


Sieci P2P

Tribler 

Janina Mincer-Daszkiewicz

Systemy rozproszone

MSUI, II rok

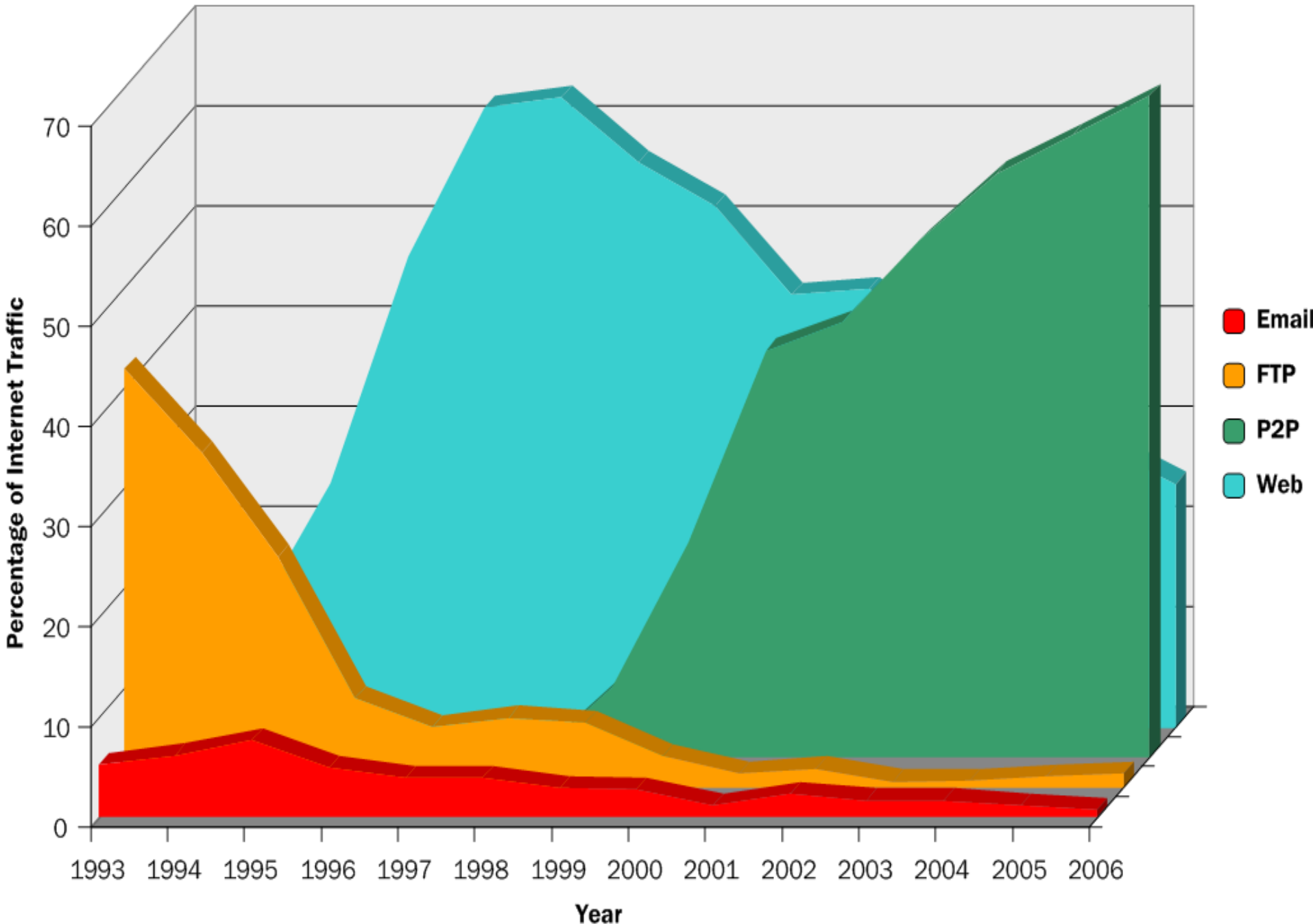
Materialy i rysunki zaczerpnięto z następujących źródeł

1. Definicje terminów, Wikipedia (wersja angielska)
2. Definicja sieci P2P, <http://portalwiedzy.onet.pl>
3. J.A. Pouwelse, P. Garbacki i inni, *Tribler: a social-based peer-to-peer system*, Concurrency and Computation: Practice and experience, 2007
4. Projekt Tribler, <http://www.tribler.org/>
5. Michel Meulpolder, *TriblerCampus: An Integrated Peer-to-peer Platform for File Distribution in Course Management Systems*, MSc Thesis, Delft, University of Technology, 2006

Definicja

- **P2P**, sieć **równorzędna**, sieć typu **każdy z każdym**. Jest to model architektury sieciowej oparty na równoważności wszystkich jej węzłów. W sieci P2P każdy komputer dysponuje podobnymi możliwościami oraz może inicjować połączenia. Nie ma ustalonej hierarchii ani centralnego serwera. Ten sam komputer może równocześnie pełnić rolę serwera i klienta, czyli pobierać dane z innych komputerów i udostępniać swoje zasoby wszystkim pozostałym komputerom.

CacheLogic Research | Internet Protocol Trends 1993 to 2006



Historia sieci P2P

- **Napster** (1999) – sieć P2P składająca się z centralnego serwera oraz tysięcy czy nawet milionów klientów (W 2001 r., po czasowym wprowadzeniu filtrów w wyszukiwarce utworów, w wyniku przegranych procesów sądowych serwis zakończył działalność, powracając jako legalny, płatny serwis pod koniec 2003 r.)
- **Gnutella** – prezentowała odmienne podejście do problemu wymiany plików: wszystkie komputery były równorzędne (nieefektywne wyszukiwanie plików, krążące po sieci zapytania zapychały łącza, stanowiąc około połowy ruchu tworzonego przez Gnutellę)

Historia sieci P2P - cd

- **Kazaa** – wykorzystuje protokół FastTrack Jeśli któraś maszyna spełnia określone warunki (m.in. dysponuje szybkim łączem, mocnym procesorem i pojemną pamięcią), zyskuje status superwęzła. Taki wyróżniony komputer zaczyna działać trochę jak serwer z Napstera. Ważną cechą FastTracka jest jego samoorganizacja. O funkcji komputera nie decyduje człowiek, a program.
- **eDonkey** – powrót do idei centralnego serwera. Podstawą są serwery nazywane hubami. Do nich podłączają się komputery klienckie. Podobnie jak w Napsterze, wyróżniona maszyna pośredniczy w szukaniu plików. W przeciwieństwie do pierwszej sieci P2P hubów jest wiele i są prowadzone przez różne osoby i organizacje (2006 – ugoda z RIAA, zaprzestano dystrybucji klienta)

Historia sieci P2P - cd

- **Kademlia** (ostatnio przechrzczona na Kad) to sieć całkowicie rozproszona. Korzysta z DHT. Z racji braku serwerów łączenie z siecią realizuje się poprzez wskazanie jakiegokolwiek już podłączonego węzła. Sieć sama się organizuje i dostraja, by osiągnąć lepszą wydajność zależnie od liczby użytkowników i możliwości ich połączeń.
- **BitTorrent** – korzysta z protokołu, którego celem jest odciążenie łączy serwera udostępniającego pliki. Jego największą zaletą w porównaniu do protokołu HTTP jest podział pasma pomiędzy osoby, które w tym samym czasie pobierają dany plik. Oznacza to, że użytkownik w czasie pobierania wysyła fragmenty pliku innym użytkownikom.

Ważne pojęcia

- **Rozproszona tablica mieszająca** (ang. *Distributed Hash Table*, w skrócie: **DHT**) – rozproszony system pamięci używany do przechowywania dużych ilości danych. Na system DHT składa się pewna liczba stacji roboczych, które dzielą między sobą zbiór kluczy.
- **Skrót** (ang. *hash*). Losowo wyglądający ciąg znaków, który powstał z użyciem odpowiedniej funkcji matematycznej na podstawie zawartości zbiorów. Skrót jednoznacznie identyfikuje plik – jeśli wyszukamy zbiory o tych samych skrótach, możemy być pewni, że są to te same dane.
- **Ziarno** (ang. *seed*). Użytkownik posiadający kompletny plik i udostępniający go innym osobom.

Ważne pojęcia – cd

- **Węzeł śledzący** (ang. *tracker*). Serwer przekazujący informacje (adresy IP) o innych użytkownikach pobierających plik.
- **.torrent** Metaplik zawierający informacje niezbędne do rozpoczęcia pobierania pliku (np. adres węzła śledzącego)
- **Reputacja** (ang. *reputation*). Ocena przeszłego zachowania węzła w sieci, mierzona jego współczynnikiem współdzielenia
- **Współczynnik współdzielenia** (ang. *sharing-ratio*). Stosunek ilości danych wysłanych do pobranych.
- **Pijawka** (ang. *leacher*). Użytkownik, który pobiera dużo więcej danych niż sam udostępnia, korzystając np. ze zmodyfikowanej wersji oprogramowania klienckiego.

Ważne pojęcia – cd

- **Filtr Blooma** (ang. *Bloom filter*). Efektywna pamięciowo probabilistyczna struktura danych używana do sprawdzania przynależności elementu do zbioru. Możliwe są przypadki *false positive*, niemożliwe *false negative*.
Przykład: filtr Blooma zawiera słownik poprawnych słów, będzie akceptował wszystkie słowa, które należą do słownika i odrzucał prawie wszystkie, które nie są (tzn. niektórych nie odrzuci). Zależnie od współczynnika błędu wynikowy zbiór może wymagać jedynie bajtu na słowo
Używane np. w Google do implementacji **BigTable** (szybka, skalowalna DBMS) do redukcji przeszukiwania dysku w poszukiwaniu nieistniejących wierszy i kolumn

Ważne pojęcia – cd

- **Tit for tat** (odpowiednik polskiego *wet za wet*). Możliwy schemat wymiany plików, bazujący na uczciwej wymianie: dane są wysyłane do tych węzłów, które także przysyłają dane. Stosowany bardzo ściśle może nie być optymalny, np. nowe węzły nie mają co wysyłać

Tit for tat to wysoce efektywna **strategia w teorii gier** dla dylematu więźnia, który wielokrotnie podejmuje decyzje

Dylemat więźnia: sypnąć kolegę, czy trzymać z nim sztamę?

Jeśli Ty sypniesz, a on nie, to on dostanie 10 lat, a Ty nic

Jeśli obaj będziecie sypać, to obaj dostaniecie 5 lat

Jeśli obaj będziecie milczeć, to obaj dostaniecie 6 miesięcy

Ważne pojęcia – cd

- **Optimistic unchoking** (ang. *choke*, dławić się). Mechanizm wymiany plików stosowany przez klienta BitTorrent. Klient rezerwuje część dostępnej przepustowości na wysyłanie kawałków do losowych węzłów (a nie tylko dobrze znanych partnerów, tzw. preferowanych węzłów) w nadziei znalezienia jeszcze lepszych partnerów i dania szansy nowym węzłom na dołączenie do roja (ang. *swarm*)

Wyzwania w sieciach P2P

⇒ Liczba użytkowników sieci P2P jest rzędu milionów ⇐

Badania wydajności sieci BitTorrent pozwoliły zidentyfikować następujące wyzwania:

- **decentralizacja funkcjonalności sieci** pomiędzy węzłami; centralne serwery muszą być przez kogoś utrzymywane, mogą stanowić wąskie gardło, punkty awarii, zagrożenie dla bezpieczeństwa, z drugiej strony zaś ułatwiają dostęp do informacji.

Współcześnie nie istnieje w pełni zdecentralizowana efektywna sieć P2P. BitTorrent nie jest w pełni zdecentralizowany, do szukania treści wykorzystuje strony sieci Web i węzły śledzące

Wyzwania w sieciach P2P - cd

- **dostępność sieci** nie powinna zależeć od dostępności żadnego konkretnego węzła ani elementu centralnego (mniej niż 4% węzłów ma czas aktywności powyżej 10 godzin)
- zachowanie **integralności systemu i zaufania** między węzłami. Dane mogą być atakowane na kilku poziomach: wskaźniki do treści, metadane, sama treść (jak usuwać zanieczyszczone, ang. *polluted*, dane?)
- zapewnienie **właściwej zachęty** (ang. *incentive*) do udostępniania własnych zasobów jest kluczowe dla zapewnienia współpracy i osiągnięcia dobrej wydajności

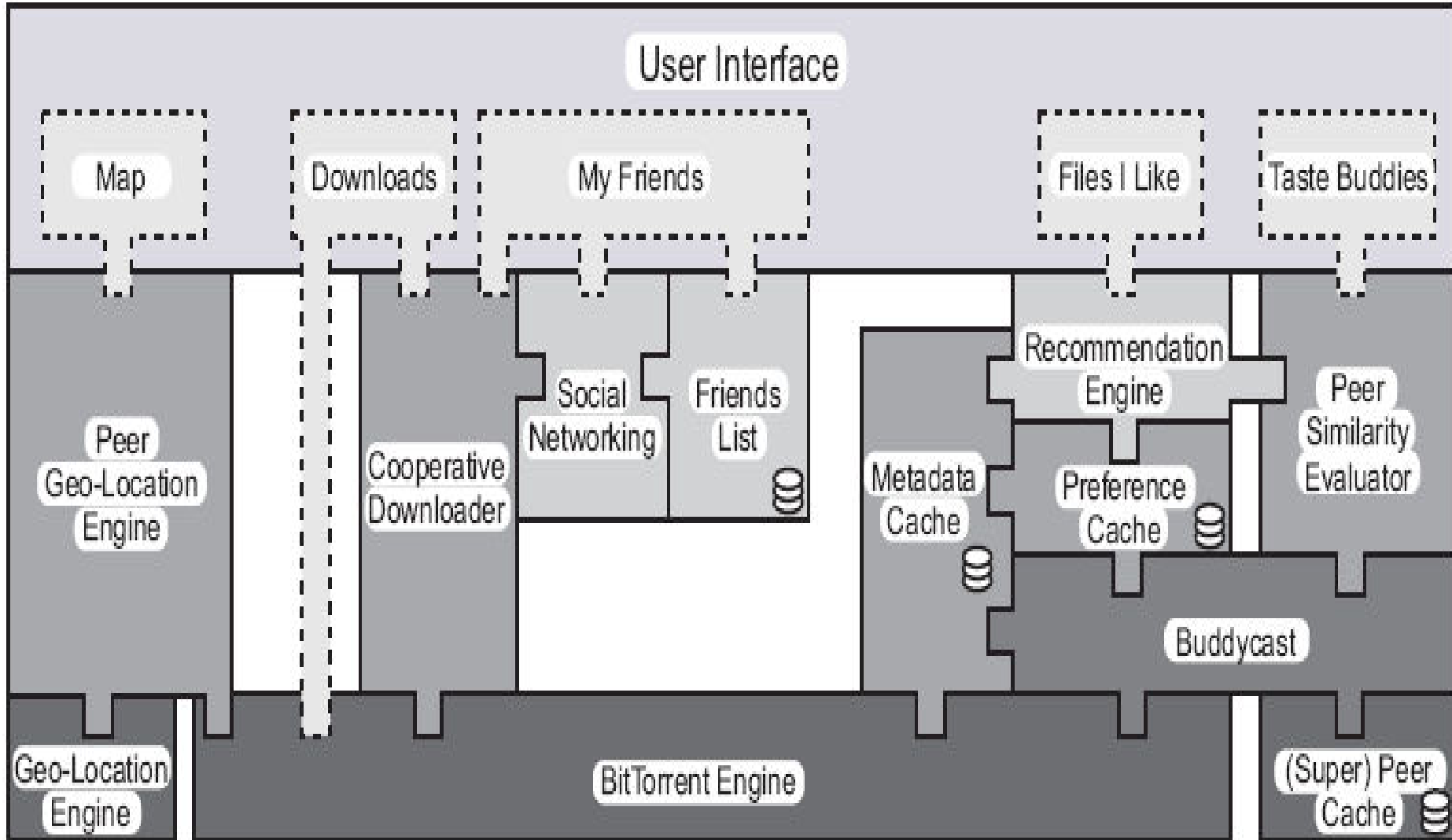
Wyzwania w sieciach P2P - cd

- osiągnięcie **przezroczystość sieci** (ang. *network transparency*) poprzez rozwiązanie problemu powodowanego przez dynamiczne numery IP, pudełka z adresami NAT (ang. *NAT boxes*), ściany ogniowe. W Internecie nastąpiły duże zmiany w rezultacie czego węzły w sieci nie mogą już swobodnie przesyłać treści bez pomocy innych węzłów pełniących rolę mediatorów
⇒ Zdaniem projektantów sieci Tribler ich protokół odpowiada na te wyzwania dzięki podtrzymywaniu grup społecznościowych ←

Tribler z lotu ptaka ...

- Nazwa **Tribler** pochodzi od słowa *Tribe* (plemię), w celu podkreślenia związku z **sieciami społecznościowymi** (ang. *social networks*)
- W sieci Tribler zgromadzone dane o istnieniu grup znajomych i osób o podobnych gustach wykorzystuje się przy **poszukiwaniu treści** (ang. *content discovery*), **rekomendowaniu treści** (ang. *content recommendation*) i **pobieraniu plików** (ang. *dowloading*)
- Zbudowana na bazie protokołu BitTorrent
- Głównie ma służyć do transmisji telewizji internetowej (YouTube zagrożony?)
- Projekt badawczy, ale także działająca sieć (8 tys. uz.) ☺

Architektura Triblera



Architektura – grupy społecznościowe

- Moduł Social Networking jest odpowiedzialny za przechowywanie i dostarczanie informacji o grupach społecznościowych (ich członkach, używanych przez nich ostatnio numerach IP itp.)
- Podobny gust dotyczący treści jest podstawą do tworzenia społeczności on-line z altruistycznym zachowaniem względem siebie
- Tribler ułatwia rozpoznanie przyjaciela, wroga (ang. *polluter*), nowego użytkownika sieci
- Rezygnuje się zatem z anonimowości i ułatwia formowanie grup społecznościowych
- Każdy użytkownik wybiera pseudonim (ang. *nickname*), który jest automatycznie przekazywany między węzłami

Architektura - megaschowki

- Praktycznie żadne sieci P2P nie utrzymują trwale informacji o poprzedniej aktywności w sieci, zwykle wymieniane są pytania o pliki i metadane plików, a ignorowane inne informacje
- Tribler utrzymuje w węzłach **megaschowki**, w których przechowuje dotyczące danego węzła informacje o relacjach społecznych, czasach aktywności węzłów, podobieństwie gustów
- Informacje te są wymieniane w ramach grup społecznościowych przy użyciu **algorytmu epidemicznego** (ang. *epidemic algorithm*) **BuddyCast**

Architektura – megaschowki cd

- Cztery megaschowki na rysunku:
 - *Friends List* – informacja o sieciach społecznościowych
 - *(Super) Peer Cache* – informacja o węzłach i super-węzłach
 - *Metadata Cache* – metadane plików
 - *Preference Cache* – listy preferencji innych węzłów
- Rozmiary schowków – poniżej 10 MB
- Narzut na uaktualnianie przechowywanych informacji - np. dla metadanych rzędu 600 KB/dzień (znośnie)
- Dzięki schowkom poszukiwanie treści może się odbywać lokalnie zamiast przez sieć

Architektura – szukanie treści

- Szukanie treści jest kluczowe dla sieci P2P
- Obecne rozwiązania bazują na: *query flooding*, *distributed hash tables*, *semantic clustering* (tego nie będziemy omawiać)
- Tribler łączy **ludzi o podobnych gustach** (**kumpli**, ang. *taste buddies*) oraz dąży do pełnej replikacji metadanych
- Moduł **Files I like** służy do przechowywania preferencji plików (w skali 1 .. 5). Domyślnie lista zawiera ostatnio ściągnięte pliki
- **Peer Similarity Evaluator** porównuje listy preferencji i określa podobieństwo gustów dwóch węzłów
- **Recommendation Engine** tworzy listę plików przypuszczalnie najbardziej pożądanym przez użytkownika

Architektura – pobieranie i inne

- Moduł BitTorrent Engine pobiera pliki przy użyciu protokołu zgodnego z BitTorrent. Może korzystać z udogodnień modułu Collaborative Downloader w celu przyśpieszenia transmisji badając dostępną przepustowość ładowania plików swoich aktywnych przyjaciół
- Głównym celem **interfejsu** jest ułatwienie tworzenia grup społecznościowych oraz wizualna identyfikacja potencjalnych współpracowników. Podczas pobierania pliku użytkownik widzi uczestniczące w transmisji węzły na mapie (Peer Geo-Location Engine)

Architektura – bootstrapping

- **Bootstrapping** – znajdowanie innych węzłów po instalacji oprogramowania
- Węzeł automatycznie, kontaktuje się z jednym z superwęzłów, ale tylko raz w celu otrzymania inicjalnej listy innych węzłów w systemie (komunikat Buddycast)
- Specjalny **rój sieci wirtualnej** (ang. *overlay swarm*), do którego należy każdy węzeł, który nie ma węzła śledzącego i który jest używany w celu wyszukiwania treści i węzłów, przy użyciu algorytmu Buddycast

Sieci społecznościowe

- Poważnym problemem w systemach współdzielenia plików jest **ograniczenie sesji** (ang. *session boundary*) – cała informacja kontekstowa jest gubiona gdy użytkownik odłącza się od sieci. Ze względu na dynamiczne IP trudno jest przechowywać taką informację pomiędzy sesjami
- Wprowadzono zatem trwałe, unikatowe i bezpieczne **identyfikatory węzłów** (PermIDs), które są kluczami publicznymi wymienianymi mailowo
- Tworzenie sieci będzie szybsze, gdy powstanie możliwość importowania kontaktów z innych sieci, jak MSN, GMail

Sieci społecznościowe – cd

- Do rozpraszania i porównywania parami zawartości megaschowków stosuje się **filtry Blooma** – bardzo gęste struktury danych podobne do DHT do przechowywania i testowania przynależności do zbioru. Znacząco redukują one zapotrzebowanie algorytmów epidemicznych na szerokość pasma
- Rozmiar filtru Blooma zależy od liczby oczekiwanych połączeń węzłów
- Pomiar: osoba ma średnio 243 przyjaciół i 9147 przyjaciół swoich przyjaciół
- Z tego wynika, że wystarczy 260 bajtów żeby odszukać wspólnych przyjaciół swoich przyjaciół dwóch węzłów przy użyciu filtru Blooma

Algorytm Buddycast

- Każdy węzeł przechowuje w swoim megaschowku pewną liczbę osób o podobnych gustach (kumpli) z ich listami preferencji i pewną liczbę losowych węzłów (o nieznanym preferencjach)
- Cyklicznie (np. co 15 s) węzeł łączy się z:
 - jednym ze swoich kumpli (ang. *exploitation*) albo
 - losowym węzłem (ang. *exploration*)żeby wymienić komunikat lub żeby odpowiedzieć na otrzymany komunikat

Algorytm Buddycast - cd

- Komunikat zawiera identyfikatory 10 kumpli i 10 pozycji z góry listy preferencji, identyfikatory 10 świeżych węzłów oraz 50 pozycji z listy preferencji węzła wysyłającego
- Po wymianie komunikatów oba węzły wstawiają otrzymaną informację do swojego megaschowka
- Każdy węzeł pamięta listę losowych węzłów, z którymi się ostatnio kontaktował i stara się pomijać je przez jakiś czas (4 godz.)
- Użytkownicy mogą wyłączyć Buddycast, jeśli chcą chronić swoją prywatność
- Losowy węzeł jest wybierany na podstawie jego ‘świeżości’, żeby zwiększyć szansę, że podczas połączenia będzie dostępny
- Kumpel jest wybierany spośród 100 ostatnich, z którymi węzeł się kontaktował, na bazie podobieństwa gustów

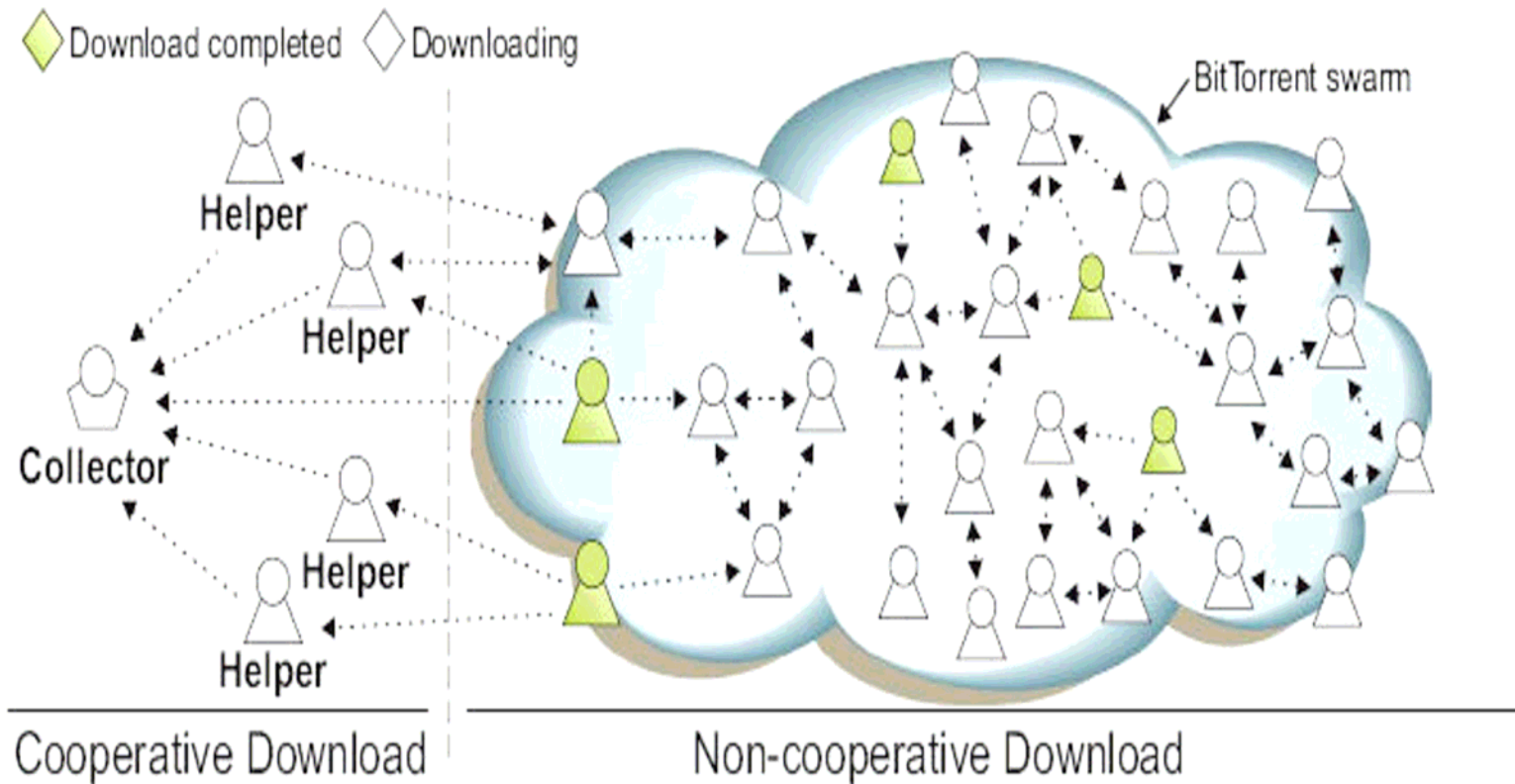
Współpraca przy pobieraniu

- Protokół 2Fast do pobierania plików na zasadzie współpracy (ang. *collaborative downloading*)
- Wczesne protokoły (np. Gnutelli) nie zapewniały **właściwej zachęty** (ang. *incentive*) do oferowania przepustowości pasma, a to sprzyja **jeźdzeniu za darmo** (ang. *freeriding*)
- Mechanizm Tit-for-Tat z BitTorrent jako pierwszy wykorzystywał taką zachętę
- Protokół 2Fast wykorzystuje grupy społecznościowe, w których członkowie, którzy sobie wzajemnie ufają, współpracują w celu zwiększenia wydajności pobierania
- Węzły z grupy, które decydują się współpracować występują w roli **pobieracza** (ang. *collector*) lub **pomocnika** (ang. *helper*)
- Pobieracz chce pobrać cały plik, a pomocnik obiecał mu pomóc

Współpraca przy pobieraniu - cd

- Pobieracz i pomocnicy zaczynają pobierać plik
- Pomocnik wybiera kawałek pliku do ściągnięcia, ale przed ściągnięciem prosi pobieracza o akceptację, którą otrzyma tylko jeśli kawałka nie ma jeszcze żaden pomocnik
- Po ściągnięciu pomocnik wysyła kawałek do pobieracza nie prosząc o nic w zamian
- Całkowity czas pobierania skrócił się 2, 3, 6-krotnie, zależnie od sposobu połączenia z Internetem

Współpraca przy pobieraniu – cd



Prace magisterskie – TriblerCampus

Course Management Systems (CMSs) are widely used in educational institutes around the world to manage course information and to distribute educational content. The existing framework of CMSs could make them an ideal platform for offering digital lectures and multimedia content as well. However, all major implementations of CMSs (e.g., Blackboard, Moodle) are based on a client-server architecture. Because of this, the performance of these systems is limited, especially regarding the distribution of large files to large numbers of people. TriblerCampus aims to overcome this problem, by integrating Tribler peer-to-peer technology with the popular CMS Moodle. The user interacts in the usual way with the CMS, while in the background large files are distributed in peer-to-peer fashion, based on the BitTorrent protocol. In order to further improve the performance and scalability of the resulting system, caching and replication mechanisms are introduced: regular peers cache recently downloaded content, and so-called partners automatically replicate copies of popular files. The resulting system makes it possible to download large files with a high and robust performance, even when the demand in the system is very high. In this thesis, the design and implementation of TriblerCampus are presented, as well as an extensive performance analysis based on mathematical analysis, simulation and experiments.

Prace magisterskie – Grid monitoring

Michał Ejdys, *A peer-to-peer Grid monitoring system*

<http://students.mimuw.edu.pl/SR/prace-mgr//ejdys/ejdys.pdf>

Grid monitoring systems aim at delivering accurate dynamic information about the Grid, which helps users with optimal utilization of the resources. Current systems have a static structure or are centralized services thus are sensitive to nodes failures, not scalable, and difficult in administrating. In this work we propose an architecture of a peer-to-peer Grid monitoring system, addressing these issues. Furthermore, we propose three information disseminating strategies and evaluate their prototype implementations.

Prace magisterskie – Anonimowość

Łukasz Jancewicz, *Anonimowa publikacja treści w Internecie*

<http://students.mimuw.edu.pl/SR/prace-mgr/jancewicz/jancewicz2007-05-24.pdf>

Publikacja treści w Internecie zwykle wiąże się z ryzykiem ujawnienia tożsamości publikującego, co może doprowadzić do przykrych dla niego konsekwencji. Istnieją aplikacje zapewniające całkowitą anonimowość w sieci, ale ich działanie jest bardzo powolne, a używanie niewygodne. Z kolei aplikacje służące do wymiany plików są nieprzystosowane do zapewniania anonimowości. W pracy przedstawiono projekt i realizację sieci Sandstorm, która zapewnia częściową anonimowość dla nadawców, nie dla odbiorców oraz aplikacji służącej do wymiany plików, korzystającej z tej sieci. ...

Podsumowanie

- Sieci 2P2 to nie tylko budzące emocje procesy sądowe, lecz przede wszystkim bardzo aktywna dziedzina badawcza z wieloma otwartymi problemami
- Znajdują coraz więcej zastosowań, także w innych dziedzinach informatyki
- Mogą być przedmiotem pracy magisterskiej