Aktiivinen tiedon suodatus PDA-laitteilla

Mikko Niinivaara

Espoo 10.10. 2000

Ubiquitous Computing -seminaarin kirjoitelma

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

Sisältö

Termien selitykset

Dokumentti

Mikä tahansa yksikkö informaatiota, josta on olemassa (tai josta voidaan muodostaa) metatietoa. Tyypillisesti tekstitiedosto, jossa metatietoina esim. otsikko, tekijä, päivämäärä jne.

Konteksti (*Context)*

Kontekstilla tarkoitetaan tässä kirjoitelmassa sitä tilannetta, jossa käyttäjä dokumenttejaan kulloinkin tarvitsee. Kontekstiin liittyy mm. aika, paikka ja aktiviteetti (eli mitä käyttäjä parhaillaan on tekemässä).

PDA, kämmentietokone (*Personal Digital Assistant, Palmtop Computer*)

Nimensä mukaisesti kämmenelle mahtuva tietokone. Malleja on useita erilaisia, osassa on näppäimistö, osassa tieto syötetään kynällä.

Relevanssi, relevanssiarvo (*relevance*)

Tiedonhaussa käytetty termi. Kuvaa sitä kuinka hyvin dokumentti vastaa käyttäjän sen hetkistä tiedon tarvetta. Dokumentit luokitellaan relevanssiarvonsa (esim. 1-100) avulla ja käyttäjälle näytetään vain tietyn relevanssiarvon ylittävät dokumentit.

Johdanto

Kämmentietokoneet eli PDA-laitteet ovat ideaalisia laitteita ajankohtaisen tiedon seurantaan, sillä PDA-laitteet ovat mukana kannettavia ja usein ainakin laajennettavissa toimimaan langattomassa verkossa. PDA-laitteisiin littyy kuitenkin pöytäkoneisiin verrattuna rajoituksia pienen näytön koon ja käytettävän siirtokaistan leveyden suhteen. Tämän vuoksi sekä näytöllä tulevan että verkon yli siirettävän tiedon määrää kannattaa pyrkiä rajoittamaan. Yksi tapa rajoittaa tiedon määrää on erilaisten tiedonsuodatustekniikoiden käyttäminen.

Kirjoitelman ensimmäinen luku käsittelee perinteistä tiedon suodatusta, jossa dokumenttien relevanssi lasketaan matemaattisten mallien avulla. Käyttäjän tulee antaa säännöt, joiden mukaan suodatus tapahtuu. Metodin etuna on se, että käyttäjälle on selvää miten dokumentin relevanssiarvo määräytyy. Haittana voidaan pitää sitä, että käyttäjän pitää itse muodostaa joskus varsin monimutkaisetkin suodatussäännöt. Ensimmäinen luku perustuu lähinnä lähteisiin [Mat96] ja [Fol92].

Toisessa luvussa käsitellään oppivia järjestelmiä, joilla voidaan paikata puhtaan sääntöpohjaisen järjestelmän heikkouksia. Oppivat järjestelmät perustuvat metaforalle henkilökohtaisesta avustajasta, joka ajan myötä oppii mitkä dokumentit ovat milläkin ajan hetkellä tärkeitä. Oppivan suodatusjärjestelmän etuna on sen sopeutuvuus käyttäjän muuttuviin kiinnostuksiin sekä se, että käyttäjän ei tarvitse osata kirjoittaa monimutkaisia sääntölausekkeita. Huonona puolena on mm. se, että aluksi järjestelmä ei toimi kovin automaattisesti. Toinen luku perustuu lähteisiin [Mit94] ja [Mae94].

Kolmannessa luvussa esitellään tiedonsuodatustapa, jossa käytetään hyväksi tietoa käyttäjän kontekstista. Tässä PDA laite onkin ideaalinen apuväline, sillä jo pelkästään päivän kalenterimerkinnöistä voidaan päätellä hyvin paljon siitä, missä ja mitä tekemässä käyttäjä milloinkin on. Tämän lisäksi PDA -laitteesta voidaan saada tietoa käyttäjän ja muiden samassa verkossa olevien laitteiden sijainnista. Näillä tiedoilla voidaan tehostaa kahta edellämainittua suodatusmenetelmää hyvin tehokkaasti, jolloin päästään siihen, että PDA laitteesta on jatkuvasti saatavilla sillä hetkellä ajankohtaista tietoa (=relevantit dokumentit), ilman että käyttäjän tarvitsee niitä itse "kaivaa esille". Kolmas luku perustuu lähteisiin [Sch94] ja [Fit93].

Kirjoitelman tarkoituksena ei ole antaa syvällistä teknistä selontekoa em. tekniikoista vaan esitellä yleisellä tasolla keinoja parantaa dokumenttien hallintaa erityisesti PDA-laitteilla. Tekniikoiden toimintaan käytännössä tutustutaan seminaariin liittyvässä esitelmässä.

1. Sääntöpohjainen suodatus

1.1 Teknisesti

Tiedon suodatus on hyvin läheistä sukua tiedonhaulle (*Information Retrieval*). Kummassakin tavoitteena on selvittää dokumenttijoukosta käyttäjän kannalta relevantit dokumentit ja minimoida epärelevanttien dokumenttien osuus. Tiedon suodatusta käytetään yleensä, kun dokumenttijoukosta halutaan poistaa epärelevantit dokumentit, tiedonhaulla taas pyritään löytämään relevantit dokumentit.Tiedon suodatusta käytetään, kun käyttäjällä on pitkäaikaisia muuttumattomia kiinostuksen kohteita, kun taas tiedonhakua käytetään kertaluontoiseen yksittäisten dokumenttien hakuun. Esimerkiksi jos etsin Altavistalla jääkiekkosivuja, joissa esiintyy vaikkapa sana “Espoo-Blues”, on kyseessä tiedonhaku. Jos taas ohjelma seuraa jääkiekkosivustojen muutoksia ja lähettää sanan “Espoo-Blues” havaitessaan ko. sivun sähköpostina, on kyseessä tiedon suodatus.

Yksi tiedon suodatuksen aikaisimmista sovelluksista oli SDI (*Selective Dissemination of Information)* –järjestelmä, joka kehitettiin ilmoittamaan tiedemiehille, kun heidän alaltaan julkaistiin uusia kiinnostavia dokumentteja. Käyttäjä antoi järjestelmälle luettelon avainsanoista, jotka kuvasivat käyttäjän kiinnostuksen kohteita. Näitä sanoja käyttäen SDI sitten yritti ennustaa, mitkä julkaistuista artikkeleista mahtaisivat kyseistä tiedemiestä kiinnostaa. Yksittäisten avainsanojen käyttöön liittyy kuitenkin selviä ongelmia. Yksittäiset sanat pystyvät harvoin yksiselitteisesti ilmaisemaan käyttäjän kiinnostuksia. Sama sana saatta esim. merkitä montaa asiaa tai saman asian ilmaisemiseen voidaan käyttää useita sanoja. Ongelmaan törmää usein myös Internetin hakukoneita käyttäessään. SDI –järjestelmä ei koskaan saavuttanut suurta suosiota.

Niin nykyaikainen tiedon suodatus kuin –hakukin perustuu dokumenttien samankaltaisuuden määrittelyyn. Ensimmäinen dokumenttien samankaltaisuutta mitannut malli oli vektorimalli (*vector model*). Siinä dokumenttia kuvaa matriisi, jonka arvoina ovat sanojen esiintymistiheydet dokumentissa. Dokumenttien samankaltaisuusarvo lasketaan näistä matriiseista sisätulon tai kosinitulon avulla. Myös käyttäjän antaman suodatusehdon voidaan ajatella olevan dokumentti, jonka kanssa tarpeeksi erilaiset dokumentit suodatetaan. Tätä mallia vaivaavat kuitenkin osin samat ongelmat kuin SDI-järjestelmää, koska se perustuu oletukseen, että sanat ja dokumentit ovat keskenään täysin riippumattomia.

Latent Semantic Indexing (LSI) on vekorimallin laajennus, jolla pyritään paikkailemaan vektorimallin ongelmia. LSI:ssä pyritään selvittämään dokumenttien ja niissä esiintyvien sanojen (termien) yhteyksiä. Oletuksena on, että dokumenteissa olevien sanojen esiintymistä voidaan löytää piileviä semanttisia rakenteita. Näiden rakenteiden etsimiseen käytetään tilastollisia menetelmiä. LSI:llä päästään siihen, että jos kahta eri sanaa on käytetty semanttiselta rakenteeltaan samanlaisissa dokumenteissa, on niiden esitys LSI-mallissa sama. Sekä dokumentit että käyttäjän haku- tai suodatusehdot esitetään sitten tämän semanttisen rakenteen avulla. LSI-malli löytää käyttäjän ehtojen kannalta semanttisesti saman tyyppiset dokumentit, vaikka niissä ei esiintyisikään samoja sanoja kuin käyttäjän antamassa ehdossa. LSI onkin osoittautunut hyväksi perusmalliksi niin tiedonhaulle kuin –suodatuksellekin.

Samanlaisia sääntöpohjaisia suodatusmenetelmiä käyttävät pohjimmiltaan myös seuraavissa luvuissa esitellyt metodit, mutta niissä sääntöjen muodostamista on pyritty automatisoimaan.

1.2 Käyttäjän kannalta

Käyttäjän kannalta eri sääntöpohjaiset suodatusmallit eivät eroa juurikaan toisistaan. Käyttäjä antaa sääntönsä boolen –logiikka ja/tai säännöllisiä lausekkeita muistuttavalla sääntökielellä, josta muodostetaan varsinainen hakuehto. Sääntölausekkeista muodostuu kuitenkin helposti varsin monimutkaisia. Perus suodattimien muodostukseen riittävät kuitenkin yleensä kohtuullisen yksinkertaiset säännöt, kuten kaikki Inernetin hakukoneita käyttäneet tietävät. Käytettäessä selkeitä sääntöjä on myös käyttäjälle selvää miksi tietyt dokumentit suodattuivat pois tuloksesta.

Tyypillinen käyttötilanne:

Käyttäjä haluaa nähdäkseen ne sähköpostit Mark Matthewsilta, joissa esiintyy sana Motorola (englannin) monikossa tai posessiivimuodossa. Käyttäjän tulee syöttää seuraavalainen suodatusehto (esimerkki on CLUES –järjestelmästä):

```
From: *Mark Matthews*
^Subject:*[^a-zA-Z0-9]+Motorola+((([^a-zA-Z09]|es|s’ ?s)
+.*|([^a-zA-Z0-9)|es|s|’?s)*$)).*
```

Helppoa, eikö totta? Yllättäen myös CLUES -järjestelmässä sääntöjen muodostusta on pyritty automatisoimaan mahdollisimman pitkälle.

2. Oppivat suodatusjärjestelmät

2.1 Teknisesti

Koneoppimista hyödyntävät agentit (*Personal Software Assistant, Personal Digital Assistant)* perustuvat metaforaan henkilökohtaisesta avustajasta. Aluksi avustaja ei tiedä käyttäjän mieltymyksiä eikä siitä välttämättä ole kovin paljon hyötyä. Ajan myötä austaja kuitenkin oppii käyttäjän mieltymykset seuraamalla käyttäjän työtapoja.

Jotta oppivan järjestelmän käytöstä saataisiin kaikki hyöty irti, tulee kahden peruslähdtökohdan toetutua: ohjelman käytön tulee sisältää suuri määrä toistuvia toimintoja ja näiden toistuvien toimintojen tulee olla erilaisia eri käyttäjillä. Jos jälkimmäinen ehto ei toteudu, päästään hyvään lopputulokseen jo ennalta ohjelmoidulla sääntöpohjaisella järjestelmällä. Dokumenttien seurannassa ehdot täyttyvät kuitenkin kevyesti. Eri käyttäjiä kiinnostavat (myös samoissa tilanteissa) eri dokumentit ja jopa saman käyttäjän kiinnostukset muuttuvat ajan myötä. Tyypillisesti tällainen dokumenttien seuranta on myös toistuvaa ja pidempiaikaista.

Oppiva agentti oppii neljällä tavalla. Agentti seuraa jatkuvasti käyttäjän suorittamia toimintoja tämän käsitellessä dokumenttejaan. Pidemmän ajan kuluessa (tyypillisesti viikkoja tai kuukausia) agentti löytää käyttäjän toiminnoista yhteneväisyyksiä (patterns). Esimerkiksi käyttäjä saattaa tallettaa lähes aina “Ubiquitous Computing” uutisryhmään lähettämänsä sähköpostit kansioon “Ubic”. Samalla tavalla agentti voi etsiä käyttäjän lukemista dokumenteista (esim. LSI:tä käyttäen) yhteneväisyyksiä ja tämän jälkeen ehdottaa käyttäjälle saman tyyppisiä dokumentteja automaattisesti.

Toinen tapa oppia on käyttäjän antama suora ja epäsuora palaute. Kun käyttäjä hylkää agentin antaman ehdotuksen on kyseessä epäsuora palaute. Käyttäjä voi esimerkiksi muuttaa dokumenttiensa lukemisjärjestystä, jättää jotkut dokumentit lukematta tai lukea dokumentteja joita agentti ei ehdottanut (eli suodatti pois). Käyttäjä voi myös suoraan ilmoittaa agentille vaikkapa että “Älä enää koskaan tee näin!” tai “Tämä artikkeli on syvältä!”, tällöin on kyseessä suora palaute.

Kolmas tapa oppia on käyttäjän antamat esimerkit. Käyttäjä voi vaikkapa lähettää itselleen muka-sähköpostia tietyltä henkilöltä ja jättää muut kuin lähettäjä -kenttä tyhjäksi. Tämän postin käyttäjä sitten ohjaa tiettyyn kansioon ja käskee agenttia tekemään aina näin. Samoin käyttäjä voi käskeä tuomaan heti näytölle sellaiset dokumentit, joissa esiintyy sana “Espoo-Blues”. Käytännössä tämä tapa on hyvin lähellä perinteistä suodatussääntöjen laatimista.

Neljäs tapa on kysyä neuvoa muilta agenteilta, joilla voi olla enemmän kokemusta vastaavasta tilanteesta. Esimerkiksi jos agentti ei tiedä mitä tehdä sähköpostiviestillä, jonka lähettäjäksi on merkitty Nicholas Negroponte, se voi kysyä mitä muut agentit (=niiden käyttäjät) tekevät moiselle postille. Jos suurin osa agenteista suosittaa, että dokumentille tulisi antaa korkea relevanssiarvo, voi agentti näyttää dokumentin heti käyttäjälle vaikka tämä ei koskaan aikaisemmin olekaan saanut postia ko. henkilöltä. Käyttäjä voi halutessaan kertoa agentilleen keiden käyttäjien agenteilta kannatta kysellä toimintaohjeita missäkin tilanteessa. Esimerkiksi UbiComp seminaarin pitäjän agentilta voisi irrota hyviä suosituksia aihetta käsittelevistä dokumenteista. Agentti voi vastaavasti myös itse oppia minkä agentin (tai agenttien) suositukset osuvat usein yksiin käyttäjän mieltymysten kanssa.

2.2 Käyttäjän kannalta

Käyttäjän kannalta oppivan järjestelmän etuja on se, että käyttäjän ei tarvitse osata syöttää monimutkaisia sääntölausekkeita vaan järjestelmä muodostaa niitä itse. Hyötynä on myös se, että järjestelmä mukautuu käyttäjän muutuviin kiinostuksen kohteisiin (joskin vasta ajan myötä). Oppiva agentti voi myös epäsuorasti käyttää kontekstia hyväkseen esim. juuri valitsemalla tietyt agentit neuvojikseen tai havaitsemalla yhteyden kellonajan ja luettujen artikkelien välillä. Tällöin käyttäjä huomaa saavansa vaikkapa Vallilan bussiaikataulut eteensä aamusella sopivasti ennen laitokselle lähtöä. Ongelmana käyttäjän kannalta on se, että aluksi agentista ei juuri ole iloa. Tätä voidaan kuitenkin helpottaa esiasentamalla agentille jotkut perus oletukset käyttäjän kiinnostuksista.

Tyypillinen käyttötilanne:

Käyttäjä on tähän mennessä lukenut heti Teuvo Olvilta saapuneet viestit, joskus NHL tulokset ja siirtänyt aina $$$Money fast –dokumentin kanssa samankaltaiset viestit roskakoriin. Tilanne näytöllä voisi olla vaikka seuraavan näköinen (Teuvo Olvin viesti on luonnollisesti jo näytetty):

Uudet dokumentit:

Tekijä:	Pvm:	Varmuus:	Toiminto:	Otsikko:
Teuvo Olvi	16.10.2000	96%	Näytä heti	Tänään kaljalle?
Www.sportti.fi	16.10.2000	53%	Talleta	NHL tulokset
Joe Smith	08.10.2000	90%	Roskiin	$$$Money fast…

Haluatko tallettaa dokumentin "NHL tulokset" vai saada sen heti näytölle?

[Talleta] [Näytä] [ ] Tee aina näin

[Poista]

3. Kontekstin käyttö suodatuksessa

3.1 Teknisesti

Yleensä käyttäjän kontekstiin katsotaan kuuluvan käyttäjän (PDA:n) sijainnin ja myös hänen lähellään olevat muut käyttäjät ja laitteet. Teknisesti laitteiden sijainnin määritys voidaan hoitaa esim. jakamalla verkko huoneen kokoisiin soluihin, jolloin laitteen sijainti voidaan päätellä siitä, minkä solun alueella laite on. Voidaan käyttää myös GPS (*Global Positioning System*) –järjestelmää tai päätellä sijainti yksinkertaisesti seuraamalla laitteiden välistä verkkoliikennettä. Käyttäjän sekä verkon laitteiden sijainnista voidaan sitten päätellä mitä tekemässä ja kenen kanssa käyttäjä milläkin hetkellä on, eli yritetään päätellä mitkä dokumentit käyttäjää sillä hetkellä kiinnostaisivat. Myös esim. lähimmän verkkotulostimen sijainnin selvitääminen yms. onnistuu, mutta tämä ei yleensä ole dokumenttien suodatuksen kannalta relevanttia. Erityisesti käyttäjän ja laitteiden sijaintitiedosta on hyötyä silloin, kun johonkin laitteseen tai paikkaan liittyy kiinteästi yksi tai useampia dokumentteja. Esimerkiksi, kun käyttäjä saapuu kopiokoneen läheisyyteen, saa hän PDA-laitteensa näytölle kopiokoneen käyttöohjeet.

PDA-laitteissa mahdollisia tiedon lähteitä siitä, mitkä dokumentit käyttäjää mahtavat kiinnostaa löytyy muitakin. PDA-laitteita on perinteisesti käytetty taskukalenterin korvikkeena, joten niihin kuuluu luonnolisena osana kalenteriohjelmisto. Tyypillisesti käyttäjä täyttää ohjelmaan tiedot missä ja/tai mitä tekemässä hän päivän mittaan on. PDA-laitteita käytettäessä ei siis välttämättä tarvita lainkaan käyttäjän sijainnin selvittävää laitteistoa, joskin käyttäjän todellisesta tarkasta sijainnista on tietysti paljon hyötyä. Tämän lisäksi PDA-laitteissa on usein osoitekirjaohjelmisto jonka tietoja voidaan myös käyttää suodatuksen apuna.

Perinteisen sääntöpohjaisen suodatuksen apuna kalenterin tietoja voidaan käyttää esim. etsimällä kalenterista sanat tietyn ajanjakson sisällä (esim. kaksi viikkoa) ja käyttää näitä suodatussääntöjen muodostamiseen. Käyttäjän kalenteri voi olla vaikkapa seuraavan näköinen (esimerkki jälleen CLUES-järjestelmästä):

Time:	Monday	Tuesday
9	Texas Instruments visit	at AVIOS '96 all day
10	show MailCall	fax: 408-506-9987
11		
12	lunch w/Mark Matthews	
13		keynote
14	confirm airplane departure	
15	pack for trip	
16	catch 16:15 train to Logan	
17		banquet starting 17:45

Kalenterimerkinnöistä muodosteetaan kaksi listaa: toisessa ovat kaikki avainsanat eli sanat jotka eivät ole ns. stop-sanoja (englannissa esim. and,but,all), toiseen listaan kerätään erisnimet. Erisnimet kerätään omaan listaansa, koska niillä suodattaminen on luotettavampaa kuin yksittäisillä sanoilla. Esimerkkitapauksessa listat olisivat seuraavanlaisia:

Avainsanat:	Erisnimet:
Texas	Texas Instruments
Instruments	MailCall
lunch	Mark Matthews
Mark	Logan
Matthews	AVIOS
confirm	
airlane	
departure	
…	

Erisnimistä voidaan muodostaa suodatussääntö, joka tulkitsee relevanteiksi kaikki dokumentit joiden otsikoissa joku erisnimistä esiintyy. Näin muodostettu suodatin havaitsisi relevantiksi esimerkiksi dokumentin (sähköpostin), jonka otsikko olisi “Final agenda for **Texas Instruments** visit”. Avainsanoja voidaan verrata dokumenttien sisälltöön LSI -menetelmällä, jolloin sanojen yksittäiset esiintymät eivät vielä nosta dokumentin relevanssiarvoa turhan korkeaksi.

Osoitekirjaa voidaan esimerkkitapauksessa käyttää sen selvittämiseen kenelle merkinnöissä esiintyvä faksinumero kuuluu. Jos faksinumeron haltija selviää, voidaan tämän nimellä muodostaa suodatusehto.

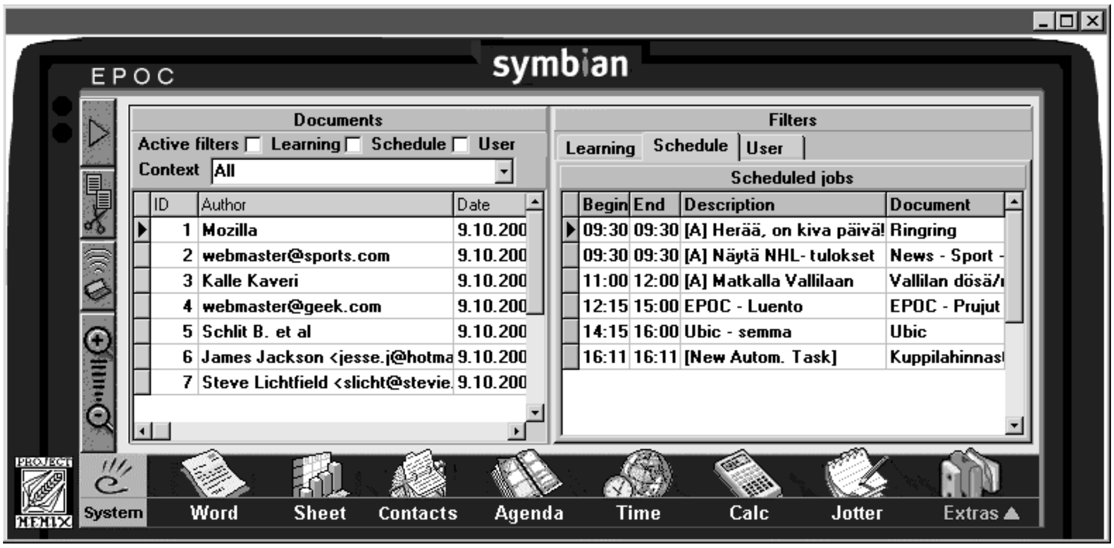
Täysi hyöty kontekstista sadaan, kun se yhdistetään oppivaan järjestelmään. Tällöin suodatussääntöjä muodostettaessa voidaan ottaa huomioon se, mitä dokumentteja käyttäjä on missäkin tilanteessa tarvinnut. Esimerkiksi agentti voi havaita että aina, kun käyttäjä on C-kielen laskujarjoituksissa (tämä tieto voi tulla kalenterista tai käyttäjän sijainnin selvittävältä laitteistolta), avaa käyttäjä uusimman niistä dokumenteista, joiden otsikoissa esiintyy merkkijono "C-Kielen laskuharjoitukset". Samoin agentti voi huomata, että aina noin tuntia ennen päivän ensimmäistä luentoa tai laskuharjoitusta käyttäjä on kiinnostunut Vallilan bussiaikatauluista ja tuoda ne automaattiseti näyttöön. Tällöin edellisestä oppivien agenttien yhteydessä olleesta esimerkistä poiketen järjestelmä sopeutuu siihen, että laitokselle ei mennä joka päivä samaan aikaan. Näiden yhteneväisyyksien (patterns) löytämiseen järjestelmä käyttää luvussa 2. esiteltyjä menetelmiä, joten teknisesti kontekstin käyttö agentin apuna ainoastaan lisää käytettävissä olevaa informaatiota eikä edellytä muutoksia käytettyihin algoritmeihin.

3.2 Käyttäjän kannalta

PDA-laitteen käyttäjän kannalta kontekstin käyttö (yhdistettynä edellisissä luvuissa oleviin tekniikoihin) on erittäin hyvä tapa suorittaa dokumenttien suodatusta. Ihaneellisessa tapauksessa käyttäjän ei tarvitse tehdä mitään ylimääräistä työtä saadakseen relevantit dokumentit näytölleen. Riittää, että hän käyttää PDA-laitetta siihen tarkoitukseen, johon se luonnollisestikin soveltuu, eli tärkeiden dokumenttien mukana kuljettamiseen ja päivittäisen kalenterin ylläpitämiseen. Myös suodatussääntöjen syöttö helpottuu, kun apuna voi käyttää kontekstia. Käyttäjä voi esim. antaa säännön jonka mukaan 15 minuuttia herätyksen (joka tulee PDA-laitteesta) jälkeen hän haluaa näyttöön edellisenä yönä mahdollisesti pelattujen NHL-otteluiden tulokset.

Tyypillinen käyttötilanne:

Oikealla näkyvät päivän kalenterimerkinnät. Yksittäisiin tapahtumiin on liitetty dokumentti tai dokumenttijoukko, joka näytetään tapahtuman yhteydessä. Vasemmalla puolella ovat lukemattomat dokumentit. Koska mitään suodatusta eikä dokumenttien tarkastelukontekstia ole valittu, näkyvät kaikki uudet dokumentit satunnaisessa järjestyksessä.



Lähteet:

[Fit93] Fitzmaurice G.,
Situated Information Spaces and Spatialy Aware Palmtop Computers, Communications of the ACM 1993, No. 7 39-49

[Fol92]Foltz P., Dumais T.,
Perosnalized Information Delivery: An Analysis of Information Filtering Methods, Communications of the ACM 1992, No. 12, 51-60

[Mae94] Maes P.,
Agents that Reduce Work and Information Overload,
Communications of the ACM 1994, No. 7, 31-40

[Mat96] Matthew M., Schmandt C.,
CLUES: Dynamic Personalized Message Filtering,
Proc. of ACM CSCW'96 113-121

[Mit94] Mitchell T. et al,
Experience with a Learning Personal Assistant,
Communications of the ACM 1994, No. 7, 81-91

[Sch94] Schlit B. et al,
Context-Aware Computing Applications,
IEEE Wokshop on Mobile Computing Systems and Applications 1994