互联网内容传输

张喆

zhezhang@njupt.edu.cn

通信与信息工程学院

目　录

互联网内容与流量

- 互联网过去与现在：
  - 过去：端到端通信，如email
  - 现在：内容分发，如看视频

互联网内容与流量

- 端到端通信VS内容分发：
  - 通信模式有何区别?
    - 端到端通信：sender与receiver是唯一确定的
    - 内容分发：不在乎sender，即内容的存储位置不重要
      - 多媒体内容：目前已超过网络流量的86%

互联网内容与流量

- 互联网内容与流量的特点：
  - 流量变化快：从传统的FTP文件传输与email占主导，变化为web流量，再到电影与音乐的P2P分享，再到目前的多媒体内容。
  - 流量有高度的偏斜性（highly skewed）：不同业务的流量差异性极大。

互联网内容与流量

- 流量分类：
  - Elephant flow：long traffic flow，如大文件传输流
  - Mice flow：short traffic flow，如控制流
  - 该种分类认为Elephant flow占少数，而mice flow占多数

互联网内容与流量

- 互联网内容流行度：
  - 并非每一个内容都有着相同的流行度
  - 通过实验得出：
    - 当有N个电影，那么第k个电影的请求量大约为C/k，其中C计算如下：
    - $C = 1/(1 + 1/2 + 1/3 + 1/4 + 1/5 + \ldots + 1/N)$
    - 该结果称为Zipf's law，由哈佛大学George Zipf命名。

互联网内容与流量

- Zipf分布：长尾效应（绝大多数的内容并不流行）

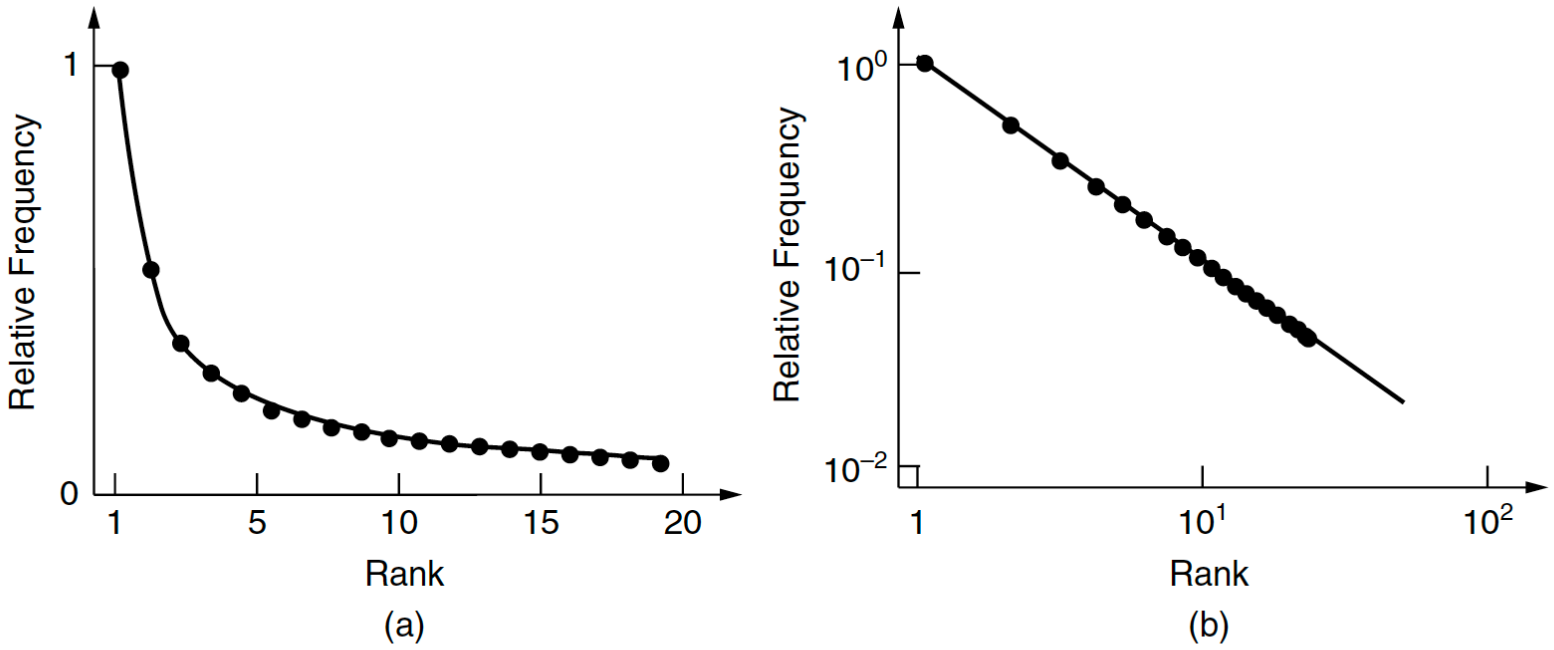
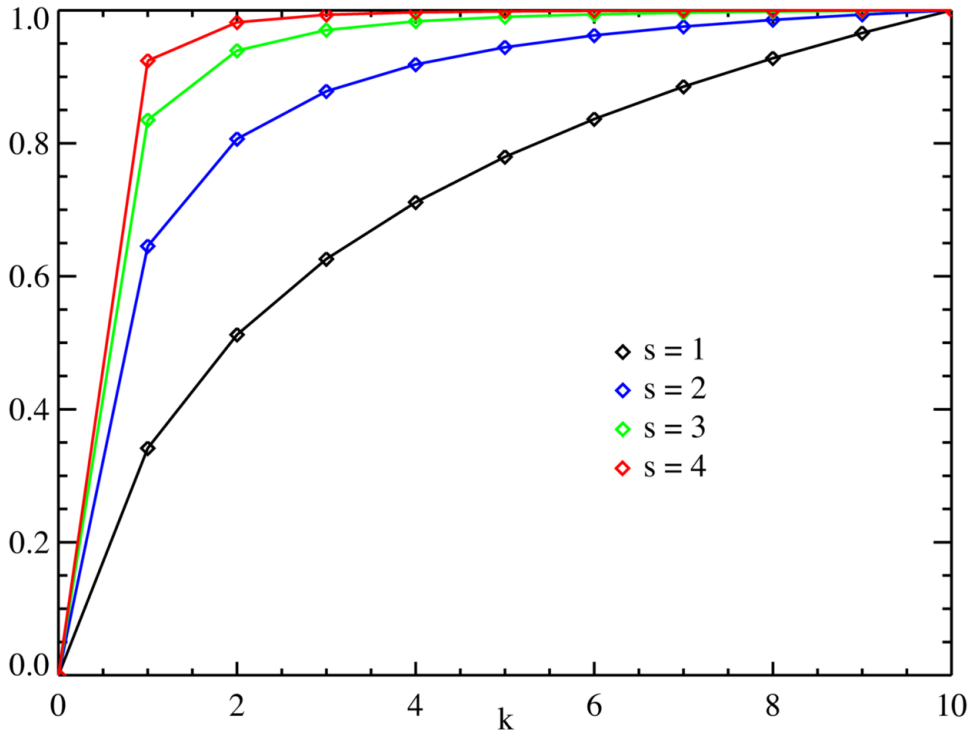


Figure 7-64. Zipf distribution (a) On a linear scale. (b) On a log-log scale.

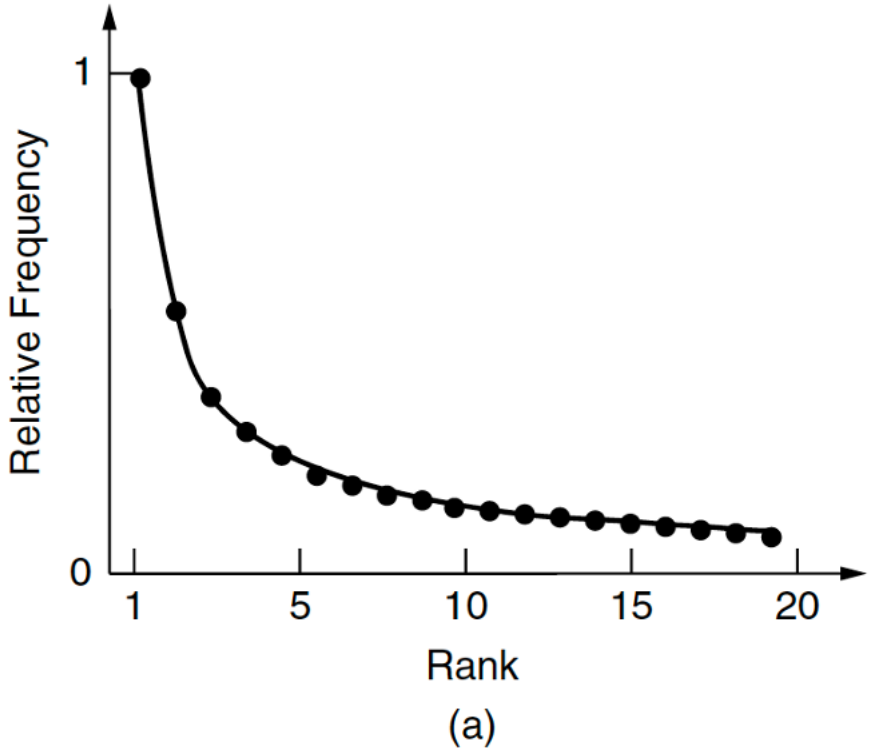
互联网内容与流量

- Zipf-distribution是power-law（2-8分布）的一种
- 右图为Zipf- distribution的CDF（cumulative distribution function）图



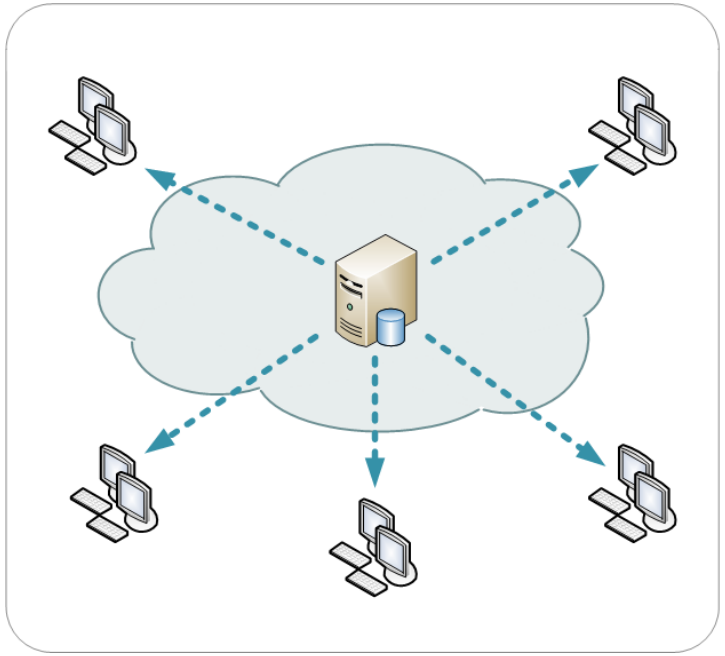
互联网内容与流量

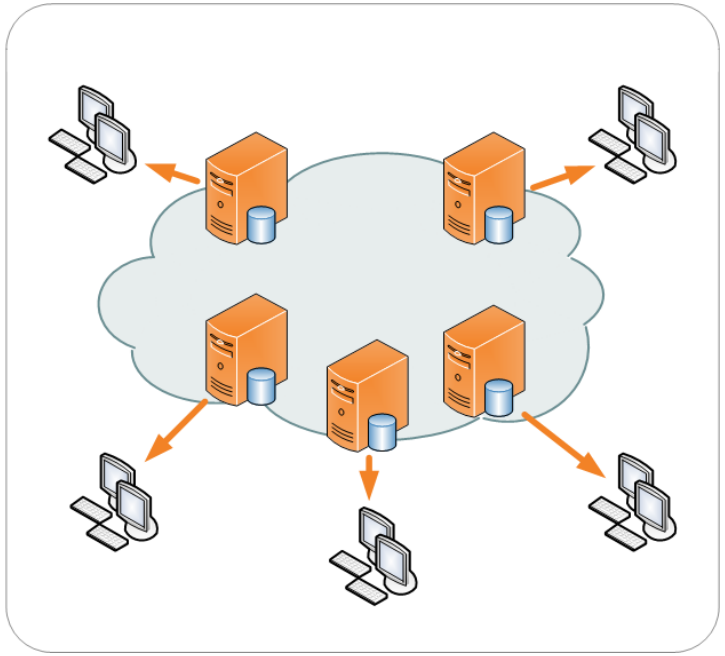
- 长尾衰减率：
  - 并非总是保持一致的
  - 在特定情况下会呈现出指数级衰减 $e^{-t/a}$，如放射性原子的半衰期。
    - 此刻就不再符合Zipf-distribution了。
    - 区别：Zipf's law需要考虑长尾部分的内容，而指数衰减则可以忽略。



互联网内容与流量

- 如何处理流行内容与不流行内容呢?





目　录

多媒体内容的基本概念

- Video *bit rate*: effective number of bits per second of the video after encoding
- Higher bit rate == higher perceptual quality
- CBR: (constant bit rate): fixed bit-rate video
- VBR: (variable bit rate): different parts of the video have different bit rates, e.g., changes in color, motion, etc.
  - For VBR, we talk about average bit-rate over video's duration

网络多媒体内容的常见类型

- On-demand streamed video/audio
  - Can begin playout before downloading the entire file
  - Ful video/audio stored at the server: able to transmit faster than audio/video will be rendered (with storing/buffering at client)
  - e.g., Spotify, YouTube, Netflix, Bilibili
- Conversational voice or video over IP
  - interactive human-to-human communication limits delay tolerance
  - e.g., Zoom, Tencent Meeting
- Live streamed audio, video
  - e.g, sporting event on sky sports，抖音直播
  - Can buffer a little, but must be close to the “live edge” of content

视频分辨率与带宽的关系

- 带宽=分辨率×帧率×每像素比特数×压缩比
- 例如：对于一个1080p（1920×1080）分辨率、30 fps的视频，假设平均每像素使用24比特（RGB图，3字节），没有任何压缩的情况下，每秒钟的数据量可以计算如下：

  数据量=分辨率×帧率×每像素比特数=1920×1080×30×24≈1.5 Gbp
- 如果采用H.264压缩算法的话，压缩比可达50:1～200:1，所需带宽会锐减至7.5～30 Mbps。

视频编码

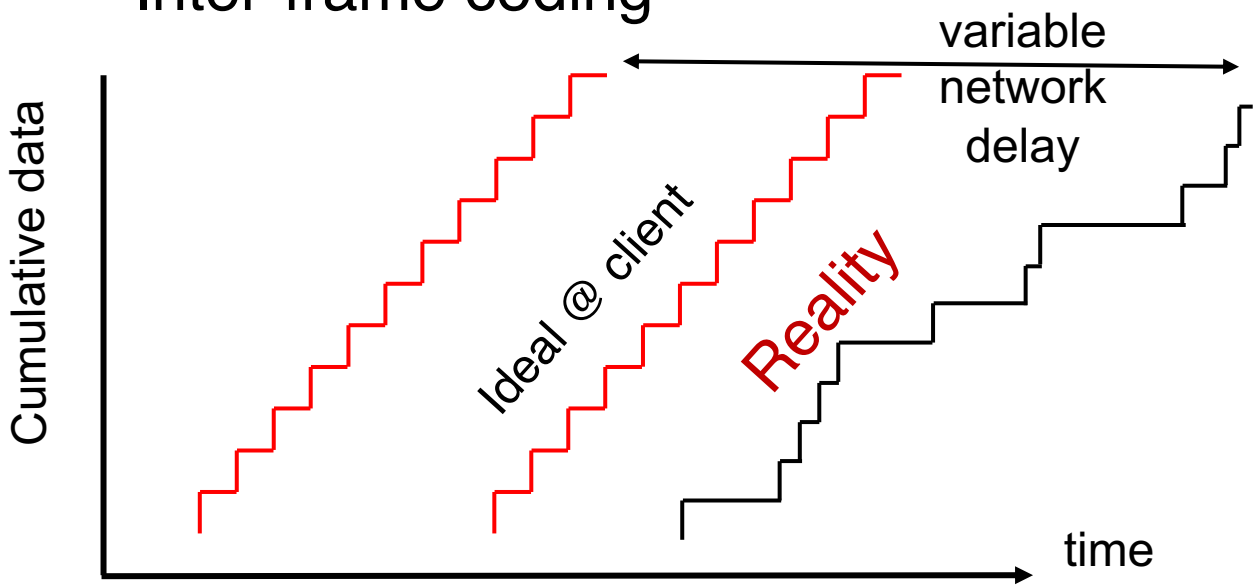
Video representation:
Pixels in Frames

Bits played out per second (can vary over video's lifetime)

Codec

Intra-frame coding

Inter-frame coding



Buffer at the client to hold frames initially until playout delay t_p

Choosing t_p is hard! Don't know buffer fill rate a priori

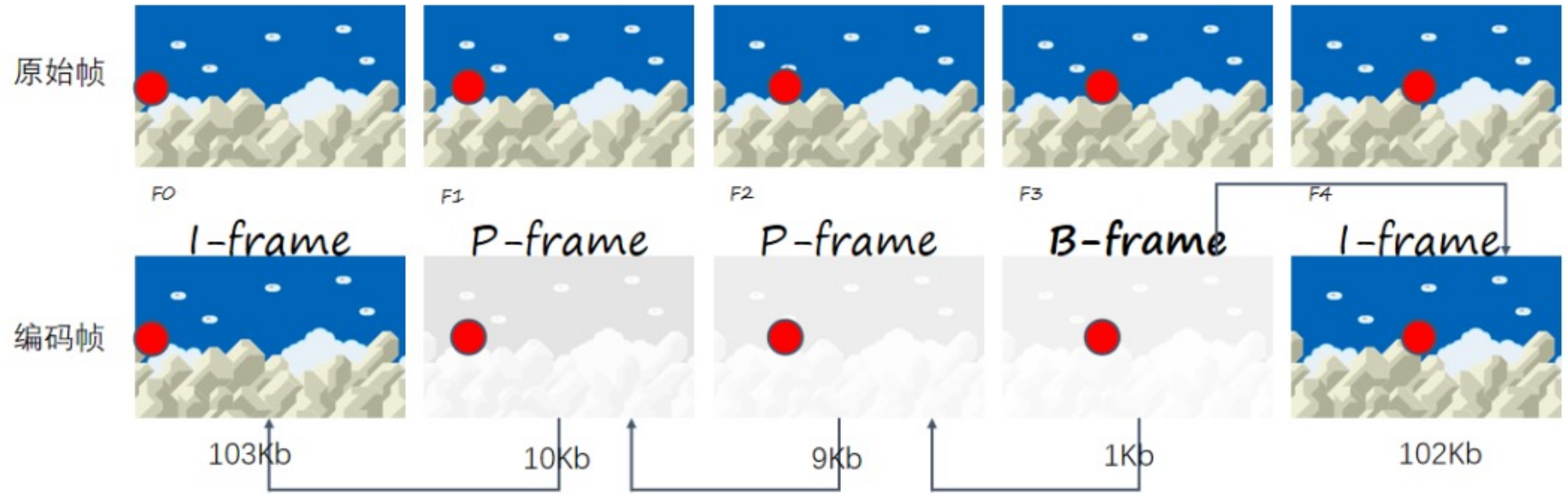
Adaptive bit-rate selection

视频编码

- Intra-frame coding （帧内编码）：
  - 仅在单个帧内进行压缩，不依赖其他帧的信息。
  - E.g.，JPEG、H.264的I帧
  - 特点：每个帧独立编码，解码时不需要其他帧，适合编辑和随机访问。
- Inter-frame coding（帧间编码）：
  - 利用相邻帧之间的时间冗余进行压缩，通过预测和补偿来减少数据量。
  - 例子：H.264的P帧和B帧。
  - 特点：通常比帧内编码效率更高，但解码复杂度更高，需要访问多个帧。

INTER-FRAME CODING（帧间编码）- H.264

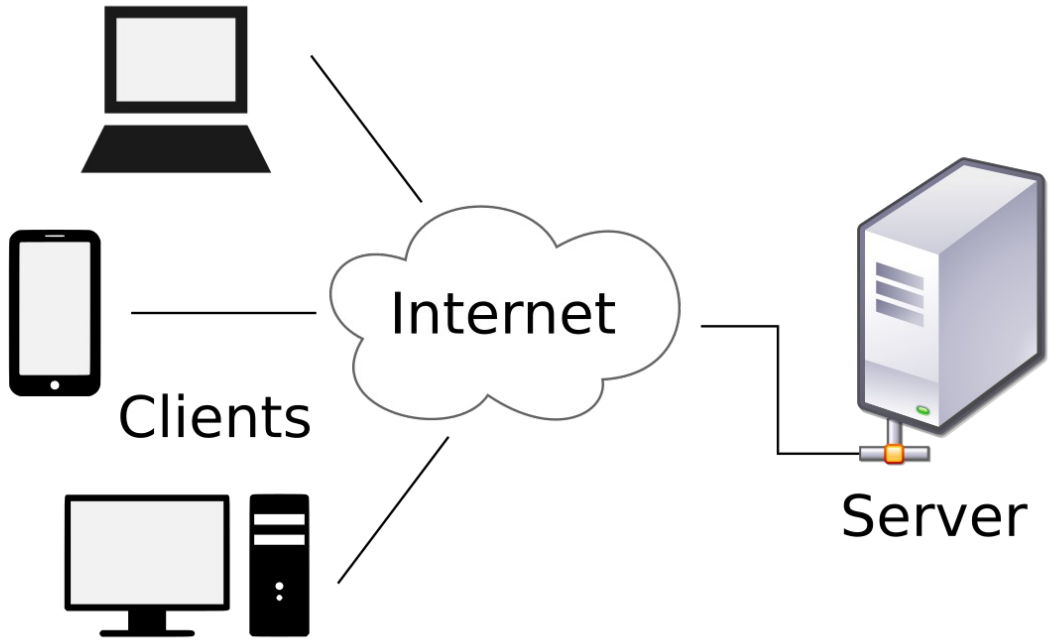
- **I帧**：完整独立编码的帧，用于关键帧和随机访问。
- **P帧**：前向预测编码帧，基于前面的参考帧进行压缩。
- **B帧**：双向预测编码帧，基于前后帧进行压缩，提供最高的压缩效率。

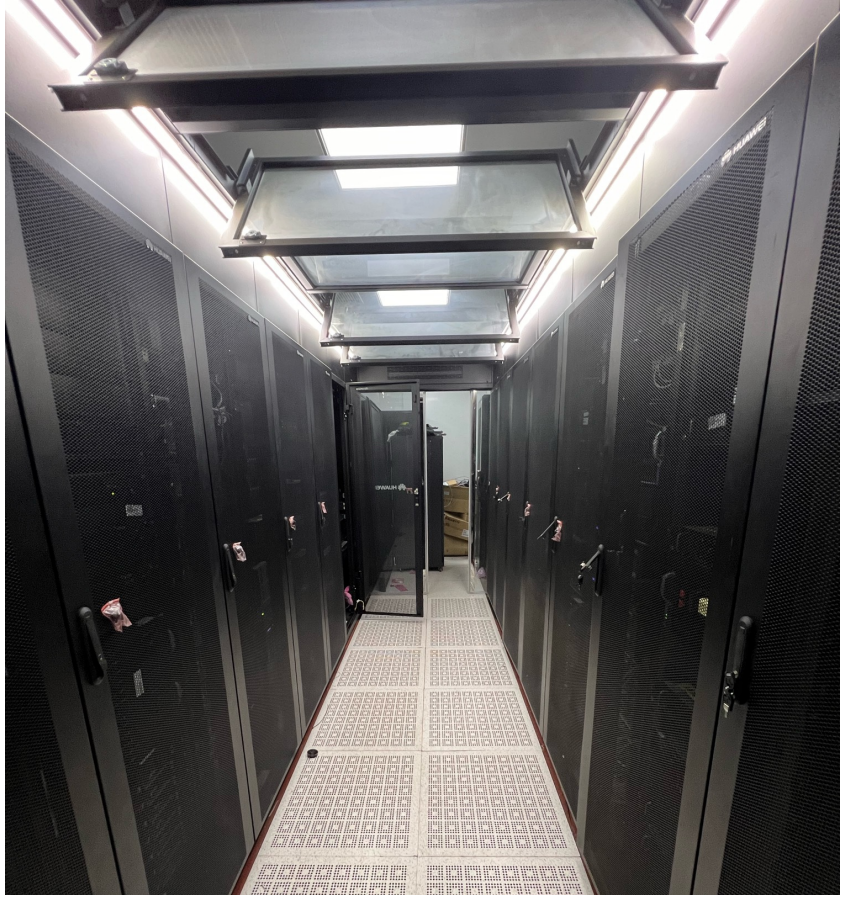


目　录

SERVER FARMS AND WEB PROXIES

- 如何提升网站的性能呢?
  - 从服务器侧：使用更多高性能服务器->构成server farms
  - 从用户侧：proxy cache（代理缓存）

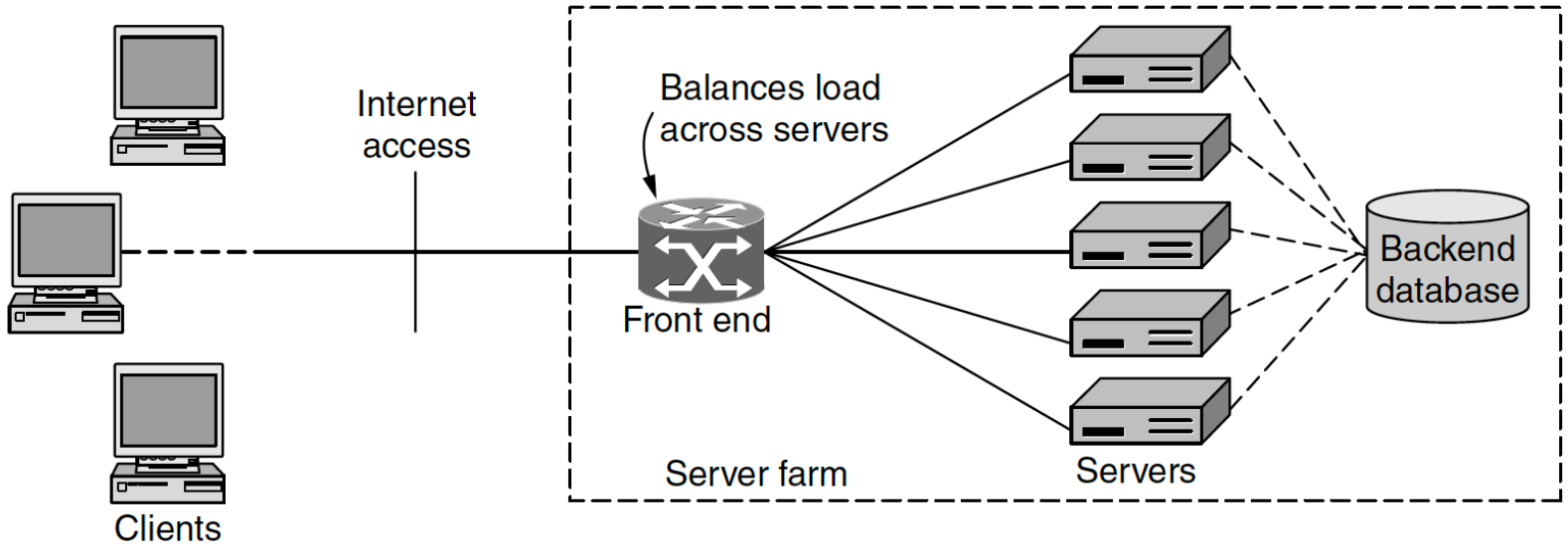




SERVER FARMS AND WEB PROXIES

SERVER FARMS AND WEB PROXIES

- Server farm模型：



SERVER FARMS AND WEB PROXIES

- Server farm模型要求：
  - 众多的服务器需要在逻辑上能够被认为是一台服务器。
  - 否则的话，相当于在并行运行的众多不同的web服务器。

SERVER FARMS AND WEB PROXIES

- 如何实现上述需求呢?
- 方案一：利用DNS来重定向
- 方案二：设置前端节点（通常为交换机或路由器）
- 方案三：在前端设置load balancer
- 方案二和三都是基于前端的方案（Front end）

SERVER FARMS AND WEB PROXIES

- 方案一：
  - DNS收到web URL请求时，通过轮询（round-robin）的方式从服务器群中选择其中一台服务器的IP，并将该IP返回给用户。
  - 效果：不同用户通过访问不同服务器来实现浏览同一网页的效果。

SERVER FARMS AND WEB PROXIES

- 方案二：
  - Front end将所有请求广播给所有服务器。
  - 每台服务器根据预设协议只回复其中的一小部分请求。
    - 比如16台服务器只根据源IP地址的最后4位来决定是否应答。
  - 效果：绝大多数的广播数据包都被丢弃了，导致带宽的极大浪费。

SERVER FARMS AND WEB PROXIES

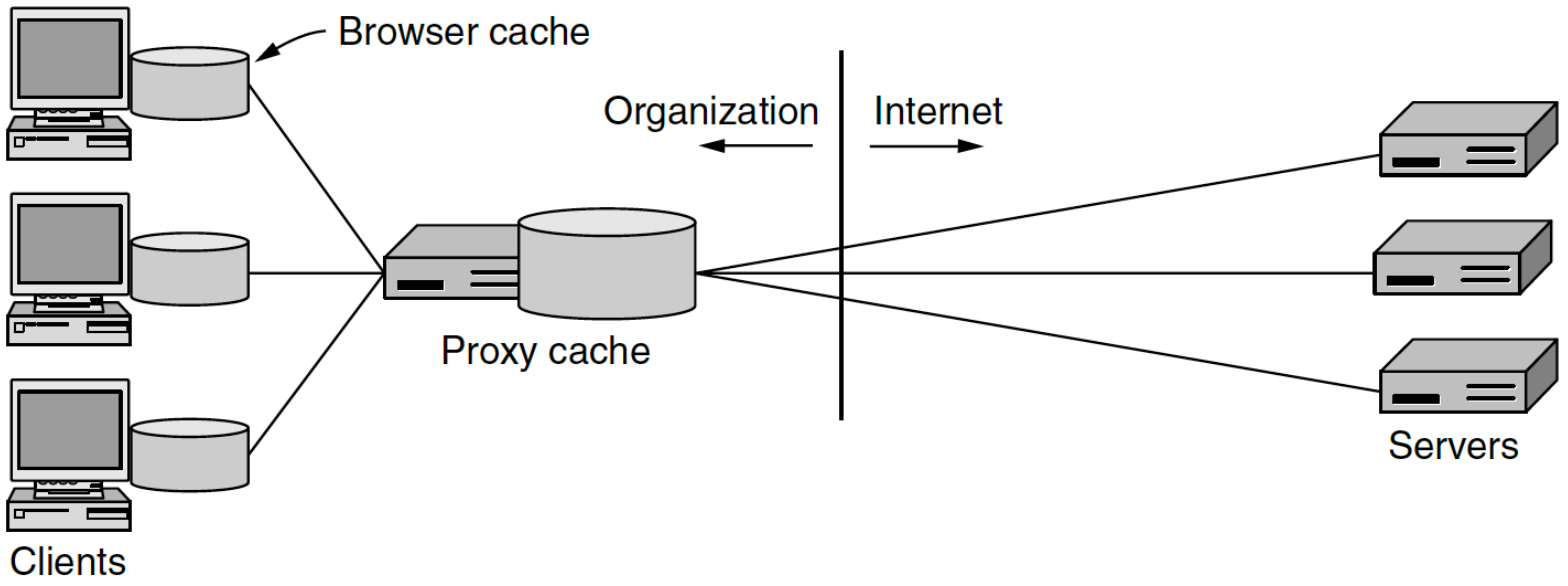
- 方案三：
  - 采用load balancer（本质还是交换机或路由器）来识别TCP、HTTP等更多种类的数据。
  - 最简单的方法是load balancer通过轮询的方式来将请求数据挨个传给不同服务器。

SERVER FARMS AND WEB PROXIES

- Web proxies:
  - A Web proxy is used to share a cache among users. A proxy is an agent that acts on behalf of someone else, such as the user.
  - 通过将web页面缓存在本地来降低传输时延以及网络带宽负载。

SERVER FARMS AND WEB PROXIES

- Web proxy cache架构图



SERVER FARMS AND WEB PROXIES

- Web proxy可以提供多级缓存：
  - Upstream proxy：
    - 当本地代理没有缓存本地内容时，则需要访问upstream proxy
  - Downstream proxy：
  - 比如，可以有company proxy，ISP proxy等。

SERVER FARMS AND WEB PROXIES

- Web proxy还可以提供更多其他功能：
  - 如过滤内容，很多公司可以通过配置代理来过滤掉访问特定站点。
  - 还可以提供隐私保护。
    - 通过匿名的方式来使得服务器无法获知用户身份信息。

SERVER FARMS AND WEB PROXIES

- 缓存算法：
  - 缓存决定算法：决定哪些内容应该被缓存下来。
  - 缓存替换算法：当缓存空间耗尽时，决定哪些内容应该被替换掉以腾出空间来缓存新内容。

SERVER FARMS AND WEB PROXIES

- 典型算法：
  - 典型缓存决定算法：缓存所有内容。
  - 典型缓存替换算法：LRU(least recently used), LFU (least frequently used)。
    - LRU：优先替换最近未被访问的内容。
      - 可以通过链表结构实现LRU。
    - LFU：优先替换最近被访问次数最少的内容。
      - 可以通过设置计数器来实现LFU。

Thank You