

Welcome back to [occ]!

## Week 2 Housekeeping:

- ▶ You did amazing things this Summer! Thanks for sharing.
- ▶ Keep website “bug reports” coming! Thank you.
- ▶ Plan for the week.
- ▶ All readings for Wednesday and Friday are posted.
- ▶ Watch email for a practice “assignment” on Canvas.
- ▶ Discussion Questions to be explained/distributed Wednesday.
- ▶ Never too early to meet with Sneha (or me).

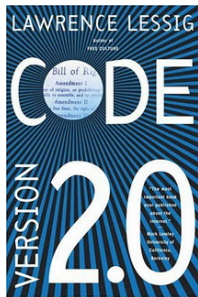
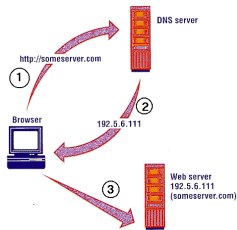
Welcome back to [occ]!

### Week 2 Housekeeping:

- ▶ You did amazing things this Summer! Thanks for sharing.
- ▶ Keep website “bug reports” coming! Thank you.
- ▶ Plan for the week.
- ▶ All readings for Wednesday and Friday are posted.
- ▶ Watch email for a practice “assignment” on Canvas.
- ▶ Discussion Questions to be explained/distributed Wednesday.
- ▶ Never too early to meet with Sneha (or me).

# Today's Flight Plan:

- ▶ **Foundations:** Code and other architectures of control.
- ▶ **Dynamic:** Governance & regulation
- ▶ **Cases:** TCP/IP, the WELL.
- ▶ **(Challenge:** Managing conflict, rules, and power within communities.)



## Today's Flight Plan:

- ▶ **Foundations:** Code and other architectures of control.
- ▶ **Dynamic:** Governance & regulation
- ▶ **Cases:** TCP/IP, the WELL.
- ▶ **(Challenge:** Managing conflict, rules, and power within communities.)



These are the touchstones for today's class.

- First, we're going to address some of this through a group demonstration of how TCP/IP works.
- Then we'll talk more about Lessig's ideas on the subject.
- Then we return to the WELL to apply Lessig's ideas and bring out some of the key tensions there.
- Synthesize key takeaways.

## Packet routing principles:

Key rules here:

- We're going to count how many hops it takes.
- Packet start positions will be "random"
- Your job is to transfer the packet closer to its destination ("best effort delivery").
- No peeking inside packets ("neutral network").
- (Round 2) If Sneha taps you on the shoulder, you are down for maintenance.
- (Round 3) If Sneha taps you on the shoulder, you are going to start routing things to the wrong destination.
- End of round 3: endpoints open your packets.

2014-09-29

└ TCP/IP (grossly simplified) demo!

Packet routing principles:

► Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.

## Packet routing principles:

- Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.

Key rules here:

- We're going to count how many hops it takes.
- Packet start positions will be "random"
- Your job is to transfer the packet closer to its destination ("best effort delivery").
- No peeking inside packets ("neutral network").
- (Round 2) If Sneha taps you on the shoulder, you are down for maintenance.
- (Round 3) If Sneha taps you on the shoulder, you are going to start routing things to the wrong destination.
- End of round 3: endpoints open your packets.

2014-09-29

└ TCP/IP (grossly simplified) demo!

#### Packet routing principles:

- ▶ Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.
- ▶ I'm assigning the person in each corner an "address" on the board.

## Packet routing principles:

- ▶ Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.
- ▶ I'm assigning the person in each corner an "address" on the board.

Key rules here:

- We're going to count how many hops it takes.
- Packet start positions will be "random"
- Your job is to transfer the packet closer to its destination ("best effort delivery").
- No peeking inside packets ("neutral network").
- (Round 2) If Sneha taps you on the shoulder, you are down for maintenance.
- (Round 3) If Sneha taps you on the shoulder, you are going to start routing things to the wrong destination.
- End of round 3: endpoints open your packets.

2014-09-29

└ TCP/IP (grossly simplified) demo!

#### Packet routing principles:

- ▶ Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.
- ▶ I'm assigning the person in each corner an "address" on the board.
- ▶ Your job is to get the packet closer to its destination & to keep count of the cumulative number of hops for each packet.

## Packet routing principles:

- ▶ Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.
- ▶ I'm assigning the person in each corner an "address" on the board.
- ▶ Your job is to get the packet closer to its destination & to keep count of the cumulative number of hops for each packet.

Key rules here:

- We're going to count how many hops it takes.
- Packet start positions will be "random"
- Your job is to transfer the packet closer to its destination ("best effort delivery").
- No peeking inside packets ("neutral network").
- (Round 2) If Sneha taps you on the shoulder, you are down for maintenance.
- (Round 3) If Sneha taps you on the shoulder, you are going to start routing things to the wrong destination.
- End of round 3: endpoints open your packets.

2014-09-29

└ TCP/IP (grossly simplified) demo!

#### Packet routing principles:

- ▶ Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.
- ▶ I'm assigning the person in each corner an "address" on the board.
- ▶ Your job is to get the packet closer to its destination & to keep count of the cumulative number of hops for each packet.
- ▶ No peeking inside packets!

## Packet routing principles:

- ▶ Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.
- ▶ I'm assigning the person in each corner an "address" on the board.
- ▶ Your job is to get the packet closer to its destination & to keep count of the cumulative number of hops for each packet.
- ▶ No peeking inside packets!

Key rules here:

- We're going to count how many hops it takes.
- Packet start positions will be "random"
- Your job is to transfer the packet closer to its destination ("best effort delivery").
- No peeking inside packets ("neutral network").
- (Round 2) If Sneha taps you on the shoulder, you are down for maintenance.
- (Round 3) If Sneha taps you on the shoulder, you are going to start routing things to the wrong destination.
- End of round 3: endpoints open your packets.

- ▶ Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.
- ▶ I'm assigning the person in each corner an "address" on the board.
- ▶ Your job is to get the packet closer to its destination & to keep count of the cumulative number of hops for each packet.
- ▶ No peeking inside packets!
- ▶ More instructions tba in subsequent rounds...

## Packet routing principles:

- ▶ Four packets need to make it from "random" start positions to their destination (1 of 4) corners of the room.
- ▶ I'm assigning the person in each corner an "address" on the board.
- ▶ Your job is to get the packet closer to its destination & to keep count of the cumulative number of hops for each packet.
- ▶ No peeking inside packets!
- ▶ More instructions tba in subsequent rounds...

Key rules here:

- We're going to count how many hops it takes.
- Packet start positions will be "random"
- Your job is to transfer the packet closer to its destination ("best effort delivery").
- No peeking inside packets ("neutral network").
- (Round 2) If Sneha taps you on the shoulder, you are down for maintenance.
- (Round 3) If Sneha taps you on the shoulder, you are going to start routing things to the wrong destination.
- End of round 3: endpoints open your packets.



# Analyze the network we just made

2014-09-29

└ Analyze the network we just made

Key observations/questions:

- Why does addressing matter?
- Does identity of machines/people behind them matter?
- End-to-end principle: what about this network is “dumb”?
- Anything that can be broken up into a packet can travel through the network! e.g., different packet contents.
- Robust to failure of many nodes.
- Robust to some infections (viruses).
- Threats include: knocking out whole neighborhoods; overwhelming a single point; breaking address system

## Key design elements & rules?

## Effects of the architecture?

└ Analyze the network we just made

Key observations/questions:

- Why does addressing matter?
- Does identity of machines/people behind them matter?
- End-to-end principle: what about this network is “dumb”?
- Anything that can be broken up into a packet can travel through the network! e.g., different packet contents.
- Robust to failure of many nodes.
- Robust to some infections (viruses).
- Threats include: knocking out whole neighborhoods; overwhelming a single point; breaking address system

**Key design elements & rules?**

**Effects of the architecture?**

**Potential freedoms?**

Key observations/questions:

- Why does addressing matter?
- Does identity of machines/people behind them matter?
- End-to-end principle: what about this network is “dumb”?
- Anything that can be broken up into a packet can travel through the network! e.g., different packet contents.
- Robust to failure of many nodes.
- Robust to some infections (viruses).
- Threats include: knocking out whole neighborhoods; overwhelming a single point; breaking address system

**Key design elements & rules?**

**Effects of the architecture?**

**Potential freedoms?**

**Potential abuses?**

└ Analyze the network we just made

Key observations/questions:

- Why does addressing matter?
- Does identity of machines/people behind them matter?
- End-to-end principle: what about this network is “dumb”?
- Anything that can be broken up into a packet can travel through the network! e.g., different packet contents.
- Robust to failure of many nodes.
- Robust to some infections (viruses).
- Threats include: knocking out whole neighborhoods; overwhelming a single point; breaking address system

## Key design elements & rules?

## Effects of the architecture?

## Potential freedoms?

## Potential abuses?

## Vulnerabilities to disruption?

## Analyze the network we just made

### Key observations/questions:

- Why does addressing matter?
- Does identity of machines/people behind them matter?
- End-to-end principle: what about this network is “dumb”?
- Anything that can be broken up into a packet can travel through the network! e.g., different packet contents.
- Robust to failure of many nodes.
- Robust to some infections (viruses).
- Threats include: knocking out whole neighborhoods; overwhelming a single point; breaking address system

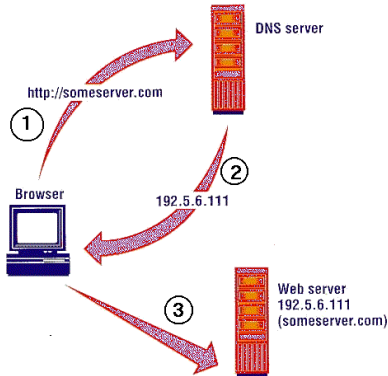
Key design elements & rules?

Effects of the architecture?

Potential freedoms?

Potential abuses?

Vulnerabilities to disruption?



Analyze the network we just made

Key design elements & rules?

Effects of the architecture?

Potential freedoms?

Potential abuses?

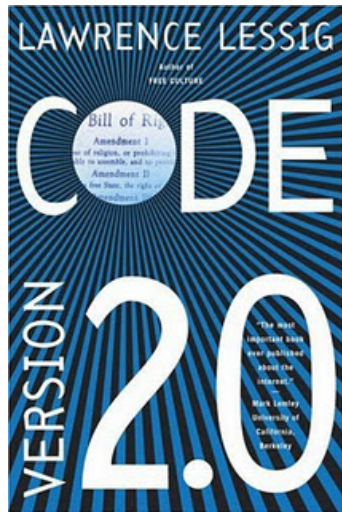
Vulnerabilities to disruption?



Key observations/questions:

- Why does addressing matter?
- Does identity of machines/people behind them matter?
- End-to-end principle: what about this network is “dumb”?
- Anything that can be broken up into a packet can travel through the network! e.g., different packet contents.
- Robust to failure of many nodes.
- Robust to some infections (viruses).
- Threats include: knocking out whole neighborhoods; overwhelming a single point; breaking address system

## Connect this to Lessig's core concepts:



└ Connect this to Lessig's core concepts:



### Architectures of control:

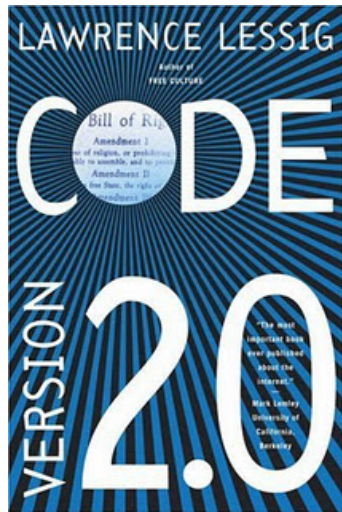
- Rules/laws
- Norms
- Code
- Prices (markets)

### Label in examples:

- specific architectures/manifestations
- opportunities & strengths
- limitations & constraints
- vulnerabilities

## Connect this to Lessig's core concepts:

- ▶ What are “architectures of control?”



└ Connect this to Lessig's core concepts:



Architectures of control:

- Rules/laws
- Norms
- Code
- Prices (markets)

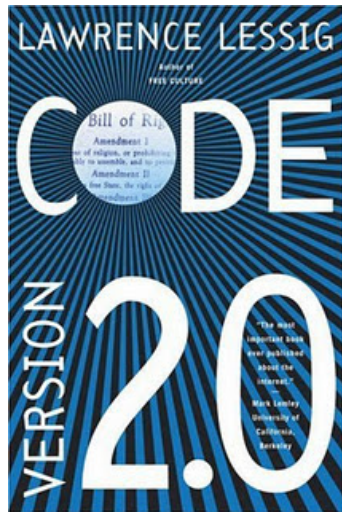
Label in examples:

- specific architectures/manifestations
- opportunities & strengths
- limitations & constraints
- vulnerabilities



## Connect this to Lessig's core concepts:

- ▶ What are “architectures of control?”
- ▶ How does Lessig's example of “Ibex” (the abusive student in his law class) illustrate this idea?



└ Connect this to Lessig's core concepts:

- ▶ What are “architectures of control?”
- ▶ How does Lessig's example of “Ibex” (the abusive student in his law class) illustrate this idea?



Architectures of control:

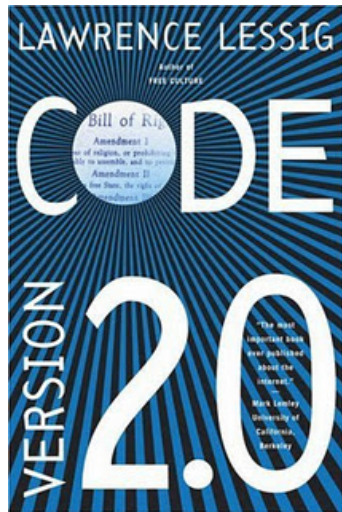
- Rules/laws
- Norms
- Code
- Prices (markets)

Label in examples:

- specific architectures/manifestations
- opportunities & strengths
- limitations & constraints
- vulnerabilities

## Connect this to Lessig's core concepts:

- ▶ What are “architectures of control?”
- ▶ How does Lessig's example of “Ibex” (the abusive student in his law class) illustrate this idea?
- ▶ Lessig argues that architectures of control are unavoidable. Why?



└ Connect this to Lessig's core concepts:

Architectures of control:

- Rules/laws
- Norms
- Code
- Prices (markets)

Label in examples:

- specific architectures/manifestations
- opportunities & strengths
- limitations & constraints
- vulnerabilities

- What are “architectures of control?”
- How does Lessig's example of “Ibex” (the abusive student in his law class) illustrate this idea?
- Lessig argues that architectures of control are unavoidable. Why?



## Connect this to Lessig's core concepts:

- ▶ What are “architectures of control?”
- ▶ How does Lessig's example of “Ibex” (the abusive student in his law class) illustrate this idea?
- ▶ Lessig argues that architectures of control are unavoidable. Why?
- ▶ w/r/t architectures, Lessig says “limitations can be freedoms.” What does this mean?



└ Connect this to Lessig's core concepts:

- ▶ What are “architectures of control?”
- ▶ How does Lessig's example of “Ibex” (the abusive student in his law class) illustrate this idea?
- ▶ Lessig argues that architectures of control are unavoidable. Why?
- ▶ w/r/t architectures, Lessig says “limitations can be freedoms.” What does this mean?



### Architectures of control:

- Rules/laws
- Norms
- Code
- Prices (markets)

### Label in examples:

- specific architectures/manifestations
- opportunities & strengths
- limitations & constraints
- vulnerabilities

**What sorts of conflicts arose on WELL?**

**Which (if any) of these conflicts seem particularly likely online? Why (not)?**

**What sorts of resolutions were reached? How?**



└ Back to the WELL...

What sorts of conflicts arose on WELL?

Which (if any) of these conflicts seem particularly likely online? Why (not)?

What sorts of resolutions were reached? How?



My old slide from Friday. Didn't discuss then. Framing questions for next activity, in which I'm going to ask you to tackle some of this (using Lessig's ideas) in relation to what you read about the WELL (Hafner).

## Critical analysis in groups of four:

- ▶ Identify architectures of control in the WELL (rules/laws, norms, code, prices, etc.)
- ▶ Analyze at least two strengths, freedoms, or opportunities afforded by the WELL.
- ▶ Analyze at least two weaknesses, constraints, and threats afforded by the WELL.
- ▶ Create at least two proposals to help the WELL managers avoid abuses such as those perpetrated by “Mandel.”
- ▶ Prepare to discuss your analysis and proposals with the full class.



### └ Critical analysis in groups of four:

- ▶ Identify architectures of control in the WELL (rules/laws, norms, code, prices, etc.)
- ▶ Analyze at least two strengths, freedoms, or opportunities afforded by the WELL.
- ▶ Analyze at least two weaknesses, constraints, and threats afforded by the WELL.
- ▶ Create at least two proposals to help the WELL managers avoid abuses such as those perpetrated by “Mandel.”
- ▶ Prepare to discuss your analysis and proposals with the full class.



# Big takeaways?

Online Communities & Crowds

2014-09-29

└ Big takeaways?

Big takeaways?

## Big takeaways?

**Code (like most architectures of control) is made by people.**

└ Big takeaways?

## Big takeaways?

**Code (like most architectures of control) is made by people.**

**Architectures encode values and choices.**

└ Big takeaways?

Code (like most architectures of control) is made by people.

Architectures encode values and choices.



## Big takeaways?

**Code (like most architectures of control) is made by people.**

**Architectures encode values and choices.**

**“Limitations” can be features. “Features” can also be limitations.**

## Big takeaways?

**Code (like most architectures of control) is made by people.**

**Architectures encode values and choices.**

**“Limitations” can be features. “Features” can also be limitations.**

**Different regimes offer different degrees of “amenability to control” with real consequences.**

Code (like most architectures of control) is made by people.

Architectures encode values and choices.

“Limitations” can be features. “Features” can also be limitations.

Different regimes offer different degrees of “amenability to control” with real consequences.

## Next time: back to the commons!

- ▶ Reading: Selections from Mancur Olson's *The Logic of Collective Action*.
- ▶ Reading: Elinor Ostrom brief 2-pager "Understanding Collective Action."
- ▶ Consider: In what ways are communities like the WELL similar/different from more traditional examples of collective action or commons?

Next time: back to the commons!

- ▶ Reading: Selections from Mancur Olson's *The Logic of Collective Action*.
- ▶ Reading: Elinor Ostrom brief 2-pager "Understanding Collective Action."
- ▶ Consider: In what ways are communities like the WELL similar/different from more traditional examples of collective action or commons?

## Next time: back to the commons!

- ▶ Reading: Selections from Mancur Olson's *The Logic of Collective Action*.
- ▶ Reading: Elinor Ostrom brief 2-pager "Understanding Collective Action."
- ▶ Consider: In what ways are communities like the WELL similar/different from more traditional examples of collective action or commons?

**Thanks and see you soon!**

Next time: back to the commons!

- ▶ Reading: Selections from Mancur Olson's *The Logic of Collective Action*.
- ▶ Reading: Elinor Ostrom brief 2-pager "Understanding Collective Action."
- ▶ Consider: In what ways are communities like the WELL similar/different from more traditional examples of collective action or commons?

Thanks and see you soon!