Plan

* Feedback summary
* Recitation Qs
* Background on DCs
* Queue game
* DCTCP

Logistics

* DP prelim report due TODAY 5pm.
* Participation check-in #2 out this weekend.
* Midterm on Tuesday, April 6
  ↳ example exams online

1. What is the goal of DCTCP?
   - ↳ Increase perf for DCs (latency, ...)
   - ↳ No new hardware
2. How does DCTCP differ from TCP?
   - ↳ ECN — early congestion feedback
3. Why does DCTCP differ from TCP?
   - ↳ Characteristics of DC traffic different than net traffic.

What makes a great research result? (Spielman)

1. Beautiful theory.

2. Works in practice.

3. Solves a problem that people care about.

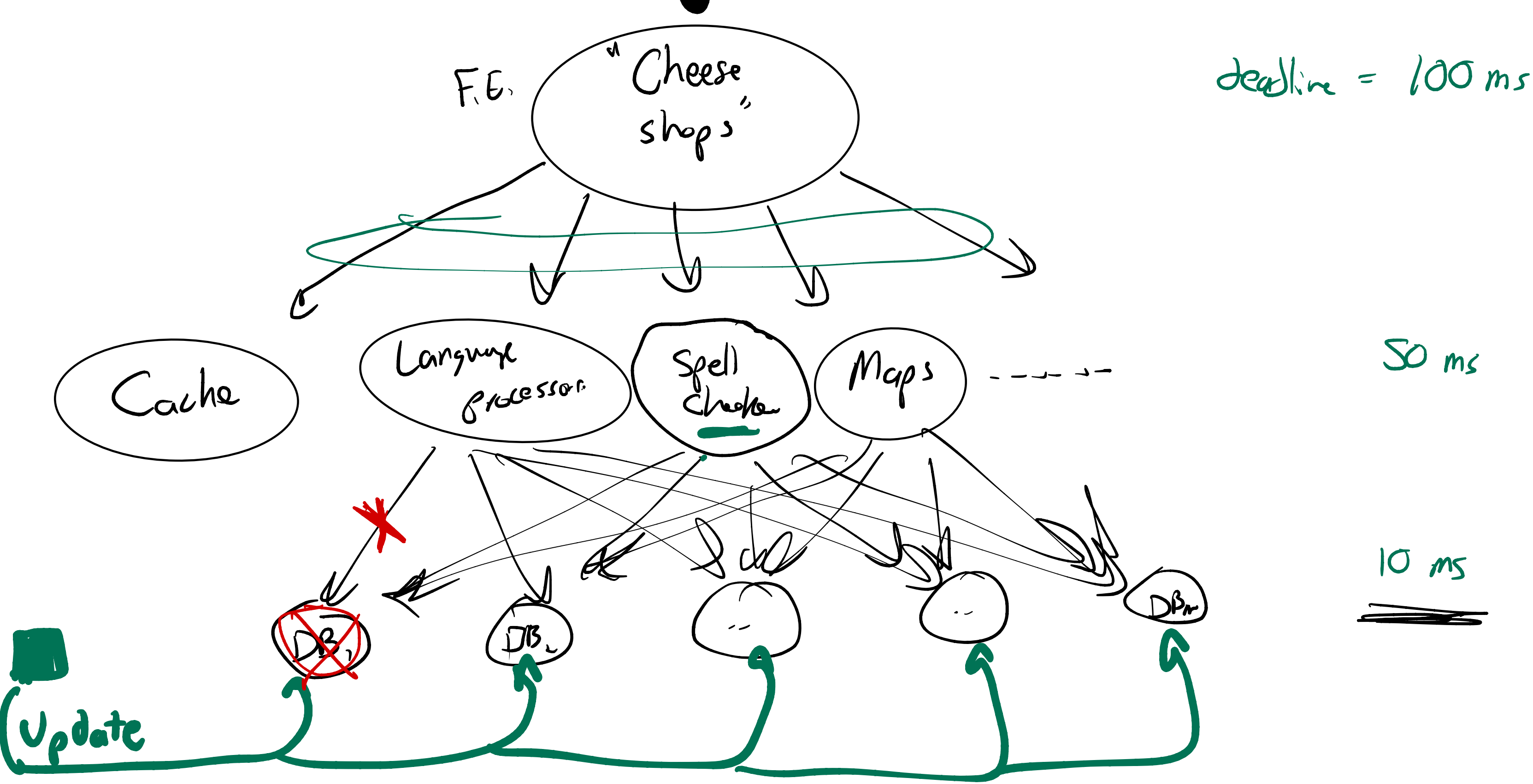
Types of flows in DC

1. Query flows ~2 kB
2. Bg. traffic ~50 MB
3. Short message ~1 MB

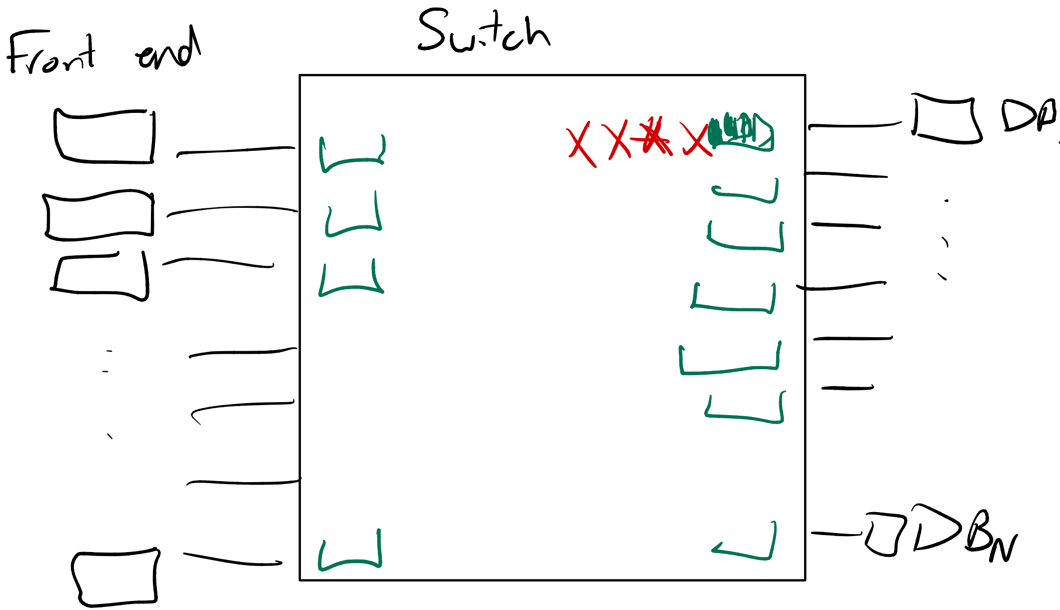
throughput

latency sensitive

e.g. Search


F.E.
"Cheese shops"
deadline = 100 ms
Cache
Language processor
Spell checker
Maps
50 ms
DB1
DB2
DBn
10 ms
Update

Queues

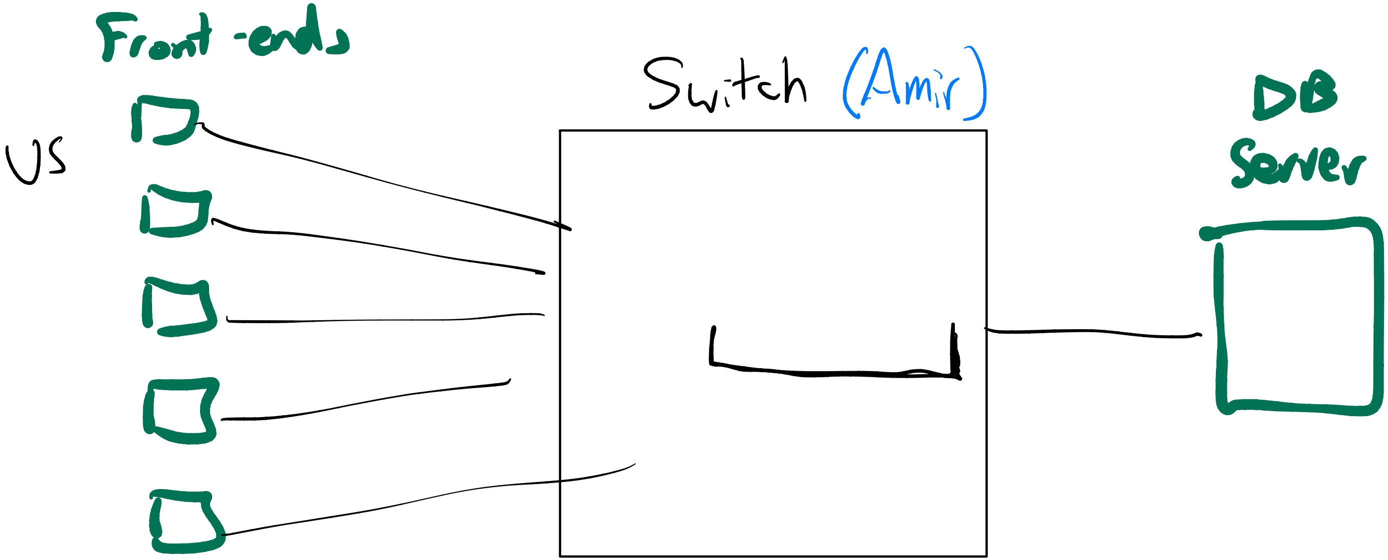


Two worries

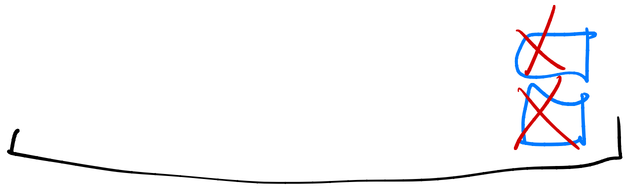
1. Dropped packets

2.

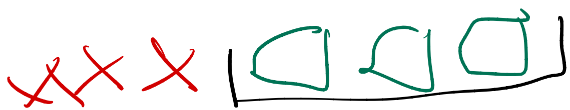
Queue Game



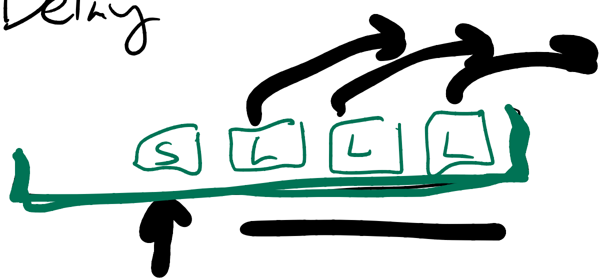
1. One short flow ✓



2. Many short flow same time "Incast"



3. Queuing Delay

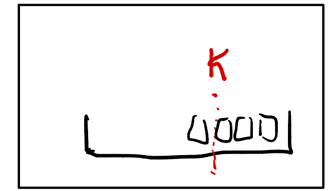


What does DCTCP do to fix?

SWITCH

- Explicit congestion notification.
  - ↳ Set bit on packet
  - ↳ Difference w/ prior art: switch sets bit early

Switch



SENDER

- 1st start getting ECN → slow down gradually
  - ↳ Normal TCP → really slows down → sharply

Why doesn't this work on Internet?

- Need to modify both ends and switch ?!
- Convergence time depends on RTT

↓

- Feedback is too slow

Plan

* Feedback summary
* Recitation Qs
* Background on DCs
* Queue game
* DCTCP

Logistics

* DP prelim report due TODAY 5pm.
* Participation check-in #2 out this weekend.
* Midterm on Tuesday, April 6
  ↳ example exams online

DCTCP

1. What is the goal of DCTCP?
   ↳ improve on TCP in DCs low latency, high throughput

2. How does DCTCP differ from TCP?
   ↳ More clever use of explicit congestion notification (ECN)

3. Why does DCTCP differ from TCP?
   ↳ Take advantage of unique properties of DC

   { 0.1 ms RTT
     ~50 ms RTT

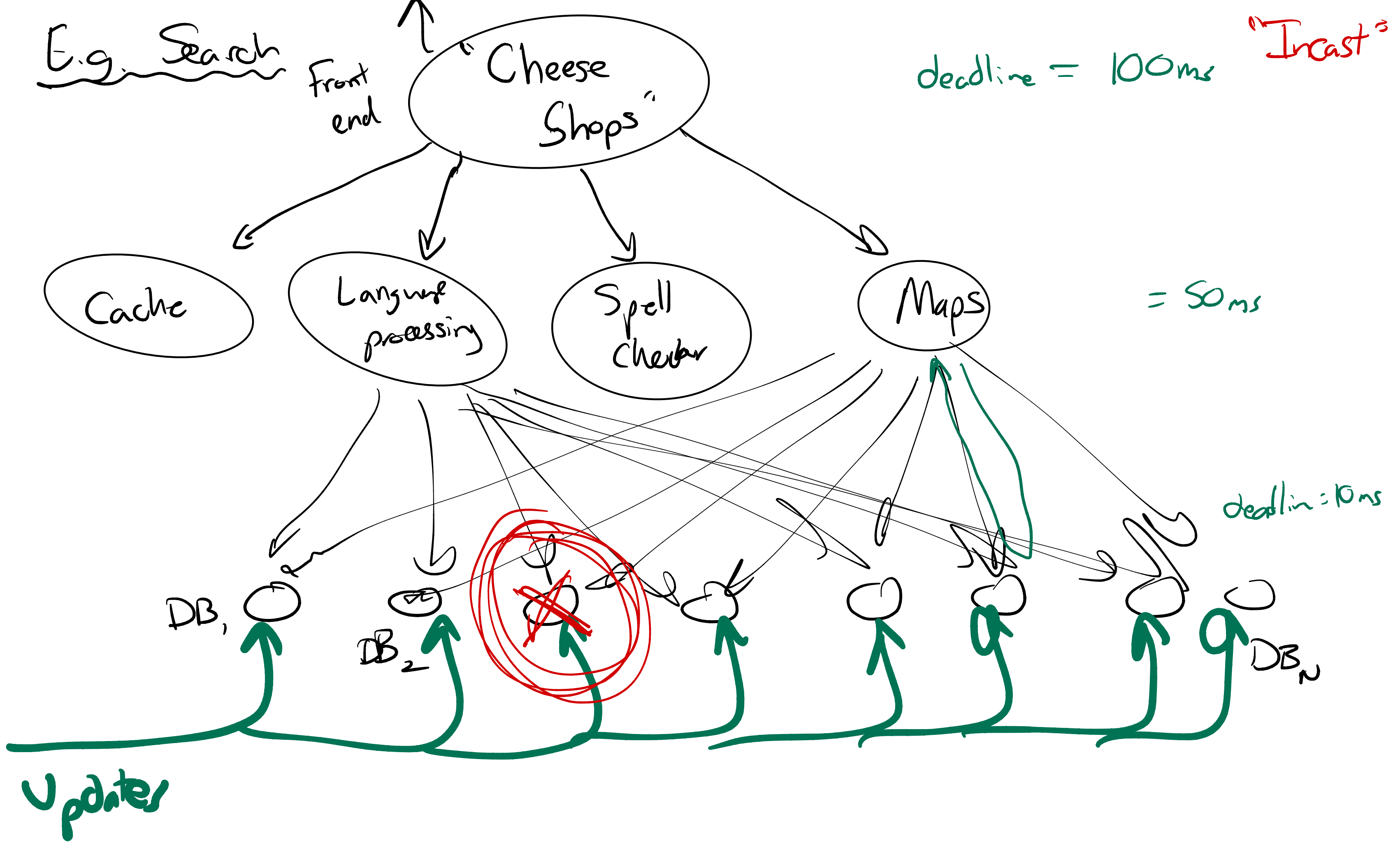
What makes a great research result? (Spielman)

1. Beautiful theory
2. Works in practice
3. Solves a problem that people care about.

Types of flows

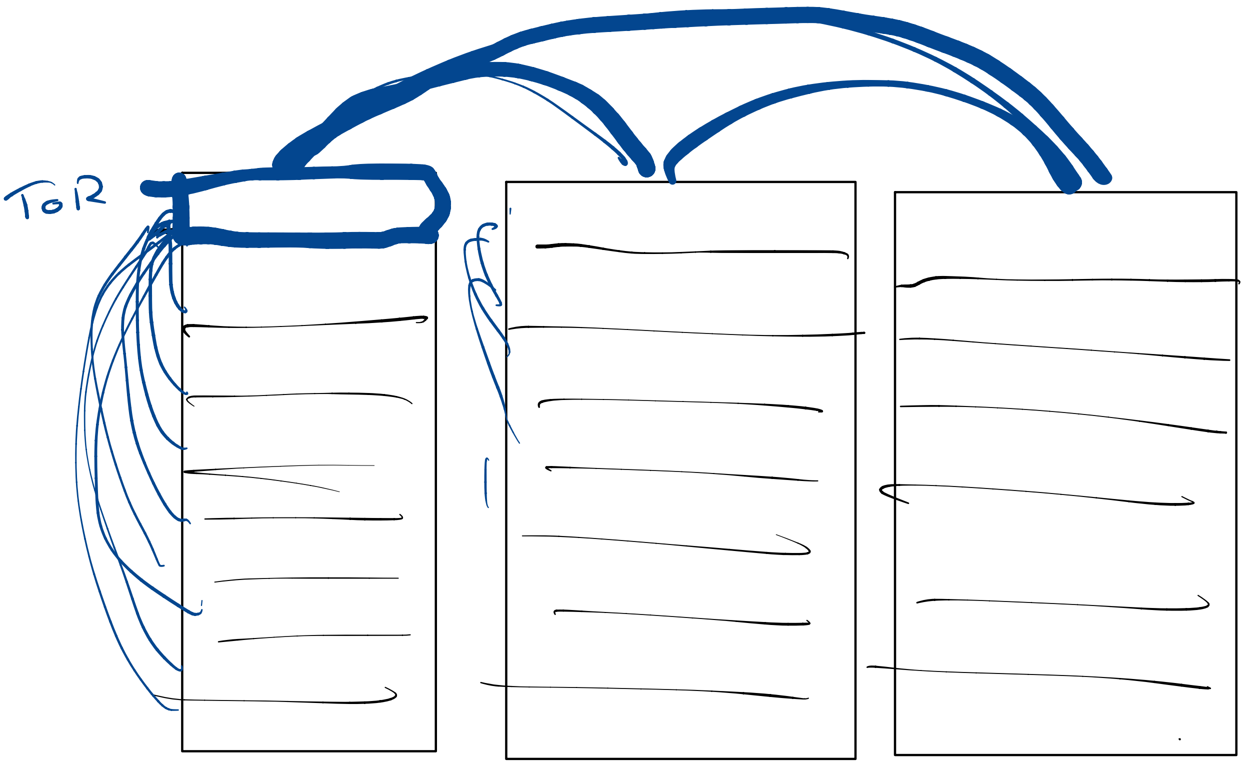
1. Short/low-latency "Query" ~2 kB
2. Long .... background flows ~100 MB
3. Short msg traffic ~1 MB

E.g. Search



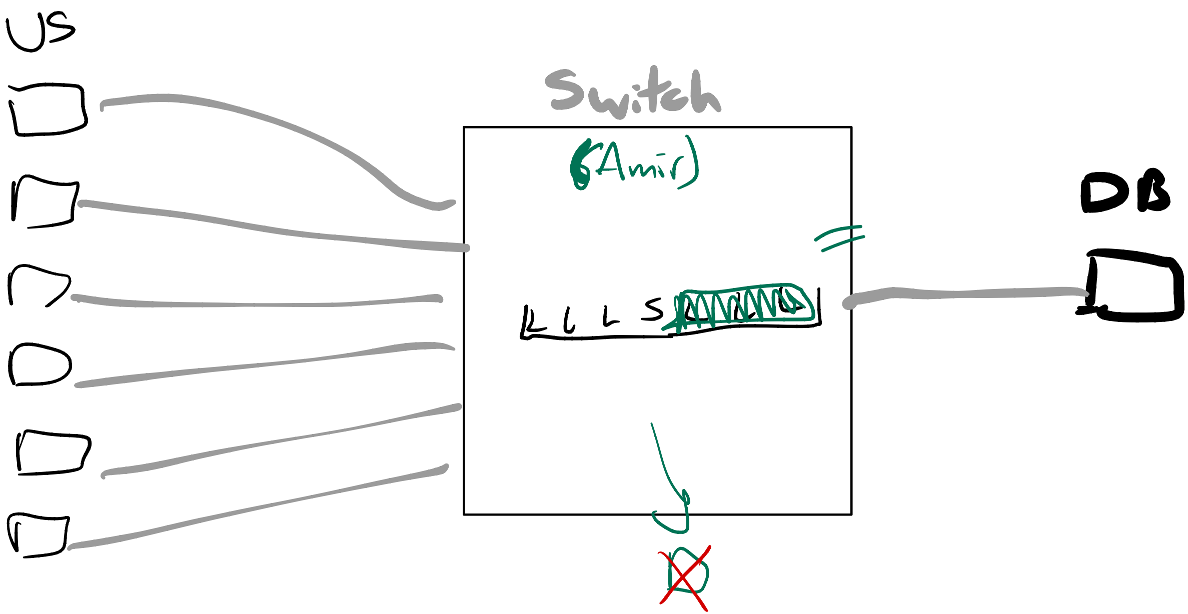
Front end
"Cheese Shops"
"Incast"
deadline = 100ms
Cache
Language processing
Spell Checker
Maps
= 50ms
deadline = 10ms
DB1
DB2
DBN
Updates

Queues


ToR

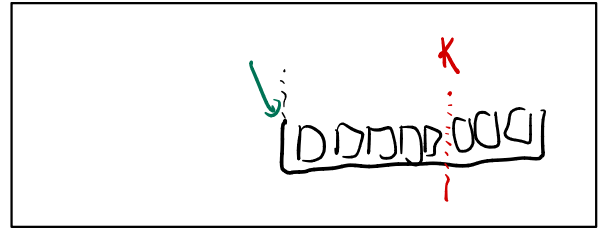
Queue Game

1. Drop packet
2. Queuing delay



DCTCP

Switch

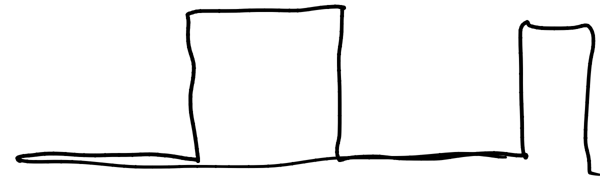


SWITCH

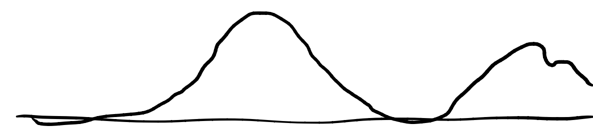
- Early notification of congestion.

Normal : Wait under buffer full

TCP



DCTCP



SENDER

- Keeps estimate of queue length
  ↳ adjusts tx rate accordingly

How is this different from normal TCP?
↳ TCP: Cut window size in half
↳ Quantitative approach

Why doesn't this work on Internet?

- Scale .... Convergence time = time before every endpoint tx at "right" rate
  - Depends on RTT DC 0.1 ms, Net 50 ms
- Deployment
- Less structure !
-