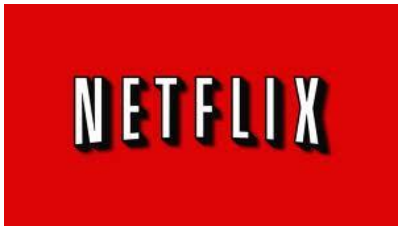
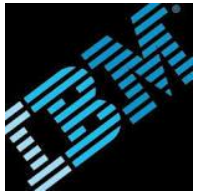
Today's service/company landscape include ...





Today's service/company landscape include ...

Today's service/company landscape include ...

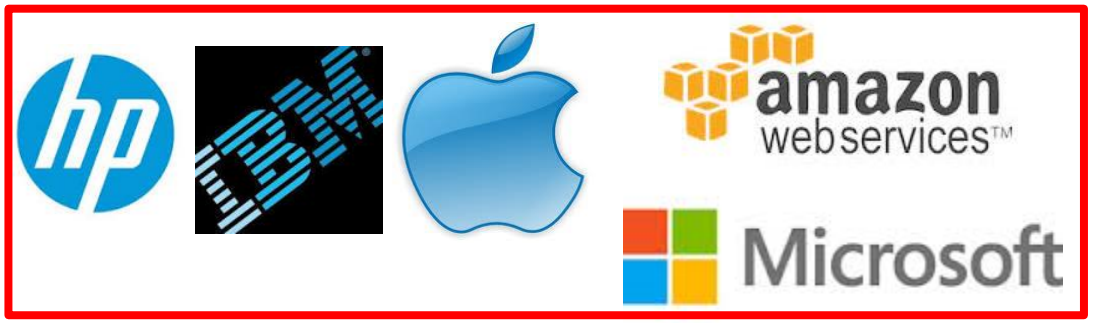
Equipment manufacturers
(also sell services and help
Operate networks)

Today's service/company landscape include ...

Network operators

Equipment manufacturers
(also sell services and help
Operate networks)

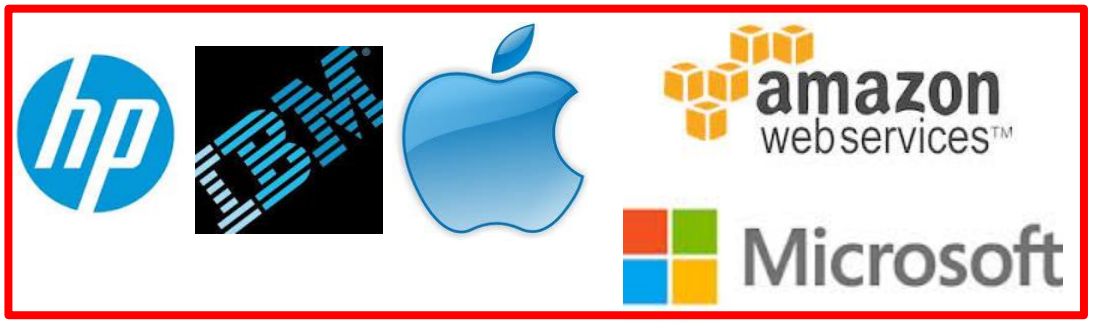
Today's service/company landscape include ...



Enterprise solutions and network service (e.g., data center solutions and cloud providers)

Today's service/company landscape include …

Content delivery networks



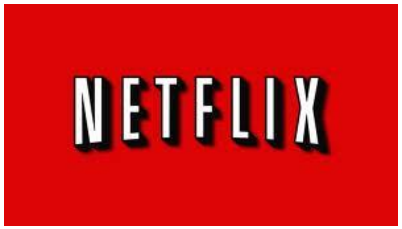
Enterprise solutions and network service (e.g., data center solutions and cloud providers)

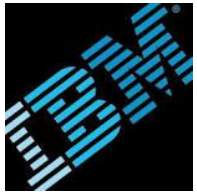
Today's service/company landscape include ...

End user services (e.g., web-based social networks, search, communication, and streaming)

Today's service/company landscape include ...

However, "boundaries" are getting fuzzy …

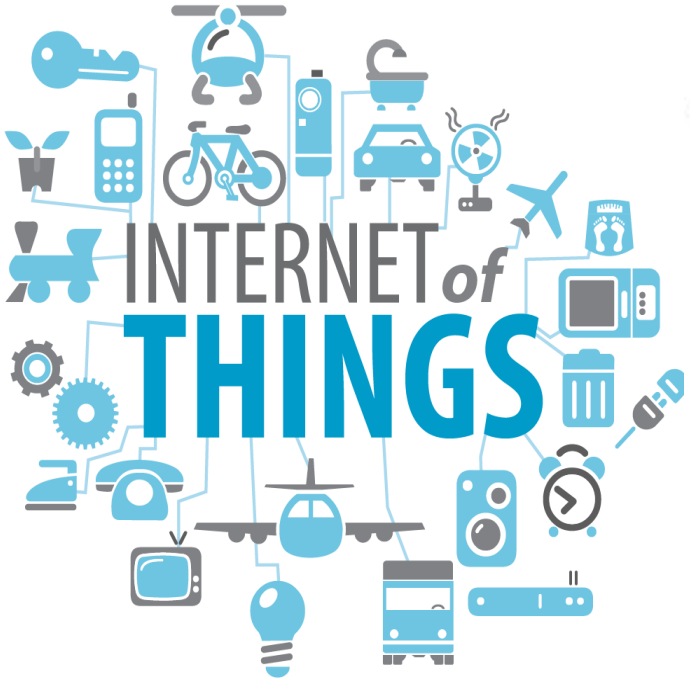


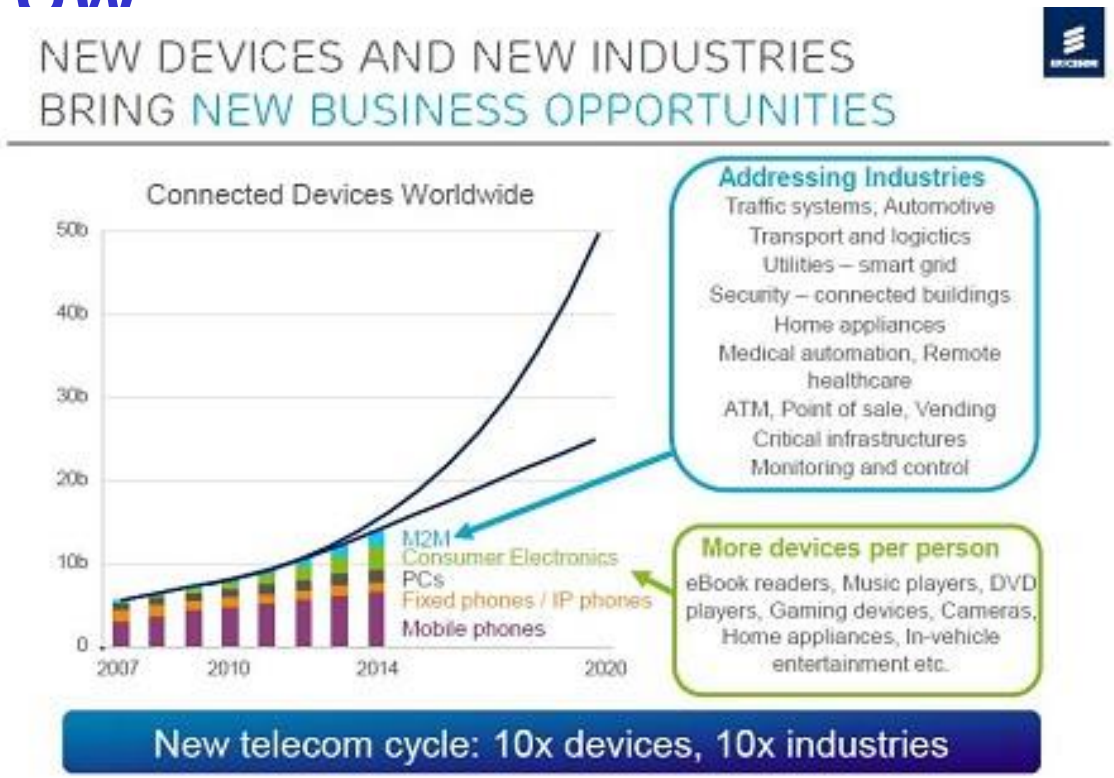


Some common applications today …

- World Wide Web (WWW)
- Remote login (telnet, rlogin, ssh)
- File transfer
- Peer-to-peer file sharing
- Cloud computing/services
- Instant messaging (chat, text messaging, etc.)
- Live and video-on-demand streaming
- Internet phone (Voice-Over-IP)
- Distributed games

… and tomorrow





The 2020 vision

- Everything that can be connected will be connected
  - 50B devices (perhaps more like 500B …)
- IoT and smart cities
  - Machine-to-machine
- High-definition 3D streaming to heterogeneous clients

Scalable Content Delivery

- Use of Internet for content delivery is massive … and becoming more so (e.g., majority of all IP traffic is video streaming content)
- Variety of approaches: HTTP-based Adaptive Streaming (HAS), broadcast/multicast, batching, replication/caching (e.g. CDNs), P2P, peer-assisted, …
- In these slides, we only provide a few high-level examples

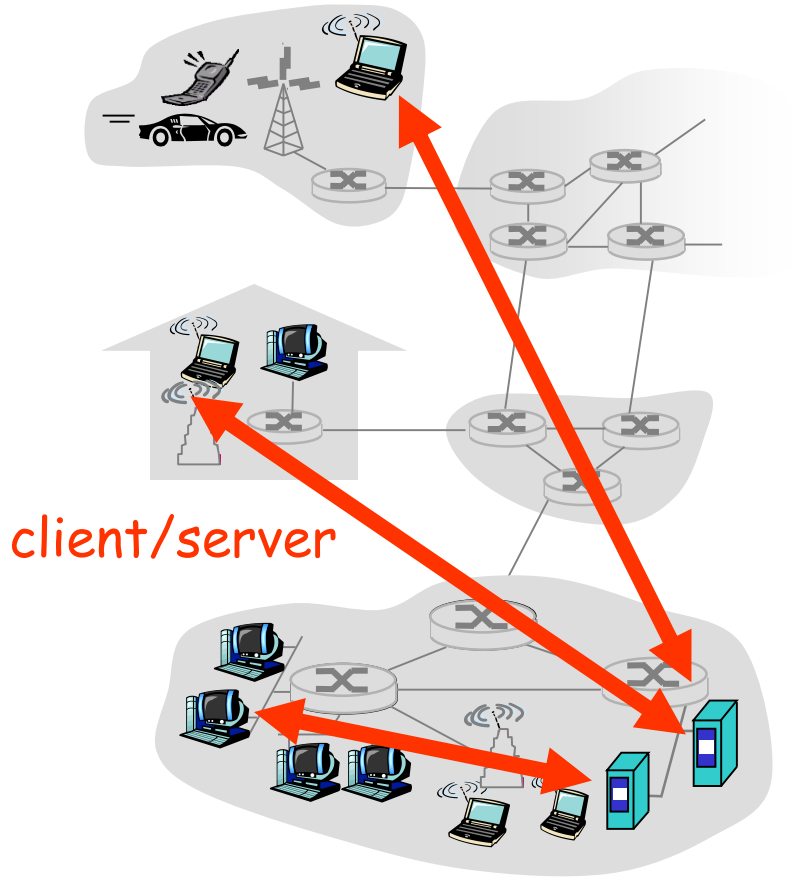
Service models

- Client-server (one-to-one)
- Peer-to-peer (machines can act as both client and server)

- Multicast/broadcast (one-to-many and many-to-many)
  - Application layer, IP-based, and down at the MAC-layer
- Replication: ISP-based caching, CDNs, cloud, and other third-party solutions

Client-server architecture

Client/server model has well-defined roles.



server:

- always-on host
- permanent IP address
- server farms for scaling

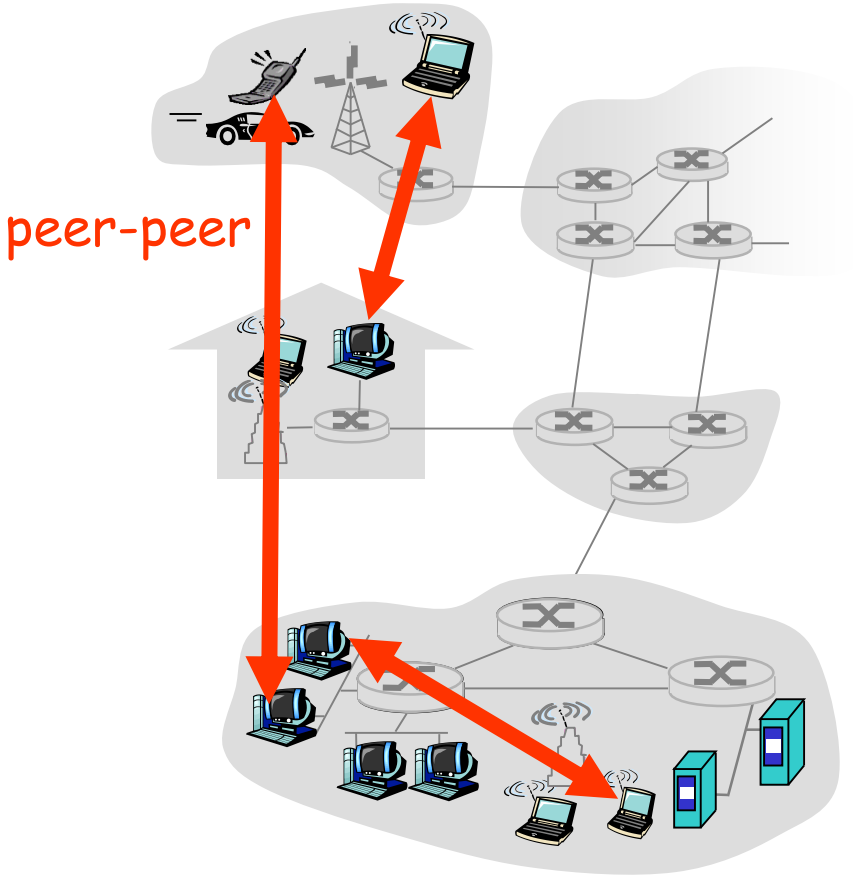
clients:

- communicate with server
- may be intermittently connected
- may have dynamic IP addresses

Pure P2P architecture

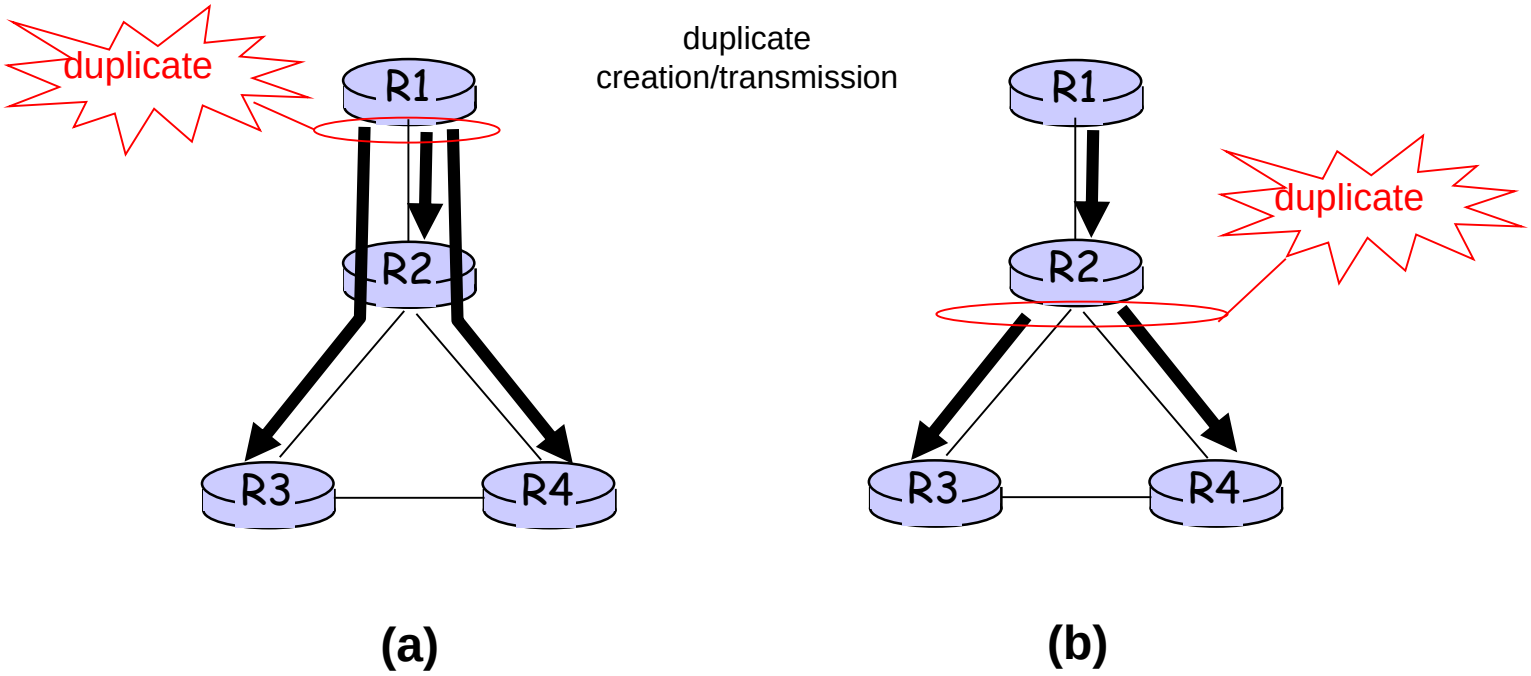
No fixed clients or servers: Each host can act as both client and server at any time

- *no* always-on server
- arbitrary end systems directly communicate
- peers are intermittently connected and change IP addresses



One-to-many delivery

Multicast/Broadcast



Source-duplication versus in-network duplication.
(a) source duplication, (b) in-network duplication

Also, application-layer multicast …

Evolved Multimedia Broadcast/Multicast Service (eMBMS) in LTE-advanced

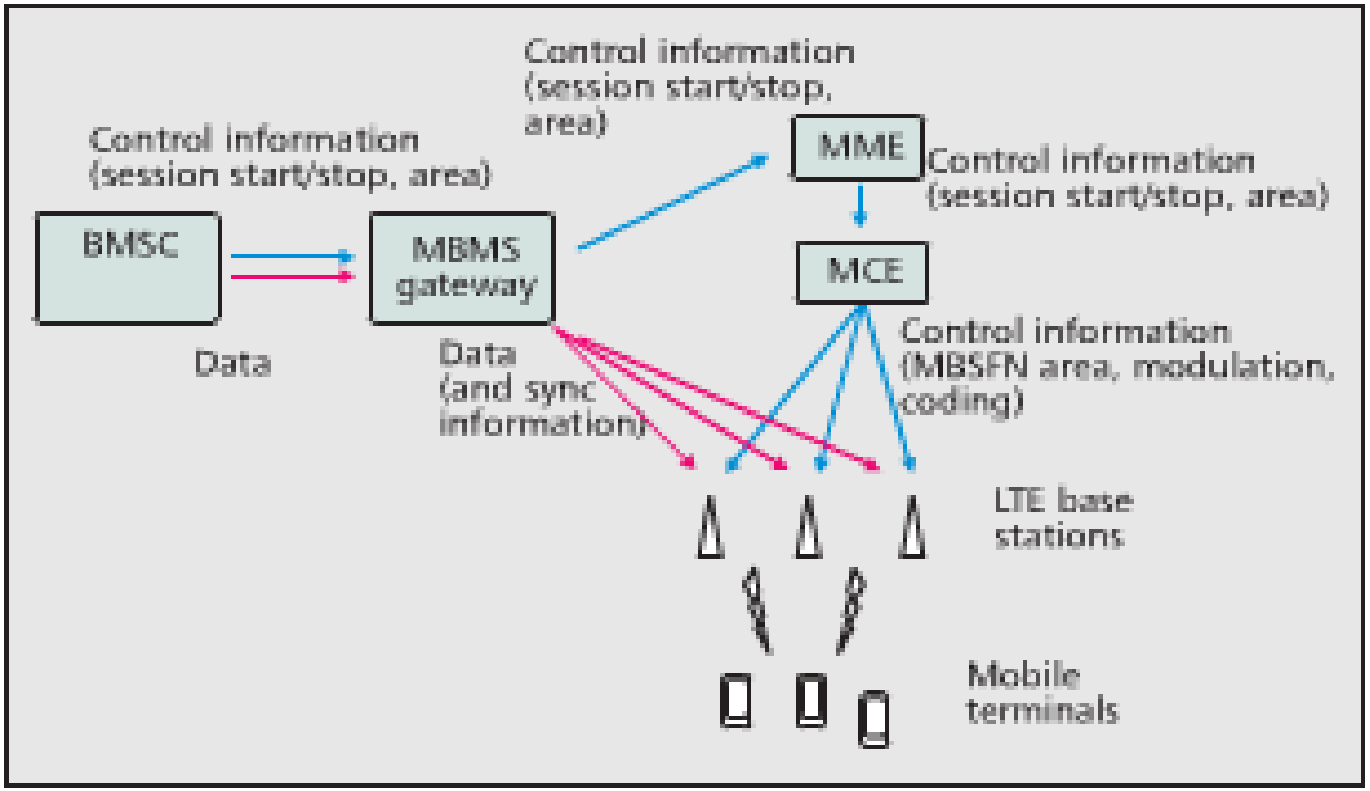


Figure 4. *RAN architecture for SFN across LTE base stations.*

- Separation of control plane and data plane

Image from: Lecompte and Gabin, Evolved Multimedia Broadcast/Multicast Service (eMBMS) in LTE-Advanced: Overview and Rel-11 Enhancements, IEEE Communications Magazine, Nov. 2012.

Evolved Multimedia Broadcast/Multicast Service (eMBMS) in LTE-advanced

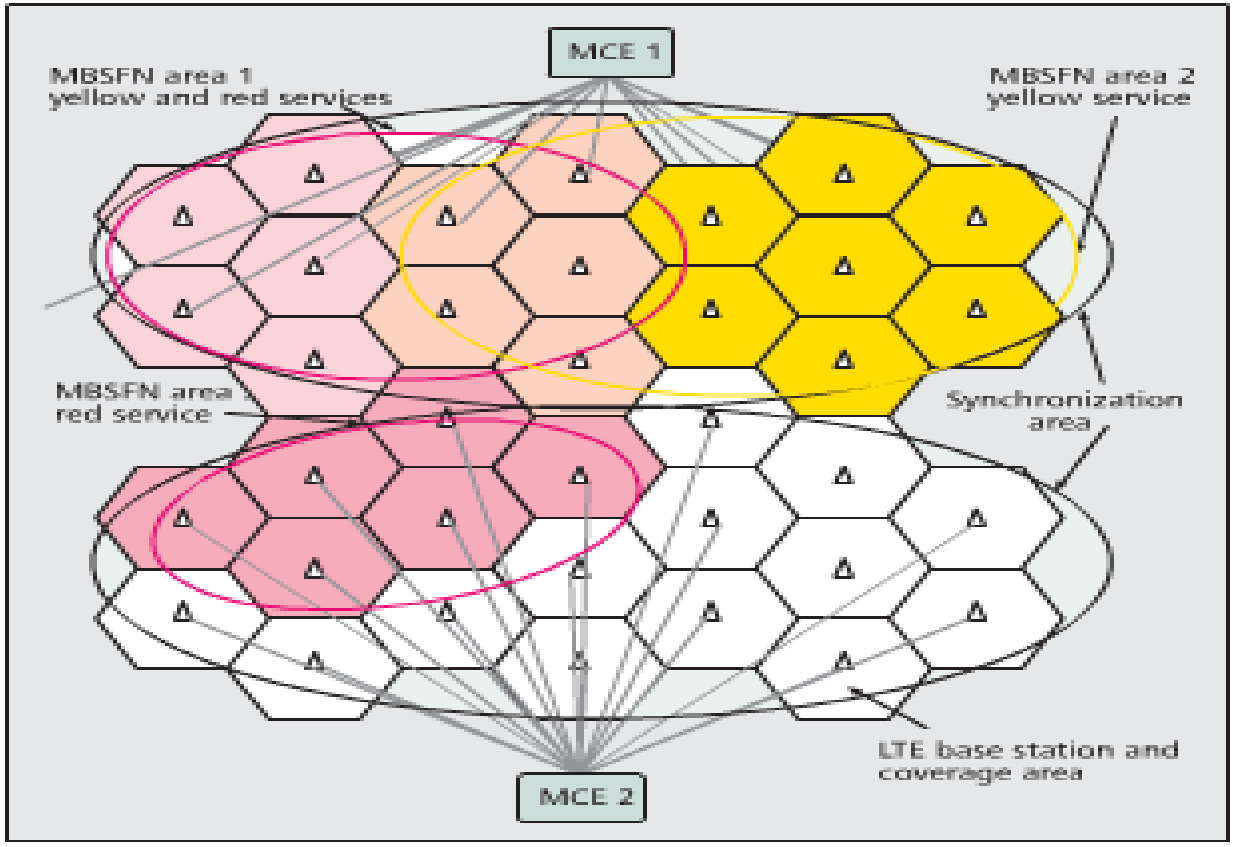


Figure 5. *Example with two MBMS services with different services areas.*

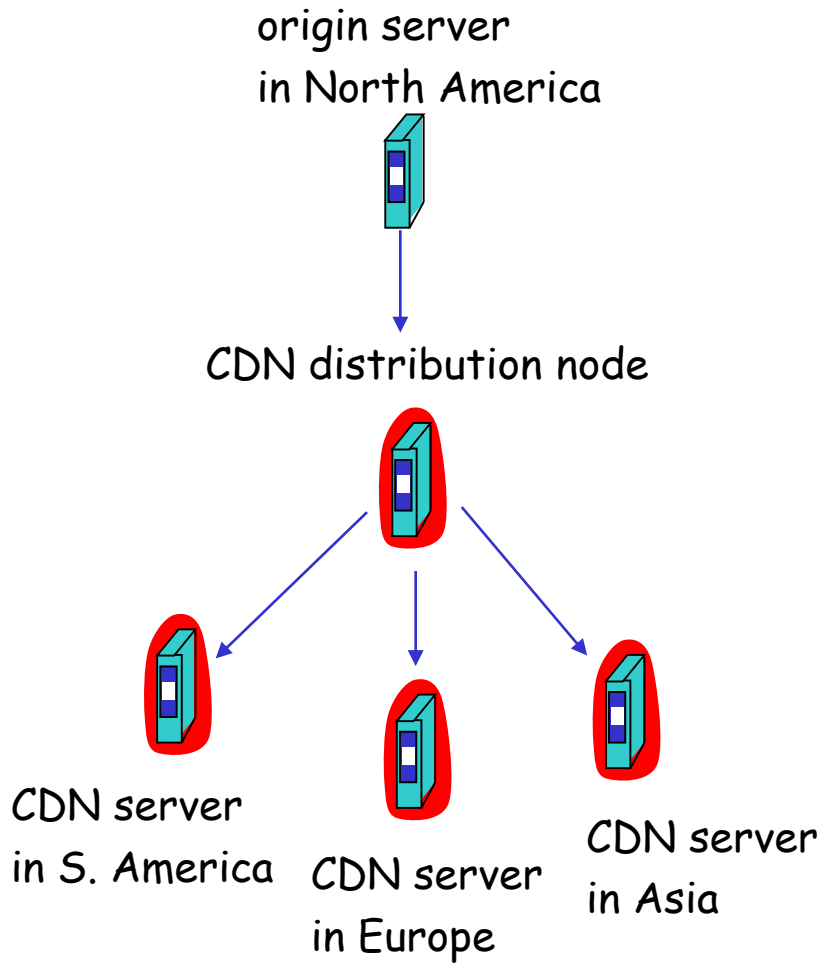
- MBMSFN and use of services areas

Image from: Lecompte and Gabin, Evolved Multimedia Broadcast/Multicast Service (eMBMS) in LTE-Advanced: Overview and Rel-11 Enhancements, IEEE Communications Magazine, Nov. 2012.

Content distribution networks (CDNs)

Content replication

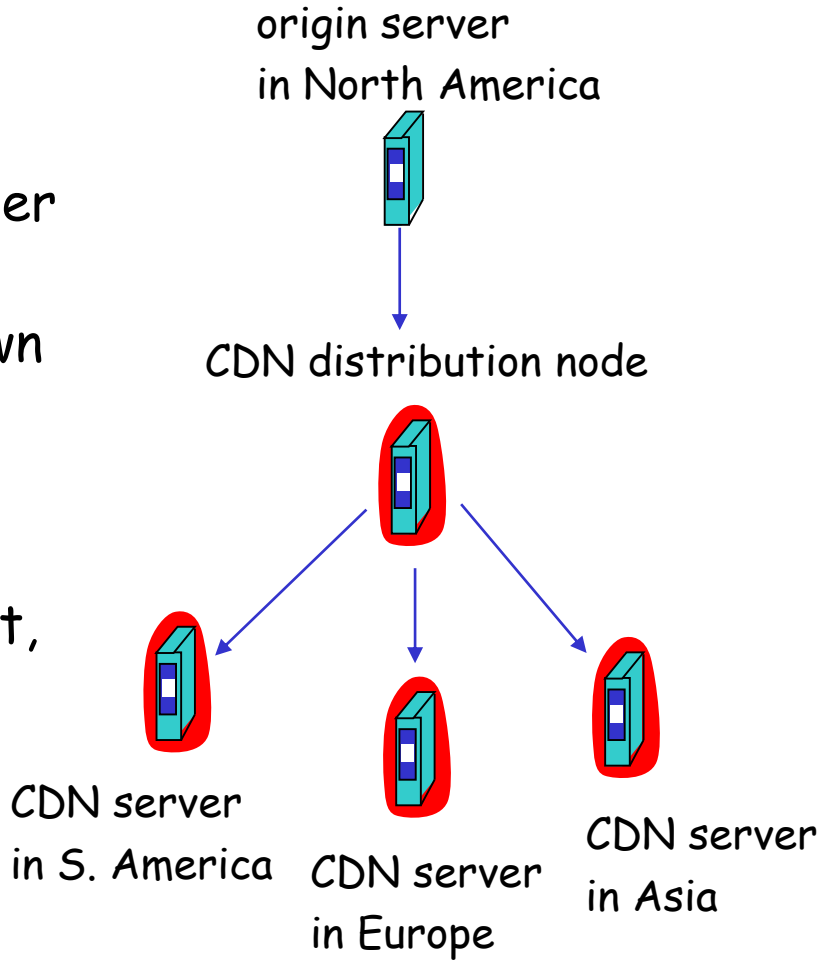
- replicate content at hundreds of servers throughout Internet (often in edge/access network)
- content "close" to user reduce impairments (loss, delay) of sending content over long paths



Content distribution networks (CDNs)

Content replication

- CDN (e.g., Akamai, Limewire) customer is the content provider (e.g., CNN)
- Other companies build their own CDN (e.g., Google)
- CDN replicates customers' content in CDN servers.
- When provider updates content, CDN updates servers



TDDD36: Peer-to-peer

The Akamai CDN

Akamai slides borrowed from Dr. M. Kasbekar's keynote at GreenMetrics '10

- Akamai deployment
  - 66,000+ servers
  - 120+K CPUs, 400+K disks
  - Located in 1000+ datacenters in rented space
- Akamai Traffic
  - Petabytes delivered each day
  - Recent traffic record of 3.5 Tbps
  - Growing extremely fast

Energy efficiency of this massively distributed platform

- Areas that we can't control
  - Datacenter PUE
  - Energy source
  - Hardware's energy efficiency
  - Growth in the web traffic
- Areas that we do control
  - Server software efficiency
  - Operational practices around traffic management
- Currently, datacenters are inefficient
  - PUE of 1.7-3.0
  - Any savings in the reduction of machine count are amplified by the same factor

Trends

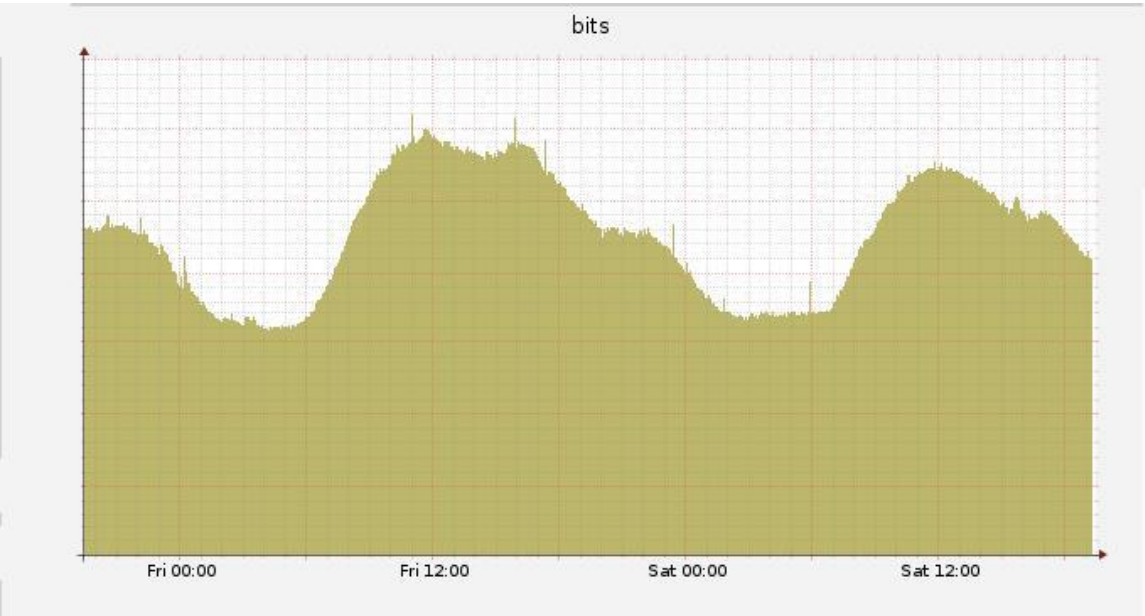
General web traffic

against

- Software downloads
- Transactional application acceleration
- Large-footprint long-tail content
- Media downloads and full movie delivery

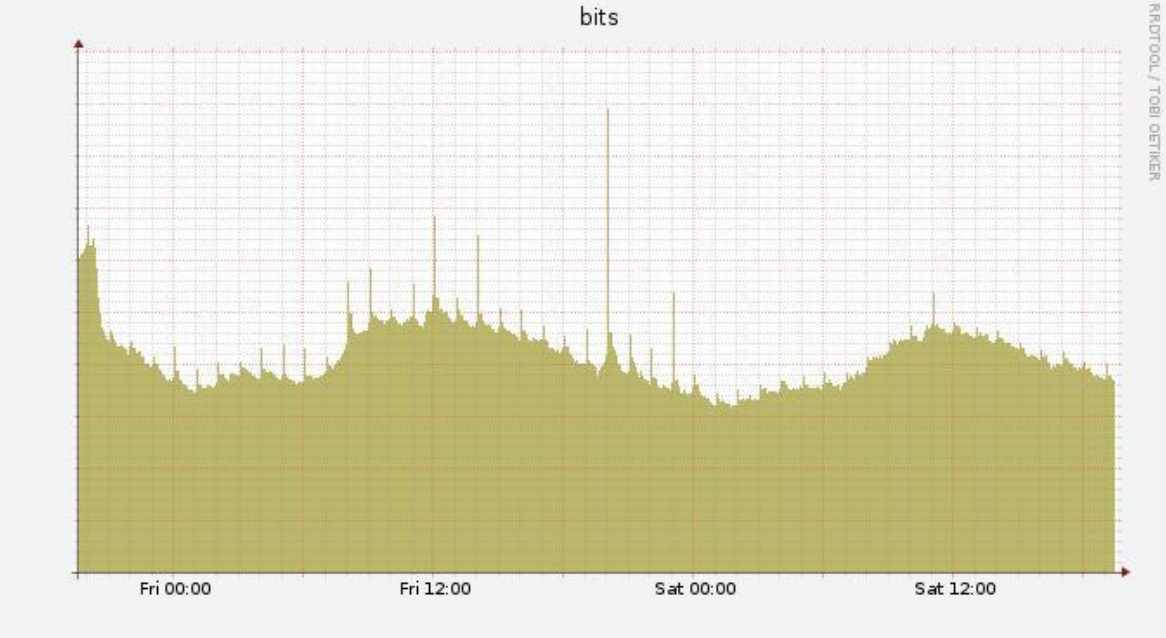
General web traffic

- Selling point: High performance for cacheable content
- Content type: html, images, stylesheets, javascripts
- Peak to valley ratio = 1.8
- Peak traffic at 1000-1600 EST



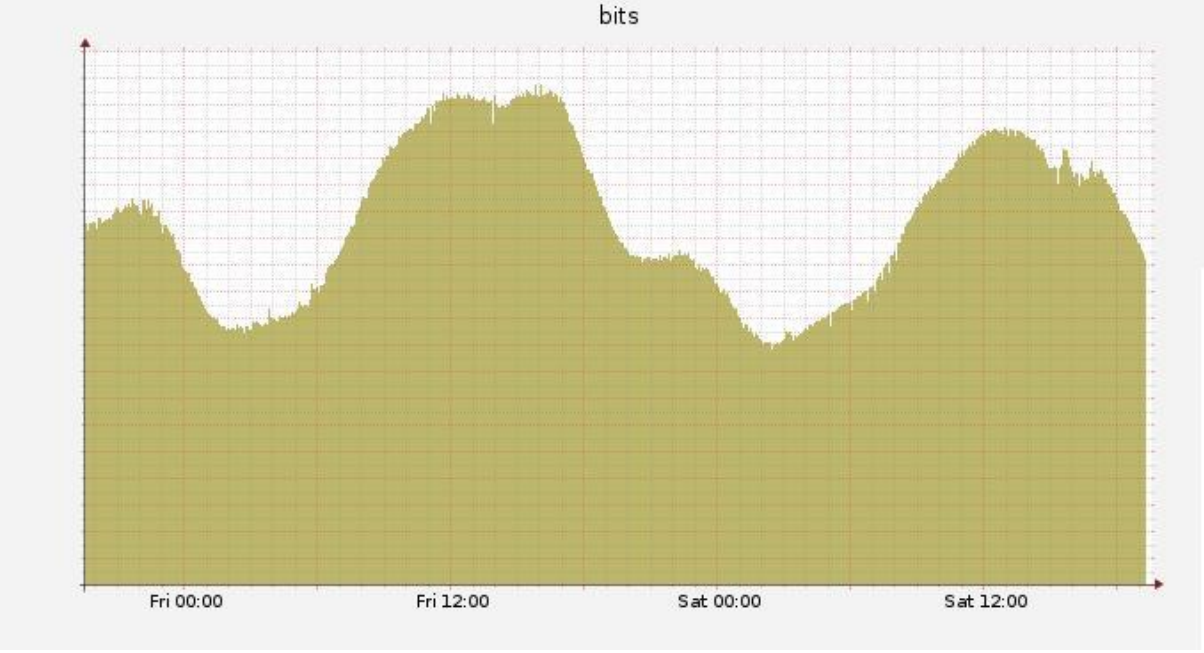
Software downloads

- Selling point: software downloads and frequent updates
- Requirements: cheap delivery and origin offload
- Content type: very large files, high cache hit rates
- Peak to valley ratio = 1.4
- Peak traffic hr: early in the morning, but fairly flat



Large-footprint long-tail traffic

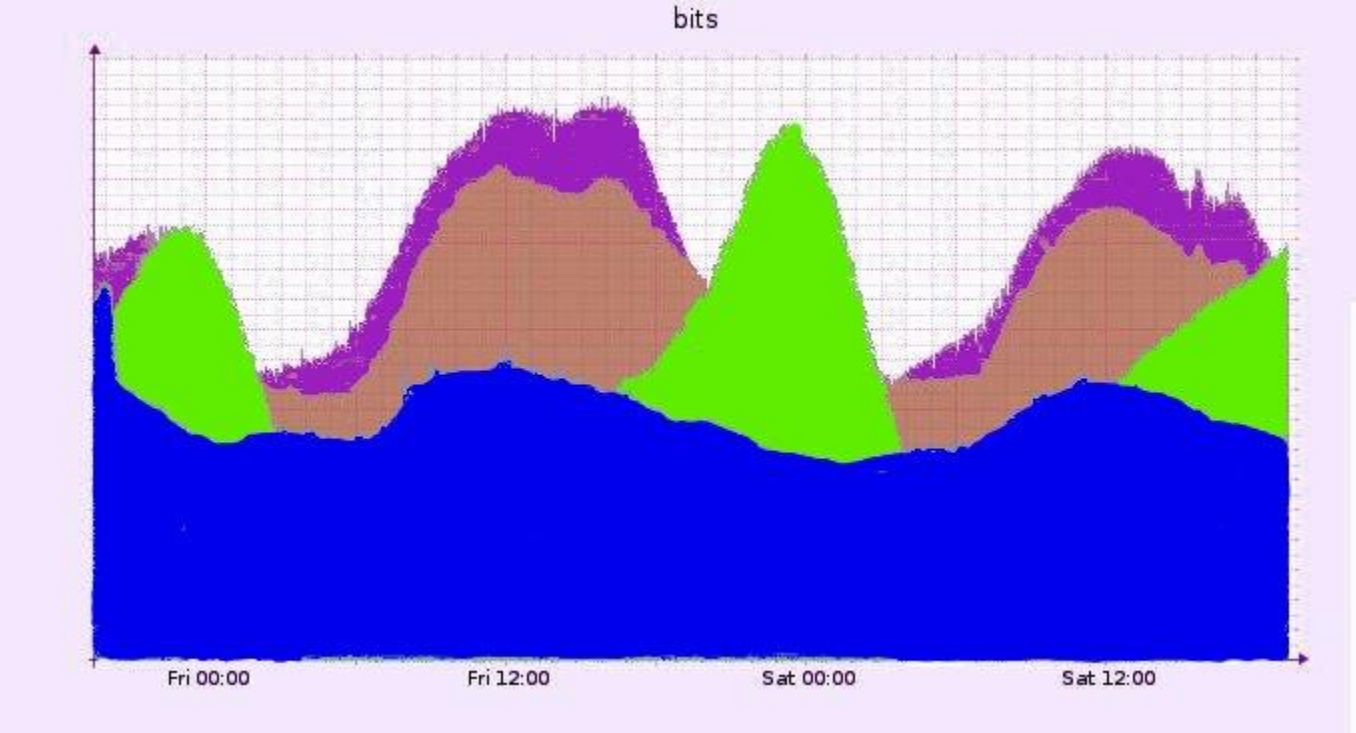
- Selling point: delivery of user-generated content
- Requirement: Extremely high origin offload
- Content type: thumbnails - small videos, bad cacheability
- Peak to valley ratio = 2
- Peak traffic at 1200-1800 EST



Media downloads and movie delivery

- Selling point: movie and media delivery
- Requirement: real-time performance
- Content-type: very large media files
- Peak to valley ratio = 7
- Peak traffic at 2200-0100EST

Combined traffic pattern



Not to scale

Metrics

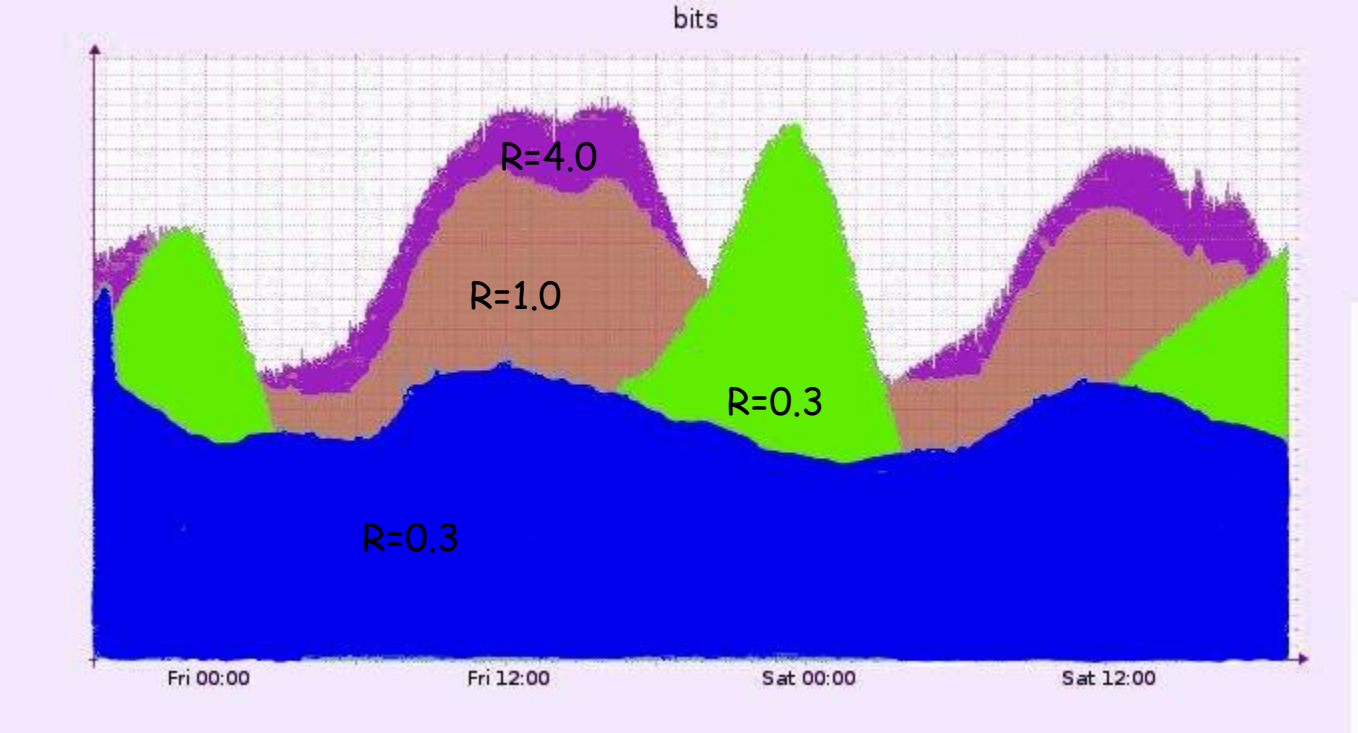
- Different resources in a server
  - CPU, Disk I/O, Memory, Network Bandwidth, Various internal limits
- Define U = utilization of the most constrained resource for a given level of traffic
- Define a metric for resource-intensiveness of a class of traffic, and normalize:
  - R = U * constant / traffic

R for different classes of traffic

Category	R	CPU	DISK
General	1.0	100%	60-70%
Software downloads	0.3-0.4	100%	<50%
Large-footprint long-tail	1.5-4.0	70-95%	100%
Media downloads	0.3-0.4	100%	<50%

Low R = High efficiency

Combined traffic pattern



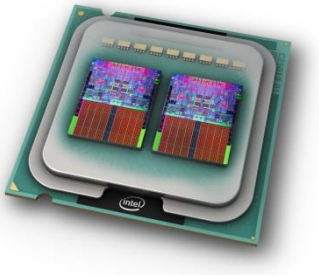
Not to scale

Trends

- Summary
  - Four large classes of traffic
  - Different requirements from the customer
  - Different traffic patterns
  - Different levels of resource intensiveness

Systems thinking

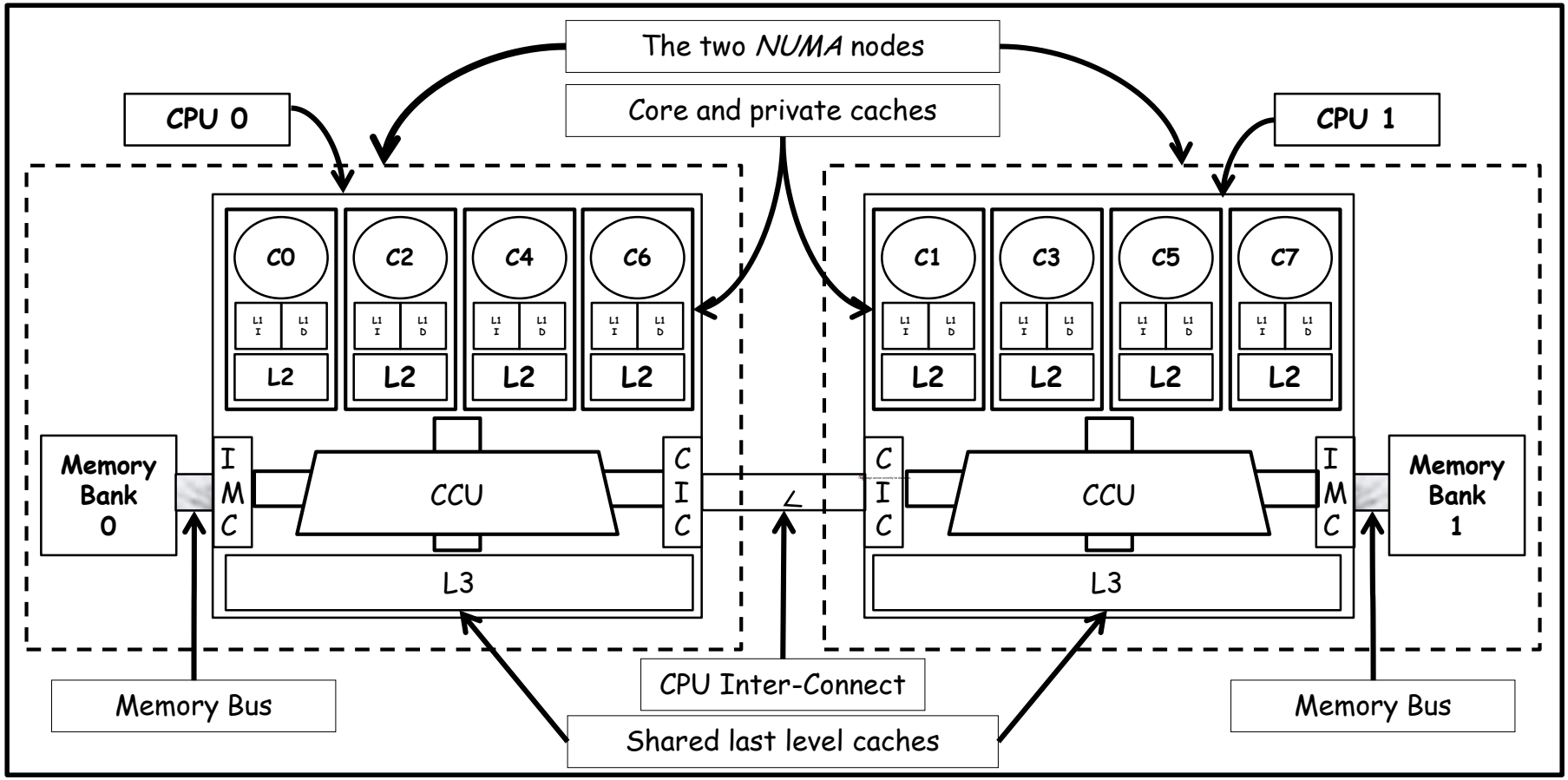
- We want to understand the full system and the ecosystem it operates within; e.g.,
  - Understanding the full system
  - Looking at the parts and how they interact
- This course provide many examples ...



Multicore systems

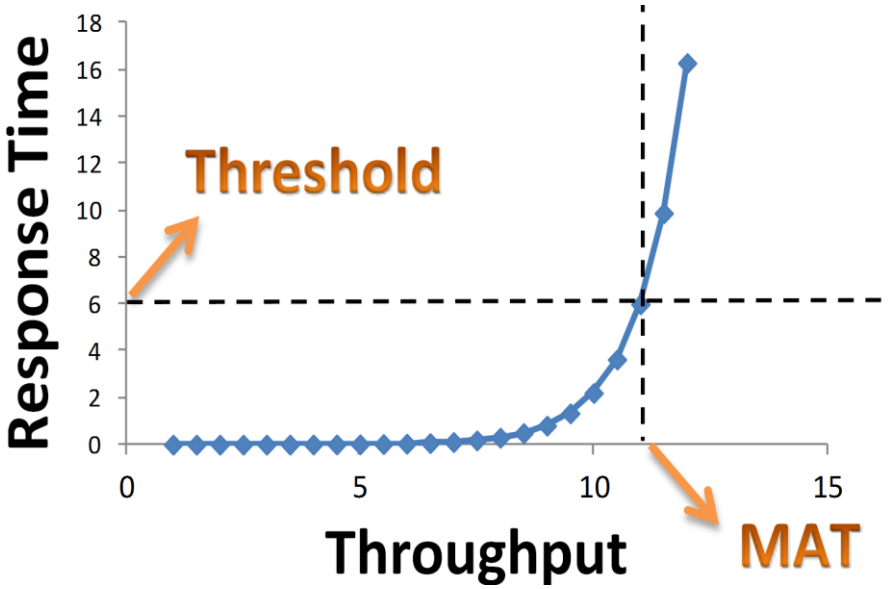
NUMA Architecture

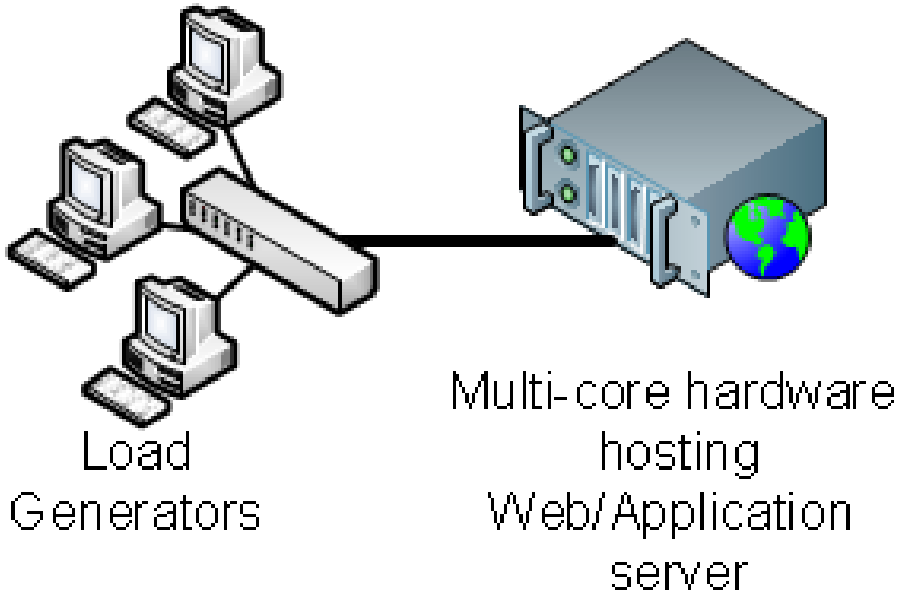
An example of a two processor eight core NUMA system



Scalability Evaluation Measurements

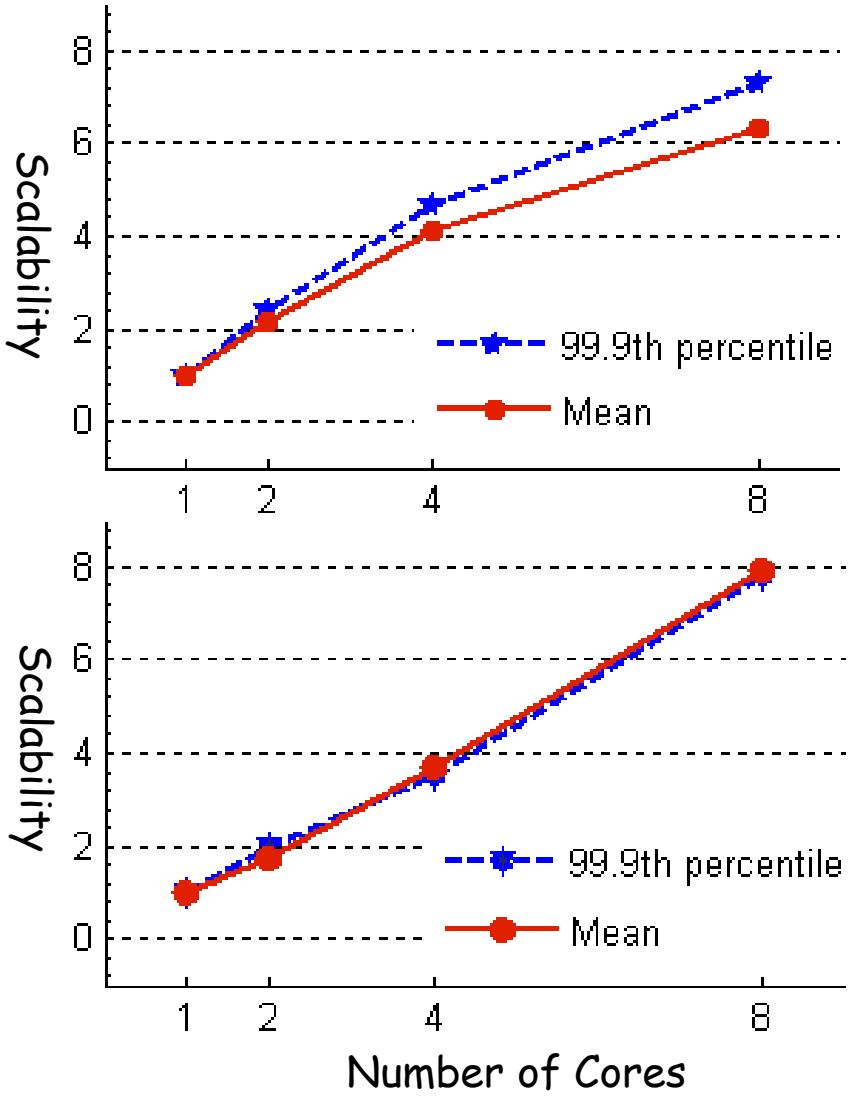
- E.g., Measure Web server scalability for workloads [ICPE '13]
  - Typically want to provide some 99% response time
  - Example scalability measure: Maximum Achievable Throughput (MAT)





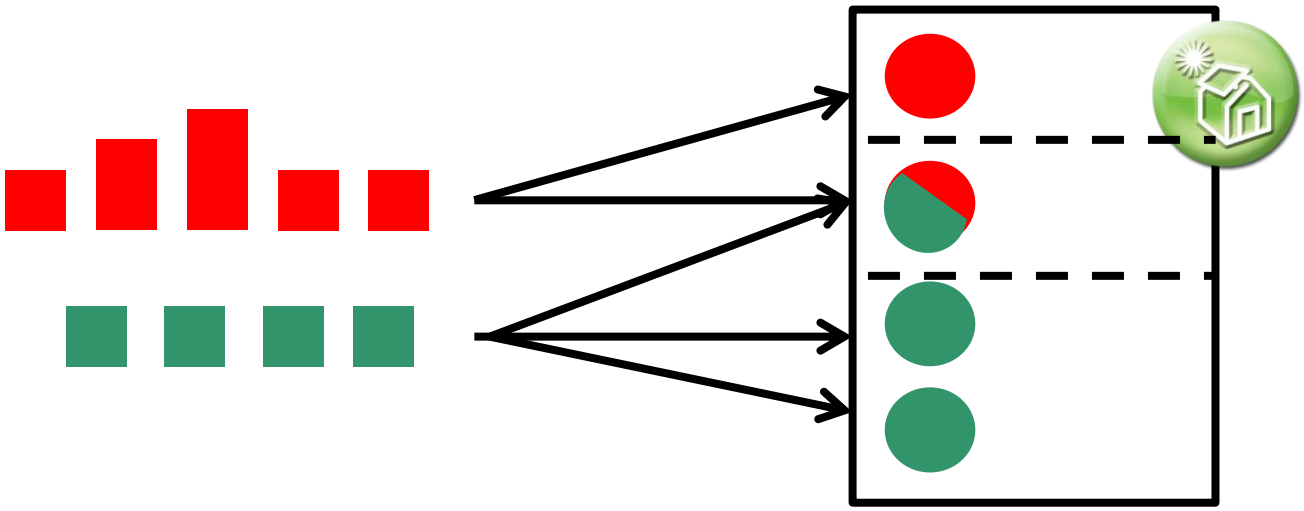
RESULTS

- TCP/IP Intensive workload
  - Sub-linear
    - Maximum Achievable Throughput
      - 146,000 req/sec
- SPECweb Support workload
  - Almost linear
    - Maximum Achievable Throughput
      - 23,000 req/sec

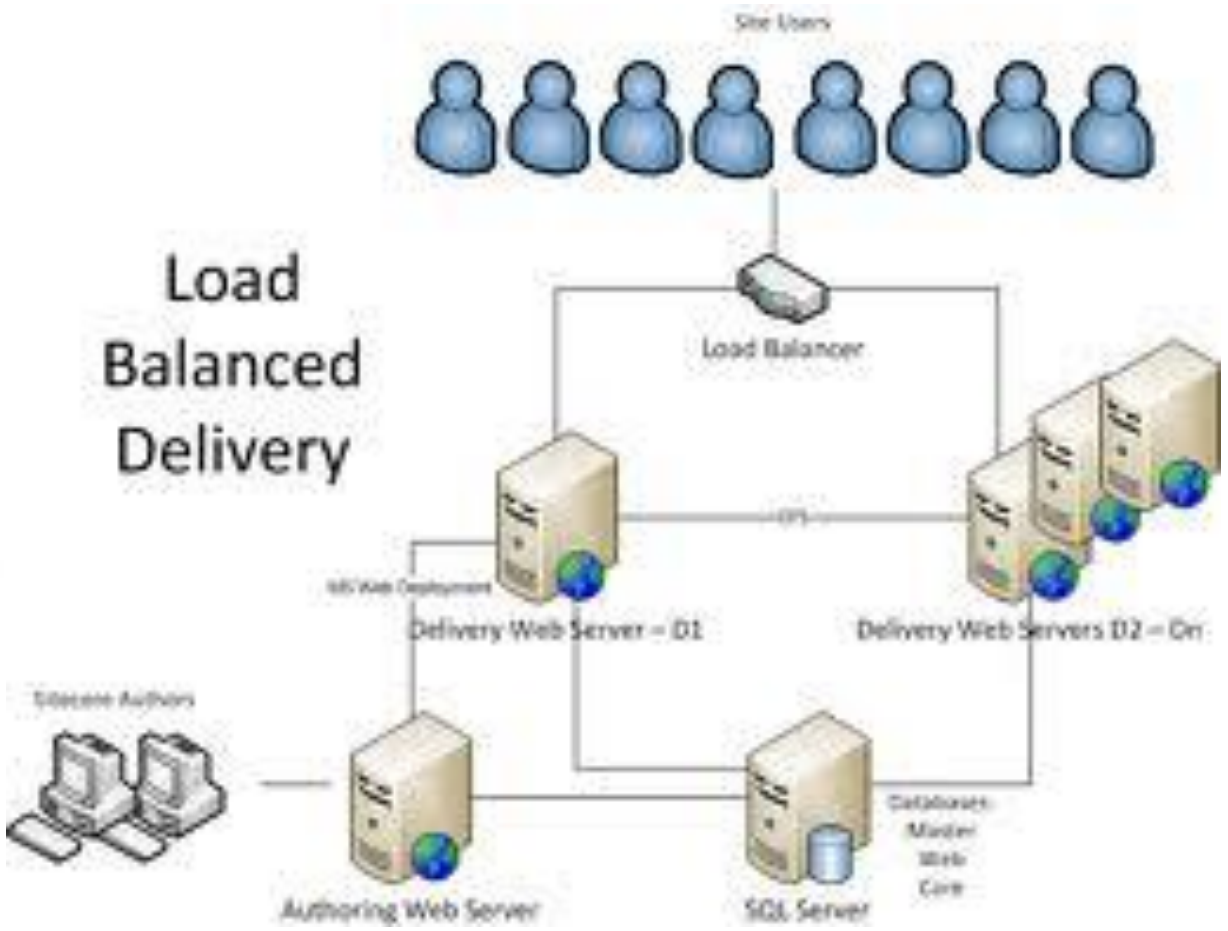


Identification of bottlenecks

- E.g., memory, CPU, network, cache hierarchy, interconnect bus, scheduler, …
  - Black-box testing
  - Low-level instrumentation
- Multiple workloads …



Often many servers (and racks)



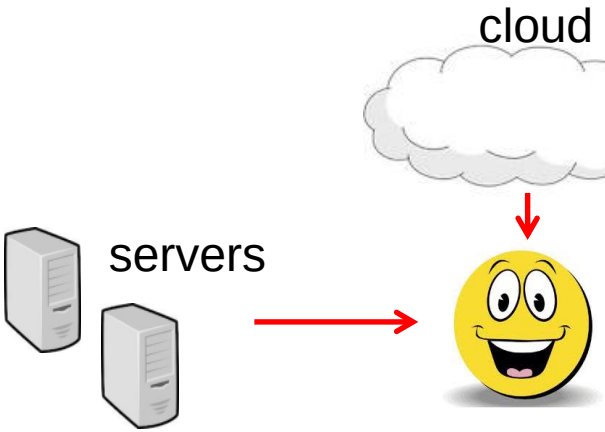
… cost-efficient delivery …

Example problem

- Minimize content delivery costs

	Bandwidth	Cost
Cloud-based	Elastic/flexible	$$$
Dedicated servers	Capped	$

How to get the best of two worlds?



… and from who?