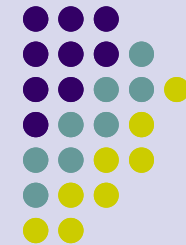


Úvod do P2P sítí



Přednášky z Distribuovaných systémů
Ing. Jiří Ledvina, CSc.

Úvod



- Základní rozdělení P2P systémů
 - sdílející obsah
 - vyhledávací obsah
- Sdílení obsahu
 - přímý přenos mezi uzly – identické pro všechny uzly
 - strukturované nebo nestrukturované ukládání dat
 - automatická replikace dat

Úvod



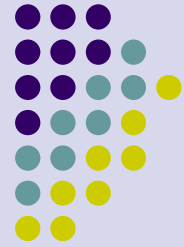
- Jiné dělení P2P systémů
 - Pro paralelní výpočty
 - Mnohonásobné výpočty
 - Rozdělené výpočty
 - Přístup k souborům a informacím
 - Sdílení obsahu
 - Přístup k souborům
 - Filtrování, vyhledávání
 - Kooperující (RT)
 - Instant messaging
 - Sdílené aplikace
 - hry

Úvod



- Rozdělení podle decentralizace
 - Čisté P2P systémy
 - Členové jsou servery i klienti (servent)
 - Bez centrálního serveru
 - Bez centrálního směrovače
 - Hybridní P2P
 - Centrální server pro udržování informací o členech
 - Odpovídání na dotazy (indexy)
 - Členové udržují informaci (soubory)
 - Směrování pomocí převodu reference – adresa
 - Kombinované
 - obojí

Vyhledávání v P2P sítích



- Vyhledávání obsahu
 - centralizované
 - decentralizované
 - založené na pravděpodobnostním vyhledávání



P2P směrování

- P2P overlay networks
 - Členové jsou uzly
 - Sousední členové jsou propojeni virtuálními hranami (TCP, IP adresa)
 - Změna topologie
 - Nový uzel musí nabootovat informaci
 - Periodický ping
 - Test životaschopnosti při přenosu zpráv
 - Rekonfigurace – přidávání uzlů, ztráta souseda

Typy P2P



- P2P systémy
- Nestrukturované
 - Napster (centralizované)
 - Gnutella (distribuované)
 - Kazaa/FastTrack (hierarchické)
- Strukturované
 - Chord – uzly v kruhu
 - Pastry – overlay síť, ID
 - Scribe - kruh
 - CAN



Model klient/server

- Architektura klient/server
 - server musí být výkonný, spolehlivý, známý zdroj dat
 - klienti požadují data od serverů
 - často používaný, úspěšný model
- Omezení architektury klient/server
 - těžko dosažitelná škálovatelnost
 - představuje úzké místo z hlediska chyb
 - vyžaduje administraci
 - mohou existovat nevyužitá zdroje
 - o řešení těchto problémů se snaží P2P sítě

Model peer-to-peer



- P2P počítání
 - P2P počítání je sdílení výpočetních zdrojů a služeb přímou výměnou mezi systémy
 - Tyto zdroje a služby zahrnují výměnu informace, cykly zpracování, vyrovnávací paměti a diskové paměti pro ukládání souborů
 - P2P počítání využívá existujících výpočetních zdrojů, pamětí, propojení počítačů a dovoluje uživatelům využívat společný výpočetní výkon ku prospěchu všech



Architektura P2P sítí

- Architektura P2P sítí
- všechny uzly jsou jak klienti, tak i servery
 - zpracovávají a spotřebovávají data
 - jakýkoliv uzel může zahájit spojení
- neexistuje centrální zdroj dat



Charakteristiky P2P sítí

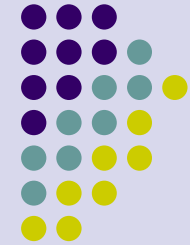
- Charakteristiky P2P sítí
 - klienti jsou současně servery i směrovače
 - uzly se podílí úložišti dat, paměti, CPU
 - uzly jsou autonomní (bez administrace)
 - síť je dynamická : uzly se mohou často připojovat i odpojovat
 - uzly spolupracují přímo, každý s každým (ne prostřednictvím serverů)
 - uzly mohou být velice odlišné



Vlastnosti P2P sítí

- Výhody P2P sítí
 - efektivní využívání zdrojů
 - využívají nevyužité zdroje (paměť, CPU) v síti
 - škálovatelnost
 - možnost replikace
 - geografická distribuovanost
 - neexistuje centrální bod (chyby)
 - jednoduchá administrace
 - uzly se organizují samy
 - není třeba dislokovat servery aby byly pokryty požadavky
 - systému je vlastní odolnost proti chybám, replikace, vyrovnání zátěže

Aplikace P2P sítí



- Aplikace P2P sítí
 - sdílení souborů (Napster, Gnutella, Kazaa)
 - síťové hry (DOOM)
 - sdílené aplikace (ICQ, sdílené tabule)
 - distribuované výpočty (seti@home)

Aplikace P2P sítí



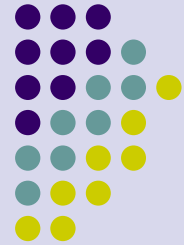
- **Sdílení souborů**
 - Napster, Gnutella, Kazaa, Freenet
 - Rozsáhlé sdílení souborů
 - Zpřístupnění souborů (hudba, video) na vlastním počítači ostatním
 - Ostatní uživatelé se připojí do sítě, vyhledávají soubory a stahují si je
 - Problém – zasahování do autorských práv

Napster



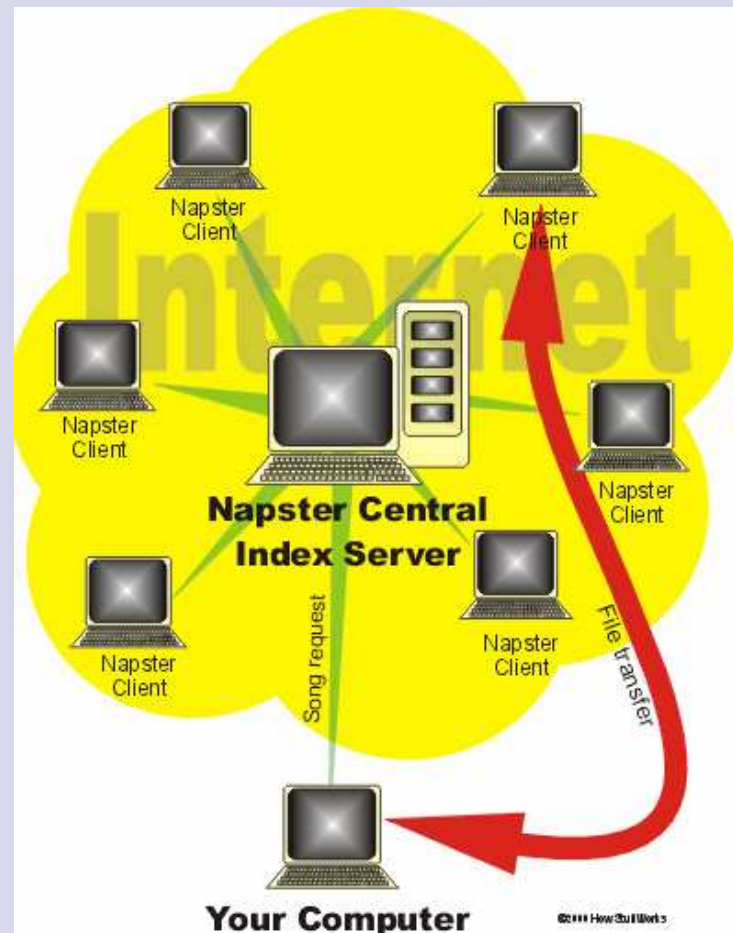
- 5/1999 – 2001 (soud), 2002 - konec
- sdílení hudebních souborů
- centralizované řešení
- uživatelé zapíší seznam svých souborů do Napster serveru
- další uživatelé pošlou požadavek se seznamem požadovaných souborů do Napster serveru (vyhledávání podle klíčů)
- Napster server pošle seznam IP adres počítačů, které soubory obsahují
- Uživatel se připojí přímo k vybranému uživateli a soubor stáhne

Napster



- Centrální Napster server
 - Zajišťuje správné odpovědi
 - Je však úzkým místem z hlediska škálovatelnosti
 - Je také úzkým místem z hlediska chyb
 - Citlivý na DoD
- Prohledávání je centralizované
- Přenos souborů je přímý (peer-to-peer)

Napster



Gnutella



- dovede sdílet libovolné soubory
- na rozdíl od Napster provádí decentralizované prohledávání
- dotazuje se sousedů
- sousedé se dotazují svých sousedů, atd.
- počet úrovní prohledávání je dán TTL
- uzly, které obsahují vyhledávané soubory odpoví

Gnutella



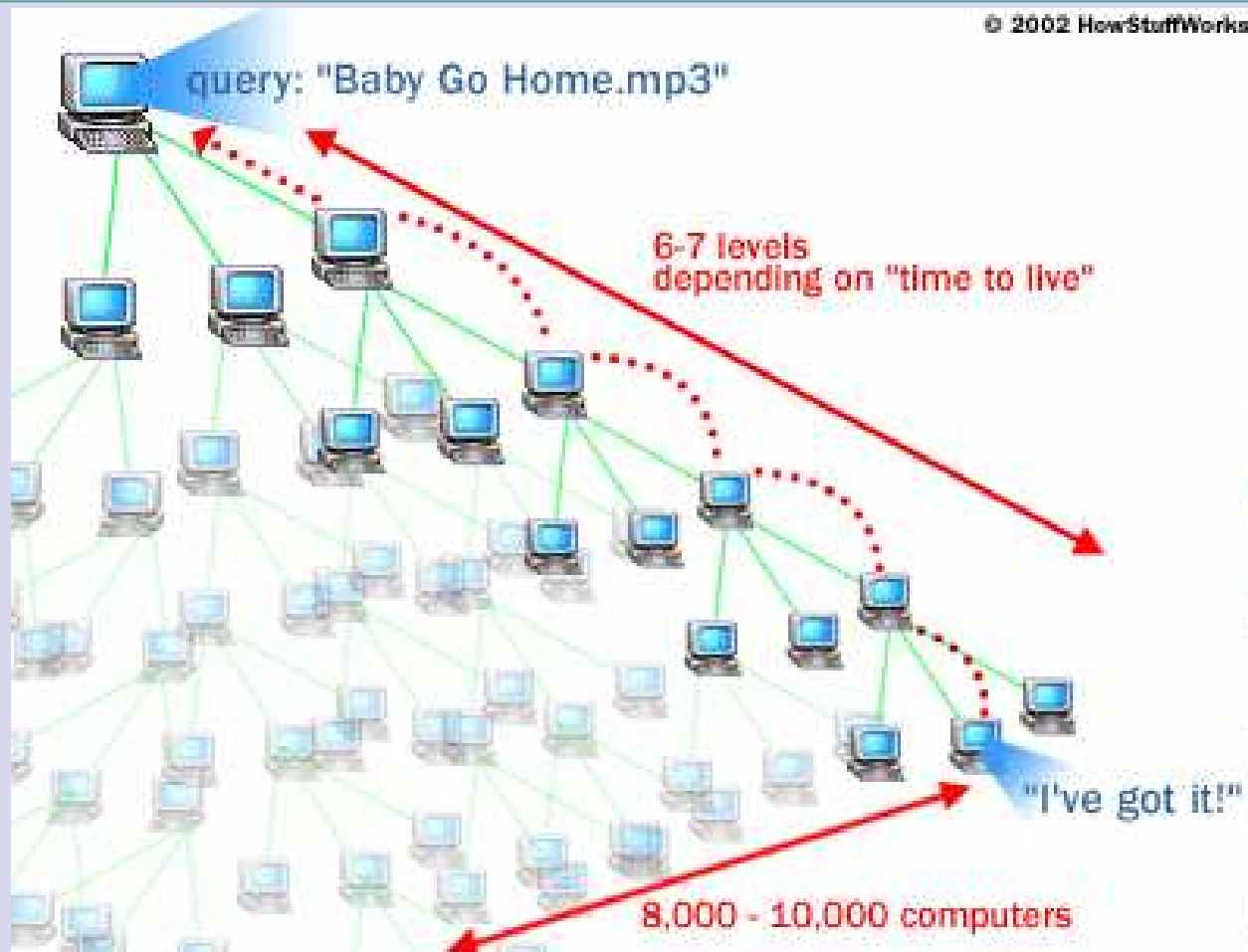
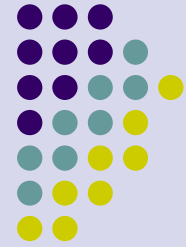
- decentralizované prohledávání
 - odstraněn problém místa citlivého na chyby
 - není tak citlivý na DoS
 - nemůže zaručit správné výsledky
- záplavové dotazování
 - prohledávání je distribuované, ale dosud neškálovatelné
- Stahování pomocí HTTP GET request

Gnutella



- Zprávy
 - Advertisement
 - Query
 - Push Request
- Formát zprávy
 - Message ID (16) - párování
 - Function ID (1) – typ
 - TTL (1)
 - Hops (1)
 - Payload Length (4)

Gnutella





Kazaa (sít' FastTrack)

- hybrid centralizované Napster a decentralizované Gnutella
- super-peer vystupují jako lokální centra vyhledávání
 - každý super-peer je obdobou Napster serveru pro malou část sítě
 - super-peer jsou vybírány systémem automaticky na základě jejich parametrů (paměť, šířka pásma, ...) a dostupnosti (čas pro připojení)



Kazaa (sít' FastTrack)

- uživatelé přenesou svůj seznam souborů do super-peer
- super-peer si periodicky vyměňují seznam souborů
- uživatelé posílají dotazy do super-peer
- systém slouží ke sdílení souborů
- existuje možnost nedsílet data, pouze je stahovat



Anonymita P2P sítí

- Napster, Gnutella ani Kazaa nezajišťují anonymitu
 - Uživatelé vědí kde co je a kdo co požaduje
- Freenet
 - Navržen mimo jiné k zajištění anonymity

Freenet



- 6/1999
- data jsou přenášena v opačném směru než dotaz
 - není možné zjistit, je-li uživatel iniciátorem nebo pouze data přenáší dál
 - není možné zjistit, jestli uživatel data posílá dál nebo je také spotřebovává
- chytré dotazy
 - požadavky jsou směrovány do správného uzlu postupně

Freenet

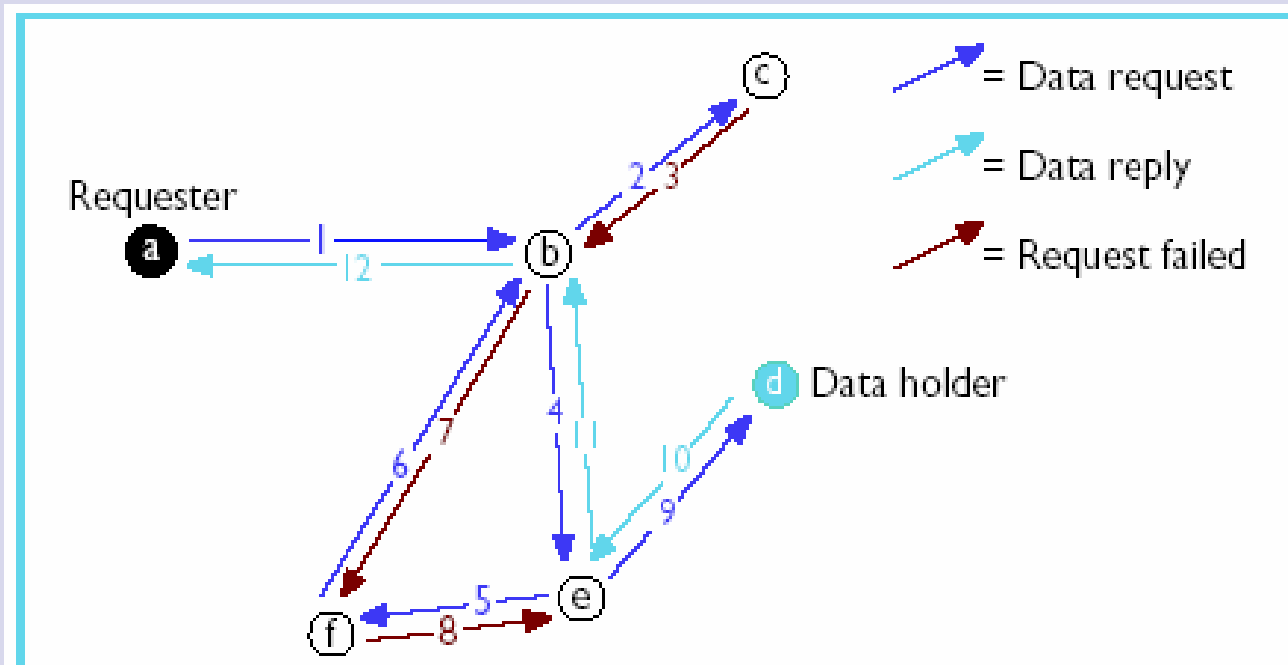
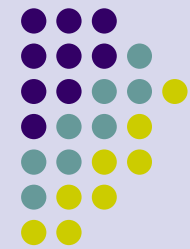


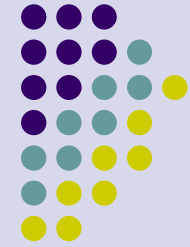
Figure 1. Typical request sequence. The request moves through the network from node to node, backing out of a dead-end (step 3) and a loop (step 7) before locating the desired file.



Strukturované P2P sítě

- druhá generace P2P sítí
- samo organizující se struktura
- vyrovnávání zátěže
- odolné proti chybám
- založeno na distribuované hashovací tabulce

Distribuované hashovací tabulky DHT



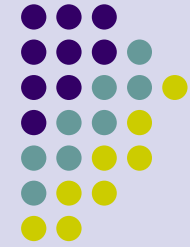
- Distribuované hashovací tabulky (Distributed Hash Tables – DHT)
 - distribuovaná verze hashovací tabulky (datová struktura)
 - ukládá páry (klíč, hodnota)
 - klíč je podobný jménu souboru
 - hodnota může být obsah souboru
 - cíl: efektivní vkládání, prohledávání, rušení párů (klíč, hodnota)
 - každý uzel ukládá do systému podmnožinu párů (klíč, hodnota)
 - základní operace: nalezení uzlu, který obsahuje klíč
 - mapování klíčů na uzly
 - efektivně směřovat požadavky vložení, prohledání, rušení do tohoto uzlu

Distribuované hashovací tabulky DHT



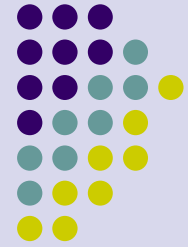
- Základní rozhraní DHT
 - identifikátor uzlu (Node id): m-bitový identifikátor (obdoba IP adresy)
 - klíč: posloupnost slabik
 - hodnota: posloupnost slabik
 - operace:
 - put(key, value) – uložení do uzlu, odpovídajícímu klíči
 - value=get(key) – obnovení hodnoty spojené s klíčem z odpovídajícího uzlu

Distribuované hashovací tabulky DHT



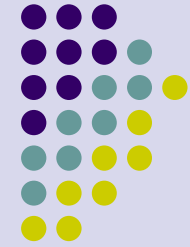
- Aplikace DHT
 - nad rozhraním DHT mohou být realizovány služby
 - sdílení souborů
 - archivační služby
 - databáze
 - jmenné a adresářové služby
 - chat
 - publikování/čtení

Distribuované hashovací tabulky DHT



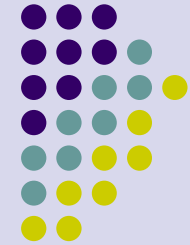
- Požadované vlastnosti DHT
 - rovnoměrné mapování klíčů do všech uzlů sítě
 - každý uzel udržuje informaci pouze o malém počtu uzlů
 - efektivní směrování zpráv do uzlů
 - připojení a odpojení uzlů ovlivní pouze malý počet uzlů

Distribuované hashovací tabulky DHT



- Směrovací protokoly DHT
 - DHT rozhraní bylo realizováno v několika implementacích
 - Chord (MIT)
 - Pastry (Microsoft Research UK, Rice University)
 - Tapestry (UC Berkeley)
 - CAN – Content Addressable Network (UC Berkeley)
 - SkipNet (Microsoft research US, Univ. Of Washington)
 - Kademlia (New York University)
 - Viceroy (UC Berkeley)
 - P-Grid (EPFL Switzerland)
 - Freenet (Ian Clarke)
 - Tyto systémy jsou často označovány jako P2P overlay networks

Chord



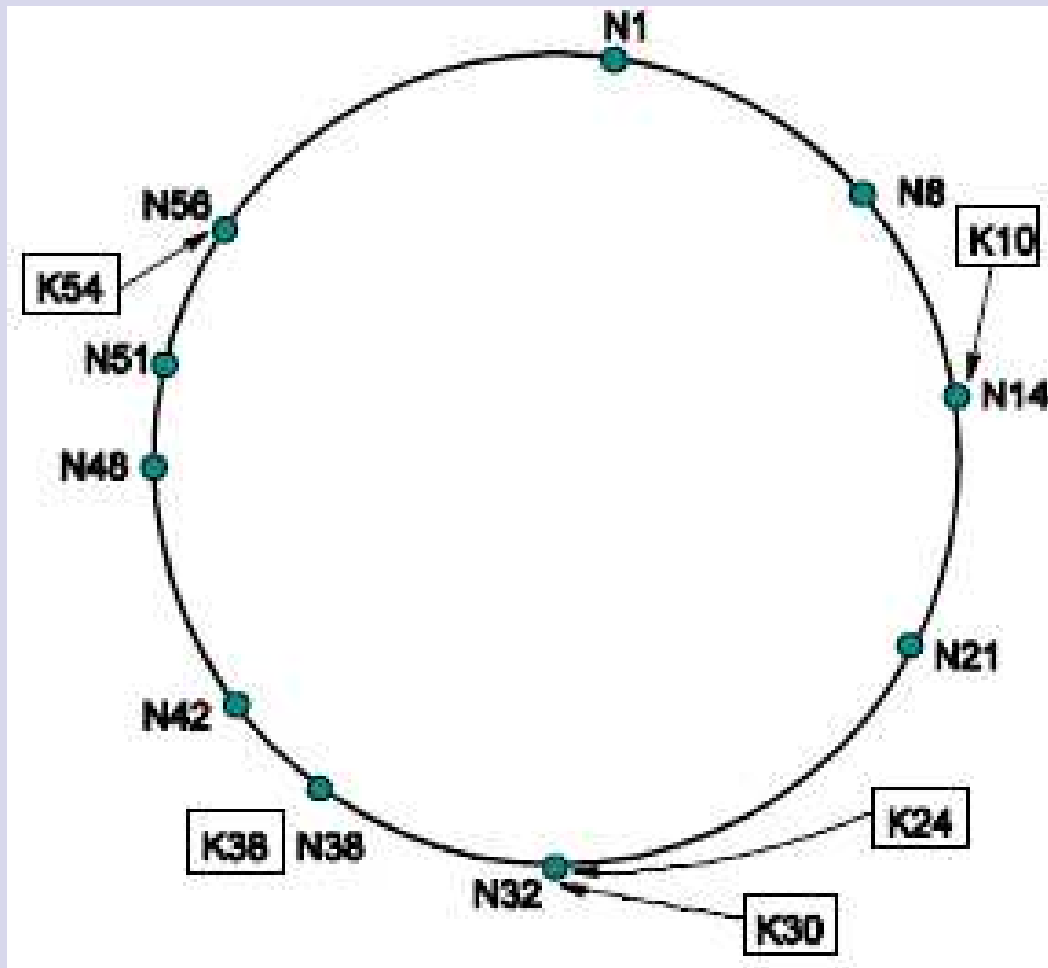
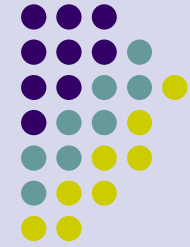
- Node id: unikátní m-bitový identifikátor (hash IP adresy, jiné unikátní ID)
- Klíč: m-bitový (160) identifikátor (hash sekvence slabik – SHA-1)
- Klíč – SHA-1 – ID
- IP adresa – SHA-1 - ID
- Hodnota: sekvence slabik
- API
 - Insert(key, value) – uložení klíče a hodnoty do r uzlů
 - Lookup(key)
 - Update(key, newval)
 - Join(n)
 - Leave()

Chord



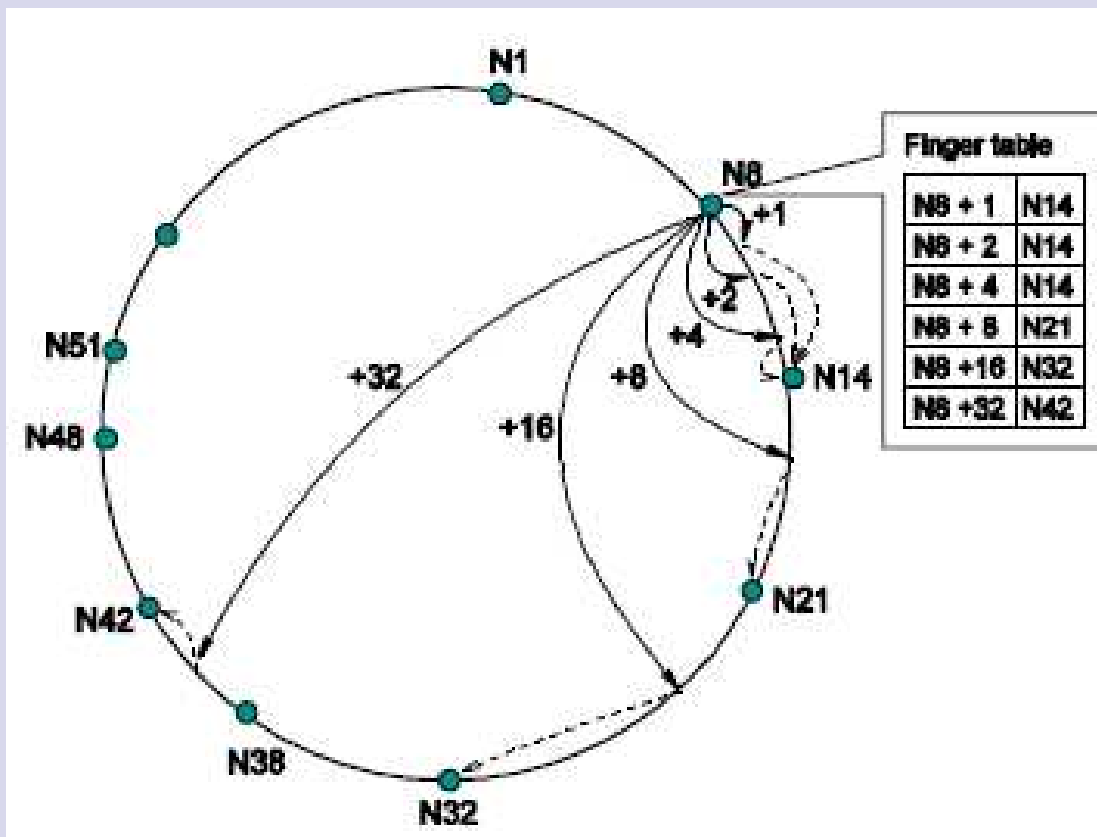
- Uzly jsou organizovány do kruhu identifikátorů
- Klíče jsou přiřazeny číselně následujícím uzlům
- Hashovací funkce zajišťuje distribuci uzlů i klíčů v kruhu

Chord



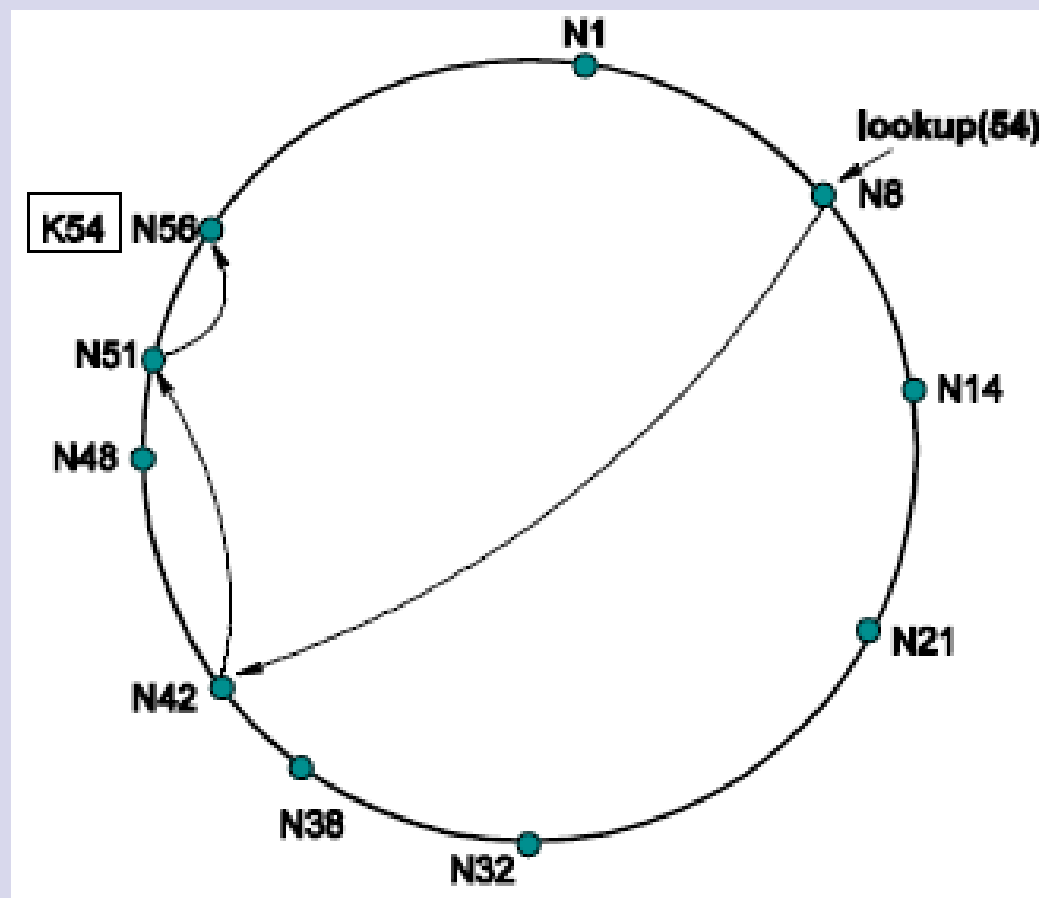
- Vyhledávací tabulka
 - velikost tabulky $O(\log N)$

Chord

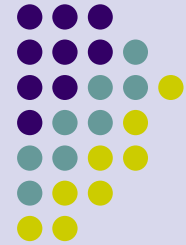


- Vyhledávání
 - hledání uzlu s číslem nejblížeším vyšším než je hodnota klíče

Chord



Chord



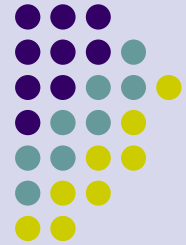
- Vlastnosti
 - v systému s N uzly a K klíči s vysokou pravděpodobností
 - každý uzel obsahuje nanejvýš K/N klíčů
 - každý uzel udržuje informaci o přibližně $O(\log N)$ jiných uzlech
 - počet prohlížených uzlů závisí na $O(\log N)$
 - není zaručeno dodání výsledků
 - není zaručena konzistentnost replik
 - uzly v kruhu mohou být umístěny v síti kdekoliv (umístění bez vazby na síť)

Pastry



- rozhraní podobné jako Chord
- předpokládá lokalizaci sítě pro minimalizaci přeskoků
- nový uzel potřebuje znát nejbližší uzel, aby bylo dosaženo lokality

CAN



- CAN (Content-Addressable Network)
 - založeno na d-dimenzionálním kartézském systému
 - každý uzel vlastní odlišnou zónu v prostoru
 - každý klíč je hashován na bod prostoru

CAN

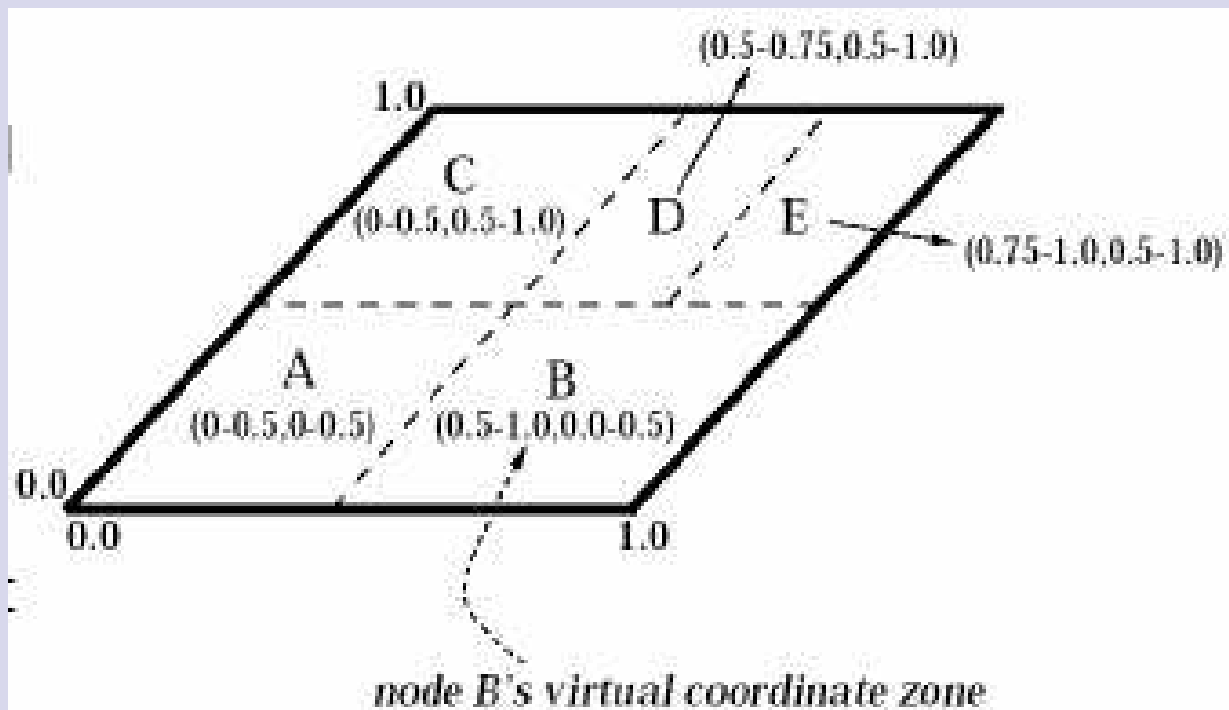
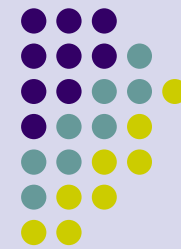
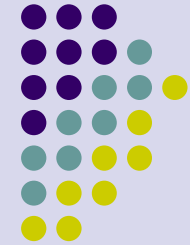


Figure 1: *Example 2-d space with 5 nodes*

Bit Torrent



- Bram Cohen
- TCP pro přenosy
- Dělení souboru na kousky (16kB)
 - Každý kousek zabezpečen SHA-1
- Stahování přes Web server
 - Supernova.org (+zrcadla)
 - Obsahuje metadata soubor (.torrent)
 - SHA-1 pro všechny kousky souboru
 - Mapování kousků do souboru
 - Odkaz na tracker (umístění částí souborů)



Bit Torrent

- Seed (zdroj) – vytváří .torrent soubor
- Tracker – centrální server udržující seznam členů ve swarmu
- Swarm – soubor uzlů zúčastněných na distribuci souborů
- Člen se spojí se SWARMem aby získal TRACKER a z něho seznam členů, ke kterým se pak připojí

Bit Torrent



- Překrývání částí souborů
 - Rovnoměrné překrytí
 - Dobré využití šířky pásma
 - Záloha souborů, distribuované kopie
 - Náhodný výběr člena pro stahování
 - Odolný vůči výpadkům členů