

# “LyX Branches” – avoimen lähdekoodin ja opetusmateriaalin kehittämisen keissi

Martin Vermeer

29. joulukuuta 2003

## 1 Ongelman kuvaus

### 1.1 Monikielisyys opetuksessa

Suomi on kaksikielinen maa. TKK on tosiasiallisesti monikielinen yliopisto.

Minulle on toistuvasti tullut vastaan tarve toimittaa tenttikysymykset sekä suomen- että ruotsinkielisinä, joskus myös englanninkielisinä. Myös luentomonisteita ja muuta opetusmateriaalia kaivataan joskus erikielisinä: vuonna 2001 minulla oli ensimmäistä kertaa uusi opintojakso Navigoinnin menetelmät, joka kuuluu Paikannus ja navigointi -pääaineeseen. Opiskelijoiden joukossa oli muutama jatko-opiskelija, joista yksi oli ulkomaalainen. Luentomoniste oli kuitenkin vain suomen kielellä.

Vuonna 2001 ratkaisin ongelman sillä tavalla, että annoin opiskelijalle luettavaksi kirjan sivut, jotka sisälsivät suurin piirtein samat asiat. Päätin kuitenkin, että v. 2003, kurssin seuraavana luennointikertana, olisi valmis englanninkielinen versio olemassa. Ja näin olikin... ja taas yksi ulkomaalainen opiskelija. Suomea osaamattomien opiskelijoiden määrä on kurseillani aina pieni, mutta ratkaisu on löydettävä.

Anoin ja sain rahat tähän käännöstyöhön TKK:n Opetuksen kehittämisvaroista.

### 1.2 Käännöstyö

Keväällä 2002 pestasin opiskelijan kääntämään luentomoniste englanniksi. Opiskelija oli viettänyt harjoittelukauden Kanadassa ja osasi aika hyvin englantia, ei kuitenkaan täydellisesti. Jouduin korjailemaan tekstiä jonkin verran, etenkin ammattisanastoa. Ajansäästö oli kuitenkin huomattavaa.

Tämä käännöstyö tehtiin käyttämällä LyX -dokumenttiprosessoria, joka on TKK:n paikallisverkon UNIX- ja Linux-koneissa. Opiskelija työskenteli kotona käyttäen oma koti-PC:tään ja siihen TKK:n kampuslisenssien luvalla asennettua Secure Shelliä sekä UNIX-grafiikkasoftaa (X-palvelinta).

Työssä käytettiin LyXin standardiversiota ja siihen lisättyä “kludge” eli rumasti improvisoitua ratkaisua. Perusidea oli kuitenkin hieno: olisi yksi lähdedokumentti, joka sisältäisi sekä suomen- että englanninkielisen version siten, että vastaavat suomen- ja englanninkieliset tekstifragmentit näkyisivät vierekkäin samassa kuvaruudussa. Muutokset voitaisiin näin tehdä yht’aikaa molemmissa kielissä, jotta välttyttäisiin synkronoinnin ongelmalta kahta erillistä dokumenttia käytettäessä.

Kuvaruudussa maalattiin suomenkielinen teksti siniseksi ja englanninkielinen magentaksi. Tulostus erikseen suomen- tai englanninkieliseksi PDF-dokumentiksi tapahtui varta vasten laaditun L<sup>A</sup>T<sub>E</sub>X-koodipätkän avulla, muuttamalla vain yhden parametrin arvo ykkösestä kakkoseksi.

### 1.3 Uuden ominaisuuden kehittäminen

Tämän kokemuksen rohkaisemana päätin, että LyXiin pitäisi saada aikaan “virallinen” ratkaisu tällaiseen käyttöön, eli kahden eri kieliversion toimittaminen yhdessä dokumentissa. Tietääkseni mitään muu tekstinkäsittelysovellus ei mahdollista tätä. Kutsuin tätä ominaisuutta nimellä “Branches” eli *haaraumat*. Jokainen haarauma voi olla kieli tai esim. teknisen dokumentaation tapauksessa tuotteen eri versioiden hieman eroavat dokumentaatiot. Tai harjoitusten laatimisen tapauksessa harjoitusten vastaukset eli “opettajan versio” voi olla yksi haarauma ja niin edelleen.

Haaraumien käytön toisena etuna on, että ne dokumentin osat, jotka ovat samoja molemmissa kielissä — kaavat, kuvat, taulukot —, kirjoitetaan vain kerran.

## 2 LyXin lyhyt kuvaus

Miljoonat ihmiset tuntevat ns. WYSIWYG-tekstinkäsittelyohjelmistot (“What You See Is What You Get”), kuten Microsoft Wordin, WordPerfectin tai StarOffice Writen. Niiden idea on, että jo kuvaruudussa näkyy, millaiselta teksti tulee näyttämään tulostuksen jälkeen paperilla.

WYSIWYGin hieno puoli on, että se on helppoa oppia ja varmakäyttöinen. Tulos on juuri sitä, mitä se lupaa olla.

Kuitenkin WYSIWIGillä on myös ongelmia ja rajoituksia. “What You See Is *All* You Get”. Siksi kaikki tekstinkäsittelysovellukset sisältävät myös ei-WYSIWYG-ominaisuuksia, kuten mm. itsenumeroivia otsikkotyylejä. Jos käyttää niitä, ei tarvitse koskaan itse numeroida otsikoita ja ne menevät myös automaattisesti sisällysluetteloon. Mainittakoon myös toistuvat sivuotsakkeet — vielä erikseen parillisille ja parittomille sivuille — ja itsenumeroituvat sivut, “pehmeät” tavutusmerkit jne.

Aivan toisenlainen lähestymistapa dokumenttien laatimiseen on *merkintäkieli*, jonka tunnetuin tyyppiesimerkki on www-sivuilla käytetty HTML, Hypertext Mark-up Language. Kuitenkin periaate on paljon vanhempi. Jo ensimmäisissä UNIX-järjestelmissä oli ohjelma nimeltä troff, jonka avulla saa-

tiin sopivilla merkinnöillä varustettu teksti tulostumaan kauniina paperitulos-  
teena. Nuorempi on  $\TeX$ , kalifornialaisprofessori Donald KNUTHIN laatima  
tietokoneladontakieli. Sitä käytetään edelleen hyvin laajasti matemaatispainot-  
teisten tekstien latomiseksi tieteellisiä aikakauslehtejä varten.

$\TeX$ in suosittu variantti on  $\LaTeX$ , Leslie LAMPORTIN laatima  $\TeX$ -makropaketti,  
kauniina erikoisominaisuutena pesäeron tekeminen dokumentin rakenteen ja  
sen visuaalisen ilmeen välillä.

Jos haluat tietää, miksi ihmeessä monet ihmiset käyttävät edelleen  $\LaTeX$ ia kun  
kerran hyviä WYSIWYG-ohjelmia on olemassa, riittää  $\LaTeX$ illa ladotun kir-  
joitelman silmämääräinen vertailu Microsoft Wordilla laadittuun, vastaavaan  
kirjoitelmaan. Varsinkin jos on kyse tieteellisestä tekstistä. Typografinen laatu  
on kuin toiselta planeetalta.

Tämä oli  $\text{LyX}$ in luomisen motivaatio: yhdistää  $\LaTeX$ in teknis-esteettistä  
ylivoimaisuutta visuaaliseen, melkein WYSIWYG<sup>1</sup>-tyyliseen, käyttötapaan.  
 $\text{LyX}$  näyttää pinnallisesti katsoen tavalliselta tekstinkäsittelysovellukselta,  
mutta sievän ulkokuorensa alla työskentelee  $\LaTeX$ in ladontakoneisto. " $\LaTeX$   
ilman kyyneleitä".

Tällä hetkellä  $\text{LyX}$ issä on n. 200 000 koodiriviä.

## 3 $\text{LyX}$ in kehitysprosessi ja työkalut

### 3.1 Kehittäjien yhteisö ja historia

$\text{LyX}$ in elinkaari on jo melkein vuosikymmenen mittainen. Ensimmäisiä versioi-  
ta kehitti saksalainen Matthias ETTRICH vuosina 1994–95. Versio 0.10.7 oli v.  
1997 jo varsin käyttökelpoinen. Versio 1.0 julkaistiin helmikuussa 1999. Tätä  
versiota voidaan kutsua valmiiksi, kun aikaisemmat versiot olivat vielä niin  
keskeneräisiä, että niitä ei oikein kehtaisi antaa tavallisen loppukäyttäjän käyt-  
töön. Tällä hetkellä stabiiliversio on 1.3; kehitystyön kohteena oleva versio  
tulee olemaan 1.4 kun se julkaistaan, luultavasti vuoden 2004 aikana.

Tämän hetken aktiivikehittäjiä on puolisen tusinaa. Norjalainen Lars Gullik  
BJØNNES vastaa www- ja CVS-palvelimista ja kehitysprosessin koordinoimi-  
sesta. Aktiivikehittäjien lisäksi on monia satunnaisia kehittäjiä. He ovat usein  
kiinnostuneita tietyistä erityisominaisuuksista, joita he ovat lisänneet  $\text{LyX}$ iin  
ja joita he ylläpitävät. Oman lukunsa muodostavat kääntäjät, jotka ovat kään-  
täneet valikkotekstit yms n. 30 eri kielelle — mukaan lukien kreikka, venäjä,  
heprea, arabia ja kiina! Tämä on jatkuvaa työtä, koska  $\text{LyX}$  kehittyi koko ajan.

$\text{LyX}$  on kirjoitettu C++ kielellä, käyttäen hyväksi sen tarjoamia objektiorien-  
toituneen ohjelmoinnin mahdollisuuksia. Myös STL:ää (Standard Template  
Library) käytetään hyväksi, sekä Boostin (<http://www.boost.org>) tar-  
joamia hyödyllisiä apuvälineitä, mm. älykkäät pointerit. (Kehittäjien postitus-  
listalla keskustellaan usein siitä, kuinka pitkälle on mentävä modernin C++-  
kielen käytössä, koska monella kehittäjällä on hieman vanhemmat järjestelmät  
käytössään.)

<sup>1</sup>Eli, kuten  $\text{LyX}$ -kotisivu asian ilmaisee: WYSIWYM, What You See is What You Mean.

LyXin kehityksessä tärkeä maamerkki oli myös päätös koodin modularisoimiseksi siten, että se toimisi yhtä hyvin eri graafisten käyttöliittymäratkaisujen kanssa: GUI-I, “GUI<sup>2</sup> Independence”.

Pääsin tämän kehitysyhteisön jäseneksi kirjoittautumalla LyXin kehittäjien postituslistalle (`lyx-devel@lists.lyx.org`) ja toimittamalla sinne ensimmäiset parannusehdotukset, lähdekoodin muutosreseptit eli “pätsit<sup>3</sup>”, tähän listaan. Kommentit kyllä tulivat! Minulla ei ollut yhtään C++-kehitystyön kokemusta... vain proseduraaliset kielet, kuten C ja Pascal ja Fortran, olivat tuttuja. Tekemällä oppii.

## 3.2 Graafiset työkalupakit

LyX on graafinen sovellus. Käyttäjä on sen kanssa vuorovaikutuksessa graafisen käyttöliittymän eli GUI:n kautta. LyXin tapauksessa liittymä perustuu X-ikkunointijärjestelmään, UNIX-maailman standardi-ikkunointiratkaisuun.

X sellaisenaan ei vielä riitä luomaan graafista käyttöliittymää. Periaatteessa se kyllä onnistuisi, mutta vain hyvin hankalasti. Yksinkertaisetkin asiat vaatisivat satoja ohjelmarivejä. Siksi on kehitetty erilaisia ns. *graafisia työkalupakkeja*.

Niistä vanhin on Motif. Se on tutun näköinen, kantikas, harmaa. Windows 3.1 on Motifin johdannainen. Kaunis se ei ole, toimiva kyllä. LyXin eräs varhaisversio oli todella kirjoitettu Motifia varten.

Hieman nuorempi on Xforms. Se on tuskin Motifia paremman näköinen mutta sitä huomattavasti monipuolisempi ja helppokäyttöisempi.

Kaiken tuorein saavutus UNIX-maailman graafisissa käyttöliittymissä on *graafiset käyttöympäristöt* kuten Gnome ja KDE. Nämä ovat paljon enemmän kuin vain graafisia työkalupakkeja — ne ovat kokonaisia ympäristöjä, joissa kaikki sovellukset käyttäytyvät samalla, tutulla, integroidulla tavalla. Aivan kuten Windowsin ja Macin sovellukset käyttäytyvät, tai pitäisi käyttäytyä.

LyXin ensimmäinen versio käytti Motifia. Kuitenkin jo LyXin ensimmäinen kehittäjä Matthias ETTRICH muutti ohjelman käyttämään Xforms-työkalupakkia. Tämä on edelleen LyXin standardiratkaisu. Nykyisin Qt, KDE-ympäristön käyttämä graafinen työkalupakki, on lähes yhtä hyvin tuettu.

LyXissä on toteutettu teknisenä ratkaisuna GUI-I eli “GUI Independence”: graafiset ja ei-graafiset osat on huolellisesti erotettu toisistaan niin, että uuden graafisen työkalupakin liittäminen LyXin käyttöön olisi mahdollisimman suoraviivainen. Graafisen työkalupakin vaihto onnistuu helposti vain uudelleen konfiguroimisella ja kääntämisellä. GUI-I on osa MVC (Model-View-Controller) -arkkitehtuuria, jota LyX myös käyttää.

<sup>2</sup>GUI: Graphical User Interface.

<sup>3</sup>En. *patch*. Luultavasti nimi juontaa juurensa vaatteiden korjailussa käytetyistä kangaspaikoista.

### 3.3 Kotisivu, CVS, GNU-työkalut, make, autoconf

LyXillä on oma kotisivusto: <http://www.lyx.org>.

Kehitystyössään LyXin kehittäjät käyttävät järjestelmää nimeltä CVS, Concurrent Versioning System, joka on ns. *versiohallintajärjestelmä*. Tämä palvelimella pyörivän järjestelmän tietokanta pitää sisällään LyXin nykyisen ja kaikkien aikaisempienkin versioiden lähdekoodit. Lähdekooditiedostojen eri versiot tallennetaan tehokkaasti perusversiona ja erotustietueina, ns. *deltoina*. Kehittäjä voi "imuroida" järjestelmästä LyXin nykyversion (tai periaatteessa vanhemman version) omaan kehityskäyttöön, jonka jälkeen koko sen hakemistopuu on hänen omalla paikallisella kovalevyllä. Sitten, kehitystyön päätyttyä käyttäjä voi syöttää tekemänsä muutokset taas takaisin CVS-palvelimeen — edellyttäen, että hänellä on siihen lupaa.

CVS voi olla, ja tavallisesti on, monen kehittäjän yht'aikaisessa käytössä ("Concurrent"). Jos kuitenkin kaksi kehittäjää yrittävät samanaikaisesti muuttaa saman tiedoston samaa aluetta syntyy ristiriita, jonka ratkaiseminen jää toisen (hitaamman) kehittäjän manuaaliseksi tehtäväksi.

CVS-palvelimen tietokannasta on olemassa myös *www-vedos*, jota voi katsella selaimella LyXin kotisivustolla. Näin kaikilla, myös ulkopuolisilla, on mahdollisuus tarkistaa mitä LyXin kehiteillä olevalla versiolla on viime aikoina tapahtunut tai seurata jokaisen tiedoston muutoshistoria.

CVS:n lisäksi tarvitaan kehitystyössä monia muita työkaluja, jotka tunnetaan kollektiivisesti GNU-työkaluketjuna:

- editori (joko *emacs* tai *vi*),
- kääntäjä (*gcc*, GNU Compiler Collection, sisältää myös hyvän C++-kääntäjän),
- *make* (valikoidun kääntämisen hallintajärjestelmä, jonka avulla voi kääntää LyXin sadoista lähdekooditiedostoista vain ne, jotka ovat itse muuttuneet viime käynnöksen jälkeen tai riippuvat tiedostosta, joka on muuttunut) ja
- *autoconf/automake/libtool* työkalut, jotka auttavat kääntämään ja linkittämään ohjelman ja sen tarvitsemat monet kirjastot oikein, toimivaksi binääriksi, isäntäkoneen tai sen käyttöympäristön lukuisista mahdollisista omituisuuksista huolimatta.

## 4 "Branches"-ominaisuuden kehitys

"Branches" eli "haaraumat"-ominaisuutta yritettiin ensin toteuttaa käyttämällä *branch*-nimistä attribuuttia dokumentin merkkien eli *kirjasinleikkauksen* (fontin) ominaisuutena. Toisin sanoen jokaisella dokumentin merkillä oli näiden fonttiominaisuuksien — kuten koon tai värin — lisäksi vielä yksi lisäominaisuus: *branch*, jonka arvoksi sallittiin kahdeksan eri värin nimeä: *white*, *black*, *red*, *green*, *blue*, *cyan*, *magenta* ja *yellow*. Näitä värejä käytettiin itse

asiassa myös tiettyyn haaraumaan kuuluvien dokumentin tekstiosien värjäämiseksi kuvaruudulla.

Tämä tekstin ominaisuuksiin perustuva ratkaisu oli tavallaan vain yllä (osassa 1.2) mainitun "kludgen" jalostettu laitos. Sen kehittämisen tuloksena laadittiin pätsi, josta löytyy seuraava viesti (12. kesäkuuta 2003): <http://www.mail-archive.com/lyx-devel@lists.lyx.org/msg57376.html>.

Tätä ratkaisua ei kuitenkaan haluttu hyväksyä LyXin osaksi. Se oli liian improvisoitu ja sen suurimpana hankaluutena pidettiin sitä, että *käyttäjän* tehtäväksi jäisi eri haaraumien ja käyttövärien välisen vastaavuuden muistaminen.

Päätettiin luoda dokumentin sisälle tietorakenne nimeltä BranchList, joka sisältäisi käyttäjän määrittämiä haaraumia ominaisuuksineen. Tiettyyn haaraumaan kuuluvat dokumentin tekstipalaset laitettaisiin *upotuksiin* (en. *inset* — dokumentin sisältämä objekti, tekstin lomassa oleva "säiliö").

Ensimmäinen upotuspohjainen ratkaisu julkaistiin 31. heinäkuuta 2003, ks. viesti: <http://www.mail-archive.com/lyx-devel@lists.lyx.org/msg59560.html>.

Tämän jälkeen ratkaisua kehiteltiin edelleen: esimerkiksi *värien* käyttöä mahdollistettiin haaraumaupotuksen pohjaväriksi, yksi käyttäjän määrittämä, mielivaltainen väri jokaiselle haaraumalle. Vasta elokuun 17. päivänä julkaistiin "*Branches mega-patch*", joka syötettiin CVS:ään.

Voimme seurata *haaraumat*-ominaisuuden kehitystä myös CVS-tietokannan muutosmerkintöjen avulla. Yritin ensin toteuttaa haaraumaupotusta jo olemassa olevan *Notes*-upotuksen varianttina. Samalla yleistin *Notes*-upotuksen arkkitehtuurin niin, että sen avulla voitaisiin myös toteuttaa kaikenlaisia muita annotaation tyyppejä.

Näin *Notes*-upotusta kehitettiin edelleen aikana 8. heinäkuuta – 22. syyskuuta, ja CVS:n sisältämät merkinnät keskeisen tiedoston *insetnote.C* kohdalla löytyvät tästä: <http://www.lyx.org/cgi-bin/viewcvs.cgi/lyx-devel/src/insets/insetnote.C>. Aikaviivalla löytyy 8 pätsiä nimellä VERMEER.

*Haaraumat*-ominaisuuden lopullista, upotuspohjaista toteutusta, jota kehitettiin aikana 17. elokuuta – 22. syyskuuta, voidaan seurata tiedoston *insetbranch.C* avulla: <http://www.lyx.org/cgi-bin/viewcvs.cgi/lyx-devel/src/insets/insetbranch.C>. Nimellä VERMEER löytyy 4 pätsiä. Ensimmäinen niistä oli yllämainittu "mega-patch", joka itse asiassa loi tämän tiedoston.

Tietenkin tähän työhön kuului paljon muitakin tiedostoja kuin vain kaksi yllä mainittua: mm. *BranchList.[Ch]*, *FormBranch.[Ch]*, *ControlBranch.[Ch]*, vain muutaman mainitakseni. Kuitenkin yllä mainituista merkinnöistä saa hyvän käsityksen työn kulusta. Se antaa myös osviittaa, miten perusteellisesti ja julkisesti avoimen lähdekoodin kehitysprojekti on dokumentoitu. (Laittomasti kopioitujen koodien salakuljettaminen näin julkiseen projektiin ei todellakaan olisi suositeltavaa!)

Eräs LyXin upotusten hyödyllinen ominaisuus on, että ne voidaan "laskostaa" ja taas avata. Se toimii siten, että jos valitsee haaraumista yhden paperille tu-

lostettavaksi, niin myös sen haarauman kaikki upotukset kuvaruudulla aukeavat automaattisesti ja kaikki muut laskostuvat...

## 5 Johtopäätökset

Verkkodesia-projektiin liittyvänä vapaaehtoistyönä lisättiin LyX-dokumenttiprosessoriin *haaraumat*-ominaisuus monikielisten dokumenttien tuottamiseksi samasta lähdekoodista. Työ tehtiin suurilta osin kesäkuukausina, kesämökillä aktiivisesti ja viihtyisästi vietetyn ajan lomassa.

Työn kuvauksesta saa jonkinlaisen käsityksen siitä, kuinka kurinalaista avoimen lähdekoodin ohjelmistojen kehitysprosessi on ja miten ryhmän dynamiikka toimii mahdollisimman laadukkaan lopputuloksen aikaansaamiseksi.

LyXin seuraava versio uusine ominaisuuksineen tulee TKK:n tiede- ja opetusyhteisön käyttöön v. 2004 aikana, kun UNIX- ja Linux-työasemat ja niihin asennetut sovellukset päivitetään. LyXiä voidaan käyttää myös Windows-työasemalta käsin: se edellyttää ssh:n (Secure Shell, ks. <http://www.hut.fi/atk/lisenssit/mikroturva/ssh/>) ja X-palvelimen (esim. <http://www.hut.fi/atk/lisenssit/xwinpro/>) asennusta. Linux-työaseman käyttö sen sijaan ei edellytä erityistoimenpiteitä.