Data Center Networks

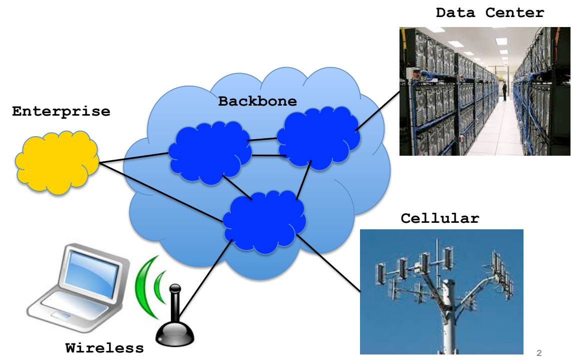
Jennifer Rexford

COS 461: Computer Networks

Lectures: MW 10-10:50am in Architecture N101

http://www.cs.princeton.edu/courses/archive/spr12/cos461/

Networking Case Studies



Cloud Computing

Cloud Computing

- Elastic resources
  - Expand and contract resources
  - Pay-per-use
  - Infrastructure on demand
- Multi-tenancy
  - Multiple independent users
  - Security and resource isolation
  - Amortize the cost of the (shared) infrastructure
- Flexible service management

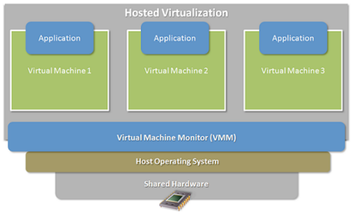
Cloud Service Models

- Software as a Service
  - Provider licenses applications to users as a service
  - E.g., customer relationship management, e-mail, ..
  - Avoid costs of installation, maintenance, patches, …
- Platform as a Service
  - Provider offers platform for building applications
  - E.g., Google's App-Engine
  - Avoid worrying about scalability of platform

Cloud Service Models

- Infrastructure as a Service
  - Provider offers raw computing, storage, and network
  - E.g., Amazon's Elastic Computing Cloud (EC2)
  - Avoid buying servers and estimating resource needs

Enabling Technology: Virtualization

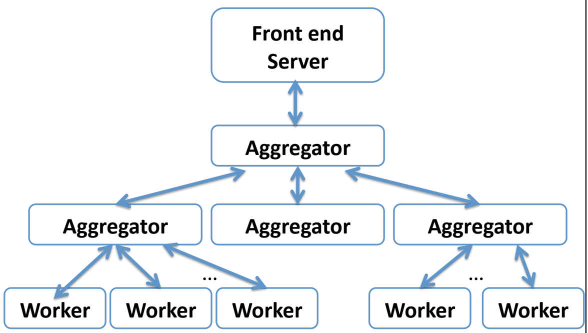


- Multiple virtual machines on one physical machine
- Applications run unmodified as on real machine
- VM can migrate from one computer to another

Multi-Tier Applications

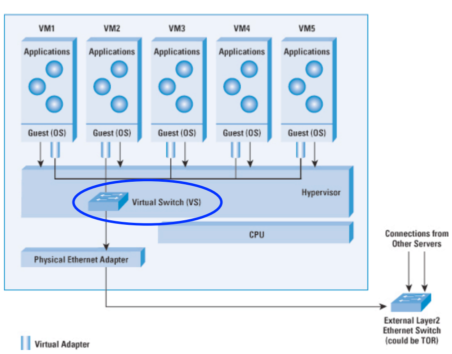
- Applications consist of tasks
  - Many separate components
  - Running on different machines
- Commodity computers
  - Many general-purpose computers
  - Not one big mainframe
  - Easier scaling

Multi-Tier Applications



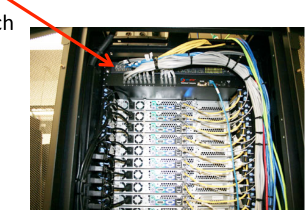
Data Center Network

Virtual Switch in Server

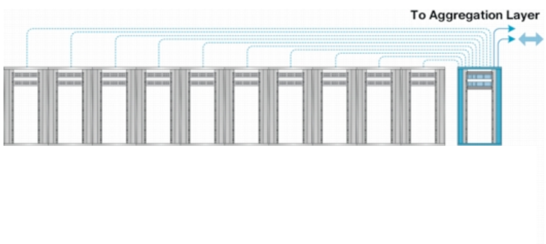


Top-of-Rack Architecture

- Rack of servers
  - Commodity servers
  - And top-of-rack switch
- Modular design
  - Preconfigured racks
  - Power, network, and storage cabling



Aggregate to the Next Level



13

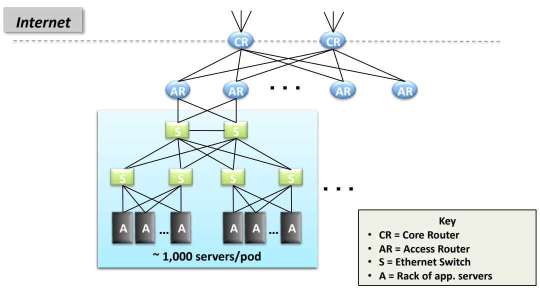
Modularity, Modularity, Modularity

- Containers

- Many containers

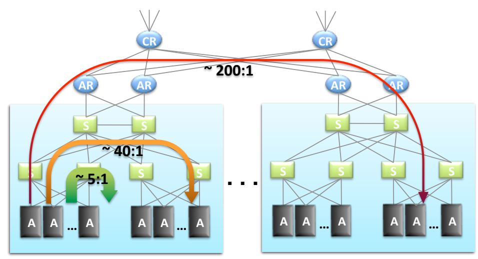
14

Data Center Network Topology



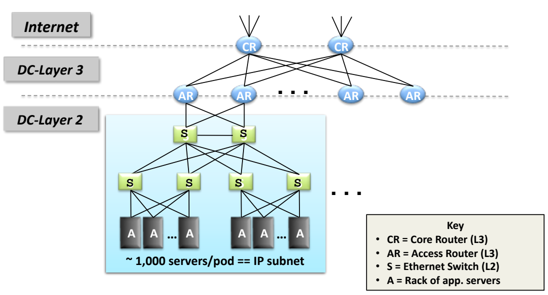
15

Capacity Mismatch



16

Data-Center Routing



17

Reminder: Layer 2 vs. Layer 3

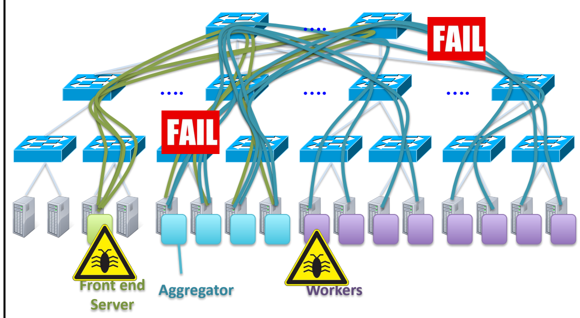
- Ethernet switching (layer 2)
  - Cheaper switch equipment
  - Fixed addresses and auto-configuration
  - Seamless mobility, migration, and failover
- IP routing (layer 3)
  - Scalability through hierarchical addressing
  - Efficiency through shortest-path routing
  - Multipath routing through equal-cost multipath
- So, like in enterprises…
  - Connect layer-2 islands by IP routers

18

Case Study: Performance Diagnosis in Data Centers

http://www.eecs.berkeley.edu/~minlanyu/writeup/nsdi11.pdf

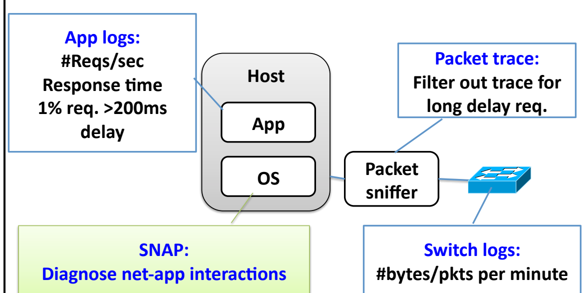
Applications Inside Data Centers



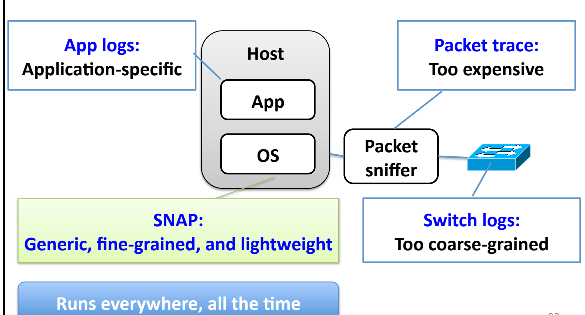
Challenges of Datacenter Diagnosis

- Multi-tier applications
  – Hundreds of application components
  – Tens of thousands of servers
- Evolving applications
  – Add new features, fix bugs
  – Change components while app is still in operation
- Human factors
  – Developers may not understand network well
  – Nagle's algorithm, delayed ACK, etc.

Diagnosing in Today's Data Center



Problems of Different Logs



TCP Statistics

- Instantaneous snapshots
  – #Bytes in the send buffer
  – Congestion window size, receiver window size
  – Snapshots based on random sampling
- Cumulative counters
  – #FastRetrans, #Timeout
  – RTT estimation: #SampleRTT, #SumRTT
  – RwinLimitTime
  – Calculate difference between two polls

Identifying Performance Problems

- **Sender App** – Not any other problems
- **Send Buffer** – Send buffer is almost full — Sampling
- **Network**
  - #Fast retransmission
  - #Timeout
  - (Direct measure)
- **Receiver**
  - RwinLimitTime (Direct measure)
  - Delayed ACK (Inference)
  - *diff(SumRTT)/diff(SampleRTT) > MaxDelay*

25

SNAP Architecture

At each host for every connection

Collect data

Direct access to OS

- Polling per-connection statistics:
  - Snapshots (#bytes in send buffer)
  - Cumulative counters (#FastRestrans)
- Adaptive tuning of polling rate

26

SNAP Architecture

At each host for every connection

Collect data → Performance Classifier

Classifying based on the life of data transfer

- Algorithms for detecting performance problems
- Based on direct measurement in the OS

27

SNAP Architecture

At each host for every connection

Collect data → Performance Classifier → Cross-connection correlation

Direct access to data center configurations

- Input
  - Topology, routing information
  - Mapping from connections to processes/apps
- Correlate problems across connections
  - Sharing the same switch/link, app code

SNAP Deployment

- Production data center
  - 8K machines, 700 applications
  - Ran SNAP for a week, collected petabytes of data
- Identified 15 major performance problems
  - Operators: Characterize key problems in data center
  - Developers: Quickly pinpoint problems in app software, network stack, and their interactions

29

Characterizing Perf. Limitations

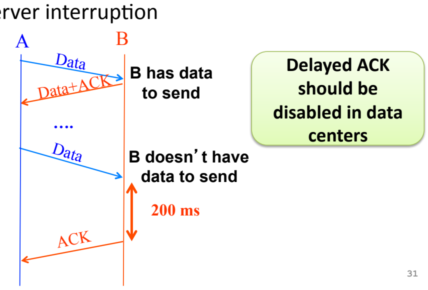
#Apps that are limited for > 50% of the time

- **Sender App** — 551 Apps
  - Bottlenecked by CPU, disk, etc.
  - Slow due to app design (small writes)
- **Send Buffer** — 1 App
  - **Send buffer not large enough**
- **Network** — 6 Apps
  - **Fast retransmission**
  - **Timeout**
- **Receiver**
  - 8 Apps — **Not reading fast enough (CPU, disk, etc.)**
  - 144 Apps — **Not ACKing fast enough (Delayed ACK)**

30

Delayed ACK

- Delayed ACK caused significant problems
  - Delayed ACK was used to reduce bandwidth usage and server interruption



31

Diagnosing Delayed ACK with SNAP

- Monitor at the right place
  - Scalable, low overhead data collection at all hosts
- Algorithms to identify performance problems
  - Identify delayed ACK with OS information
- Correlate problems across connections
  - Identify the apps with significant delayed ACK issues
- Fix the problem with operators and developers
  - Disable delayed ACK in data centers

32

Conclusion

- Cloud computing
  - Major trend in IT industry
  - Today's equivalent of factories
- Data center networking
  - Regular topologies interconnecting VMs
  - Mix of Ethernet and IP networking
- Modular, multi-tier applications
  - New ways of building applications
  - New performance challenges

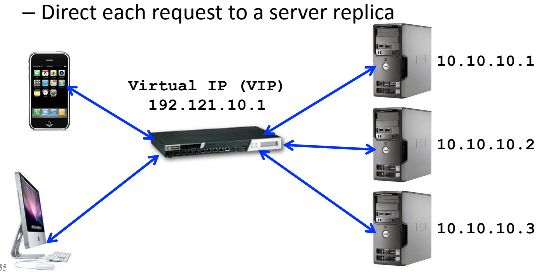
33

Load Balancing

34

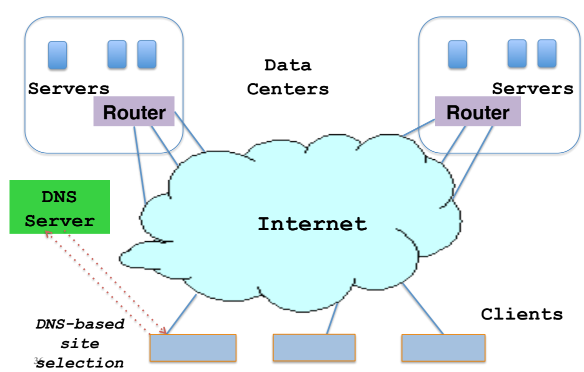
Load Balancers

- Spread load over server replicas
  - Present a single public address (VIP) for a service
  - Direct each request to a server replica

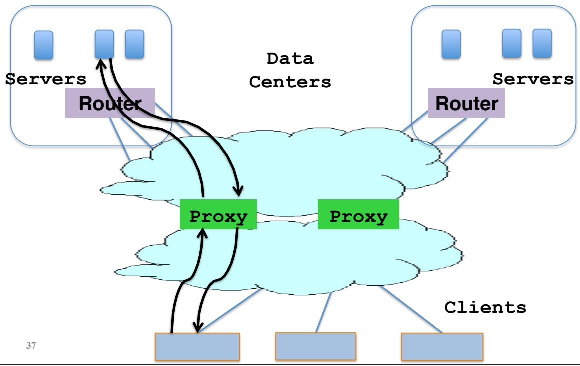


35

Wide-Area Network



Wide-Area Network: Ingress Proxies



37