

Merkityksellinen ikkunoiden yhdistely

Tuukka Hastrup
Tuukka.Hastrup@iki.fi

28.2.2006

Tiivistelmä

Ikkunointijärjestelmiä voidaan kehittää ikkunoiden yhdistelyn visuaalisista tehosteista edelleen sisällön merkitykset huomioiviksi Semantic Desktop -tutkimuksen ja Focus+Context-esitystavan yhdistelmänä. Tavoitteena on olemassaolevien sovellusten kanssa yhteensopivasti parantaa järjestelmän käytettävyyttä ja helpottaa sovellusten toteuttamista osissa. Toteutus pohjautuu Xgl:ään ja Compiziin jälkimmäiseen tehtävänä liitännäisenä, joka toteuttaa RDF-tietovarannon sovelluksia ja yhdistelytehostetta varten. Käytettävyyden arvioimiseksi esitellään koeasetelma.

1 Johdanto

Graafiset käyttöliittymät perustuvat yleisesti perinteisiin ikkunointijärjestelmiin, joissa sovellusohjelmilla on näytöllä käytettävissään erillisiä suorakulmaisia alueita.

Laitteistojen teho ja ikkunointijärjestelmien kehittyminen ovat viime aikoina mahdollistaneet *ikkunoiden yhdistelyn* eli sen, että sovellusten tuotama ikkunoiden sisältö käsitellään ja yhdistellään ikkunointijärjestelmässä mielivaltaisesti ennen näytöllä esittämistä. Tämän sovelluksina on nähty ensin ikkunoiden läpinäkyvät alueet, seuraavaksi läpikuultavat ikkunat ja viimeisimpänä ikkunoiden reaaliaikaiset muodonmuutokset.

Tässä raportissa esitellään eräs idea alkuperäisen ikkunamallin kehittämiseksi niin, että ikkunoiden yhdistely voi perustua visuaalisten tehosteiden lisäksi ikkunoissa näkyvien asioiden merkityksiin. Idea pohjautuu toisaalta Semantic Web (merkitysten verkko) ja edelleen Semantic Desktop (merkitysten työpöytä) -tutkimukseen, toisaalta hypertekstiin ja sen Focus+Context

(kohde ja asiayhteys) -esitystapaan. Tarkoituksena on olemassaolevien sovellusten kanssa yhteensopivasti sekä parantaa järjestelmän käytettävyyttä että helpottaa sovellusten toteuttamista osissa.

Seuraavassa luvussa esitellään aiempi tutkimus ja teoria, kolmannessa aiemmat toteutukset ja nyt kehitettävä ohjelmisto, sekä neljännessä luvussa toimivuuden käytettävyystutkimus.

2 Teoria

2.1 Semantic Desktop

Semantic Web [1] (merkitysten verkko) on idea internetin nykyisen hyper-tekstin kehittämiseksi niin, että tiedot olisivat koneisesti käsiteltävissä. Tämän tavoitteen saavuttamiseksi on kehitetty RDF-malli hajautettuun tiedon esittämiseen sekä sen XML-dokumenttien merkitsemiseen soveltuva syntaksi. RDF-mallin suunnittelussa on otettu huomioon, että eri ohjelmistot voivat käsitellä samaa tietoa eri merkitystasoilla ja sekä tieto että merkitykset kehittyvät hajautetusti ilman koordinoitumahdollisuutta. Matalimmalla tasolla näkyvät asiat ja niiden yhteydet.

Semantic Desktop [5] (merkitysten työpöytä) tuo idean uudelle alueelle, henkilökohtaiseen tietojenkäsittelyyn. Semantic Webin tavoitteet ja tekniikat soveltuvat pitkälti myös sovellusohjelmiin, kun toivotaan niiden toimivan yhteen ja koordinoitumahdollisuus puuttuu. Dokumenteille ja niiden osille, käyttäjää kiinnostaville asioille voidaan antaa URI:t (Uniform Resource Identifier, yhdenmukainen resurssin tunniste), jolloin sovellukset voivat käsitellä asioiden yhteyksiä ja merkityksiä.

2.2 Ikkunointiarkkitehtuuri

Graafisten käyttöliittymien arkkitehtuuri on pitkälti säilynyt samana ensimmäisistä toteutuksista lähtien. Käyttöjärjestelmän prosessit voivat asiakas-palvelin-mallin mukaisesti [2, s. 136–138] viestiä ikkunointijärjestelmän kanssa, joka varaa niille suorakulmaisen alueen, ikkunan, näytöltä graafisen käyttöliittymän esittämiseen. Ikkunoihin kohdistuvan syötteen, näppäimistöltä tai hiirestä, ikkunointijärjestelmä ohjaa prosesseille. Muu viestintä sovellus-ohjelmien välillä tapahtuu pääasiassa ikkunointijärjestelmästä riippumatta.

Ikkunointijärjestelmä tarjoaa integrointiominaisuutena sovellukselle mahdollisuuden nielaista (*swallow*) toinen sovellus osaksi omaa ikkunaansa. Tämä mahdollistaa lähinnä selaimen videoliitännäiset. Monimutkaisemmat integroinnit toteutetaan nykyään prosessinsisäisesti.

Ikkunoiden yhdistely [4] ei muuta viestintää muuten, mutta ikkunat voivat näkyä käyttäjälle käsiteltyinä. Ikkunoiden sisältö ei ole suoraan näytöllä vaan puskurissa, josta ikkunointijärjestelmä käyttää sisältöjä näyttöä piirtäessään, esimerkiksi projisoi niitä eri muotoihin.

2.3 Focus+Context

Hypertekstitutkimuksen kohteena on asioiden esittäminen osina ja niiden välisinä yhteyksinä, alunperin tekstikappaleina ja linkkeinä niiden välillä, myöhemmin myös multimediana hypermediassa.

Focus+Context (kohde ja asiayhteys) -esitystapa hypermedialle perustuu siihen, että kullakin hetkellä tietty asia on esittämisen kohteena ja yhteyksien avulla esitetään asiayhteyttä.

3 Ohjelmointiosuus

3.1 Xgl ja Compiz

Nykyisin useimmilla laitteistoilla ja useimmissa käyttöjärjestelmissä käytetty ikkunointijärjestelmästandardi on X Window System. Standardi määrittelee ikkunointijärjestelmän arkkitehtuurin ja viestintämuodon sovellusohjelmien ja ikkunointijärjestelmän välille. Yksi sovellusohjelma toimii ikkunanhallintaohjelmana (*window manager*), jolloin sen tehtävänä on tarjota käyttöliittymä ikkunoiden siirtämiseen, koon muuttamiseen ja sulkemiseen. Standardista on vapaaohjelmistototeutus X.Org, jota käytetään yleisesti varsinkin avoimissa käyttöjärjestelmissä.

Ikkunoiden yhdistelyä varten standardissa on laajennos Composite (yhdistelmä), joka mahdollistaa yhden sovellusohjelman toimia yhdistelynhallintaohjelmana (*compositing manager*). Yhdistelynhallintaohjelma voi siirtää ikkunat näytöltä puskurimuistiin ja piirtää näytön käyttämällä näitä ikkunoita kuvina.

X.Orgissa on vaihtoehtona kokeellinen toteutus Xgl, joka sekä toteuttaa yhdistelylaajennoksen että mahdollistaa ikkunoiden käytön tekstuureina OpenGL:llä toteutetussa yhdistelyssä [3]. Helmikuussa 2006 julkaistiin yhdistetty ikkunan- ja yhdistelynhallintaohjelma Compiz, joka Xgl:n kanssa käytettynä pystyy monenlaisiin graafisiin tehosteisiin. Sovellusohjelmat viestivät ikkunointijärjestelmän kanssa kuten ennenkin, joten niihin ei tarvita muutoksia.

3.2 Ohjelmiston rakenne

Merkityksellinen ikkunoiden yhdistely on tässä työssä tarkoitus toteuttaa Compizin liitännäisenä (*plugin*). Suunniteltu liitännäinen toteuttaa Focus+Contex-näkymän ikkunoille ja niiden välisille merkityksille. Merkitykset saadaan sovellusohjelmilta RDF-muodossa. Viestintä sovellusohjelmien ja liitännäisen välillä toteutetaan TCP/IP:llä. Sovellusohjelmia täytyy joko muokata tai niihin toteuttaa laajennoksia (*extension*), niin että sovellusohjelmat ilmoittavat merkityksistä.

Liitännäisen yksi tärkeä osa vastaanottaa RDF-tiedot ja säilyttää ne tietovarastoarkkitehtuurin mukaisesti [2, s. 145–146] muodossa, jossa tarvittavat haut saadaan toteutettua tehokkaasti. Toinen tärkeä osa on yhdistelytehoste. Aktiivinen ikkuna voidaan näyttää keskellä näyttöä suurena. RDF-tietovarastosta voidaan hakea siihen liittyviä muita ikkunoita ja ne voidaan näyttää pienempinä ympärillä.

Yhdistelyefekti voidaan aktivoida tietyllä näppäinyhdistelmällä, jolloin perinteinen ikkunoiden järjestys on edelleen käytettävissä mutta uusi järjestys saadaan käyttöön haluttaessa.

4 Kokeellinen osuus

Ohjelmointiosuuden perusteella nähdään, millaisia ongelmia merkityksellisen ikkunoiden yhdistelyn toteuttamisessa voi olla. Itse ideaa sen sijaan täytyy testata kokeellisesti, jotta voitaisiin todeta siitä saatavat hyödyt käytettävyydessä.

Eräässä yleisesti käytettävyyden testaamisessa käytetyssä koeasetelmasa mitataan aika, joka käyttäjältä kuluu tietyn selaustehtävän suorittamiseen selaimessa. Jos ohjelmointiosuudessa saadaan toteutettua laajennos esimerkiksi Mozilla Firefoxiin, voitaisiin kokeessa testata erikseen selausaika merkityksellisellä ikkunoiden yhdistelyllä sekä ilman sitä. Vertailtavana voi olla myös välilehtien käyttö.

Jotta mahdolliset erot käytettävyydessä tulisivat esille, koetehtävän tulee olla sellainen, että sen tehokkaassa suorittamisessa tarvitaan useampaa ikkunaa. Esimerkiksi voidaan pyytää selaamalla löytämään ja vertailemaan tietyn sivuston erilaisia sivuja.

5 Yhteenveto

Raportissa kuvattiin idea ikkunanhallintajärjestelmän kehittämiseksi niin, että ikkunoiden yhdistelyä voidaan tehdä ikkunoiden sisältöjen merkitysten

perusteella. Aiheen lähtökohtana esiteltiin Semantic Desktop, ikkunointijärjestelmien arkkitehtuuri sekä hypermedian Focus+Context-esitystapa. Toteutusympäristöstä esiteltiin X.Org sekä siihen ikkunan- ja yhdistelyhallintaohjelma Compiz. Toteutus kuvattiin Compizin liitännäisenä, jonka tehtävät jakautuvat RDF-tietovaraston ylläpitoon ja yhdistelytehosteeseen. Merkityksellistä ikkunoiden yhdistelyä voidaan arvioida lopuksi kuvatulla koejärjestelyllä.

Tulevaisuudessa voidaan tutkia millaisia merkityksiä sovellusohjelmista saadaan ja miten sekä merkitysten hyödyllisiä esitystapoja yhdistelyssä. Kuvattu järjestelmä sopisi myös siihen, että sovellusohjelmat voisivat viestiä haluamaansa yhdistelyn ulkoasua niiltä osin kuin ikkunaan piirtäminen ei riitä.

Viitteet

- [1] Tim Berners-Lee, James Hendler ja Ora Lassila, *The Semantic Web*, Scientific American, May 2001.
- [2] Kai Koskimies ja Tommi Mikkonen, *Ohjelmistoarkkitehtuurit*, Talentum, Helsinki, 2005.
- [3] Peter Nilsson ja David Reveman, *Glitz: Hardware Accelerated Image Compositing using OpenGL*, Proceedings of the 2004 USENIX Annual Technical Conference, saatavilla WWW-muodossa <URL: <http://www.usenix.org/publications/library/proceedings/usenix04/tech/>>, 10.6.2004.
- [4] Thomas Porter ja Tom Duff, *Compositing digital images*, SIGGRAPH '84: Proceedings of the 11th Annual Conference on Computer Graphics and interactive Techniques (toim. H. Christiansen), ACM Press, New York, NY, 1984, s. 253-259.
- [5] Leo Sauermann, *The Semantic Desktop - a Basis for Personal Knowledge Management*, Proceedings of the I-KNOW 2005, 5th International Conference on Knowledge Management.