

Resurssien kohdistus

Sakari Savolainen

Helsinki 14.2.2012

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

Pro Gradu –seminaarin kirjallinen alustus

Sisältö (sivunumeroitu osuus on tässä asiakirjassa)

Abstrakti	1
1 Johdanto	1
2 Avainsanoja ja käsitteenmuodostusta	2
3 Avun kohdistusmekanismeja	3
3.1 Mekanismien esittelyä	3
3.1.1 GRID-teknologia	4
3.1.2 Verkkoyhteisöt ja vertaistuki	7
3.1.3 Expertise Finding -teknologia	10
3.1.4 I-Help-järjestelmä	13
3 Mallinnus ja agentit	15
3.1 Henkilökohtaiset agentit	15
3.1.5 Internetin hakukoneiden kohdistusmekanismit	19
3.2 Apu-mekanismi	20
3.2.1 Apu-mekanismin konteksti	21
3.2.2 Match-algoritmi	22
3.2.3 Kohdistusmekanismin toiminta	27
3.3 Mekanismien vertailua	
3.3.1 GRID-teknologia ja Apu-mekanismi	
3.3.2 Verkkoyhteisöt, vertaistuki ja Apu-mekanismi	
3.3.3 Expertise Finding -teknologia ja Apu-mekanismi	
3.3.4 I-Help-järjestelmä ja Apu-mekanismi	
3.3.5 Internetin hakukoneiden kohdistusmekanismit ja Apu-mekanismi	
4 Apu-mekanismin evaluointi	
5 Tulokset	
Lähteet	
Liitteet	
LIITE 1 Dictionary	
LIITE 2 ER-kaaviot	
LIITE 3 Taulukaaviot	
LIITE 4 Taulut	
LIITE 5 Testikyselyjen tulokset	
LIITE 6 Ohjelman käyttöohje	
LIITE 7 Moduulikaaviot	
LIITE 8 Moduulikuvaukset	
LIITE 9 Ohjelmakoodi	

Abstrakti

Tietotekniikkaa hyödyntävissä organisaatioissa, yhteisöissä ja yksityisillä ihmisillä ja heidän hoitamisissaan työ- ja muissa tehtävissä esiintyy paljon tietoteknisen tuen ja avun tarvetta. Mikäli avun löytää itse tai työkaverilta helposti, asiat ratkeavat nopeasti. Mutta mikäli apua ei ole käden ulottuvilla, sitä pitäisi saada muualta. Apuresursseja on olemassa kaikissa organisaatioissa ja yhteisöissä ja resurssit täytyisi saada kohdistettua nopeasti ja tehokkaasti tarvitsijalle. Resurssien kohdistusmekanismeja on kaikkialla ja niiden ominaisuudet vaihtelevat. Tässä työssäni vertaan muutamia avunkohdistusmekanismeja itse kehittämäni kohdistusmekanismiin. Vertailupohjana käytän mekanismeja, joita on GRID-teknologiassa, vertaistuessa, verkkoyhteisöissä, expertise finding -teknologiassa, I-Help-järjestelmässä ja Internetin hakukoneissa. Oman mekanismini olen evaluoinut järjestelmän itsensä keräämän transaktioaineiston perusteella. Testitapaukset ovat olleet oppilaitoksen tyypillisiä avunhakutapauksia.

1 Johdanto

Tarpeita ja niitä täyttäviä resursseja ja resursseja tarpeisiin kohdentavia mekanismeja on kaikkialla. Lintuemo syöttää poikasiaan, postinkantaja toimittaa postit perille verkkomyymälän sivusto huolehtii tilauksista ja toimituksista. Tarkastelen työssäni tietotekniikkaan liittyviä kohdistusmekanismeja, jolloin tietotekniikka on sekä auttamassa avun tarvitsijoita että aiheuttamassa avun tarvetta. Tietotekniikan parissa työskentelevät opetusalan ammattilaiset ja oppilaitosten henkilökunta tarvitsevat usein tukea tietoteknisiin kysymyksiin ja ongelmiin. Yksi syy tähän on tietotekniikan runsas hyödyntäminen opetusallakin. Apua saattaa saada kollegoilta, tutuilta tai tietoteknisestä tukipalvelusta sekä Internetistä ja kirjallisista lähteistä. Mutta aina ei sopivaa tukea ole saatavilla ja täytyy itse yrittää tarkoa pulma. Apua kannattaa pyytää, koska usein ongelmat ratkeavat siten nopeammin, varsinkaan, jos omat yritykset ratkaista pulmaa eivät ole tuottaneet tulosta. Kuitenkin, jos paikkakunta- tai organisaatiokohtaisesti olisi verkosto ja työkalu, jonka avulla voisi tarjota apua ja jolta voisi tarvittaessa pyytää apua, olisi tällainen vapaaehtoisuuteen perustuva ja hyvin toimiva vertaistuki usein korvaamaton ja verkosto voisi palvella näin saman alan toimijoita ja kehittää alaa yleensäkin. Kaikilla ihmisillä on vahvuuksia työyhteisössään. Nämä vahvuudet voi asettaa vuorovaikutteisesti tarjolle, auttamaan samankaltaisia tehtäviä tekeviä, tietotekniikan parissa työskenteleviä tai tietotekniikkaa hyödyntäviä henkilöitä. Rajoituksia tällaiseen toimintaan aiheuttavat esimerkiksi liikesalaisuuksien varjelu tai tilanne, jossa työaikaa pitäisi käyttää toisen organisaation hyväksi ilman korvausta.

Tässä työssä keskityn Riihimäen seudun opettajien ja muiden alalla toimivien ja tietotekniikan parissa työskentelevien henkilöiden auttamisen tarkasteluun. Henkilöitä ovat opettajat, hallintohenkilökunta, free lance -kouluttajat jne. Toimialue (domain) on tässä työssä rajattu edellämainittuihin henkilöryhmiin ja tietoteknisiin kysymyksiin. Tarkastelua on myöhemmin kuitenkin helppo laajentaa niin haluttaessa maantieteellisesti, muihin organisaatioihin ja myös uusille aloille. Toisaalta sovelluksen käyttöä ei rajoiteta eli se on täysin avoin. Tämän työn tarkastelu rajoittuu edellä kuvatulla tavalla, jolloin käsittelen tiettyä, tarkasti rajattua henkilö- ja asiasegmenttiä.

Maantieteellisesti virtuaalisesti muodostuva yhteisö voisi sijaita missä tahansa, koska datan tuottaminen tietokantaan, testaus ja evaluointi tapahtuvat automaattisesti ja niin, että suoritan itse testiajot ja kyselyt, ilman, että ulkopuoliset henkilöt testaisivat mekanismeja. Tämä vapauttaa testustilaisuuksien järjestämiseltä ja tekee evaluoinnin joustavaksi. Sovelluksen asettamista tuotantokäyttöön voi myöhemmin harkita.

Konstruoimani järjestelmä on auttamispalvelu, joka koostuu ihmisistä ja tietokantapohjaisesta verkkosovelluksesta, jota käytetään avoimen Internet-palvelun avulla. Ihmiset antavat ja pyytävät apua ja Internetissä toimiva sovellus yhdistää avun tarvitsijoita ja avun saajia tarjoamalla hakijalle tietoa avun antajista. Kyse on siis pääsääntöisesti vertaistuesta, vaikka ei rajaa pois muunluonteistakaan apua, esimerkiksi organisaatioiden tietoteknistä tukea.

Ohjelma on Internet-selaimen välityksellä toimiva avoin verkkosovellus, joka etsii avun hakijoille sopivia avun tarjoajia ja yhdistää nämä annettujen hakuparametrien ja kohdistusmekanismin avulla toisiinsa. Avunsaaja ottaa sitten yhteyden

avun tarjoajaan sähköpostitse, puhelimitse tai muilla keinoin ja henkilöt sopivat jatkotoimenpiteistä keskenään. Näin järjestelmä on karkea, 'robusti', eikä rajoita tai pakota mihinkään ja konkreettiset auttamistoimenpiteet jäävät henkilöiden yhteydenottojen ja keskustelujen varaan.

Ohjelmaa käytetään Internet-selaimella siten, että avun tarvitsija valitsee ennalta määritellyn neliportaisen luokittelun mukaan parhaiten hänen tarpeitaan vastaavan hakusanayhdistelmän ja kirjoittaa lisäksi vapaasanakenttään mahdollisen lisäavainsanan. Hakusanojen ja mahdollisesti ohjelman vapaasanasta löytämän lisävaimen perusteella ohjelma etsii parhaiten tarpeeseen sopivan avun. Sopivia henkilöitä voi löytyä useita, jolloin tulos listataan hakijalle siten, että listassa sopivin on ensimmäisenä. Avun tarjoajien määrän listassa on nollasta kolmeen. Hakutapahtumista kertyy ja tallentuu tietoja, mm. haun onnistumisesta ja haun tyyppistä. Myös näitä tietoja käytetään myöhemmissä hauissa hakukriteereinä.

Järjestelmän avulla tarjotaan ja haetaan ensisijaisesti apua käytännön tietoteknisiin pulmiin (it- tai ict-pulmiin; Information Technology; Information and Communication Technology) ja -tilanteisiin, joita voisivat olla jonkin sovellusohjelman käyttöön liittyvät kysymykset, tietoteknisen kurssin sijaisen saaminen pitämään jokin oppitunti, sopivan materiaalin löytäminen (tietotekniikan) opetukseen, ohjelman tai laitteen asentaminen, taulukkolaskentaongelman selvittäminen tai ohjelmointipulman ratkaisu.

Perustietoja järjestelmään syöttävät henkilöt itse ja keskitetty syöttökin on mahdollista esimerkiksi oppilaitoskohtaisesti, jolloin joku sovittu henkilö kerää tiedot ja syöttää ne järjestelmään. Tässä työssä on lähtökohtana se, että riittävä määrä perustietoa on jo olemassa eli oletuksena on se, että tiedon kriittinen massa järjestelmän toiminnan kannalta on jo saavutettu.

Varsinainen kysymykseni hakee vastausta siihen, toimiiko tällainen mekanismi ja toimiiko se hyvin nyt asetetuilla ehdoilla - saisivatko ihmiset sen avulla apua pulmiinsa. Mikä on erona ja/tai yhtenevyytenä muihin vastaavatyypisiin mekanismeihin, vertaistukeen mm. opiskelijoiden keskinäiseen auttamiseen rakennetuissa järjestelmissä tai muunlaisissa resource matching -palveluissa. Suomenkielinen vastine resource matching toiminnolle voisi olla resurssi-tarve-kohdistus tai apu-tarve-kohdistus tai vain avun kohdistus. Millaisia vahvuuksia ja heikkouksia tällä järjestelmällä on verrattuna muihin ja miksi? Osakysymyksenä on lisäksi match-tekniikan toimivuus verrattuna muihin vastaavanlaisiin tekniikoihin.

Mekanismi toimii siten, että käyttäjä pääsee käyttämään resource matching -kohdistusohjelmaa Internet-selaimen avulla missä tahansa, missä on Internet-yhteydellä varustettu tietokone. Käyttäjä valitsee ohjelman tarjoamista valmiista listoista hakusanayhdistelmän, joka parhaiten vastaa hänen avuntarvettaan. Hakusanaluokituksen pohjana on käytetty ACM-CCS98-luokitusta, jota on muokattu ympäristöön sopivaksi mm. yksinkertaistamalla sitä. Lisäksi käyttäjä voi kirjoittaa vapaatekstikenttään hakusanan, joka ilmaisee keskeisen avuntarpeen ja tarkentaa sitä. Ohjelma etsii hakuparametrien avulla tietokannasta parhaiten pyyntöä vastaavat avun tarjoajat ja listaa heidät ruudulle paremmuusjärjestyksessä.

Tämän jälkeen hakija lähettää sähköpostia avun tarjoajalle tai ottaa muuten yhteyttä häneen ja voi samalla valaista tarvettaan lisää. Mahdolliset aputapahtumaan liittyvät käytännön toimet ja mahdolliset korvaukset sovitaan myös yhteydenoton jälkeen tapauskohtaisesti. Oleellista on sopivien avuntarjoajien löytäminen. Työn tavoite on tutkia resource matching -mekanismeja ja sen mahdollisuutta palvella apuneuvona määritellyssä kontekstissa ja tehdä maisteritutkinnon opinnäytetyö, pro gradu -tutkielma.

Teksti on tutkielmassa kirjoitettu siten, että sen voi lukea ilman tietoteknisen toteutuksen seuraamista. Näin lukija voi keskittyä mekanismin ideaan ja sen evaluointiin. Toisaalta tietoteknisestä toteutuksesta kiinnostunut lukija löytää työstä ja sen liitteistä täydellisen kuvauksen myös tietoteknisen toteutuksen yksityiskohdista, joiden perusteella ohjelmisto on konstruoitavissa ja testattavissa.

2 Avainsanoja ja käsitteenmuodostusta

Työ aihepiiriin kuuluu seuraavia keskeisiä käsitteitä:

resource matching, resurssi-tarve-kohtaaminen, apu-tarve-kohtaaminen, avun kohdistus, kohdistusmekanismi, MAS, Multi Agent Systems, moniagenttiarkkitehtuuri, vertaistuki, vertaisapu, peer help, peer-to-peer help, sovellusohjelmointi, Internet-sovellus, verkkosovellus, tietokantasovellus, eksperttiys

3 Avun kohdistusmekanismeja

Ihmiset ratkovat organisaatioissa ongelmia joka päivä ja tarvitsevat usein muiden apua ratkaisujen löytämiseen. Organisaatioille ja tieteellisille yhteisöille ovat tärkeitä sellaiset järjestelmät, jotka auttavat etsimään sopivaa tietoa ja eksperttiyttä.

Tietotekniikkaa hyödynnettäessä käyttäjille syntyy avun tarvetta, kysymyksiä ja ongelmia. Tämä lienee kaikille tietotekniikan käyttäjille tuttua. Monet tilanteet ja ongelmat pystyy selvittämään itsekin, mutta silti kaikki tarvitsevat joskus muiden apua. Tällöin ongelmaa selvittävä henkilö etsii sellaisia kollegoja tai muita henkilöitä, jolla olisi riittävät tiedot ja taidot auttaa.

Oppilaitosympäristössä käytetään hallinnon ohjelmistoja, toimisto-ohjelmistoja, sähköpostia, Internet-materiaalia, intranettiä, lähiverkkoa, erilaisia palvelimia, opetusohjelmia ja oppilaitoksesta riippuen opetuksessa tarvittavaa työkaluohjelmistoa, kuten esimerkiksi grafiikkaohjelmia, suunnitteluohjelmia ja ohjelmointikielten kääntäjiä ja monenlaisia kehitysympäristöjä. Tietoteknisin välinein hoidetaan lukujärjestysten laadinta, opiskelijatietojen rekisteröinti, asiakirjat tuotanto, Intranet- ja Internet-sivustojen ylläpito, opetusmateriaalien laadinta ja oppituntien pitämisessä hyödynnetään myös tietotekniikkaa, mm. henkilökohtaisilla koneilla ja dataprojektorin välityksellä. Opiskelijat käyttävät oppitunneilla ja niiden ulkopuolella sähköpostia, toimisto-ohjelmistoa ja oppimisalustoja. Kommunikointiin käytetään sähköpostia henkilökunnan kesken ja opettajien ja opiskelijoiden välillä. Kurssien hallinnointiin käytetään lisäksi oppimisalustoja, kommunikointiohjelmistoja ja Internet-sivustoja palveluineen.

Avun tarve on monitahoista, lähtien yksinkertaisista ohjelmien käyttöön liittyvistä kysymyksistä aina vaativaa ja laajaa eksperttiyttä edellyttäviin vaikeisiin kysymyksiin. Yhteisöissä ja organisaatioissa avun saatavuus vaihtelee. Ohjelmat sisältävät usein opasteita, organisaatioilla on tukihenkilöitä ja tietoverkoissa on runsaasti erilaisia apua tarjoavia ohjelmia, palvelimia, keskustelualueita jne. Myös vertaisapua käytetään, kysymällä neuvoa esimerkiksi työkaverilta tai muulta tutulta.

Esimerkkinä on päätoimisen tietotekniikkaopettaja Elmon tilanne, jossa hän on rutiinitehtävänä tarkistanut pitämänsä kokeet ja luvannut opiskelijoille tulokset kurssin Internet-sivulle seuraavalla viikolla. Kokeiden tarkistaminen sujuu rutiinilla. Koe oli viimeinen välikoe ja kurssin arvosanat tulee myös määritellä. Kurssitulokset tulevat etuajassa kurssin sivustolle, mutta arvosanojen syöttäminen ei onnistukaan oppilaitoksen opiskelijahallinnon järjestelmään. Tukihenkilö on kaksi päivää koulutuksessa ensi viikolla ja nyt on viikonloppu. Arvosanojen syöttö voi toki odottaa, mutta Elmo päättää yrittää selvittää tilanteen heti maanantaiaamuna. Silloin hän tarvitsisi apua henkilöiltä, jotka tuntevat oppilaitoksen ohjelmistoa riittävästi.

3.1 Mekanismien esittelyä

Organisaatioissa ja järjestelmissä varastoidaan tietoa ja tuota tietoa haetaan moniin tarpeisiin. Tarve määritellään ja sen perusteella tehdään haku. Näin tapahtuu tarpeen täyttämiseksi tarkoitettu kohdistus, jossa resurssi ja tarve kohdistetaan toisiinsa. Erilaisia kohdistusmekanismeja on runsaasti ja niiden ominaisuudet vaihtelevat. Esittelen joitakin järjestelmiä, joissa on kohdistusmekanismi. Valitut mekanismit ovat tietoteknisiä ja järjestelminä poikkeavat toisistaan melkoisesti. Mainitut järjestelmät olen valinnut esiteltäviksi mielenkiintoisten kohdistusominaisuuksiensa takia. Näin on saatu laajempi kuva siitä, millaisia kohdistusmekanismeja on olemassa ja millaisina toteutuksina. Yhdistäviä tekijöitä mekanismeilla ovat (avun) tarpeen määrittely, resurssien tunnistaminen ja valinta sekä tulosten kohdistus hakijalle.

Valitut mekanismit ovat GRID-teknologian, verkkoyhteisöjen Internet-palvelujen, Expertise Finding -teknologian, I-Help-järjestelmän ja Internetin hakupalvelujen osana.

GRID-teknologiassa kohdistetaan ja allokoidaan tietoverkkojen avulla globaalisti tietokonekapasiteettia laskentatehtävien tarpeisiin. GRID:ssä tehdään resurssien jakamista ja koordinoitua ongelmanratkaisua dynaamisten ja multi-institutionaalisten, virtuaalisten organisaatioiden välillä (ks. mm. [Tangmunarunkit et al. 2003] ja [Kee Y. et al. 2006]).

Verkkoyhteisöjen sivustoilla käytetään mm. verkoissa toimivia keskustelupalstoja ja FAQ (Frequently Asked Questions) -palveluja, hakupalveluja ja -tietokantoja, joilla avun hakija kohdistaa tarvittavan avun mm. muotoilemalla keskustelupalstalle kysymyksen. Yhteisön jäsenistä yksi tai useampi voi vastata kysymykseen ja muodostunut keskustelu voi jatkua, kunnes kysyjä saa avun. Tällöin kohdistetaan yhteisön eksperttiyttä avun hakijalle. Yhteisöistä yksi esimerkki on suomalainen ohjelmointiin keskittynyt Mureakuha-sivusto [Mureakuha 2008]. Katso myös Ohjelmointiputka-sivusto [Ohjelmointiputka 2009], MySQL-tietokannan sivusto [MySQL 2008] ja Moodle-oppimisolun sivusto [Moodle 2009].

Expertise Finding -teknologioita, joilla kohdistetaan avun hakijoita ja eksperttiyttä, on rakennettu lukuisiin järjestelmiin, kuten esimerkiksi Answer Garden. CSCW (Computer-Supported Cooperative Work) on tutkimuskohde ja teknologia, jossa sosiaalisista verkostoista haetaan apua ja informaatiota kysymyksiin ja ongelmiin. Katso mm. [McDonald, Ackerman 1998].

Yliopistoissa on kehitetty ohjelmistoja, jotka tarjoavat opiskelijoille vertaisapua opiskelussa, kohdistuen avun hakijoita ja avun tarjoajia. Yksi tällainen mielenkiintoinen järjestelmä on Kanadassa, Saskachewanin yliopistossa kehitetty I-Help, Intelligence Helpdesk, joka tarjoaa vertaistukea opiskelijalta opiskelijalle ja toteutuksessa käytetään Javan agenttitekniologiaa. Agentit tekevät apuresurssien tunnistuksen ja kohdistuksen hakijalle. Katso mm. [Greer et al. 2001].

Internetissä toimivat hakukoneet käyttävät erilaisia algoritmeja, jotka kohdistavat hakijalle mahdollisimman hyviä hakutuloksia. Haut tapahtuvat enimmäkseen asettamalla tekstimuotoisia hakutekijöitä, joiden perusteella hakukoneet tekevät valinnan. Katso mm. [Ekonoja, Lahtonen, Mäntylä 2003].

3.1.1 GRID-teknologia

Grid on uusi teknologia, joka mahdollistaa resurssien jakamisen ja koordinoitun ongelmanratkaisun dynaamisissa ja multi-institutionaalisissa organisaatioissa. GRID-teknologiassa kohdistetaan globaalisti tietokoneiden laskentatehoa missä tahansa esiintyviin laskentatarpeisiin. Gridejä (verkoja) käytetään yhdistämään maantieteellisesti erillään olevat, hajautetut laskenta- ja dataresurssit ja toimittamaan näiden resurssien palvelut heterogeenisille käyttäjäyhteisöille. Katso [Eurogrid 9999], [Griphyn 9999] ja [Teragrid 9999]. Resurssit voivat kuulua eri instituutioille, omata erilaisen käyttöpolitiikan ja erilaiset vaatimukset hyväksyttävillä pyynnöillä. Grid-sovelluksilla voi silti olla myös erilaisia rajoitteita, jotka voidaan tyydyttää vain tietyn tyyppisillä resursseilla, resursseilla, joilla on jotain erityisominaisuuksia.

Ennen kuin resurssi (tai joukko resursseja) voidaan allokoida suorittamaan jotakin sovellusta, käyttäjän tai agentin on valittava resurssit, jotka ovat sopivia sovelluksen vaatimuksiin nähden. Tätä prosessia, jossa valitaan resursseja sovellusten vaatimusten mukaan, kutsutaan resurssien kohdistusmekanismiksi (resource matching mechanism). Grid-ympäristössä, jossa resurssit tulevat ja menevät, on toivottavaa ja joskus välttämätöntä automatisoida resurssien kohdistus, jotta sovelluksien vaatimukset tulisivat karkeasti ottaen huomioitua.

Nämä vaatimukset ja politiikat on usein ilmaistu erillisissä sovellus- ja resurssimalleissa, jotka pakottavat resurssivalitsimen tekemään semanttista yhteensovitusta niiden kahden välillä. Resurssien löytämisongelma Gridissä koskee resurssien liittämistä tehtäviin niin, että tehtävien tarpeet ja resurssipolitiikka tulevat täytettyä. Tangmunarunkit, Decker ja Kesselman [Tangmunarunkit et al. 2003] ehdottavat joustavaa ja laajennettua Gridin lähestymistapaa resurssien löytämisen ratkaisuksi, käyttämällä semanttisen webin teknologioita. He ovat suunnitelleet ja tehneet prototyypin ontologiaperustaisesta resurssivalitsimesta, joka käyttää hyväkseen ontologioita, taustatietämystä ja sääntöjä tehdäkseen resurssivalintaa Gridissä.

Tangmunarunkitin, Deckerin ja Kesselmanin mukaan [Tangmunarunkit et al. 2003] resurssien löytämisongelma (resurssien tunnistaminen) Gridissä koskee resurssien liittämistä tehtäviin niin, että tehtävien tarpeet ja resurssipolitiikka tulevat täytettyä. Juuri resurssien allokointi (Grid Resource Allocation Management, GRAM) on se mielenkiintoinen alue, joka liittyy kohdistusmekanismeihin ja niiden ominaisuuksiin ja siten tämän työn keskeisiin kysymyksiin.

Olemassa oleva resurssin kuvaus ja resurssin valinta Gridissä on erittäin rajoitettua perinteisellä tavalla tehtynä. Traditionaalinen resurssien kohdistus, josta ovat esimerkkinä Condor Matchmaker [Solomon et al. 1998] ja Portable Batch System [MRJ 2009], on tehty perustuen symmetriselle ja attribuuttipohjaiselle kohdistukselle. Näissä järjestelmissä, resurssien ilmoittamia arvoja verrataan töiden vaatimiin arvoihin. Jotta vertailu olisi mielekäs ja tehokas, resurssien tarjoajien ja kuluttajien täytyy hyväksyä attribuuttien nimet ja arvot. Täsmällinen kohdistus ja koordinointi tarjoajien ja käyttäjien välillä tekevät tällaisen järjestelmän joustamattomaksi ja vaikeasti laajennettavaksi, kun niihin halutaan lisätä uusia piirteitä tai käsitteitä. Lisäksi, heterogeenisessä ja moni-institutionaalisessa ympäristössä, kuten Gridissä, on vaikeaa pakottaa syntaksia ja semantiikkaa resurssikuvauksiin. Kuva 1 esittää GRID:n toimintaa. Resurssien pyytäjä (hakija, Resource Requester) voi pyytää resurssien allokointia, kun on saanut kohdistajalta tiedon sopivista resursseista.

GRID:n resurssikohdistus.

Kuva 1. Resurssien kohdistus GRID:ssä: kohdistaja (Resource Matchmaker) kerää resurssi-informaatiota (3) resurssien tarjoajista (Resource Provider). Kun hakija (Resource Requester) lähettää pyynnön (1) kohdistajalle kohdistusta varten, kohdistaja hakee sopivat resurssit ja palautaa (2) hakijalle tiedon niistä. Hakija voi sitten pyytää resurssia käyttöönsä (4). Kuvan on kirjoittaja muokannut mukaellen lähdettä [Tangmunarunkit et al. 2003].

Symmetrinen ja attribuuttipohjainen kohdistus on jäykkää ja vaikeaa laajentaa. Tangmunarunkitin, Deckerin ja Kesselmanin [Tangmunarunkit et al. 2003] ehdottama ontologioihin perustuva kohdistaminen on joustavampaa, perustuen semanttisen webin käyttöön. Ontologia-perustainen kohdistus muodostuu kolmesta tekijästä. Ensimmäinen on ontologia, joka käsittää käyttöalueen mallin ja sanakirjan esitettävälle resurssi-ilmoituksille ja työtilauksille. Toinen on käyttöalueen taustatietämys, käsittäen lisätietoa käyttöalueesta ja kolmantena ovat kohdistussäännöt, jotka määrittävät sen, milloin resurssi vastaa työtilauksen kuvausta.

Kohdistusongelma voidaan esittää tällöin formaalisti seuraavasti, kuten kuvassa 2. Alfa on kaikkien tallennuspaikan resurssi-ilmoitusten joukko, O on käyttöalueen ontologia, B on käyttöalueen taustatietämyksen muodostama joukko ja R on binaaripredikaatti match:n määrittelemä kohdistussääntöjen joukko ja johto-operaattori on, kuten johto-operaattori klassisessa logiikassa. Tällöin annetulle kyselylle tai ilmoitukselle q on kohdistus joukko.

$$\{\alpha \mid OUBURUM \models_{match} match(\alpha, q)\}$$

Kuva 2. Kohdistusongelman formaali esitys joukkona.

Ontologia-perustainen resurssien kohdistus

Ontologia-perustainen kohdistajamme on kehitetty semanttisen webin teknologioille. Tässä kappaleessa kokoamme ensin kuvatut kohdistajamme piirteet, kuvaten sen arkkitehtuuria ja lopuksi metodologiaa. Ontologia-perustaisen kohdistajan piirteet ovat

resurssien ja pyyntöjen asymmetrinen kuvaus; kehyksessämme resurssit ja pyynnöt on mallinnettu ja kuvattu erikseen; semanttisesta kohdistuksesta kahden mallin välillä huolehditaan; asymmetrisestä kuvaamisesta johtuen ei resurssien tarjoajien ja kuluttajien välillä tarvita ennen uuden kuvaussanaston lisäämistä; tämä ei ole mahdollista

Symmetrisessä ja attribuuttiperustaisessa kohdistuksessa

jakaminen ja ylläpidettävyys; ontologiat ovat jaettavissa ja helpompia ylläpitää ja ymmärtää kuin yksinkertainen attribuuttilista

kahdenväliset rajoitteet; pyynnön kuvaus sallii spesifioida rajoitteet pyynnön ehdoilla; samoin, jokainen resurssi voi myös itsenäisesti ilmaista käyttöpolitiikkansa (eli määrittellen, kenellä on pääsy sallittu) estäen tarvittaessa pääsyn jollekin sovelluksille/pyynnöille; kohdistaja ottaa huomioon jokaisen resurssin ja pyynnön rajoitteet etsiessään kohdistusta

mahdollisuus kuvata kohdistettavien ehtojen ensisijaisuutta; sekä pyyntö että resurssi voivat määrittellä ehtojen tärkeysjärjestystä, mikäli pyynnöissä tai resursseissa löytyy useampia osumia

monitahoinen kohdistus; käyttäjä voi lähettää pyynnön, joka tarvitsee useita resursseja samanaikaisesti ja jokainen niistä on määritelty itsenäisellä määrittelylauseella; joukkokohdistuspiirre lisätään myös tulevaisuudessa

yhteneväisyystarkistus; kohdistaja voi käyttää käyttöalueen (domain) tietoa tunnistaakseen epäyhtenevyydet resurssikuvauksissa ennen kuin se hyväksyy resurssin; yhteneväisyystarkistus voidaan myös tehdä pyyntönä, jotta voidaan tarkistaa, ettei resurssivaatimuksissa ole ristiriitoja; esimerkiksi resurssin tai pyynnön määrittelyssä `OperatingSystem="Windows2000"` ja `CPUFamily="Sparc"` pitää hylätä

ilmaisukykyisyys; asymmetrisestä kuvaamisesta johtuen pyyntö voidaan muotoilla erityisesti tietyn toimialueen sovelluksille; korkean tason sovelluksen piirteitä voidaan näin tarjota käyttäjälle; lisäksi kohdistaja voi määrittellä automaattisesti tiettyjä korkean tason vaatimusmäärittelyjä

joustavuus ja laajennettavuus; uusia käsitteitä voidaan helposti lisätä ontologiaan, esimerkiksi tiukasti parina toimivat koneet tai MPI-sovellus (Message Passing Interface, katso [MPI 2009]); lisäksi uudet rajoitteet, esimerkiksi MPI-sovellus vaatii tiukasti parina toimivat koneet ja tämä voidaan helposti lisätä sääntöihin

Kohdistussäännöt määrittelevät kohdistusrajoitteet pyyntöjen ja resurssien välillä. Nämä säännöt on toteutettu käyttäen TRIPLE-sääntökielellä. Kohdistussäännöt voisivat näyttää esimerkiksi seuraavalta (kuva 3).

```
FORALL Data, Background @match(Data,Background,Ontology){
FORALL X,Y X[matches->Y] <-
  X[rdf:type->GR:JobRequest]@Data
  and Y[rdf:type->GR:ComputerSystem]@rdfschema(Data,Ontology)
  and ((X.GR:RequestResource.GR:RequiredMemory)@Data)[matchesMEM->(Y.GR:RunningOS)@Data]
  and ((X.GR:RequestResource.GR:RequiredOS)@Data)[matchesOS->(Y.GR:RunningOS)@Data]
  and ((X.GR:RequestResource.GR:RequiredFS)@Data)[matchesFS->(Y.GR:HostedFileSystem)@Data]
  and ((X.GR:RequestResource.GR:RequiredCPU)@Data)[matchesCPU->Y].
// checking OperatingSystem requirement
FORALL X,Y X[matchesOS->Y] <-
  X[rdf:type->GR:OSRequirement]@Data
  and Y[rdf:type->GR:OperatingSystem]@Data
  and ((X.GR:OSType)@Data)[substitutes->(Y.GR:OSType)@Data]@Background.
// checking FileSystem Requirement
FORALL X,Y X[matchesFS->Y] <-
  X[rdf:type->GR:FSRequirement]@Data
  and Y[rdf:type->GR:FileSystem]@rdfschema(Data,Ontology)
  and (X.GR:MinDiskSpace)@Data =< (Y.GR:AvailableSpace)@Data.
}
```

Kuva 3. Katkelma TRIPLE-sääntökielellä ilmaistusta kohdistussäännöistä.

90pivien kompleksisten resurssijoukkojen löytäminen ja hankinta laajoista hajautetuista tietokonejärjestelmistä on todellinen haaste ja myös kriittistä sovellusten tehon kannalta. Tämä on esitetty uudenlainen tapa resurssiervalintaprobleemalle ja uusi ratkaisu resurssien valinta- ja sitomisongelmalle, jota kutsutaan integroiduksi valinnaksi ja sitomiseksi. Yhdistelyoperaattorit resurssienkuvauskielessämme ja tehokkaassa datan organisoinnissamme mahdollistavat lähestymistavassa allokoida kompleksisia resurssikokoelmia tehokkaasi, myös kilpalutilanteissa. Empiirinen evaluaatio osoittaa, että integroitu lähestymistapa voi tuottaa merkittävästi parempia ja korkealaatuisempia ratkaisuja suuremmilla onnistumismäärillä ja alemmilla kustannuksilla kuin traditionaalinen, erillinen lähestymistapa. Onnistumistaso integroidulla lähestymistavalla sietää 15-60 % alemman resurssien saatavuuden kuin erillinen lähestymistapa. Lisäksi kaikilla pyynnöillä on lähes 98 % onnistumismahdollisuus ja ne voidaan palvella kuudessa sekunnissa miljoonan isäntäkoneen populaatiolla [Kee Y. et al. 2006].

Erillinen valinta- ja sitomisalgoritmi on seuraava

1. Valitse jokaiselle spesifikaation entiteetille valintojen joukko.
2. Ota yksi jokaisen entiteetin valinnoista ja muodosta ratkaisukandidaatti, asettaen jokainen valinta, huomioiden entiteettien väliset suhteet. Toista tätä, kunnes N ratkaisukandidaattia on muodostettu.
3. Poista paras ratkaisukandidaatti ja yritä sitoa se.
4. Jos mikä tahansa entiteetti epäonnistuu sitomaan ratkaisua, koko ratkaisu epäonnistuu. Pyri sitomaan seuraavaksi paras kandidaatti. Jos kaikki N ratkaisukandidaattia epäonnistuvat sitomaan ratkaisua, palauta virhe.
5. Jos jokainen entiteetti on sitonut ratkaisukandidaatin onnistuneesti, palauta ratkaisu sovellukselle.

[Kee Y. et al. 2006]

Ja tehokkaampi integroitu valinta- ja sitomisalgoritmi on seuraava

1. Tunnista spesifikaation komponentit ja ryhmitä läheisesti toisiinsa liittyvät komponentit yhteen.
2. Valitse jokaiselle komponentille N parasta valintaa.
3. Poista jokaisen valinnan paras valinta ja pyri sitomaan ne samanaikaisesti.
4. Jos komponentti epäonnistuu sitomaan valintansa, vain komponentti epäonnistuu. Jos mikä tahansa komponentti epäonnistuu kaikkien N valintansa kanssa, palauta virhe.
5. Jos jokainen komponentti onnistuu sitomaan valintansa, lähetä sidotut resurssit ratkaisuksi koko määrittelylle, käyttäen hyväksi koottua rakennetta ja palauta ratkaisu sovellukselle. [Kee Y. et al. 2006]

3.1.2 Verkkoyhteisöt ja vertaistuki

Avoimen koodin kehitysyhteisöissä ja tietoteknisissä verkkoyhteisöissä käytetään Internetissä toimivia keskustelupalstoja ja usein kysyttyjen kysymysten (FAQ, Frequently Asked Questions) palstoja, hakupalveluja ja -tietokantoja, HelpDesk-palvelua, download-palvelua ja Wiki-palvelua, joista avun hakija voi kohdistaa tarvittavan avun sivustossa tarjottua hakupalvelua käyttäen. Vertaistuki toteutuu yhteisöjen Internet-sivujen välityksellä siten, että keskustelupalstalle (foorumi), Chat-alueelle tai IRC-galleriaan kirjoitetaan kysymyksiä, joihin muut yhteisön jäsenet vastaavat. Verkkoyhteisöistä esimerkkejä ovat suomalaiset ohjelmointiin keskittyvät Mureakuha-sivusto [Mureakuha 2008] ja Ohjelmointiputka-sivusto [Ohjelmointiputka 2009]. Mureakuhassa on koodikirjasto useille ohjelmointikielille ja koodikokoelmaa voi selata lyhyiden selitteiden perusteella tai etsiä tarvitsemaansa hakutoiminnon avulla, asettamalla hakusanoja. MySQL-tietokannan sivusto [MySQL 2008] ja Moodle-oppimisolustan sivusto [Moodle 2009] ovat esimerkkejä avoimen koodin kehitysyhteisöistä. Myös näillä sivustoilla voi käyttää mm. tekstimuotoista hakupalvelua. Sivustoille on sijoitettu tietoa, jota pääsee lukemaan kuka tahansa. Lisäksi sivustojen laajempi käyttö edellyttää

rekisteröitymistä sivuston käyttäjäksi ja kirjautumista tämän jälkeen. MySQL on avoimeen koodiin perustuva relaatiotietokantaohjelmisto. Moodle on avoimeen koodiin perustuva oppimisolusta.

Avun kohdistus käyttäjälle tapahtuu FAQ-listoja selaamalla ja etsimällä asetettujen kysymysten avulla tarpeeseen sopivaa aineistoa. Chat-alueen ja IRC-gallerian käyttö on on-line-tyyppistä ja sopivaa apua löytää siten, että käy tekstuaalista keskustelua chat-alueella tai IRC-ralleriassa olevan tai olevien muiden vertaisten kanssa. Verkossa oleva tekstimuotoinen HelpDesk on off-line palvelu, jonne lähetettyihin viesteihin HelpDeskin hoitajat vastaavat esimerkiksi sähköpostilla. Download-palvelulla sivustoista voi ladata http- tai ftp-protokollaa käyttäen tiedostoja omalle koneelleen. Wikissä voi kirjoittaa ja muokata siellä olevaa materiaalia kaikkien sivuston käyttäjien hyödyksi.

Sivustoilla käytetään omia tai yleisiä hakupalveluja, kuten Googlea. Tekstimuotoisten hakusanojen asettaminen ja keskustelupalstojen käyttö on yhteisöjen sivustoissa tekijä, joilla apua ja sopivaa eksperttiyttä myös haetaan ja tämä liittyy keskeisesti tämän työni aiheeseen. Tekstiin perustuvan haun hakutapoja ovat avainsanahaku (keyword search), boolean haku (boolean search), esimerkkihaku (query by example), fraasihaku (phrase search), sanarunkohaku (stemming) ja sumea haku (fuzzy search).

Avainsanahaku hakee käyttäjän hakusanat sisältävät dokumentit ja yrittää järjestää tuloksen dokumenttien laadun perusteella. Boolean hakua (boolean search) käytetään tekemään mutkikkaampia hakuja käyttämällä loogisia operaattoreita AND (ja), OR (tai) ja NOT (ei). Joissakin hakukoneissa ei tarvitse käyttää AND-operaatiota vaan se on oletuksena ja hakutuloksessa vaaditaan kaikkien hakusanojen esiintymistä. Saman haun tekeminen joissain muissa hakukoneissa vaatii +-merkin lisäämistä jokaisen hakusanan eteen. NOT-operaatiota vastaa joissain hakukoneissa - (miinus, väliviiva) -merkki. Esimerkkihaku (query by example) toimii siten, että käyttäjä antaa dokumentin ja hakukone hakee samantyyppiset dokumentit. Google-hakukoneessa tämä onnistuu related-avainsanan avulla. Fraasihaussa (phrase search) hakusana sijoitetaan lainausmerkkien sisään, jolloin hakukone hakee koko merkkijonoa eli fraasia. Sanarunkohassa (stemming) haetaan sanoja pelkästään niiden alkuosan perusteella. Altavista-hakukoneessa sanarunkohaku määrittellään asteriski (*) -merkillä. Sumea haku (fuzzy search) tarkoittaa tietojen löytymistä, vaikka hakusanaa ei löytyisikään kirjoitusvirheen takia (katso mm. [Ekonoja, Lahtonen, Mäntylä 2003]). Tarkempi kuvaus Google-hakukoneen hakuagoritmista on jäljempänä kappaleessa 2.1.6 (Internetin hakukoneiden kohdistusmekanismit).

Neljän edellämainitun verkkoyhteisön Internet-sivut on esitetty alla olevissa kuvissa ja kussakin kuvassa on hakupalvelun tekstikenttä ruudun oikeassa laidassa.



Kuva 4. Mureakuha on ohjelmointiin keskittynyt suomalaisen yhteisön Internet-sivusto.



Kuva 5. Myös Ohjelmointiputka on ohjelmointiin keskittynyt suomalaisen yhteisön Internet-sivusto.



Kuva 6. Moodlen Internet-sivusto. Moodle on avoimeen lähdekoodiin perustuva oppimislustaohjelmisto, jonka ylläpitystyö on kansainvälinen.



Kuva 7. MySQL:n Internet-sivusto, joka on kansainvälisen relaatiotietokantaohjelmiston ylläpidosta vastaavan yhteisön sivusto.

3.1.3 Expertise Finding -teknologia

Eksperttiyden löytäminen (Expertise Finding) ja kohdistaminen hakijalle yrityksissä ja muissa organisaatioissa on ilmiönä kiinnostava ja sitä tutkitaan paljon. McDonald ja Ackerman [McDonald, Ackerman 1998] mainitsevat muutamia järjestelmiä, joissa tapahtuu eksperttiyden kohdistamista. Näitä ovat Answer Garden ja Answer Garden 2, Referral Web, Yenta ja Phoaks. Answer Garden on tehty helpottamaan epämuodollisia informaatiovirtoja ja kaapata niitä. Answer Garden 2:een, joka on uusi versio, tuli ulkoinen kohdistuskone. Referral Web auttaa löytämään tiettyä aihetta osaavia tutkijaeksperttejä kaikkien yhteistyössä olevien kirjoittajien joukosta. Yenta luo henkilöille henkilökohtaisia eksperttiysprofiileja tekstin sisällön analyysin perusteella. Phoaks paikallistaa henkilöitä, jotka ovat myötävaikuttaneet informaation syntyyn.

McDonaldin ja Ackermanin mukaan eksperttiyden löytäminen voidaan jakaa kahteen osaan, resurssin löytämiseen (identifiointiin) ja resurssin valitsemiseen. Eksperttiyden identifiointi on ongelma, jossa tulisi tietää, mitä informaatiota ja erityistaitoja muilla on. Eksperttiyden (resurssien) valinta tarkoittaa tarkoituksenmukaisten henkilöiden tai muiden resurssien valitsemista kulloiseenkin ongelmaan [McDonald, Ackerman 1998].

Eksperttiydessä on tasoja, vaikka järjestelmät eivät niitä kovin yleisesti tunnistaakaan. Millään mainituista järjestelmistä ei ole kovinkaan hienostunutta mallia eksperttiyden ja informaation hakemisessa. Sekä informaation hakijat että tarjoajat tekevät päättelyitä sen informaation validiudesta sen mukaan, millaista informaatiota itse saavat. Monet organisaatioon liittyvät ja sosiaaliset tekijät vaikuttavat eksperttiyden käyttöä rajoittavasti diagnostisissa tilanteissa. Eksperttiyttä voi saada teknisestä tuesta, virheenjäljityksellä tai muilla vastaavilla ohjelmallisilla toiminnoilla. Eksperttiyden paikallistaminen

Seuraavissa kappaleissa erotamme kaksi askelta eksperttiyden etsimisessä organisaatioissa, erottaen etsimisen tunnistamisen ja valinnan vaiheisiin.

Eksperttiyden tunnistus on ongelma, jossa tulee tietää, mitä informaatiota tai erityistaitoja muilla henkilöillä on. Ylläoleva esimerkki valaisi ongelmaa kuvaillen sitä, mitä tietämystä tai erityistaitoja kenelläkin on. Eksperttiyden valinta tarkoittaa tarkoituksenmukaisen eksperttiyden valintaa henkilöiden joukosta. Jos potentiaalisia eksperttejä on useita tai henkilöitä vaaditulla eksperttiydellä on useita, on valittava yksi tai useampia.

Teemme tämän erottelun analyttisesti, käsittäen, että se ei päde koko ajan ja kaikille asianosaisille. Haluamme painottaa, että eksperttiyden tunnistaminen ja eksperttiyden valinta ovat iteratiivisia, yhteenkietoutuneita käyttäytymismalleja, myös yleisesti työelämässä. Ja vielä, tämä erottelu perustuu teoriaan, dataan, kirjallisuuteen ja suunnittelukriteereihin:

S-dcog-teorian näkökulmasta, ratkaisten useita ongelmia yhteistyössä vaaditaan ensin tarvittavien resurssien tunnistamista ympäristössä ja onnistumista sitten niiden hankkimisessa (vaikka tämä saatetaankin tehdä iteratiivisesti). Joskus asianosaiset tekevät näitä kahta aktiviteettia tietoisesti, toisinaan taas aktiviteetit voivat rutinoitua.

Monet lähteistämme, mutta eivät kaikki, erottavat tunnistamisen ja valinnan. Monet tuntevat henkilöitä, joilla on johonkin tilanteeseen tarvittavaa eksperttiyttä ja ottavat heihin yhteyttä.

Näiden vaiheiden erottaminen toisistaan lupauksen suunnitella ja toteuttaa potentiaalisia järjestelmiä. Eksperttiyden hakeminen voidaan jakaa organisaatioissa kahteen vaiheeseen, erottaen toisistaan hakemisen tunnistamis- ja valitsemisvaiheet.

Eksperttiyden tunnistus

Eksperttiyden tunnistus on vaikea probleema. Mitä se on ja miten eksperttiyttä käytetään? Ihmiset kehittyvät ja muuttuvat ajan myötä. Myös ympäristö elää. Tiedon hakijalle edellämäinnittu on ensimmäinen probleema ratkaistavaksi, kun edetään siihen suuntaan, että saataisiin tarvittavaa informaatiota.

Eksperttiyden tunnistaminen on probleema, jossa selvitetään se, mitä tietoa (informaatiota) tai erityistaitoja muilla ihmisillä on.

Eksperttiyden valinta tarkoittaa tarkoituksenmukaisten taitojen omaavien henkilöiden valitsemista. Tästä myöhempänä lisää. Jos potentiaalisia henkilöitä on useampia, valitaan näistä yksi tai useampia. Teemme tämän erottelun analyttisesti niin, että pidämme mielessä sen, että tämä ei päde kaiken aikaa ja kaikille mukana oleville. Niiden ihmisten tunnistaminen, joilla on muille jaettavaa eksperttiyttä, on monien ongelmien ratkaisun kannalta ensimmäinen askel ratkaisun suuntaan. Mitä eksperttiys on ja kuinka sitä käytetään? Muuttuva ympäristö ja muuttuvat ihmiset tekevät tekevät eksperttiyden hallinnasta vaikeaa.

Joskus henkilöt tietävät muiden osaamisesta ja lähestyvät heitä. Paepcke ehdottaa tunnistamisen vaiheiden erottamista ja välittäjien analysointia. Näiden vaiheiden erottaminen antaa mahdollisuuden suunnitella ja toteuttaa potentiaalisia järjestelmiä. Vaiheiden erottaminen on analyttisesti arvokas ja osoittautuu hyödylliseksi jatkossa.

Jokapäiväinen eksperttiys: monien on vaikeaa ilmaista sitä, miten he tietävät jonkun omaavan jotain tiettyä eksperttiyttä. Monille 'kokemus' on tärkeä argumentti, kun he kuvaavat jonkun omaavan eksperttiyttä. Työntekijät oppivat väitellen, kenellä on mitään erityistaitoa tai tietoja. McDonald ja Ackerman [McDonald, Ackerman 1998] raportoivat kaksi kirjallisuudessa raportoimatonta mekanismia, joilla työntekijät tunnistavat organisaatioissa työovereidensa eksperttiyttä. Ensimmäinen on se, miten henkilöt käyttävät ihmisten tekemän työn tuloksia kerätäkseen informaatiota siitä, mitä kukakin tekee organisaatioissa. Toinen on henkilön erityinen rooli, joka auttaa löytämään oikean henkilön. Yksi tärkeä rooli on eksperttiyden portinvartijalla (expertise gatekeeper), joiden rooli on tiiviisti kytköksissä organisaatioon ja jotka palvelevat tiedon hakijoita tuomalla uutta relevanttia teknistä tietoa potentiaalisten tiedon hakijoiden ulottuville. Eksperttiyden portinvartijoiden roolista on havaittu erilaisia variaatioita, joita voidaan nimittää informaation välittäjiksi, jotka välittävät tietoa paikallisten dokumenttien välityksellä ja yhteyksien välittäjiksi, jotka tuntevat organisaation eri osista ihmisiä.

Eksperttiyden valinta

Eksperttiyden tunnistus ei vielä riitä ratkaisemaan probleemaa tai antamaan vastausta kysymyksiin. Seuraava vaihe onkin eksperttiyden valinta, jossa valitaan yksi tai useampi niistä sopivista henkilöistä, jotka on paikallistettu. Yksikään henkilö

ei todennäköisesti hallitse mitä tahansa erittäin monimutkaista järjestelmää. Vain yhden ekspertin löytävä järjestelmä ei ole kovin realistinen ja hyvä, oli kysymyksessä mikä tahansa eksperttiyttä vaativa asia. Usein hakija kuitenkin joutuu valitsemaan useista mahdollisista eksperteistä ne, joilla oletettavasti olisi juuri sopivaa eksperttiyttä. Monimutkaisen probleeman ollessa kyseessä hakija saattaa tehdä useita hakuja saadakseen apua riittävän monipuolisesti.

McDonald ja Ackerman löysivät tutkimuksessaan kolme yleistä eksperttiydenvalintamekanismia.

organisatorinen kriteeri

"the load on the source"

suorituskyky

Organisatorinen kriteeri

Organisatorisia kriteerejä voidaan kuvata peukalosäännöillä, jotka ovat 'paikallisena pitäminen'. Tällä tavoin pyritään pitämään ongelma niin lähellä sen syntypaikkaa kuin mahdollista, esimerkiksi omalla osastolla, jolloin auttajat tuntevat hyvin myös probleeman kontekstin. Toinen peukalosääntö on mennä suoraan sopivan henkilön luo, vaikka tämä olisi toisella osastollakin. Näin nopeutetaan ongelman ratkeamista (välttämällä mm. turhaa byrokratiaa). Kolmas peukalosääntö ovat edellämaintut eksperttiyden portinvartijat, joiden puoleen tulee kääntyä, mikäli apua ei muutoin löydy.

Ennen kohdistusmekanismin suunnittelua on hyvä tutustua olemassaoleviin mekanismeihin sekä tulevan mekanismin kontekstiin, sen sosiaalisiin, kognitiivisiin ja informatiivisiin taustoihin, kuten McDonald ja Ackerman edeman augment before constructing these systems. It is our goal to guide the construction of information systems that facilitate finding expertise within organizations. We believe that one can do so best with a field study of existing practice." [McDonald, Ackerman 1998]

Ohjelmistoyritysten eksperttiyden kohdistaminen

Avoimen koodin (Open Source Software, OSS) kehittäjäyhteisöissä tarjotaan yhteisöjen jäsenille vapaaehtoista ja ilmaista vertaistukea [Singh 2006]. Tuki perustuu tietämyksenhallintajärjestelmiin, jotka koostuvat mm. dokumenttien talletuspaikoista, eksperttitietokannoista, keskustelulistoista, sisältöspesifeistä hakujärjestelmistä, erilaisista suodatusteknologioista, joita käytetään yhteistyössä tehtävissä suodatusmetodeissa.

Teknisen tuen ilmoitustaulu on asynkroninen, ei käyttäjän paikallisessa koneessa (remote), useimmiten tekstipohjainen ja vapaaehtoisten tarjoama palvelu. Sillä on samankaltaisuuksia ja eroavuuksia OSS-kehitysprosessiin ja muuntuyppiseen tukeen verrattuna. Niillä kullakin on oma tutkimuskirjallisuutensa. Mielenkiintoista onkin se, kuinka käyttäjät tekevät yhteistyötä avun hakijoina yrittäen yhteisesti ratkaista ongelmiaan ja miten he jakavat löytämänsä ratkaisut muiden kanssa.

Auttamistaakan jakautuminen

Auttamistaakkaa voi jakaa vertaistuella. Auttajan (ekspertin) työkuormaa (päivittäinen ja auttajan kuorma lisänä) vaikuttaa auttajan valintaperusteisiin. Auttajan on voitava hoitaa auttaminen yleensä päivittäisen työn ohessa (lisänä). Tekninen tuki on institutionaalistunut mekanismi organisaatioissa. Miten työkuormaa mitataan? Miten työkuorma vaihtelee? 'Suljettu ovi' viestii kiireestä. Työkuormaa arvioidaan työkavereiden arvioiden perusteella. Arviot muuttuvat ajan kuluessa. Sama avun pyytäjä voi kysyä samaa asiaa useilta auttajilta saadakseen tehoa etsintäänsä. Avun pyytäjä ei halua jatkuvasti kuormittaa samaa eksperttiä.

Suorituskyky, teho

Tapa jakaa omaa eksperttilytään vaikuttaa siihen, miten apua pyydetään, samoin se, mitä eksperttilytää henkilöllä on. Ongelman ymmärtäminen ei aina ole helppoa: ongelma voi olla vaikea tai vaikea selittää, voi olla kulttuurisia ja kieliogelmia. Myös eksperttilyden määrä voi olla riittämätön. Jotkut auttajat ovat taitavampia (selkeämpiä) antamaan selkeitä, ymmärrettäviä ja ymmärrettäviä selityksiä kuin toiset. Asenteen ovat myös ratkaisevia. Kaikkia valintastrategioita ei voi automatisoida. Jos apua ei saa lähempää, haun tai järjestelmän laajentaminen voi olla välttämätöntä. Eksperttilyden tunnistaminen voi epäonnistua kolmella tavalla: yli-identifiointi (liikaa auttajia), liian vähän auttajia tai ei ollenkaan tai kukaan valituista ei ole ekspertti.

Kaupallisten ohjelmien puhelintuki

Kaupallisten ohjelmien puhelintuella ja OSS-käyttäjillä on samankaltaisia tarpeita. Puhelintuki koostuu sarjasta toimenpiteitä ja sen protokolla voi olla tiukasti määritelty. Pääasialliset erot näillä järjestelmillä liittyvät auttajien tyyppiin ja yhteydenpitovälineeseen, joskin puhelin muuttaa hieman yhteydenpidon luonnetta. Avun hakijalle puhelin on pienempi ponnistus ilmoitustaulutukeen verrattuna, koska tarpeita voidaan tarkistaa, kysely ja käsitysten korjailu on paljon helpompaa kuin tekstuaalisessa yhteydenpidossa.

Toisaalta hyvin muotoiltu tekstimuotoinen kysymys on helppo lukea ja hyvinmuotoiltu vastaus on helppoa hyödyntää uudelleen. Puhelinneuvonta keskittyy pääasiassa tiedon jakamiseen ja FAQ-tyyppiseen toimintaan. Puhelinneuvonnassa (help desk) apu kohdistuu 1-to-1-tyyppisesti suoraan neuvojalta avun hakijalle. Puhelinneuvonta voi olla ilmaista tai 'kaupallista', jolloin siitä peritään maksu. OSS:ssa help on vapaaehtoista ja ilmaista.

KMS (Knowledge Management System) eli teitämyksenhallintajärjestelmä, jota erimuotoisesti käytetään mm. help desk -toiminnoissa, koostuu dokumenttien talletuspaikoista (document repositories), eksperttitytökannoista (expert databases), keskustelulistoista (discussion lists), sisältöspesifeistä hakujärjestelmistä (context-specific retrieval systems), suodatusteknologioista (filtering technologies) ja yhteistyössä suoritettavista suodatusmetodeista (collaborative filtering methods).

CSCW:n (Computer-Supported Cooperative Work) ja yhteistyössä tapahtuvaan informaation hakuun liittyvä kirjallisuus kertovat lukuisista yrityksistä helpottaa informaation löytymistä sosiaalisten verkostojen avulla. Eksperttily on syvällistä tietämistä tai taitoa tehdä jotakin hyvin. Noudatan McDonalidin ja Ackermanin määritelmään eksperttilydestä työssäni. Määritelmän mukaan eksperttily on henkilöllä olevia tietoja ja taitoja. Eksperttily on alue ja se on erillinen eksperttilystä. Yksilöllä voi olla eri tasoista eksperttilytää erilaisista asioista. Eksperttily voi olla aiheisiin liittyvää tai proseduraalista ja se järjestetään ja arvotetaan sosiaalisilla ja institutionaalisilla mittareilla. Useilla järjestelmillä on ulkoisia eksperttilyden malleja. McDonald ja Ackerman kuvaavat keskikokoisten ohjelmistoyritysten eksperttilyden kohdistamista organisaation sisällä [McDonald, Ackerman 1998].

3.1.4 I-Help-järjestelmä

Yliopistoissa on kehitetty opiskelun tueksi tietokonepohjaisia vertaistukijärjestelmiä. Niissä opiskelijan on määriteltävä ja ilmaistava avun tarve jollain tavalla ja vastaus tarpeeseen täytyy kohdistaa käyttäjälle. Erilaisia match-mekanismeja on paljon ja tässä kuvaan Saskatchewanin yliopistossa Kanadassa kehitettyä I-Help-järjestelmää (Intelligent Helpdesk) ja sen kohdistusmekanismeja.

Järjestelmän avulla opiskelijat voivat hakea vertaistukea opiskelussa ilmenneisiin kysymyksiin. Greer et al. [Greer et al. 2001] kuvaavat I-Help-järjestelmää. Henkilöitä mallinnetaan agenttien ja tietokantaan varastoidun tiedon avulla. Henkilö voi olla avun hakija, avun antaja tai molemmissa rooleissa. Kohdistusmekanismi, joka etsii sopivia kandidaatteja auttajiksi avun hakijalle, etsii heitä järjestelmässä kirjautuneena olevista henkilöistä ja valitsee viisi parhaiten sopivaa.

Välinta perustuu mahdollisten auttajien ominaisuuksiin, joita ovat osaamisen taso, auttavaisuus, vastaamisnopeus, kognitiivinen tyyli ja virtuaalivaluutan määrä. Kun henkilöitä edustavat agentit pääsevät sopimukseen, muodostuu kahden henkilön välille keskustelu ja muut kandidaatit vapautetaan.

Auttamistapahtumista kerätään automaattisesti transaktiotietoa ja tietoja kumppanista keskustelun päätyttyä. I-Help-järjestelmää voidaan käyttää synkronisesti tai asynkronisesti siten, että yleisille keskustelupalstoille voi lähettää kysymyksiä, joihin joku vastaa myöhemmin tai edellä kuvattuun tapaan kahdenvälisessä (1-to-1) keskustelussa, synkronisesti.

Järjestelmän kuvaus

Järjestelmä on toteutettu Javalla, moniagenttiarkkitehtuurina. Deters [Deters 2000] toteaa agenteista ja vertaisavusta, että I-Helpissä käytetään henkilökohtaisia agentteja, jotka edustavat käyttäjiä ja opiskelijoita. Agentit ylläpitävät käyttäjien malleja, jotka sisältävät informaatiota käyttäjien tavoitteista (avunpyynnöt ja tämänhetkiset tavoitteet), tietoresurseja, kompetensseja ja tietoja käyttäjienvälisistä suhteista. Käyttäjät kommunikoivat agenttiansa kanssa päivittääkseen mallejaan ja asettaakseen tavoitteitaan. Agentit kommunikoivat keskenään ja hakuagenttien kanssa, etsiäkseen sopivia auttajia, avunpyynnön mukaan ja neuvottelevat sitten avun hinnasta.

Käyttäjää edustaa ohjelmistossa oleva henkilökohtainen agentti. Elektronista lähdettä vastaa sovellusagentti. Sovellusagentti huolehtii elektronisista lähteistä, samoin ohjelmistosta. Agenteilla yhteinen ontologia ja kieli. Agentti ylläpitää resursseistaan tietokantaa, henkilökohtainen agentti käyttäjän tietämyksestä ja ominaisuuksista sekä sovellusagentti sovelluksen (opetus)materiaalista tai muista tiedoista. Agentit käyvät kauppaa resursseilla (lähteillä), kun ne tarvitsevat lähteitä, joita niillä ei ole. Agenttien välille muodostuu suhteita, jotka heijastavat käyttäjien välisiä suhteita.

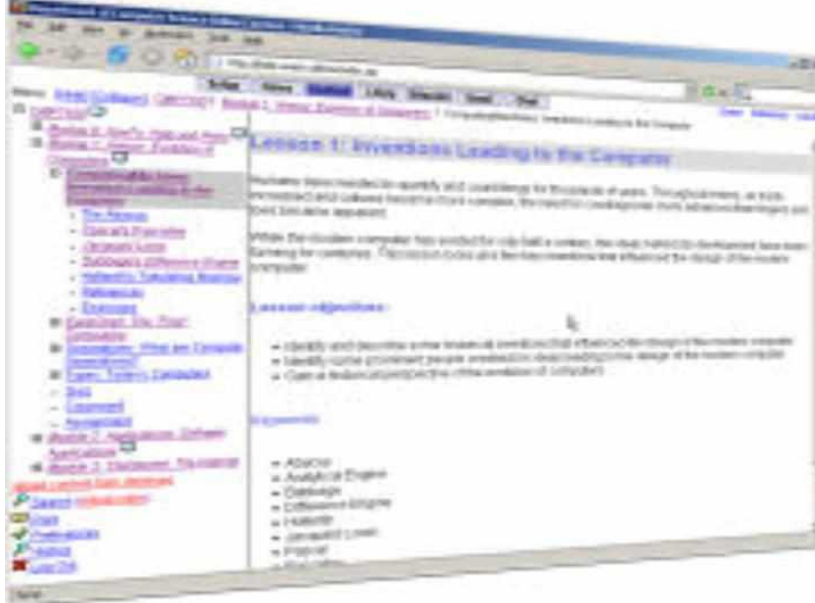
Kun käyttäjä pyytää apua, käyttäjän henkilökohtainen agentti lähettää pyynnön kohdistajalle (match maker), joka etsii sopivan toisen agentin, joko henkilökohtaisen tai sovellusagentin, käyttäen agenttien mallit sisältävää tietokantaa. Tietokannassa on esimerkiksi käyttäjän pätevyys kyseisessä aiheessa ja tiedot elektronisen materiaalin kattavuudesta aiheeseen liittyen. Kun sopiva agentti on löytynyt, agenttien neuvottelu alkaa.

Käyttäjällä (opiskelijalla) on mahdollisuus pyytää, saada apua järjestelmän tuntemilta henkilöiltä ja antaa apua muille järjestelmän tuntemille henkilöille. Vertaisapu opiskelijoiden välillä säästää opettajien resursseja ja tasaa auttamiskuormaa järjestelmän piirissä toimivien henkilöiden välillä. Katso kuormituksen jakautumisesta myös [Singh 2006] jäljempänä.

I-Help-järjestelmässä viestien lähetys avun hakijan ja auttajan välillä tapahtuu joko synkronisesti tai asynkronisesti. Synkroninen yhteydenpito tapahtuu kahden tai useamman osallistujan keskusteluna keskustelupalstalla (foorumi, chat). Asynkroninen yhteys toimii sähköpostin ja keskusteluryhmien avulla.

Molemmissa tavoissa on käytössä yhteensovituspalvelu (match making), joka etsii sopivan vertaisauttajan tietokannassa olevien käyttäjien mallien avulla. Yhteensovittajien käyttö on merkittävä lisä aiempiin avunsaantivälineisiin verrattuna. Aiempia välineitä ovat olleet materiaalin tarjoaminen verkkosivuilla, sähköposti, keskustelupalstat ja FAQ-tietokannat. Yhteensovittajat parantavat opiskelijan kannalta katsottuna hajautetun tuen saantia.

Kuvassa 8 on esitetty I-Helpin toiminnallinen rakenne. Ylhäällä oleva opiskelija pyytää apua järjestelmästä ja järjestelmä hyväksyy avunpyynnön. Pyyntö yhdistetään sopiviin käsitteisiin ja pyyntöön liitetään opiskelijan malli. Seuraavaksi järjestelmä valitsee sopivan apuresurssin. Opiskelijoista kerätään myös tietoja opiskelijan malleihin tietokantaan.



Kuva 8. I-Helpin arkkitehtuuri toiminnallisuuden kannalta esitettynä [Greer et al. 1998].

Miten avun haku tapahtuu? I-Help:ssä tarpeen ja avun yhteensovitus (matchmaking, avun kohdistus) tapahtuu siten, että käyttäjän pyytessä apua, henkilökohtainen agentti lähettää avunpyynnön yhteensovittajalle (koka on myös agentti), joka etsii agenttien mallit sisältävän tietokannan avulla sopivan vertaisauttajan agentin ja ottaa siihen yhteyden. Valintaan vaikuttaa mm. auttajan pätevyysparametri. Sopivan auttajan löytyessä agentit aloittavat neuvottelun. Jos agenttien välille syntyy sopimus auttamisesta, saa avun pyytäjä tästä ilmoituksen.

3 Mallinnus ja agentit

Agentit.

3.1 Henkilökohtaiset agentit

Henkilökohtaiset agentit ylläpitävät mm. seuraavia tietoja:

- lista käyttäjän ystävistä sekä vihollisista
- asetukset, joiden mukaan agentti neuvottelee käyttäjän puolesta
- eri lähteiden tärkeys käyttäjälle
- rahanhimo
- käyttäjän egoismi / altruismi

Käyttäjä asettaa kyseiset ominaisuudet ja ne määrittävät tapaa, jolla käyttäjä näkyy muille käyttäjille. Neuvotteluissa henkilökohtainen agentti edustaa käyttäjää. Agentit pyrkivät optimoimaan toimintaansa ja pyrkivät ennustamaan vastapuolen käyttäytymistä.

Nämä ominaisuudet käyttäjä säätää itse ja ne heijastavat tapaa, jolla käyttäjä ”näky” toisille käyttäjille. Neuvottelussa toisien agenttien kanssa henkilökohtainen agentti toimii ikään kuin käyttäjän puolesta. Agentit yrittävät optimoida toimintaansa ja ennustaa ”vastapuolen” käyttäytymistä. Sen vuoksi, ne tekevät malleja vastapuolen luonteesta ja

prioriteeteista. Neuvottelun aikana agentit ottavat huomioon aikaisemmin muodostuneet suhteet toisien käyttäjien kanssa (tarjota alennushintaa ystäville tai erikoisen kallista hintaa vihollisille...). Kun neuvottelun tuloksena syntynyt onnistunut avunvälitys on ohi, agentit tarjoavat mahdollisuutta lisätä uusi suhde malleihinsa, jolloin "ystävien" määrä lisääntyy yhdellä. Lisäksi henkilökohtaiset agentit keräävät viitteitä niihin agentteihin, joilla on informaatiota käyttäjästä, esim. diagnostiset agentit jotka ovat muodostaneet malleja käyttäjän tietämyksestä eri aloilla.

4.2 sovellus- ja diagnostiset agentit

Sovellusagentit vastaavat uutisryhmien säikeitä, web-sivuja, hakukoneita jne. Ne rakentavat oman käyttäjäprofiilin, joka kuvaa sovelluksen relevantteja käyttötapoja, perustuen kanssakäymiseen käyttäjien kanssa. Lisäksi ne käyttävät "perinteisiä" käyttäjän mallinnustekniikoita. Esim. agentti joka vastaa web-pohjaista C++ -kurssia varastoi dataa opiskelijoiden edistymisestä kurssilla liittyen luennolla opetettuihin asioihin. Se myös varastoi käyttäjien asetuksia web-kurssin käyttöliittymästä.

Diagnostiset agentit ovat sovellusagenttien alaryhmä, jotka vastaavat web-pohjaisia testituloksia, kyselykaavakkeita jne. Ne luovat käyttäjämalleja tietynlaiseen toimintaan / tietämykseen liittyen ja niillä on tietty rakenne. Esim. diagnostinen agentti "M" tarkkailee käyttäjän toimia (uutisryhmien selailua, lukemista ja viesteihin vastaamista) ja päivittää "käyttäjän innokkuus"-luokitusta (yksi osa käyttäjän sosiaalisista luonteenpiirteistä). I-Help-järjestelmässä on monia agenteja, jotka mallintavat käyttäjän tietämystä jossakin tietyssä aiheessa ja uusia agenteja voidaan myös lisätä.

4.3 Yhteensovittaja-agentit (matchmaker agents)

Yhteensovittaja-agentit pitävät yllä käyttäjien malleja eli profiileja. Nämä agentit ovat erikoistuneet käyttäjien tiettyihin malleihin, kuten jonkin asiatiedon osaamiseen. Sovittajat luovat yhteyksiä käyttäjien välille esimerkiksi sen perusteella, kuka osaa parhaiten auttaa jossakin ongelmassa tai jolla on samanlainen ajattelutapa kuin avun pyytäjällä (kognitiot). Yhteensovittaja-agentit pyytävät henkilökohtaisilta ja diagnostisilta agenteilta käyttäjäprofiileja, luodakseen malleja yhteensovittavista käyttäjistä. Auttajan tasoa määrittellen apusessioiden yhteydessä kerätyn tiedon avulla.

Henkilökohtaiset agentit infoirmoivat käyttäjiä vain silloin, kun agentit ovat päässeet sopiukseen. Käyttäjät voivat hyväksyä tai hylätä tarjouksen, mikä päätös vaikuttaa kertyvään valuuttaan. Jos käyttäjä toistuvasti hylkää agentin tarjouksen, käyttäjälle ei kerry valuuttaa. Sovittajat pitävät kirjaa myös henkilökohtaisista agenteista, jotka rikkovat sopimuksiaan. Sovittajat keskittävät informaatiota (miten?), jotta lista sopivista auttajista voitaisiin aluksi muodostaa. Sovittajat eivät kuitenkaan rakenna globaaleja malleja käyttäjistä.

I-Help-järjestelmässä tietämys käyttäjistä on hajautettu eri agenttien välille (henkilökohtaisten ja sovellusagenttien välille). Tieto on sirpaleina, keskitettyä käyttäjämallia ei ole, koska esim.

tällaisessa järjestelmässä eri agenteilla voi olla erilaisia (ristiriitaisia) näkökantoja

teknisesti keskitetty järjestelmä on hankala toteuttaa suorituskykyongelmien takia

harjoitustehtävien tekijöitä yhdistellyt matchmaker on erilainen kuin keskusteluryhmien säikeitä (kysymyksiä) yhdistellyt matchmaker

Informaationhaussa elektroniset lähteet tai palvelut sovitetaan informaationtarvitsijan kanssa. Jokainen lähde ilmaistaan järjestelmässä joko yksitellen tai sovellusagentti ryhmittelee ne aiheen mukaan. Sovellusagentti ylläpitää sovelluksensa lähteiden luettelon (mallin) ajan tasalla. Esim. keskusteluryhmän sovellusagentti ylläpitää hallussaolevien aiheiden listaa. Meklariagentti (matchmaker-agentti) ylläpitää sovellusagenttien listaa, joka on järjestetty niiden aiheiden mukaan. On tärkeää, että sovellusagenttien tekemät listat noudattavat samaa ontologiaa, jotta meklariagentti voi tehdä niistä hakemiston.

Käytännössä järjestelmän administraattori, joka on yleensä opettaja, luo kurssilla käsiteltävien aiheiden taksonomian joka palvelee ontologiana. Opiskelijoiden avunpyynnöt täytyy liittää määriteltyihin aiheisiin. Tämän järjestelmän tarkka noudattaminen aina ei ole kuitenkaan pakollista sillä muita tekijöitä kuin aiheita (jonkun tietyn opiskelijan lähettämät viestit, tietyn ajanjakson aikana lähetetyt viestit tai tiettyyn viestiin tulleet vastaukset) voidaan käyttää haun perustana. Järjestelmän redundanssi varmistaa, että vaikka agentit eivät tarkasti seuraa tiettyä taksonomiaa, sopiva lähde löytyy, kun edes agenttien osajoukko tekee niin.

I-Helpissä on useita meklariagentteja (matchmakereitä), jotka etsivät valmiita, auttamishaluisia ja -kykyisiä vertaisauttajia. Kukin agentti etsii oman kriteerinsä mukaan (kukin eri käyttötarkoitusta varten). Seuraavassa esitellään muutamia näistä:

apuaiheessa pätevimmat käyttäjät

ne käyttäjät, jotka ovat saatavilla (on-line-tilassa)

tietyn hyvän luonteenpiirteen omaavat käyttäjät (innokkuus, auttavaisuus, class ranking)

ne käyttäjät, joilla on samanlainen oppimistapa kuin avunpyytäjällä

avunpyytäjän ystävät

sama horoskooppimerkki

Kukin meklariagentti keskittyy tiettyihin ominaisuuksiin ja laatii niiden mukaan ranking-listan. Tyypillisesti meklariagentit toimivat yhteistyössä ja yhdistävät tuloksensa tuottaakseen listan, jonka käyttäjät on arvosteltu joidenkin ominaisuuksien yhdistelmän mukaan. Esim. henkilökohtaiset agentit on ohjelmoitu kutsumaan ”pätevyys”- ja ”saatavuus”- matchmakereitä. Riippuen käyttäjän asetuksista (siitä kuinka käyttäjä on ohjeistanut henkilökohtaisen agenttinsa) kutsutaan joko ajattelutavan, ystävien tai horoskooppimerkin matchmakereitä tai näiden yhdistelmää.

5 Käyttäjän mallinnus

Keskitettyä käyttäjämallia ei ole.

6 Resurssit

- resursseista ylläpidetään malleja

7 Tunnistus ja valinta

Resurssien paikallistaminen

Informaation haku on erityinen resurssienkohdistusvaihe, jossa elektronisia resursseja tai palveluja kohdistetaan tiettyihin käyttäjän tarpeisiin. I-Helpissä informaation välittäjät ovat agentteja, jotka etsivät verkkosivuja tai keskustelupalstoilla olevia ja käyttäjän pyyntöön nähden relevantteja postituksia. Kaikki informaatioreсурssit esitetään joko yksin tai ryhmänä sovellusagentin toimesta. Sovellusagentit ylläpitävät malleja niistä resursseista, joita ne esittävät. Esimerkiksi keskustelupalstan (foorumin) keskustelusäikeessä pitää yllä luetteloa relevanteista malleista, jolloin resurssit saavat resurssi-indeksin malleista, joita agentit ovat esittäneet. Kun käyttäjä tai henkilökohtainen agentti tekee informaatiopyynnön, välittäjäagentti paikallistaa agentit, jotka edustavat kaikkein relevanteimpia resursseja [Vassileva et al. 2003].

On toivottavaa, että sovellusagenttien ylläpitämät resurssien mallit noudattavat samaa ontologiaa, jotta välittäjä voi ylläpitää pysyvää ja yhtenäistä indeksiä. I-Helpissä keskustelualueen järjestelmänvalvoja, joka on tavallisesti luokan ohjaaja, tarjoaa pääkohtaluokittelun palvelemaan ontologiana. Käyttäjien informaatiokyselyjen tulee koskea jotain näistä otsikoista. Kuitenkin yleisesti ottaen tiukka kytkeytyminen taksonomiaan ei ole välttämätöntä. Muut tekijät sitäpaitsi, että otsikoita voidaan käyttää hauissa (toisin sanoen tiettyjen henkilöiden postituksia tai tietyn välein lähetettyjä postituksia tai vastauksia tiettyihin posteihin jne.). Järjestelmän runsas käyttö varmistaa, että vaikka agentit eivät tiukasti noudatakaan tiettyä taksonomiaa, relevantti resurssi voidaan silti löytää, jos ainakin osa agenteista noudattaa luokitusta [Vassileva et al. 2003].

I-Help-järjestelmän kohdistusalgorithmi on seuraavanlainen. Voidakseen nähdä, onko lukuoikeus resurssiin voimassa, käyttäjän agentin on suoritettava seuraavanlainen algoritmi

määrittele pääsy epätodeksi (FALSE)

käy sitten jokaiselle dokumentin käyttöoikeussäännölle läpi seuraava alialgoritmi

jos löytyy yksikin käyttöoikeus, aseta pääsy todeksi (TRUE)

jos löytyy yksikin ristiriita kyselyn URL:n kanssa, aseta pääsy epätodeksi (FALSE)

jos pääsy on nyt tosi, hyväksy pääsy ja poistu algoritmista

muutoin, estä pääsy resurssiin

8 Talous

I-Helpin talous (I-Help economy)

Agentit neuvottelevat avun hinnasta. Valuuttana käytetään ICU-virtuaalirahayksikköä (I-Help Currency Unit). Avun saajat maksavat avusta ja auttajat saavat palkkion avustaan. Taloudenpidon tarkoituksena on luoda dynaaminen markkinatalous, jossa agenttien huolehtiman avun tarjonnan ja kysynnän välillä vallitsee tasapaino. Osaavia käyttäjiä ei kuormiteta liikaa ja auttamiseen motivoidaan kaikkia. Talouden säätelyllä estetään myös saalistajakäyttäjien toiminta. Virtuaaliraha lisää opiskelijoiden motivaatiota; luentokurssien bonusarvosanat vastaavat virtuaalirahaa; myös maine toimii motivaattorina.

9 Esimerkki toiminnasta

Esimerkki järjestelmän toiminnasta

Järjestelmää testattiin kaksi vuotta yli 1000 yliopisto-opiskelijan kanssa Saskatchewanin yliopistossa Canadassa, ja tulokset esiteltiin artikkelissa [7]. Tästä artikkelista otamme esiin esimerkin järjestelmän toiminnasta.

Oletetaan, että opiskelijalla, joka tekee ohjelmoinnin harjoitustehtävää, on mielessään kysymys. Hän delegoi avunpyynnön henkilökohtaiselle agentilleen. Tämä yrittää löytää toisen agentin, joka voi olla joko sovellusagentti tai toinen henkilökohtainen agentti, joka tarjoaa auttavaa informaatiota (lähde). Tämä lähde voi olla elektroninen, kuten luennoijan tai toisen opiskelijan tekemä web-sivu (tätä vastaa sovellusagentti) tai I-Helpin keskusteluryhmän keskustelu tai säie (jota vastaa keskusteluryhmän agentti). Lähde voi olla myös toinen opiskelija, joka on samaan aikaan verkossa ja jolla on pätevyys auttaa ongelmassa (häntä vastaa toinen henkilökohtainen agentti).

Eri agenteilla on yhteinen koodikieli, jolla lähteet(resurssit) indeksoidaan. Tämä koodikieli perustuu luennoitavan asian aiheeseen, tehtäviin jne. Koodikieli on yleensä luennoitsijan luoma ja se perustuu kurssin aiheiden luonnokseen ja sitä voi laajentaa myöhemmin, mikäli tarvetta on. Avuntarjoajan etsii ns. yhteensovittaja, matchmaker. Nämä

Yhteensovittajat ylläpitävät tietokantaa eri agenttien tietämyksestä sekä käyttäjien ja sovellusten muista ominaisuuksista.

Jos sopivaa avuntarjoajaa ei löydy, matchmaker luo järjestetyn listan käyttäjistä, jotka ovat sillä hetkellä verkossa ja joilla on tietoa aiheesta. Tämä lista lähetetään avuntarvitsijan agentille ja agentti aloittaa neuvottelun listan ensimmäisen agentin kanssa avuntarjoamisen hinnasta ICU-virtuaalirahayksikköinä. Kun neuvottelu on onnistunut, avuntarjoajan agentti ilmoittaa tästä kyseiselle käyttäjälle kysyen haluaako hän auttaa. Jos vastaus on ei, silloin matchmaker aloittaa neuvottelun listan seuraavan agentin kanssa. Jos taas käyttäjä suostuu, avataan chat-linja näiden käyttäjien välille ja apu välitetään sitä tarvitsevalle. Kun linja katkaistaan, avautuu ikkunassa toisen osapuolen arvioimiseen käytettävä lomake. Matchmaker käyttää kerättävää informaatiota käyttäjien profiilien päivittämiseen agenttitietokannassa.

Kuva 2. Henkilökohtainen agentti (kuvassa oikealla) on löytänyt mielenkiintoisen keskusteluryhmän viestin liittyen käyttäjän avunpyyntöön. Julkisen keskustelufoorumin käyttöliittymä koostuu neljästä ikkunasta, joissa voidaan katsoa tietyn aiheen viestejä, vastata viesteihin, poistaa viestejä sekä löytää nopeasti huomionarvoiset viestit.

3.1.5 Internetin hakukoneiden kohdistusmekanismit

Tyypillisellä Internet-selaimen välityksellä toimivalla hakukoneella on indeksi, jossa on tiedot hakukoneella löytyvistä sivuista. Indeksini muodostetaan tietyillä kriteereillä. Käyttäjän tehdessä hakua hän kirjoittaa hakutermiksi hakuohjelmalle sanan, osittaisen sanan tai useita sanoja ja käyttää muitakin hakukoneen sääntöjen mukaisia hakutermin asettamistapoja. Hakuohjelma käy hakutermiä avulla läpi indeksini ja hakukoneesta riippuen palauttaa koneen sääntöjen mukaan järjestetyn tuloksen. Tässä tapahtuu (avun) haku ja löydettyjen resurssien palauttaminen tietyssä järjestyksessä hakijalle (avun kohdistus).

kuva!

Mikäli hakijalla on tiedossa suora osoite johonkin sopivaa tietoa sisältävään sivustoon, voi tiedon hakija pelkällä selauksella löytää hakemansa. Kuitenkin on usein tarpeen käyttää hakukoneiden palveluja. Kappaleessa 3.1.2 viitattiin jo tekstimuotoisten hakujen tyypeihin, joita ovat avainsanahaku, boolean haku, esimerkkihaku, fraasihaku, sanarunkohaku ja sumea haku.

Avainsanahaku hakee käyttäjän hakusanat sisältävät dokumentit ja yrittää järjestää tuloksen dokumenttien laadun perusteella käyttäen esimerkiksi tekniikkaa, jossa lasketaan hakusanan esiintymiskerrat tekstissä tärkeyden perusteena.

Boolean hakua (boolean search) käytetään joillakin hakukoneilla, jolloin pystytään tekemään mutkikkaampia hakuja käyttämällä loogisia operaattoreita AND (ja), OR (tai) ja NOT (ei). Joissakin hakukoneissa ei tarvitse käyttää AND-operaatiota vaan se on oletuksena ja hakutuloksessa vaaditaan kaikkien hakusanojen esiintymistä. Saman haun tekeminen joissain muissa hakukoneissa vaatii +-merkin lisäämistä jokaisen hakusanan eteen. NOT-operaatiota vastaa joissain hakukoneissa - (miinus, väliviiva) -merkki.

Esimerkkihaku (query by example) toimii siten, että käyttäjä antaa dokumentin ja hakukone hakee samantyyppisiä dokumentteja. Google-hakukoneessa tämä onnistuu related-avainsanan avulla.

Fraasihaussa (phrase search) hakusana sijoitetaan lainausmerkkien sisään, jolloin hakukone hakee koko merkkijonoa eli fraasia. Hakufraasi voisi olla esimerkiksi "hirsirakentajan perustaidot".

Sanarunkohassa (stemming) haetaan sanoja pelkästään niiden alkuosan perusteella, esimerkiksi sanarungolla pilvi löytyisivät sanat pilvipouta, pilvimuodostelma jne. Altavista-hakukoneessa sanarunkohaku määrittellään asteriski (*) -merkillä, esimerkiksi hakutermillä kaiku* löytää sanat kaikuluotain, kaikunuora jne.

Sumea haku (fuzzy search) tarkoittaa tietojen löytymistä, vaikka hakusanaa ei löytyisikään kirjoitusvirheen takia (katso mm. [Ekonoja, Lahtonen, Mäntylä 2003]).

Tämän hetken yksi yleisimmin käytetyistä hakukoneista on Google, joka on samalla sitä kehittävä yrityksen nimi. Google-hakukoneen haut perustuvat yrityksen kehittämään PageRank-tekniikkaan. Tässä tekniikassa ohjelmisto tekee samanaikaisia laskelmia nopeasti ja analysoi linkkirakennetta ja päättelee sen perusteella, mitkä sivut ovat tärkeimpiä. Tämän jälkeen ohjelmisto suorittaa hypertekstin täsmäytysanalyysin löytääkseen haun kannalta olennaiset sivut.

Googlen sivustossa [Google 2009] on esitelty PageRank-tekniikkaa seuraavasti:

"PageRank-teknologia: PageRank arvioi Web-sivujen merkitystä objektiivisen mittaustavan avulla, jossa ratkaistaan yli 500 miljoonaa eri muuttujaa ja 2 miljoonaa termiä sisältävä kaava. Suorien linkkien laskemisen sijaan PageRank tulkitsee linkin sivusta A sivulle B olevan yhden äänen. Sitten PageRank arvioi sivun merkityksen laskemalla sen saamat äänet.

PageRank ottaa huomioon myös jokaisen sivun tärkeyden laskemalla sivujen ääniä. Joidenkin sivujen äänillä on suurempi merkitys, ja niistä tulevien linkkien avulla linkitetty sivu saa suuremman arvon. Tärkeät sivut saavat korkeamman PageRank-arvon, ja ne näkyvät hakutulosten kärjessä. Googlen tekniikka hyödyntää internetin kollektiivista älyä määritellesään sivun merkityksen. Ihmiset eivät voi vaikuttaa tai manipuloida hakutulosten sijoituksiin, ja juuri siksi käyttäjät luottavat Googleen objektiivisena tietolähteenä, jota maksettu sijoittelu ei ole pilannut."

Hypertekstin täsmäysanalyysi tarkoittaa sitä, että hakukone analysoi sivujen sisältöä tarkasti, mm. kirjasimet ja sanojen sijainnin sivulla sekä sivua lähellä olevien sivujen sisällön. Näin esimerkiksi meta-tiedoilla ei voi vaikuttaa suoraan sivun tärkeyteen haun kannalta.

Hakualgoritmi toimii seuraavasti:

web-palvelin lähettää pyynnön indeksipalvelimille

hakupyynnö siirtyy asiakirjapalvelimiin, jotka noutavat tallennetut asiakirjat; kaikista hakutuloksista luodaan näytekappaleet

hakutulokset palautetaan käyttäjälle sekunnin sadasosassa

Katso [Google 2009].

3.2 Apu-mekanismi

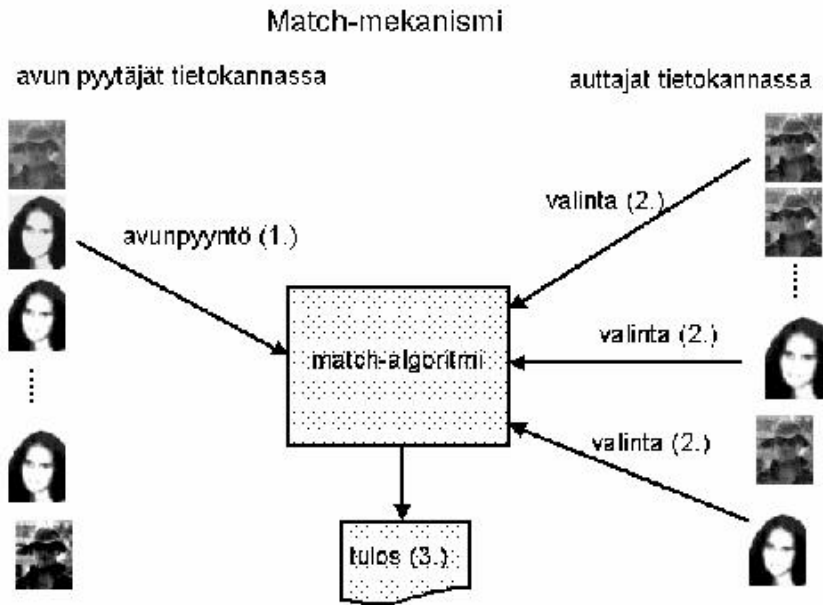
Olen kehittänyt tietokantapohjaisen sovelluksen apuresurssien kohdistamiseksi apua tarvitseville. Apu-mekanismi kohdistaa hakijalle tietoteknistä apua, jonka tiedot ovat mekanismin saatavilla sen tietokannassa. Hakija määrittelee avun tarpeensa Internet-selaimen välityksellä toimivalla verkkosovelluksella ja mekanismi etsii mahdollisimman hyvin tarpeet täyttäviä auttajakandidaatteja (henkilöitä) ja esittää kandidaatit listana hakijalle. Hakija päättää jatkotoimista valitsemalla listasta auttajan, johon ottaa yhteyttä.

Mekanismin ytimenä on Match-algoritmi, joka tekee auttajien valinnan hakusanojen perusteella siten, että eniten osumapisteitä saaneet kandidaatit valitaan hakijalle esitettävään listaan.

Match-mekanismi käsittää henkilöitä ja tietokantapohjaisen verkkosovelluksen, jonka ytimenä on resurssien tunnistuksen tekevä match-algoritmi. Henkilöillä on roolina avun hakijan rooli, auttajan rooli tai molemmat. Mekanismi yhdistää toisiinsa avun hakijan ja auttajat siten, että avun hakijalle tarjotaan aina vähintään yhden auttajan tiedot. Haun tulos tosin lasketaan onnistuneeksi vasta, kun kolme hakusanaa täsmää tarjotun auttajan tietoihin. Kuvassa 1 on esitetty match-mekanismin rakenne. Ensin hakija tekee avunpyynnön (1), jonka jälkeen match-algoritmi hakee auttajakandidaatit (2) ja tulostaa auttajaluettelon (3). Tämä on auttajaresurssien tunnistusvaihe. Kohdistuksen valintavaiheessa hakija valitsee tuloslueettelosta auttajan ja ottaa tähän yhteyttä. [Tangmunarunkit et al. 2003] ja [McDonald, Ackerman 1998] ovat mm. kuvanneet kohdistuksen kaksivaiheisuutta.

Avun hakija asettaa 1 - 4 hakutekijää ja määrittelee niillä avun, jota tarvitsee. Vain ensimmäinen hakutekijä on

pakollinen, mutta paremman hakutuloksen saa, kun hakutekijöitä on kolmesta neljään. Kun hakutekijät on asetettu, match-algoritmi tekee auttajien valinnan tietokannasta.



Kuva 10. Match-mekanismi.

3.2.1 Apu-mekanismin konteksti

Organisaatio muodostuu tässä työssä oppilaitoksista, varsinkin pienistä, yhden tai muutaman ihmisen yrityksistä, julkishallinnon yksiköistä ja yksityisistä henkilöistä, jotka käyttävät tietotekniikkaa hallinnossa, tuotannossa, myynnissä, opetuksessa ja viestinnässä. Yksiköiden henkilöt ovat tekemisissä toistensa kanssa yhteistyön merkeissä monella tavalla. Organisaatiota voisi kutsua virtuaaliseksi siinä mielessä, että se koostuu hallinnollisesti erillisistä yksiköistä, jotka saattavat olla maantieteellisesti hajallaan, vaikkakin pääasiassa samalla talousalueella. Organisaation henkilöt ovat verkostoituneet toistensa kanssa monitahoisesti mm. kehittämään opetusta, tuotantoa ja muuta organisaation toimintaa työtehtäviensä puitteissa.

Oppilaitoksissa tietotekniikkaa käyttävät henkilökunta ja opettajat. Henkilökunta muodostuu mm. toimiston väestä, vahtimestarista, rehtorista ja vararehtorista. Opettajat ovat lehtoreita tai tuntiopettajia. Tyypillistä tietotekniikan käyttöä on toimisto-ohjelmien käyttö hallinnon tehtävissä ja viestintä tavallisimmin sähköpostilla. Tehtäviä ovat esimerkiksi oppilasrekisterin ja muiden rekistereiden ylläpito, asiakirjojen laadinta, lukujärjestyksen laadinta ja Internet- ja Intranet-sivustojen päivittäminen, oppimateriaalin laadinta ja tiedon haku.

Yritysten ja julkishallinnon väki koostuu tietotekniikan suunnittelijoista ja hallinnon henkilöistä. Heille tyypillisiä tietoteknisiä tehtäviä ovat asiakirjojen laadinta, julkaisujen tuottaminen paperimuotoisena ja Internet-käyttöön, rekistereiden ylläpito ja erilaisten kehitysympäristöjen käyttö. Yksityiset henkilöt käyttävät tietotekniikkaa huviksi ja hyödykseen ja paljolti toimisto-ohjelmien ja Internetin erilaisten palvelujen muodossa.

Laitteiston osalta ympäristö organisaatiossa muodostuu pääosin mikrotietokoneista, niitä yhdistävistä lähiverkoista, Internetistä, palvelimista ja erilaisista oheislaitteista. Ohjelmistona on mm. käyttöjärjestelmiä, toimisto-ohjelmia, tietoliikenneohjelmistoa, tietokantoja, ohjelmointikieliä, suunnitteluohjelmia ja monia hallinnon ohjelmistoja.

Tärkeänä yhdistävänä tekijänä organisaation yksilöiden ja henkilöiden välillä on tietotekniikan käyttö työtehtävissä ja tähän käyttöön liittyvät kysymykset, ongelmat ja tarpeet. Tietoteknistä osaamista on laajasti eri yksiköissä ja avun kohdistamisen tehostaminen yksiköiden ja henkilöiden välillä avun tarpeen ilmaantuessa tehostaisi toimintaa ja edistäisi

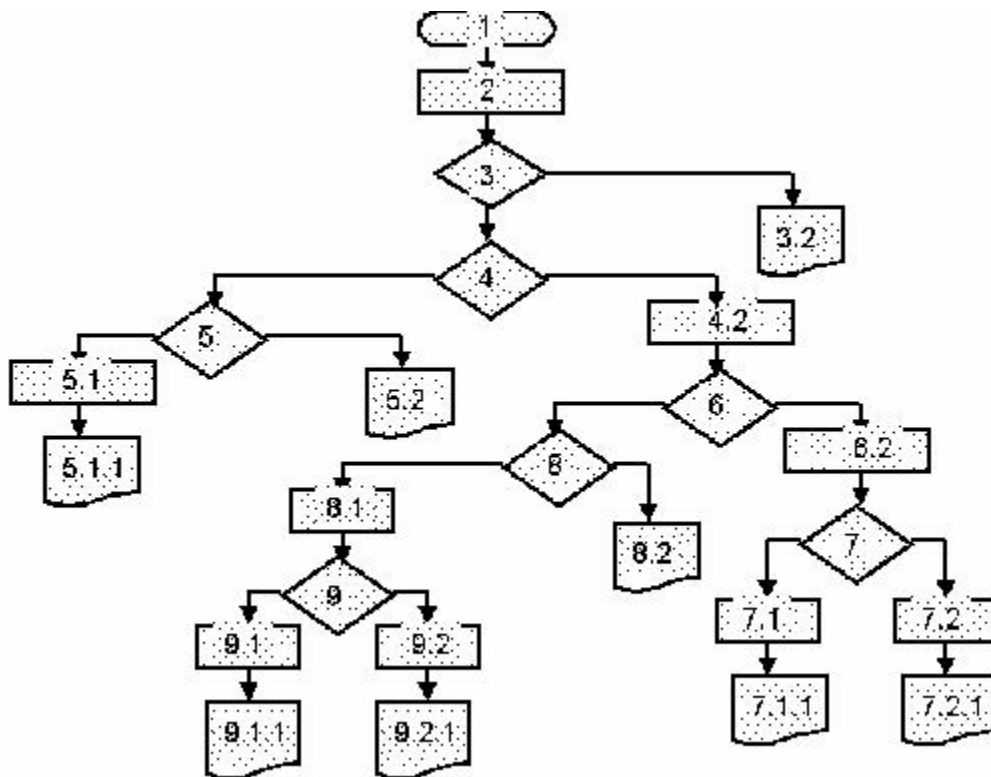
Oppimista työtehtävissä.

Usein tulee tilanteita, joissa henkilö törmää tietotekniseen ongelmaan tai tilanteeseen, jota ei osaa itse ratkaista ja jota myöskään välittömästi saapuvilla olevista henkilöistä kukaan ei osaa ratkaista. Tällöin avuksi tarvittaisiin mekanismi, jonka välityksellä saisi yhteyden henkilöön tai henkilöihin, joilta saa apua. Tässä työssä kuvaan mekanisme, jota käytetään kohdistamaan apu sen tarvitsijalle siten, että avun tarvitsija voisi ottaa yhteyden henkilöön, joka osaa auttaa syntyneessä avun tarpeessa. Organisaation kaikki henkilöt kuuluvat apumeکانismiin avun hakijoina, auttajina tai molempina.

Hakusanaluokat nousevat muutoin oppilaitosympäristöstä, mutta luokka yksi perustuu ACM CSS98 -luokitukseen ja noudattaa sen ylintä tasoa. Muut luokat ja niiden avainsanat olen kerännyt oppilaitosympäristöstä Riihimäen seudun oppilaitoksista, joissa olen työskennellyt ja joissa edelleen työskentelen. Tässä on auttanut pitkäaikainen työkokemus ja opintosuunnitelmien ja kurssien ja infrastruktuurin kehittymisen seuraaminen. ACM-luokitus on laaja luokitus, joka kattaa tietoteknisen terminologian. Muut luokitukset ovat konkreettisempia ja niiden tarkoituksena on auttaa käyttäjää löytämään käytännönläheinen luokitus kulloisellekin avun tarpeelleen. Toinen luokka on avun tyyppien luokka, jossa on ilmaistuna avainsanoja tavallisimmille tarpeille, joita avun tarvitsija kohtaa. Kolmas luokka on tuotteiden, ohjelmien ja työkalujen luokka, joka sisältää mm. yleisten ohjelmallisten työkalujen tuotenimiä. Neljäs luokka tarkentaa edelleen avun antajien ilmoittamia osaamisalueita, kuten esimerkiksi 'asennukset'.

3.2.2 Match-algoritmi

Match-algoritmi (Kuva 4.) tekee auttajien valinnan tuloslistaan hakuavainten perusteella. Lisäksi valintaan vaikuttavat auttajien poissaolotiedot, kieltäytymiset ja kuormitus. Valinta on siinä mielessä robusti, että yksikin hakutekijän täsmääminen tuottaa tuloslistan (tulosteet 5.2 ja 7.2.1).



Kuva 11. Match-algoritmi.

Algoritmi on seuraavanlainen.

1. Hakutekijät on valittu ja valinta voi alkaa.
2. Etsi tietokannasta kaikki ne henkilöt, joiden rooli on auttaja. Auttajahenkilöt tunnistetaan roolinumerosta kaksi. Karsi

Radusta listasta ne henkilöt, joiden paikalla-tieto osoittaa henkilön olevan poissa (0), esimerkiksi lomalla. Valitse uuteen listaan jäljellejääneistä ne henkilöt, joilla on parametrinaan ensimmäisestä valikosta valittu hakusana.

3. Jos tuloslista on tyhjä...

3.2 ...anna tuloksena viesti tyhjästä hakutuloksesta.

4. Jos jäljelle jääneessä listassa on enemmän kuin kolme henkilöä...

4.2 ...ota ne, joilla on parametrinaan toisesta listasta valittu hakusana.

6. Mikäli jäljelle ei jäisi yhtään henkilöä...

6.2 valitse hakutulos1:stä ne, joilla on kolmas, neljäs tai molemmat hakutekijät.

7. Jos tuloksena on nolla auttajaa...

7.1 ...valitse hakutulos2:sta yhdestä kolmeen kandidaattia ja järjestä heidät.

7.1.1 Tulosta yhdestä kolmeen auttajakandidaattia.

7.2 ...valitse hakutulos1:stä sopivin auttajakandidaatti.

7.2.1 Tulosta yksi auttajakandidaatti.

5. Jos hakutuloksessa on vain yksi auttajakandidaatti...

5.2 ...tulosta yksi auttajakandidaatti.

5.1 ...muutoin järjestä tulos...

5.1.1 ...tulosta kaksi tai kolme auttajakandidaattia.

8. Jos hakutuloksessa on vain yksi auttajakandidaatti...

8.2 ...tulosta yksi auttajakandidaatti.

8.1 Valitse hakutulos2:sta ne, joilla on kolmas, neljäs tai molemmat hakutekijät.

9. Mikäli jäljelle ei jäisi yhtään henkilöä...

9.2 valitse hakutulos2:sta sopivin kandidaatti.

9.2.1 Tulosta yksi auttajakandidaatti.

9.1 valitse hakutulos2:sta yhdestä kolmeen sopivinta kandidaattia ja järjestä heidät.

9.1.1 Tulosta yhdestä kolmeen auttajakandidaattia.

Lopullinen kahden tai kolmen henkilön järjestäminen tuloslistaan tapahtuu siten, että osumakertojen ja kieltäytymisten määrät lasketaan yhteen ja summasta vähennetään hakusanaosumien arvot seuraavan taulukon mukaan:

Hakutekijä 1, vähennä 1

Hakutekijä 2, vähennä 2

Hakutekijä 3, vähennä 3

Hakutekijä 4, vähennä 4

Näin laskettu arvo määrää auttajan sijoituksen tuloslistassa siten, että pienimmän arvon omaava on listan kärjessä, toiseksi suurimman arvon omaava toisena ja kolmanneksi suurimman arvon omaava viimeisenä. Jos kandidaateilla on samoja arvoja, järjestys määräytyy nimen mukaan aakkostaen nousevasti.

Kun lopullinen hakutulos on valmistunut, hakutuloksen henkilöistä tallentuu tieto henkilön valinnasta, hakusanayhdistelmästä ja järjestysnumerosta listassa, sekä muut transaktiotiedot.

Mikäli hakusanayhdistelmä olisi esimerkiksi

E. Data (data)

Pika-apu

MS Access

lomake

voisi haun tulos olla esimerkiksi seuraava:

Tomi Tanskanen

E. Data (data), Pika-apu, MS Access, lomake

tomi.tanskanen@koulu1.fi

019 5256 225

050 7262 776

Koulukatu 15 a 7

00110 Lahti

Marja Heino

E. Data (data), Pika-apu, MS Access

marja.heino@firma1.fi

040 7333 7346

Vaaratie 8

00300 Lahti

Timo Virtanen

E. Data (data)

timo.virtanen@koulu2.fi

019 5256 233

040 5562 442

Käyttö ja transaktiot

Järjestelmää perustettaessa tietokantaan viedään perustiedoiksi hakusanat, käyttöoikeustiedot ja järjestelmänhoitaja ensimmäiseksi käyttäjäksi. Kaikki muu tieto järjestelmään kertyy sitä käytettäessä, transaktioiden yhteydessä.

Datan kertyminen

Käyttötapahtumia ovat järjestelmän perustaminen, avun hakijan rekisteröityminen, avun hakijan kirjautuminen, avun hakijan tyytyväisyyskysely, auttajan rekisteröityminen, auttajan kirjautuminen, auttajan tietojen muuttaminen, järjestelmän automaattinen evaluointi, hakusanojen lisääminen, hakusanojen muuttaminen ja keskeisimpänä avun haku (kuvat 1 ja 2).

Haun yhteydessä kertyvä transaktiodata

haun tunniste (yksilöi haun)
aikaleima
hakija_id
hakijan tyytyväisyys asteikolla 0 - 5 (0 = täysin tyytymätön, 5 = erittäin tyytyväinen)
auttajan kieltäytyminen

haun tekninen onnistuminen

0 - ei auttajakandidaatteja
1 - yksi tai useampi kandidaatti hakutekijöillä 1, 2 tai molemmat
2 - vähintään yksi kolmella hakutekijällä (onnistumisen minimi)
3 - vähintään yksi neljällä hakutekijällä
4 - kaikki kolme neljällä hakutekijällä (teoreettinen maksimi)
uusi hakusanakandidaatti

auttajien valituksi tuleminen ja sijoitus tuloslistassa (1, 2 tai 3)

Avun hakijan malli

Avun hakijan malli koostuu tunnuksesta, joka yksilöi kunkin avun hakijan järjestelmässä, nimestä, jonka hakija on valinnut rekisteröityessään järjestelmään, sähköpostiosoitteesta, hakujen määrälaskurista, käytetyistä hakusanayhdistelmistä ja niiden aikaleimoista, hakutuloksista ja tyytyväisyystiedoista.

tunnus

nimi

sähköposti

hakujen määrä

hakusanayhdistelmät

hakujen ajankohdat

hakutulokset

tyytyväisyys

Avun hakijan yleinen malli

Akun hakijan yleinen malli käsittää tunnuksen, nimen, hakujen määrän yhteensä järjestelmässä, hakusanayhdistelmien laskurit ja aikaleimat, hakutulosten keskiarvo ja tyytyväisyyskeskiarvo.

tunnus
nimi
sähköposti
hakujen määrä yhteensä
hakusanayhdistelmät
hakujen ajankohdat
hakutulokset
tyytyväisyys

Auttajan malli

Auttajan malli muodostuu yksilöivästä tunnuksesta, nimestä, sähköpostiosoitteesta, osuneiden hakujen määrästä ja keskiarvosta, hakusanayhdistelmistä, joilla auttaja on tullut valituksi ja näiden aikaleimoista, kieltäytymisistä, poissa/paikalla-tiedosta ja auttajaa kuvaavista hakusanoista.

tunnus
nimi
sähköposti
valituksi tulojen määrä (osuneiden hakujen määrä)
sijainti tuloslistassa
hakusanayhdistelmät, joilla valittu
ajankohdat
kieltäytymiset
poissa
omat hakusanat

Auttajan yleinen malli

tunnus
nimi
sähköposti
valituksi tulojen määrä yhteensä (osuneiden hakujen määrä yhteensä)
sijainti tuloslistassa
hakusanayhdistelmät, joilla valittu
ajankohdat
kieltäytymiset
poissa
omat hakusanat

Järjestelmän malli

hakuja
onnistuneita hakuja
hakusanoja

Hakutapahtuman malli

Järjestelmä voisi validoida itseään

- eniten käytetyt hakusanayhdistelmät, top 10
- vähiten käytetyt hakusanayhdistelmät, top 10

3.2.3 Kohdistusmekanismin toiminta

Järjestelmää ja sen kohdistusmekanismia on tarkoitus testata ja evaluoida ilman ulkopuolisia testaajia. Siksi olen tuottanut apumekanismini tietokantaan ohjelmallisesti auttajia PHP-kielisellä apuohjelmalla. Jokainen auttaja on saanut jokaisen neljän luokan jonkin (yhden) vahvuuden, kompetenssin. Kompetenssit on arvottu PHP-kielen rand-funktiolla siten, että kunkin neljän luokan jokin kompetenssi on annettu auttajalle. Tilanteena tällainen on teoreettinen, mutta suuri auttajien massa on näin kompetensseiltaan tasaisesti jakautunut ja pyrkii kuvaamaan tilannetta, jossa auttajien kriittinen massa on saavutettu.

Auttajien testidata tuotettiin tai tuotetaan tarvittaessa uudelleenkin siten, että ensin ohjelma lisäsi uuden henkilön henkilö-tauluun ja hänelle generoitiin tekninen nimi ja sähköpostiosoite ja uuden henkilön rooliksi asetettiin 'auttaja' (henkilo_rooli-taulu). Sitten laskettiin luokan yksi avainsanojen määrä ja arvottiin tämän määrän perusteella yksi numero esimerkiksi niin, että jos avainsanoja oli 12, arvottiin numeroista 1-12 yksi numero ja sitä vastaava avainsana asetettiin auttajan kompetenssiksi (hlo_hakusana-taulu). Tämä sama toimenpide tehtiin kaikille neljälle avainsanaluokalle. Näin tuotettiin tuhat (1000) auttajaa, joilla on siis kullakin neljä kompetenssia, yksi kustakin luokasta.

Kun kunkin auttajan neljä hakusananumeroa laskettiin yhteen, saatiin summa, jota verrataan muiden auttajien vastaavaan summaan. Näiden kaikkien hakusananumeroiden summien keskiarvoksi tuli 30,9, maksimin keskiarvoille ollessa 38,5 ja minimin ollessa 23,3. Tämä kertoo kompetenssien hajonnan olevan auttajilla varsin tasaista ja se tulnee vaikuttamaan hakutuloksiin niin, että ne ovat hyvin tasaisia. Tämä tarkoittaisi sitä, että hakutulokset ovat tasaisen hyviä.

SQL-kyselyllä nähdään esimerkki datasta. Mukaan otetaan neljä auttajahenkilöä tunnistenumeroilla 1-4. Henkilöiden tunnisteet alkavat numerosta yksi. Järjestelmä on generoinut numerot. Kysely on seuraava:

```
select distinct * from henkilo, hlo_hakusana, hakusana
where henkilo.henkilo_id = hlo_hakusana.hlo_id
and henkilo.henkilo_id < 5
and hlo_hakusana.hakusana_id = hakusana.hakusana_id;
```

Kyselyn tulos hiukan editoituna näyttää seuraavalta:

```
1 henkilö1 sposti1 000 000 0000 1 1 13 13 L. Teaching ( 1
1 henkilö1 sposti1 000 000 0000 1 1 15 15 Oppitunti 2
1 henkilö1 sposti1 000 000 0000 1 1 41 41 ASP 3
1 henkilö1 sposti1 000 000 0000 1 1 67 67 hallinto 4
```

```
2 henkilö2 sposti2 000 000 0000 1 2 9 9 I. Computing 1
2 henkilö2 sposti2 000 000 0000 1 2 23 23 Opastus 2
2 henkilö2 sposti2 000 000 0000 1 2 50 50 UNIX 3
2 henkilö2 sposti2 000 000 0000 1 2 57 57 tietoturva 4
```

```
3 henkilö3 sposti3 000 000 0000 1 3 5 5 E. Data (data 1
3 henkilö3 sposti3 000 000 0000 1 3 23 23 Opastus 2
3 henkilö3 sposti3 000 000 0000 1 3 49 49 Windows 3
3 henkilö3 sposti3 000 000 0000 1 3 56 56 ohjelmointi 4
```

4 henkilö4 sposti4 000 000 0000 1 4 3 3 C. Computer S 1
 4 henkilö4 sposti4 000 000 0000 1 4 14 14 Pika-apu 2
 4 henkilö4 sposti4 000 000 0000 1 4 35 35 COBOL 3
 4 henkilö4 sposti4 000 000 0000 1 4 57 57 tietoturva 4

Kuva 6. Neljä auttajaa kompetensseineen.

Sarakkeet ovat (kenttänimiltään: henkilo_id, henkilo_nimi, henkilo_sposti, henkilo_puh, paikalla, hlo_id, hakusana_id, hakusana_id, hak_sana ja taso_nro) henkilön yksilöivä tunniste, henkilön automaattisesti generoitu nimi, sähköposti ja puhelinnumero, paikallaolotieto (1 tarkoittaa, että henkilö on paikalla), henkilön tunniste hlo_hakusana-taulussa, hakusanan tunniste hlo_hakusana- ja hakusana-tauluissa, selkokieline hakusana ja hakusanan tasonumero (luokka). Hakusanan tunniste ja selkokielliset hakusanat pienessä näytteessä antavat viitteitä siitä, että kompetenssit muodostanevat varsin satunnaisia kombinaatioita. Tämä on datan tuottamisessa ollut tavoitteena.

Hakusanalukissa on hakusanoja seuraavasti. Hakusanalukka ja taso_nro ovat tässä synonyymejä. Määrät on laskettu SQL-kyselyillä seuraavasti:

```
SELECT count(*) FROM hakusana WHERE taso_nro = 1;
```

tulos: 12

```
SELECT count(*) FROM hakusana WHERE taso_nro = 2;
```

tulos: 10

```
SELECT count(*) FROM hakusana WHERE taso_nro = 3;
```

tulos: 28

```
SELECT count(*) FROM hakusana WHERE taso_nro = 4;
```

tulos: 18

Edellä kuvattu testidata, henkilöt, heidän roolinsa ja kompetenssinsa ovat mukana, kun kohdistusmekanismin toimivuutta testataan. Edellä mainitussa tilanteessa, jossa hakusanalukissa on 12, 10, 28 ja 18 hakusanaa, olisi teoreettisten neljän sanan kombinaatioiden määrä 60480.

Kun sitten avun hakijat tekevät pyyntöjä järjestelmän avulla, on avun haun testitapaukset määritelty ja onnistuneiden hakujen ominaisuudet määritelty, jolloin paras tapa tehdä haku tiettyä tarvetta varten voidaan myös määrittellä. Asettamalla hakuparametrit tietyllä tavalla saadaan paras hakutulos. Hakuparametrit asetetaan siten, että käyttäjä valitsee ohjelman tarjoamista vaihtoehdoista sellaiset, jotka kuvaavat hänen mielestään parhaiten käsillä olevaa tarvetta tai ongelmaa.

Olen kuvannut tekstuaalisesti kymmenen tilannetta, jossa käyttäjä tai käyttäjät tarvitsevat eri tyyppistä tietoteknistä tukea. Kustakin tilannekuvauksesta on tehty yhteenveto, jossa tulkitaan tilanteen vaatimaa apua. Lisäksi tilannekuvauksesta on muodostettu tietoteknisten termien ja parametrikandidaattien luettelo, jonka perusteella käyttäjä valitsee ohjelman tarjoamista hakusanoista parhaimmin tarvetta kuvaavat. Tämä kuvaa sitä prosessia, jonka käyttäjä todellisessa tilanteessa tekee tavalla tai toisella, mielestään, paperille tai muilla tavoin. Mikäli jokin luettelo ei tarjoa sopivaa avainsanaa, valinta jätetään tekemättä ja hakuprosessi perustuu valittuihin avainsanoihin.

Vertaamalla tarvetapauksien hakuja ja niiden onnistumista, voidaan muodostaa kuva hakuparametrien tehokkaasta valinnasta ja määritellä mekanismin paras käyttötapa ja kuvata se, miten mekanismi toimii kuvatuilla testitapauksilla. Hakutulosten perusteella voidaan verrata tarpeen määrittelyn avainsanoja ja auttajien kompetenssimääreitä ja nähdään, kuinka hyvä vastaavuus näillä on. Saako avun hakija sellaisia auttajia hakutulokseen, jotta hänen ongelmansa ratkeaa?

Kun tuhannen auttajan data on testattu, siitä tehdään varmuuskopio MySQL:n dumppiohjelmalla ja data poistetaan tietokannasta. Tämän jälkeen tuotetaan uusi data samalla tavalla kuin aiemmin, mutta auttajia tuotetaan nyt vain sata. Tällä datalla tehdään samat haut kuin tuhannen auttajan datallakin ja näin saadaan vertailumateriaali tilanteesta, jossa auttajia olisi vain kymmenesosa ensimmäisen testin määrästä. Hakujen tuloksia voidaan tarkastella erillisinä ja lisäksi hakujen tuloksia voidaan verrata toisiinsa.

Eähteet

[Deters 2000]

Deters R., Developing and deploying a multi agent system, Proceedings of the fourth international conference on Autonomous agents, Barcelona, Spain, DOI Bookmark: <http://doi.acm.org/10.1145/336595.337346>, ISBN:1-58113-230-1, 175 - 176, 2000. ACM 2000 1-58113-230-1/00/6.

[Dom et al. 2003]

Dom B., Eiron I., Cozzi A., Zhang Y., GraphBased Ranking Algorithms for Email Expertise Analysis.

DMKD'03, June 13, 2003, San Diego, CA, USA. Copyright 2003 ACM 1-58113-763-x. DMKD03: 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, pages 42-48, 2003.

[Ekonoja, Lahtonen, Mäntylä 2003]

Ekonoja A., Lahtonen T. ja Mäntylä J., Tiedonhaku, 2003. Informaatioteknologia - Jyväskylän yliopiston IT-tiedekunta ja avoin yliopisto. Internetsivusto: <http://appro.mit.jyu.fi/doc/tyovaline/hakukoneet/> (tulostettu 6.6.2009).

[Eurogrid 9999]

Eurogrid: Application testbed for european grid computing.

Internet-sivusto: <http://www.eurogrid.org>, tulostettu 99.99.9999.

[Greer et al. 1998]

Greer, J., McCalla, G., Cooke, J., Collins, J., Kumar, V., Bishop, A., and Vassileva, J., The Intelligent Helpdesk: Supporting Peer-Help in a University Course. Proceedings of ITS 1998: Fourth International Conference on Intelligent Tutoring Systems, 494-503, August, San Antonio, Texas, Springer-Verlag: Berlin, Germany.

[Greer et al. 2001]

Greer J., McCalla G., Vassileva J., Deters R., Bull S., Kettel L., Lessons Learned in Deploying a Multi-Agent Learning Support System: The I-Help Experience, ARIES Laboratory, Department of Computer Science, University of Saskatchewan, Canada, 2001. Oppimisen moniagenttijärjestelmän käytössä opittua: I-Help-kokemus. Tiedosto: Greer_2001.pdf

[Griphyn 9999]

Griphyn—grid physics network.

Internet-sivusto: <http://www.griphyn.org/index.php>, luettu 99.99.9999.

[Google 2009]

Internet-sivusto: <http://www.google.com/corporate/tech.html>

(tulostettu 6.6.2009).

Formal Models for Expert Finding in Enterprise Corpora

SIGIR'06, August 6–11, 2006, Seattle, Washington, USA. Copyright 2006 ACM 1-59593-369-7/06/0008 ...\$5.00.

Tiedosto: D:\Hy\gradu\taustamateriaali\expert_finding\ft_gateway.cfm3.pdf

[Kee Y. et al. 2006]

Kee Y.-S., Yocum K. ja Chien A. A., Improving Grid Resource Allocation via Integrated Selection and Binding, Henri Casanova, Information and Computer Sciences Department, University of Hawai'i at Manoa, Proceedings of the 2006 ACM/IEEE SC'06 Conference (SC'06), SC2006 November 2006, Tampa, Florida, USA.

0-7695-2700-0/06 2006 IEEE.

[KQML 9999]

KQML

Internet-sivusto: <http://www.koders.com/java/fid17EA05BDD7FEAC86B09F9BDB66AE87C4B46C4C53.aspx>, tulostettu 99.99.9999.

[McDonald, Ackerman 1998]

McDonald D. W., Ackerman M. S., Just Talk to Me: A Field Study of Expertise Location, 1998. Department of Information and Computer Science, University of California, Irvine, CSCW 98 Seattle Washington USA, ACM 1998 1-113-009-0/89/11.

[McDonald, Ackerman 2000]

McDonald W., Ackerman M. S., Expertise Recommender:

A Flexible Recommendation System and Architecture.

Department of Information and Computer Science, University of California, Irvine, CA 92697-3425.

CSCW'00, December 2-6, 2000, Philadelphia, PA.

Copyright 2000 ACM 1-58113-222-0/00/0012.

[Moodle 2009]

Internet-sivusto: <http://moodle.org/>, tulostettu 6.6.2009.

[MPI 2009]

Internet-sivusto: http://en.wikipedia.org/wiki/Message_Passing_Interface, tulostettu 25.6.2009.

[MRJ 2009]

The portable batch system.

Internet-sivusto: <http://pbs.mrj.com>, tulostettu 99.99.9999.

[Mugdal, Vassileva 2000]

An influence diagram model for multi-agent negotiation, Chhaya Mugdal ja Julita Vassileva

7695-0625-9/00 2000 IEEE.

Tiedosto: 08858520_ieee1.pdf

[Murekauha 2008]

Internet-sivusto: <http://murekauha.com/>, tulostettu 15.6.2008.

[MySQL 2008]

Internet-sivusto: <http://www.mysql.com/>, tulostettu 15.6.2008.

[Ohjelmointiputka 2009]

Internet-sivusto: <http://www.ohjelmointiputka.net/>, tulostettu 6.6.2009.

[Shachter 1988]

Shachter, R., Evaluating Influence Diagrams. *Operations Research*. Vol. 34. No. 36, 1988, 871-882.

[Singh 2006]

Singh V., Twidale M. B., Rathi D., Open Source Technical Support: A Look at Peer Help-Giving. *Proceedings of the 39th Hawaii International Conference on System Sciences - 2006*. 0-7695-2507-5/06 (C) 2006 IEEE.

[Solomon et al. 1998]

Solomon M., Raman R. ja Livny M., Matchmaking distributed resource management for high throughput computing. In *Proceedings of the Seventh IEEE International Symposium on High Performance Distributed Computing*, Chicago, IL, July 1998.

[Tangmunarunkit et al. 2003]

Tangmunarunkit H., Decker S., Kesselman C., Ontology-based Resource Matching in the Grid — The Grid meets the Semantic Web, 2003. <http://epicenter.usc.edu/docs/iswc03.pdf> [13.2.2007], University of Southern California, Information Sciences Institute, 18-Jul-2003, 12:33, 84K.

[Teragrid 9999]

The teragrid project.

Internet-sivusto: <http://www.teragrid.org>, tulostettu 99.99.9999.

[Vassileva et al. 1999]

Vassileva J., Greer J., McCalla G., Deters R., Zapata D., Mudgal C., Grant S., A Multi-Agent Approach to the Design of Peer-Help Environments, in *Proceedings of AIED '99*, Le Mans, France, July, 1999, 38-45.

[Vassileva et al. 2003]

Vassileva J., McCalla G., Greer J., Multi-Agent Multi-User Modeling in I-Help, *User Modeling and User-Adapted Interaction archive*, Volume 13, Issue 1-2 (February-May 2003), Pages: 179 - 210, 2003, ISSN:0924-1868, Publisher Kluwer Academic Publishers, Hingham, MA, USA, DOI Bookmark: 10.1023/A:1024072706526.