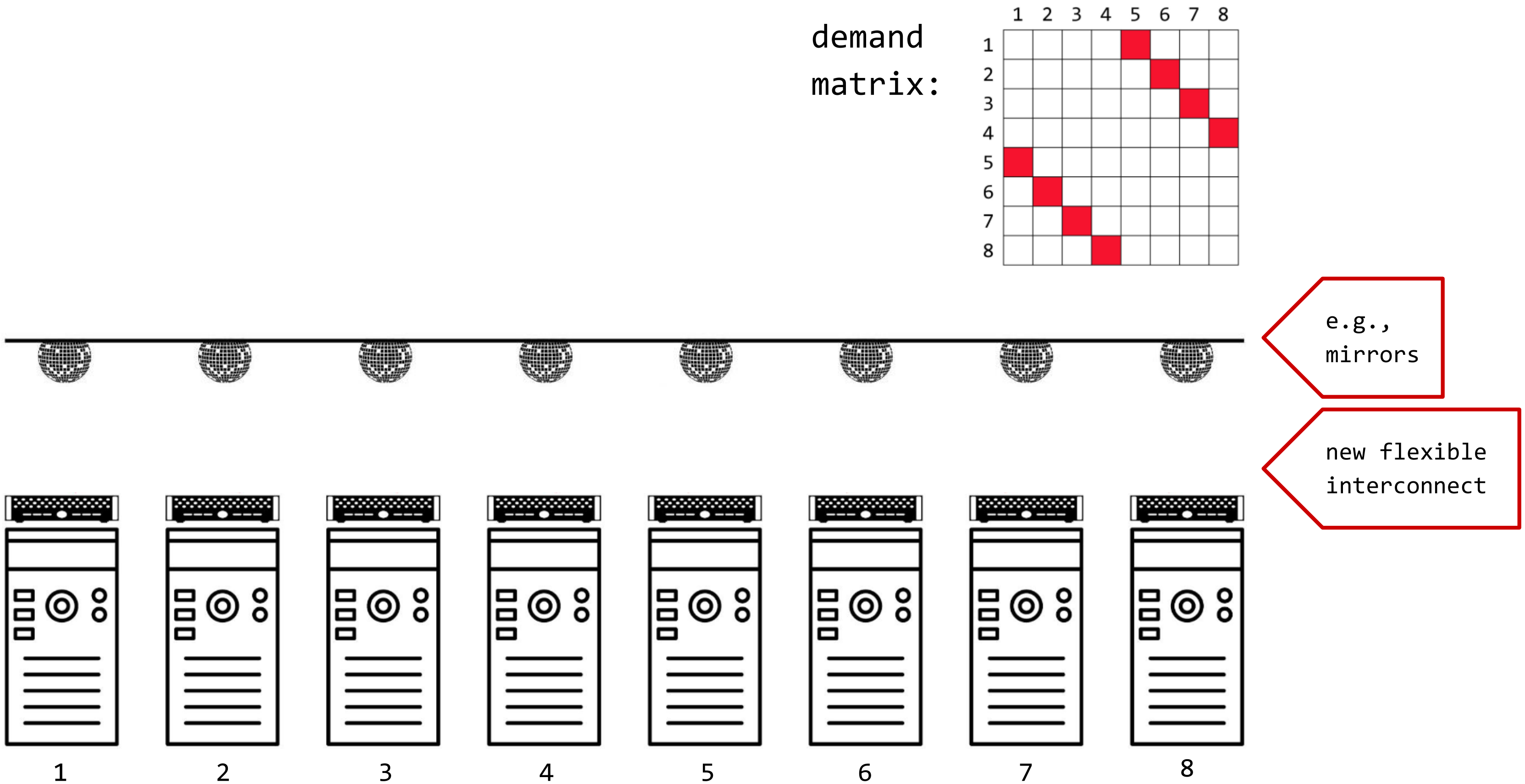
A Vision

Flexible and Demand-Aware Topologies



A Vision

Flexible and Demand-Aware Topologies

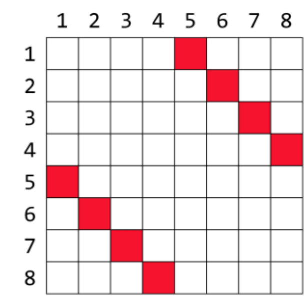


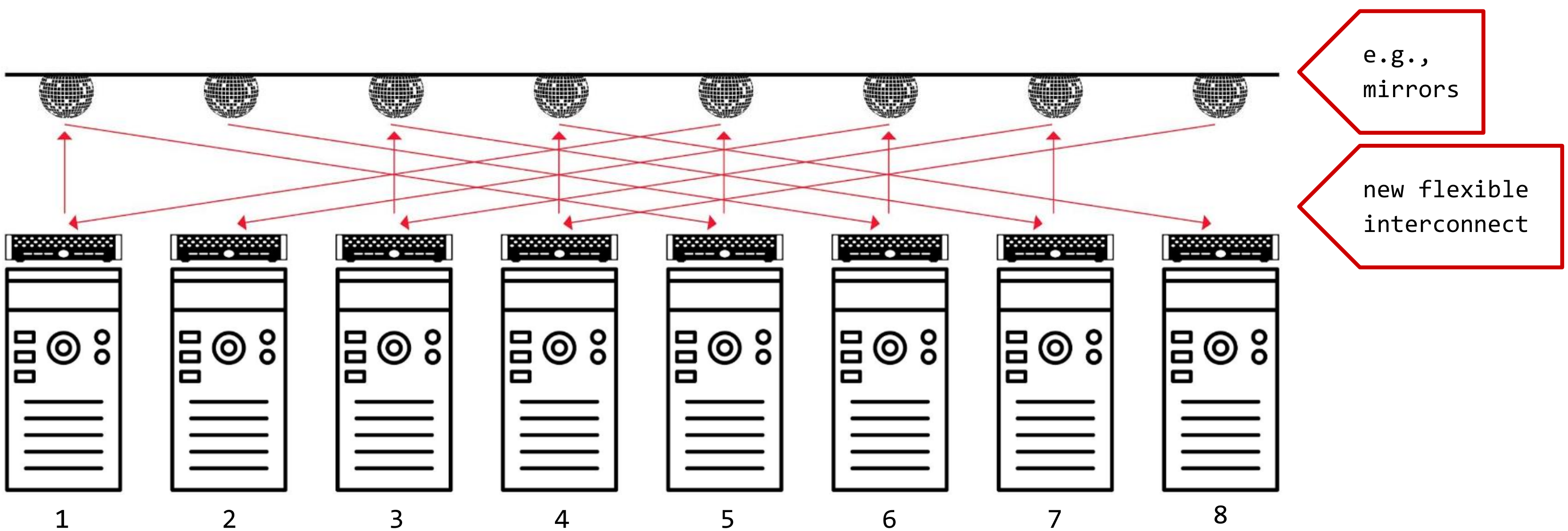
A Vision

Flexible and Demand-Aware Topologies

Matches demand

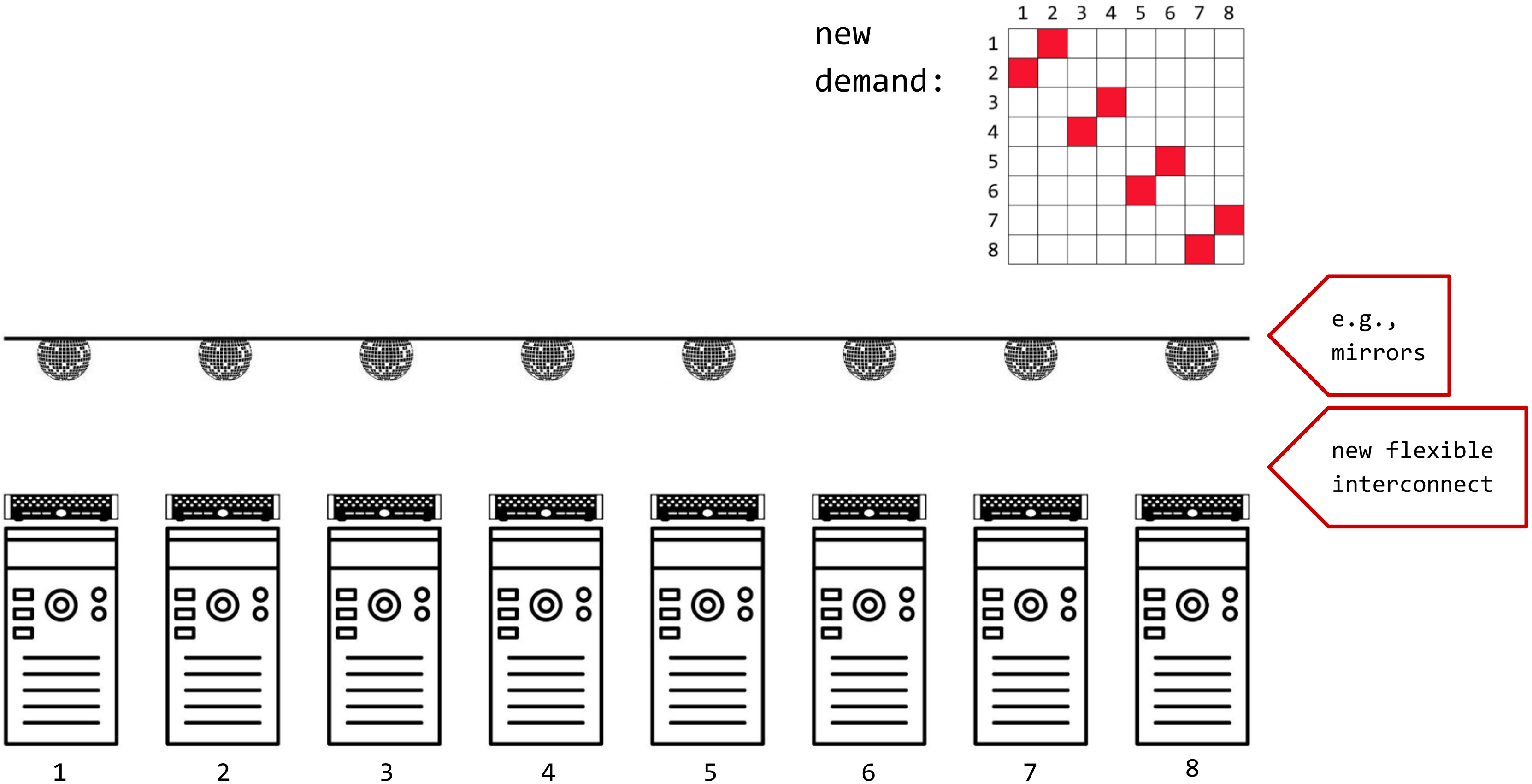
demand matrix:





A Vision

Flexible and Demand-Aware Topologies

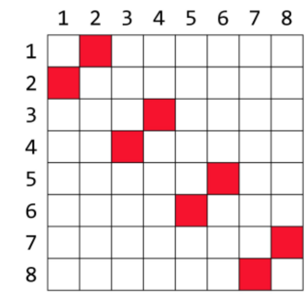


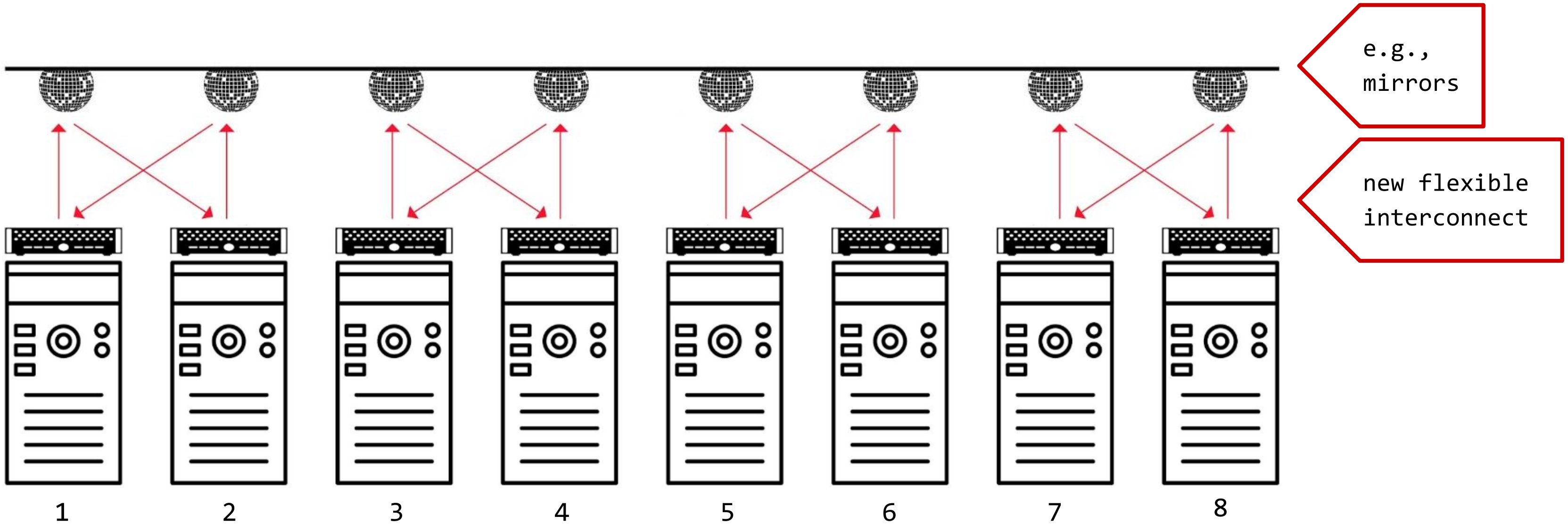
A Vision

Flexible and Demand-Aware Topologies

Matches demand

new demand:



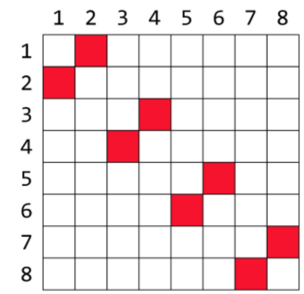


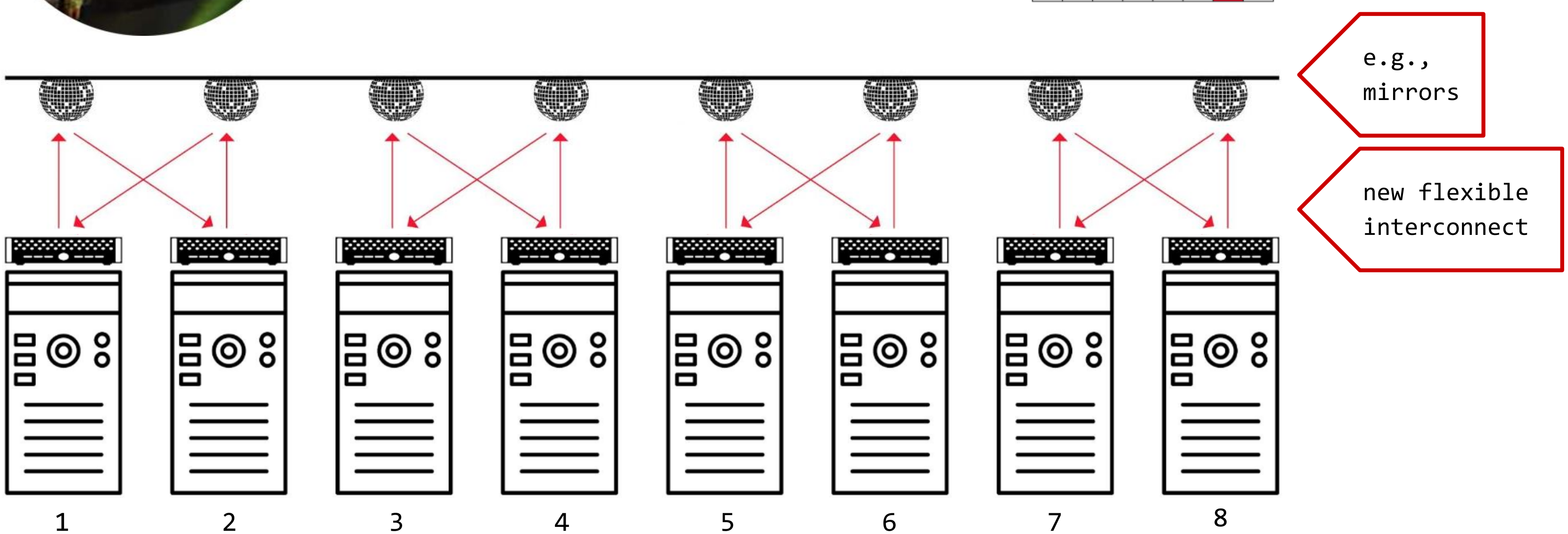
A Vision

Flexible and Demand-Aware Topologies

Self-Adjusting Networks

new demand:



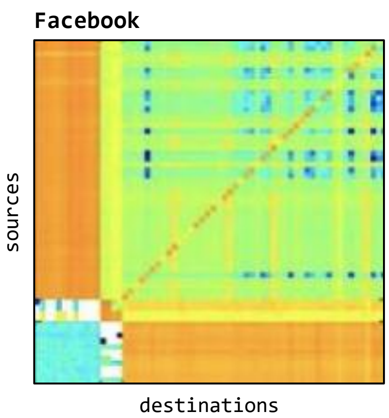


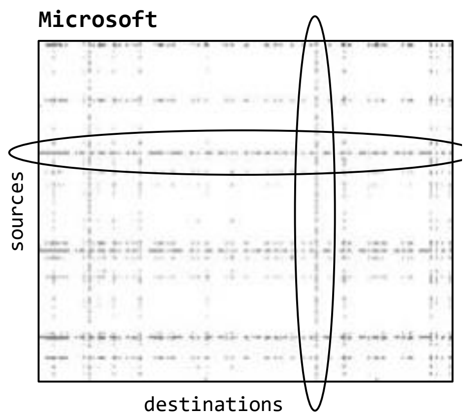
The Motivation

Much Structure in the Demand

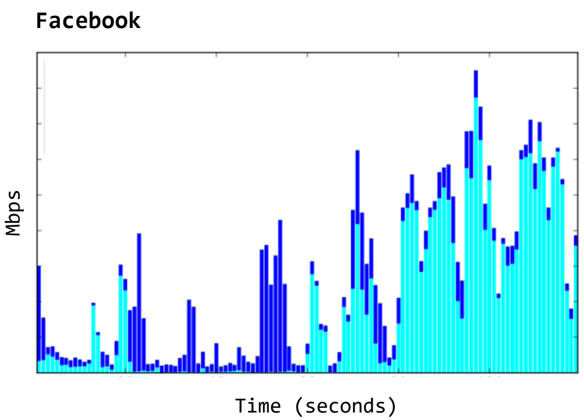
Empirical studies:

traffic matrices sparse and skewed





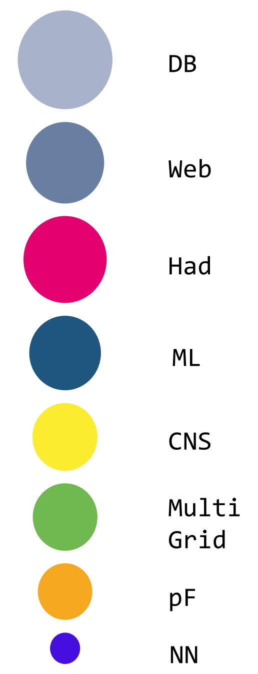
traffic bursty over time



The hypothesis: can be exploited.

Recent Representation of Trace Structure:

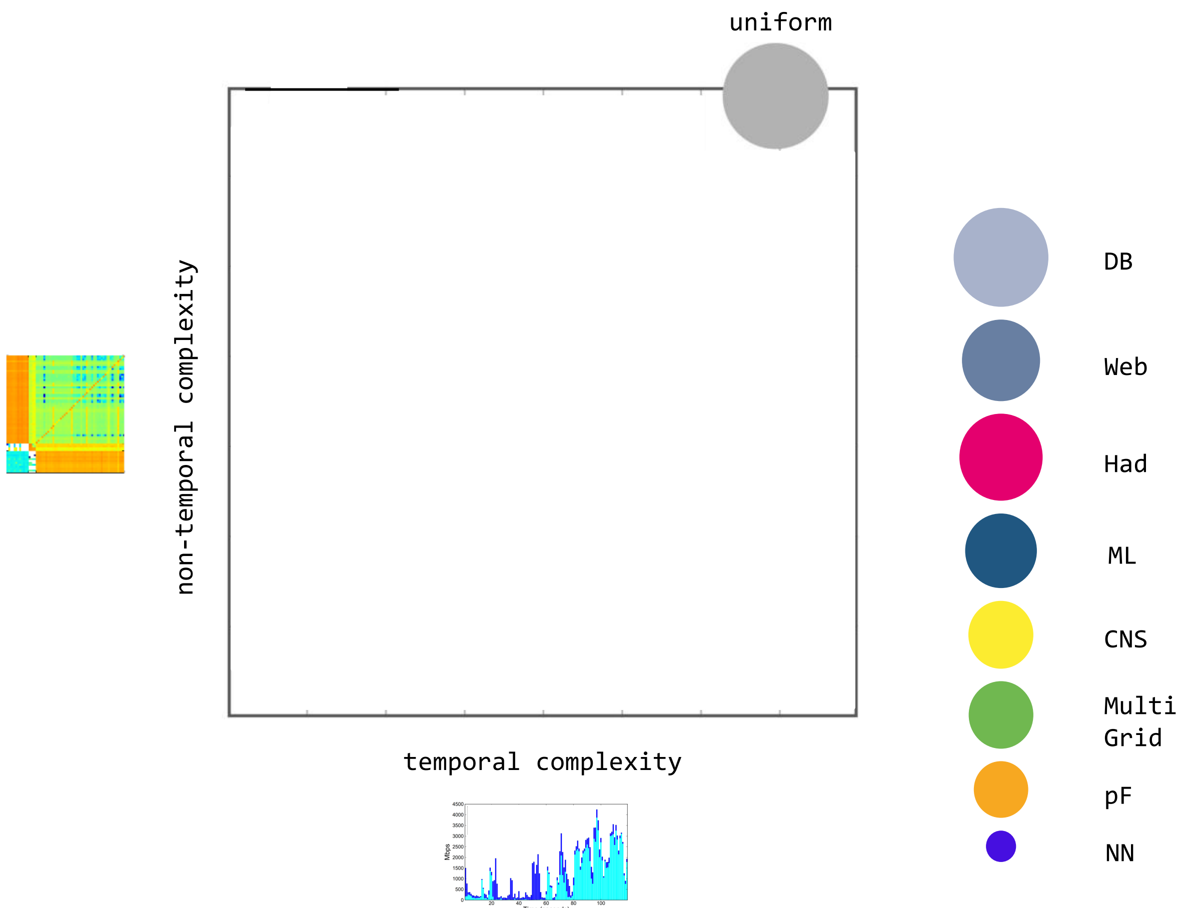
Complexity Map



Griner et al., SIGMETRICS 2020

Recent Representation of Trace Structure:

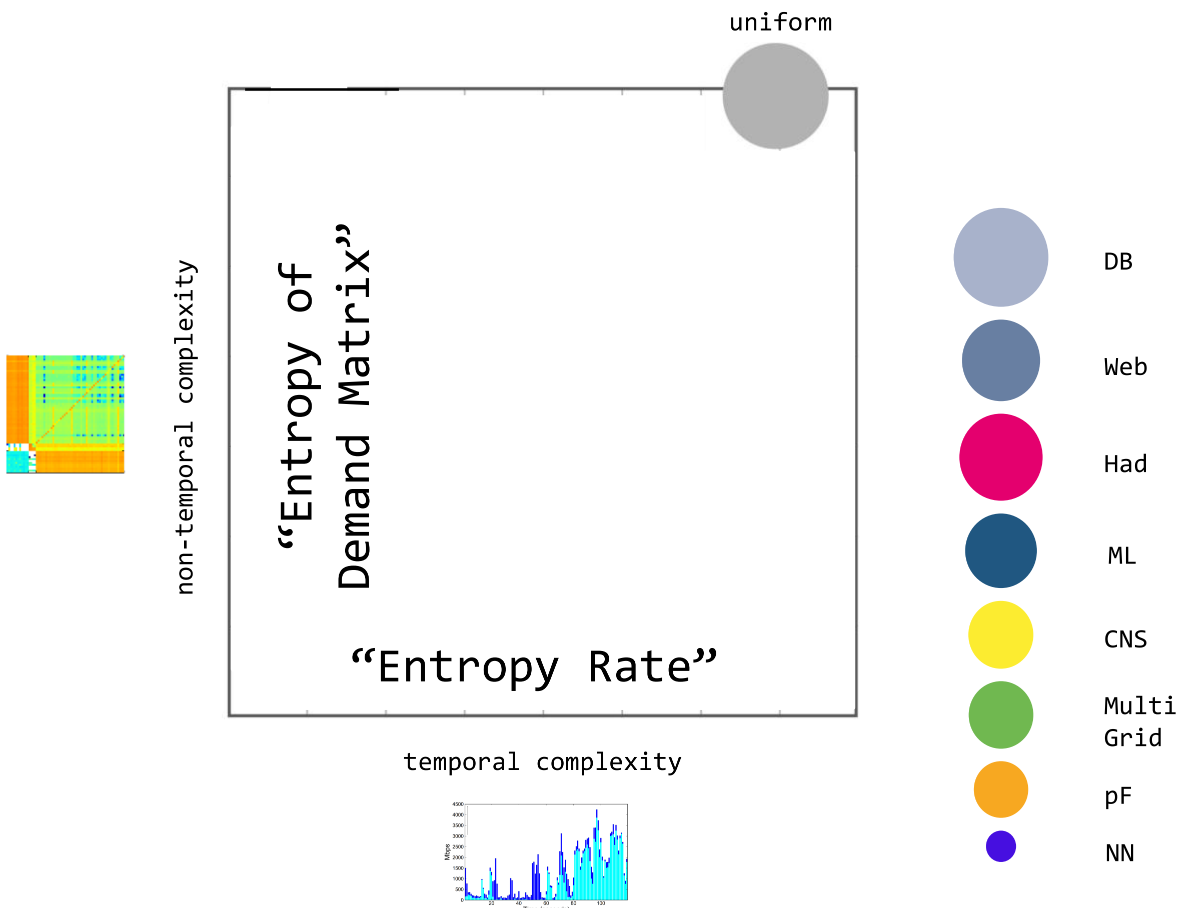
Complexity Map



Griner et al., SIGMETRICS 2020

Recent Representation of Trace Structure:

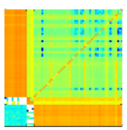
Complexity Map

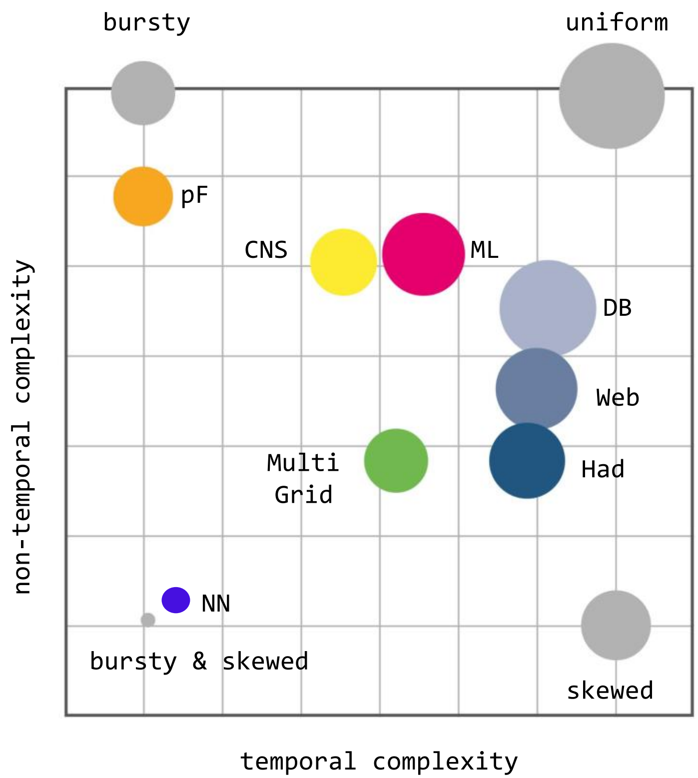


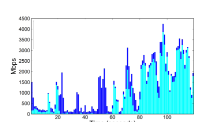
Griner et al., SIGMETRICS 2020

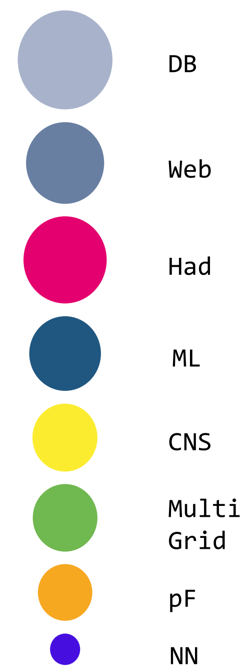
Recent Representation of Trace Structure:

Complexity Map







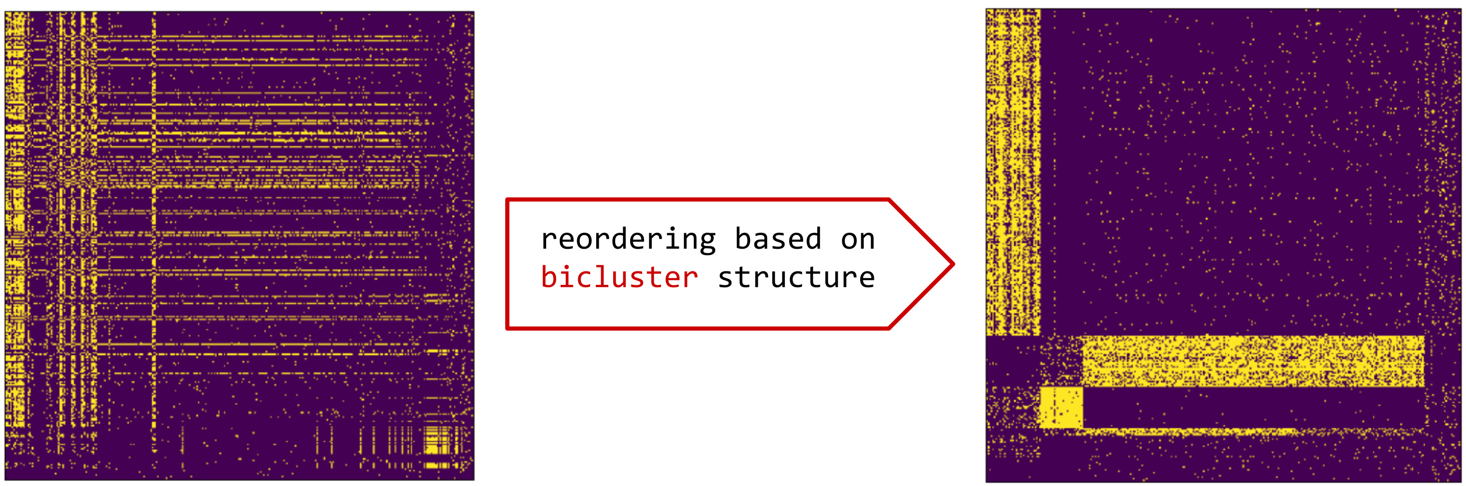


Different structures!

Griner et al., SIGMETRICS 2020

Traffic is also clustered:

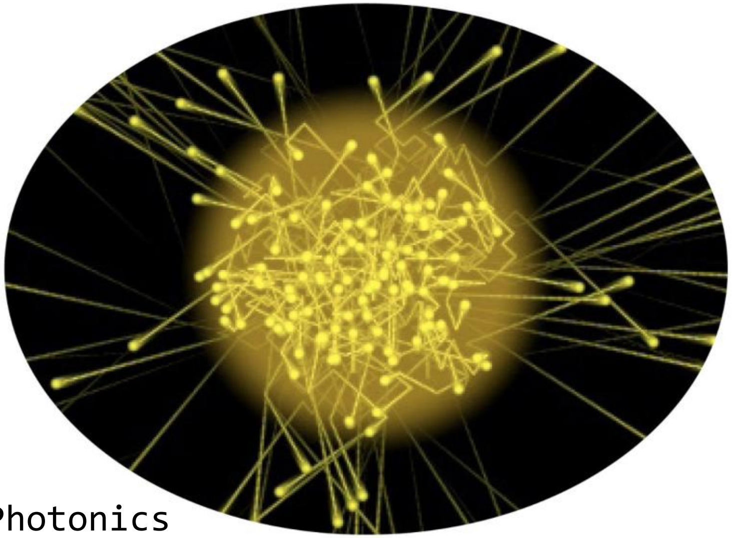
Small Stable Clusters



Opportunity: *exploit* with little reconfigurations!

Förster et al., Analyzing the Communication Clusters in Datacenters. WWW 2023

Sounds Crazy?
Emerging Enabling Technology.



Photonics

H2020:

“Photonics one of only five key enabling technologies for future prosperity.”

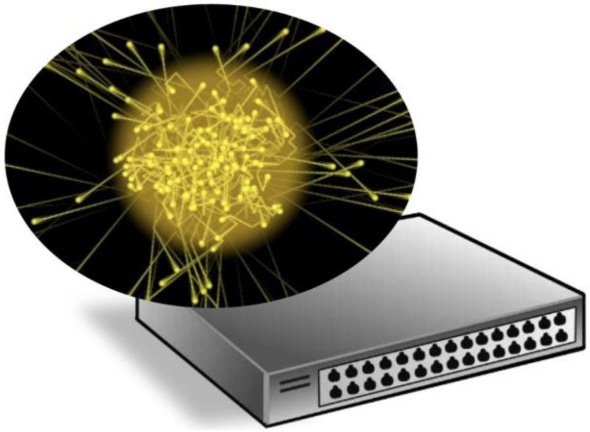
US National Research Council:

“Photons are the new Electrons.”

Enabler

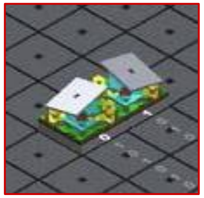
Novel Reconfigurable Optical Switches

- **Spectrum** of prototypes
  - Different sizes, different reconfiguration times
  - From our ACM **SIGCOMM** workshop OptSys



Prototype 1

Moving antenna (ms)



Prototype 2

Moving mirrors (mus)

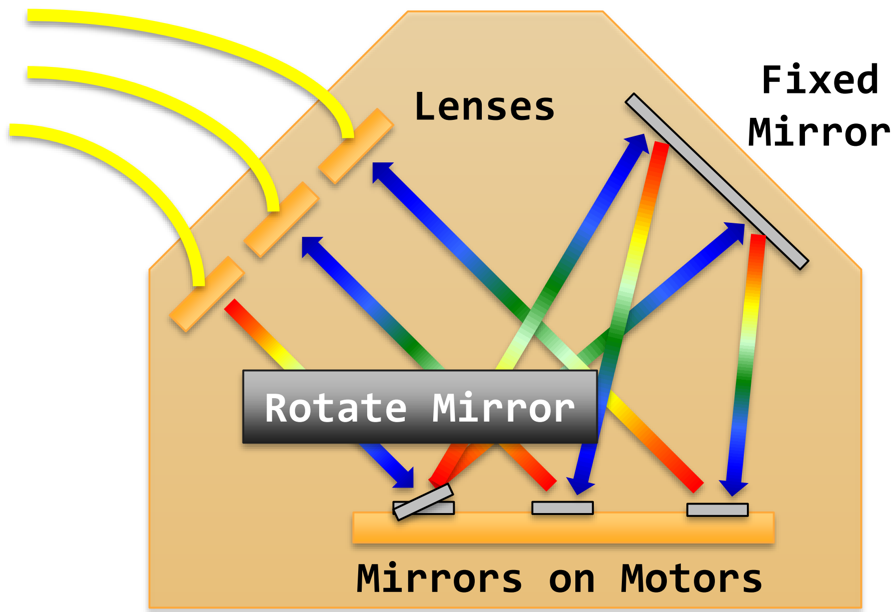
Prototype 3

Changing lambdas (ns)

Example

Optical Circuit Switch

- ⇢ Optical Circuit Switch rapid adaption of physical layer
  - ⇀ Based on rotating mirrors



Optical Circuit Switch

By Nathan Farrington, SIGCOMM 2010

First Deployments

E.g., Google


Systems

Jupiter evolving: Reflecting on Google's data center network transformation

August 24, 2022

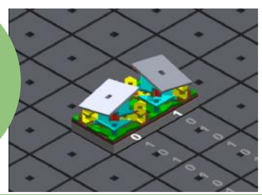
Amin Vahdat

VP & GM, Systems and Services Infrastructure

The Big Picture

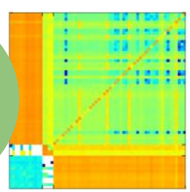
New!

Flexibility



More!

Structure

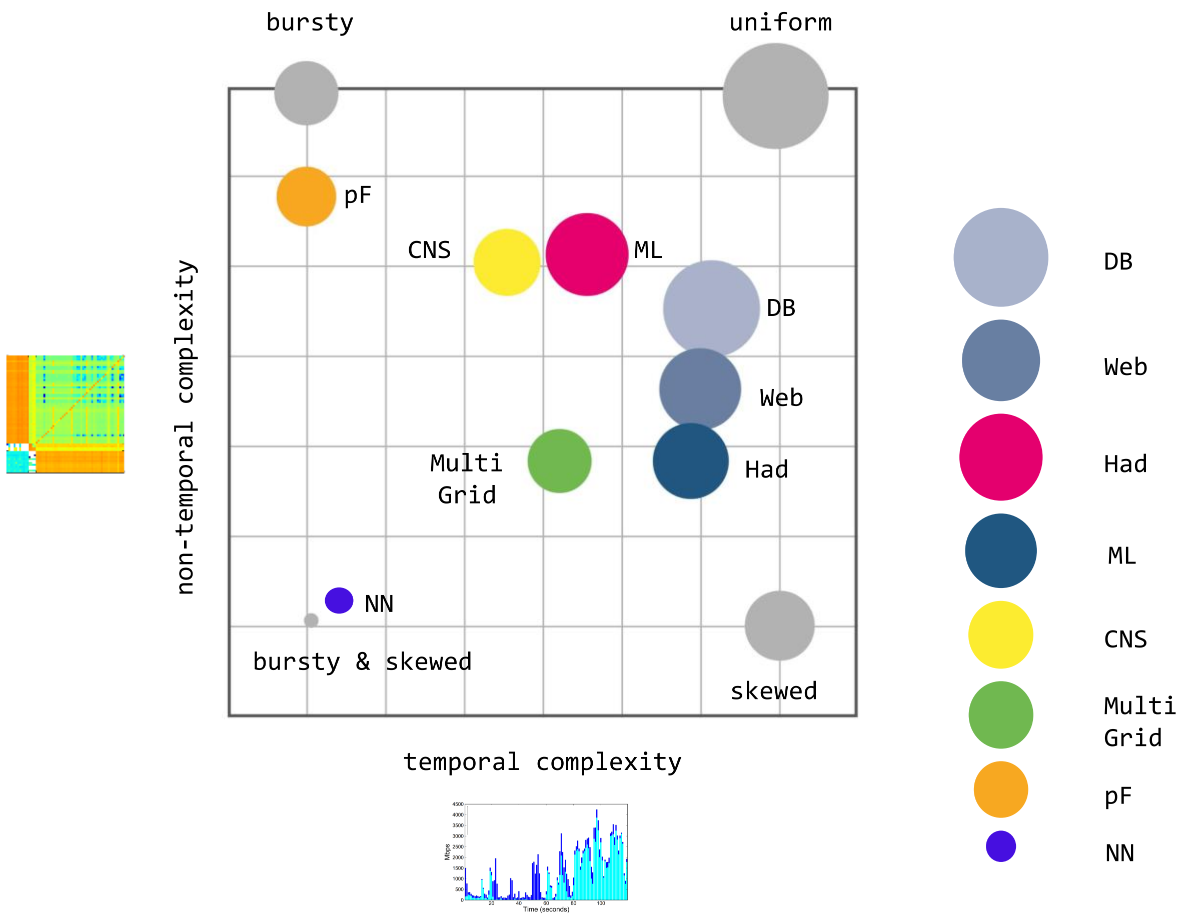


Self-Adjusting Networks

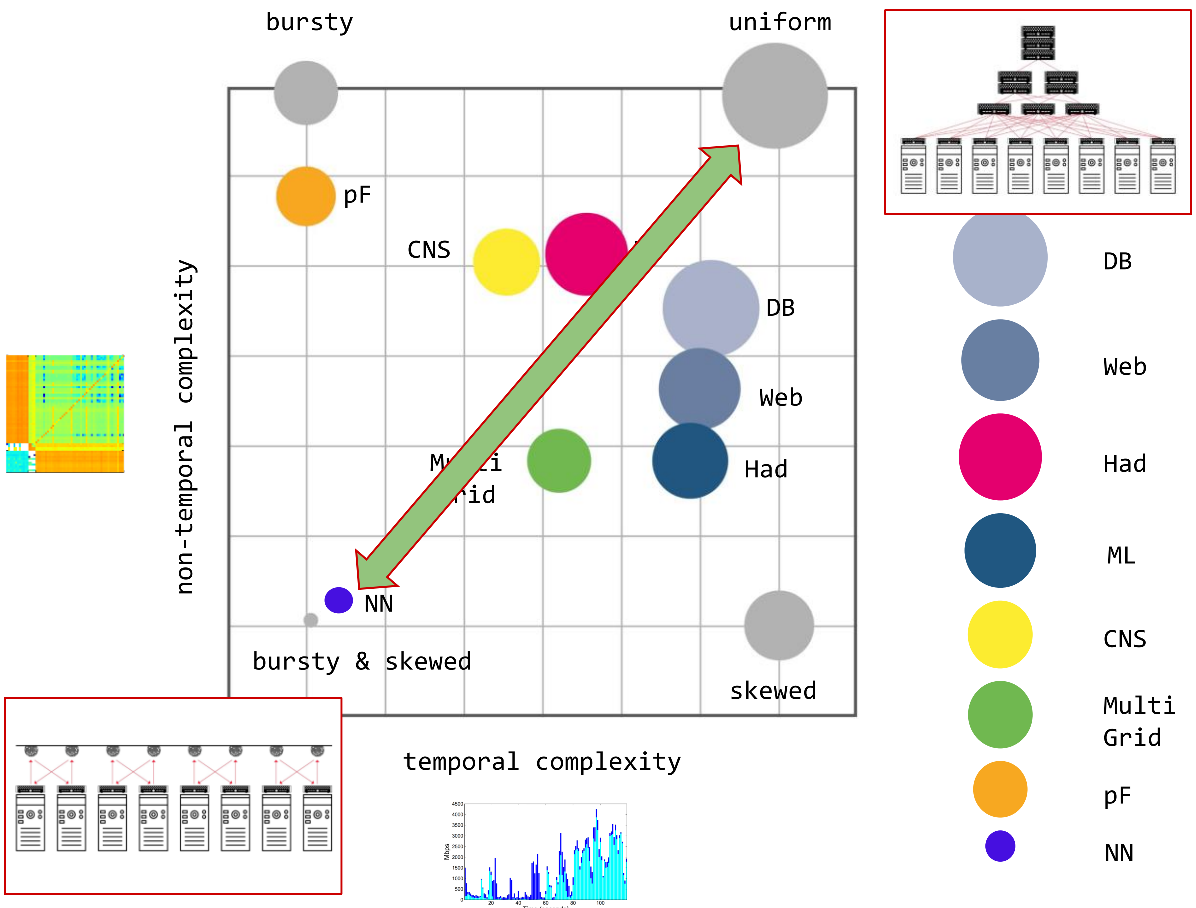
Efficiency

Now is the time!

Potential Gain



Potential Gain



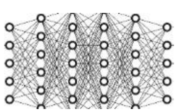
Unique Position

Demand-Aware, Self-Adjusting Systems

Everywhere, but mainly
in software

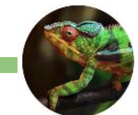
Algorithmic trading

Recommender systems

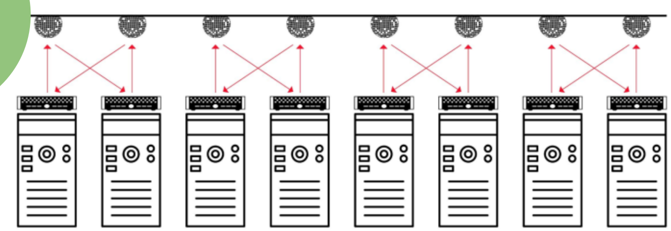


Neural networks

VS



Our focus in this talk:
in hardware



Tech Diversity

Design Spectrum of (R)DCNs

Diverse topology components:

⇀ demand-oblivious and demand-aware

Demand-oblivious

Demand-aware

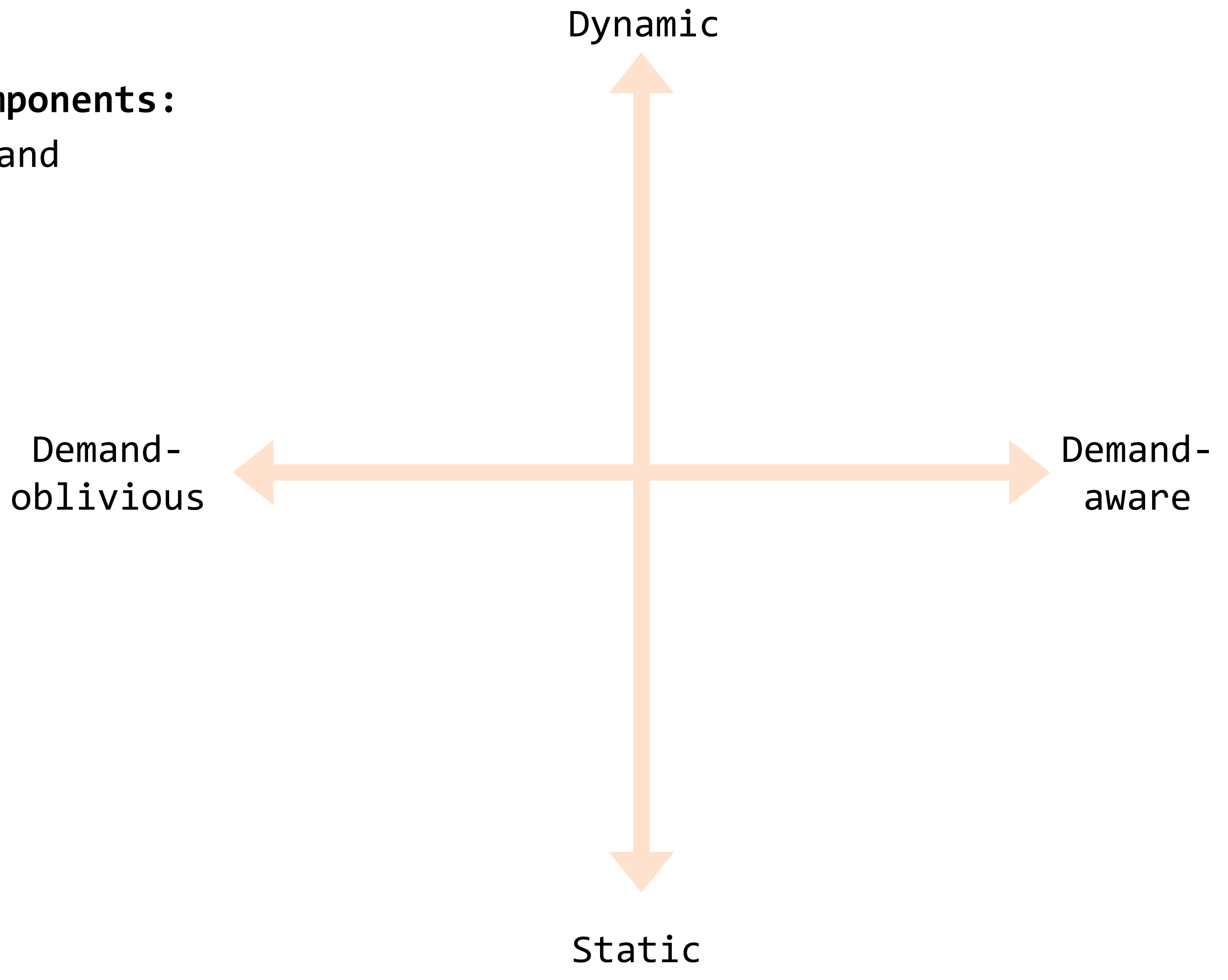
Tech Diversity

Design Spectrum of (R)DCNs

Diverse topology components:

⇀ demand-oblivious and
demand-aware

⇀ static vs dynamic



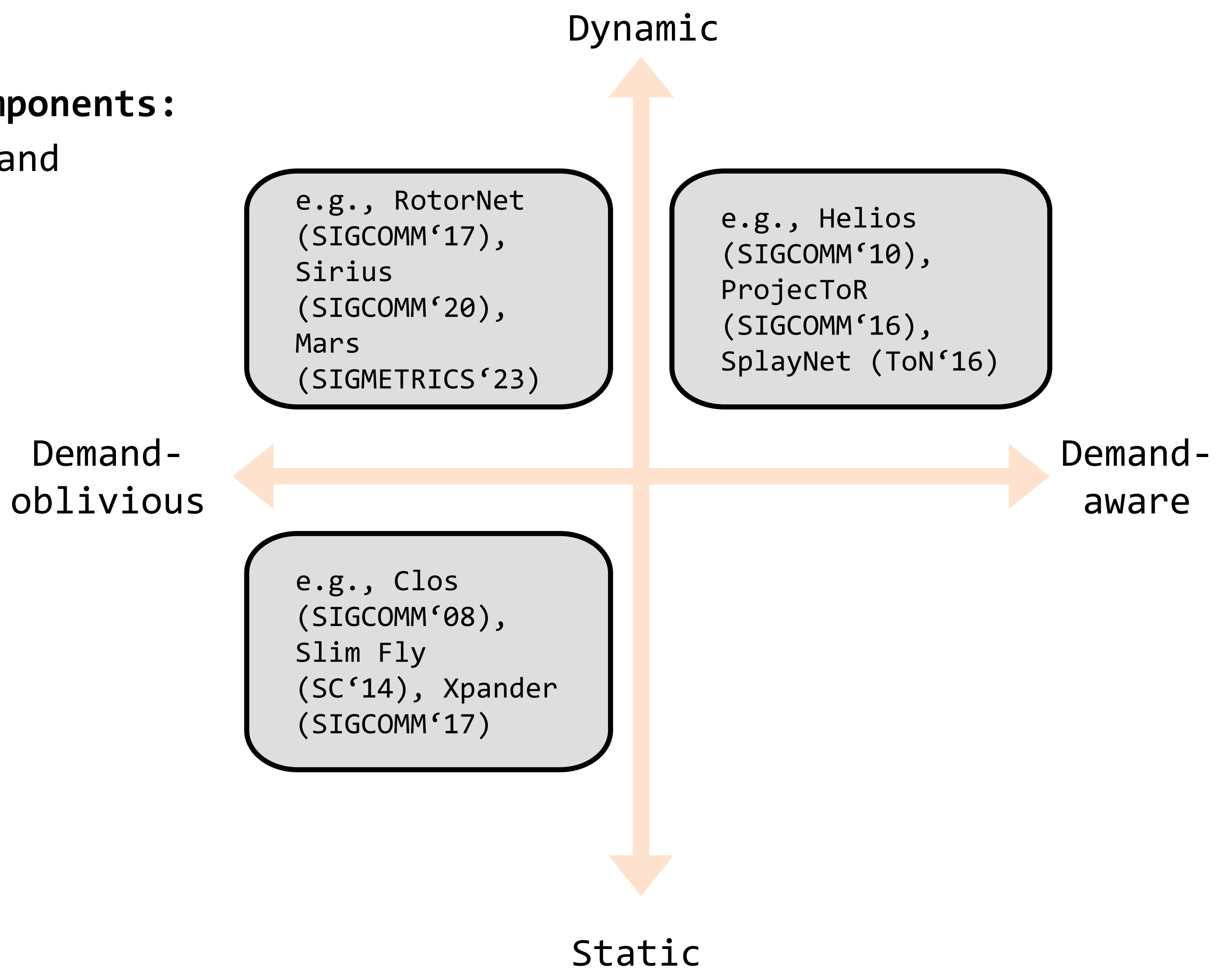
Tech Diversity

Design Spectrum of (R)DCNs

Diverse topology components:

⇀ demand-oblivious and demand-aware

⇀ static vs dynamic



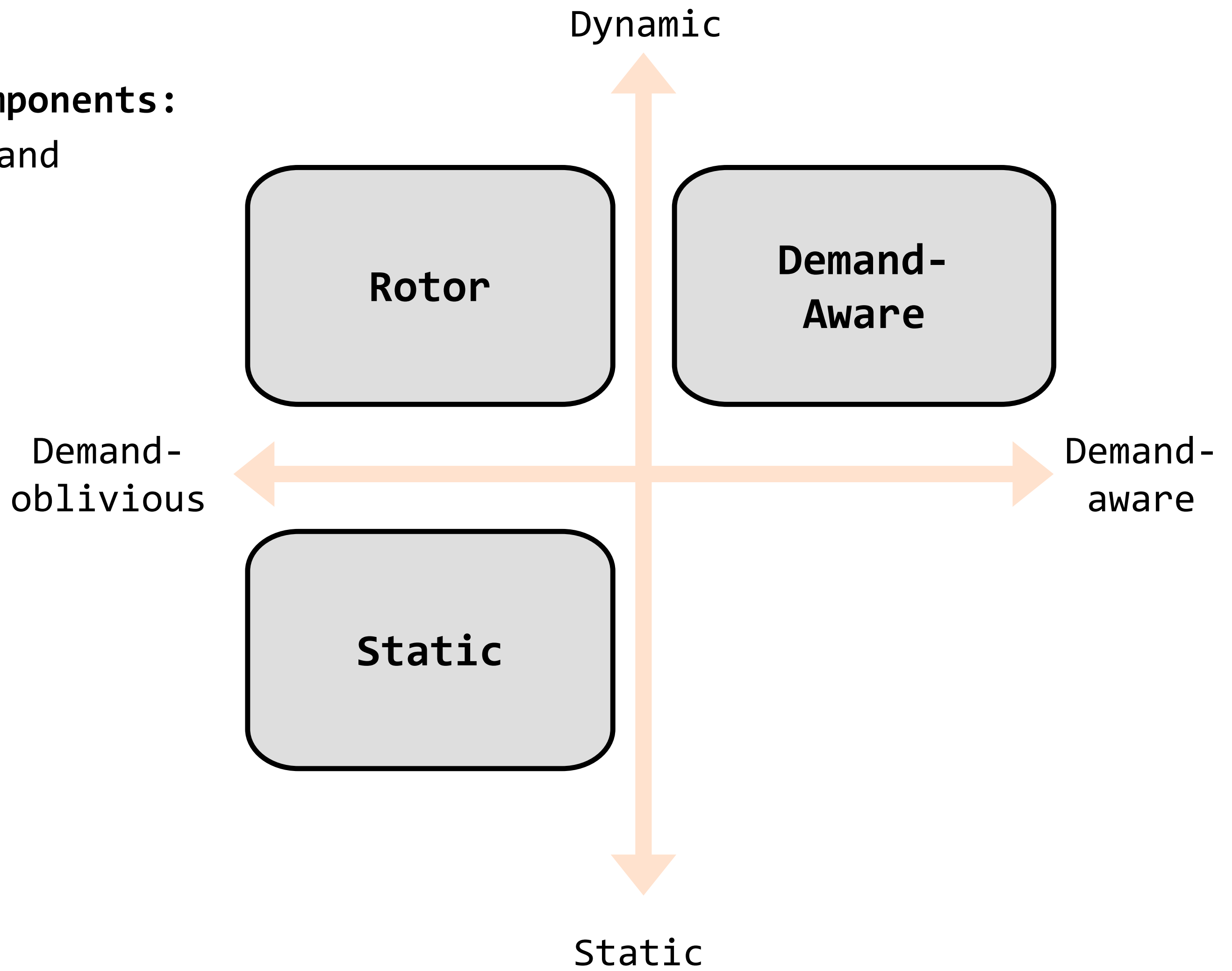
Tech Diversity

Design Spectrum of (R)DCNs

Diverse topology components:

⇀ demand-oblivious and
  demand-aware

⇀ static vs dynamic



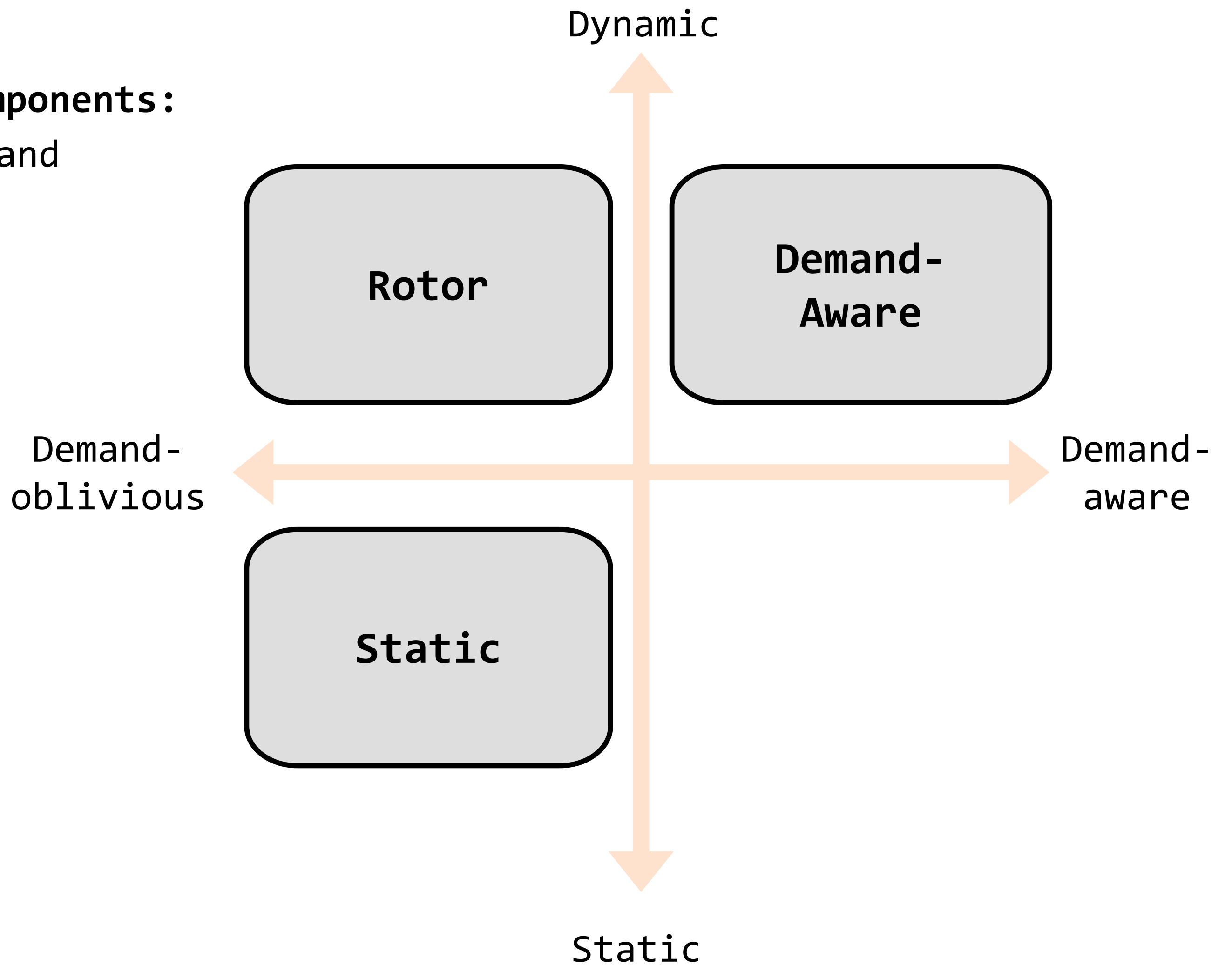
Tech Diversity

Design Spectrum of (R)DCNs

Diverse topology components:

⇀ demand-oblivious and demand-aware

⇀ static vs dynamic



Which approach is best?

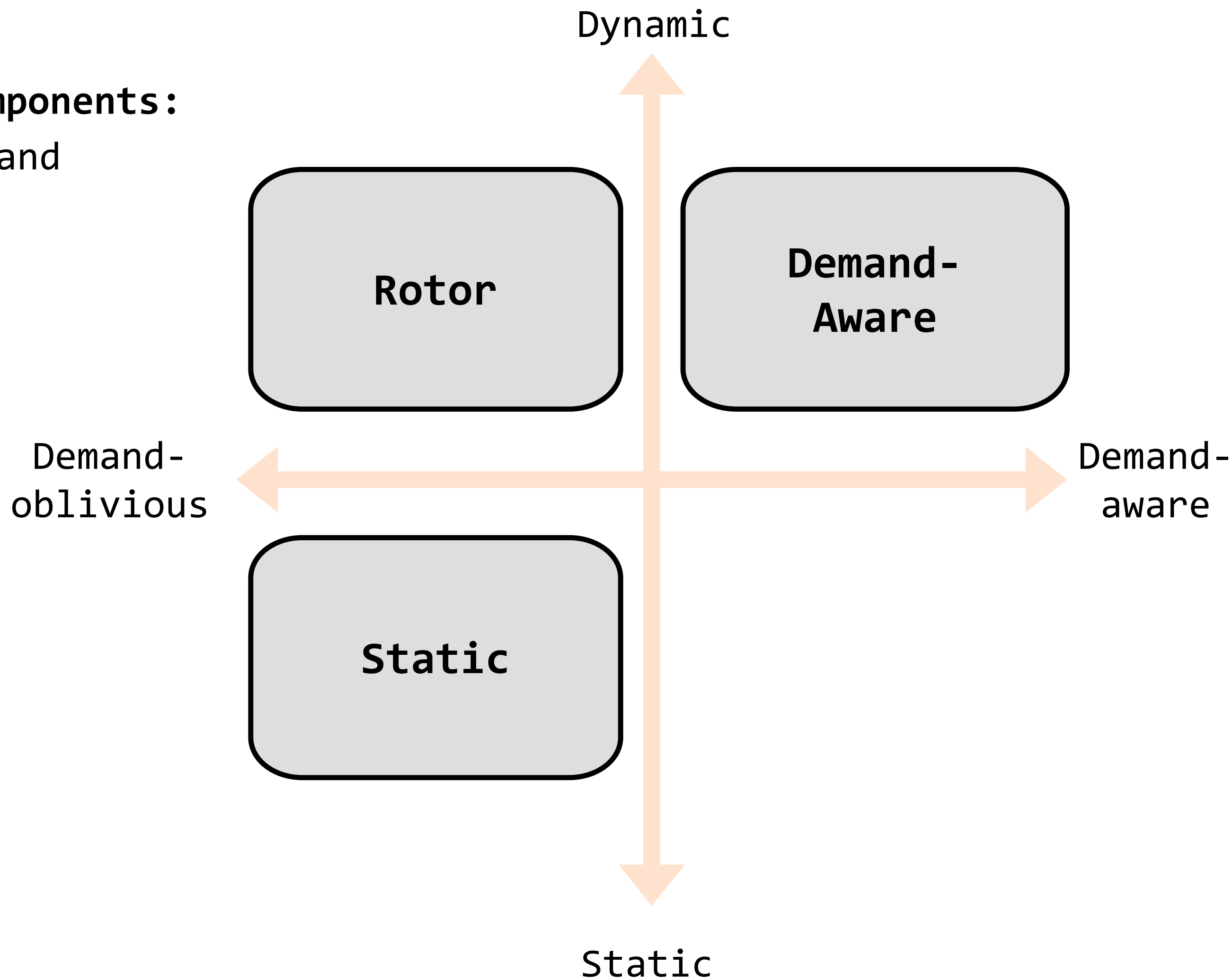
Tech Diversity

Design Spectrum of (R)DCNs

Diverse topology components:

⇀ demand-oblivious and demand-aware

⇀ static vs dynamic



Which approach is best?

As always in CS: It depends…

Depends on: Traffic

Diverse patterns:

⇀ Shuffling/Hadoop:
all-to-all

⇀ All-reduce/ML: ring or
tree traffic patterns

⇀ Elephant flows

⇀ Query traffic: skewed

⇀ Mice flows

⇀ Control traffic: does not evolve
but has non-temporal structure

Diverse requirements:

⇀ ML is bandwidth hungry,
small flows are latency-
sensitive

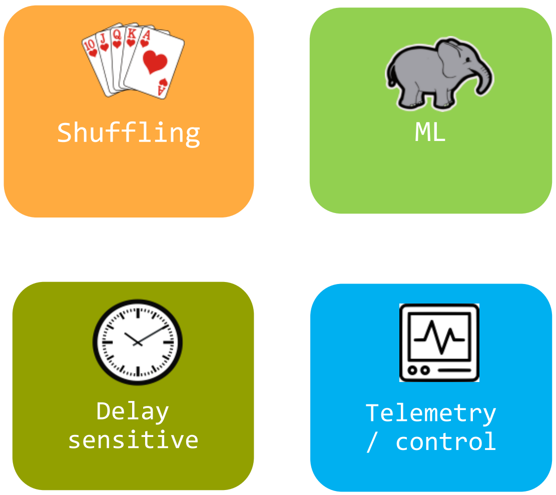
Shuffling
All-to-All

ML
Large flows

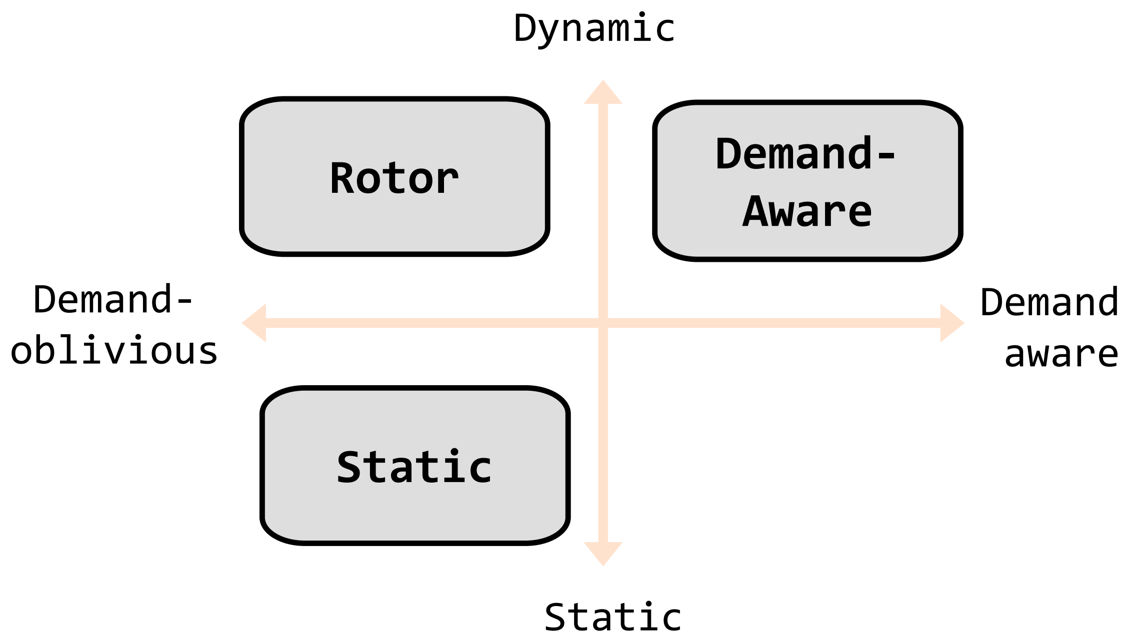
Delay
sensitive

Telemetry
/ control

Examples:
Match or Mismatch?

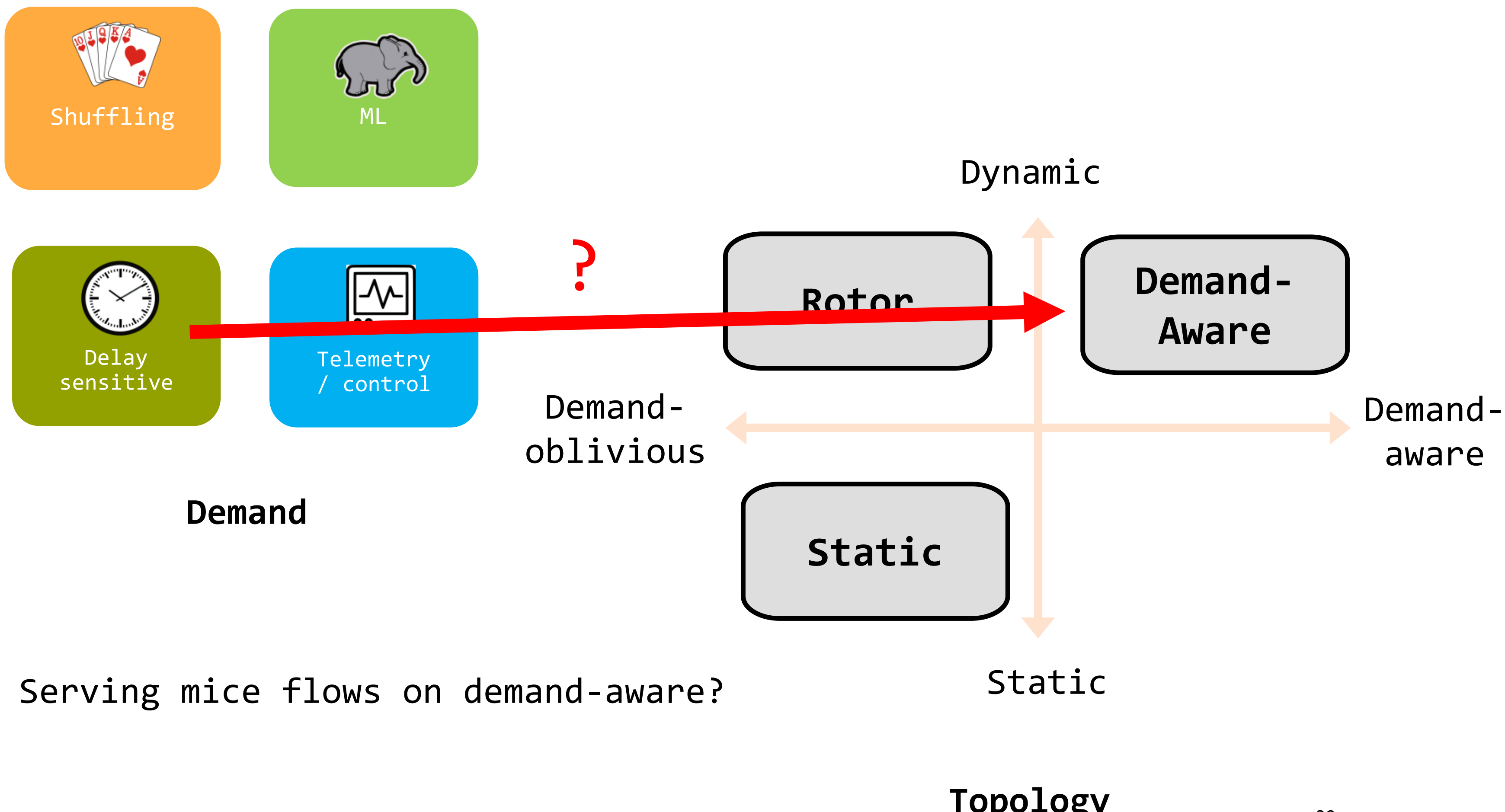


Demand



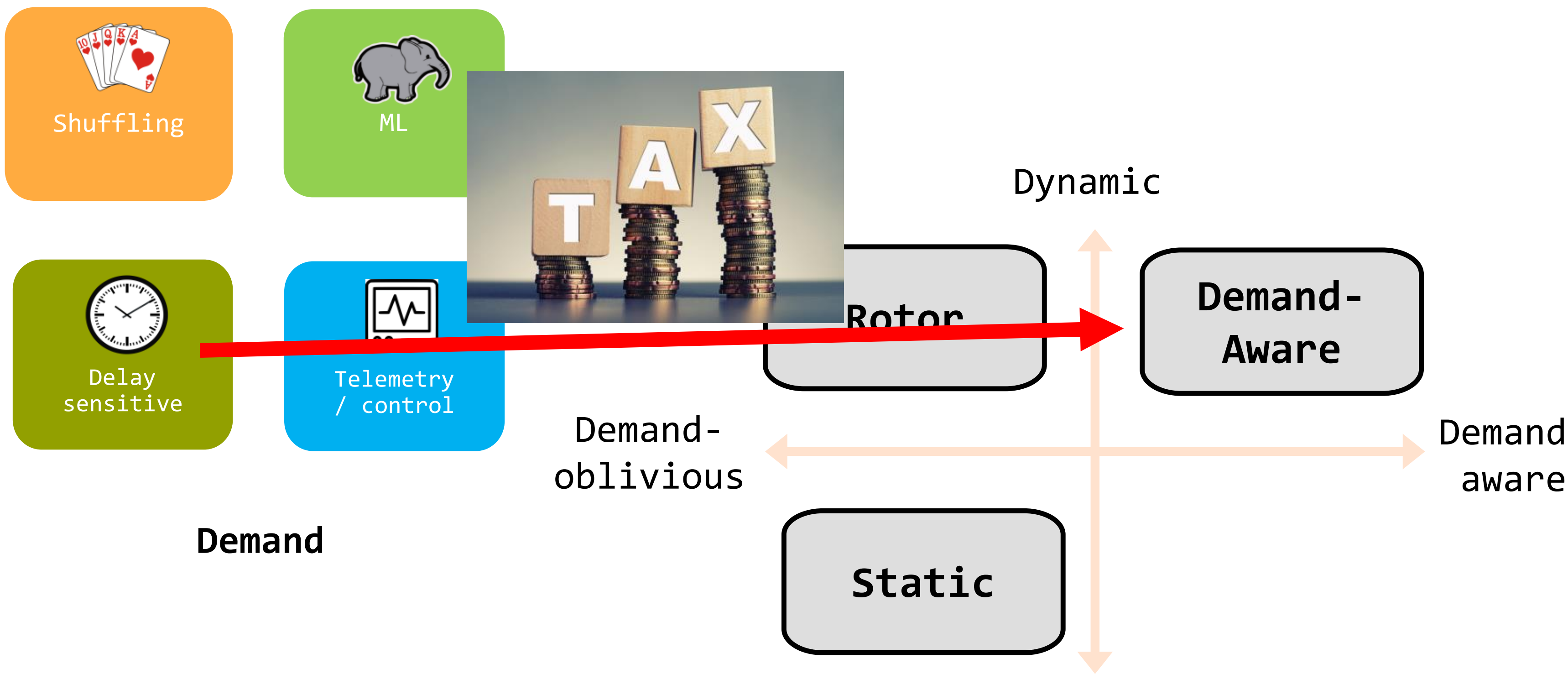
Topology

Examples:
Match or Mismatch?



Serving mice flows on demand-aware?

Examples:
Match or Mismatch?



Demand

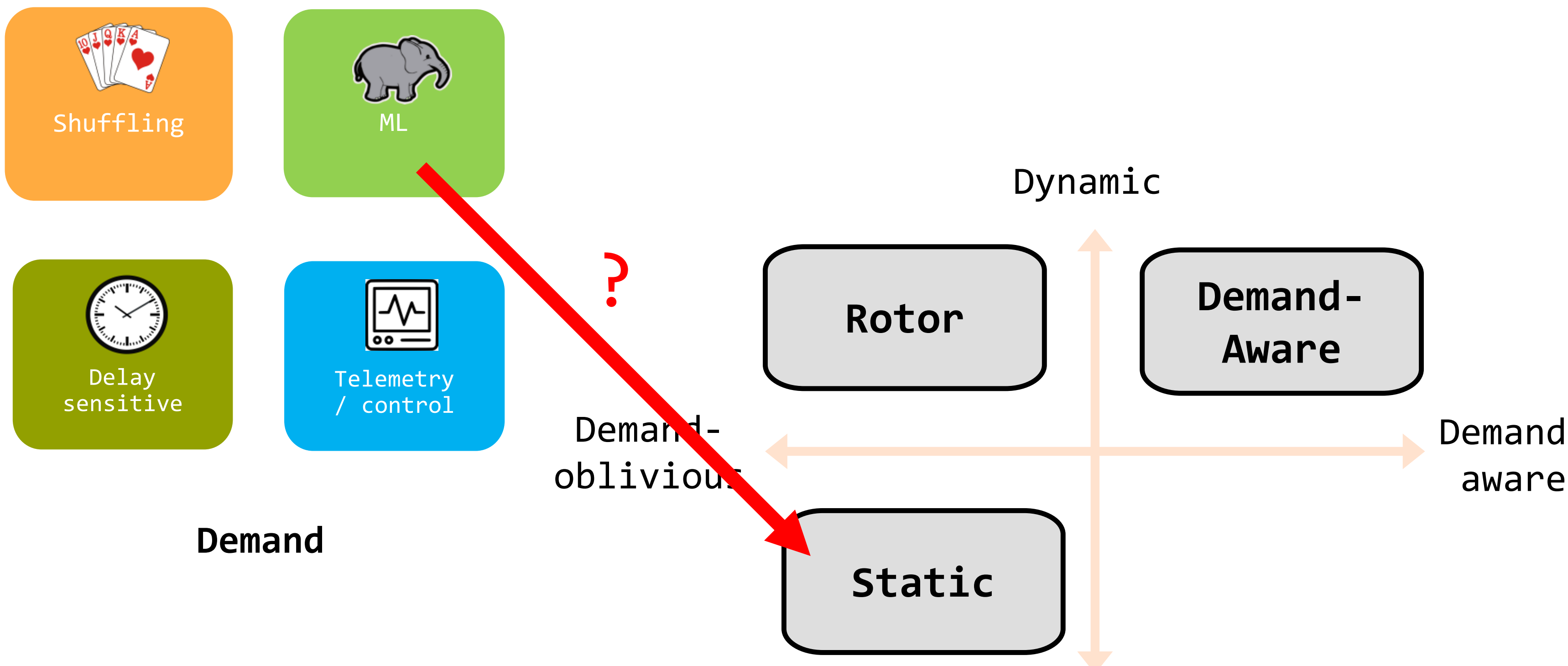
Serving mice flows on demand-aware?

Bad idea! Latency tax.

Static

Topology

Examples:
Match or Mismatch?

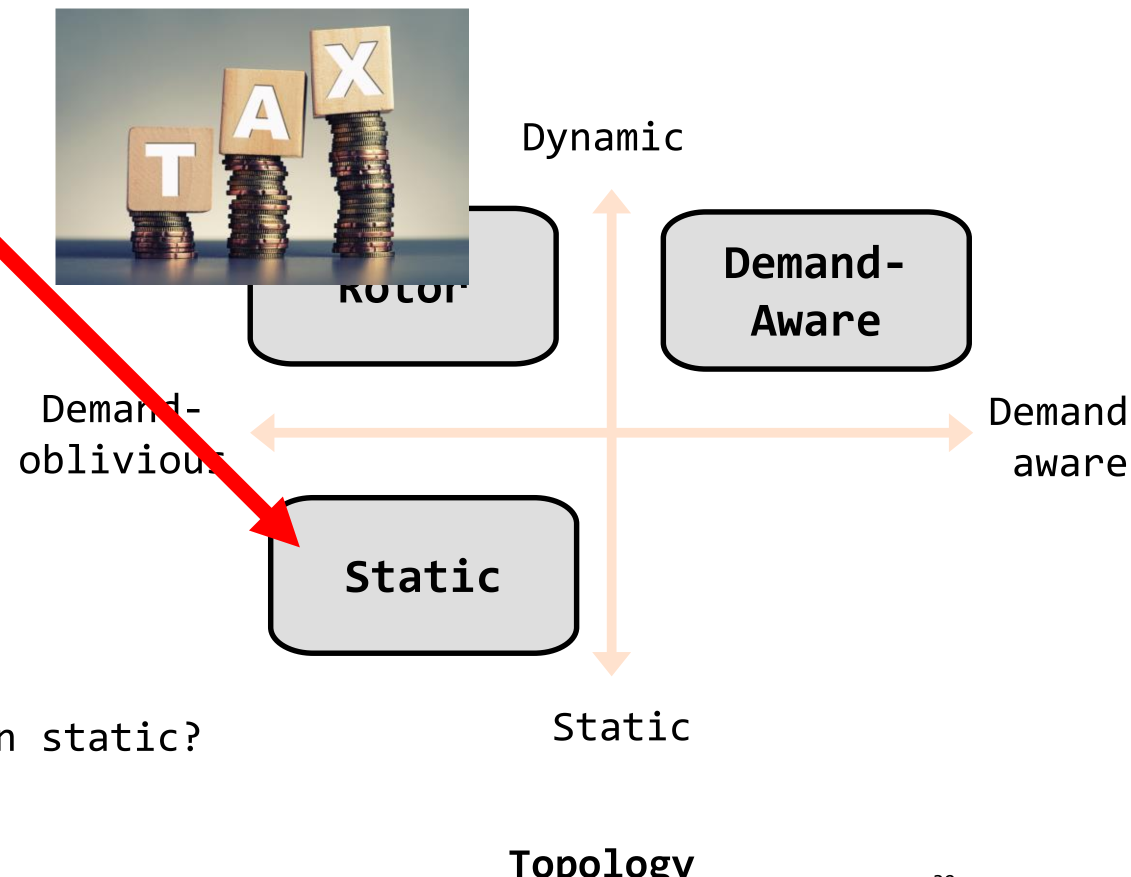


Serving elephant flows on static?

Static

Topology

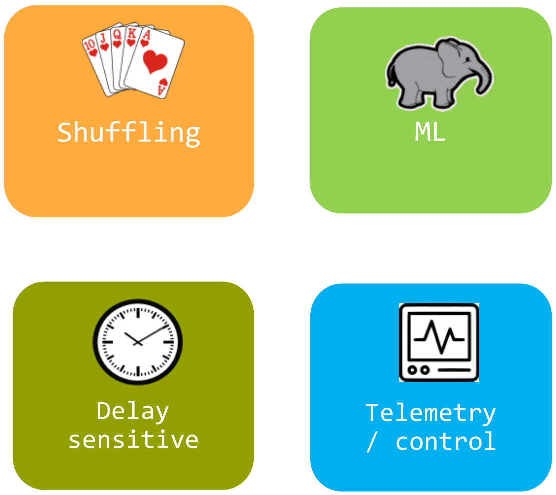
Examples: Match or Mismatch?



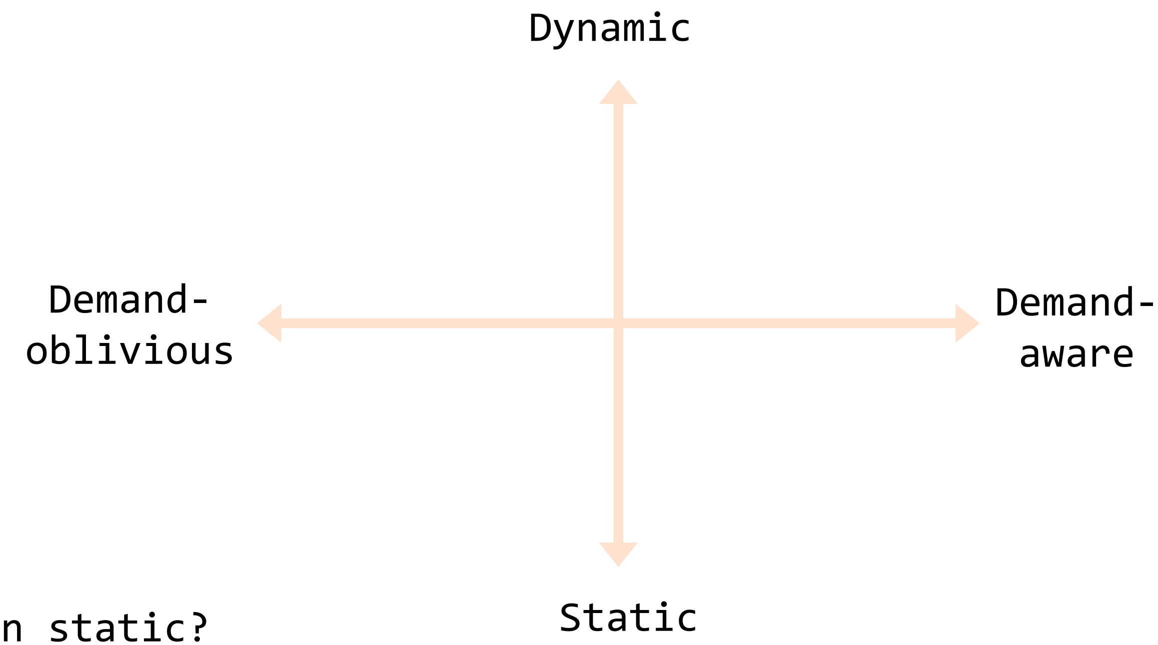
Serving elephant flows on static?

Bad idea! Bandwidth tax.

Examples:
Match or Mismatch?



Demand

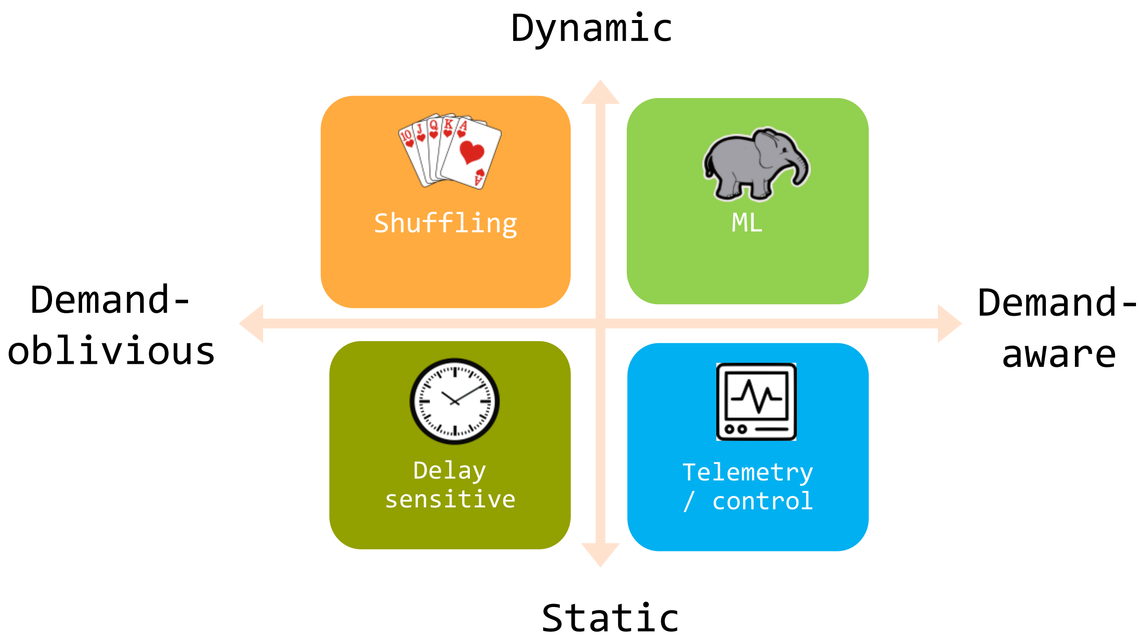


Topology

Serving elephant flows on static?

Bad idea! Bandwidth tax.

Optimal Solution: It's a Match!



We have a first approach:

Cerberus*** serves traffic on the “best topology”! (Optimality open)

* Griner et al., ACM SIGMETRICS 2022

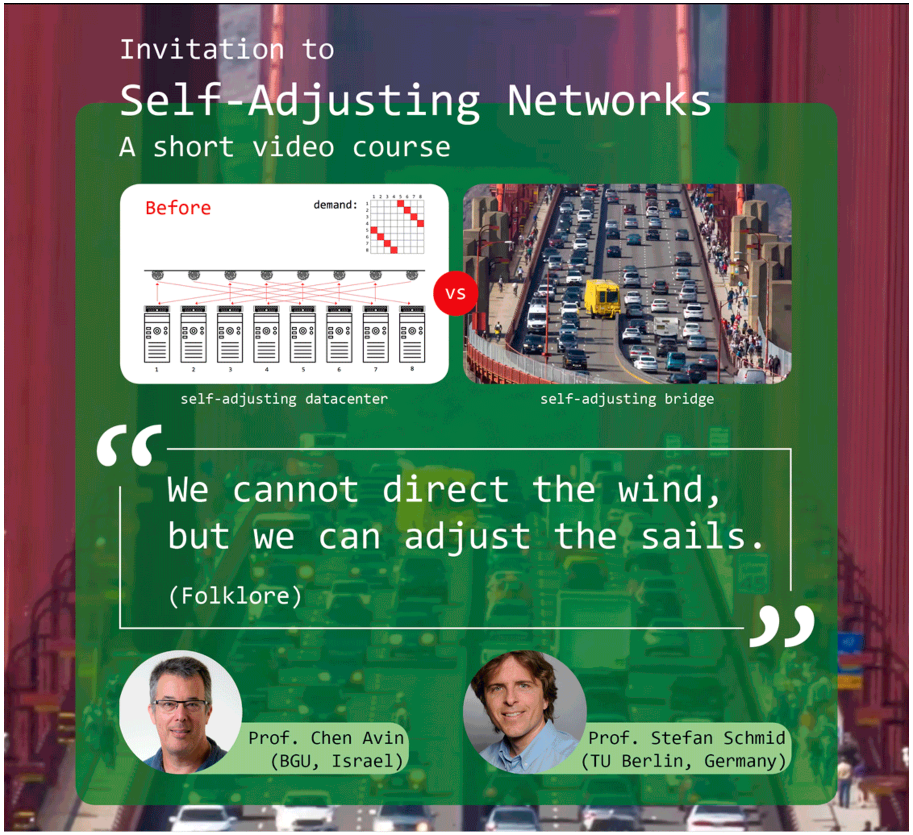
“Zukunftsmusik”

⇢ So far: tip of the iceberg

⇢ Many more challenges
- ⇀ Shock wave through *layers*:
  impact on routing and congestion control?
- ⇀ *Scalability* of control in dynamic graphs:
  local algorithms? Greedy routing?
- ⇢ Complexity of demand-aware graphs
  (*pure vs hybrid*, e.g., SplayNet)
- ⇀ *Application-specific* self-adjusting networks:
  e.g., for AI, or similar to *active dynamic networks* (independent sets, consensus, …)
- ⇀ etc.

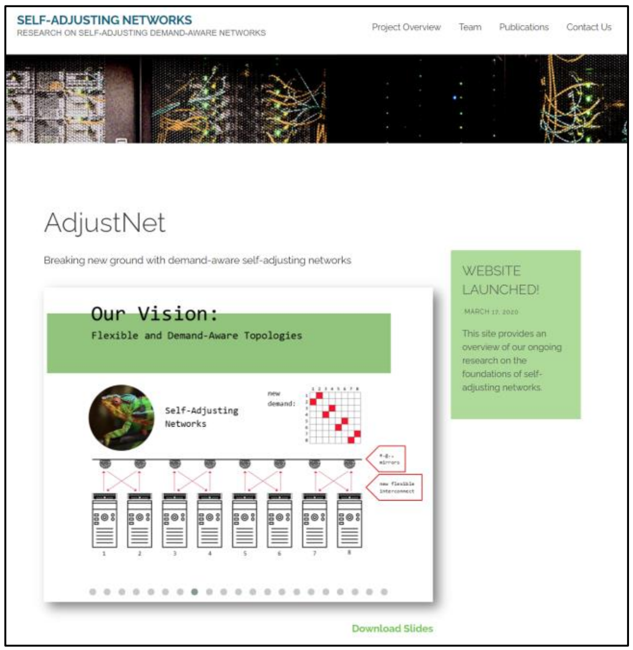
Thank you!

Online Video Course

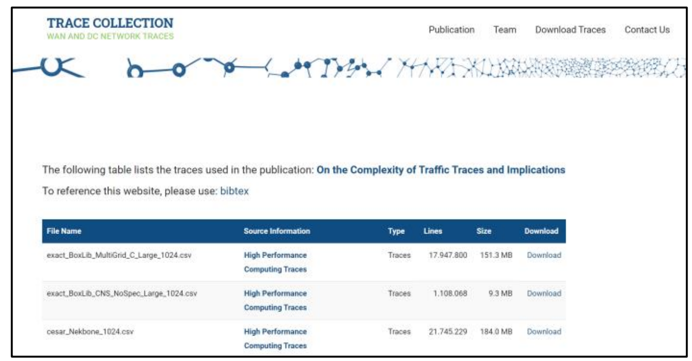


https://self-adjusting.net/course

Websites

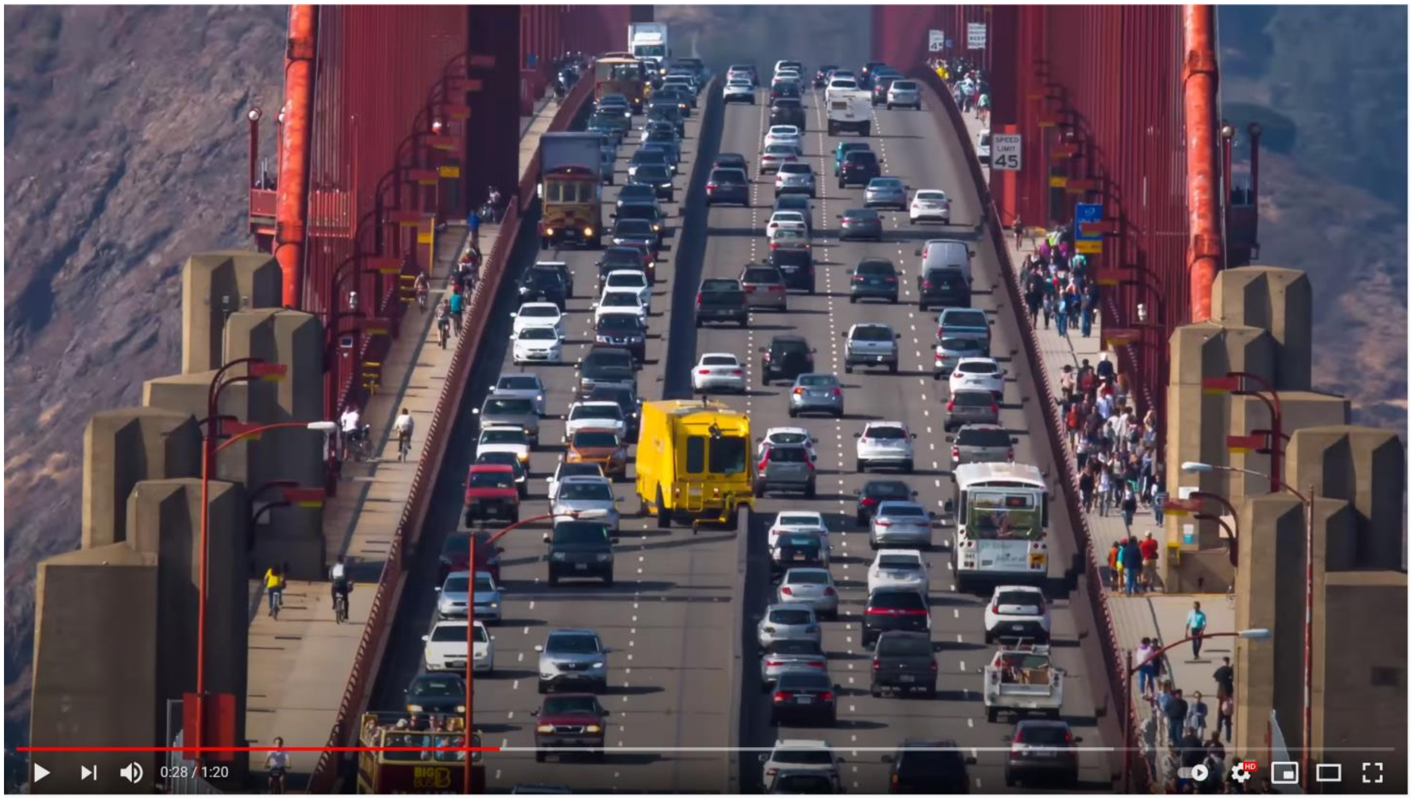


http://self-adjusting.net/
Project website



https://trace-collection.net/
Trace collection website

Questions?



Golden Gate Zipper

Selected References

On the Complexity of Traffic Traces and Implications
Chen Avin, Manya Ghobadi, Chen Griner, and Stefan Schmid.
ACM SIGMETRICS, Boston, Massachusetts, USA, June 2020.

Survey of Reconfigurable Data Center Networks: Enablers, Algorithms, Complexity
Klaus-Tycho Foerster and Stefan Schmid.
SIGACT News, June 2019.

Analyzing the Communication Clusters in Datacenters
Klaus-Tycho Foerster, Thibault Marette, Stefan Neumann, Claudia Plant, Ylli Sadikaj, Stefan Schmid, and Yllka Velaj.
The Web Conference (WWW), Austin, Texas, USA, April 2023.

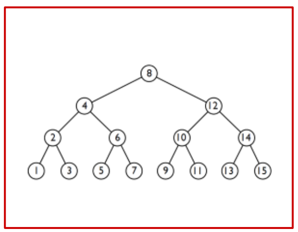
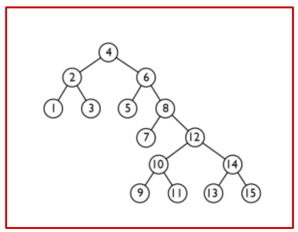
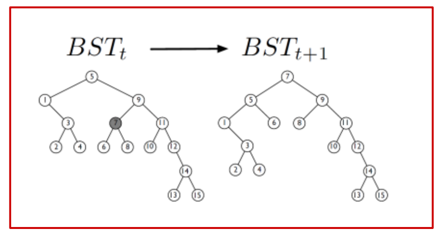
Duo: A High-Throughput Reconfigurable Datacenter Network Using Local Routing and Control
Johannes Zerwas, Csaba Györgyi, Andreas Blenk, Stefan Schmid, and Chen Avin.
ACM SIGMETRICS and ACM Performance Evaluation Review (PER), Orlando, Florida, USA, June 2023.

Credence: Augmenting Datacenter Switch Buffer Sharing with ML Predictions
Vamsi Addanki, Maciej Pacut, and Stefan Schmid.
21st USENIX Symposium on Networked Systems Design and Implementation (NSDI), Santa Clara, California, USA, April 2024.

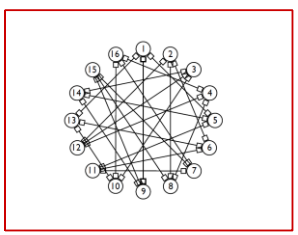
Cerberus: The Power of Choices in Datacenter Topology Design (A Throughput Perspective)
Chen Griner, Johannes Zerwas, Andreas Blenk, Manya Ghobadi, Stefan Schmid, and Chen Avin.
ACM SIGMETRICS and ACM Performance Evaluation Review (PER), Mumbai, India, June 2022.

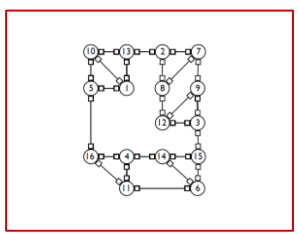
Why "Self-Adjusting" Networks?

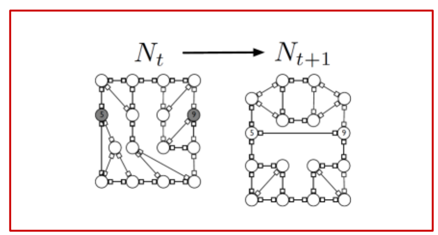
Connection to Datastructures & Coding

Traditional BST (Worst-case coding)	Demand-aware BST (Huffman coding)	Self-adjusting BST (Dynamic Huffman coding)
		

More structure: improved **access cost** / shorter **codes**





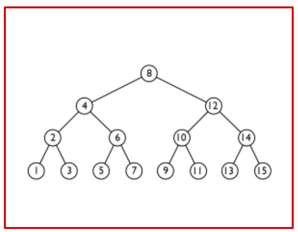


Similar **benefits**?

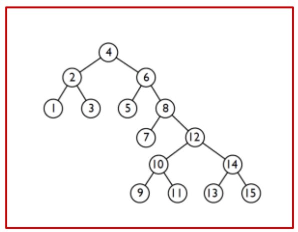
Why “Self-Adjusting” Networks?

Connection to Datastructures & Coding

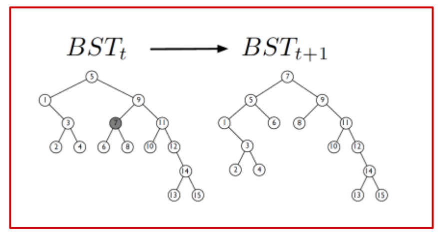
Traditional BST
(Worst-case coding)



Demand-aware BST
(Huffman coding)

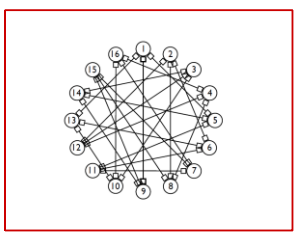


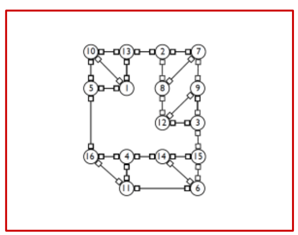
Self-adjusting BST
(Dynamic Huffman coding)

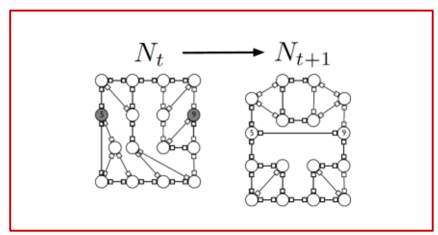


More than an analogy!

More structure: improved **access cost** / shorter **codes**







Similar **benefits**?

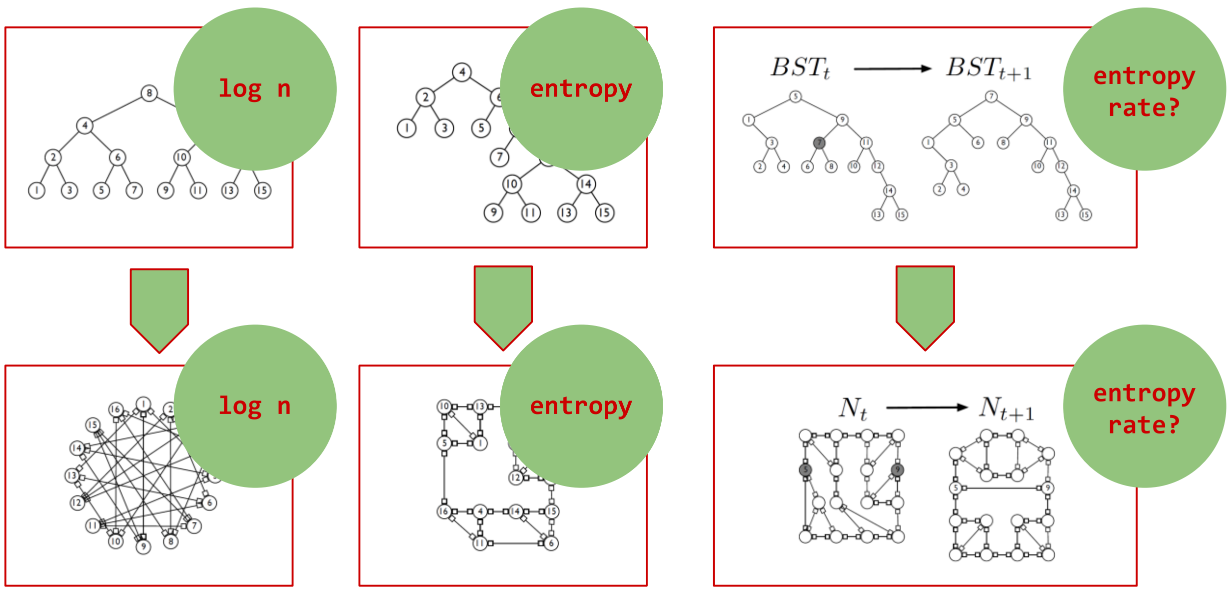
Why "Self-Adjusting" Networks?

Connection to Datastructures & Coding

Traditional BST
(Worst-case coding)

Demand-aware BST
(Huffman coding)

Self-adjusting BST
(Dynamic Huffman coding)



Reduced expected **route lengths**!

More than
an analogy!

Generalize methodology:
... and transfer entropy bounds and algorithms of data-structures to networks.

First result:
Demand-aware networks of asymptotically optimal route lengths.

Credence Teaser

Credence: Augmenting Datacenter Switch Buffer Sharing with ML Predictions

Vamsi Addanki
TU Berlin

Maciej Pacut
TU Berlin

Stefan Schmid
TU Berlin

Abstract

Packet buffers in datacenter switches are shared across all the switch ports in order to improve the overall throughput. The trend of shrinking buffer sizes in datacenter switches makes buffer sharing extremely challenging and a critical performance issue. Literature suggests that push-out buffer sharing algorithms have significantly better performance guarantees compared to drop-tail algorithms. Unfortunately, switches are unable to benefit from these algorithms due to lack of support for push-out operations in hardware. Our key observation is that drop-tail buffers can emulate push-out buffers if the future packet arrivals are known ahead of time. This suggests that augmenting drop-tail algorithms with predictions about the future arrivals has the potential to significantly improve performance.

This paper is the first research attempt in this direction. We propose CREDENCE, a drop-tail buffer sharing algorithm aug-

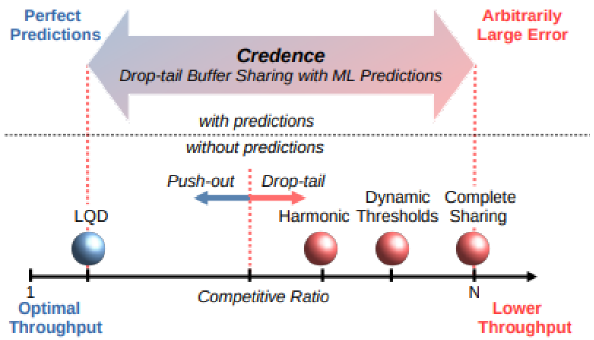


Figure 1: Augmenting drop-tail buffer sharing with ML predictions has the potential to significantly improve throughput compared to the best possible drop-tail algorithm (without predictions), and unlock the performance that was