Tutorial Part I Outline

- Introduction to Data Center Networks
- Data Centers Networks Requirements
- Data Center Network Topologies
  - The Real World
  - Research and Academic Proposals
Introduction to Data Center Networks
Cloud and Data Centers

- Cloud: The Next Generation of Large-Scale Computing
  - Infrastructure as a Service (IaaS)
  - Platform as a Service (PaaS)
  - Software as a Service (SaaS)
- Cloud needs support of large-scale elastic data centers
  - Massive number of servers
  - Massive amount of storage
  - Orchestrated together with a Data Center Network
  - Virtual Machine support
- Example: Google and its Services
Inside Google’s Data Center
A Server Room in Council Bluffs, IA Data Center
Inside Google’s Data Center

A Campus Network Room in Council Bluffs, IA Data Center
Inside Google’s Data Center
Central Cooling Plant in Google’s Douglas County, GA Data Center
Data Center Application Requirements

• Data centers typically run two types of applications
  – outward facing (e.g., serving web pages to users)
  – internal computations (e.g., MapReduce for web indexing)

• Workloads often unpredictable:
  – Multiple services run concurrently within a DC
  – Demand for new services may spike unexpectedly
    • Spike of demands for new services mean success!
    • But this is when success spells trouble (if not prepared)!

• Failures of servers are the norm
Data Center Costs [Greenberg 2008]

- **Total cost varies**
  - upwards of $1/4 B for mega data center
  - server costs dominate
  - network costs significant

- **Long provisioning timescales:**
  - new servers purchased quarterly at best

<table>
<thead>
<tr>
<th>Amortized Cost*</th>
<th>Component</th>
<th>Sub-Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>~45%</td>
<td>Servers</td>
<td>CPU, memory, disk</td>
</tr>
<tr>
<td>~25%</td>
<td>Power infrastructure</td>
<td>UPS, cooling, power distribution</td>
</tr>
<tr>
<td>~15%</td>
<td>Power draw</td>
<td>Electrical utility costs</td>
</tr>
<tr>
<td>~15%</td>
<td>Network</td>
<td>Switches, links, transit</td>
</tr>
</tbody>
</table>
Overall Data Center Design Goal [Greenberg 2008]

• Agility – Any service, Any Server
• Turn the servers into a single large fungible pool
  – Let services “breathe” : dynamically expand and contract their footprint as needed
    • We already see how this is done in terms of Google’s GFS, BigTable, MapReduce
  – Equidistant end-points with non-blocking core
  – Unlimited workload mobility
• Benefits
  – Increase service developer productivity
  – Lower cost
  – Achieve high performance and reliability

• These are the three motivators for most data center infrastructure projects!
Data Center Networks
Requirements
Tutorial Part I Outline

• Introduction to Data Center Networks
  ➢ Data Centers Networks Requirements
• Data Center Network Topologies
  — The Real World
  — Research and Academic Proposals
Data Center Network Requirements

• Uniform high capacity
  – Capacity between servers limited only by their NICs
  – No need to consider topology when adding servers
    • => In other words, high capacity between any two servers regardless of which racks they are located!

• Performance isolation
  – Traffic of one service should be unaffected by others

• Ease of management: “Plug-&-Play” (layer-2 semantics)
  – Flat addressing, so any server can have any IP address
  – Server configuration is the same as in a LAN
  – Legacy applications depending on broadcast must work
Data Center Network Requirements

- Requirements for scalable, easily manageable, fault tolerant and efficient Data Center Networks (DCN):
  - R1: Any VM may migrate to any physical machine without a change in its IP address
  - R2: An administrator should not need to configure any switch before deployment
  - R3: Any end host should efficiently communicate with any other end hosts through any available paths
  - R4: No forwarding loops
  - R5: Failure detection should be rapid and efficient

- Implication on network protocols:
  - A single layer2 fabric for entire data center (R1&R2)
  - Mac forwarding tables with hundreds of thousands entries (R3)
  - Efficient routing protocols which disseminate topology changes quickly to all points (R5)
Data Center Network Topologies
Tutorial Part I Outline

• Introduction to Data Center Networks
• Data Centers Networks Requirements
  ➢ Data Center Network Topologies
    – The Real World
    – Research and Academic Proposals
Tutorial Part I Outline

• Introduction to Data Center Networks
• Data Centers Networks Requirements
  ➢ Data Center Network Topologies
    ✓ The Real World
      – Research and Academic Proposals
Clos Networks

- Charles Clos 1953
- Circuit Switching
- Non-blocking
- Multistage

Source: http://upload.wikimedia.org/wikipedia/en/9/9a/Closnetwork.png
Clos Networks

- 3-stage Clos
Clos Networks

- Folded Clos: Leaf and Spine

Leaf switch: aggregate traffic from servers to the core of the network
Spine switch: core network
Leaf-spine network: a leaf is one hop from another
An Example Clos Network

40G Leaf/Spine

Source: https://s3.amazonaws.com/bradhedlund2/2012/40G-10G-leaf-spine/clos-40G.png
Typical Data Center Topology Today

• **Data Center Network topology:**
  – End hosts connects to top of rack (ToR) switches
  – ToR switches contains 48 GigE ports and up to 4 10 GigE uplinks
  – ToR switches connect to one or more end of row (EoR) switches

• **Forwarding:**
  – Layer 3 approach:
    • Assign IP addresses to hosts hierarchically based on their directly connected switch.
    • Use standard intra-domain routing protocols, eg. OSPF.
    • Large administration overhead
Typical Data Center Topology Today

- Layer 2 approach:
  - Forwarding on flat MAC addresses
  - Less administrative overhead
  - Bad scalability
  - Low performance

- Middle ground between layer 2 and layer 3:
  - VLAN
  - Feasible for smaller scale topologies
  - Resource partition problem

- End host virtualization:
  - Needs to support large addresses and VM migrations (e.g. vMotion)
  - In layer 3 fabric, migrating the VM to a different switch changes VM’s IP address
  - In layer 2 fabric, migrating VM incurs scaling ARP and performing routing/forwarding on millions of flat MAC addresses.
An Example from Cisco’s Recommendation

Internet

Core Router

Access Router

Data Center Layer 3

Layer 2

Load Balancer

L2 Switches

TOR Switches

Server Racks

Layer 3
An Example from Cisco’s Recommendation

• Hierarchical Network: 1+1 redundancy
  • Equipment higher in the hierarchy handles more traffic, more expensive, more efforts made at availability ➔ scale-up design
  • Servers connect via 1 Gbps UTP to Top of Rack switches
• Other links are mix of 1G, 10G; fiber, copper
Data Center Fabric

• Abstraction of the data center network into a single orchestrated entity
  – Definition depends on different vendors, defined to their convenience

• Typical attributes of a fabric: Open (standards based), Scalable (cost and power), Intelligent (auto discovery), Secure (isolation and virtual network support), Resilient (fault-tolerant, stateless), Flexible (auto provisioning), Modular
Data Center Topologies: Research and Academic
Tutorial Part I Outline

- Introduction to Data Center Networks
- Data Centers Networks Requirements
  - Data Center Network Topologies
  - The Real World
  - Research and Academic Proposals
DCN Topology Taxonomy

Data Center Networks

Fixed Topology
- Tree-based
  - Basic tree
  - Fat tree
  - Clos network
- Recursive
  - DCell
  - BCube
  - MDCube
  - FiConn

Flexible Topology
- Fully Optical
  - OSA
- Hybrid
  - c-Through
  - Helios
DCN Topology Taxonomy

Data Center Networks

Fixed Topology
- Tree-based
  - Basic tree
  - Fat tree
  - Clos network
- Recursive
  - DCell
  - BCube
  - MDCube
  - FiConn

Flexible Topology
- Fully Optical
  - OSA
- Hybrid
  - c-Through
  - Helios
Tree-based Topologies

- Single port on each computer
- Scale up by adding more levels of switches or more ports on switches
- Basic tree, fat tree, Clos Network, ...
- Also can be classified as switch-centric topologies
Fat-Tree Topology
Fat-Tree Topology

- Fat-Tree: a special type of Clos Network
  - K-ary fat tree: three-layer topology (edge, aggregation and core)
  - Split fat tree into k pods
  - Each pod consists of \((k/2)^2\) servers & 2 layers of \(k/2\) k-port switches
  - Each edge switch connects to \(k/2\) servers & \(k/2\) aggr. switches
  - Each aggr. switch connects to \(k/2\) edge & \(k/2\) core switches
  - \((k/2)^2\) core switches: each connects to \(k\) pods
  - Each pod supports non-blocking operation among \((k/2)^2\) hosts
  - Each source and destination have \((k/2)^2\) paths
Fat-Tree Topology

- **Fat-Tree Properties**
  - Identical bandwidth at any bisection
  - Each layer has the same aggregated bandwidth

- **Can be built using cheap devices with uniform capacity**
  - Each port supports same speed as end host
  - All devices can transmit at line speed if packets are distributed uniform along available paths

- **Great scalability: k-port switch supports $k^2/4$ servers**
Fat-tree Topology

- Layer 2 switch algorithm: data plane flooding!
- Layer 3 IP routing:
  - shortest path IP routing will typically use only one path despite the path diversity in the topology
  - if using equal-cost multi-path routing at each switch independently and blindly, packet re-ordering may occur; further load may not necessarily be well-balanced
  - Aside: control plane flooding!