Annemari Auvinen

CHEDAR: VIKASIETOINEN VERTAISVERKKOALUSTA JA SEN KÄYTTÄMÄT ALGORITMIT

Tietotekniikan pro gradu -seminaari

24.4.2003

Jyväskylän yliopisto

Tietotekniikan laitos

Sisältö

1 JOHDANTO 1

2 VERTAISVERKOT 2

2.1 MÄÄRITELMÄ 2

3 CHEDAR 5

3.1 ALGORITMIT 6

3.1.1 Overtaking 6

3.1.2 Node Selection 6

3.1.3 Node Removal 7

3.1.4 Overload Estimation 7

3.1.5 Resource Reply -viestin reititys 7

LÄHTEET 8

1 Johdanto

Pro graduni tarkoituksena on tutkia, minkälaisen verkon topologian Chedar-vertaisverkkoalustan algoritmit muodostavat. Tutkimustyökaluna käytettävä Chedar-vertaisverkkoalusta on kehitetty hajautettujen resurssien löytämiseksi vertaisverkosta. Chedarin käyttämät algoritmit on suunniteltu mukautumiskykyisiksi vaihteleviin tilanteisiin, ja niiden käyttäytymistä arvioidaan myös mittaamalla niiden suorituskykyä useista työasemista koostuvassa tietoliikenneverkossa.

2 Vertaisverkot

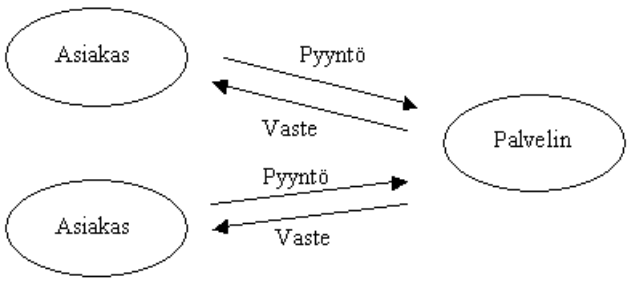
Vertaisverkko (Peer-to-Peer, P2P) on vanhin arkkitehtuuri kommunikoinnissa: puhelimet ovat vertaisverkkoja, samoin IP-reititys. Alkujaan 1960-luvun lopulla Internet suunniteltiin P2P-järjestelmäksi. ARPANET:in tarkoituksena oli jakaa laskentaresursseja Yhdysvalloissa verkkoarkkitehtuurissa, jossa jokainen isäntä toimii tasavertaisena solmuna.

1994 Internetin käyttäjämäärä kasvoi räjähdysmäisesti, mikä muutti Internetin rakennetta. Miljoonat uudet käyttäjät olivat kiinnostuneita sähköpostin lähettämisestä ja WWW-sivujen katselemisesta verkon yksityiskohtien sijaan. Selaimet kuten monet muutkin Internetin kaupallistumisen aikaan syntyneet sovellukset perustuivat asiakas-palvelin –protokollaan. Internetistä muokkautui verkko, jossa on suhteellisen pieni määrä erikoistuneita palvelimia ja paljon näitä käyttäviä asiakkaita.

Nykyiset vertaisverkkoteknologiat palauttavat Internetin sen alkuperäiseen tavoitteeseen, missä jokainen tarjoaa sekä kuluttaa.

2.1 Määritelmä

Yleisimmin verkoissa käytössä oleva arkkitehtuuri on asiakas-palvelin mallinen. Siinä asiakas ottaa yhteyden tunnettuun palvelimeen, lataa dataa ja sulkee yhteyden. Suosittu data on heikommin saatavilla, kun pyynnöt kuormittavat keskitettyä palvelinta. Arkkitehtuuri on myös altis sensuurille ja teknisille vioille.

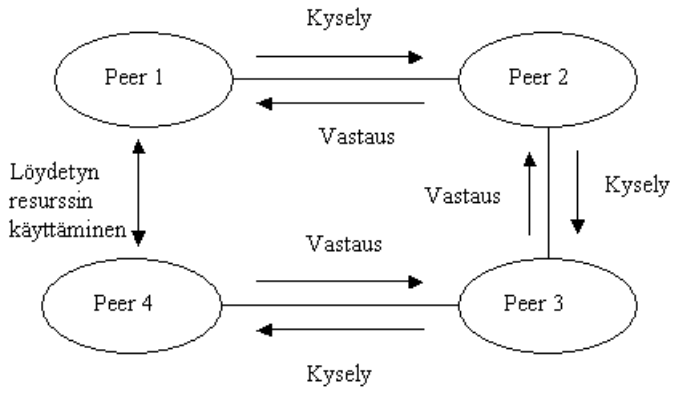


Kuva 1: Kommunikointi asiakas-palvelin -arkkitehtuurissa

Puhdas vertaisverkko muodostuu hajautetun verkon koneista eli vertaisista (peers), jotka keskustelevat keskenään ilman keskitettyä pistettä. Verkossa ei ole erikseen palvelimia ja asiakkaita vaan käytännössä jokainen kone toimii sekä palvelimena että asiakkaana. Kone tarjoaa muille verkon koneille palveluita kuten laskenta-aikaa, tiedostoja tai levytilaa ja itse vastaavasti käyttää muiden tarjoamia palveluita.

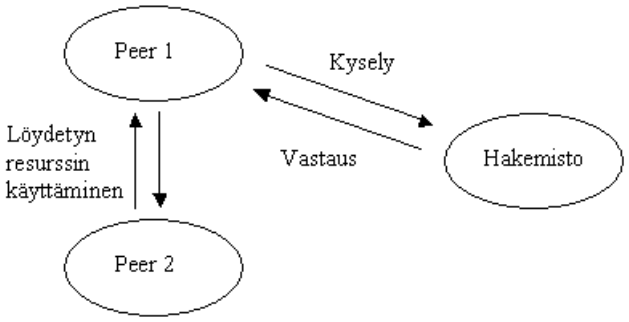
Vertaisverkon ja asiakas-palvelin –mallin ero voidaan määritellä myös sen avulla, kuka omistaa ja hallinnoi laitteistoa, joka palveluita tarjoaa. Asiakas-palvelin –arkkitehtuurissa se on yleensä jokin organisaatio, kun taas vertaisverkkosovelluksissa kukin verkon käyttäjä. Voidaan siis sanoa, että vertaisverkko on hajautettu myös kustannusten ja hallinnan kannalta. Asiakas-palvelin –malliin verrattuna vertaisverkoissa myös verkon reunoilla olevat koneet voivat tuottaa kuluttamisen lisäksi. Yleensä tiedon tuottaminen tapahtuu verkon keskellä palvelimissa.

Vertaisverkossa resurssien haku suoritetaan välittämällä kyselyä verkossa koneelta toiselle. Kun kysely saapuu koneelle, jolta kyseltävä resurssi löytyy, se luo vastauksen, joka välitetään takaisin kyselyn lähettäjälle samaa reittiä kuin kysely tuli. Saadessaan vastauksen kyselyynsä kone ottaa yhteyttä suoraan siihen koneeseen, jolta resurssi löytyi ja lataa esimerkiksi tiedoston. Tiedostojen jakamiseen tarkoitettu Gnutella on puhdas vertaisverkkoprotokolla.



Kuva 2: Resurssien haku puhtaassa vertaisverkossa

Puhtaiden vertaisverkkojen lisäksi on olemassa ns. hybridejä vertaisverkkoja (hybrid peer-to-peer). Musiikinlevityksestä tutuksi tullut Napster on yksi tällaisista järjestelmistä. Siinä välittäjänä toimiva palvelin ylläpitää hakemistoa käyttäjiltä löytyvistä musiikkitiedostoista. Kukin käyttäjä ilmoittaa välittäjäpalvelimelle tietonsa ja tarjoamansa tiedostot. Käyttäjän halutessa jonkin häneltä löytymättömän tiedoston kysytään välittäjältä, keneltä kyseinen tiedosto löytyy. Varsinainen tiedoston lataus tapahtuu suoraan koneelta toiselle. Napsterissa haku tapahtuu palvelin-asiakas –mallin mukaan, mutta tiedoston lataus on puhdas P2P.



Kuva 3: Hybridi vertaisverkko

Vertaisverkon etuja ovat sen vikasietoisuus, skaalautuvuus ja tiedon korkea saatavuus. Koska verkossa ei ole keskitettyä pistettä, jonkin koneen poistuminen verkosta ei lamauta koko järjestelmää.

3 Chedar

Chedar on resurssien etsintään kehitetty vertaisverkkoalusta. Sen tarkoituksena on, että verkon koneet eli solmut kommunikoivat keskenään täysin itsenäisesti ilman keskitettyä pistettä.

Verkon topologian hallintaan liittyviä viestejä ovat yhteyspyynnöt ja -vastaukset, naapurien kyselyt ja vastaukset sekä koneen tarjoamien resurssien kyselyt ja vastaukset. Näiden viestien avulla luodaan yhteyksiä sekä kerätään naapureista tietoa. Varsinainen pääasia on resurssikyselyiden ja vastauksien käsittely. Kun tarvitaan jotakin resurssia, lähetetään resurssikysely muille solmun naapureille kulloinkin halutun hakualgoritmin avulla. Mikäli resurssi löytyy, solmu lähettää vastauksen, joka kuljetetaan alkuperäisen kyselyn lähettäjälle.

Chedar tallentaa aktiivisten yhteyksien lisäksi myös historiatietoja yhteyksistä. Historiassa on tallessa kaikki yhteydet, jotka solmu on joskus saanut tietoonsa. Solmu pitää yllä aktiivisista yhteyksistä yhteyden naapureiden lukumäärää ja kaikista osumatietoja sekä tietoa, koska yhteys on viimeksi vastannut ja koska sinne on lähetetty pyyntö ja onko tuo lähetys onnistunut vai epäonnistunut. Osumatietoja on kolmenlaisia. Ensimmäistä kasvatetaan joka kerta, kun yhteydeltä saadaan vastaus resurssikyselyyn, toista kasvatetaan, kun yhteyden resurssia ollaan käytetty ja kolmatta päivitetään, kun halutaan arvostella käytetty resurssi. Lisäksi aktiivisista yhteyksistä pidetään yllä niiden naapurien suhteellisia osumia, jotka kertovat paljonko vastauksia on saapunut koneelle sen yhteyden kautta. Kun vastausviesti saapuu koneelle, se lisää sen yhteyden osumia, joka vastauksen lähetti. Sen lisäksi lisätään viestin välittäneen yhteyden naapurin suhteellisia osuma-arvoja ja overtaking-algoritmilla tarkistetaan hypätäänkö yhteys yli.

Chedarissa ei ole kiinnitettyä lukumäärää solmun yhteyksille vaan yhteyksiä lisätään ja poistetaan solmun kautta kulkevan liikenteen mukaan. Mikäli viestien koko sekunnissa ylittää käyttäjän määrittelemän arvon, Chedar tiputtaa pois huonoimman yhteyden. Mikäli liikenne on vähäistä, Chedar pyrkii aktiivisesti luomaan uusia yhteyksiä.

3.1 Algoritmit

3.1.1 Overtaking

Overtaking-algoritmia käytetään topologian optimoinnissa.. Algoritmin tarkoituksena on, että kone siirtyy lähemmäksi hyviä koneita eli niitä, joilta saadaan eniten vastauksia, hyppäämällä yli yksi naapuri.

Vastausviestin saapuessa ja osumatietojen lisäyksien jälkeen overtaking-algoritmilla katsotaan, onko jonkin yhteyden naapurin suhteellinen osuus kaikkien yhteyden naapurien ja sen osuma-arvojen summaa suurempi, ja jos näin on, luodaan yhteys tähän naapuriin ja tiputetaan pois nykyinen yhteys.

Esimerkiksi yhteyden säilyttämät tiedot voisivat olla seuraavanlaiset:

	Osumatiedot
Yhteys	2
1. naapuri	7
2. naapuri	19

Tässä tapauksessa toisella naapurilla on enemmän osumia kuin muilla yhteensä. Näin ollen yhteys tiputettaisiin pois ja tilalle luotaisiin yhteys toiseen naapuriin.

3.1.2 Node Selection

Liittyessään vertaisverkkoon Chedar-solmu etsii ensimmäisenä luotavaa yhteyttä niiden yhteyksien joukosta, joihin se oli ennen poistumistaan verkosta liittyneenä. Mikäli näitä yhteyksiä ei ole tai koneen pitää myöhemmin lisätä uusi yhteys, se etsii sopivaa yhteyttä historiatiedoista. Koska yhteyttä ei haluta yrittää samaan yhteyteen kuin joka on juuri tiputettu pois, haetaan vain niiden yhteyksien joukosta, joihin ei ole yritetty yhteyttä määrätyn ajan sisällä onnistuneesti.

3.1.3 Node Removal

Valittaessa poistettavaa yhteyttä, pyritään poistamaan huonoin mahdollinen yhteys. Chedarin kannalta huonoin yhteys on se yhteys, jonka osuma-arvo on kaikista pienin.

3.1.4 Overload Estimation

Algoritmi seuraa liikenteen määrää kilotavuina sekunnissa ja sen ylittäessä annetun rajan tiputetaan Node Removal-algoritmilla yksi yhteys pois. Mikäli liikenteen määrä jää alle määritellyn rajan, algoritmi pyrkii muodostamaan uusia yhteyksiä.

3.1.5 Resource Reply -viestin reititys

Jokainen kone tallentaa välittämistään resurssikyselyviesteistä kyselyn id:n sekä kaksi edellistä konetta, jonka kautta viesti koneelle saapui.

Vastausviesti reititetään takaisin kyselyn lähettäjälle samaa reittiä kuin kysely saapui eli lähetetään sille yhteydelle, jolta kysely tuli. Mikäli yhteys ei ole enää kunnossa, esimerkiksi naapuri on poistunut verkosta, yritetään luoda yhteys edelliseen solmuun ja lähettää viesti sille. Mikäli tämäkään ei onnistu, viimeisenä keinona yritetään ottaa yhteys kyselyn lähettäjään ja välittää vastausviesti suoraan lähettäjälle.

Lähteet

[1] Oram Andy, ”Peer-to-Peer: Harnessing the Benefits of a Distruptive Technology”, O’Reilly Books, 2001.

[2] Schollmeier R., *A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications*, In Proceedings of the First International Conference on Peer-to-Peer Computing, 2002.