1. The Cloud
   - Collection of machines under the control of a single company, connected into a datacenter network
   - Let's us change more things about the underlying infrastructure
   - Backs much of what we do on the Internet today
   - Use Cases (for this lecture)
     - Storing data. Lots of different types.
     - Computing over that data
     - Renting out machines for others (cloud computing)

2. Datacenter Networks
   - Collections of physical machines on racks; network connections within a rack are very fast
   - Datacenter network = many racks connected together via a scalable and redundant topology (example in lecture: a "clos" topology, where all of the racks have top-of-rack switches; each top-of-rack switch in a cluster connects to the same set of aggregate switches, and each aggregate switch connects to the same set of core switches.
   - Multi-path routing lets us load-balance traffic across multiple paths at the same time, but have to be careful about how we do this
     - Round-robin'ing every packet has negative effects on TCP
     - Better to keep each flow on the same path throughout, can do this without much storage by hashing the flow identifier and using the output to dictate the path
     - If we want to be congestion-aware in this process, likely need a centralized controller

3. Centralized control
   - Such control also lets the datacenter move VMs around if needed (e.g., to change the flow of traffic in the network). Requires another layer of indirection via naming

4. Discussion
   - Datacenters vs. CDNs
     - Both under single administrative control, both largely abstracted away from end users
     - CDN is for a different purpose, concerned with global distribution, has to contend with the Internet in a way that datacenter networks don't
   - Datacenters vs. the Internet
     - Centralized control means we can change things. Topology design, routing, addressing, etc. all look different.
     - Decisions to make in a datacenter
- Performance: are network transfers within the datacenter involved? Can we minimize them?
- What computation should exist in the datacenter vs. client-side?
- What if machines fail? Do we replicate data? How do we keep it consistent (answers after spring break!)
- What about cost? Both monetary and environmental; datacenters require a lot of energy (and sometimes water) to cool