

# ● 网络计算模式

- 概述
- 企业计算
- 网格计算和云计算
- P2P网络、CDN网络和物联网
- 社会计算



# P2P网络

## ∞P2P网络概念

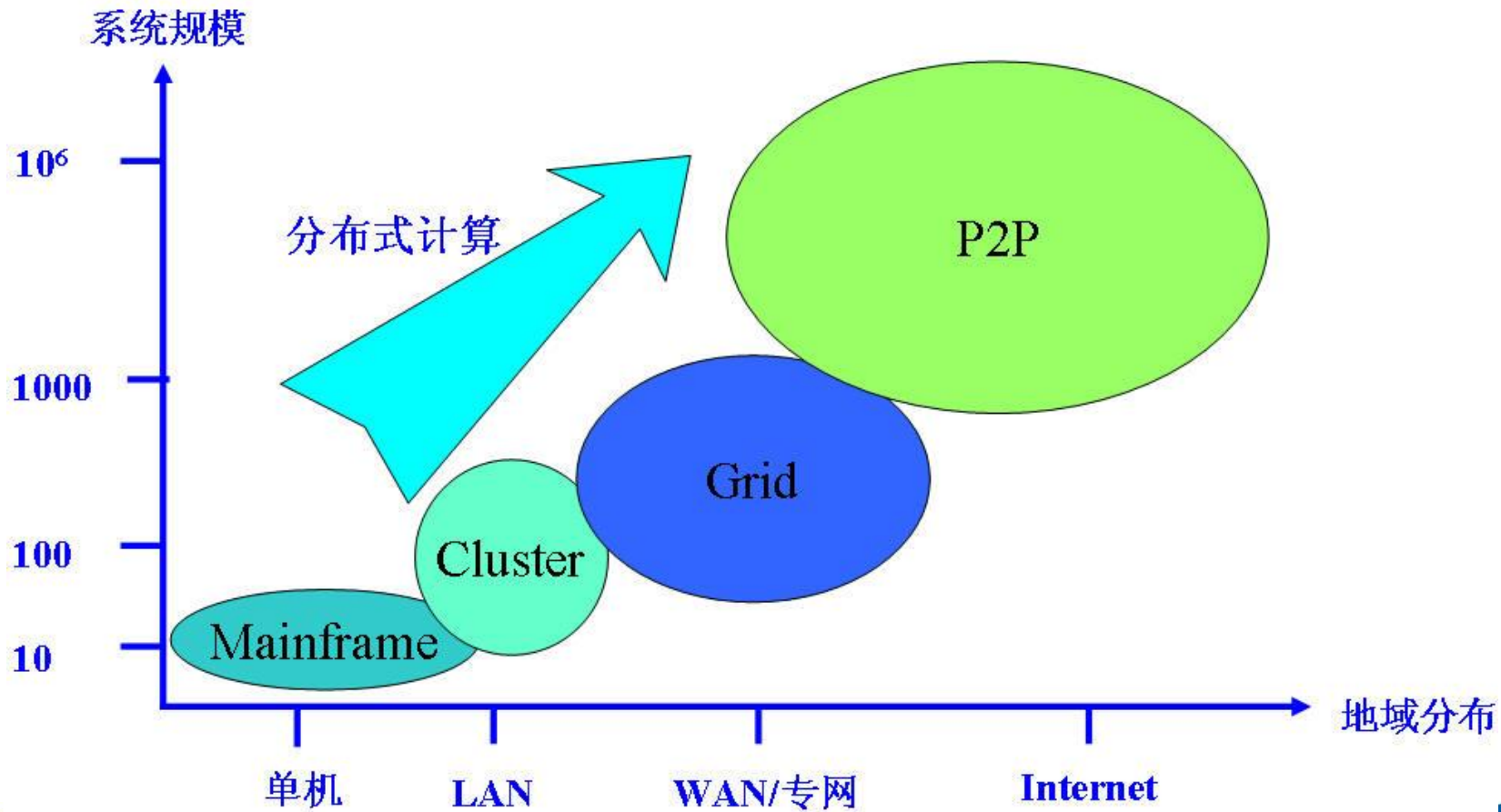
### 什么是P2P?

◆对等网络(Peer-to-Peer Networks)是分布式系统和计算机网络相结合的产物，在应用领域和学术界获得了广泛的重视和成功，被称为“改变Internet的一代网络技术”。



# P2P网络

## ∞P2P网络概念



# P2P网络

## ∞P2P网络概念

### 什么是P2P?

- ◆P2P: Peer to Peer对等网络。
- ◆peer指网络结点，在行为上是自由的——任意加入、退出，不受其它结点限制，匿名；在功能上是平等的——不管实际能力的差异；在连接上是互联的——直接/间接，任两结点可建立逻辑链接，对应物理网上的一条IP路径。
- ◆充分利用网络带宽、节点资源，提高工作效率。



# P2P网络

## ∞P2P网络概念

∞ 信息交流与计算机网络

∞ 世界上最大的广域网——Internet

∞ 计算机网络的管理与控制的方式

- ◆ 集中式：Server/Client 结点；易管理、工作效率低、可扩展性差

- ◆ 分布式：distributed system，结点地位平等

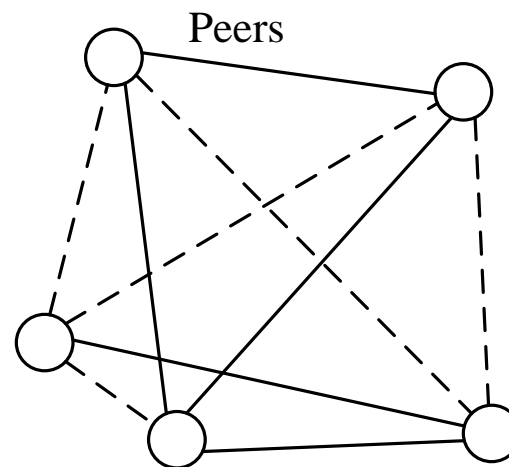
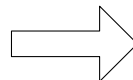
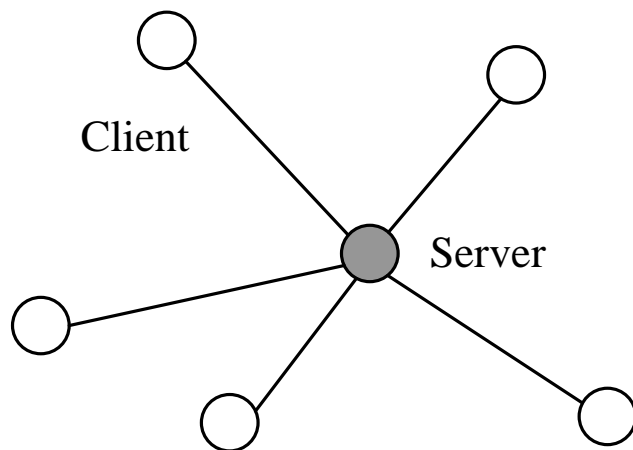
- ◆ 混合式：hybrid system



# P2P网络

## ∞P2P网络概念

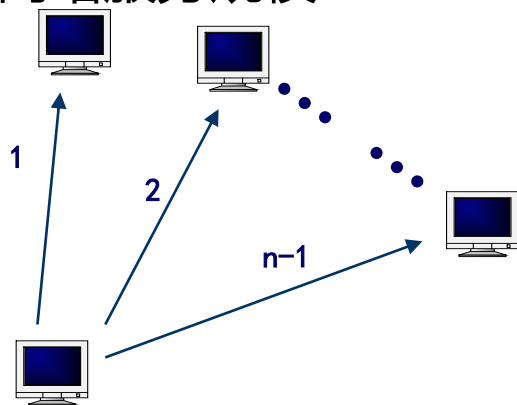
◆从C/S到P2P



# P2P网络

## ∞P2P网络概念

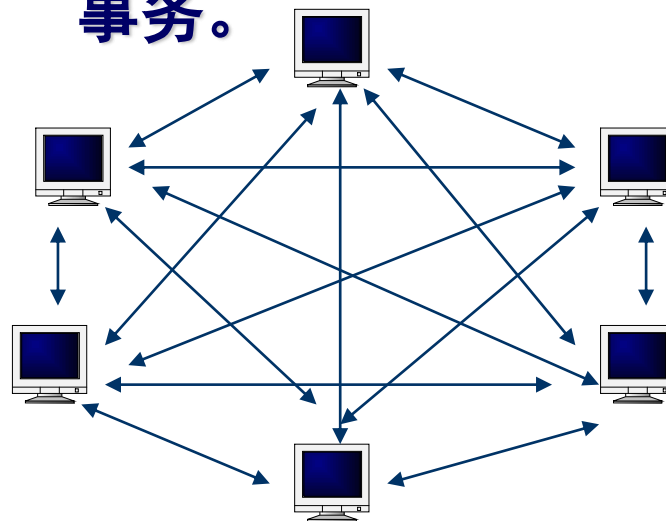
### ◆网络服务规模



A: Sarnoff 'law : 规模是 $O(n)$

➤ Sarnoff 'law: 效益规模是 $O(n)$ : 网络是广播媒介, 任1发送者(设备)和多个 $(n-1)$ 接收者(设备)。

➤ Metcalfe 'law: 效益规模是 $O(n^2)$  网络是全互连媒介, 任何1个设备可与其它 $n-1$ 个交互, 同时存在 $n(n-1)=n^2-n$ 个并发执行的事务。



B: Metcalfe 'law : 规模是 $O(n^2)$



# P2P网络

## ∞P2P网络的发展

- ∞ 1999年Internet上第一个应用Napster, 半年发展了5000万用户。
- ∞ 其后涌现Gnutella, KaZaA, BitTorrent, eDonkey/eMule, Skype
- ∞ 发展的原因:
  - ◆ 占据Internet一半以上的带宽
  - ◆ 学术界重视





# P2P网络

## ∞P2P网络的发展

∞ 1999年Internet上第一个应用Napster，半年发展了5000万用户。

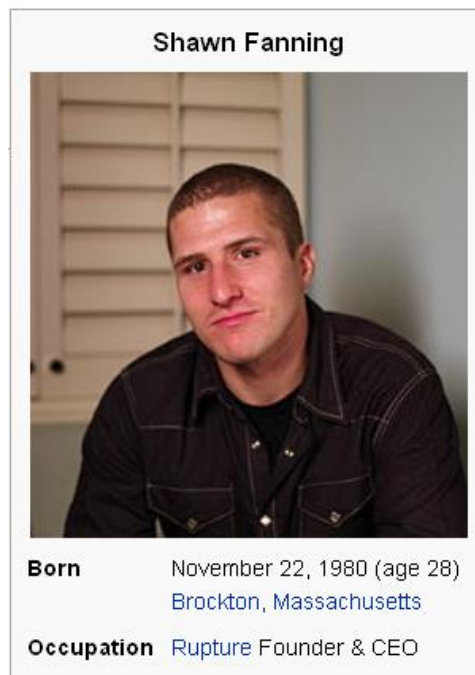
∞ 溯源：Napster

-- 1999年，18岁的美国学生Shawn Fanning

-- 宿舍开发，朋友共享mp3

-- 半年5000万用户！

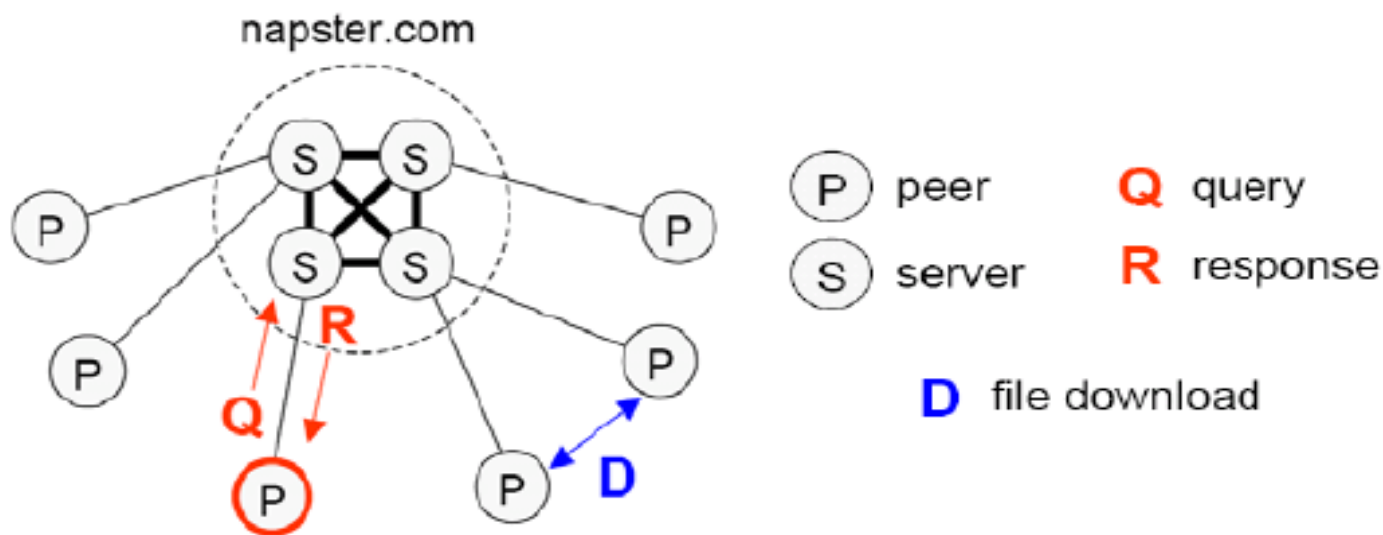
-- 2001年，版权纠纷，被迫关闭。



# P2P网络

## ∞ P2P网络的发展

### ∞ Napster运行原理



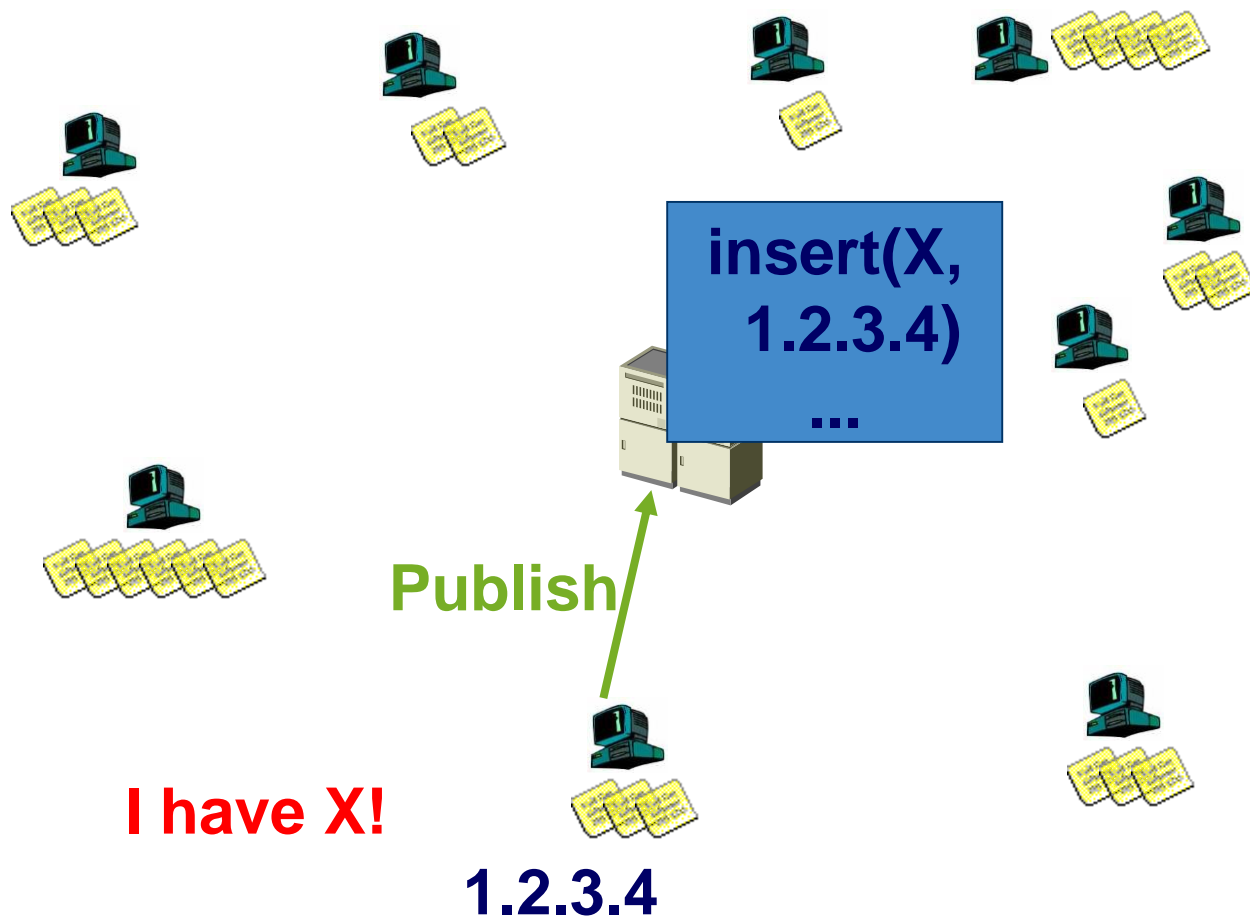
◆ Napster是众所周知的音乐交换系统。每个节点登录到服务器上并发送它们的文件清单，发布查询到服务器上查找哪些节点是它们拥有的想要的文件，并直接与目标节点连接下载文件。



# P2P网络

## ∞ P2P网络的发展

### ∞ Napster运行原理

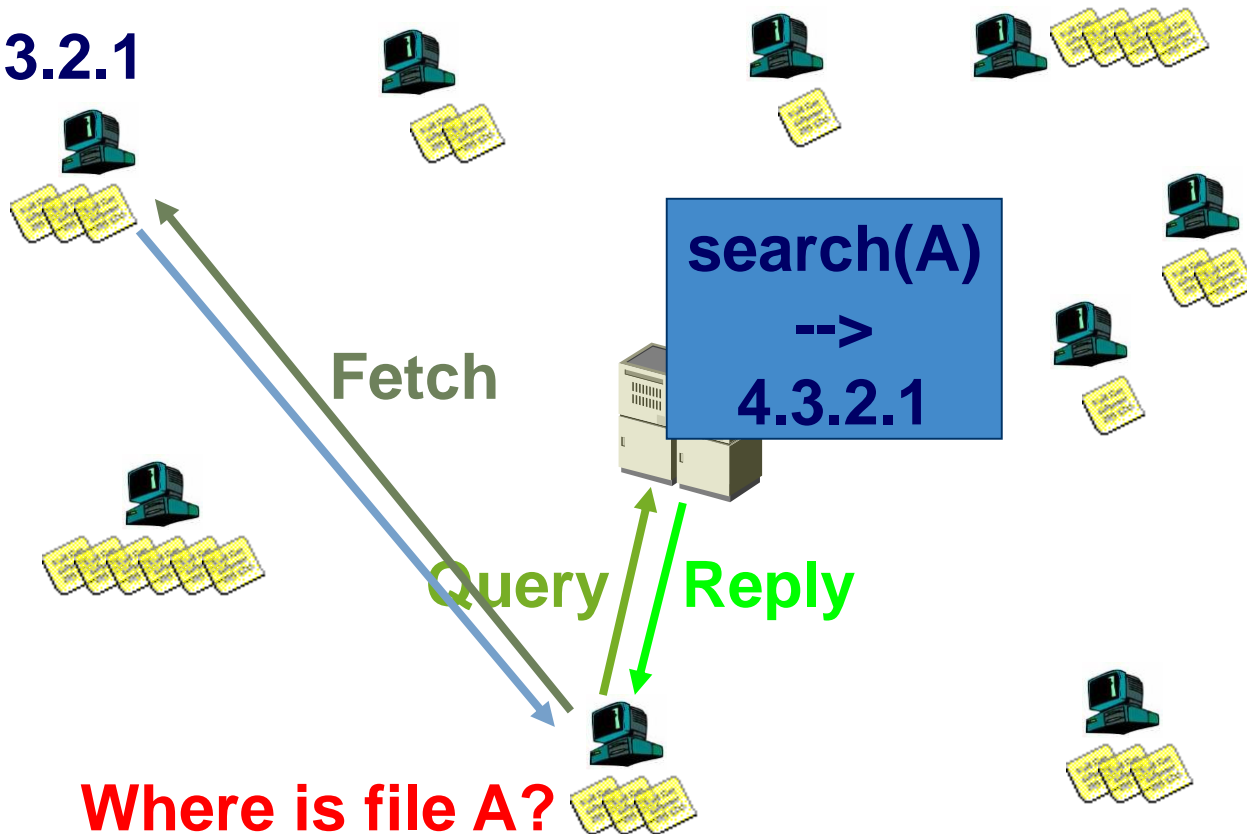


# P2P网络

## ∞ P2P网络的发展

### ∞ Napster运行原理

4.3.2.1



# P2P网络

## ∞P2P网络的发展

∞ 从计算模式的发展来理解P2P的必然性。

◆ 主机终端计算模式

◆ C/S计算模式

◆ B/S计算模式

◆ P2P计算模式



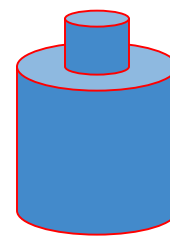
# P2P网络

## ∞ P2P网络的发展

### ∞ 主机终端计算模式

终端

主机端



计算模式支点

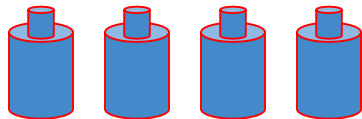


# P2P网络

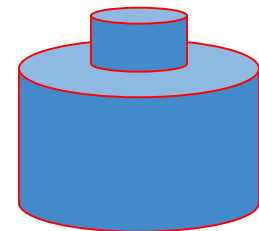
## ∞ P2P网络的发展

∞ 随着时间的发展

终端



主机端



计算模式支点

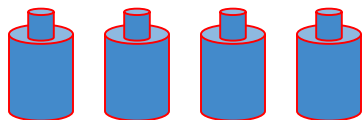


# P2P网络

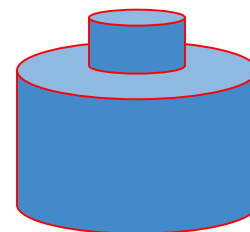
## ∞ P2P网络的发展

### ∞ C/S计算模式

客户机端



服务器端



计算模式支点





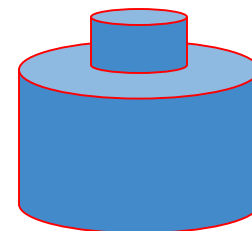
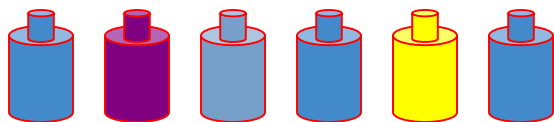
# P2P网络

## ∞ P2P网络的发展

∞ 异构性问题出现

客户端

服务器端



计算模式支点

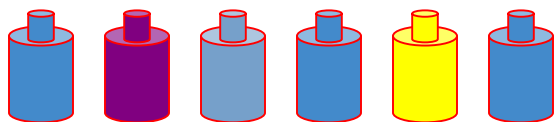


# P2P网络

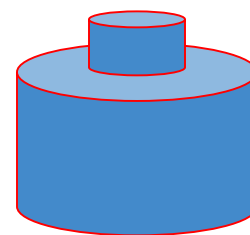
## ∞ P2P网络的发展

∞ 为了解决异构性的问题...B/S

客户机端



服务器端



计算模式支点

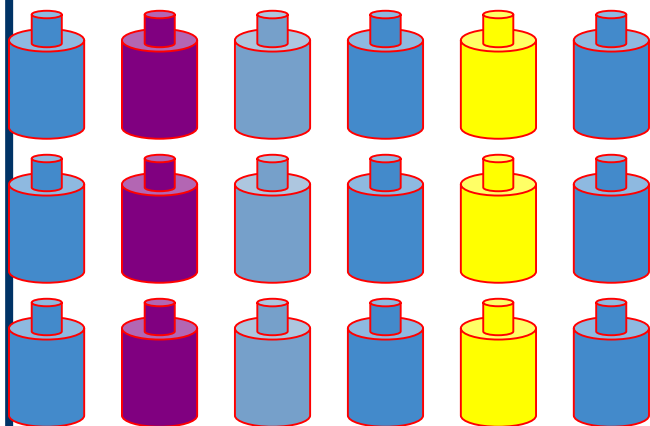


# P2P网络

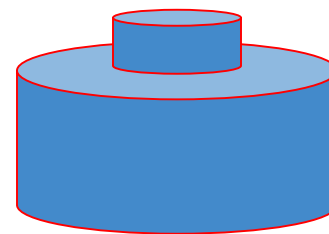
## ∞ P2P网络的发展

∞ 随着Internet的迅猛发展

客户机端



服务器端



计算模式支点

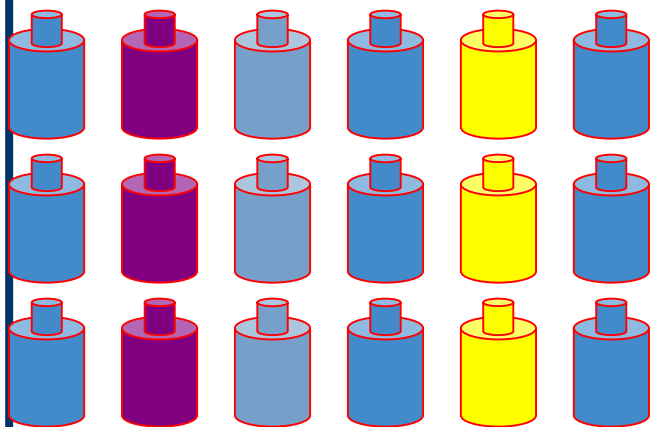


# P2P网络

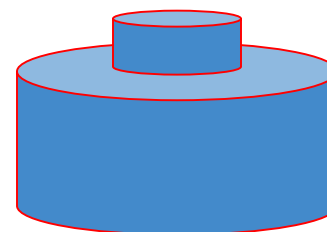
## ∞ P2P网络的发展

∞ 如何利用终端资源

客户机端



服务器端



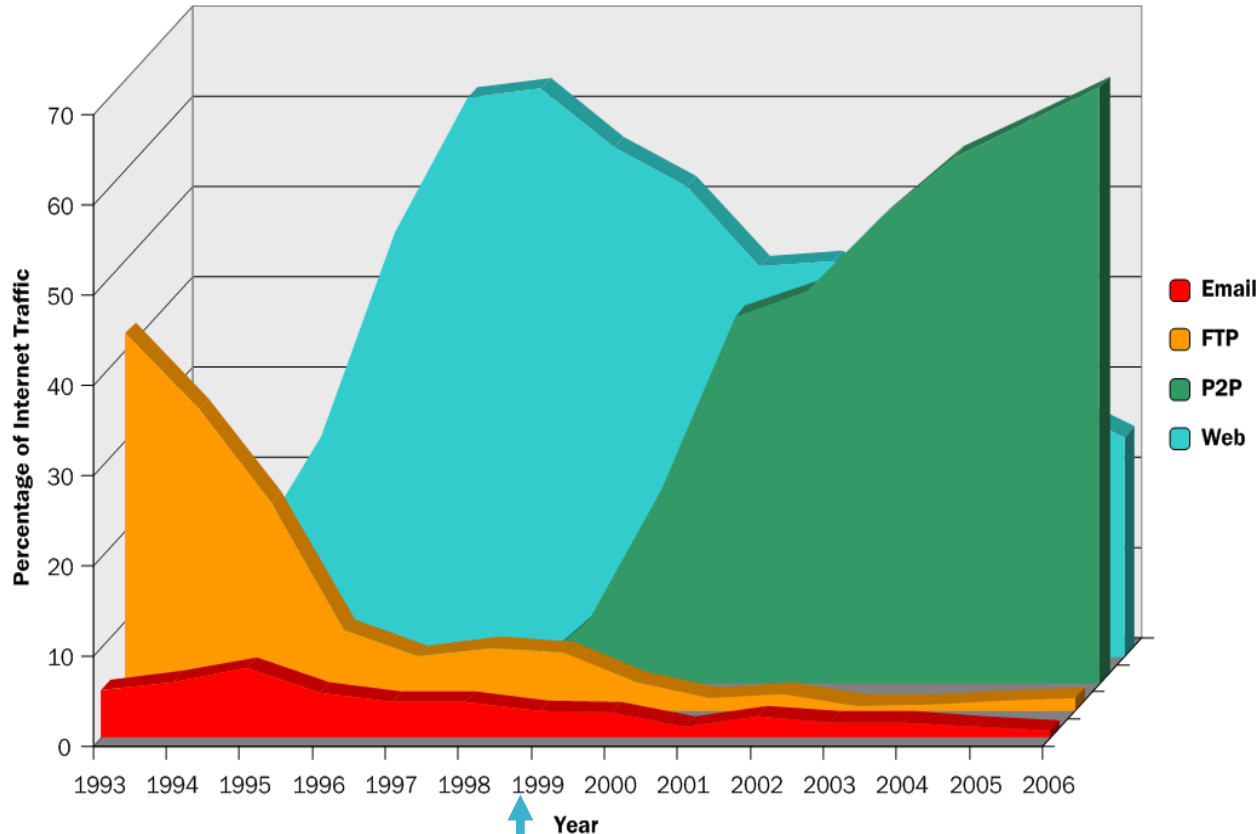
计算模式支点



# P2P网络

## ∞ P2P网络的发展

CacheLogic Research | Internet Protocol Trends 1993 to 2006



**P2P**  
represented  
~65% of  
Internet  
Traffic at  
end 2006

∞ 1999: *Napster*, first widely used p2p-application



# P2P网络

## ∞P2P网络的发展

∞ 不同类型P2P网络几乎同时出现，无明确界定，大致分类：

- ◆混合式P2P网络：C/S、P2P模式的混合

- ◆无结构P2P网络：分布/松散的结构

- ◆结构化P2P网络：准确、严格的结构

∞ P2P网络实质上指TCP/IP模型(四层模型)中应用层的工作方式，包括结点工作模式与计算模式等；下面三层通常采用标准、单一的工作方式，本身并没有集中式与分布式之分，只是为应用层不同的工作方式提供底层的服务支持。



# P2P网络

## ∞P2P网络的发展

∞ P2P网络的核心机制，是在应用层建立逻辑上的覆盖网络(overlay network)，屏蔽下三层的工作细节。

◆设计和实现P2P网络应解决的基本问题

✱路由和定位、查询和搜索、动态结点算法、容错性



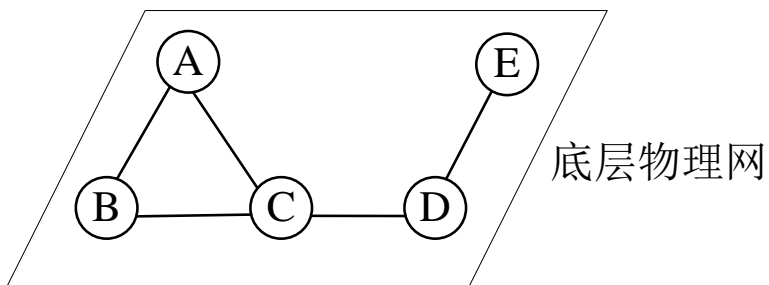
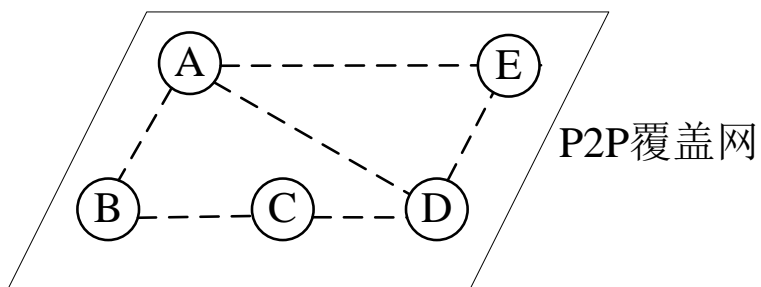
# P2P网络

## ∞ P2P网络的发展

### ∞ P2P网络的增强机制

◆数据复制、缓存、分片；负载均衡；拓扑一致性；匿名、声誉、信任、安全性

◆拓扑不一致问题



如：覆盖网上  $A \rightarrow E$

实际上：  $A \rightarrow C \rightarrow D \rightarrow E$





# P2P网络

## ∞P2P网络的发展

∞ 第一阶段：1999 ~ 2000，应用→研究

- ◆1999年，18岁的Shawn Fanning发布Napster软件，立刻风行，为Hybrid P2P体系的代表，因版权问题2001年关闭网站。
- ◆2000年，第一个无结构P2P网络Gnutella（很快关闭），与Napster目标同为交换文件。
- ◆几乎同时“自由、安全、匿名”的Freenet目标是信息共享平台。
- ◆KaZaA（无结构）引入超结点，开发结点异构性。



# P2P网络

## ∞P2P网络的发展

∞ 第一阶段：1999 ~ 2000，应用→研究

◆eDonkey：文件分块、同一文件可并行下载，网络结构类似KaZaA。

◆2000年8月，出版商O' Reilly组织P2P峰会目的：诠释P2P的意义，描述P2P的作用，消除P2P的负面影响。

◆同时，Intel成立P2P工作组，02年发布.NET基础架构上的P2P Accelerator Kit及P2P安全软件包，可建立P2P安全Web应用；IBM、HP等也利用P2P技术进行开放存储与网络共享打印。



# P2P网络

## ∞P2P网络的发展

∞ 第二阶段：2001 ~ 2003，学术→商用

- ◆2001年，产生结构化structured P2P体系
- ◆IEEE成立P2P专业会议，ACM发表经典论文
- ◆P2P经典模型及应用体系：Chord, CAN, Tapestry, Pastry, CFS, OceanStore, PAST
- ◆学术团体及技术组织成立或完善了专门的P2P研究组，MIT, U.C. Berkeley, Microsoft/Rice Univ., Stanford
- ◆2002年，P2P专业会议IPTPS及其它如SIGCOMM等网络通信、分布式系统领域的重要会议关注P2P



# P2P网络

## ∞P2P网络的发展

∞ 第二阶段：2001 ~ 2003，学术→商用

◆2002年10月BitTorrent推出，2003年世界范围流行(尤其是在中国)。BitTorrent使用基于文件的分布式服务器，共享同一文件的用户构成一个独立的子网，单点失效不会影响整个网络，网络不易被关闭；分片优化、阻塞控制从而充分利用网络资源。它的快速文件分发下载的特性引起网络多媒体文档下载的一个浪潮。

◆2003年实质进入一个稳定期，在解决了P2P网络最核心的问题后，学术界将重点放在其性能增强，安全性、实用系统开发上。



# P2P网络

## ∞P2P网络的发展

∞ 第三阶段：2004~迄今，共识→实用

◆P2P网络的主要问题已解决，核心机制、整体架构已形成，在重大问题形成共识的基础上，追求g更细节、更高效、更实用，试图整合不同的P2P应用系统（类似web推出后的应用）

◆05年底，Springer在其LNS(Lecture Notes in Computer Science)系列中出版Peer-to-peer system and application, 提供了对P2P领域权威、全面的总结与展望 <http://www.informatik.uni-trier.de/~ley/db/conf/p2p/p2p2005lns.html>

◆P2P目前更侧重应用拓展以及与其它领域的研究结合



# P2P网络

## ☞ P2P网络的定义和概念

- ☞ 1、Peer-to-peer is a **type of Internet network** allowing a group of computer users with the same networking program to connect with each other for the purposes of directly accessing files from one another's hard drives.
- ☞ 2、Peer-to-peer networking (P2P) is **an application** that runs on a personal computer and shares files with other users across the Internet. P2P networks work by connecting individual computers together to **share files** instead of having to go through a central server.



# P2P网络

## ∞ P2P网络的定义和概念

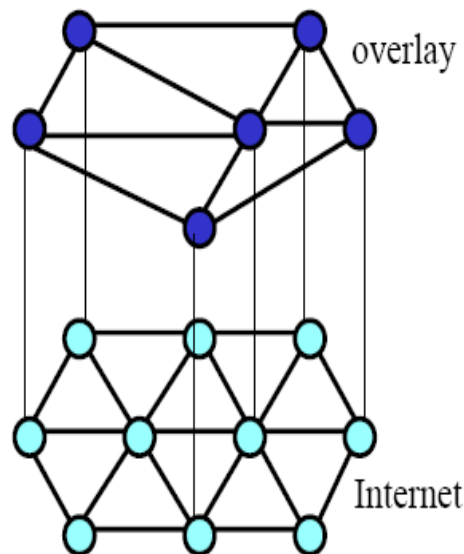
- ∞ 3、P2P是一种分布式网络，网络的参与者共享他们所拥有的一部分硬件资源（处理能力、存储能力、网络连接能力、打印机等），这些共享资源需要由网络提供服务 and 内容，能被其它对等节点(Peer)直接访问而无需经过中间实体。在此网络中的参与者既是资源（服务和内容）提供者（Server），又是资源（服务和内容）获取者（Client）。
- ∞ P2P is a **special distributed system** on the application layer, where each pair of peers can communicate each other through the routing protocol in P2P layers.



# P2P网络

## ☞ P2P网络的定义和概念

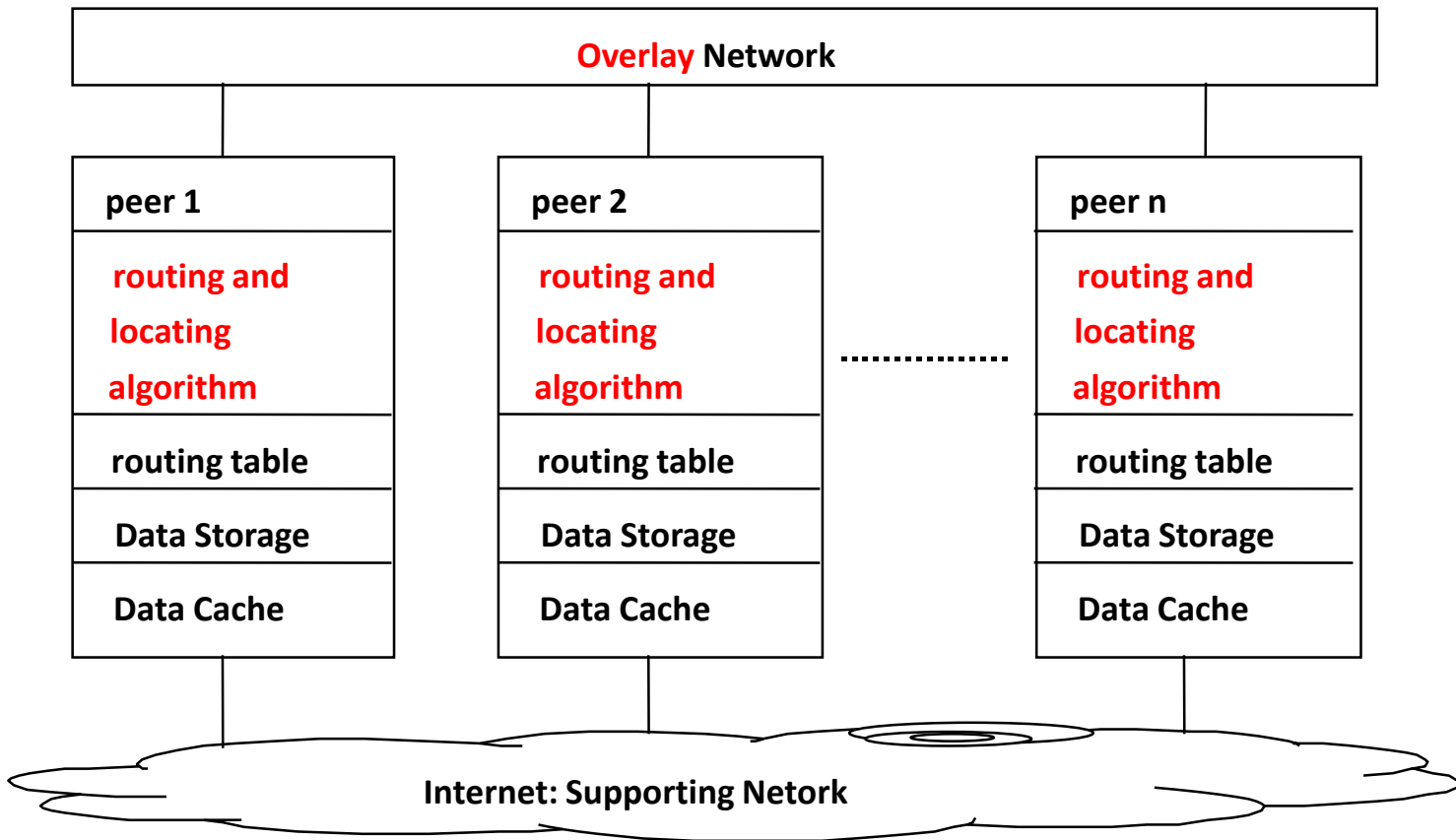
- ☞ **Overlay网络**（为实现某一应用目标，若干peer互联形成一个overlay network）。**在现有网络体系结构上新加一层。**
- ☞ Overlay网络是在**应用层之上**的网络。由对等点连结而成的网络。
- ☞ 对等联网模式的理念——在资源共享中，表现为 “**人人为我，我为人人**” 的思想。





# P2P网络

## Overlay网络模型



A Generic Topological Model of P2P Systems



# P2P网络

## ∞P2P网络的典型应用

- ◆ 文件共享
- ◆ 媒体播放
- ◆ 数据存储
- ◆ 分布计算等



# P2P网络

## ☞ P2P网络的优势

### ☞ 提高网络工作效率

- ◆ 对比C/S，结点间的协作效率更高。
- ◆ 结构化P2P有严格拓扑结构，基于DHT，将网络结点、数据对象高效均匀地映射到覆盖网中，路由效率高。

### ☞ 充分利用网络带宽

- ◆ P2P不通过服务器进行信息交换，无服务器瓶颈，无单点失效，充分利用网络带宽，如BT下载多个文件，可接近实际最大带宽，HTTP及FTP很少有这样的效果。



# P2P网络

## ∞P2P网络的优势

### ∞ 开发了每个网络结点的潜力

- ◆ 结点资源：计算能力及存储容量。
- ◆ 个人计算机并非永久联网，是临时性的动态结点，称为“网络边缘结点”。
- ◆ P2P使内容“位于中心”转变为“位于边缘”，计算模式由“服务器集中计算”→“分布式协同计算”。



# P2P网络

## ∞ P2P网络的优势

### ∞ 具有高可扩展性(scalability)

- ◆ 可扩展性衡量，当网络结点总数增加时：结点负载如何改变、为适应规模扩大而需要增加的额外设备的数量、任意两个网络结点通信效率如何改变（尤其是路由效率）。
- ◆ P2P网络中，结点间分摊通信开销，无需增加设备，路由跳数增量小。

### ∞ 良好的容错性

- ◆ 冗余方法：空间换取容错。
- ◆ 周期性检测：时间换取容错(结点自适应状态维护)。



# P2P网络

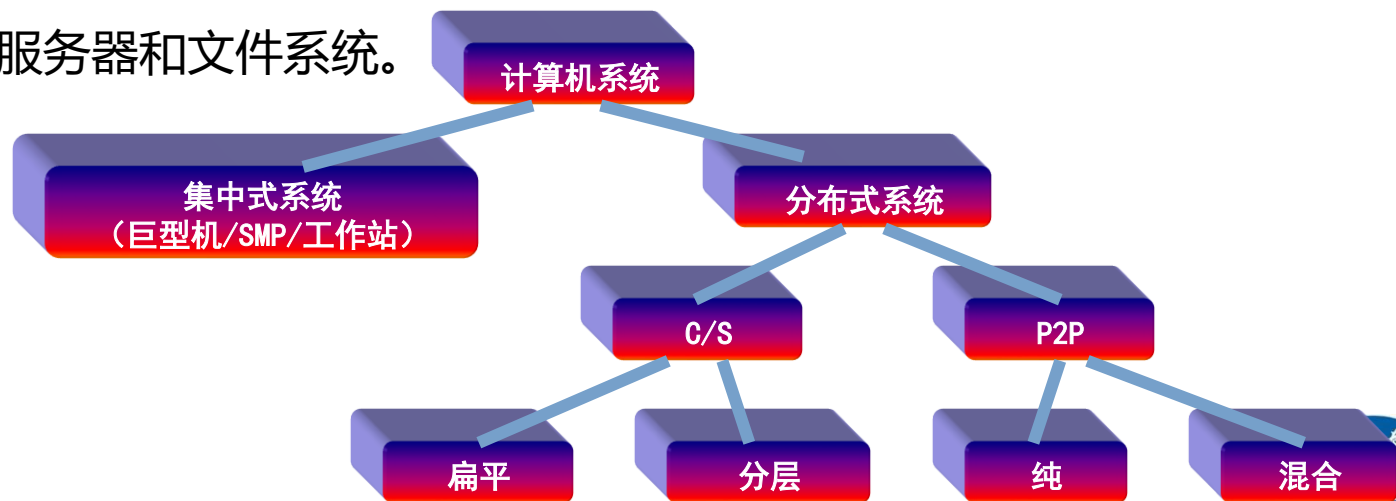
## 计算机系统的架构划分

所有的计算机系统可分为集中式和分布式两类

分布式可进一步划分为C/S和P2P模式

C/S模式可划分为

- ◆ 扁平：所有的客户端仅仅和单个服务器（含重复服务器）通信，如传统的中间件；
- ◆ 分层：提高可扩展性，某层的服务器又作为更高层的客户端：如DNS服务器和文件系统。



# P2P网络

## ∞P2P体系结构（分三类）

### ∞ 1、完全集中式

- ◆ 研究目标及重点是应用模式从**C/S模式向对等模式**的转变。
- ◆ 优点：应用模式消除了应用服务器的瓶颈问题并缓解了应用流量的不均衡性，在目录服务器获取资源索引信息之后的所有数据的交换都是在节点间完成的。简单易部署。可以模糊查询。
- ◆ 缺点：**单点失效**。尽管可以用并行服务器解决。
- ◆ 拓扑结构：非结构化、集中式。
- ◆ 典型代表：Napster



# P2P网络

## ∞ P2P体系结构（分三类）

### ∞ 2、非集中式、非结构化

- ◆ 研究目标和重点是**去除体系结构上的单点失效**问题。对象查询是分布式的。查询是逐跳的，泛洪式直到成功或失败或超时。
- ◆ 优点：自组织的管理模式使得整个系统的鲁棒性得以大幅度增强。可以模糊查询。
- ◆ 缺点：**消息传递（泛洪、回溯等）的资源定位模式**制约了网络规模的可缩放性。查询效率低。
- ◆ 典型代表：Gnutella、Freenet、KaZaA
- ◆ 拓扑结构：非结构化、非集中式、无规则分布式



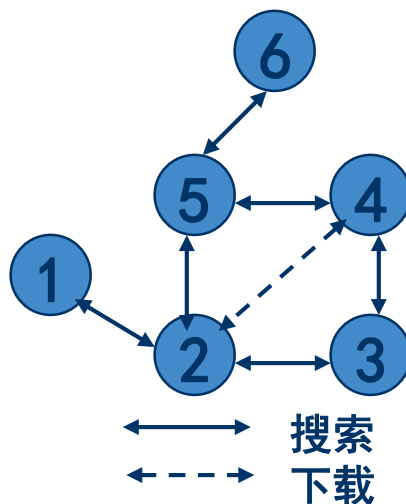


# P2P网络

## ☞ 泛洪请求模式

### ☞ 过程

- ◆ 每个Peer的请求直接广播到连接的Peers
- ◆ 各Peers又广播到各自的Peers
- ◆ 直到收到应答或 达到最大泛洪步数(典型5-9)



# P2P网络

## ☞ 泛洪请求模式

### ☞ 特点

- ◆ 无广告性共享资源
- ◆ Gnutella 使用该算法,限于公司内通信有效
- ◆ 大量请求占用网络带宽,可扩展性并不一定最好

### ☞ 改进

- ◆ Kazaa 设立Super-Peer客户软件,以集中大量请求
- ◆ BT 文件分块
- ◆ Cache最近请求



# P2P网络

## ∞P2P体系结构（分三类）

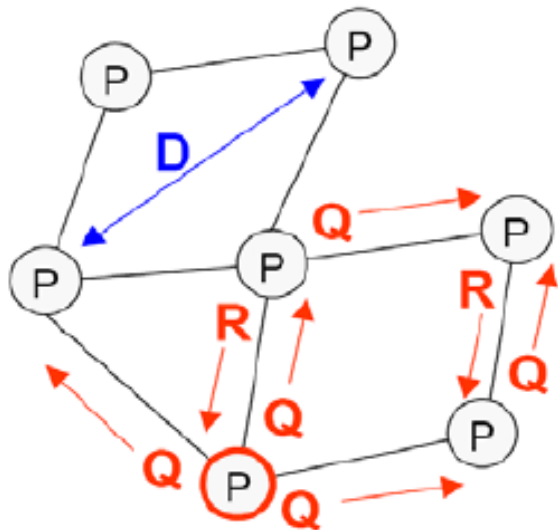
### ∞ 2、非集中式、非结构化

- ◆ 典型代表：Gnutella
- ◆ 2000年3月，Nullsoft公司
- ◆ Justin Frankel & Tom Pepper: Winamp发明人
- ◆ 版权问题→上线一个半小时关闭
- ◆ 无结构P2P系统代表
- ◆ **其思想和代码被复制、改写、继承**



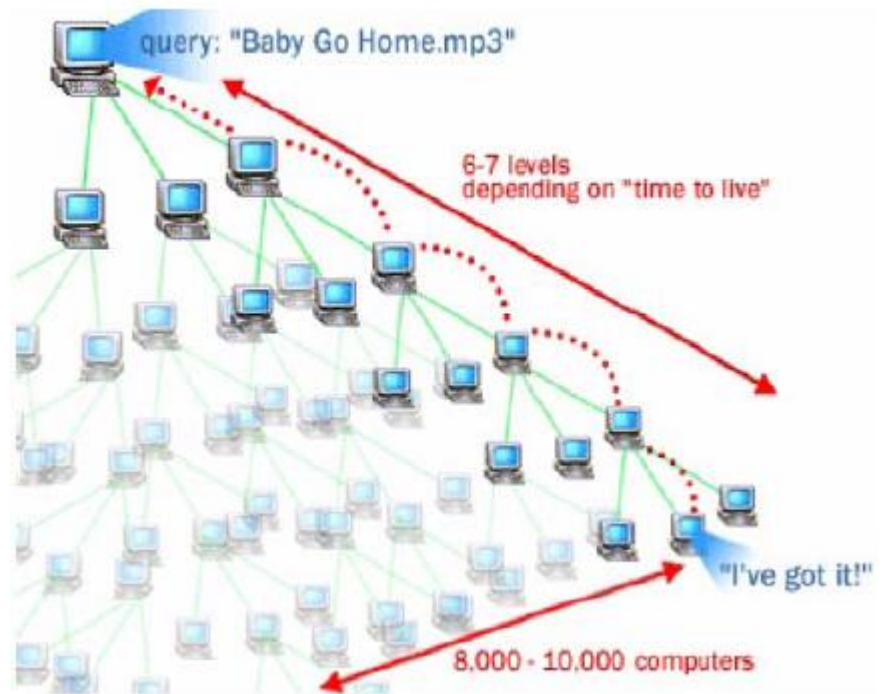
# P2P网络

## ∞Gnutella运行原理、泛洪问题



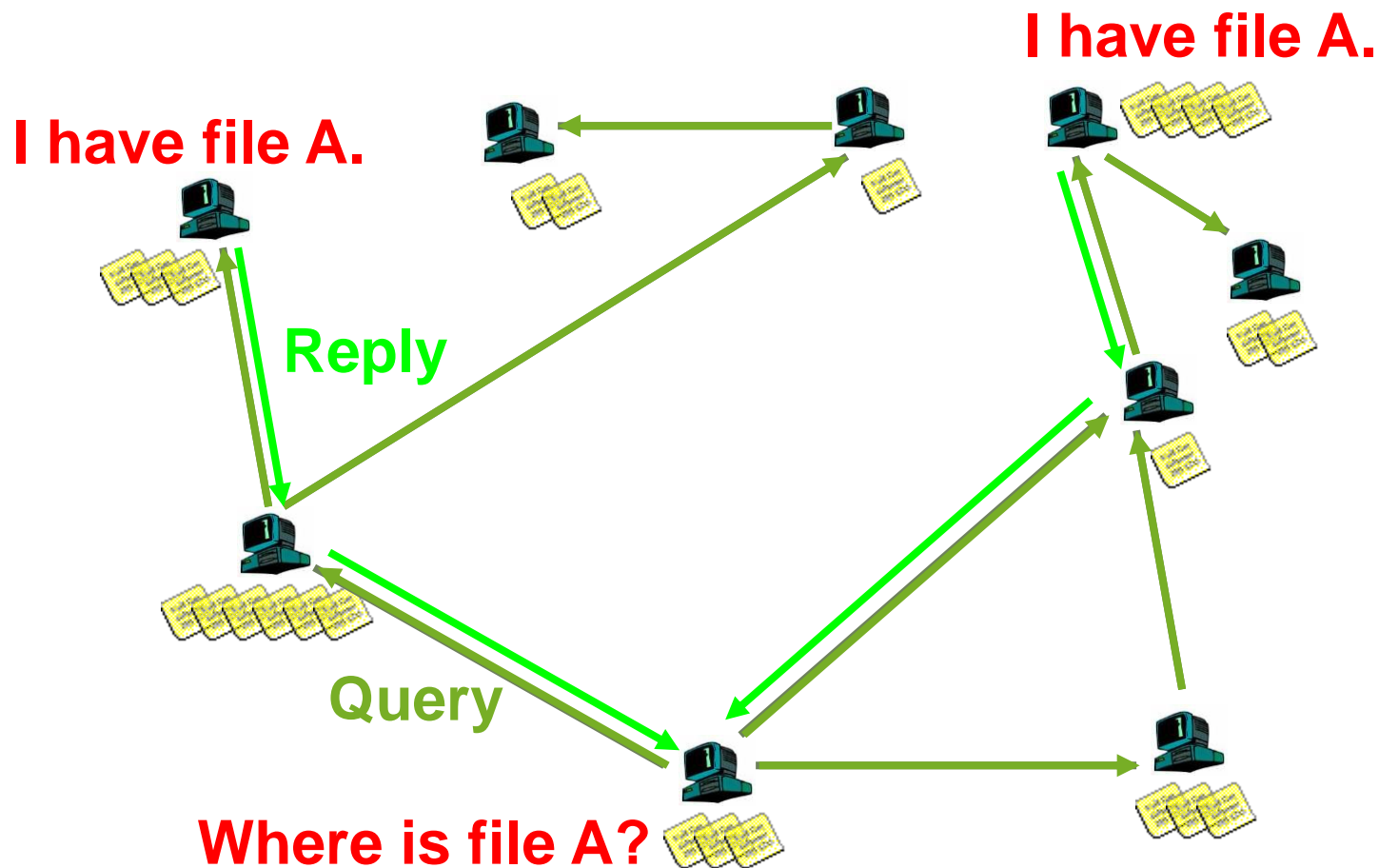
(P) peer      Q query  
(S) server    R response

D file download



# P2P网络

## ∞Gnutella运行原理、泛洪问题



# P2P网络

## ∞P2P体系结构（分三类）

### ∞ 3、非集中式且结构化

- ◆ 需要解决的问题则是如何增强网络规模的可缩放特性。对象查询也是分布式的。使用**DHT技术**构造结构化拓扑。对象的查询也是逐跳的执行，经过确定的步跳可以确信是成功的。
- ◆ 拓扑结构：非集中式、**结构化**。
- ◆ 典型P2P网络如：CAN、Chord、Oceanstore等。
- ◆ 优点：在资源管理过程中同时拥有自组织特性、规模的强可缩放特性以及部署的廉价性等等。为规模庞大的资源整合及共享提供了可能性。
- ◆ 缺点：节点仅存在局部视图。缺少权威第三方的控制。不支持模糊查询。



# P2P网络

## ☞文档路由模型(Document Routing Model)

- ☞ 这种文件路由模型需要用**分布式哈希表(Distributed Hash Tables, DHT)**,这是**有结构对等网络**采用的搜索方法,也是有结构和无结构对等网络的**根本区别**。是确定性的算法。
- ☞ 在这种模型下,每个对等体都有一个ID,每个文件有一个关键字Key,当宣告一个关键字为K1的文件时,先通过哈希映射得到对应的 $K1 \rightarrow ID1$ ,然后将该文件存到ID号为ID1的节点,文件的存放过程需要将文件路由到该节点ID1。反过来,当查找一个关键字为K1的文件时,先进行哈希映射得到 $K1 \rightarrow ID1$ ,然后将该文件从ID号为ID1的节点上取到该文件,从该网络中取文件需要将请求消息路由到ID1节点,然后文件从ID1节点原路返回。



# P2P网络

## 文档路由模式

### 过程

- ◆ 每个网上Peer**分配一个随机ID**,并知道其他Peers的给定号码
- ◆ 当共享文件发布到系统上时,根据文件名字和内容Hash成为ID
- ◆ 每个Peer将根据**该ID向该文件路由**
- ◆ 该过程重复执行,直到最近的PeerID是现行Peer的ID
- ◆ 每个路由操作还**保持文件副本在本地**
- ◆ 当Peer请求某文件时,该请求将用**该文件的ID到达Peer**,过程重复直到发现文件副本,最终文件下载到请求源端





# P2P网络

## ∞四大结构化模型

∞ 2001年, SIGCOMM (网络通信顶尖会议)

-- Chord: Ion Stoica等 (Berkeley、MIT)

-- CAN: Ratnasamy等 (Berkeley、AT&T)

∞ 2001年, 其它两个模型

-- Pastry: Rowstron等 (微软、Rice)

-- Tapestry: 赵燕斌等 (Berkeley)



# P2P网络

## ☞四大算法实现文件路由

☞ Chord/CAN/Tapestry/Pastry

☞ 目标相同

- ◆ **减少**路由到指定文件的P2P跳数
- ◆ **减少**每个Peer必须保持的**路由状态**

☞ 算法异同

- ◆ 都保证算法的跳数与Peer群组的大小相关
- ◆ 或都指出算法能以高概率完成
- ◆ 方法上的差别很小



# P2P网络

## ∞四大算法实现文件路由

∞ Chord/CAN/Tapestry/Pastry

∞ Chord

- ◆ 每个Peer保持**LogN其他Peer**的踪迹(N是群组的全部Peer数)
- ◆ 当Peer加入或离开时,高优化算法版本仅需关注LogN个Peers的变化



# P2P网络

## ∞四大算法实现文件路由

∞ Chord/CAN/Tapestry/Pastry

∞ CAN

- ◆ 每个Peer保持**少于 $\text{Log}N$** 个其他Peers的踪迹
- ◆ 在插入和删除时仅这些Peers受影响
- ◆ 其路由表较小,但到达的路径较长
- ◆ 可能**更适合动态通信**



# P2P网络

## ∞四大算法实现文件路由

∞ Chord/CAN/Tapestry/Pastry

∞ Tapestry与Pastry很相似

◆ 除减少跳数外,还积极削减每个P2P跳上的时延



# P2P网络

## ☞ 路由表

### ☞ 路由表内容

- ◆ id - 文件标识符
- ◆ next\_hop - 存储文件id的另一个节点
- ◆ file - 保存在本地的id标识文件

<i>ID</i>	<i>Next_hop</i>	<i>file</i>
	...	

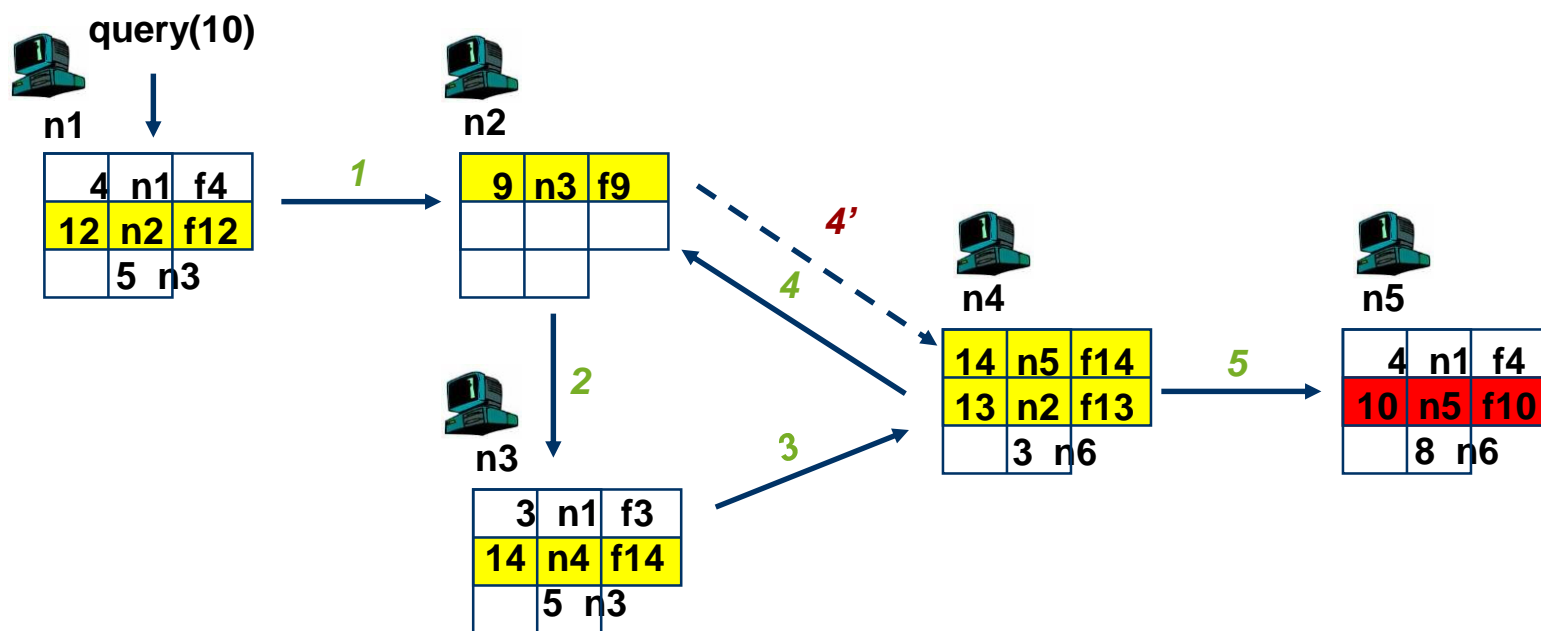
### ☞ 搜索过程

- ◆ 如果文件id存储在本地，停止搜索，上传文件
- ◆ 如果不在本地，**搜索路由表中最接近的id**，将请求**转到next\_hop**
- ◆ 如果所有节点都没有找到，返回失败，返回路由表中**下一个最接近的id**



# P2P网络

## 文件路由原理



- 网络趋向于一个小世界—small world, 类似六度分隔 (Six Degrees of Separation) 理论
- 因此, 大部分查询只需经过少量跳数



# P2P网络

## ∞四大算法实现文件路由

### ∞ 结构化重叠路由

- ◆ **加入**：开始时，联系一个“bootstrap”节点，加入分布式数据结构，**获得一个节点id**
- ◆ **发布**：向数据结构中最近的节点发布**文件id的路由信息**
- ◆ **搜索**：向路由表中最近的节点查询文件id，数据结构保证查询会找到发布节点
- ◆ **获取**：两个选项
  - \* 查询到的节点保存有文件，则从查询结束的节点获取
  - \* 查询到的节点返回结果：节点x有文件，则从节点x获取





# P2P网络

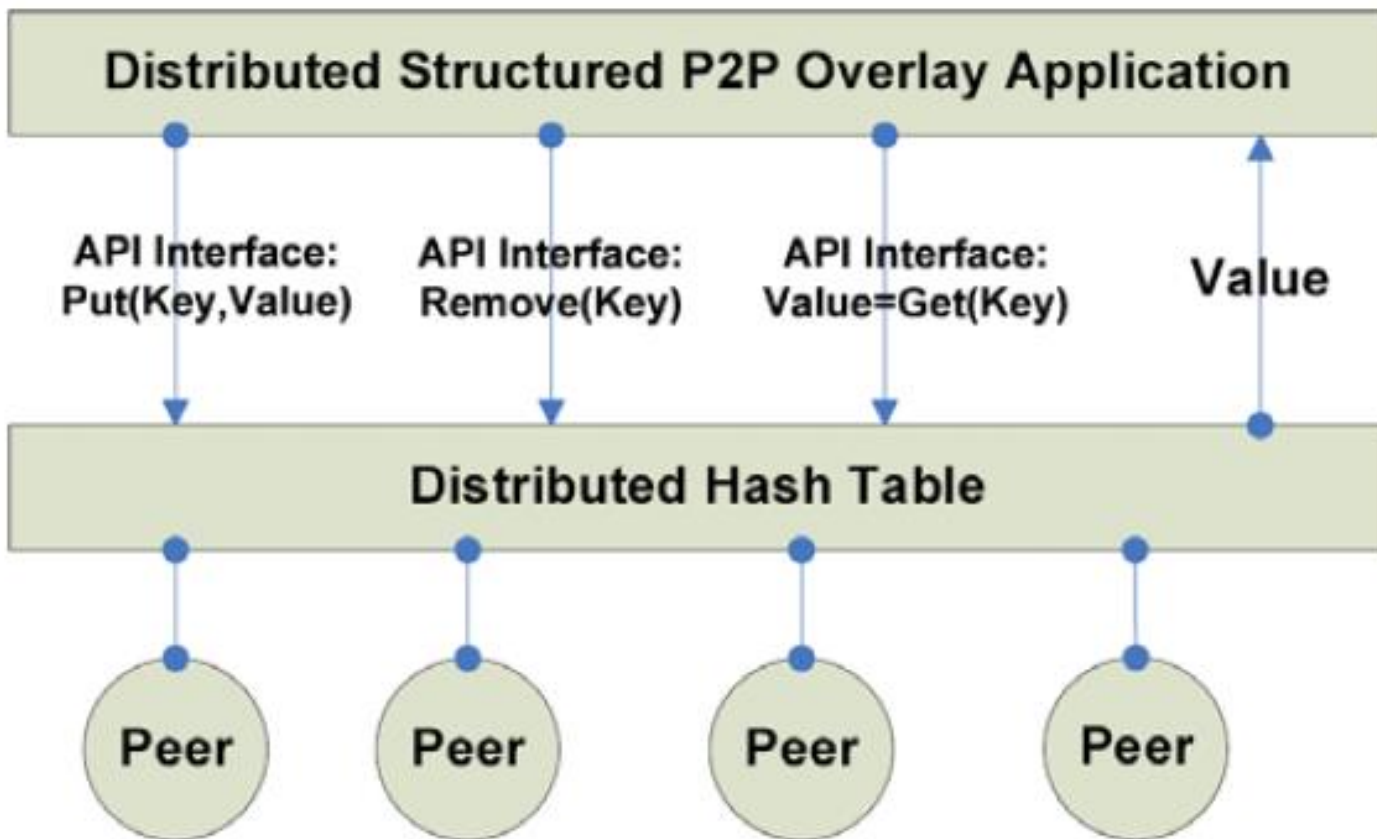
## 第二代P2P

- ◆ **没有集中的目录服务器**，但是**拓扑结构**有意义。这个结构意味着P2P网络拓扑被紧紧的控制。比如： Mesh, Ring , d-dimension Torus, K-ary tree。
- ◆ 使用**DHT技术**，有较好的可伸缩性和查询效率。提供负载均衡和确定性的搜索保证。但是容错性或弹性不好，尤其是在恶意攻击下。文件不是被 随机地而是以特定的位置放置，这样使得连续的查询更加容易满足。
- ◆ 它使用精确的定位算法和特殊的路由协议使得搜索效率提高。支持精确查询不能支持模糊查询。



# P2P网络

## 第二代P2P——DHT



# P2P网络

## 第二代P2P的代表

- ◆ Tapestry :用于覆盖网络的定位和路由机制。Tapestry具有自我管理、容错和灵活平衡负载等特点。
- ◆ Pastry: 是一个路由定位协议, 与Tapestry有许多相似性。
- ◆ CAN: 可以在Internet规模的大型对等网络上提供类似哈希表的功能, 提供查询服务。CAN具有可扩展、容错和完全自组织等特点。。
- ◆ CHORD: 是一个分布式的非集中式的P2P查询服务, 存储关键字/值对。给定一个关键字 (key), 将key映射到某个结点。如果对等网络应用的每个数据都分配一个key, 那么对等网络中的数据查找问题就可以用Chord很容易地解决了。



# P2P网络

## ∞P2P查询算法

- ◆ 目标查询的算法研究是P2P网络的关键部分。算法的优劣决定了P2P网络性能的优劣。
- ◆ 随机地和确定性地，非泛洪和泛洪式的，集中式的和非集中式的。
- ◆ 除了Napster集中式的查询外，所有其他的P2P网络使用分布式的和逐跳的转发请求查询方式。



# P2P网络

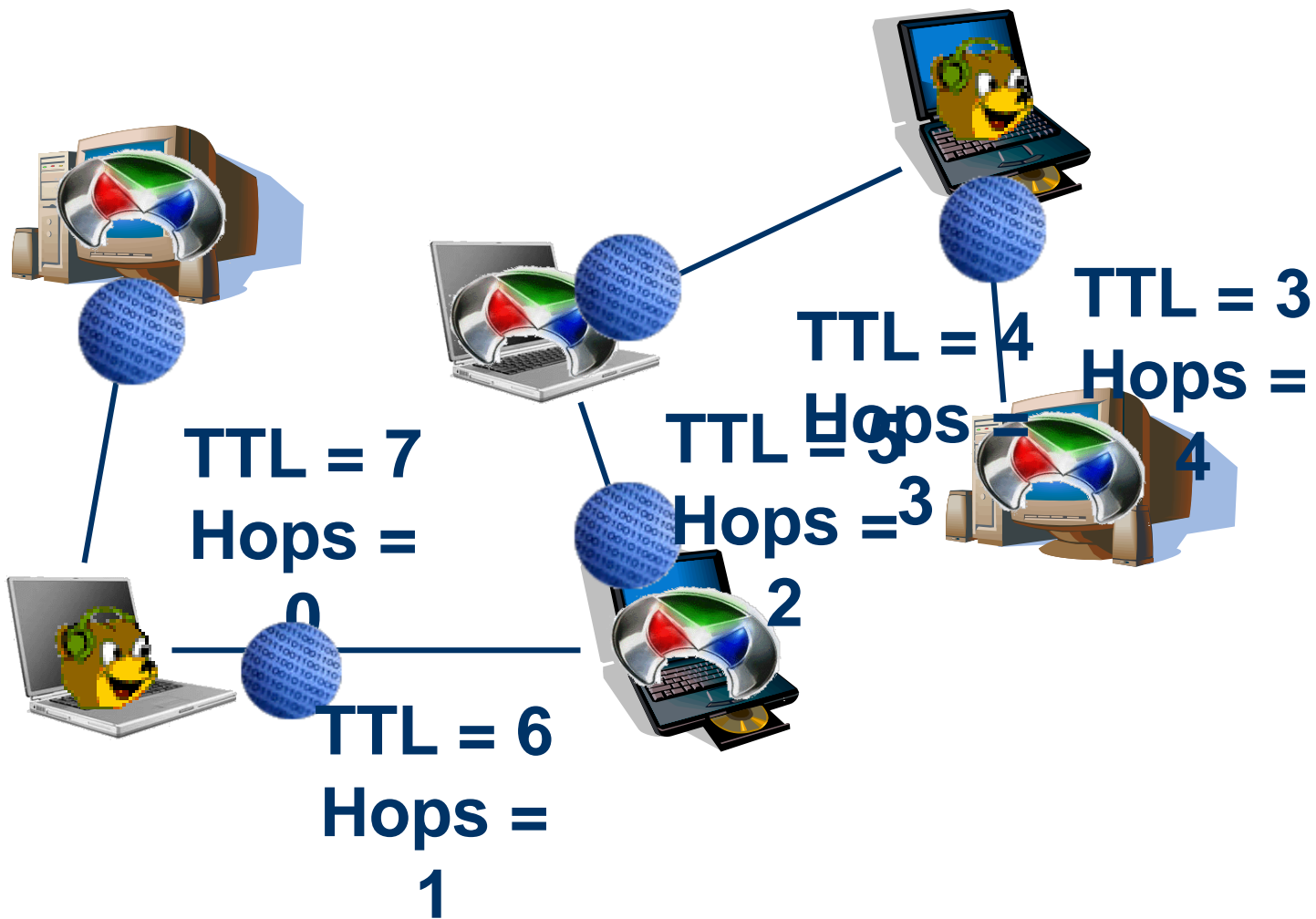
## ∞P2P查询算法

- ◆ **随机查询**被转发到一个随机选择的邻居节点。通常有一个较长的路径长度。应用于非结构化P2P网络中。如Gnutella。
- ◆ **泛洪式查询方法**是在纯粹分布式结构中采用的方法,不需要向目录服务器报告共享的信息,而是将请求泛洪到直接相连的邻居,直到收到响应,或者达到了最大的泛洪步数。由于这种模型需要很多的网络带宽来进行资源的搜索工作,因而这种模型的扩展性并不是很好,但是在像公司这样的小型团体里还是很有效的。
- ◆ 为了解决扩展问题,引入了**混合式**结构,把查询请求集中到超级节点,这样就减少了网络带宽的消耗。



# P2P网络

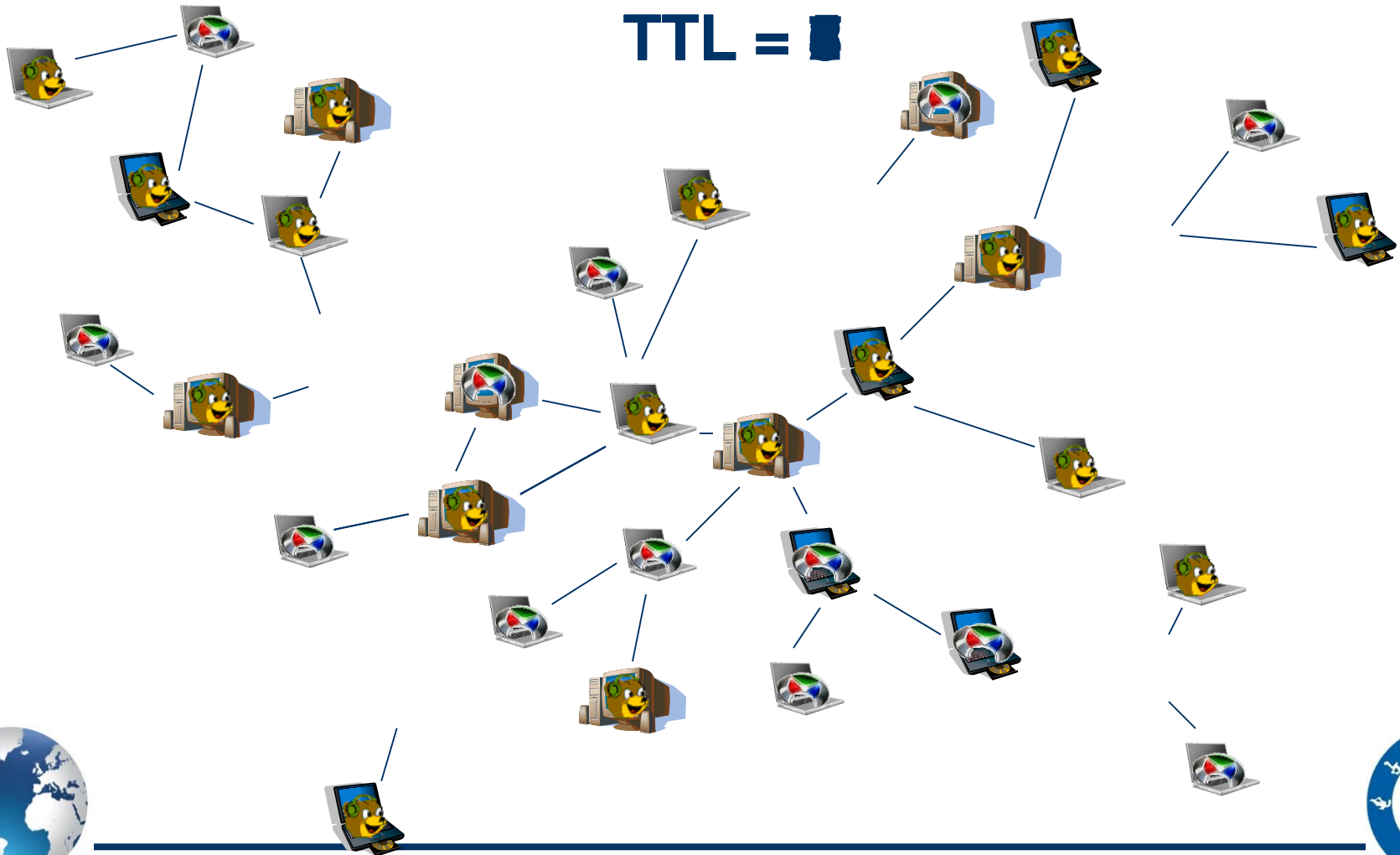
## ⌘ P2P查询算法——TTL/Hops Example



# P2P网络

## ∞P2P查询算法——TTL/Hops Example

TTL = 4



# P2P网络

## ☞ 泛洪式算法的改进

☞ 有各种方法可以控制和减少在分布式无结构P2P网络中基于泛洪搜索方法带来大量不必要流量,这大致可归为三类

- ◆ 基于改进转发机制的方法;
- ◆ 基于缓存的方法;
- ◆ 拓扑结构的优化。
- ◆ 这些改进方法会使得**无结构P2P网络**在将来会有更广阔的应用。





# P2P网络

## ∞P2P网络中的安全问题

- ◆ P2P作为分布式的网络模型,在系统安全中面临着巨大的挑战,它需要在没有中心节点的情况下,提供身份验证、授权、数据信息的安全传输、数字签名、加密等机制。还要避免大量信息传输拥塞、病毒的入侵和恶意代码攻击等问题。
- ◆ P2P网络的安全问题,归结为以下三类: 1)数据安全问题,包括数据的完整、真实和保密; 2)网络对等节点安全问题,对等机相互信息和个人环境安全; 3)信息安全问题,网络信息内容和版权的管理。
- ◆ 研究内容: 上述三类问题即可以作为研究对象。



# P2P网络

## ∞P2P网络中的信任问题

- ◆ 在纯分布式对等网络中,没有中心控制,具有高度的动态性、自治性和异构性,即:每个用户参与网络是随机的、自愿的,并且不同的用户有不同的能力和可靠性,由此导致不可靠的服务质量及大量欺诈行为的存在,网络的可用性差。目前解决这一问题的有效方法是在P2P网络中建立信誉体制。
- ◆ 所谓信誉体制是指用户通过自己的过去经历或他人的推荐来选择符合自己要求的交互端的一种体制。此体制能激励用户提供可靠高质量的服务,节省时间和通信开销,减少交互风险和损失,促进网络的良性发展。



# P2P网络

## ∞P2P网络中的激励机制

- ∞ 在P2P 系统中所有节点往往更多地表现出自兴趣和理性,单个节点的目标往往是使自身的网络效用 最大化,这样导致了以下问题:
  - ◆ (1) 搭便车( free riding)问题, 70%的Gnutella用户不共享任何文件,接近50%的 文件查询命中仅来自1%的Gnutella用户;
  - ◆ (2) 公共的悲剧 ( tragedy of common)问题,即网络资源作为非排他性占有的公共资源,被大多数 P2P节点无节制地使用.
- ∞ 如何使节点根据自身的期望,主观上愿意参与共享其资源不仅是困扰P2P系统的问题,同样也严重 困扰着大规模计算资源共享系统.有论文指出了使用微支付(micro payment)和虚拟货币( virtual currency)的方法来解决P2P文件共享网络中的共享激励问题。



# P2P网络

## ∞ P2P网络中的其他问题

### ∞ 资源放置

在P2P系统中，并非个人资源（比如数据）都放置在各自的机器上，很可能是所有机器共同管理资源，比如在P2P存储系统中经常采用分布式哈希表放置数据，各人数据可能放置在他人的机器上，于是如何进行资源放置就成了必须回答的第一个问题。



# P2P网络

## ∞ P2P网络中的其他问题

### ∞ 资源定位

数据的查找与资源放置方法是直接相关的。对于以DHT方式放置的数据，可以直接定位，但在多数文件共享系统中，用户的文件都是放在各自的机器上，如何知道哪些机器放有用户需要的数据就成为一个关键问题，常常需要较大规模的搜索才可以完成。资源定位就是研究如何更有效率地找到需要的资源所处的位置，尤其是一些在网络中稀有（rare）的数据。



# P2P网络

## ∞ P2P网络中的其他问题

### ∞ 资源获取

资源定位后就需要获得资源，有些资源并不能直接获得，比如计算资源、大文件、流媒体资源。这里的问题主要在于如何才能更高效的获取资源，或者说如何使一些热点资源服务更多的需要该资源的用户，通常这需要尽量发挥P2P系统中所有参与者的能力。而使用P2P模式充分地利用了所有结点的带宽资源，使得并行下载能力得到了极大的扩展。



# P2P网络

## ∞性能

### ∞ P2P系统目标:

- ◆ 聚合分散网络上的存储容量(Napster/Gnutella)和计算周期(SETI@home) 来改进系统的性能

### ∞ 影响非集中化性能三类资源

- ◆ 处理/存储/网络
- ◆ 网络资源中的带宽和时延是主要因素



# P2P网络

## ∞性能

### ∞ 中心协调系统(Napster SETI@home)

- ◆ Peers的协调和仲裁通过中心服务器进行
- ◆ 混合P2P以克服这些脆弱点

### ∞ 非集中协调系统(Gnutella Freenet)

- ◆ 用消息传递机制搜索信息和数据
- ◆ 查询搜索的带宽与发送消息数





# P2P网络

## ∞性能优化的关键技术

### ∞ 复制(Replication)

- ◆ 把对象/文件的拷贝放在请求Peers附近,最小化连接距离
- ◆ 改变数据时必须保持数据拷贝的一致性



# P2P网络

## ∞性能优化的关键技术

### ∞ 高速缓存(Cache)

- ◆ 减少获取文件/对象路径的长度,进而减少Peers间交换消息数
- ◆ 这一减少很有意义-Peers间通信时延是严重的性能瓶颈
- ◆ Freenet:命中文件传播到请求者途中所有节点高速缓存它
- ◆ 目标是最小化时延,最大化请求吞吐率,很少高缓大数据



# P2P网络

## ∞性能优化的关键技术

### ∞ 智能路由和网络组织

- ◆ 社交“小世界”现象, 6熟人链找到生人
- ◆ 局部搜索策略, 代价与网络规模成子-线性增加
- ◆ OceanStore/Pastry网络上积极移动数据提高性能

