Ongelmakenttä

WWW-museo

Tekijät:

Kim Josefsson (KJ)
Samppa Saarela (SS)
Eetu Mäkelä (EM)
Taneli Rantala (TR)
Robert Holmberg (RH)
Pasi Lehtimäki (PL)

Helsinki 7. maaliskuuta 2002

Versio 1.0

HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

Sisältö

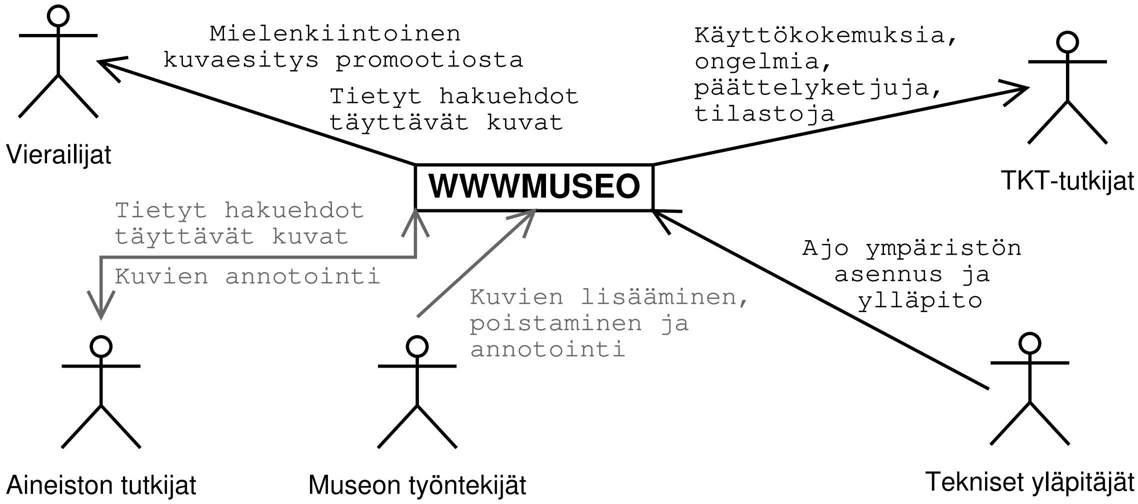
1 Johdanto (KJ)

Tässä dokumentissa käsitellään Helsingin yliopiston museon WWW-näyttelyn toteuttamiseen liittyviä ongelmia ja rajoituksia. Tämä ohjelmistotuotantoprojekti liittyy Helsingin yliopistossa tehtävään Semantic Web -tutkimukseen, jossa tutkitaan mm. tekoälyn hyödyntämistä älykkään Internetin toteuttamisessa. Projektin tehtävämäärittely löytyy Liiteestä 2. Dokumentissa pohditaan järjestelmän toteuttamiseen liittyviä ongelmia ja määritellään tärkeimpiä käsitteitä ja järjestelmän osia.

Luku 1 sisältää johdannon ja dokumentin versiohistorian. Luvussa 2 esitellään WWW-näyttelyn sidosryhmät. Luvussa 3 kerrotaan, mitä ontologiat ja fasetit ovat. Luvussa 4 kerrotaan, miten ontologioita käytetään hyväksi WWW-näyttelyssä suunnistamisessa. Luvussa kuvataan ontologoiden yhteys HiBrowse-navigointimalliiin ja museovierasta kinnostaviin mielenkiintoisiin linkkeihin: näyttely osaa ehdottaa vierailijoille linkkejä kiinnostaviin kuviin. Myös ontologioiden suhdetta käyttöliittymään ja järjestelmän tilan analysointiin käsitellään tässä luvussa. Luvussa 5 kerrotaan, minkälaista tietoa museon kuvien XML-korteissa on, ja missä muodossa tieto on. Lisäksi kerrotaan, miten korttien tietoa muutetaan ja muokataan projektin tarpeisiin sopivaksi. Luvussa 6 esitellään mm. RDF ja RDFS, eli järjestelmän rakentamisessa käytettäviä tärkeitä välineitä ja käsitteitä. Loppuun on liitetty kuvaus alkuperäisestä projektin tehtävänannosta ja näyte museon kuvatietokannan XML-kortista.

1.1 Versiohistoria

Versio	Päivämäärä	Tehdyt muutokset
0.90	27.2.2002	Lähes valmis versio.
0.91	1.3.2002	Korjailtiin kuvia ja viimeisteltiin tekstiä.
1.0	6.3.2002	Tehtiin FTR:ssä ehdotetut muutokset. Valmis versio.



Kuva 1: WWW-museon sidosryhmät. Käyttäjäryhmiä, joiden suhde järjestelmään on kuvattu harmaalla nuolella ei oteta huomioon nyt toteutettavassa osajärjestelmässä.

2 Sidosryhmät (SS)

Järjestelmällä on viisi sidosryhmää: vierailijat, aineiston tutkijat, museon työntekijät, järjestelmän jatkokehittäjät ja tekniset ylläpitäjät (kuva 1). Järjestelmän sidosryhmät on kuvattu luvuissa 2.1 - 2.5. Kuvauksissa mahdollisesti mainitut käyttötapaukset eivät ole järjestelmän varsinaisia käyttötapauksia, jotka määritellään suunnitteludokumentissa. Osaa tässä mainituista toiminnoista ei edes ole tarkoitus toteuttaa, joten osa kuvatuista käyttäjäryhmistä ei ole nyt toteutettavan järjestelmän todellisia käyttäjäryhmiä, vaan kuvaukset on kirjoitettu todelliseen käyttöön suunnatulle järjestelmälle. Luvussa 2.6 eritellään nyt toteutettavan osajärjestelmän käyttäjäryhmien eroavuudet alla kuvattuihin.

2.1 Vierailijat (SS)

Vierailijoilla tarkoitetaan ketä tahansa museon www-sivuille sattuvaa, promootiosta kiinnostunutta henkilöä, jota ensisijaisesti kiinnostaa mielekkäällä tavalla esitetty kuvallinen esitys promootiosta. Mielekkäällä tavalla tarkoitetaan tässä yhteydessä sitä, että järjestelmä pyrkii ehdottamaan kutakin vierailijaa todennäköisimmin kiinnostavia kuvia ja aiheita jatkolinkkeinä. (Luku 4.2.)

2.2 Aineiston tutkijat (SS)

Tutkija tarvitsee täsmätietoa tarkkaan määritellystä promootioaineistoon liittyvästä tekijästä. Tutkija saattaa olla kiinnostunut esimerkiksi vain tietyn aikakauden promootioista, tietyistä henkilöistä (rehtori, promovoitava jne.) tai jostain tietystä rakennuksesta.

Tutkijoille voisi myös esimerkiksi tarjota mahdollisuuden annotoida kuvia, jolloin järjestelmän tietosisältö kasvaisi dynaamisesti ja ainakin periaattessa muokkautuisi käyttäjiensä näköiseksi.

Yksittäisen tutkijan tavoite saattaa olla esimerkiksi tekeillä olevan julkaisun kuvittaminen määrätyn henkilön promootiokuvalla tai vaikkapa ikonografinen tutkimus yliopistolla harjoitettavasta musiikista. Järjestelmään toteutettavan HiBrowse-käyttöliittymän (luku 4.1) tulisi mahdollistaa hakuehdon tarkka määrittäminen mielekkäällä tavalla, jotta tutkijan olisi helppo löytää kaikki tätä kiinnostavat kuvat.

2.3 Museon työntekijät (SS)

Helsingin yliopiston Museon työntekijät keräävät aineiston näyttelyä varten ja vastaavat sen siirrosta digitaaliseen muotoon sekä ylläpidosta. Näyttely koostuu pääosin kuvista ja niihin liittyvästä metatiedosta. Kuvien lisäksi näyttelyyn kuuluu promootion sanallista kuvausta, esimerkiksi yleiskuvaus, tapahtumien ja eri henkilöiden tehtävien kuvaukset. Museon työntekijät ovat siis vastuussa järjestelmän tietosisällöstä.

Tässä projektissa ei keskitytä tietosisällön ylläpitämiseen, vaan lähtökohtana on jo olemassaoleva kuva- ja metatietotietokanta, josta valitaan promootioon liittyvät osat. Uusien näyttelyiden luominen ja puutteellisten tai virheellisten tietojen korjaaminen toteutetaan mahdollisissa jatkoprojekteissa.

2.4 Järjestelmän jatkokehittäjät (SS)

Järjestelmän jatkokehittäjillä tarkoitetaan tässä yhteydessä lähinnä laitoksella vaikuttavia Älykäs WWW -tutkimusalueeseen (Semantic Web) keskittyviä tutkijoita.[1] Heitä kiinnostaa itse ohjelman toiminta ja sen parantaminen. Olennaista on mahdollisuus tutkia uudella tekniikalla toteutetun järjestelmän toimintaa ja toimivuutta. Tutkija haluaa tietää, miten järjestelmän mielenkiintoisten kuvien päättely etenee ja kuinka hyvin tässä onnistutaan. Myös erilaiset tilastotiedot järjestelmän käytöstä ovat mahdollisesti tutkijan kiinnostuksen kohteita.

Tämän käyttäjäryhmän huomioiminen asettaa lisävaatimuksia lähinnä dokumentoinnille: dokumenteista tulisi käydä ilmi myös vaihtoehtoisia toteutustapoja ja käytettävät käsitteet on määriteltävä myös suhteellisen yleisellä tasolla (tämä dokumentti).

2.5 Tekniset ylläpitäjät (SS)

Tekniset ylläpitäjät vastaava järjestelmän toiminnasta ja ylläpidon teknisestä puolesta, esimerkiksi WWW-palvelimen toiminnasta, konfiguroinnista ja päivittämisestä.

2.6 Toteutettavan osajärjestelmän käyttäjäryhmät (SS)

Seuraavassa on kuvattu erot toteutettavan osajärjestelmän ja tuotantokäyttöön suunnatun jatkoprojekteissa kehitettävän järjestelmän välillä:

- ”Vierailijat” ja ”Aineiston tutkijat” ovat samaa käyttäjäryhmää joille tarjotaan HiBrowse ja mielenkiintoiset linkit.
- ”Museon työntekijät” eivät ole järjestelmän varsinainen käyttäjäryhmä. Heille ei toteuteta tässä projektissa tietosisällön ylläpitoliittymää. Periaatteessa he voivat kuitenkin vaikuttaa järjestelmän tietosisältöön muokkaamalla metatieto-kuvauksia tai muita kuvauksia, kuvia tai jopa ontologiaa käsin tai muiden ohjelmien avulla.

3 Ontologia (SS)

Ontologiatutkimuksen peruskysymys on ”mitä on?” Tutkimuksen kohde on johonkin tiettyyn alueeseen (domain) D kuuluvien asioiden (käsitteiden) katego-

[1] http://www.cs.helsinki.fi/u/eahyvone

riat ja näiden väliset suhteet. Pyrkimyksenä on muodostaa yhtenäinen käsitteistö asioista, joiden uskotaan kuuluvan ongelma-alueeseen. Nykyään ontologiatutkimuksessa ei niinkään pyritä muodostamaan yleistä mallia kaikista olemassaolevista asioista. Tällaisen muodostaminen olisikin käytännössä mahdotonta, sillä vastaus kysymykseen ”mitä on olemassa” riippuu ensinnäkin siitä, keneltä kysytään ja toiseksi käytettävästä kielestä. Esimerkiksi eskimoiden kielessä on huomattavasti enemmän lunta tai sen tyyppejä kuvaavia sanoja kuin suomen kielessä. Annetun ongelma-alueen D ontologia on siis aina tehty jonkin kieltä L käyttävän henkilön näkökulmasta. [Sow00]

Ontologioita on tutkittu aina tieteen historiassa. Oman näkemyksensä ovat tarjonneet niin Platon, Aristoteles, Kant kuin Peirce ja jokainen esimerkiksi olio-ohjelmointia harrastanut. Olio-ohjelmoinnissa ohjelma on tarkoitus järjestää tietyn ongelma-alueen hierarkisen ontologian mukaisesti. Esimerkiksi Javan Swing-käyttöliittymäpakkaus on yksi esimerkki käsitteen käyttöliittymä Java-kielellä määritellystä ontologiasta.

Ontologioita on mahdollista esittää hyvin monilla eri tavoilla, niin luonnollisella kielellä kuin formaalistikin, esimerkiksi predikaattilogiikalla, SQL-kielellä tai Prologilla. Nykytrendinä on kuitenkin pyrkimys datan esitysmuodon mahdollisimman suureen riippumattomuuteen sovellusalustasta, joten XML:n ilmestymisen jälkeen onkin esitetty tätä esitystapaa hyväksikäyttäviä ontologioiden määrittelykieliä. Kuten ohjelmointikielissäkin, on näiden (esim. DAML+OIL[2] tai RDFS) ontologiakielien ilmaisuvoimissa eroja. Siinä, missä yhdellä on mahdollista ilmaista syklisiä perimishierarkioita ja osallistumisrajoitteita (cardinality), ei tämä toisella kielellä ole välttämättä mahdollista.

Tässä projektissa ontologioita käytetään hyväksi kuvien luettelointitietojen antamisessa (annotoinnissa). Pitäydymme ontologioissa, jotka on mahdollista määritellä RDF:n (luku 6.1) käyttämän RDF Schema määrittelyn versiolla 1.0 (luku 6.2).

3.1 Esimerkki ontologiasta (SS)

Ontologiat muodostavat siis käsiteverkkoja, jotka on mahdollista kuvata esimerkiksi olio-paradigman avulla esimerkiksi luokkakaaviotekniikalla (UML). Tässä luvussa kuvatussa esimerkissä näkökulmaksi voidaan ottaa Sowan [Sow00] esi-

[2]http://www.daml.org

tys.

Yleiskäsitteiden jakautuminen hierarkisesti alikäsitteiksi kuvataan oliomallissa periytymisenä. Esimerkiksi luokalla `Virka` voisi olla aliluokkinaan `YlemmätToimihenkilöt` ja `AlemmatToimihenkilöt`. Yliopiston `Rehtori` puolestaan perii luokan `YlemmätToimihenkilöt`. [Kuva 2.)

Pelkkä perimishierarkia ei kuitenkaan ole riittävä kuvaamaan mitään todellista ontologiaa, joten mukaan tulevat myös luokkien väliset yhteydet eli ominaisuudet. Yllä kuvatussa viran ontologiassa esimerkiksi alemmilla toimihenkilöillä on esimies-yhteys johonkin ylempään toimihenkilöön. Jokaiselle viralle tulee myös olla mahdollista määritellä hoitaja sekä taho (virasto, yritys, laitos tms.), mihin virka liittyy. Hoitaja on tässä tapauksessa aina jokin ihminen, eli toisin sanoen luokan `Virka` ominaisuus *Hoitaja* on arvoltaan luokan `Ihminen` ilmentymä. Ihminen puolestaan on toisaalta nisäkäs ja toisaalta maalla elävä eläin. (Kuva 3.)

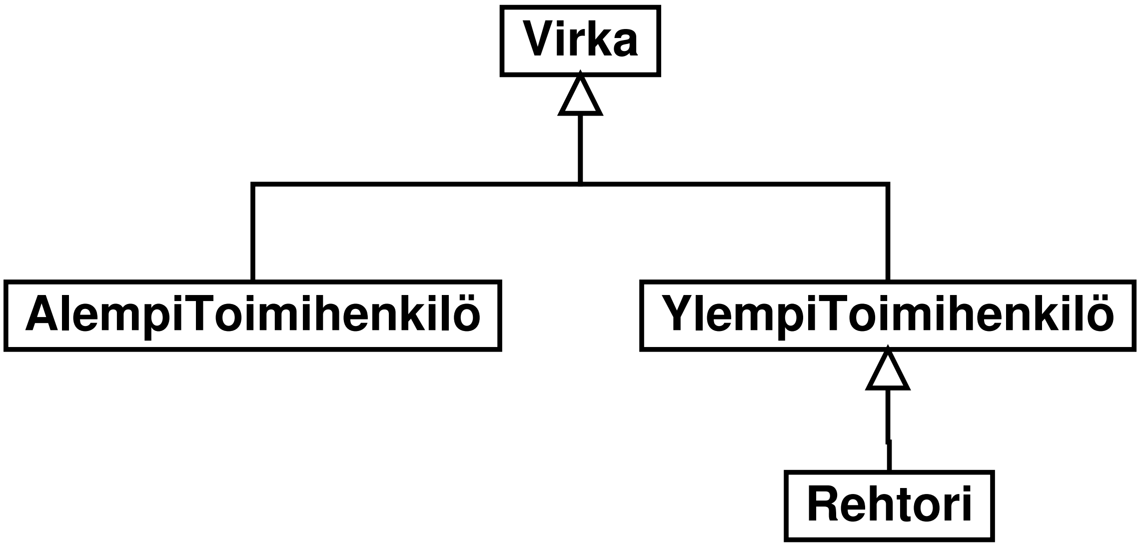
Kuvien annotoinnissa ontologian avulla (esimerkiksi RDF:llä) on pohjimmiltaan kyse siitä, että kuvan selitys eli metatieto esitetään tavalla jolla sitä pystytään käsittelemään ihmisen kannalta mielekkäällä tavalla automaattisesti. [3] Luvussa 4 esitellään muutamia tapoja, joilla tätä ontologian avulla esitettyä metatietoa voidaan käyttää hyväksi historiallisen kuva-aineiston virtuaalinäyttelyssä.

3.2 Fasetti (SS)

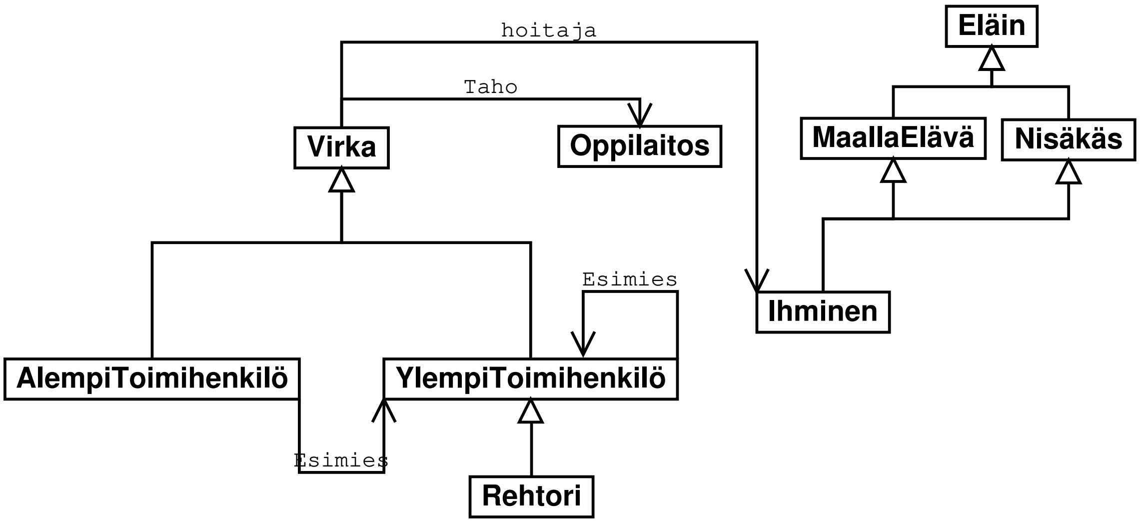
Fasetilla tai käsiteryppäällä tarkoitetaan puumaista käsitehierarkiaa eli tiettyä näkökulmaa käsiteltävään hierarkisesti luokiteltuun aineistoon. Koska ontologiassa periytymisen näkökulma tarjoaa nimenomaan tämänkaltaisen luokittelun annetulle ongelmakentälle olematta kuitenkaan moniperinnän vuoksi fasetti, on myös syytä määritellä fasetin ja ontologian suhde.

Tässä yhteydessä käyttämämme ontologian määrittely rajoittaa ontologiat perimisen suhteen syklittömiksi verkoiksi. Yhteen fasettiin kuuluu valitusta solmusta (juuri) kaikki juuri-lehti-polut. Kuvassa 4 on eläinfasetti yllä kuvatusta ontolo-

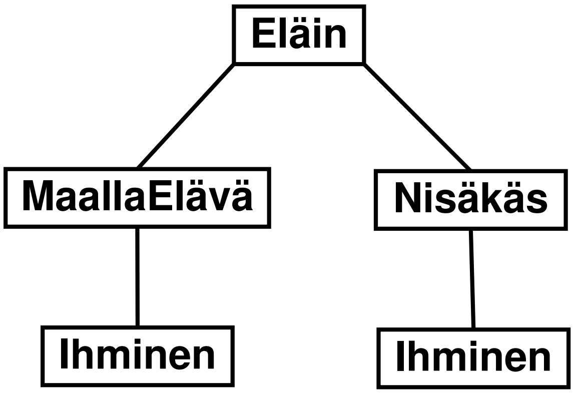
[3]Semantic web -projekti (www.w3c.org/2001/sw/) pohjautuu ontologioiden käyttöön www-sivujen tai muiden lähteiden annotoinnissa. Ontologioilla pyritään esittämään kuvauksessa käytettävien käsitteiden merkitykset tavalla, joka mahdollistaa automaattisen käsittelyn, eli toisin sanoen kuvaukset pyritään esittämään siten, että ”tietokonekin ymmärtää niiden merkityksen”. Toimiiko tämä sitten käytännössä ihmisen kannalta mielekkäällä tavalla jää nähtäväksi, ja isommassa mittakaavassa tämänkin projektin toivotaan tuovan lisävaloa juuri tähän kysymykseen.



Kuva 2: Esimerkki virka-käsitteen ontologiasta, jossa huomioitu vain perimissuhde.



Kuva 3: Esimerkki virka-käsitteen ympärille muodostuvasta ontologiasta. Kuvassa tavallinen nimetty nuoli kuvaa lähtöluokan ominaisuutta, jonka tyyppi on nuolen osoittama luokka.



Kuva 4: Eläin-fasetti kuvassa 3 määritellystä ontologiasta.

giasta.

4 Ontologia-avusteinen selailu (RH)

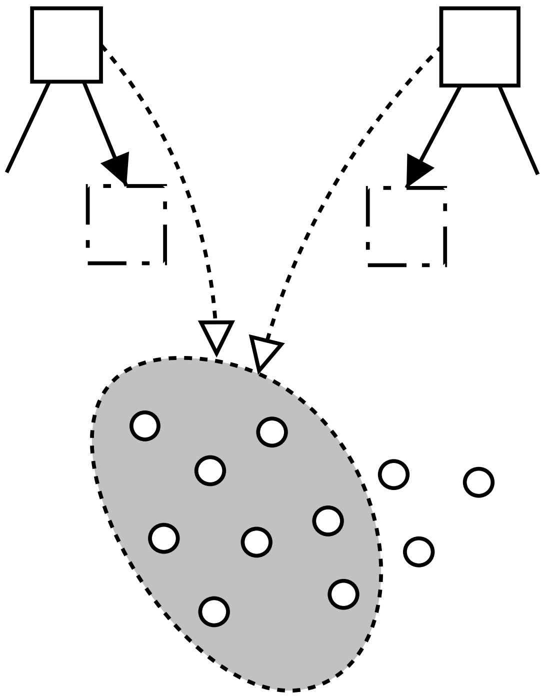
Ontologioissa navigointi on ontologian avulla luokitellun dokumenttijoukun selailua. Tässä luvussa esitetään kaksi selkeästi poikkeavaa tapaa.

4.1 HiBrowse-malli

HiBrowse [Hib02] on malli ontologioissa navigoimiseen. Ontologian fasetit ovat puurakenteita ja HiBrowse-mallissa muutetaan kerralla aina yhden fasetin *aktiivista solmua*. Ontologioista oli lisätietoa luvussa 3. Kaikkien ontologioiden fasettien aktiivisten solmujen joukko muodostaa mallin nykyisen *tilan*, tai suodatuksen, joka suodattaa ontologian kuvaamaa dokumenttijoukkoa (tässä tapauksessa kuvajoukkoa) niin, että tiettyyn tilaan kuuluva dokumenttijoukko on kaikkien sen tilan aktiivisten solmujen kuvaamien dokumenttijoukkojen leikkaus. Aktiiviseen tilaan kuuluva dokumenttijoukko voidaan kutsua *aktiiviseksi dokumenttijoukoksi* (kuva 5).

Esimerkissämme (Kuvat 5-11) on kahden fasetin (puut) suodattama (katkoviivanuolet) dokumenttijoukko (ympyrät). Kuvissa fasettien aktiiviset solmut on merkitty aina ehjällä reunalla kun taas muut solmut on merkitty katkoviivareunalla. Dokumenttijoukon aktiivinen osa on merkitty tummalla taustalla.

Aktiivisen solmun siirto yhdessä fasetissa muuttaa myös ontologian aktiivista

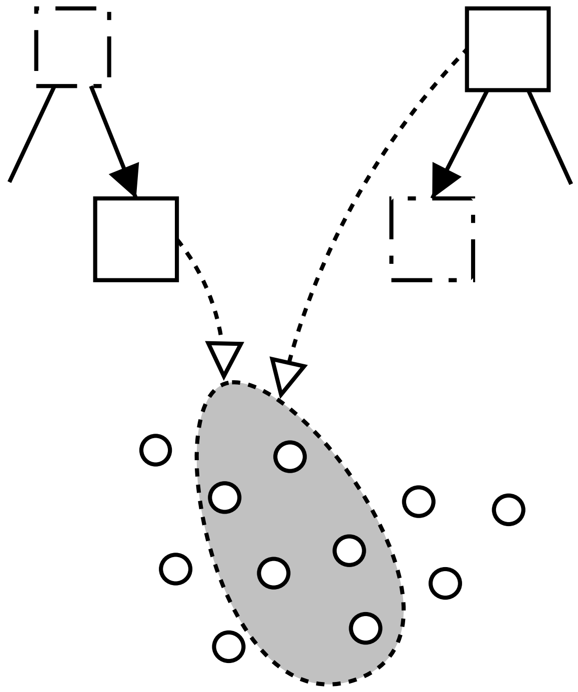


Kuva 5: Aktiivinen dokumenttijoukko.

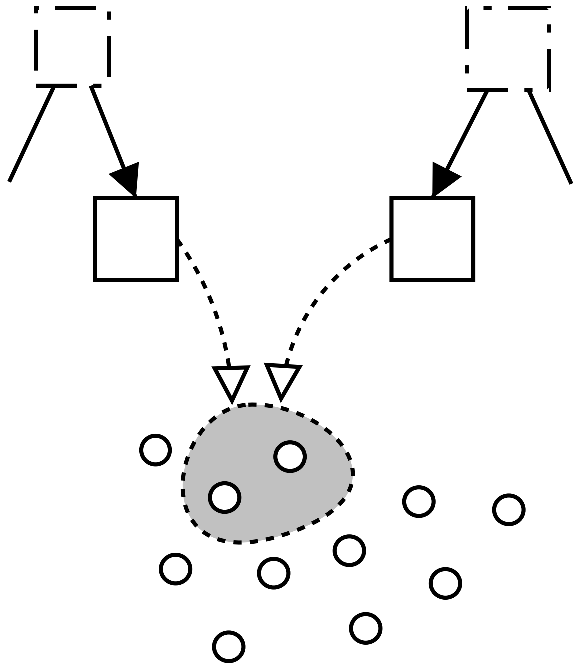
dokumenttijoukkoa. Aktiivisen solmun siirto alaspäin puussa tarkentaa ontologian kuvaamaa dokumenttijoukkoa (kuvat 6 ja 7). Aktiivista dokumenttijoukkoa voi tarkentaa edelleen (kuva 8), mutta aktiivisten solmujen kuvaamien dokumenttijoukkojen leikkaus voi myös jäädä tyhjäksi (kuva 9). Tyhjiä dokumenttijoukkoja antaviin tiloihin siirtyminen olisi mukavaa estää käyttöliittymätasolla piiloittamalla tällaiset siirtymät faseteissa.

Koska fasetit ovat puurakenteita, niissä voi myös siirtyä muulla tavalla kuin pelkästään ylöspäin ja alaspäin. Käyttäjää kiinnostavat mm. ”sisarustilat” (esim. kuvat 6 ja 7) sijaitsevat samalla tasolla. Itse asiassa navigointi voi käyttöliittymätasolla olla hyvinkin vapaata, eli käyttäjä voi mistä tahansa solmusta siirtyä mihin tahansa toiseen solmuun. Fasettipuiden lehdissä löytyvät itse dokumentit, tai ainakin yksittäiset linkin niihin.

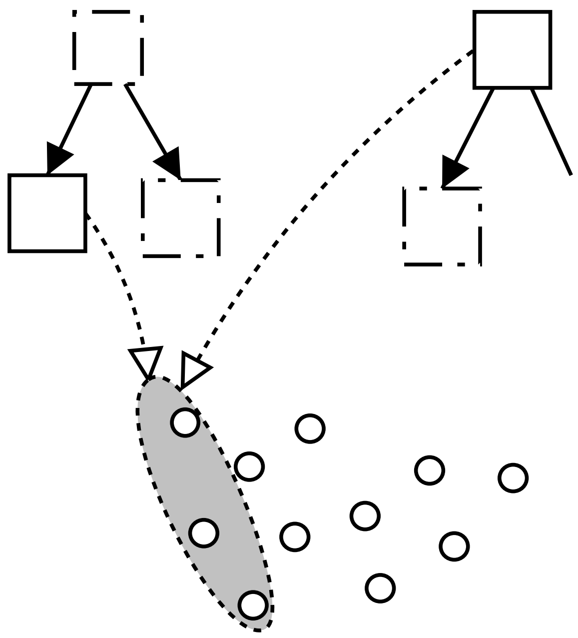
Yksi potentiaalinen ongelma HiBrowse-mallissa on ontologiassa olevien solmujen moniperinnän salliminen. Tämä tarkoittaa että jokin ontologiassa oleva solmu voi kuulua moneen fasettiin (solmu X kuvassa 10), joka voisi muodostua ongelmaksi liikuttaessa fasetissa ylöspäin, puun juurta kohti. Mikäli eteen tulisi tällöin kaksi mahdollista isäsolmua, on valittava se isäsolmu joka kuuluu siihen faset-



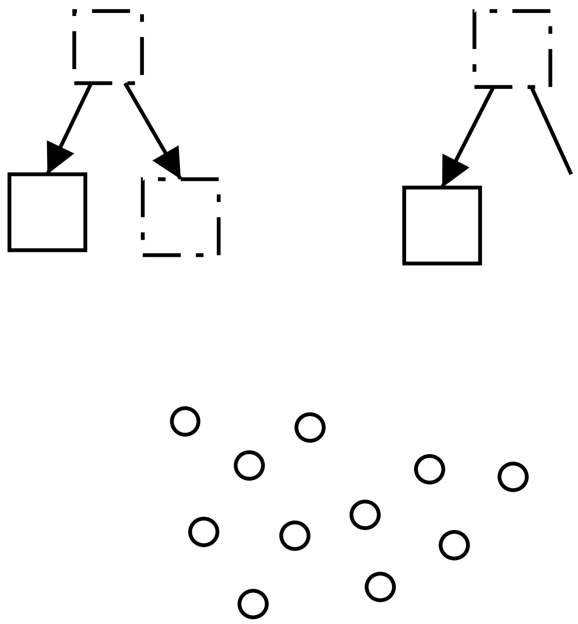
Kuva 6: Aktiivisen dokumenttijoukon tarkennus.



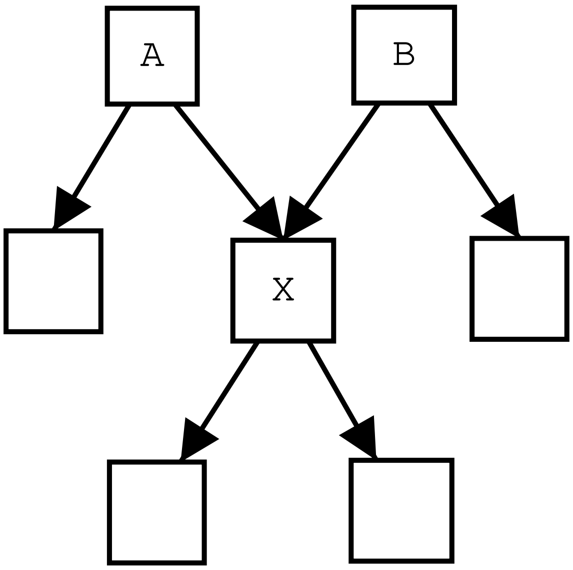
Kuva 7: Aktiivisen dokumenttijoukon tarkennus.



Kuva 8: Aktiivisen dokumenttijoukon muuttuminen kun fasetissa siirretään aktiivinen solmu kahdella siirrolla sisarukseensa (vrt. kuva 7).



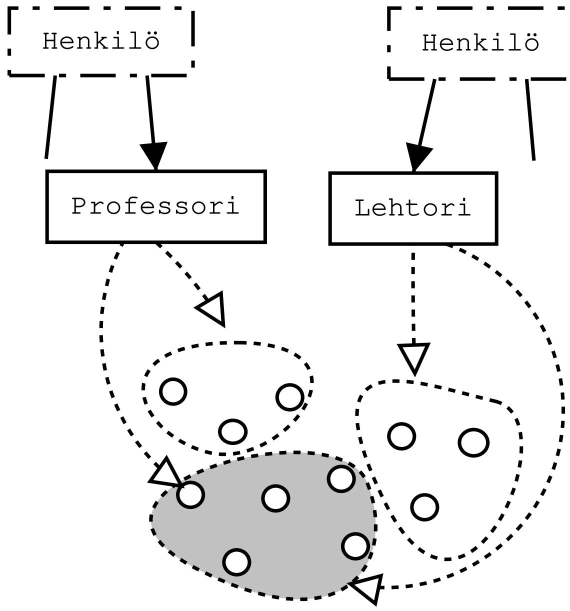
Kuva 9: Tyhjä aktiivinen dokumenttijoukko.



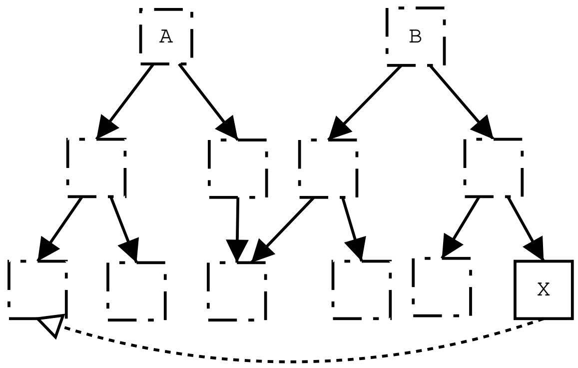
Kuva 10: Moniperintä ontologiassa, jossa on moniperivä solmu X ja kaksi fasettia, A ja B.

tiin jossa liikutaan. Tarvittaessa moniperivä solmu voidaan duplikoida kaikkiin fasetteihin johon se kuuluu. Tämä tarkoittaa että moniperintää fasetin sisällä ei sallita. Ongelmiksi jäävät vielä kahteen fasettiin kuuluvien solmujen mahdollisesti aiheuttamat suodatuksen ristiriidat jotka voivat selaillessa yllättäen johtaa tyhjään tai muuten odottamattomaan dokumenttijoukkoon. Nämä ristiriidat on mahdollisuuksien mukaan havaittava, ja siirto tällaisiin tiloihin on mukava estää joko järjestelmä- tai käyttöliittymätasolla. Syklejä ontologiassa ei saa esiintyä.

Toinen ongelma on kiinteiden fasettien asettamat rajoitukset. Mikäli henkilöfasetteja olisi vain yksi, suodatukseen voisi valita vain professorin tai lehtorin, muttei molempia. Ratkaisu tähän on toteuttaa fasetit fasettiluokkina (esim. henkilöfasettiluokka) joista käyttäjä voi luoda useamman ilmentymän. Luokan tilaan tulisi kuulua sen ilmentymien tilojen aktiivisten dokumenttien yhdiste tai leikkaus. Nyt voidaan luoda kaksi henkilöfasettiilmentymää, joista toinen suodattaa professorin ja toinen lehtorin, ja henkilöfasettiluokan suodatus on siis 'professori ja lehtori' (Kuva 11).



Kuva 11: Fasettiluokan suodatus ilmentymäfasettien suodatusten leikkauksena (tummataustaiset dokumentit). Vaaleataustaiset dokumentit jäävät leikkauksen ulkopuolelle.



Kuva 12: "Mielenkiintoinen"ominaisuuslinkki (X:llä on ominaisuus Y) ontologiassa.

4.2 Mielenkiintoiset linkit

Voidaan katsoa, että johonkin tarkasteltavissa olevaan dokumenttiin liittyy aina tietty määrä muita, aiheeltaan samoja tai samantyyppisiä dokumentteja, tai dokumentteja joiden aihe esiintyy tarkasteltavissa olevassa dokumentissa. Näitten dokumenttien löytäminen selailemalla voi olla hankalaa, varsinkin jos niitten olemassaolosta ei tiedetä mitään. HiBrowse-mallin tueksi voidaankin esittää jokaisen dokumentin yhteydessä ns. *mielenkiintoisia linkkejä* muihin dokumentteihin. RDF-standardin mukaan ontologioissa voi solmujen välillä esiintyä paitsi periytymistä myös vapaamuotoisempia ominaisuussuhteita joita voi käyttää hyväksi. Esimerkkinä jollakin yliopiston rakennuksella voi olla talonmies. Kiinnostavuusyhteys voidaan siis määritellä näitten suhteiden mukaan, ja linkit tehdään lisättäessä dokumentteja ontologiaan.

On päätettävä, sallitaanko linkkejä ontologian aktiivisen tilan ulkopuolella oleviin dokumentteihin vai ei. Mikäli tällaiset linkit sallitaan, linkin takana olevat kuvat eivät välttämättä kuulu yksiselitteisesti mihinkään HiBrowse-mallin tilaan ja sen kanssa synkronisointi järkevällä ja johdonmukaisella tavalla on miltei mahdotonta. Tämän projektin puitteissa tällaiset vapaat linkit sallitaan ja synkronisointia ei edes yritetä toteuttaa.

Mikäli ontologiat ovat suuria on kymmenien tai jopa satojen linkkien näyttäminen käyttäjälle aivan turhaa, ja siksi näitä linkkejä on luokiteltava tai painotettava jollakin tavalla paremmuusjärjestykseen niin, että käyttäjälle näytetään vain esim. korkeintaan kymmenen parasta linkkiä. Luokitteluun on ontologioiden koosta aiheutuen luontevinta käyttää tekoälymenetelmiä manuaalisen luokittelun sijasta. Myös yhteisösuodatusmalli saattaa olla käyttökelpoinen, eli käyttäjät saisivat äänestää hyviä linkkejä, ja paljon ääniä saaneet linkit voitaisiin katsoa erityisen mielenkiintoisiksi. Samoin eniten seuratut linkit voitaisiin katsoa muita mielenkiintoisiksi.

4.3 Sidokset käyttöliittymään

Luonteva yhteys käyttöliittymään olisi kaksijakoinen käyttöliittymämalli, jonka yhdessä osassa näytettäisiin HiBrowse-mallin mukainen fasettiselain ja toisessa osassa aktiivinen dokumenttijoukko ja sille määritellyt mielenkiintoiset linkit. Tilanmuutos selainosassa muuttaisi dokumenttiosaa vastaamaan selaimen aktiivista tilaa ja mielenkiintoisen linkin seuraaminen dokumenttiosassa muuttaisi vastaavasti selainosaa, mikäli tällainen synkronointi katsotaan mieluisaksi tehdä. Käyttäjälle myös oma polun seuraaminen taaksepäin olisi hyödyllinen ominaisuus. Tällöin vierailluista tiloista voitaisiin muodostaa linkitetty lista jossa voidaan siirtyä sekä taaksepäin että eteenpäin. Mikäli käyttäjä peruuttaa ja valitsee jonkun muun tilan kuin se mikä olisi ollut listassa seuraavana, listassa kaikki aktiivista tilaa seuraavat tilat tyhjennetään ennen uutta siirtymää. Toiminto on tällöin oleellisesti samanlainen kuin web-selaimissa käytetty back/forward-logiikka. Tässä projektissa oman polun seuraaminen jää lisäominaisuudeksi joka toteutetaan jos aikaa riittää.

5 Aineisto (EM)

5.1 Aineiston kuvaus

Aikaisemmin mainittuja tekniikoita sovelletaan projektissa Helsingin Yliopiston museon valokuva-arkiston aineistoon. Viidentoistatuhannen kuvan tietokannasta on eroteltu 509 teemaan "promootio"liittyvää kuvaa, joista virtuaalinäyttely koostuu. Varsinaisten kuvien lisäksi aineistoon kuuluvat myös kuvien arkistokortit.

Käytettävät kuvat on tallennettu JPG-muotoisiin kuvanumeron mukaan nimettyihin tiedostoihin. Alunperin MOPSI-järjestelmässä (Yliopiston museon kuvatietokanta, New Generation Software 1993) olleet kortit on käännetty projektia varten XML:ksi (esimerkkikortti: Liite 1). Käännös toteutettiin Mopsi-järjestelmän tiedostoon tulostus ja tulostuspohjaominaisuuksia käyttäen. (tulostuspohja: Liite 3)

5.2 Ontologioihin sijoittelu

Aineistoon ei ole eksplisiittisesti talletettu HiBrowse-selailun (luku 4.1) vaatimia tietoja kuvan sisällön sijoittumisesta eri ontologioihin. Mielenkiintoisten linkkien (luku 4.2) luomiseen käytettäviä suhdeominaisuuksiakaan ei alkuperäistietokannassa ole talletettu erikseen, vaan käytännöllisesti katsoen lähes kaikki kuvasisältöjen sijoituksessa olennainen informaatio on pakattu kahteen kuvakortin kenttään, vapaan tekstin osioon ja viitesanoihin.

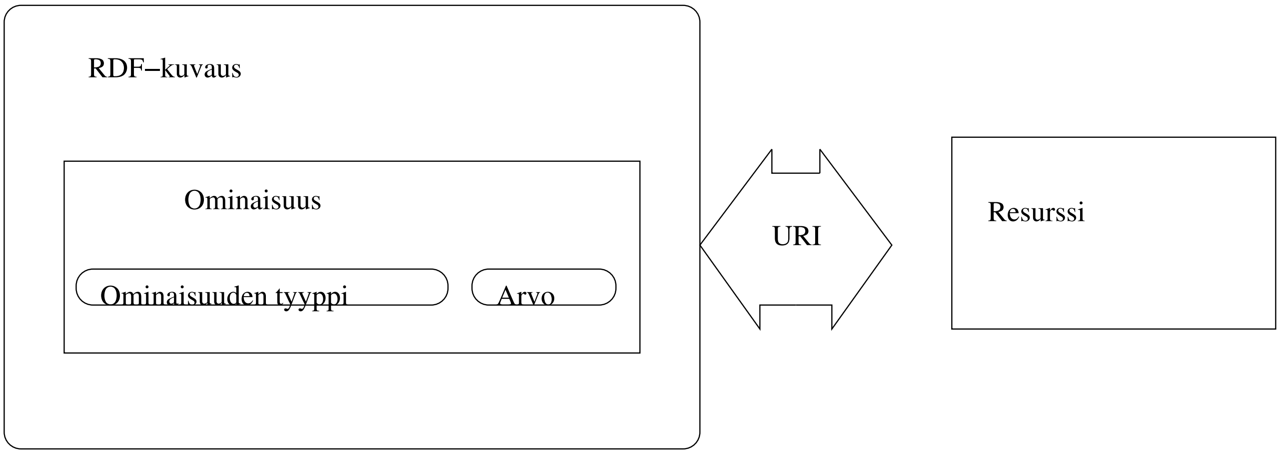
Tarvittavat ontologiasijoitukset ja ominaisuudet on siis pakko lisätä kortteihin selailuohjelmistoa varten. Tehtävään on mahdollista kehittää apuohjelmisto, joka tulee mitä luultavimmin perustumaan kuvakorttien tekstien ja ontologioiden käsitenimien sekä erikseen määriteltyjen eri ontologiakäsitteiden instanssinimien välillä tehtäviin tekstivertailuihin, joiden pohjalta kuvan mahdollisia sijoituksia fasetteihin esitetään käyttäjän varmistettaviksi. Sen suunnitteleminen ja kehittäminen ei kuitenkaan kuulu tämän aliprojektin tehtäviin. Projektissa tarvittavat sijoitukset ja ominaisuudet saadaan projektin ulkopuolelta tai tarpeen vaatiessa määritellään käsin.

6 Tärkeitä välineitä ja käsitteitä

6.1 RDF (TR)

Resource Description Framework (RDF) on XML-pohjainen kieli kuvausten (eng. description) luomiseen. Se on kieli informaation tulkitsemiseen ja olioiden välisten suhteiden luomiseen [Hje01] RDF-kieltä käytetään resurssien (eng. resources), kuten WWW-sivujen, kuvaamiseen XML-tekstien pohjalta. Mistä tahansa resurssista, jolle voidaan määrätä URI (Uniform Resource Identifier, eräänlainen globaali nimi), voidaan laatia RDF-kuvaus.Kuva 13 on R. Ianellan käsitys RDF-kuvauksesta [Ian99] ja alla oleva koodi on esimerkki RDF:n käytöstä [Ian99].

```
<? xml version="1.0"?>
<RDF xmlns= "http://w3.org/TR/1999/PR-rdf-syntax-1999-19990105#"
```



Kuva 13: RDF-kuvaus

xmlns:DC= "http://purl.org/DC#»
<Description about = "http://dctc.com.au/report.html»
<DC:Title> The Future of Metadata </DC:Title>
<DC:Creator> Jacky Crystal</DC:Creator>
<DC:Date>1998-01-01</DC:Date>
<DC:Subject>Metadata, RDF, Dublin Core</DC:Subject>
</Description>
</RDF>

Ensimmäinen rivi ilmaisee, että kyseessä on XML-dokumentti. Kaikki RDF-dokumentit ovat aina myös XML-dokumentteja. Seuraavat kaksi riviä määrittelevät kaksi nimiavaruutta, RDF-nimiavaruuden, joka tulee kaikkiin RDF- ja RDFS-dokumentteihin (RDFS käsitellään seuraavassa aliluvussa) ja DC-nimiavaruuden. DC- eli Dublin Core, nimiavaruus on kirjastokäyttöön kehitetty kielioppi. Description-tagien välissä tekstissä kuvataan neljällä attribuutilla about-attribuutin sisältämää resurssia, joka yksilöidään URI:lla "http://dctc.com.au/report.html".

6.2 RDFS (TR)

RDFS eli Resource Description Framework Schema, on kieli resursseihin liittyvien ominaisuuksien kuvaamiseen. Esimerkkejä ominaisuuksista resurssille 'kirja' voisivat olla 'tekijä', 'nimi' ja 'aihe' [BrG00]. RDFS toimii hyvin samaan tapaan kuin XML-kielen DTD:t (Document Type Definition), sillä erotuksella että RDFS tarjoaa tiedon tulkintatavasta RDF-dokumentissa [BrG00]. Toisin kuin yleensä XML-dokumenttien validoinnissa, ei RDF-dokumenttien validoimiseen riitä pelkkä syntaktinen skeema: tarvitaan myös niiden rajoitteiden, jotka seuraavat RDF:llä mallinnetusta resurssista/resursseista, määritteleminen. RDF-skeemassa ei myöskään määritellä primitiivisiä tietotyyppejä kuten (merkkijono, kokonaisluku jne.) XML:n DTD:ssä. Tietotyyppien määrittely RDFS:ssä perustuu luokkien (classes), ominaisuuksien (properties), rajoiteominaisuuksien (constraint properties) käyttöön. Esimerkiksi resurssit ovat ilmentymiä luokista. Se, minkä luokan ilmentymiä ne ovat ilmaistaan ominaisuudella *rdf:type*. Luokat itsessään noudattavat hierarkista periytymistä: esimerkiksi koira on nisäkkään aliluokka, joka taas on eläimen aliluokka [BrG00]. Tätä periytymistä kuvataan ominaisuudella *rdfs:subClassOf*. Erona Java-kielen kaltaisiin luokkamäärittelyihin on se, että ominaisuuksissa itsessään määritellään se, mihin luokkiin niitä voidaan soveltaa. Tähän tarkoitukseen tarvitaan rajoitteita *rdfs:domain* ja *rdfs:range*. Rajoitteista ensimmäistä käytetään ilmaisemaan mitä arvoja ominaisuus voi saada, jälkimmäistä taas siihen, missä luokissa ominaisuutta voi käyttää.

6.3 JSP (PL)

JSP eli Java Server Pages on tekniikka, jolla voidaan luoda serverin päässä dynaamisia web-sivustoja. Tyypillisesti JSP:tä kirjoitettaessa javakoodi on kirjoitettu HTML:n sekaan. JSP:n versio 1.1 mahdollistaa custom tagien käytön, joiden avulla javakoodi voidaan kirjoittaa omaan tiedostoonsa (JSP tag library) ja HTML:n sekaan kirjoitetaan vain XML-tageja muistuttavia JSP-tageja.

6.4 Java Applet (PL)

Java Applet on asiakas päässä pyörivä java-ohjelma, jolla voidaan lisätä toiminnallisuutta web-sivustolle. Applet ei kuitenkaan voi suorittaa kaikkia tavallisen java sovelluksen toimintoja. Java-applet lisätään HTML-sivulle applet-tagilla, jolla voi olla alielementteinä param-tageja.

6.5 Lyhyitä määritelmiä

Seuraavassa taulukoituna lyhyitä määritelmiä käsitteistä ja työkaluista, joita voidaan käyttää hyödyksi projektin toteutuksessa.

Termi:	Määrittely:
CSS	Cascading Style Sheets. Merkkauskieli, jolla voidaan määritellä esim. HTML dokumenttien ulkoasu. Tarkempaa tietoa osoitteessa www.w3.org/Style/CSS/.
DBXML	XML-tietokanta. Tarkempaa tietoa osoitteesta www.dbxml.org/.
Fasetti	Hakukäsite. Muodostuu saman aihealueen sisällä samaa aspektia kuvaavista asioista tai niitä tarkoittavista käsitteistä. Niiden käsitteiden joukko, joita voidaan pitää haussa vaihtoehtoisina. Fasetti on siis käsiteryväs.
HTML	Hypertext Markup Language. WWW-sivujen kuvaamiseen käytettävä kieli. Tarkempi kuvaus osoitteessa www.w3.org/TR/html4/.
JavaBeans	Teknologia, jolla voidaan luoda ohjelmakomponentteja Java-ympäristössä. Katso lisää osoitteesta http://java.sun.com/products/javabeans/docs/.
JavaScript	Selaimen tulkitsema kieli, jolla voidaan lisätä HTML-sivuihin toimintoja, joita pelkkä HTML ei mahdollista.
JDBC	Java-kielen rajapinta, jonka avulla voidaan käsitellä tietokantoja.
Jena	Javalla toteutettu tyäkalu, jolla voidaan käsitellä RDF-malleja. Tarkempi kuvaus osoitteessa www.hpl.hp.com/semweb/jena-top.html.
JRE	Java Runtime Environment. Sunin kehittämä ohjelmointiympäristö. Tarkempaa tietoa osoitteessa http://java.sun.com/.

JSP	JavaServer Pages. Käyttää XML:n kaltaisia tageja kapseloimaan logiikan, joka luo web-sivun. Tarkempaa tietoa osoitteessa http://java.sun.com/products/jsp/.
Ontologia	Tiedon riippuvuussuhteiden esittäminen puumaisena rakenteena. Puun juurena on esim. talo, jonka lapsia ovat ylä- ja alakerta ja lapsenlapsia ullakko ja keittiö.
RDF	Resource Description Framework. Kieli, jolla voidaan yhdistää erilaista XML-muodossa olevaa tietoa toisiinsa ontologioiden avulla. Tarkempaa tietoa osoitteessa www.w3.org/RDF/.
SQL	Structured Query Language. Kieli, jolla voidaan muokata ja käyttää tietokantaa.
XHTML	Extensible Hypertext Markup Language. XML-yhteensopiva HTML-standardia mukaileva WWW-sivujen kuvauskieli.
XML	Extensible markup language. Metakieli rakenteisten dokumenttien esittämiseen. Tarkempaa tietoa osoitteessa www.w3.org/TR/2000/REC-xml-20001006.

Lähteet

BrG00 Brickley, D. , Guva, R., Resource Description Framework (RDF) Schema Specification 1.0, W3C Candidate Recommendation 27 March 2000, www.w3c.org/TR/rdf-schema.

Hje01 Hjelm, J., Creating the Semantic Web with RDF ,Wiley, 2001.

Ian99 Ianotello, R., An Idiots guide to the Resource Description Framework, 1999.

Sow00 Sowa, John F. 2000: Knowledge Representation - Logical, Philosophical, and Computational Foundations, Brooks/Cole, USA.

Hib02 The key role of classification and indexing in view-based searching: http://www.hud.ac.uk/schools/cedar/pollifla.html (5.3.2002).

Liite 1. Esimerkki kuvatietokannan kortista XML-muodossa.

<kuva>
<nro>
5680
</nro>
<kuvatiedosto>
kuvat/kuvat/5680.jpg
</kuvatiedosto>
<kuvausaihe>
Professori Eskola antaa kuningas Kustaa VI Aadolfille kunniatohtorin hatun filosofisen tiedekunnan promootiossa 1952. Korokkeella edessä kukkakimput kädessään (vasemmalta) tasavallan presidentti J.K. Paasikivi, Ruotsin kuningatar ja rouva Alli Paasikivi, jonka takana ilmeisesti K.A. Fagerholm rouvansa kanssa. Pedellinä Gunnar Strömberg, ylivahtimestarina Långholm.
</kuvausaihe>
<viitesanat>
Juhla, promootio 25.5.1952, filosofinen tiedekunta, Kustaa VI Aadolf,
kunniatohtori, 1950luku
</viitesanat>
<kuvatyyppi>
vk
</kuvatyyppi>
<vari>
mv
</vari>
<vedoslkm>
1
</vedoslkm>
<kuvakoko>
18x22
</kuvakoko>
<vaakapysty>
v
</vaakapysty>
<kehys>

ei
</kehys>
<negakoko>
-
</negakoko>
<lasifilmi>
-
</lasifilmi>
<origrepro>
-
</origrepro>
<kuvaamo>
-
</kuvaamo>
<kuvaaja>
-
</kuvaaja>
<kuvausaika>
1952
</kuvausaika>
<kayttooikeudet>
on
</kayttooikeudet>
<sijoitus>
Juhlat ja tilaisuudet, promootiot, filosofinen tiedekunta, 1952
</sijoitus>
<hankinta>
Lahjoitus
</hankinta>
<kuvasarja>
-
</kuvasarja>
<luovuttaja>
Geologian laitos
</luovuttaja>
<kunto>

-
</kunto>
<muuttiedot>
-
</muuttiedot>
<luetteloija>
ke
</luetteloija> <luettelointiaika>
19921001
</luettelointiaika>
</kuva>

Liite 2. Alkuperäinen tehtävänanto

SEMANTIC WEB:
YLIOPISTON MUSEON VIRTUAALINÄYTTELY

Tutkimusalue:

Tekoäly, WWW-teknologiat, informaatiojärjestelmät.

Asiakkaat:

Eero Hyvönen (Tietojenkäsittelytieteen laitos) ja Kati Heinämies (Helsingin yliopiston museo).

Ohjaaja:

Matti Nykänen

Työn aihepiiri:

Helsingin yliopiston museolla on laaja historiallinen kuvarkisto. Museo tulee saamaan uudet upeat tilat maamme paraatipaikalta Senaatintorilta ja näyttelytoimintaa ollaan jatkossa voimakkaasti kehittämässä. Tähän liittyen museo on ideoimassa yhdessä tietojenkäsittelytieteen laitoksen kanssa uusimman tietotekniikan soveltamismahdollisuuksia kokoelmien hallintaan ja näyttelytoimintaan liittyen.

Työn kuvaus:

Työn tavoitteena on kehittää Yliopiston museon kuvatietokantaan perustuva WWW-näyttely. Tutkimuksellisena haasteena on selvittää uusien Semantic Web teknologioiden soveltamismahdollisuuksia. Erityisen kiinnostuksen kohteena on tutkia, miten kuvamateriaalien sisällöt voitaisiin kuvata (semanttiset ontologiat), millaisilla semanttisella hakemistostoilla, kuten aihekartoilla (Topic Maps) loppukäyttäjille voitaisiin tarjota näitä kiinnostavia näkymiä kokoelmiin (esimerkiksi promootio-teema) ja miten järjestelmä voitaisiin toteuttaa uusilla RDF(S)-perustaisilla tietokantaratkaisuilla.

Tulokset:

Konkreettisena lopputuloksena syntyy tavalliselle museokävijälle tarkoitettu selainpohjainen "älykäs"käyttöliittymä kokoelmatietoihin ja (osallistujamäärästä riippuen) ontologiaperustainen luetteloinnin apuväline, jonka avulla kokoelmatietoihin voidaan liittää semanttisia metakuvauksia. Hanke sivuaa laitoksella ja HIIT:ssä vireillä olevaa laajempaa tutkimushanketta Semantic Web -teemaan liittyen.

Toteutusympäristö:

Yliopiston museon tietokantajärjestelmä.

Erityisvaatimukset:

Osallistujilta edellytetään hankkeen luonteesta johtuen vahvaa ohjelmointitaustaa ja perustietoja XML-teknologioista, tietokannoista ja tekoälystä. Hankkeelle pyritään saamaan jatkossa rahoitusta ja tarjoamaan osallistujille aihepiiriin liittyviä opinnäytetöiden aiheita.

Liite 3. XML-tulostuspohja Mopsi-järjestelmään.

```
<kuva>
<nro>  %<nro>
</nro>
<kuvatiedosto>
%<kuva>
</kuvatiedosto>
<kuvausaihe>
%<kuvausaihe>
</kuvausaihe>
<viitesanat>
%<viitesanat>
</viitesanat>
<kuvatyyppi>
%<kuvatyyppi>
</kuvatyyppi>
<vari>
%<kuvavari>
</vari>
<vedoslkm>
%<vedoslkm>
</vedoslkm>
<kuvakoko>
%<kuvakoko>
</kuvakoko>
<vaakapysty>
%<vaakapysty>
</vaakapysty>
<kehys>
%<kehys>
</kehys>
<negakoko>
%<negakoko>
</negakoko>
<lasifilmi>
```

%<lasifilmi>
</lasifilmi>
<origrepro>
%<origrepro>
</origrepro>
<kuvaamo>
%<kuvaamo>
</kuvaamo>
<kuvaaja>
%<kuvaaja>
</kuvaaja>
<kuvausaika>
%<kuvausaika>
</kuvausaika>
<kayttooikeudet>
%<kayttooikeudet>
</kayttooikeudet>
<sijoitus>
%<sijoitus>
</sijoitus>
<hankinta>
%<hankinta>
</hankinta>
<kuvasarja>
%<kuvasarja>
</kuvasarja>
<luovuttaja>
%<luovuttaja>
</luovuttaja>
<kunto>
%<kunto>
</kunto>
<muuttiedot>
%<muuttiedot>
</muuttiedot>
<luetteloija>

%<luetteloija>
</luetteloija>
<luettelointiaika>
%<luettelointiaika>
</luettelointiaika>
</kuva>