CS 268
Graduate Computer Networks

Sylvia Ratnasamy

Spring 2023

About me: Sylvia Ratnasamy

- Background
  - PhD from UC Berkeley
  - Joined UCB in 2011 after ~10 years in industry research
  - Recent stints at a startup and at Google
  - Networking has been my focus throughout

We have a GSI!

- Shishir Patil
  - 4th yr PhD student with Joey Gonzalez and Prabal Dutta
  - Interests: Systems for ML, ML at the edge, Sky computing

Today's lecture

- What is this course about?
- Course logistics

Goals of this course

- To become familiar with Internet research
- To learn how to analyze a proposal
  - Critique problem selection *and* solution
- To get some practice in the art of expressing your point of view
- To apply what you've learnt by doing an original piece of work

What's involved? Quick glance.

- Read ~50 papers
- You submit a review for each paper
- We'll discuss them together in class
- Each student will lead the in-class discussion for one paper
- You define, execute, and present a research-oriented class project
- No exams or problem sets
- 6-7 fantastic guest speakers! (MIT, Google, UW, Microsoft)

Why study the Internet?

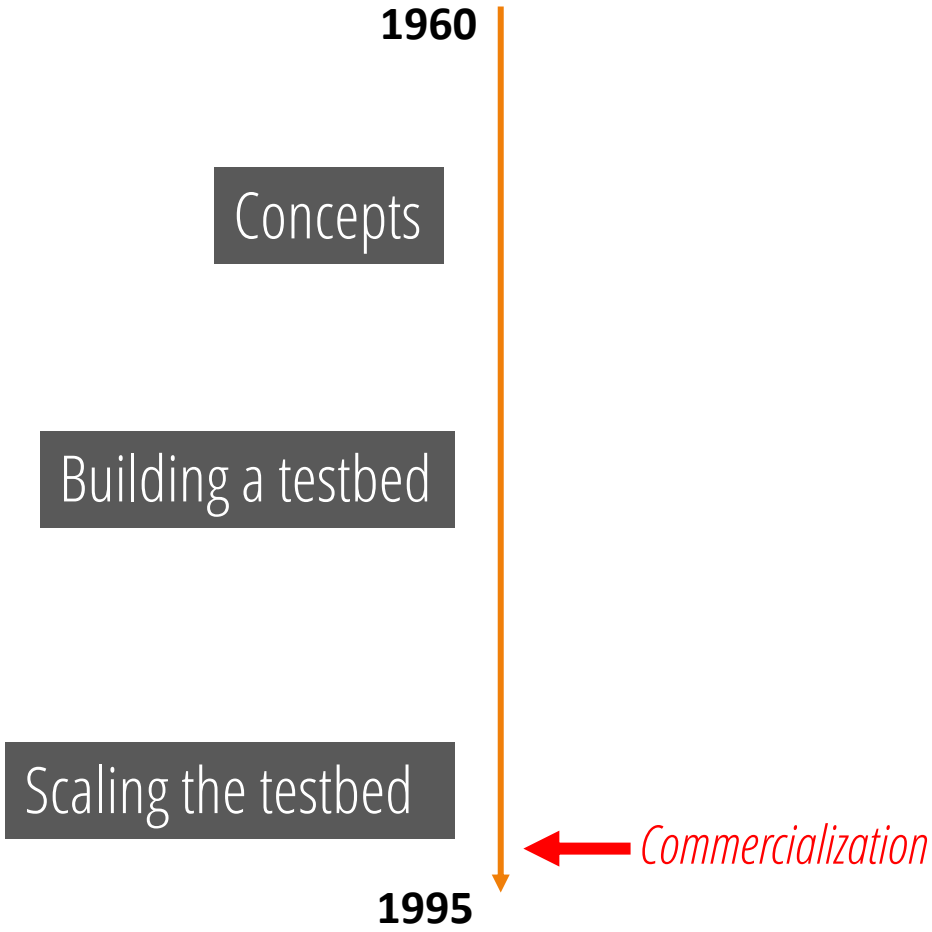
- As a system design problem, the Internet is *uniquely challenging*
- As a practical artifact, it has been *uniquely impactful*

Evolution in (and on) the Internet

Three broad phases

1. Building a global data communication network
2. Scaling; and the emergence of a commercial ecosystem (ISPs, vendors)
3. Innovation; and a shifting commercial ecosystem (CSPs, last mile operators)

Phase 1: Building a global data communication network



packet switching

statistical multiplexing

TCP/IP

routing

DNS

email

HTTP

Congestion Control

Military interest in a communication infrastructure to withstand a nuclear attack

Scientists want to share data

Impact: transformed how humans communicate

With email as the “killer app”, we went:

- From voice to data communication
- From synchronous to asynchronous communication
- From one-to-one to group communication

A lesson in the power of inventing a fundamentally new capability!

Designing for the Internet is uniquely challenging

- A federated system
- Tremendous scale
- Enormous diversity
- Enormous dynamic range
- Prone to failure
- Distributed and asynchronous

Designing for the Internet is uniquely challenging

- Too complex for theoretical models
- "Working code" doesn't mean much

Addressing these challenges led to a new design paradigm
(one that changed computer science!)

Impact: the Internet design paradigm

- Decentralized control
- Modularity through layering
- Best-effort
- IP as “narrow waist” interface
- Dumb network / smart endpoints
- The end-to-end design principle
- Fate-sharing

A radical departure from systems at the time

Now ubiquitous in how we design large-scale systems

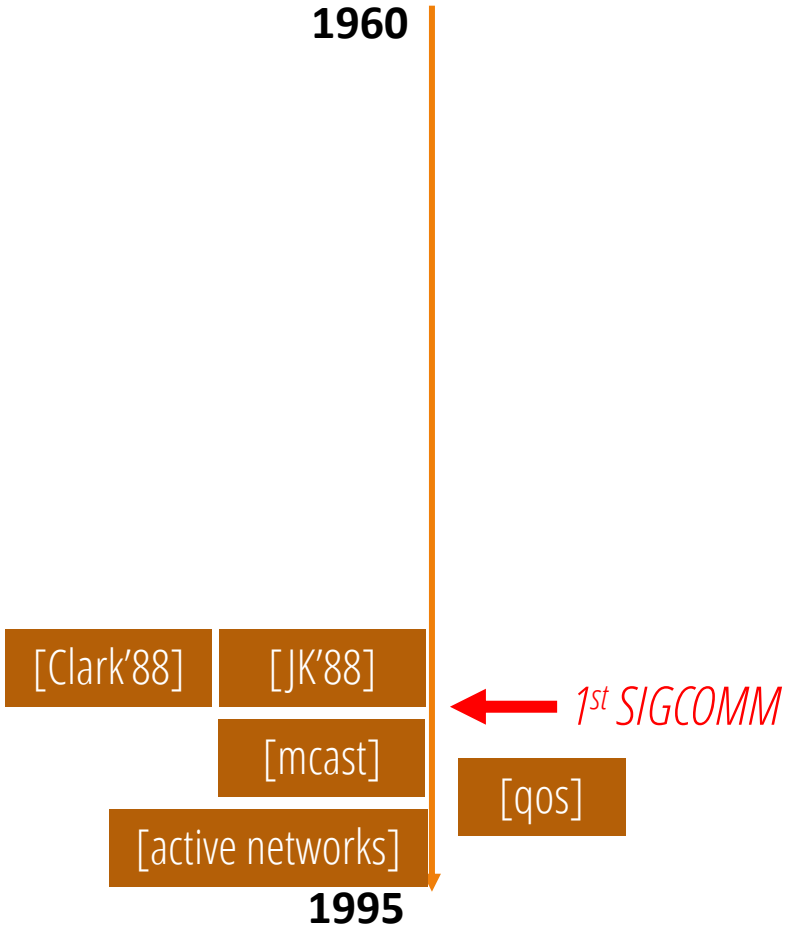
Example: a best-effort service model

- Fundamental question: what's the right service model that a network should support?
  - "contract" between network and its users/end-hosts

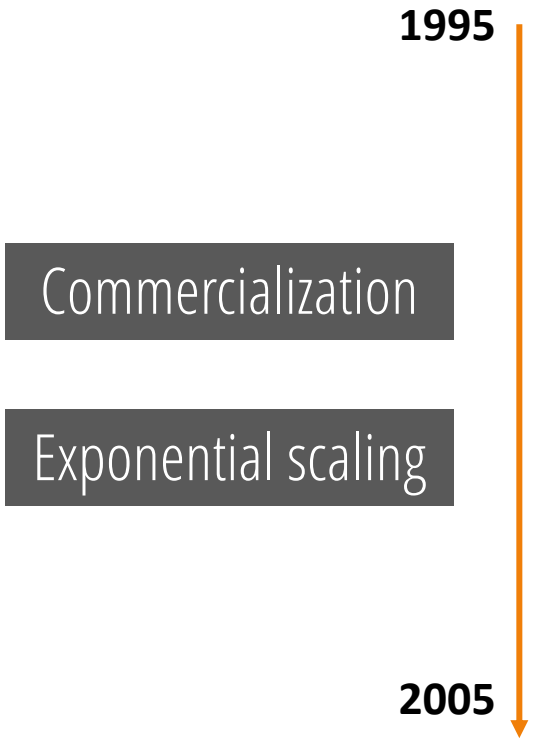
- Some possibilities:
  - "guarantee that data will be delivered"
  - "guarantee that data will be delivered within X time"
  - "return a confirmation of successful delivery or an error"

- Instead, what the Internet supports: "best effort" delivery of data
  - No guarantee on whether or when data will be delivered
  - No notification of outcome!

Phase 1 readings: the classics



Phase 2: Scaling & the emergence of a commercial ecosystem



BGP

Firewalls

Traffic engineering

NAT

Content delivery networks

High-speed routers

"Content is king" – *Bill Gates*

Proliferation of PCs

Impact: everything moves to the Internet

- Old content
- New content
- Brick-and-mortar businesses
- Banks, enterprises, government, …

A lesson in the power of building a *general* capability!

Challenge: many gaps in the original design

- Decentralized control
- Layering
- Best-effort
- IP as “narrow waist” interface
- Dumb network / smart endpoints
- The end-to-end design principle
- Fate-sharing

No operational support!

Security not addressed

No way to isolate/differentiate traffic/customers

Routing does not reflect business relationships

No blueprint for how we upgrade and evolve

Impact: growing the Internet design paradigm

- The role of measurement
- The role of policy
- The role of management
- The role of security
- …

Now standard components in operating any large-scale system

Impact (research): shining a spotlight

Tremendous innovation on top of the Internet infrastructure but challenging to innovate within the infrastructure itself

Overcoming the Internet Impasse through Virtualization

A Blueprint for Introducing Disruptive Technology into the Internet*

Tussle in Cyberspace: Defining Tomorrow's Internet

David D. Clark
MIT Lab for Computer Science
ddc@lcs.mit.edu

John Wroclaw
MIT Lab for Computer
itw@lcs.mit.ed

A Clean Slate 4D Approach to Network Control and Management *

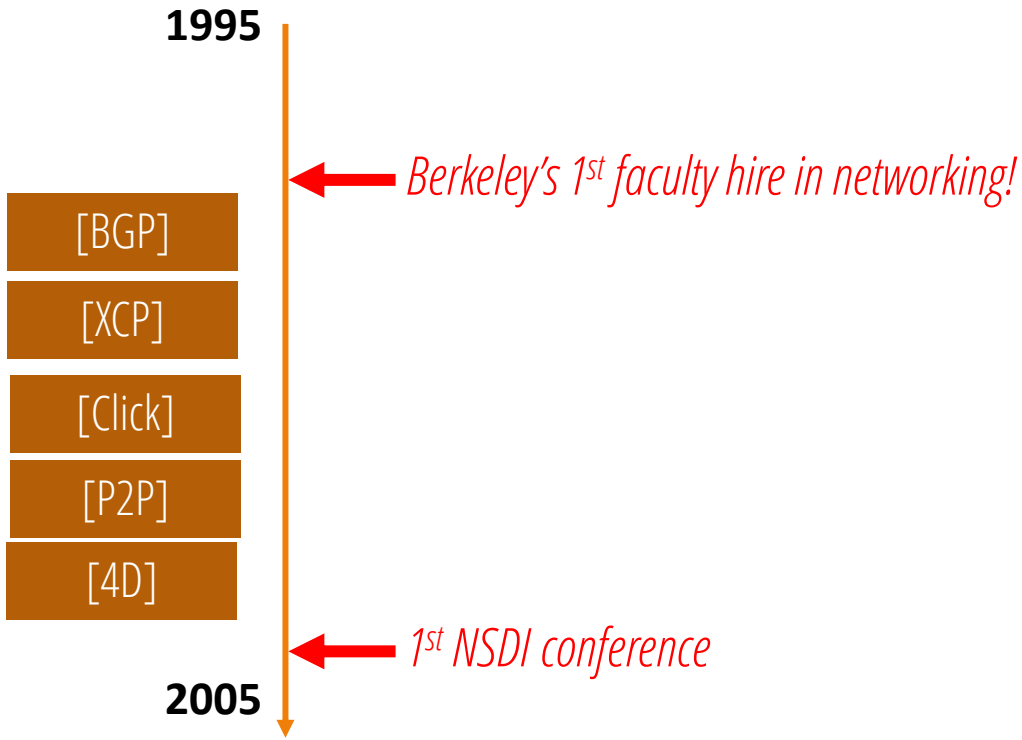
Albert Greenberg, Gisli Hjalmtysson, David A. Maltz, Andy Myers,
Jennifer Rexford, Geoffrey Xie, Hong Yan, Jibin Zhan, Hui Zhang

Impact (research): shining a spotlight

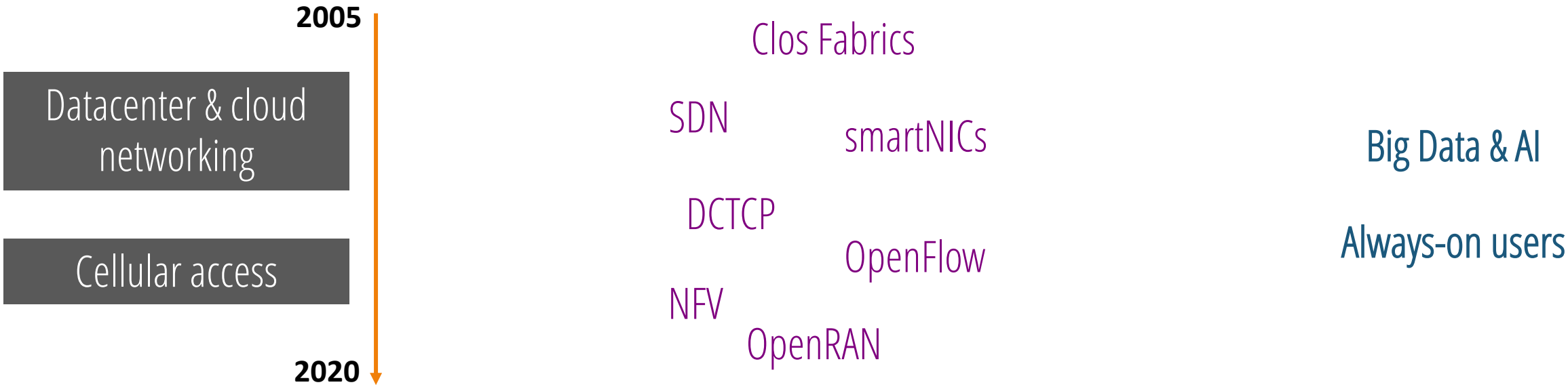
How do we architect for innovation in a complex ecosystem?
(vendors, operators, users)

- How do we reconcile federation with competition and innovation?
- How do we reconcile a universal service model with evolution?
- …

Phase 2 readings: more classics and new foundations



Phase 3: Innovation and a shifting ecosystem



Impact: the Internet is everywhere

Transforming everything

- The way we generate and consume content (live streaming, search)
- The way we do business (advertising, in-app purchases)
- The way we have relationships (facebook, twitter)
- The way we learn (wikipedia, MOOCs, chatGPT)
- The way we govern (e-voting, censorship, cyber-warfare)
- The way we cure disease (digital health, remote medicine)

Challenge: more gaps in the design …

- Decentralized control
- Layering
- Best-effort
- IP as “narrow waist” interface
- Dumb network / smart endpoints
- The end-to-end design principle
- Fate-sharing

Critical goals for a hyperscale cloud operator:

Efficiency

Performance

Guaranteed availability

Need to innovate *within* the infrastructure to achieve these

… leading to new design paradigms

- **Programmable** infrastructure
- Centralization/consolidation
- The rise of customization
- Architecting for high performance

Many lessons in architecting for innovation and high performance

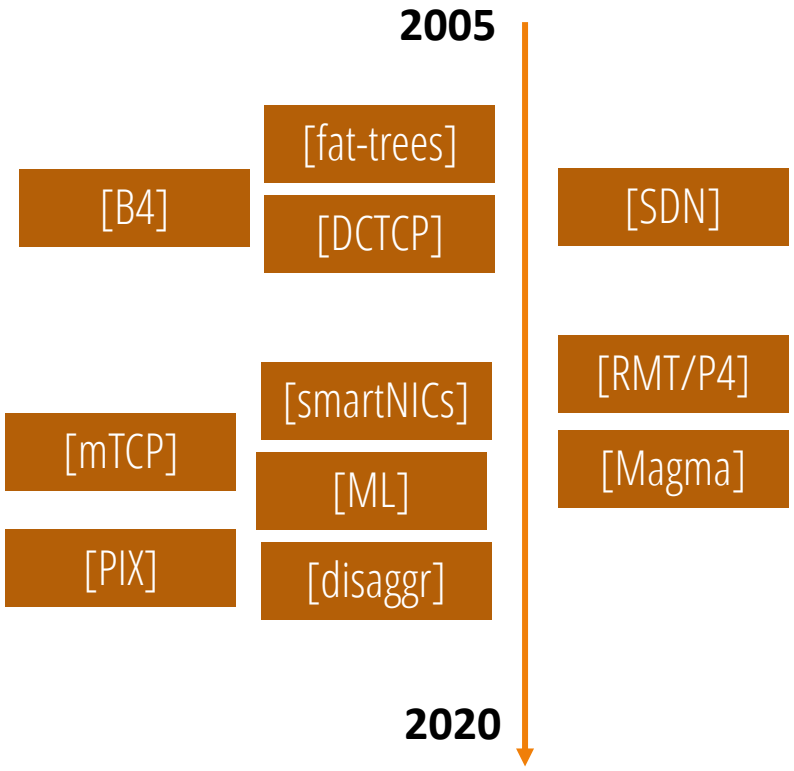
Impact: new design paradigms

- **Programmable** infrastructure (*vs.* "dumb network)
- Centralization/consolidation (*vs.* federation and distributed control)
- The rise of customization (*vs.* standardized general-purpose solutions)
- Architecting for high performance (*vs.* generality)

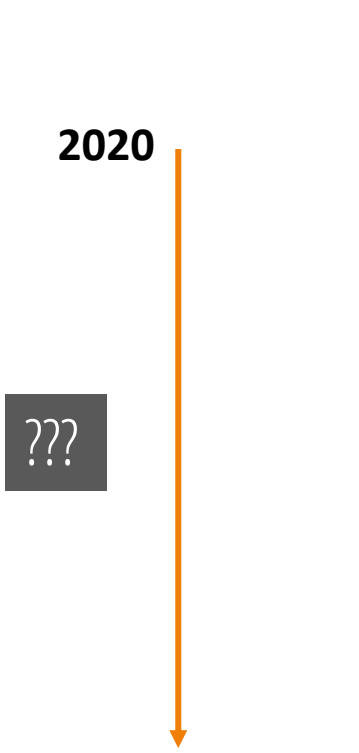
Many lessons in architecting for innovation and high performance

But only within a single provider and we're still understanding the tradeoffs!

Phase 3 readings: modern classics and the state-of-the-art



Finally, we'll speculate on the next phase



IoT, drones, satellite networks, …

The Internet interacting with the physical world

Blurring the boundary between compute and network infrastructure

Grappling with specialization & networking vs. the Internet

Societal issues: access, privacy, transparency

Growing calls for decentralization and democratization

Why study the Internet?

- As a system design problem, the Internet is *uniquely challenging*
- As a practical artifact, it has been *uniquely impactful*
- Constant evolution means we always have new challenges and opportunities

An incredible lesson in system design and architecting for innovation

And fun, because we're still debating the big questions

- Decentralized control → *But centralization simplifies management*
- Layering → *But collapsing layers improves performance*
- Best-effort → *But we need performance guarantees and verification*
- IP as "narrow waist" interface → *Specialization*
- Dumb network / smart endpoints → *smartNICs, programmable switches*
- The end-to-end design principle → *in-network computation*

Internet Research

- Epicenter is SIGCOMM, HotNets, NSDI
  - Also SOSP/OSDI, Mobicom, IMC, *etc.*
- Inherently broad in scope
  - Systems people but also theorists, PL, economists, hardware folks, etc.
- Big-picture architectural discussions tend(ed?) to dominate
  - But also new technologies, discoveries, methods, apps, *etc.*

Today's lecture

- What is this course about?
- Course logistics

Course information

- Website: https://cs268.io
- Announcements: Ed
- Reach me and Shishir via Ed (preferred) or email
- Prereq: CS168 or equivalent (http://cs168.io)

Enrollment and the waitlist

- I will not be increasing the class size
  - If you're planning to drop, please do so quickly

- Undergrads:
  - We hope to have enrollment codes by end of week at which point you can join the waitlist
  - In the meantime, please email Shishir so we know how to contact you

Grading

Paper reviews and class participation	30%
Paper presentation	30%
Term project	40%

- Can miss up to 3 paper reviews with no penalty (don't need to inform us when you do)
- Frequent absenteeism will affect your grade

Reading papers

- Plenty of advice out there
- My take: don't overthink it. Just read it. Start to end. In depth.
- Then set the paper aside and think
    - What problem are they solving? -- in 1-2 sentences
    - Replay their motivating arguments – do they make sense?
    - Replay how their solution works for a simple example -- can you? If not, go back to step 1.
- Once you're sure you understand, start critiquing
    - Is the problem important? ambitious? hard? have a long shelf-life?
    - Is the solution effective? Under what conditions does it break?
    - What other approaches are possible? Etc.

Reviewing papers

- Write a short review for each paper
  - Review ≠ summary !!
  - Length: ~one page
  - Be honest

- Submissions via google forms – Shishir will post the links on Ed.

- Reviews are due 5pm the night before class

Typical format of the review

1. What is the problem being addressed?
2. Do you believe the problem is/was important?
   - Explain your thinking
   - Consider context
3. What is the solution's main insight (nugget)?
4. Do you think the solution is a good one? (strengths and weaknesses)
   - Explain
5. Did you enjoy the paper?

Lecture format

- 40 minutes per paper
  - ~15 minutes presentation and initial discussion
  - ~25 minutes deep-dive discussion all together

- Class discussion
  - Come prepared to discuss the main ideas, gaps, etc
  - We will all learn from each other
  - Let's have **no open laptops** in class

Presenting Papers

- Each of you will be responsible for presenting and leading the discussion on one paper
- Skim the syllabus on cs268.io and look for lectures marked as “Student Led” ; you will be presenting one of these papers
  - Shishir will post instructions for submitting your presentation preferences on Ed
  - Submit your preferences by 01/23 and we’ll do assignments by 01/25
  - We’ll follow a first-come-first-serve policy for assignments
- Student presentations will start on 01/31

Presenting papers (cont'd)

- Organize presentation to suit your style
  - Summarize-then-discuss, discuss-as-you-go, ...

- Some tips/expectations
  - Assume the class has read the papers →recap, don't explain
  - Go beyond the assigned paper: your job is to give the class a more complete picture
  - Prepare a set of questions to initiate discussions

- Send me and Shishir your slides at least THREE days before your presentation

- Recall: this is 30% of your grade!

Research Project

- Investigate new ideas and solutions in a class research project
  - Define the problem
  - Execute the research
  - Write up and present your research
- Undergrads must partner with a grad student
  - Highly encourage everyone to work with a partner
- Deadlines and details for project proposal and checkpoints will be posted on Ed
- Start early and consider scope

Recap: this course

- We will read and review ~50 papers

- Three components to your grade
  - Project (40%)
  - Paper presentation (30%)
  - Paper reading and class participation (30%)

Next Lecture

- Classics in Internet design
  - Required reading: Internet design principles [Clark'88]
  - Active networks [Tennenhouse and Wetherall'95]
  - Optional reading (but highly encouraged): end-to-end principle [Clark'81]

Recap: Your Immediate Action Items

- Read and review Clark'88 and ActiveNets
- Submit your preferences for papers you'd like to present
- If you plan to drop this course, please let us know promptly (and officially drop)

Thanks!