

Veisuuvekotin

Dynaamisia virsinuotteja verkossa

Jukka Stenlund

23. elokuuta 2013

SIBELIUS-AKATEMIA
Klassisen musiikin osasto
Kirkkomusiikin oppiaine
Projekti (5p22)

Sisältö

1	Johdanto	5
2	Laulukokoelmien sähköinen julkaiseminen	9
2.1	Sähköisen median mahdollisuudet	9
2.2	Sähköiset julkaisualustat	11
2.2.1	Laitealustat	11
2.2.2	Ohjelmistoalustat	12
2.3	Nuottikuva kuvana ja informaationa	13
2.4	Digitaalisten nuottikokoelmien julkaiseminen	15
2.5	Nuottien julkaiseminen verkossa	16
3	Sähköisesti julkaistuja laulukokoelmia	25
3.1	Virsikirjajulkaisuja	25
3.1.1	Verkkovirsikirja	25
3.1.2	eVirsikirja	26
3.1.3	virret.net-palvelu	26
3.1.4	Suomen Rauhanyhdistysten Keskusyhdistyksen sähköiset laulukirjat	27
3.2	Muiden laulujen kokoelmia	28
3.2.1	Christian Virtual Hymnal	28
3.2.2	iReal b	29
4	Uuden sähköisen virsikirjajulkaisun laatiminen	31
4.1	Palveltavat henkilöt ja tilanteet	31
4.2	Toivottavat ominaisuudet	32
4.3	Projektin puitteissa julkaistavien virsien valinta	36
5	Virsikoodi – menetelmä virsinuottien informaation esittämiseen tekstin muodossa	41
5.1	Koodaukselle asetetut vaatimukset	41
5.2	Malliedition valinta	42

5.3	Koodauksen yleiset piirteet	43
5.4	Virsinuottien merkintöjen koodaaminen	44
6	Veisuuvekotin – virsilähtöinen järjestelmä laulunuohtien verkkojulkaisuun	63
6.1	Toiminnot	63
6.2	Toteutustekniikka	67
6.2.1	Tietoteknisten käsitteiden esittely	67
6.2.2	Järjestelmän arkkitehtuuri ja käytetyt teknologiat . . .	68
6.2.3	Tietokannan rakenne	72
6.2.4	Virsikoodina esitetyn informaation muuntaminen oliomalliksi	74
6.2.5	Lauluinformaation esittämiseen käytetty oliomalli . . .	75
6.2.6	ABC-muotoisen esityksen tuottaminen oliomallista . .	83
6.2.7	Käyttöliittymäluokat	84
6.2.8	Tietoturva	89
6.2.9	Kehitysympäristö ja testaaminen	91
6.3	Jakelu ja lisensointi	91
7	Projektin kulku	93
8	Loppukatsaus	99
	Lähteet	101
A	Laulukirjasto	107
B	Kysely sähköisen virsikirjan potentiaalisille käyttäjille	111
C	Esimerkkejä koodatuista virsistä	117
D	Veisuuvekotin tietokannan taulujen rakenne	119
E	GNU GENERAL PUBLIC LICENSE	125

Luku 1

Johdanto

Tämän projektin tarkoituksena on kartoittaa mahdollisuuksia, joita tietotekniikka tarjoaa laulukokoelmien, erityisesti Virsikirjan¹ laulujen julkaisemiseen sähköisessä muodossa. Käsillä olevassa kirjallisessa työssä esitellään erilaisia julkaisutekniikoita ja olemassaolevia sähköisiä laulukirjajulkaisuja. Eri-tyinen painopiste projektissa on kuitenkin uuden sähköisen virsikirjajulkaisun laatimisessa. Tätä julkaisua varten projektissa laadittiin Veisuuvekotin-niminen järjestelmä, jonka avulla laulukokoelmia voidaan julkaista verkkopalvelun muodossa. Järjestelmän avulla perustettiin virsinuotteja tarjoava verkkopalvelu osoitteeseen `veisuuvekotin.net`². Projektin arvostelua varten määriteltiin, että projektin käytännöllinen osuus eli Veisuuvekottimen rakentaminen ja `veisuuvekotin.net`-palvelun perustaminen kattavat 70 % projektin kokonaislaajuudesta, kun taas projektin kirjallinen osa eli käsillä oleva dokumentti kattaa loput 30 %.

Veisuuvekotin kehitettiin erityisesti Virsikirjan laulujen sähköistä julkaisemista ajatellen, mutta järjestelmän avulla voidaan julkaista myös muuta vastaavaa lauluaineistoa³. Projektissa perustettuun `veisuuvekotin.net`-palveluun talletettiin sävelmä yli sataan Virsikirjan virteen. Osaan näistä kirjoitettiin yksi tai useampia soinnutuksia reaalisointumerkein, ja osaan talletettiin sanoitus tai ensimmäisen säkeistön alkusanat. Virsikirjan aineistoa

¹Virsikirjalla tarkoitetaan tässä dokumentissa Suomen evankelis-luterilaisen kirkon Virsikirjaa, joka hyväksyttiin kirkolliskokouksessa 13. helmikuuta 1986.

²Projektin päättyessä veisuuvekotin.net-palvelua ei vielä avattu yleiseen käyttöön, koska järjestelmän toimintavarmuus ei ollut tasolla, jota yleisessä käytössä olevalta verkkopalvelulta voidaan edellyttää. Lisäksi projektista rajattiin pois joidenkin sellaisten toimintojen toteuttaminen, jotka olisivat tarpeen palvelun laajemman käyttämisen kannalta.

³Vastaavalla lauluaineistolla tarkoitetaan tässä säkeistömuotoisia lauluja, joissa jokainen säkeistö lauletaan samalla rytmillä, ja jotka voidaan nuotintaa yksiaanisesti käyttämättä nuottikuvissa merkintöjä, joita Virsikirjan nuottikuvissa ei käytetä. Virsikirjan nuottikuvissa käytetyt merkinnät käydään läpi luvussa 5.

on aikaisemminkin julkaistu sähköisessä muodossa, mutta Veisuuvekkottimen avulla laadittu verkkopalvelu tarjoaa mahdollisuuden muokata nuottikuvia valitsemalla sävellajeja, säkeistöjä ja soinnutuksia sekä kirjoittamalla lauluihin omia soinnutuksia. Tällainen muokattavuus on uutta Suomen evankelis-luterilaisen kirkon Virsikirjan sähköisen julkaisemisen saralla.

Tätä dokumenttia ei ole kirjoitettu tieteellisen tutkielman muotoon, joka edellyttäisi tutkimuskysymyksen, aineiston ja metodin määrittelyä. Projektiin liittyy kuitenkin kaksi pääkysymystä:

1. Millaisia ominaisuuksia digitaalisessa virsikirjassa pitäisi olla, jotta se palvelisi virsien laulamista?
2. Millä tavoin voidaan toteuttaa virsikirja, jossa on nämä ominaisuudet?

Edelliseen kysymykseen kirjoittajalla oli hypoteesi jo projektin alkaessa. Hypoteesia testattiin esittelemällä virsikirjaan kaavailtuja ominaisuuksia teologien ja kirkkomuusikoiden yhteisessä seminaarissa ja symposiumissa sekä kirkkohallituksessa. Otoskooltaan merkittävää, kvalitatiivisen tutkimuksen kriteerit täyttävää kyselytutkimusta tai muuta vastaavaa ei kuitenkaan tehty. Jälkimmäiseen kysymykseen vastausta etsittiin käytännön kautta, yrittämällä toteuttaa hyödyllisiksi arveltuja ominaisuuksia. Vastaukseksi saatiin, että ominaisuudet voidaan toteuttaa esimerkiksi niillä tekniikoilla, joita tässä projektissa sovellettiin.

Veisuuvekkottimen toteutuksessa hyödynnettiin useita avoimeen lähdekoodiin perustuvia tekniikoita, ja projektissa tuotettu lähdekoodi julkaistiin sivustolla `veisuuvekotin.net`. Lähdekoodilla tarkoitetaan toimintaohjeita, jotka ohjelmoija kirjoittaa tietokoneelle jollakin ohjelmointikielellä. Tämä koodi käännetään tai tulkitaan muotoon, jossa se voidaan tarjota tietokoneelle suoritettavaksi, mutta jossa sitä on ihmisten vaikea lukea. Lähdekoodia ei välttämättä julkaista lainkaan. Näin liikeyritykset, jotka tuottavat tietokoneohjelmia, varjelevat ammattisalaisuuksiaan: kilpaileville yrityksille ei haluta paljastaa, miten ohjelmat on toteutettu. Toisinaan tietokoneohjelman oikeuksien omistaja päättää kuitenkin julkaista lähdekoodin ja antaa luvan kenelle tahansa kehittää ohjelmaa eteenpäin. Tässä projektissa kirjoittaja pitää Veisuuvekkottimen ohjelmakoodin tekijänoikeudet itsellään, mutta lisensoi koodin GPL v. 3 -lisenssin alaisuudessa (kts. liite E). Lisenssi antaa kenelle tahansa luvan käyttää ohjelmakoodia, jaella sitä eteenpäin ja jatkokehittää sitä, kunhan samat oikeudet luovutetaan eteenpäin ja sovelletaan kaikkiin niihin tietokoneohjelmiin, joiden osaksi GPL-lisenssoitua ohjelmakoodia otetaan. Myös abcjs-kirjasto on lisensoitu GPL:n alaisuudessa, ja niinpä kirjaston käyttäminen osana Veisuuvekotinta edellyttikin Veisuuvekkottimen lisensoimista samalla lisenssillä.

Kansainvälistä kehittäjäyhteisöä ajatellen Veisuuvekottimen lähdekoodi kirjoitettiin englanniksi. Koodissa Veisuuvekotin-nimen vastineena on ilmaus HymnGadget⁴, ja tätä nimeä suositellaan käytettäväksi järjestelmästä kansainvälisissä yhteyksissä. Tässä dokumentissa käytetään kuitenkin suomenkielistä Veisuuvekotin-nimeä.

Projekti edellytti tekijältä sekä informaatioteknologian että musiikkialan tuntemusta. IT-alan tuntemusta kirjoittajalle kertyi tietojenkäsittelytieteen opinnoissa, joissa hän suoritti luonnontieteiden kandidaatin tutkinnon Helsingin yliopistossa ennen kirkkomusiikin opintojen aloittamista Sibelius-Akatemiassa. Kirjoittaja on myös ohjannut harjoitusryhmiä ja ohjelmoinnin harjoitustöiden tekemistä tietojenkäsittelytieteen kursseilla Helsingin yliopistossa sekä työskennellyt lyhyen ajan ohjelmoijana IT-alan yrityksessä.

⁴Englantia puhuvalle yleisölle sanan *veisuuvekotin* merkitys voitaisiin kääntää esimerkiksi ilmaisulla *a gadget for the act of singing hymns*.

Luku 2

Laulukokoelmien sähköinen julkaiseminen

2.1 Sähköisen median mahdollisuudet

Sähköinen media tarjoaa paperiseen verrattuna uusia mahdollisuuksia liittyen muun muassa tallennustilaan, tiedon etsimiseen, automaattiseen tiedon muokkaamiseen sekä mahdollisuuteen siirtää ja monistaa tietoa siirtelemättä ja monistamatta esineitä.

Paperille painetuissa julkaisuissa sisällön määrää joudutaan rajoittamaan, jottei julkaisu kasvaisi mitoiltaan tai painoltaan epäkäytännöllisen suureksi. Nykyinen tietotekniikka mahdollistaa valtavien tietomäärien tallentamisen hyvin pieneen tilaan, ja tietoverkkojen avulla voidaan pienellä päätelaitteella, esimerkiksi puhelimella, olla yhteydessä tietovarantoihin, jotka sijaitsevat toisaalla ja joiden tallennusmedian koolla ja painolla ei ole päätelaitteen käyttäjän kannalta merkitystä.

Paperisiin kirjoihin voidaan painaa sisällysluetteloja ja hakemistoja, minä lisäksi tärkeät kohdat voidaan merkitä sivujen väliin asetettavien kirjanmerkkien tai sivujen reunaan liimattavien liuskojen avulla. Sähköinen media mahdollistaa kuitenkin paljon monipuolisemmat ja nopeammat hakuominaisuudet: esimerkiksi tekstikatkelmaan perustuva haku on helppo toteuttaa.

Paperille painettu kirja ei juurikaan pysty muokkaamaan omaa sisältöään tai sisältönsä esitystapaa, vaikka lastenkirjoistakin tuttujen luukkujen, vedettävien liuskojen ja sen sellaisten avulla voidaan toteuttaa hieman muokattavuutta. Tietokoneistetussa sähköisessä mediassa voidaan sen sijaan toteuttaa hyvin monipuolista automaattista tietojenkäsittelyä, joka mahdollistaa esimerkiksi erilaisten esitysten tuottamisen samasta aineistosta käyttäjän mieltymysten ja tarpeiden mukaisesti. Yhteislaulunotit ovat mielenkiintoinen

sähköisen julkaisemisen erityistapaus, koska niihin liittyy muokkaustoimenpide, transponointi, joka täyttää kolme ehtoa:

- Kyseessä on tarpeellinen toimenpide. Laulajien äänialat vaihtelevat, joten lauluja on voitava laulaa eri korkeuksilta.
- Toimenpide on ihmiselle suhteellisen hankala suorittaa reaaliaikaisesti.
- Toimenpide on täysin mekanisoitavissa ja sellaisena soveltuu hyvin tietokoneen suoritettavaksi.

Paperimedian tapauksessa tiedon siirtäminen edellyttää aina fyysisten esineiden kuten lehtien ja kirjojen siirtämistä paikasta toiseen. Tiedon monistaminen taas vaatii fyysisten esineiden monistamista. Sekä esineiden monistaminen että esineiden siirtely ovat hitaita ja kalliita operaatioita, jos on monistettava paljon tai siirrettävä esineitä pitkien välimatkojen päähän. Sähköisen median tapauksessa, tietoverkkoja käytettäessä, ei tiedon siirtäminen edellytä päätelaitteiden eikä verkkojen ylläpitämiseen tarvittavien esineiden siirtelyä, eikä tiedon monistaminen päätelaitteiden monistamista. Tästä johtuen tiedon julkaiseminen ja levittäminen sähköisesti on ainakin julkaisijalle ja levittäjälle halvempaa kuin paperimedian tapauksessa. Päätelaitteiden hankkimisesta koituu kustannuksia ja operaattorit laskuttavat verkkoyhteyksistä, mutta kuluttajalle, joka on jo hankkinut päätelaitteen ja joka maksaa kiinteää kuukausihintaa verkkoyhteydestään, ei yksittäisen sähköisen julkaisun hankkimisesta koidu uusia päätelaite- eikä verkkoyhteyskustannuksia.

Mahdollisuus sähköisessä muodossa olevan tiedon siirtämiseen tietoverkossa esineitä siirtelemättä tarkoittaa myös, että tiedon laittaminen tarjolle verkkoon parantaa tiedon saatavuutta olettaen, että verkkoyhteydellä varustettuja päätelaitteita, joihin tiedon tarvitsijalla on pääsy, on hyvin saatavilla. Suomalaisissa kotitalouksissa on usein kiinteä Internet-yhteys ja verkkoselaimella varustettu tietokone. Monilla työ- ja opiskelupaikoilla on helppo päästä käsiksi verkkomateriaaleihin työ- tai opiskelupaikan tietokoneita ja verkkoyhteyttä käyttäen. Ihmisillä, jotka kantavat aina mukanaan verkkoselaimella varustettua matkapuhelinta, on operaattorinsa dataverkon peittoalueella pääsy verkon julkiseen tietoaaineistoon puhelimen avulla.

Sähköisen median tarjoama mahdollisuus julkaisujen muokkaamiseen ja tiedon siirtämiseen halvalla tekee mahdolliseksi vuorovaikutteiset, yhteisölliset julkaisut, joissa lukijat voivat kommentoida julkaisuja ja jakaa kommenttinsa kaikkien julkaisun lukijoiden kanssa. Esimerkiksi uutisia julkaisevat verkkosivustot voivat tarjota lukijoille mahdollisuuden kommentoida sekä alkuperäistä uutista että aiempia uutiseen kirjoitettuja kommentteja.

Verkossa toimiva sähköinen nuottijulkaisu voi tarjota käyttäjilleen esimerkiksi mahdollisuuden ladata nuottien yhteyteen omia äänitettyjä tai videokuvattuja performanssejaan julkaisun kappaleista tai vaikkapa mahdollisuuden kirjoittaa nuotteihin omia soinnutuksiaan ja jakaa nämä kaikkien julkaisun käyttäjien kesken. Paperisen kirjan marginaaliin voi kirjoittaa omia kommenttejaan ja paperille painettua nuottia voi soinnuttaa kynää käyttämällä, mutta paperille tehtyjen merkintöjen jakaminen julkaisun koko käyttäjäyhteisön kesken on käytännössä mahdoton tehtävä, koska kaikki julkaisun käyttäjät eivät edes ole merkintöjä tehneen tiedossa.

2.2 Sähköiset julkaisualustat

2.2.1 Laitealustat

Laitealustalla tarkoitetaan tietokoneohjelmista puhuttaessa fyysisiä laitteita, joilla ohjelmaa käytetään. Sähköisen laulukirjan laitealustan olisi hyvä täyttää muutamia ehtoja:

1. Laitteelle pitäisi voida kirjoittaa ohjelmia, jotka toteuttavat sähköiseen virsikirjaan halutut toiminnot.
2. Laitteen näytön pitäisi olla riittävän suuri, ja näytön resoluution¹ pitäisi olla riittävän korkea, jotta nuotteja olisi miellyttävä lukea.
3. Laite pitäisi voida ottaa mukaan laulutilanteeseen.
4. Laite pitäisi voida asettaa nuottitelineelle.
5. Spontaaneja laulutilanteita ajatellen olisi kätevää, jos laite olisi käyttäjillään aina mukana.
6. Laitteita tulisi olla ihmisillä entuudestaan, jotta sähköisen laulukirjan käyttäjäksi ryhtyminen ei vaatisi laitehankinnan tekemistä.
7. Laitteen pitäisi herättää mahdollisimman vähän huomiota laulutilanteessa. Tällä näkökohdalla on merkitystä erityisesti sähköisestä virsikirjasta puhuttaessa, koska virsiä lauletaan usein hartaustilanteissa.

Parhaillaan yleistyvät taulu- ja sormitietokoneiksi kutsutut laitteet täyttävät hyvin ehdot 2, 3 ja 4, mutta huonommin ehdon 6. Sylimikrojen ongelmana on ainakin ehto 4, puhelinten taas ehto 2 (fyysisen koon osalta).

¹Resoluutiolla tarjotaan näytön kuvapisteen lukumäärää. Mitä korkeampi resoluutio, sitä tarkempia kuvia näytölle voidaan piirtää.

Kaikki nämä laitteet täyttävät ehdon 1, jos puhelimista tarkastellaan vain älypuhelimia.

Sähköisten kirjojen eli E-kirjojen lukemista varten on kehitetty laitteita, joiden ominaisuudet on optimoitu nimenomaisesti kirjojen lukemista varten. Näiden laitteiden näytöt perustuvat niin kutsuttuun sähköiseen musteeseen, ja näyttöjen ominaisuudet poikkeavat tietokoneiden näytöistä. Sähköisellä musteella toteutetun näytön kuva on jossain määrin verrattavissa painettuun kuvaan. Nämä näytöt kuluttavat energiaa ainoastaan silloin, kun kuvaa päivitetään, joten kirjaa luettaessa ainoastaan sivua käännettäessä kuluu energiaa. Akut kestävät näin ollen pitkään. Sähköiseen musteeseen perustuvien näyttöjen kuvan vaihtaminen toiseen kestää kuitenkin pitkään verrattuna tietokoneissa käytettyihin näyttöihin. Liikkuvaa kuvaa ei voida näillä näytöillä esittää, mikä asettaa rajoituksia käyttöliittymille. Myös ehto 1 saattaa muodostua sähköisten kirjojen lukulaitteiden kohdalla ongelmalliseksi, koska nämä laitteet on räätälöity tiettyyn käyttötarkoitukseen.

2.2.2 Ohjelmistoalustat

Yksikään tietokoneohjelma ei toimi kaikissa mahdollisissa laitteissa. Toisaalta useinkaan ei ohjelmaa tarvitse kirjoittaa yhdelle tietylle laitteelle. Yleensä ohjelmat kirjoitetaan pikemminkin tietylle käyttöjärjestelmälle kuin tietylle laitteelle. Ohjelma voidaan kirjoittaa esimerkiksi Microsoft-yhtiön Windows-käyttöjärjestelmälle, jolloin ohjelma toimii hyvinkin erilaisissa tietokoneissa, kunhan näissä koneissa on käyttöjärjestelmänä Windowsin versio, jota ohjelma tukee.

Windowsilla on vankka asema sekä pöytä- että sylimikroissa. Apple-yhtiön Mac-tietokoneissa käytetään kuitenkin OS X -järjestelmää. Linux-käyttöjärjestelmällä on myös huomattava määrä käyttäjiä. Applen iPad-sormitietokoneissa ja iPhone-älypuhelimissa puolestaan käytetään iOS-käyttöjärjestelmää. Toinen älypuhelimissa ja sormitietokoneissa suosittu käyttöjärjestelmä on Android. Niinpä esimerkiksi Windowsille kirjoitettua ohjelmaa ei voida käyttää kuin osassa niistä laitteista, jotka fyysisten ominaisuuksiensa puolesta soveltuisivat ohjelman käyttämiseen.

Eri ympäristöihin ei samaa ohjelmaa tarvitse kuitenkaan kirjoittaa kokonaan uudestaan. Ohjelmat pyritään usein kirjoittamaan siten, että ne toimisivat useammassa ympäristössä suhteellisen pienillä muutoksilla. Java-kielellä kirjoitetut ohjelmat toimivat periaatteessa ilman mitään muutoksia kaikissa ympäristöissä, joihin on saatavilla nk. Java-virtuaalikone. Java-ohjelmat kirjoitetaan ajatellen tätä kuvitteellista konetta, ja todellisille koneille on kirjoitettu ohjelmia, jotka saavat nämä koneet käyttäytymään Java-virtuaalikoneen tavoin.

Yksi, nykyään usein käytetty tapa ratkaista lukuisten laite- ja ohjelmistoympäristöjen ongelma on tehdä tietokoneohjelmasta verkkoselaimella käytettävä. On olemassa tekniikoita, jotka ovat hyvin laajasti yhteensopivia eri verkkoselainten kanssa, jolloin näillä tekniikoilla toteutettua ohjelmaa voidaan käyttää eri verkkoselaimilla, jotka toimivat eri päätelaitteissa.

Usein verkkoselaimella käytettävät palvelut vaativat jatkuvaa verkkoyhteyttä, koska palvelun arkkitehtuuri perustuu asiakkaan päätelaitteen ja verkkopalvelimen toistuvaan kommunikointiin. On kuitenkin myös mahdollista toteuttaa palvelu siten, että verkkosivun lataamisen yhteydessä käyttäjän koneelle siirretään kaikki palvelussa tarvittava data ja ohjelmakoodi, jota sen jälkeen suoritetaan käyttäjän päätelaitteessa. Tällöin verkkoyhteyttä tarvitaan vain palvelun käyttämisen alussa.

2.3 Nuottikuva kuvana ja informaationa

Jotta nuottikuva voitaisiin julkaista digitaalisissa medioissa, on nuottikuvan sisältämä informaatio ensin muunnettava digitaaliseen muotoon eli jonoksi bittejä². Jos nuottikuva on laadittu käsin, voidaan kuva skannata, jolloin skanneri tuottaa bittijonon, joka voidaan tulkita skannattua nuottia vastaavaksi kuvaksi. Jos nuottikuva on laadittu tietokoneella käyttäen nuotinnusohjelmaa, on nuotinnusohjelma nuottia talletettaessa tuottanut bittijonon, jonka ainakin kyseinen ohjelma itse osaa tulkita takaisin nuoteiksi.

Digitaalisia muotoja, joihin nuottikuva voidaan muuntaa, on useita. Nämä muodot eli informaation kuvaamisen tavat voidaan jakaa kahteen luokkaan sen mukaan, kuvataanko nuottikuvan visuaalista ulkoasua vai merkitysisältöä. Ulkoasun kuvaamisen tavat voidaan edelleen jakaa kahteen luokkaan sen mukaan, millaisella käsitteellisellä tasolla kuvaaminen tapahtuu.

Käsitteellisesti matalalla tasolla tapahtuvaa nuotin ulkoasun kuvaamista edustaa rasterigrafiikkaan perustuvan kuvatiedoston laatiminen nuotista. Kun kuva talletetaan rasterigrafiikkana, käsitellään kuvaa ikään kuin pienistä

²Tietokoneet kykenevät käsittelemään eri muodoissa olevaa informaatiota, koska riippumatta siitä, onko kyseessä esimerkiksi teksti, kuva vai ääni, informaatio on koodattu biteiksi, joita voidaan kuvata vaikkapa luvuilla 1 ja 0. Tällöin kaikki tietokoneille talletetut tiedostot ovat ikään kuin ykkösten ja nollien jonoja riippumatta siitä, onko kyseessä esim. doc-muotoinen tekstidokumentti, mp3-muotoinen äänitiedosto, jpg-muotoinen kuvatiedosto tai sib-muotoinen Sibelius-nuotinnusohjelman tiedosto. Koneilla, joilla voidaan käsitellä bittejä, voidaan siis käsitellä hyvin eri tyyppisiä tietosisältöjä. Jotta tekstejä, kuvia, musiikkia ym. voitaisiin tallettaa bittijonoina, on sovittava, millaisilla bittijonoilla mitäkin asioita esitetään, eli miten informaatio koodataan biteiksi. Tämän sopimuksen pohjalta talletettuja bittijonoja on myöhemmin mahdollista tulkita takaisin teksteiksi, kuviiksi, musiikiksi jne.

pisteistä koostuvana mosaiikkina. Kunkin pisteen väri talletetaan erikseen. Käsitteellisesti korkeammalla tasolla tapahtuvaa ulkoasun kuvaamista edustaa vektorigrafiikka. Vektorigrafiikan idea on, että kuva talletetaan käyttämälle geometrisia käsitteitä. Esimerkiksi nuottiviivaston viiva voidaan vektorigrafiikkana tallettaa kirjaamalla ylös, että kuvaan tulee piirtää jana, ja kirjaamalla lisäksi ylös janan alku- ja loppupisteiden sijainnit kuvassa.

Jos on tarpeen tallettaa valokuvia, jotka voivat esittää mitä tahansa, on tallettaminen helpointa tehdä rasterigrafiikan ideaa noudattaen. Esimerkiksi henkilön kasvoja esittävän kuvan purkaminen geometrisiksi kuvioiksi olisi haastava tehtävä. Jos sen sijaan halutaan tallettaa vaikkapa tekninen piirustus, joka on laadittu piirustusohjelmalla, joka tarjoaa työkaluja geometrinen kuvioden piirtämiseen, on luontevaa tallettaa kuva vektorigrafiikkana. Nuottinohjelmilla laaditut nuottikuvat ovat geometrisesti ottaen melko yksinkertaisia, joten näiden tallettaminen kuvana voitaisiin tehdä vektorigrafiikan keinoin. Tällöin kuvien kirjattaisiin sisältävän suoria ja kaarevia viivoja, soikioita ym.

Rasterigrafiikka kuvaa kohteensa aina vain likimääräisesti. Mitä pienempiin pisteisiin kuva jaetaan, sitä tarkemmin kuva saadaan talletettua. Vektorigrafiikkana voidaan geometriset kuvat tallettaa matemaattisen tarkasti, jolloin niitä voidaan esimerkiksi skaalata vapaasti laadun heikentymättä. Esimerkiksi ihmiskasvoja talletettaessa päädyttäisiin vektorigrafiikkaakin käyttäessä likimääräiseen kuvaamiseen, koska ihmiskasvot voidaan kuvata geometrisin käsittein vain likimääräisesti.

Kun talletettu kuva esitetään kuvapisteistä koostuvalla näytöllä, joudutaan vektorigrafiikkakin lopulta muuntamaan rasterigrafiikaksi. Muunnos voidaan kuitenkin tehdä siten, että kuvapisteiden määrä vastaa näyttöruudun kuvapisteiden määrää, jolloin kuva on niin tarkka kuin kyseisellä näyttöruudulla pystytään esittämään.

Jos nuottikuvaa talletettaessa halutaan vain, että talletetun tiedon perusteella saadaan luotua talletettua vastaava nuottikuva, on käytännöllistä tallettaa nuotti kuvana käyttäen jotain sellaista kuvien tallettamisen tapaa, jota käytetään mahdollisimman yleisesti. Tällöin kannattaa käytännössä tallettaa kuva jossain muodossa, jota käytetään muunkinlaisten kuvien kuin nuottikuvien tallettamiseen. Tällöin talletetun tiedon voi myöhemmin tulkita kuvaksi sellaisillakin välineillä, joita suunniteltaessa ei olla ajateltu nuottikuvia lainkaan. Tällaista lähestymistapaa edustaa esimerkiksi nuotin tallettaminen JPEG-tiedostoksi, jota käytetään digitaalisten valokuvien tallettamiseen. Jos ei olla talletettu nuotin ulkoasua vaan nuottikuvan merkityssisältö, tarvitaan erityisesti nuotti-informaation tulkitsemiseen tarkoitettu väline, jotta informaatio saataisiin muunnettua nuottikuvaksi.

Nuottikuvaa talletettaessa ei välttämättä tarvitse eikä kannata tallettaa lainkaan nuottikuvan visuaalista ilmiä. Voidaan tallettaa myös nuottikuvan merkityssisältö. Jos nuottikuva esittää esimerkiksi yksiäänistä melodiaa, voidaan tallettaa tieto siitä, mikä on kappaleen sävellaji ja mitkä ovat melodian nuottien korkeudet ja kestot. Jotta tällaista tietoa voitaisiin tallettaa, on sovittava, millaisilla bittijonoilla musiikillisia käsitteitä kuvataan.

Ei kuitenkaan ole välttämätöntä sopia vastaavuutta suoraan musiikillisten käsitteiden ja bittijonojen välille. Koska ihminen ajattelee bittijonoja luontevammin kirjaimia ja näistä muodostuvia sanoja ja laajempia kokonaisuuksia, voidaan sopia, miten musiikillinen informaatio esitetään tekstin muodossa. Koska monia tietokoneella käsiteltäviä asioita on ihmisten näkökulmasta luontevaa esittää tekstinä, on luotu menetelmiä tekstin esittämiseksi bittijonoina, ohjelmia, jotka tekevät muunnoksia tekstistä bittijonoiksi ja takaisin ja ohjelmia, joilla tekstiä voidaan muokata. Jos laaditaan yksiselitteinen kuvauskieli, jolla esimerkiksi mikä tahansa perinteisellä länsimaisella nuottikirjoituksella esitettävissä oleva melodia voidaan kuvata tekstin muodossa, voidaan tällaista tekstiä kirjoittaa samoilla tekstieditoriohjelmilla, joilla kirjoitetaan esimerkiksi ohjelmakoodia, ja tekstin muuntaminen tallettaviksi bittijonoiksi voidaan tehdä noudattaen samoja standardeja, joita muutenkin käytetään tekstin biteiksi koodaamiseen.

Nuottikuvan sisältämän informaation tallettamisella musiikkia kuvaavia käsitteitä käyttäen saavutetaan merkittävä etu. Olettakaamme, että nuotista talletetun informaation pohjalta haluttaisiin tuottaa nuottikuva, joka esittäisi talletettua teosta transponoituna uuteen sävellajiin. On suhteellisen helppoa kirjoittaa tietokoneohjelma, joka muuntaa uuteen sävellajiin kuvauksen teoksesta, mikäli kuvaamiseen on käytetty sellaisia käsitteitä kuin kappaleen sävellaji tai nuottien korkeudet. Jos sen sijaan ollaan kuvattu teoksen nuottikuva rasterigrafiikkana, on varsin haastavaa laatia tietokoneohjelma, joka muokkasi tätä esitystä siten, että teos saataisiin uuteen sävellajiin. Jos teos on talletettu vektorigrafiikkana, tehtävä on helpompi, mutta edelleen haastava verrattuna tilanteeseen, jossa ollaan talletettu teos musiikillisin käsittein.

2.4 Digitaalisten nuottikokoelmien julkaiseminen

Digitaaliseen muotoon, bittien jonoiksi saatetut nuotit voidaan tarjota käyttäjille monin eri tavoin. Seuraavassa esitellään lyhyesti kolme esimerkkiä, joista kukin edustaa erilaista lähestymistapaa. Kutakin esimerkkiä on käsitelty yksityiskohtaisemmin luvussa 3.

Suomen evankelis-luterilaisen kirkon verkkosivustolla on verkkovirsikirja [ver], joka soveltuu esimerkiksi verkkopalveluna toteutetusta sähköisestä nuottijulkaisusta. Palvelussa virsinuotteja voi katsella verkkoselaimessa samaan tapaan kuin mitä tahansa verkkosivujen sisältämää teksti- ja kuvamateriaalia. Tämän lisäksi nuotteja voi halutessaan ladata omalle tietokoneelle Microsoftin Word-tekstinkäsittelyohjelman .doc-tiedostoina.

Gracevoice Oy on julkaissut oman sähköisen virsikirjatoteutuksensa, eVir-sikirjan [Graa], epub-muodossa. epub on sähköisten kirjojen julkaisemiseen tarkoitettu tiedostoformaatti. Tässä formaatissa julkaistuja kirjoja voidaan lukea erityisesti sähköisten kirjojen lukemiseen tarkoitetuilla laitteilla sekä tietokoneilla ja älypuhelimilla, joissa on sähköisten kirjojen lukemiseen tarkoitettu ohjelma. eVir-sikirjan voi ladata Gracevoice Oy:n verkkokaupasta, mutta tuotetta ei käytetä verkkoselaimen avulla, eikä sen käyttäminen vaadi Internet-yhteyttä. Tuotteen voi tilata CD-ROM -levyllä, joten edes tuotteen hankkimiseen ei Internet-yhteyttä välttämättä tarvita.

Nuotteja voidaan julkaista myös käyttäjän koneelle asennettavan itsenäisen tietokoneohjelman muodossa. Tällainen julkaisu nivoo yhteen julkaistavan nuottimateriaalin ja tietokoneohjelman, jolla materiaalia luetaan. Esimerkiksi tästä soveltuu MusicEase Software -yhtiön tietokoneohjelma Christian Virtual Hymnal [Musc], joka sisältää useiden laulukirjojen materiaalia ja lisäksi toiminnot, joilla tästä materiaalista voidaan tuottaa käyttäjän kulloistakin tarvetta vastaavia nuottikuvia. Ohjelman ilmaisen kokeiluversion voi ladata MusicEase Softwaren verkkosivuilta.

2.5 Nuottien julkaiseminen verkossa

Verkossa julkaiseminen ylipäätään voi tapahtua monin eri tavoin. Vuosituhannen vaihteessa, tästä projektista vastaavan opiskellessa tietojenkäsittelytiedettä, aineiston julkaiseminen verkossa saattoi tapahtua siten, että tekstieditoriohjelmalla kirjoitettiin HTML-koodia, jossa varsinaisen tekstin sekaan lisättiin HTML-tageja, jotka määrittelivät, oliko kussakin tekstikatkelmassa kyseessä esimerkiksi otsikko, tekstikappale vai linkki toiseen verkkosivuun. Näin syntynyt tekstitiedosto talletettiin sitten verkkopalvelimen tiedostojärjestelmässä oikeaan paikkaan, jolloin tiedostoon pääsi käsiksi Internetin kautta kirjoittamalla verkkoselaimeen tiedoston Internet-osoitteen. Tällöin sivun julkaisijalla piti olla mahdollisuus tallettaa tiedostoja palvelimelle. HTML-tagien avulla voitiin sivuille liittää myös kuvia. Kuva, joka haluttiin osaksi verkkosivua, talletettiin palvelimelle, ja verkkosivun tiedostoon kirjoitettiin HTML-tag, joka sisälsi tiedon kuvatiedoston sijainnista palvelimella. Kuva saattoi luonnollisesti esittää esimerkiksi nuottikirjoitusta. Jotta verkkoselai-

met osaisivat näyttää kuvan osana sivua, on kuvatiedoston oltava muodossa, jota verkkoselain tai selaimeen asennettu liitännäinen tukee. Useimmat selaimet tukevat JPEG-, GIF- ja PNG-rasterigrafiikkaformaatteja. Vektorigrafiikkamuotoja tuetaan huomattavasti huonommin. SVG-vektorigrafiikkamuotoa jotkut selaimet tukevat osittain. [ima]

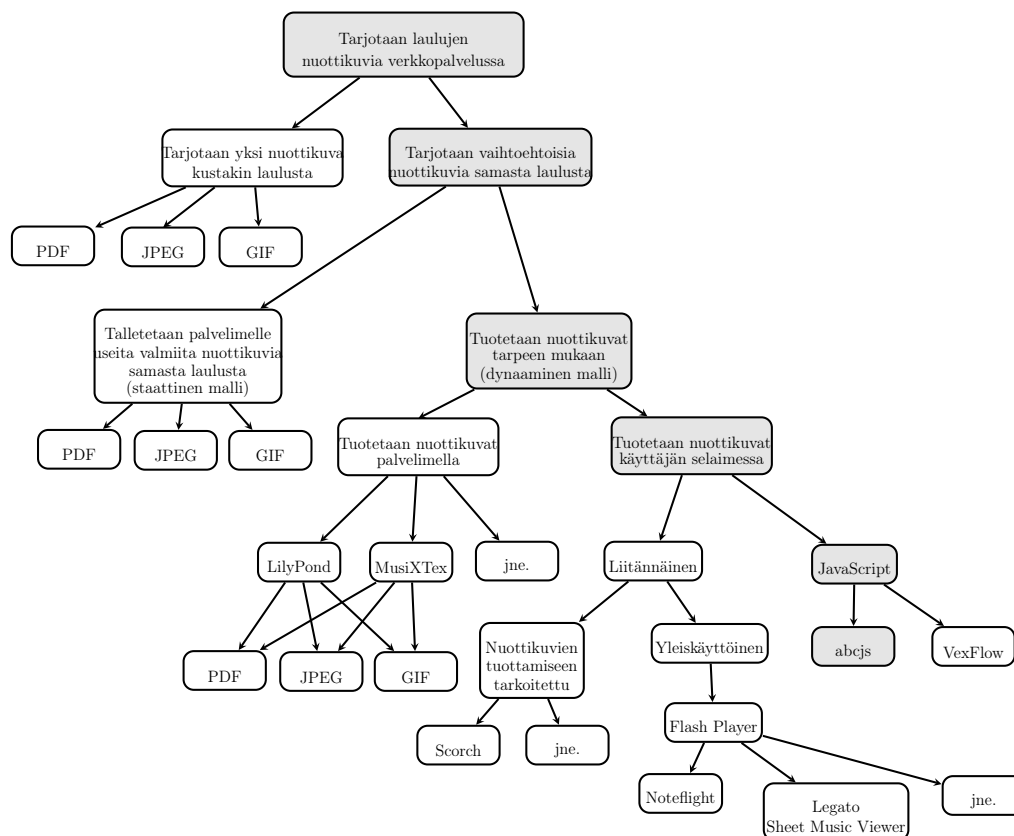
Nykyisin aineiston julkaiseminen verkossa tarkoittaa usein jonkin verkkopalvelun käyttämistä. Verkossa on lukuisia palveluita esimerkiksi blogien pitämistä varten, ja Facebookin ja muiden yhteisöpalveluiden käyttäminen on myös omanlaistaan verkossa julkaisemista. Tällaisia palveluita käytettäessä aineiston tuottajat eivät kirjoita HTML-koodeja, vaan aineiston saattaminen verkkoselainten ymmärtämään, HTML-koodattuun muotoon jää palvelun ohjelmiston tehtäväksi.

Teknisesti olisi mahdollista luoda Facebookiin sivu Virsikirjalle ja skannata virsien nuotteja tälle sivulle.³ Eri nuotinnusohjelmiin puolestaan liittyy sivustoja, joilla julkaistaan näillä nuotinnusohjelmilla kirjoitettuja nuotteja. Esimerkiksi Sibeliuksen nuotinnusohjelmaan liittyy Score Exchange -sivusto [scob], jonka kautta Sibeliuksella kirjoitettuja nuotteja voi levittää maksua vastaan tai ilmaiseksi. On olemassa myös nuottien julkaisemiseen keskittyneitä verkkopalveluita, jotka eivät liity mihinkään tiettyyn nuotinnusohjelmaan. Näistä mainittakoon esimerkiksi tekijänoikeusvapaiden nuottien jakeluun keskittynyt Petrucci Music Library, joka 22.2.2013 sisälsi 230,455 nuottijulkaisua [petb]. Kuka tahansa voi luoda palveluun itselleen käyttäjätunnukset uusien nuottien lisäämiseksi palveluun. Palvelu on toteutettu MediaWiki-järjestelmän avulla. Samalla järjestelmällä on toteutettu myös Wikipedia. Petrucci Music Libraryn nuottikuville käyttämä pääasiallinen tiedostomuoto on nykyisin PDF [peta]. PDF-tiedosto voi sisältää niin rasteri- kuin vektorigrafiikkaakin [vec]. PDF-formaattia käytetään yleisesti kaikenlaisen aineiston sähköiseen julkaisemiseen ja sille on hyvä tuki eri laite- ja ohjelmistoympäristöissä. PDF-formaatissa julkaistua nuottia on kuitenkin vaikea muokata.

Suunniteltaessa uutta verkkopalvelua laulun nuottien julkaisemista varten joudutaan tekemään useita valintoja. Halutaanko samasta laulusta tarjota vaihtoehtoisia nuottikuvia? Missä muodossa laulut talletetaan järjestelmään? Jos laulut talletetaan muussa kuin kuvallisessa muodossa, miten nuottikuvat tuotetaan? Kuvassa 2.1 esitetään, miten tehtävät valinnat johtavat uusiin valintoihin. Kuvassa on korostettu harmaalla taustavärillä vaihtoehdot, jotka

³Osa Virsikirjan lauluista on tekijänoikeuksin suojattuja, samoin kuin painetuissa virsikirjaeditioissa julkaistut nuottikuvat. Näiden oikeuksien omistajilta olisi siis saatava lupa. Virsikirja skannaamisen sijasta olisi mahdollista laatia virsistä uudet nuottikuvat, jolloin virsikirjan nuottigrafiikan tekijänoikeuksien haltijalta ei tarvittaisi lupaa. Nuottikuvista voitaisiin tässäkin tapauksessa tehdä esimerkiksi JPEG-muotoisia kuvatiedostoja, joita Facebook käsittelee kuin mitä tahansa kuvia.

tässä projektissa toteutettua verkkopalvelua laadittaessa valittiin. Seuraavassa tekstissä käydään läpi samoja päätöksiä, joita kuvan päätöspuu esittää.



Kuva 2.1: Nuottikuvien tarjoaminen verkkopalvelussa

Mikäli järjestelmässä halutaan tarjota vain yhdenlainen nuottikuva kuhunkin lauluun, kannattaa nuottikuvat tallettaa järjestelmään jossain yleisesti kuvien tallettamiseen käytetyssä tiedostomuodossa, jota verkkoselaimet tukevat hyvin. Kyseeseen tulisivat ainakin PDF⁴, JPEG ja GIF. Tämän jälkeen näitä kuvia voidaan tarjota verkossa samoilla tekniikoilla, joita käytetään esimerkiksi valokuvien tarjoamiseen verkkopalveluissa.

Mikäli kuitenkin halutaan toteuttaa verkkopalvelu, joka tarjoaisi samasta laulusta vaihtoehtoisia nuottikuvia esimerkiksi eri sävellajeissa, tämä voidaan toteuttaa kahdella vaihtoehtoisella tavalla. On mahdollista tallettaa järjestelmään useita valmiita nuottikuvia samasta laulusta tai on mahdollista

⁴PDF-tiedostojen näyttäminen voi edellyttää liitännäisen käyttämistä verkkoselaimessa.

tallettaa laulut jossain ei-kuvalisessa muodossa, josta tuotetaan nuottikuvia käyttäjän toiveiden mukaan vasta, kun käyttäjä pyytää nuottikuvaa järjestelmältä. Edellistä ratkaisumallia kutsutaan jatkossa staattiseksi, jälkimmäistä dynaamiseksi.⁵ Staattisen mallin mukainen järjestelmä voidaan teknisesti toteuttaa samoin kuin järjestelmä, joka tarjoaa vain yhden nuottikuvan kuhunkin lauluun.

Mikäli halutaan tarjota käyttäjälle mahdollisuus tuottaa palveluun sisältöä, joka on huomioitava nuottikuvissa, esimerkiksi omia soinnutuksia, ei staattista strategiaa voida soveltaa. Staattinen strategia johtaa myös helposti tarpeeseen tallettaa palvelimelle valtava määrä nuottikuvia⁶. Valittaessa dynaaminen vaihtoehto, kannattaa nuotti-informaatio tallettaa järjestelmään muodossa, jossa olevaa informaatiota on helppo käsitellä ja josta informaatio saadaan helposti muotoon, jota nuottikuvien tuottamiseen käytettävä ohjelmisto osaa tulkita. Tähän palataan jäljempänä.

Toimittaessa dynaamisen mallin mukaan jää valittavaksi, tapahtuuko nuottikuvien tuottaminen palvelimella vai käyttäjän päätelaitteessa. Nuottikuvien tuottamisessa palvelimella on se hyvä puoli, että ohjelman, joka nuottikuvat tuottaa, tarvitsee toimia vain siinä tietyssä palvelinympäristössä, jossa ohjelmaa päätetään suorittaa. Nuottikuvat voidaan tuottaa johonkin yleisesti käytettyyn kuvatiedostomuotoon, jota kaikki yleisimmin käytetyt verkkoselaimet tukevat, jolloin käyttäjän verkkoselaimelle asetetaan vain vähän vaatimuksia. Tämän ratkaisun huono puoli on, että uusien nuottikuvien tuottaminen vaatii aina viestintää palvelimen kanssa. Ratkaisu tukee siis huonosti tilanteita, joissa käyttäjä haluaisi ladata aineistoa omalle päätelaitteelleen ollessaan verkkoyhteyden piirissä (esimerkiksi seurakuntakeskuksella) ja tuottaa siitä uusia nuottikuvia esimerkiksi uusiin sävellajeihin menettettyään verkkoyhteyden (esimerkiksi leirikeskuksessa). Lisäksi nuottikuvien tuottamisesta koituu palvelimelle kuormitusta, mikä nostaa palvelimen ylläpitäjän kustannuksia ja saattaa johtaa palvelun hidastumiseen tilanteissa, joissa palvelulla on paljon yhtäaikaista käyttäjiä. Palvelun käyttäminen saattaa myös tuntua hieman tahmealta, koska uuden nuottikuvan ilmestymisnopeus on riippuvainen verkkoyhteyden sujuvuudesta.

Mikäli nuottikuvat tuotetaan käyttäjän päätelaitteessa, voidaan järjestelmä toteuttaa siten, että aiemmin ladatuista lauluista voidaan tuottaa uusia

⁵Jakoa staattiseen ja dynaamiseen käytetään yleisemminkin verkkosivuista puhuttaessa [sta, dyn].

⁶Näin tapahtuu esimerkiksi, jos halutaan, että käyttäjä voi valita haluamansa säkeistöt nuottia seuraillemaan. Jos laulussa on esimerkiksi kymmenen säkeistöä, voidaan näistä valita $2^{10} - 1 = 1023$ erilaista yhdistelmää. Jos halutaan lisäksi tarjota käyttäjälle laulun nuotti missä tahansa sävellajissa, tarvitaan jo $1023 * 12 = 12276$ erilaista nuottikuvaa yhdestä laulusta.

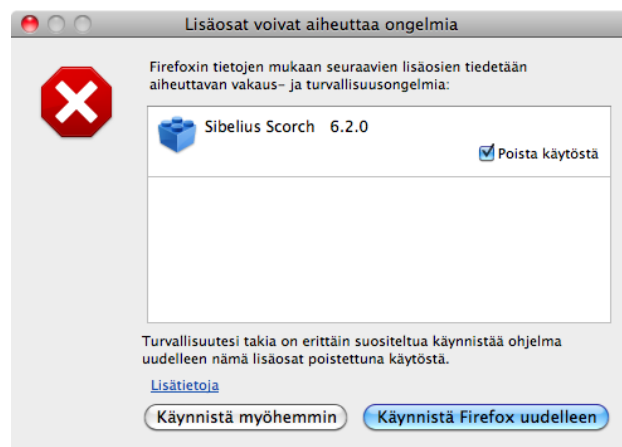
nuottikuvia, vaikka yhteys palvelimeen oltaisiin jo menetetty. Koska uuden nuottikuvan tuottaminen ei vaadi viestintää palvelimen kanssa, ei verkkoviestistä aiheudu viivettä uuden nuottikuvan ilmestymiseen käyttäjän suorittua muokkausoperaation, ja käyttökokemus säilyy jouhevampana. Ongelmana tässä ratkaisussa on, että käyttäjien verkkoselaimet tukevat vaihtelevasti eri tekniikoita, joilla nuottikuvien tuottaminen voitaisiin toteuttaa, eikä yksikään verkkoselain tarjoa niin kattavaa tukea eri tekniikoille kuin palvelinympäristö tarjoaisi.

Nuottikuvien tuottamiseen palvelimella on tarjolla useita avoimen lähdekoodin ohjelmistoja. LilyPond [lilc] ja MusiXTeX [musb] kykenevät hyvinkin monimutkaisten nuottikuvien tuottamiseen laadukkaasti. Nuottikuvien tuottamiseksi käyttäjän verkkoselaimessa voidaan hyödyntää joko verkkoselaimen liitännäisiä, joita selaimet tukevat vaihtelevasti, tai JavaScript-ohjelmointikieltä, jolla kirjoitettuja ohjelmia voidaan suorittaa yleisimmin käytetyissä verkkoselaimissa. Liitännäiset voidaan nuottien julkaisemisen kannalta jakaa kahteen kategoriaan: erityisesti nuottikuvien piirtämiseen tarkoitettuihin liitännäisiin ja yleiskäyttöisiin liitännäisiin, joiden avulla voidaan toteuttaa verkkosivuille monenlaista toiminnallisuutta, myös nuottikuvien piirtoa. Edelliseen kategoriaan kuuluu muun muassa Sibelius-nuotinnusohjelmilla tuotettujen nuottien selaimessa piirtämiseen tarkoitettu Scorch-liitännäinen [scoa], jälkimmäiseen Adobe-yhtiön Flash Player -liitännäinen [fla], joka on asennettavissa useisiin verkkoselaimiin ja jota käytetään hyvin monilla verkkosivuilla.

Verkkoselainten uudet versiot saattavat vaatia päivityksiä liitännäisiin. Valittaessa julkaisutekniikoita, jotka perustuvat liitännäiseen, tullaan siis riippuvaiseksi liitännäisten ylläpidosta. Seuraava tapauskuvaus havainnollistaa asiaa.

Tämän projektin kuluessa saatiin Firefox-verkkoselaimelta kuvassa 2.2 näkyvä viesti, jossa Firefox kehotti poistamaan Scorch-liitännäisen käytöstä, koska sen ”tiedetään aiheuttavan vakaus- ja turvallisuusongelmia”. Liitännäinen poistettiin käytöstä. Firefoxin takana oleva yhtiö tiesi verkkosivuillaan [scoc] kertoa, että Scorch-liitännäinen ei ole yhteensopiva Firefoxin version 18 ja myöhempien versioiden kanssa vaan kaatuu aina, kun ladataan noiden Firefoxin versioiden kanssa. Ilmeisesti kyse oli siis pikemminkin vakaus- kuin turvallisuusongelmasta. Koska liitännäistä käyttävään verkkosivuun haluttiin kuitenkin tutustua, yritettiin sivua katsella Safari-selaimella. Verkkosivun kohtaan, johon nuottikuvan olisi pitänyt ilmestyä, ilmestyi teksti, jossa kerrottiin, että Scorch-liitännäinen ei ole yhteensopiva selaimen kanssa ja ehdotettiin vaihtamaan Firefox-selaimen. Scorch-liitännäisen tuotta-

mut Avid-yhtiö kertoo verkkosivuillaan [run], että Scorch on yhteensopiva ainoastaan sellaisten Safari-selainten kanssa, joiden versio on aiempi kuin 5.1. Koneella, jolla sivua yritettiin katsoa, oli asennettuna Firefoxin versio 19.0 ja Safarin versio 5.1.7, joten Scorch ei toiminut kummankaan näistä kanssa. Scorch ei suostunut toimimaan myöskään Chrome-selaimen versiossa 25.0.1364.99. Enempää verkkoselaimia käytetylle koneelle ei ollut asennettuna. Mikäli tavoitteena on verkkojulkaisu, jota mahdollisimman monien olisi mahdollisimman helppo käyttää, voidaan Scorch-liitännäisen puutteellista yhteensopivuutta verkkoselainten kanssa pitää syynä olla perustamatta ratkaisua ainakaan tähän liitännäiseen.



Kuva 2.2: Ongelmia liitännäisestä

Mitä useamman verkkosivun katseleminen edellyttää tiettyä liitännäistä, sitä todennäköisemmin se on asennettuna käyttäjien selaimiin. Adobe-yhtiön Flash Player -liitännäinen on hyvin laajasti käytetty ja asennettu verkkoselaimiin. Tämä liitännäinen ei liity kiinteästi mihinkään tiettyyn sovellusalueeseen, kuten Scorch-liitännäinen liittyy nuottikirjoitukseen. Sen sijaan Flash Playerin avulla on mahdollista toteuttaa verkkosivuille periaatteessa mitä tahansa toiminnallisuutta, esimerkiksi nuottien esittämiseen liittyvää.

Flash Playerin avulla on toteutettu useita nuottikuvien tuottamiseen tarkoitettuja ratkaisuja. Noteflight on Flasp Playeriin perustuva verkkoselaimessa toimiva nuotinnusohjelma [nota], johon liittyy myös väline nuottien upottamiseksi osaksi verkkosivua. Noteflightin avulla voidaan upottaa verkkosivulle paitsi ohjelmalla itsellään tuotettuja nuotteja, myös MusicXML-tiedostomuotoon saatettuja nuotteja [notb]. Tällöin nuottikuvan piirtämisestä verkkosivulle vastaa Noteflight-ohjelmisto. Myös Legato Sheet Music

Viewer -ohjelmisto on toteutettu Flash playerin avulla. Kyseessä ei ole nuotinnusohjelma vaan ainoastaan muilla ohjelmilla tuotettujen ja MusicXML-tiedostomuodossa tallennettujen nuottien näyttämiseen ja kuuntelemiseen tarkoitettu ohjelmisto, joka mahdollistaa myös nuottien transponoimisen [leg]. Flash-liitännäistäkään ei kuitenkaan ole mahdollista asentaa kaikkiin päätelaitteisiin, jotka soveltuisivat sähköisen laulukirjan laitealustaksi. Esimerkiksi Apple-yhtiön iPad-taulutietokoneisiin ei ole mahdollista asentaa Flash Player -liitännäistä, vaikkakin on luotu ratkaisuja, joiden avulla Flash-sisältöä on mahdollista käyttää iPadissä [Nik].

Verkkosivuille on mahdollista toteuttaa toiminnallisuutta myös JavaScript-ohjelmointikielen avulla. JavaScriptiä tukevat erittäin monet verkkoselaimet, myös Applen iPad-taulutietokoneen verkkoselain⁷. JavaScriptiä on sovellettu nuottikuvien liittämiseen verkkosivuille varsin vähän. On kuitenkin olemassa ainakin kaksi avoimen lähdekoodin ohjelmistokehitysprojehtia, joissa tuotetaan JavaScript-ohjelmakoodia nuottikuvien piirtämiseen: abcjs ja VexFlow [abcb, vex].

Mikäli nuottikuvat tuotetaan dynaamisesti, ei lauluja kannata tallettaa järjestelmään kuvien tallettamiseen tarkoitettussa muodossa, kuten JPEG-tai GIF-muodossa. Pohdittaessa erilaisten, vaihtoehtoisten muotojen soveltuvuutta laulujen talletusmuodoksi, voidaan kutakin ehdolle otettua muotoa arvioida esimerkiksi seuraavien kriteerien perusteella:

1. Onko laulut on helppo saattaa valittuun muotoon?
2. Soveltuuko valittu muoto soveltuu sellaisenaan syötteen ohjelmistoil- le, jotka tuottavat nuottikuvat?
3. Onko valittua muotoa on helppo käsitellä tietokoneella esimerkiksi lau- lun transponoimiseksi?
4. Onko valitusta muodosta helppo tuottaa tietokoneella ohjelman sisäi- nen esitysmuoto, esimerkiksi oliomalli⁸, jota on helppo käsitellä esimer- kiksi laulun transponoimiseksi, ja josta on helppo kääntää informaatio muotoon, joka nuottikuvien tuottamiseen käytetty ohjelmisto kykenee tulkitsemaan?
5. Onko valittu esitystapa tiivis? Mikäli nuottikuvat tuotetaan palvelimel- la, tiiviydellä on merkitystä vain tallennustilan tarpeen kannalta. Mi- käli tallennusmuoto on samalla tiedonsiirtomuoto palvelimen ja pää- telaitteen välillä, on tiiviydellä merkitystä myös tiedonsiirron määrän kannalta.

⁷Tämän kirjoittaja varmisti kokeilemalla.

⁸Olion käsite määritellään luvussa 6.2.1

Luvussa 5 esitetään tässä projektissa suunniteltu menetelmä virsikirjan nuottikuvien informaation koodaamiseen tekstin muodossa. Menetelmä ei täytä kriteeriä 2 eikä kovin hyvin myöskään kriteeriä 3, mutta näiden puutteiden merkitystä vähentää, että menetelmä täyttää kriteerin 4. Kriteerin 5 menetelmä täyttää kohtalaisen hyvin. Kriteerin 1 menetelmä täyttää sikäli, että tekstieditoreja on saatavilla runsaasti.

Luku 3

Sähköisesti julkaistuja laulukokoelmia

3.1 Virsikirjajulkaisuja

3.1.1 Verkkovirsikirja

Suomen evankelis-luterilaisen kirkon Virsikirja on julkaistu kirkon verkkosivuilla [ver]. Tästä julkaisusta käytetään jatkossa nimeä Verkkovirsikirja. Tällä sivustolla virsien nuotit esitetään matalaresoluutioisina gif-kuvina. Nuottien tulostamista varten sivusto tarjoaa .doc-tiedostoja, joissa kuvien laatu on parempi. Sivusto ei tarjoa virsiin muussa mielessä vaihtoehtoisia nuottikuvia. Sivusto tarjoaa kuitenkin hakukoneen, jonka avulla virsiä voi etsiä käyttämällä hakuehtona kaikkia virren yhteydessä esiintyviä tekstejä, kuten virren numeroa, sanoituskatkelmaa tai sanoittajan tai säveltäjän nimeä. Hakuehtoja voi yhdistellä loogisten operaattorien avulla¹. Palvelu tarjoaa virsikirjaan myös monipuoliset hakemistot, joissa virret on luetteloitu osastoittain, numerojärjestyksessä, aakkosjärjestyksessä, säveltäjittäin, sanoittajittain sekä raamatunkohtien ja liittämivuotien mukaan.

Virsien yhteydessä on linkkejä Tauno Väinölän kirjoittamiin taustatietoihin. Verkkovirsikirja ja kirkon sivuilla julkaistu Evankeliumikirja on linkitetty kaksisuuntaisesti: Evankeliumikirjassa on kunkin kirkkopyhän yhteydessä linkit virsiin, joita suositellaan alku- ja päivän virsiksi, ja Verkkovirsikirjassa on virsien yhteydessä linkit kirkkopyhiin, joihin virret liittyvät. Verkkovirsikirjassa on linkkejä myös Suomen vanhat virsisävelmät -sivustolle [Siba], jos-

¹Hakuohjeissa mainitaan esimerkkinä haku ”armo AND Hemminki”, jolla löytyvät virret, joiden yhteydessä esiintyy sekä sana *armo* että nimi *Hemminki*, ja haku ”armo OR Hemminki”, jolla löytyvät virret, joiden yhteydessä esiintyy joko sana *armo* tai nimi *Hemminki*.

sa pääsee tutustumaan virsien sävelmien vanhoihin käsikirjoituksiin. Sivusto kuuluu osana Sibelius-Akatemian kirkkomusiikin osaston toteuttamaan Virtuaalikatedraali-sivustoon [Sibb].

Kaikkia Virsikirjan virsiä ei tekijänoikeussyistä ole julkaistu Verkkovirsikirjassa. Esimerkiksi virrestä 31 palvelu sisältää ainoastaan tekstin. Melodiasta todetaan, että ”Virren sävelmän tekijänoikeuksien haltija Fennica Gehrman Oy ei ole antanut lupaa tämän virren sävelmän julkaisemiseen verkossa ”.

3.1.2 eVirsikirja

Gracevoice-yhtiö on julkaissut Virsikirjan kaikkine virsineen maksullisena, Suomalaiseksi eVirsikirjaksi nimettynä tuotteena [Graa]. Kyseessä ei ole verkkopalvelu vaan ePub-tiedostoformaattiin perustuva sähköinen kirja, joka tosin voidaan ostaa Gracevoice-yhtiön verkkokaupasta [Grab]. ePub-kirjoja on mahdollista lukea erityisillä sähköisen kirjan lukemiseen kehitetyillä laitteilla, mutta myös tietokoneilla ja matkapuhelimilla, joissa on ePub-formaattia tukeva sähköisen kirjan lukemiseen tarkoitettu sovellusohjelma.

eVirsikirja ei mahdollista nuottikuvien muokkaamista lukuunottamatta ominaisuutta, joka mahdollistaa nuottikuvien muodon sopeuttamisen käytettävän päätelaitteen näyttöruudun muotoon liittämällä tarvittaessa joka toisen nuottiviivaston edellisen viivaston jatkoksi².

Gracevoice nostaa eVirsikirjan markkinoinnissaan [Graa] esiin seuraavia julkaisumediaan liittyviä seikkoja:

- Tekstikoon voi säätää itselle sopivaksi.
- Julkaisu toimii monissa päätelaitteissa, myös älypuhelimissa, joten sitä on helppo pitää aina mukana.
- Virren löytäminen numeron perusteella on nopeaa, koska selaamisen sijaan riittää näppäillä numero.
- Virren voi heijastaa videotykin avulla kankaalle. Mahdollisuus säätää tekstikoko on tällöinkin hyödyllinen.

3.1.3 virret.net-palvelu

Kotimaa-yhtiöt ylläpitävät virret.net-palvelua [Ky], jota sivuston pääsivulla kuvataan seuraavasti:

²Gracevoicen Joni Nisula esitteli kirjoittajalle tämän toiminnon Kirjamesseilla Helsingin Messukeskuksessa 28.10.2011.

Virret.net -palvelusta voit etsiä haluamasi virren nimen ja alkusanojen perusteella. Voit myös kuunnella virren alun.

Epäselväksi jää, mitä virren nimellä tarkoitetaan. Virsikirjoissa virret esiintyvät yleensä ainoastaan numeroituina. Esimerkiksi virsi 571 tunnetaan kyllä Suvivirren nimellä, mutta palvelu ei löydä mitään hakusanalla Suvivirsi. Virren alulla tarkoitetaan ilmeisesti sitä, että virren melodia kuullaan palvelussa ainoastaan kertaalleen. Ainakin virren 571 melodia soi palvelussa kokonaisuudessaan lukuunottamatta kertauksen huomiotta jättämistä.

Kuuntelu on teknisesti toteutettu siten, että kun virteen liittyvää kuuntele-linkkiä klikkaa, palvelimelta siirtyy käyttäjän tietokoneelle midi-tiedosto, jonka sisältämän informaation perusteella tietokoneen sisäänrakennettu syntetisaattori soittaa virren. Palveluun ei siis ole tallennettu ääninäytteitä. Nuotinnusohjelmat lukevat yleensä midi-tiedostoja, joten palvelusta on hyötyä laadittaessa virrestä omaan käyttöön räätälöity nuotti nuotinnusohjelman avulla. Jos esimerkiksi haluaa virrestä nuotin Virsikirjan nuottikuvasta poikkeavassa sävellajissa, voi käydä hakemassa virren midi-tiedostona tästä palvelusta, avata tiedoston nuotinnusohjelmalla (esimerkiksi Sibeliuksella tai Finalella), käyttää ohjelman transponointitoimintoa ja tulostaa näin syntyneen nuotin.

3.1.4 Suomen Rauhanyhdistysten Keskusyhdistyksen sähköiset laulukirjat

Suomen Rauhanyhdistysten Keskusyhdistyksen (josta myöhemmin käytetään lyhennettä SRK) verkkokaupassa [rkr] on myynnissä useita sähköisiä laulukirjajulkaisuja, jotka perustuvat eri tekniikoihin. Verkkokauppa tarjoaa linkit Android-, iOS- ja Windows-käyttöjärjestelmillä varustettujen puhelimien omiin kauppapaikkoihin sivuille, joilla myydään sovellusta, joka sisältää SRK:n kustantaman Siionin laulut -kokoelman sekä Suomen evankelisluterilaisen kirkon Virsikirjan laulujen sanat. Melodioita julkaisu ei sisällä. Suoraan omassa verkkokaupassaan SRK myy vanhempiin puhelimiin tarkoitettua Mobiililaulukirja-sovellusta. Tämän sovellus oli mahdollista asentaa kirjoittajan omaan Nokia C2-01 -puhelimeen, joten kirjoittaja pääsi tutustumaan sovellukseen käytännössä. Tämä oli tarpeen, koska sovelluksesta ei kerrota verkkokaupassa juuri mitään. Sovelluksen sai kuitenkin ladata maksuttomaan kokeilukäyttöön vuorokauden ajaksi.³

³Kirjoittajan ladatessa sovellusta puhelimeensa Opera Mini -selaimella esitti selain kysymyksen: "Sovellus ei ole luotetulta toimittajalta. Haluatko jatkaa?" Hieman siviilirohkeutta sovelluksen lataamiseen siis vaadittiin.

Laulut löytyvät sovelluksesta kirjoittamalla laulun numero tai hakusana. Kirjoittaja kokeili hakusanaominaisuutta kirjoittamalla hakusanaksi ”vaiht”. Noin kahden sekunnin kuluttua puhelimen näytölle ilmestyi lista, joka sisälsi seitsemän Siionin laulua ja kaksikymmentäkuusi virttä. Mukana oli esimerkiksi virsi 428, jossa merkkijono ”vaiht” esiintyy neljännessä säkeistössä (”Tuomme vaihtopöydälle lahjamme ja puuttemme.”)

Verkkokaupassa tarjotaan Siionin lauluja ja virsiä myös ruotsiksi. Windows-käyttöjärjestelmällä varustetuille tietokoneille SRK:n verkkokauppa tarjoaa sovellusta, joka sisältää Siionin laulut ja Virsikirjan useilla eri kielillä. Ohjelmasta ei kerrota verkkokaupassa juuri mitään, mutta sivulla olleista kuvaruutukaappauksista päätellen tämäkään julkaisu ei sisällä nuotteja.

3.2 Muiden laulujen kokoelmia

3.2.1 Christian Virtual Hymnal

MusicEase Software -yhtiö on kehittänyt Christian Virtual Hymnal -nimisen tietokoneohjelman [Musc], johon on tallennettu tekijänoikeusvapaa laulumateriaali useista eri hengellisistä laulukokoelmista⁴ reaalisointumerkkeineen ja neliäänisine satseineen. Ohjelma kykenee esittämään nuotit missä tahansa sävellajissa, ja käyttäjä voi valita, mitä elementtejä (sointumerkit, neliääniset satsit, säkeistöt) nuottikuviin otetaan mukaan. Sanoituksesta valitut säkeistöt saa haluttaessa seuraamaan nuottikuvaa. Ohjelma liittää nuotteihin haluttaessa myös sointuotekaaviot, jotka kertovat kitaristeille, miten soinnut voidaan kitarasta ottaa, sekä nauhallisille kielisoittimille tarkoitetun tabulatuurin, joka kertoo, mitä kieltä on kulloinkin soitettava ja miltä nauhalta kieltä on painettava, jotta melodia saataisiin soitettua. Ohjelma kykenee myös soittamaan nuottikuvat, mikä mahdollistaa ohjelman käyttämisen säestysautomaattina, vaikkakin musiikin hienostunut tulkinta jää ohjelmalta tekemättä.

Christian Virtual Hymnal -ohjelma ei tarjoa käyttäjille mahdollisuutta lisätä ohjelmaan uusia lauluja tai laulukokoelmia. Ohjelma on saatavissa Windows- ja OS X -käyttöjärjestelmille, joten sitä voidaan käyttää sekä PC-että Mac-tietokoneissa. Ohjelmasta voi ladata MusicEase Softwaren verkko-

⁴Ohjelma sisältää lähes kaikki tekijänoikeusvapaat laulut seuraavista kokoelmista: Celebration Hymnal®, Hymnal for Worship and Celebration, 1991 Baptist Hymnal, Lutherin Book of Worship, With One Voice, 1989 United Methodist Hymnal, 1990 Presbyterian Hymnal, New National Baptist Hymnal (21st Century Edition), Christian Life Hymnal ja 1985 Seventh-day Adventist Hymnal.

sivuilta ilmaisen kokeiluversion, joka sisältää ainoastaan hyvin suppean lauluaineiston. Ohjelman rekisteröiminen, joka tuo käyttöön koko aineiston, on maksullista.

3.2.2 iReal b

iReal b [Tec] on tietokoneohjelma, johon voi verkosta ladata reaalisointumerkein kirjoitetut soinnutukset tuhansiin eri tyyliisiin kappaleisiin. Ohjelma osaa transponoida soinnutukset mihin tahansa sävellajiin. Ohjelma osaa myös soittaa soinnutuksia pianosta, bassosta ja rummuista koostuvalla virtuaalisella bändillä. Haluttu kappale löytyy ohjelmasta (ainakin Mac OS X -versiossa, jota tämän kirjoittamisen yhteydessä kokeiltiin) kappaleen tai säveltäjän nimen sekä tyylin perusteella. Ohjelma on saatavissa älypuhelimissa ja sormitietokoneissa käytettäviin iOS- ja Android-käyttöjärjestelmiin sekä Applen pöytä- ja sylimikroissa käytettävään Mac OS X -käyttöjärjestelmään.

Luku 4

Uuden sähköisen virsikirjajulkaisun laatiminen

4.1 Palveltavat henkilöt ja tilanteet

Suunniteltaessa uutta tuotetta tai palvelua, tässä tapauksessa uutta sähköistä Virsikirjaa, on syytä määritellä, keille tuotetta tai palvelua ollaan suunnittelemassa. Tässä projektissa laadittu sähköinen virsikirjajulkaisu on suunnattu henkilöille, jotka säestävät virsiä tai ohjaavat virsien säestämistä. Tällaisia henkilöitä ovat esimerkiksi kanttorit, kirkon nuorisotyön- ja lapsityön ohjaajat sekä soittotaitoiset, virsilaulua harjoittavat maallikot. Erityisesti on ajateltu henkilöitä, joiden soitto-, soinnutus- ja transponointitaidossa on puutteita, joilla on suppea ääniala, ja jotka eivät välttämättä osaa virsiä ulkoa kovin hyvin.

Tilanteina, joita tässä projektissa laadittu sähköinen virsikirja voisi palvella, on ajateltu messuja, joissa virsiä jatkuvasti lauletaan, sekä rippileirejä ja muita tilanteita, joissa virsien säestämiseen osallistuu henkilöitä, jotka eivät ole ammattimuusikkoja, ja joissa paikalla ei välttämättä ole ketään musiikin ammattilaista. Seuraavassa esitetään muutama kuvitteellinen tilanne, jollaisia on ajateltu projektissa laadittua virsijulkaisua suunniteltaessa.

Ollaan rippileirillä. Mukana on yksi isonen, joka soittaa auttavasti kitaraa. Käy ilmi, että myös pari leiriläistä osaa ottaa kitaralla muutamän soinnun. Leirimuusikkona toimiva kesäkanttori haluaisi kaikki kolme mukaan säestämään leirijumalanpalvelusten virsiä. Leirikeskuksen virsikirjojen soinnutukset ovat kuitenkin hankalia varsinkin leiriläisille, ja sävellajitkin hieman korkeita. Kitaristeille pitäisi järjestää nuotit, joissa olisi hieman matalammat sävellajit ja soinnutukset, joita he osaisivat soittaa.

Sointumerkeistä soittamaan tottunut, vapaaseen säestykseen orientoitunut kanttori valmistelee tulevaa messua, jossa hän vastaa yksin laulujen säestämisestä. Osan hän ajatteli säestää uruilla, osan pianolla. Hän haluaisi transponoida muutaman virren uusiin sävellajeihin, jotta helpottaakseen seurakunnan laulua ja jotkut, jotta peräkkäisten laulujen sävellajisuhteet olisivat luontevat. Hän soittaisi mieluiten nuotteista, jotka olisivat valmiiksi oikeassa sävellajissa, ja joissa olisi myös sointumerkit valmiina.

Rippileirillä hengellisen herätyksen kokenut nuori haluaisi leiriltä kotiututtuaan tapailla kitaralla paria rakkaaksi tullutta virttä. Kotoa ei löydy kuin soinnuttamaton virsikirja.

4.2 Toivottavat ominaisuudet

Koska virsikirja on julkaistu sähköisessä muodossa aiemminkin, halutaan tässä projektissa keskittyä erityisesti sellaisten ominaisuuksien toteuttamiseen, joita aikaisemmat sähköiset virsikirjajulkaisut eivät toteuta. Verkkovirsikirjassa on hyvät hakemistot ja hakuominaisuudet, ja se on integroitu virsi-suosituksia kirkkovuoden pyhille tarjoavan Evankeliumikirjan verkkoversion kanssa kahdensuuntaisen linkityksen keinoin. Nämä ovat hyödyllisiä ominaisuuksia sähköiseen virsikirjaan, mutta tässä projektissa laadittuun palveluun niitä ei toteutettu. eVirsikirja ja SRK:n laulukirjat mahdollistavat virsikirjan kuljettamisen puhelimesta aina mukana. eVirsikirja ei tosin osaa transponoida nuottikuviaan eikä useimmissa eVirsikirjan nuottikuvissa ole sointumerkejä. SRK:n laulukirjoissa puolestaan ei ole nuottikuvia lainkaan. Nuottien lukeminen puhelimen pieneltä näytöltä ei kuitenkaan olisi kovin miellyttävää. Tässä projektissa ei keskitytty virsikirjan muuntamiseen puhelimesta toimivaan muotoon, vaikkakin yhtenä kriteerinä käytettyjen teknologioiden valinnassa oli, että järjestelmää olisi ainakin periaatteessa mahdollista käyttää hyvin monenlaisilla päätelaitteilla, myös puhelimilla.

Henkilöitä, jotka säestävät virsien lauluntaa ja jotka kaipaavat soittamisensa tueksi kirjoitettuja harmonisointeja, on ennen tätä projektia palvellut lähinnä paperille painettu aineisto. Neliääninen koraalikirja [kor02] tarjoaa yhden valmiin satsin kuhunkin virteen niille soittajille, jotka lukevat mieluiten stemmoiksi kirjoitettuja harmonisointeja. Niille soittajille, jotka lukevat mieluummin reaalisointumerkein kirjoitettuja soinnutuksia, on tarjolla eri vaihtoehtoja. Virsikirjasta on julkaistu painetussa muodossa kaksi kauttaaltaan reaalisointumerkein soinnutettua versiota. Kirjapajan version [soi07] soinnutukset laativat Kaj-Erik Gustafsson, Juhani Haapasalo, Aarre Joutsenvirta, Timo Kiiskinen ja Pekka Nyman. Aurinko kustannuksen version

soinnutukset laati Kullervo Puumala.

Kirjapajan versiossa sama melodia on toisinaan kirjoitettu eri sävellajeihin sen esiintyessä eri virsien yhteydessä, ja melodia on myös saatettu soinnuttaa eri tavoin. Virsien yhteydessä on viittauksia toisiin virsiin, joissa on sama melodia, ja näiden viitteiden yhteydessä mainitaan, missä sävellajissa melodia näissä rinnakkaisvirsissä on. Lisäksi mainitaan, jos melodia on rinnakkaisvirsissä soinnutettu harvemmin tai tiheämmin.¹

Reaalisointumerkein kirjoitettuja soinnutuksia valikoituihin virsiin on julkaistu muun muassa seuraavissa teoksissa:

- Kitaravirsiä [HEt02]-teos sisältää 140 virttä varustettuna kitarasäestyskäyttöön tarkoitetuilla sointumerkinnoilla.
- Soittajan virsikirjaan [PN02] on valittu 70 virttä, ja virsiin on sointumerkkien lisäksi kirjoitettu kuorosatseja ja säestysideoita bändisoittimille.
- Virsien Kirjaan [VLRN05] on valittu 46 virttä, joihin on soinnutusten lisäksi liitetty muutamia yksinkertaisia kuoro- ja bändisovituksia sekä taustatietoja.
- Virsilöytöretki [EKU09] ”on täynnä tietoa, vinkkejä, tehtäviä ja virikkeitä yli 20 virsikirjamme virrestä”. Virsiä on kirjassa sovitettu eri tavoin, ja virsiin on kirjoitettu myös sointumerkit.

Virsistä on julkaistu erilaisia soinnutuksia eri lähteissä, koska eri soittotyylit, eri soittimet ja erilaisin taidoin varustetut soittajat hyötyvät erilaisista soinnutuksista. Olisi kuitenkin käytännöllistä, jos yksi ja sama lähde tarjoaisi useita eri soinnutuksia samaan virteeseen.

Rytmi- ja musiikkiorientoituneet soittajat säestävät yleensä mieluiten sointumerkeistä, kun taas klassisemmin orientoituneet soittajat soittavat sujuvamin nuotteina kirjoitettuja säestyksiä. Niinpä sekä nuotteina että reaalisointumerkkeinä kirjoitettuja harmonisointeja tarvitaan. Jos sama lähde tarjoaisi molempia, saattaisi tämä ehkä osaltaan hieman silloittaa kuilua kahden musiikkikulttuurin välillä.

Sähköinen virsikirja voi useiden valmiiden harmonisointien lisäksi tarjota käyttäjälle mahdollisuuden kirjoittaa lauluhiin omia harmonisointejaan joko reaalisointumerkein tai neliäänisten satsien muodossa. Verkkopalveluna

¹Esimerkiksi virren 330 yhteydessä on teksti $330 = 464 (F\#m), 528 (t), 539 (Fm/t)$, jolla tarkoitetaan, että virren 330 melodia esiintyy virressä 464 fis-mollissa soinnutettuna samoin kuin virressä 330, virressä 528 taas samassa sävellajissa kuin virressä 330 mutta tiheämmin soinnutettuna, ja virressä 539 f-mollissa tiheämmin soinnutettuna.

toteutettu sähköinen laulukirja voi tarjota käyttäjille mahdollisuuden jakaa kirjoittamansa harmonisoinnit kaikkien palvelun käyttäjien kanssa. Ratkais tavaksi ongelmaksi muodostuu, kuinka välttää siltä, että käyttökelpoiset harmonisoinnit hukkuvat käyttökelvottomien joukkoon, ja kuinka käyttäjät löytävät omiin tarkoituksiinsa soveltuvat harmonisoinnit vaihtoehtojen määrän kasvaessa.

Yhteislauluja säestettäessä on usein tarpeen transponoida laulu, jotta sen ääniala vastaisi paremmin laulajien äänialoja. Sibelius-Akatemian kirkkomusiikin koulutusohjelmakin heijastaa tätä: liturgisen soiton opinnoissa harjoitellaan virsien transponoimista. Transponointitaidoiltaan puutteellista säestäjää auttaisi sähköinen virsikirja, joka osaisi transponoida nuotit automaattisesti.

Painetuissa laulukirjoissa ladotaan usein ensimmäinen säkeistö melodiaa seuraten ja muut säkeistöt melodiasta erilleen. Henkilö, joka laulaa ja soittaa samanaikaisesti joutuu näin ollen ensimmäisen säkeistön jälkeen katsomaan kahteen paikkaan laulukirjassa, elleivät melodia ja soinnutus ole vielä jääneet mieleen, vaan niitäkin on edelleen luettava. Sibelius-Akatemian kirkkomusiikin koulutusohjelman liturgisen soiton tentissä testataankin, kuinka opiskelija tästä suoriutuu. Jos jossain tilanteessa ensimmäistä säkeistöä ei aiota lainkaan laulaa, olisi parempi, että laulun ensimmäisen säkeistön sijaan ensimmäinen laulettava säkeistö ladottaisiin nuottikuvaa seuraten. Sähköinen laulukirja, jossa nuottia seuraavan säkeistön saisi valita, toisi helpotusta kahteen paikkaan katsomisen ongelmaan.

Aloitteleva soittaja ei välttämättä osaa lukea länsimaisen musiikin esittämiseen käytettyä, viiteen viivaan perustuvaa nuottikirjoitusta. Kirjoittaja on rippileireillä kesäkanttorina toimiessaan kirjoittanut kitaran ja basson soittajille tabulatuureja ja otekaavioita, joita heidän on ollut helpompi lukea kuin ”normaalia nuottikirjoitusta”. Tabulatuureissa viivat edustavat kieliä, ja viivoille kirjoitetut numerot otelaudan nauhojen sijaintia otelaudalla.

Reaalisointumerkkien yhteyteen lisätään toisinaan nuottikirjoihin diagrammit, jotka kertovat, miten sointu otetaan kitarasta. Toisinaan nämä lisätään sointumerkkien yhteyteen, toisinaan kirjassa on erillinen liite, johon on kerätty kaikkien kirjassa esiintyvien sointujen otediagrammit. Sähköinen virsikirja voisi antaa käyttäjän valita, haluaako hän otediagrammit nuottikuvaan, ja haluaako hän diagrammit sointumerkkien yhteyteen vai nuottikuvasta erilleen.

Jos sähköisen virsijulkaisun melodiat olisi mahdollista kuunnella, olisi julkaisusta hyötyä myös ihmisille, jotka eivät osaa soittaa mitään soitinta eivätkä laulaa nuoteista, mutta jotka esimerkiksi valitsevat toisinaan virsiä johonkin tilaisuuteen ja haluaisivat saada käsityksen virsien melodioista. Jos käytet-

tävissä olisi myös transponointi, voisi kukin käyttäjä säätää melodian omaa äänialaansa vastaavalle korkeudelle. Tämäkin helpottaisi opettelua.

Musiikinkuunteluohjelmissa, kuten iTunesissa [App] ja Spotifyssä [Spo], käyttäjät voivat muodostaa kappaleista soittolistoja, ja ainakin Spotify mahdollistaa listojen jakamisen kavereiden kanssa. Vastaava toiminnallisuus olisi kätevä lisä sähköiseen virsikirjaan. Kanttori voisi muodostaa sähköiseen virsikirjaansa listan tulevan messun virsistä. Ohjelmassa voisi olla toiminto, jolla listan kaikki virret tulostettaisiin yhdellä kertaa, samoin kuin esimerkiksi iTunes-ohjelmalla voidaan polttaa soittolistan kaikki kappaleet CD-levylle yhdellä komennolla. Jos kanttori soittaisi suoraan urkujen nuottitelineelle asetetun sormitietokoneen näytöltä, tarvittaisiin etukäteen tallennettua virsilistaa, jotta virrestä toiseen pääsisi helposti.

Jos virsilistaan voisi liittää tietoa tilaisuudesta, jossa virret on tarkoitus laulaa, kertyisi listoista käyttökelpoinen aiempien virsivalintojen arkisto, jota voisi hyödyntää, jos ei keksi johonkin tulevaan tilaisuuteen sopivia virsiä. Jos omat listansa voisi jakaa muiden kanssa, pääsisivät kirkkomuusikot hyödyntämään myös kollegojensa aiempia virsivalintoja. Virsilistoihin kannattaisi tallettaa virsien numerojen lisäksi tieto valituista säkeistöistä, sävellajeista ja sovituksista. Järjestelmään voitaisiin haluttaessa rakentaa tilastointiautomaatiikkaa, jolloin saataisiin helposti tietoa siitä, mitä virsiä lauletaan paljon, mitä vähän, millaisissa sävellajeissa, minä sovituksina.

Virsilistojen laatimiseen olisi kätevä kirkoista tuttua virsitaulua vastaava ilmaisumuoto, jossa virsien numerot esitetään allekkain ja virren numeron perässä on säkeistöjen numerot. Kirkkojen virsitauluissa säkeistöjen numerot erottuvat virsien numeroista pienemmän kokonsa avulla. Sähköisen virsikirjan käyttöliittymään siirrettynä sama idea voitaisiin toteuttaa sijoittamalla käyttöliittymään monirivinen tekstikenttä, johon voisi kirjoittaa esimerkiksi seuraavasti:

128:1,3-5
200
234
365:1,2,8
334:3,5,7

Edellä esitetyssä syntaksissa kaksoispiste toimii erottimenä virren numeron ja säkeistöjen numeroiden välillä. Syntaksi voi tukea joustavasti erilaisia virsilistan kirjoittamisen muotoja. Esimerkiksi ranskalaiset viivat rivin alussa voidaan hyväksyä, ja voidaan sallia virsiä toisistaan erottavien rivinvaihtojen korvaaminen esimerkiksi puolipisteillä, tai tapauksissa, joissa säkeistöjä ei esiinny, myös pilkuilla. Siis esimerkiksi:

128:1,3-5; 200; 234; 365:1,2,8; 334:3,5,7

tai

128, 200, 234, 365, 334

Tähän tapaanhan virsiä voidaan luetella vaikkapa sähköpostiviestissä. Inhimillinen lukija ymmärtää kaikki edellä esitetyt muodot, ja tietokoneohjelmaan voidaan toteuttaa jäsentäjä, joka kykenee tulkitsemaan mitä tahansa edellä esitetyistä muodoista.

Työpaikkojen tietoturvasäädökset saattavat kieltää työntekijöitä asentamasta työkoneilleen uusia ohjelmistoja. Jos sähköisen virsikirjan kokeileminen vaatisi yhteydenottoa ylläpitoon, saattaisi kirja jäädä osalta potentiaalisia käyttäjiä kokeilematta. Käyttäjillä, jotka eivät ole koskaan asentaneet uusia ohjelmia tietokoneeseen, saattaa liittyä ohjelmien asentamiseen henkinen kynnys, vaikka kyse olisi omasta tietokoneesta, jolle he saisivat asentaa, mitä haluavat. Jos sähköinen virsikirja toteutetaan verkkopalveluna, ei kokeileminen ja käyttäminen vaadi asentamista, ja aloituskynnys madaltuu huomattavasti, eritoten jos palvelun kokeileminen ei vaadi rekisteröitymistä. Verkkopalvelun etuna on myös, että palvelu voidaan toteuttaa niin, että sitä voi käyttää hyvin monenlaisilla päätelaitteilla. Verkkopalvelujen ongelmana on, että kaikkialla ei ole käytettävissä Internet-yhteyttä. Käyttäjä voi kuitenkin tallettaa verkkopalvelusta lataamaansa aineistoa omalle koneelleen, ja on mahdollista toteuttaa palvelu teknisesti siten, että etukäteen ladattua aineistoa voidaan muokata käyttäjän päätelaitteessa ilman verkkoyhteyttä.

Jos virsikirjan yhteydessä, esimerkiksi samassa verkkopalvelussa, julkais-taisiin muitakin kristillisissä yhteisöissä käytettyjä laulukirjoja, voisi hankkeella olla kristittyjen yhteyttä edistävä merkitys.

4.3 Projektin puitteissa julkaistavien virsien valinta

Vuoden 1986 virsikirjassa on 387 sävelmää, jos liiteosan sävelmiä ei oteta lukuun [PT04, sivu 479]. Tämän projektin puitteissa julkaistaviksi valittiin vain osa näistä. Tekijänoikeussyistä jätettiin pois kaikki aineisto, johon omistaa oikeuksia jokin muu taho kuin kirkon keskusrahasto. Koska tämänkin karsinnan jälkeen jäljelle jäi liian paljon virsiä syötettäväksi järjestelmään tämän projektin puitteissa, käytettiin laulujen valitsemiseen apuna kahta lähdettä. Suomen evankelis-luterilaisen kirkon verkkosivuilla on julkaistu lista [vir] virsistä, joita suositellaan laulettavaksi peruskoulussa. Taulukkoon 4.1 on kerätty näistä ne virret, joiden sävelmä on tekijänoikeudellisesti vapaa tai joiden

sävelmän tekijänoikeudet omistaa kirkon keskusrahasto². Toisena lähteenä käytettiin tutkimusta, josta ilmeni, mitä virsiä Suomen evankelis-luterilaisen kirkon jumalanpalveluksissa laulettiin eniten kirkkovuonna 2009–2010 [Pit]. Näistä virsistä on koottu taulukkoon 4.2 sävelmältään tekijänoikeudellisesti vapaat virret³. Taulukoihin koottujen virsien sävelmät talletettiin palveluun lukuun ottamatta kursivilla korostettuja virsiä, joiden nuottikuvissa oli merkintöjä, joita Veisuuvekotin ei toistaiseksi tue⁴. Eniten lauletuista virsistä talletettiin virttä 301 lukuunottamatta kumpikin sävelmä, mikäli samaan virteen oli a- ja b-sävelmä. Virrestä 301 talletettiin vain b-sävelmä, koska a-sävelmä on tekijänoikeudellisesti suojattu. Virsistä, joiden melodiat talletettiin, talletettiin myös tekstit tai alkusanat⁵ virsiin, joissa mihinkään säkeistöön ei omista oikeuksia muu taho kuin kirkon keskusrahasto. Virsistä, joiden tekstejä ei talletettu, ei talletettu edes alkusanoja.

Virsi kirjassa on useita sävelmiä, jotka esiintyvät useammassa virressä. Virsiä, joissa on sama sävelmä, kutsutaan toistensa rinnakkaisvirsiksi. Taulukoissa 4.1 ja 4.2 on sarakkeeseen *Tallennetut rinnakkaiset* kirjattu, mikä kunkin virren rinnakkaisvirret tietokantaan tallennettiin. Rinnakkaisvirsiä tallettamalla saatiin helposti kasvatettua tietokannassa olevien virsien määrää, koska kertaalleen talletettua melodiaa saatiin hyödynnettyä useasti. Veisuuvekottimen tietokannassa soinnutukset liitetään tiettyyn melodiaan, johon voi olla viite useammasta virrestä⁶. Niinpä yksi melodia soinnuttamalla saadaan automaattisesti soinnutettua kaikki virret, joissa sama melodia esiintyy: esimerkiksi soinnuttamalla mikä tahansa virsistä 60, 206, 362, 395, 497 ja 573 tullaan soinnuttaneeksi jokainen näistä.

²Virren 603 sävelmän tekijänoikeudet omistaa kirkon keskusrahasto, virsien 15, 31, 32, 59, 88, 125, 135, 177, 454, 462, 488, 492, 499, 502, 503, 509, 513, 517, 548, 581, 584, 726, 744 ja 745 sävelmien oikeudet taas muu taho.

³Taulukosta jätettiin pois eniten laulettujen virsien joukossa esiintyneet virret 135, 517, 511, 462, 509, 513, 581, 600 ja 59, koska näiden sävelmän tekijänoikeudet omistaa jokin taho.

⁴Virsisä 10 ja 18 käytetään yhdyskaaria, virressä 510 puolestaan kertausmaaleja. Veisuuvekotin ei toistaiseksi tue näitä.

⁵Koko teksti talletettiin virsiin, joiden numero oli alle 100. Suurempinumeroisiin virsiin talletettiin pääsääntöisesti vain alkusanat. Tekstien lähteenä käytettiin etupäässä Verkko-virsi kirjaa (evl.fi/virsi kirjja).

⁶Veisuuvekottimen käyttöliittymässä ei toistaiseksi ole toimintoa, jolla järjestelmälle voitaisiin kertoa, että samaan melodiaan on tarkoitus liittää useampia virsiä. Niinpä useamman virren liittäminen samaan melodiaan on tehtävä käsittelemällä tietokantaa jollain muulla työkalulla kuin Veisuuvekottimen www-käyttöliittymällä. Veisuuvekottimen käyttöliittymällä on toki mahdollista kirjoittaa sama melodia useampaan virteen, mutta tällöin järjestelmä kohtelee melodioita toisistaan erillisinä, eikä yhteen kirjoitettu soinnutus välity toisiin. Taulukoissa mainitut rinnakkaisvirret on kytketty tietokannassa samoihin melodioihin.

Järjestelmään on syötetty myös muutamia muita virsiä satunnaisin perustein. Projektin jälkeen järjestelmään voidaan tekijänoikeuksien sallimissa rajoissa lisätä virsiä esimerkiksi virsilaulannan edistämisestä kiinnostuneiden seurakuntalaisten vapaaehtoistyönä.

Taulukko 4.1: Kouluvirsiä

Numero	Alkusanat	Tallennetut rinnakkaiset
1	Hoosianna	
2	Avaja porttis	
6	Käy, kansa, Herraasi vastaan	
7	Valmistu, Herran kansa	52, 293, 425, 566
10	<i>Iloitse, kaikki kansa</i>	
13	Nyt sytytämmme	
17	Ilon päivä verraton	447
18	<i>Nyt ilovirttä veisaten</i>	
21	Enkeli taivaan	
25	Nyt seimellesi	257
26	Oi Armon lähde autuas	
30	Maa on niin kaunis	
54	Käykäämme nyt Jerusalemiin	
60	Soi kunnia ja kiitos	206, 362, 395, 497, 573
61	Ken tahtoo käydä	
63	Oi rakkain Jeesukseni	243, 292
78	Vieraalla maalla	274
79	Oi Jeesus, lähde autuuden	577, 601
84	Nyt se suuri päivä koitti	250
85	Nyt ylistämme Jumalaa	
92	On Kristus noussut kuolleista	
104	Pilvimuurista valo välähtää	
105	Aurinkomme ylösnousi	
170	Jumala on linnamme	
228a	Vaiti kaikki palvokaamme	
228b	Vaiti kaikki palvokaamme	
332a	Herraa hyvää kiittäkää	218, 428
332b	Herraa hyvää kiittäkää	
338	Päivä vain ja hetki kerrallansa	
341a	Kiitos sulle, Jumalani	441a
341b	Kiitos sulle, Jumalani	
377	Sun haltuus, rakas Isäni	472a

Jatkuu seuraavalla sivulla

Taulukko 4.1 – *Kouluvirsiä, jatkoa edelliseltä sivulta*

Numero	Alkusanat	Tallennetut rinnakkaiset
397	Kun on turva Jumalassa	182
484	Totuuden Henki	
490	Mä silmät luon ylös taivaaseen	
501	Kuule, Isä taivaan	
510	<i>Sinä, Jeesus, ymmärrät parhaiten</i>	
516	Kuulkaa, keitä Mestari	
552	Mua siipeis suojaan kätke	
571	Jo joutui armas aika	
572	Taas kukkasilla kukkulat	342, 460, 475, 530
603	Isä Jumala	

Taulukko 4.2: Sävelmiltään vapaat lauletuimmat virret

Numero	Alkusanat	Tallennetut rinnakkaiset
126	Me kiitämme sinua	114
228	Vaiti kaikki palvokaamme	
130	Jumala Isä taivaassa	
229	Saa ehtoollinen Jeesuksen	
225	Jeesus luona armopöydän	266, 472b
21	Enkeli taivaan	153, 612
134	Pyhä, pyhä, pyhä	a:218, 428
332	Herraa hyvää kiittäkää	
571	Jo joutui armas aika (Suvivirsi)	
334	Armon lapset riemuitkaa	
329	Kiitos nyt Herran	28b, 93
339	Halleluja, kiitos Herran	
221	Kiittäen nyt ylistämme	
222	Jeesus Kristus elämämme	
447	Kristus valo maailman	17
484	Totuuden Henki	180
30	Maa on niin kaunis	
224	Oi Jeesus leipä elämän	
442	Herran ristin kantajiksi	
452	Jo Herran armopöytä	8, 98
27	Juhlimaan tulkaa	

Jatkuu seuraavalla sivulla

Taulukko 4.2 – *Lauletuimpia, jatkoa edelliseltä sivulta*

Numero	Alkusanat	Tallennetut rinnakkaiset
475	Kaikkien silmät tarkkaavat	458, 485
232	Taas kutsut meidät pöytääsi	
498	Nyt kulkee halki korpimaan	
13	Nyt sytytämme kynttilän	
35	Joulu riemukas	496, 500
105	Aurinkomme ylösnousi	
236	Kätesi laske Herramme	
510	<i>Sinä Jeesus ymmärrät parhaiten</i>	
326	Sinulle kiitos, Isä maan ja taivaan	

Luku 5

Virsikoodi – menetelmä virsinuottien informaation esittämiseen tekstin muodossa

5.1 Koodaukselle asetetut vaatimukset

Tässä luvussa tarkastellaan, millaisia merkintöjä virsikirjan nuottikuvat sisältävät ja esitetään menetelmä tämän informaation kuvaamiseksi tekstin muodossa. Esitetty menetelmä on tässä projektissa kehitetty, mutta menetelmässä on vaikutteita kahdesta muusta tekstipohjaisesta notaatiosta: LilyPond-nuotinnusohjelman [lilc] notaatiosta sekä abc-notaatiosta [Wal]. Selvyyden vuoksi tässä projektissa kehitettyyn menetelmään viitataan termillä virsikoodi erotuksena LilyPond-notaatiosta ja abc-notaatiosta. Termiksi on valittu virsikoodi virsinotaation sijaan, jotta sana loisi mielikuvan tekstipohjaisesta notaatiosta. Tavoitteena oli laatia notaatio, joka täyttää seuraavat ehdot:

1. Notaatiolla voi ilmaista virsikirjan nuoteissa olevan musiikillisen informaation siten, että tämän informaation pohjalta voidaan tuottaa vastaavia nuottikuvia kuin painetuissa virsikirjoissa¹.
2. Musiikin teorian perusteet hallitsevien ihmisten pitää voida oppia notaatio vähällä vaivalla.
3. Notaation pitää olla tiivistä, jotta säästytään turhalta kirjoitusvaivalta.

¹Tuotettavien nuottikuvien ei tarvitse vastata alkuperäisiä esimerkiksi valittujen fonttien osalta, mutta kylläkin palkitusten ja viivastonvaihtokohtien osalta.

4. Notation pitää olla helposti luettavaa sekä ihmiselle että tietokoneohjelmalle.
5. Ihmisten pitää voida kirjoittaa virsinuoteissa usein toistuvat merkinnät notaatiolla helposti ja nopeasti.

LilyPond-notaatio olisi täyttänyt ensimmäisen ehdon, samoin esimerkiksi MusicXML [musa]. Muiden ehtojen suhteen kumpikaan ei tuntunut optimaaliselta ratkaisulta. Notaatiosta haluttiin sellainen, että sitä olisi myös ihmisten helppo sekä lukea että kirjoittaa, jotta notaatiota voisi editoida tavallisella tekstieditorilla käyttämättä graafista editoria, joka piilottaisi tekstipohjaisen notaation käyttäjältä ja antaisi käyttäjän operoida graafisella nuottikuvalla. Tämä oli tärkeää, koska tekstin editoiminen verkkopalvelussa on teknisesti helppo järjestää. Kaikissa yleisesti käytetyissä verkkoselaimissa on valmiit välineet tekstin editoimista varten. Niinpä virsikoodiin perustuva sähköinen laulukirja voitaisiin toteuttaa aluksi siten, että laulut syötetään tekstinä. Graafisen editorin toteuttaminen jäisi mahdolliseksi jatkokehityskohteeksi.

Tekijällä oli projektiin ryhtyessään melko hyvä käsitys siitä, millaista informaatiota virsikirjan nuottikuvat sisältävät, ja tämän käsityksen pohjalta oli mahdollista hahmotella kieltä, joka täyttäisi ensimmäisen edellä esitetystä vaatimuksista. Virsikirjan nuottikuvien läpi käyminen oli kuitenkin tarpeen. Tämän luvun esitystä laadittaessa on tarkasteltu yksityiskohtaisesti virsiä 1–100. Sen jälkeen on tutkittu tähän otokseen perustuvien havaintojen yleistettävyyttä koko virsikirjaa koskemaan.

5.2 Malliedition valinta

Suomen evankelis-luterilaisen kirkon vuoden 1986 virsikirjasta on laadittu toisistaan poikkeavat yksiaäniset nuottikuvat ainakin neljästi. Timo Leskelän ja Hanna Koskisen laatimat nuottikuvat vastaavat ilmeisesti pitkälti toisiaan, koska kumpienkin nuottikuvien yhteydessä on sama esipuhe, jossa on oma jaksonsa myös melodioiden merkitsemisestä [vir87, vir04]. Kuitenkin jo virren 1 tarkastelu paljastaa, että nuottikuvissa on eroja ainakin kaarituksen, palkituksen ja rivinvaihtokohtien osalta. Leskelän nuottikuvassa on ainoastaan yksi kaari, jolla merkitään pisteellisen neljäsosanuotin ja kahdeksasosanuotin muodostama melisma. Kun melismat koostuvat yhteen palkitetusta nuoteista, ei kaarta ole piirretty. Koskisen nuottikuvassa kaikki melismat on kaaritettu. Palkkeja käytetään Leskelän nuottikuvassa hieman Koskisen nuottikuvaa säästeliäämmin.

Antti Vuori ja Pekka Nyman ovat kirjoittaneet virsikirjan virsistä soinnuoksilla varustetut nuottikuvat [soi07], jotka poikkeavat Leskelän ja Koski-

sen nuottikuvista muutenkin kuin sointumerkkien osalta. Virsi 1 on Vuoren ja Nymanin versiossa kirjoitettu F-duuriin, kun se Leskelän ja Koskisen nuottikuvissa on G-duurissa. Palkituksen ja kaarituksen osalta virsi on kirjoitettu kuten Koskisen versiossa, kun taas rivinvaihtokohdat poikkeavat sekä Leskelän että Koskisen versiosta. Risto Ranta-aho on puolestaan laatinut nuottikuvat Aurinko kustannukselta vuonna 2013 ilmestyneeseen soinnutettuun virsikirjaan.

Tätä projektia varten valittiin virsinuottien esikuviksi Koskisen nuottikuvat. Taulukoissa esiintyvät nuottikuvaleikkeet ovat peräisin Gracevoice Oy:n julkaisemasta Suomalaisesta eVirsikirjasta [Graa], jossa käytetään Koskisen nuottikuvia.

5.3 Koodauksen yleiset piirteet

Virsikoodissa lauluihin liittyvä informaatio on jaettu melodiaan, sanoitussäkeistöihin ja soinnutukseen, ja jokainen näistä koodataan erikseen. Virsikirjassa on useita melodioita, joita käytetään useammassa virressä, ja melodian ja sanoitusten erottaminen toisistaan mahdollistaa sen, että useammasta virrestä voi olla viite samaan melodiaan. Lisäksi niin melodia-, soinnutus- kuin sanoituskoodikin pysyy yksinkertaisempänä, kun koodataan vain yhtä asiaa kerrallaan. Mikäli laulukirjaa talletetaan järjestelmään usean henkilön voimin, voidaan työt jakaa siten, että yksi kirjoittaa melodioita, toinen soinnutuksia ja kolmas sanoituksia, ja jokaiselle riittää, että tietää koodaustavan, jota sovelletaan siihen asiaan, jota itse koodaa. Haittapuolena on, ettei koodista pysty heti näkemään, miten nuotit, soinnut ja tavut osuvat toistensa kohdalle: tämän näkee parhaiten nuottikuvasta, joka koodin pohjalta tuotetaan.

Virsikoodi ei ota kantaa lauluihin liittyvien metatietojen (säveltäjä, sanoittaja...) merkitsemiseen. Liitteessä C on muutamia esimerkkejä siitä, kuinka virsien melodioita sellaisena kuin ne Koskisen nuottikuvissa esiintyvät ilmaistaan virsikoodina.

Mikäli virren jokainen säkeistö lauletaan samalla melodialla, tarvitsee virren melodia koodata vain kertaalleen. Jos kuitenkin virren melodia on riippuvainen säkeistöstä, kuten virsien 15 ja 87 tapauksissa, koodataan virteen useampi melodia.²

Virsikoodissa isot ja pienet kirjaimet samastetaan. Rivinvaihdot ja välilyönnit samastetaan, eikä peräkkäisten välilyöntien ja rivinvaihtojen lukumäärällä ole merkitystä. Oleellista on vain, onko kaksi muuta merkkiä erotettu toisistaan näillä. Tyyliohjeena on kirjoittaa nuottikuvan jokainen rivi

²Veisuuvekotiin ei toistaiseksi tue tapauksia, joissa melodia on riippuvainen säkeistöstä.

yhtenä koodirivinä ja erottaa ilmaukset koodirivin sisällä toisistaan vain yhdellä välilyönnillä.

Kielen ilmaukset erotetaan toisistaan välilyönnillä tai rivinvaihdolla. Jokainen virsikoodin ilmaus on jokin seuraavista: etumerkintä, tahtiosoitus, nuotti, tauko, palkkiryhmä (eli ryhmä nuotteja, jotka on tarkoitus yhdistää toisiinsa palkilla), hengityspilkku, tahtiviiva (joka voi olla tavallinen tai kertauksen aloittava, päättävä tai päättävä ja aloittava) tai kertausmaalimerkki.

5.4 Virsinuottien merkintöjen koodaaminen

Tämän luvun seuraaminen saattaa olla helpompaa, mikäli käsillä on Koskisen nuottikuvia käyttävä virsikirja sekä liite C, joka tarjoaa esimerkkejä nuottikuvien sisällön esittämisestä virsikoodina.

Viivastojen alkukohdat ja leveydet

Näyttää siltä, että Koskisen nuottikuvissa virsissä 1–632 kaikki viivastot ovat yhtä pitkiä ja alkavat ja loppuvat vaakasunnassa samaan kohtaan. Viivastojen yhdenmittaisuutta voidaan siis pitää oletuksena, jota ei tarvitse koodata. Virsikirjan liiteosassa, osiossa Jumalanpalvelusten tekstejä ja lauluja, on kuitenkin nuottikuvia, joissa viivastojen pituus vaihtelee, eivätkä viivastot aina ala samasta kohtaa.

Virsi virreitä kirjaa läpi käytäessä huomataan, että ainakin virren 320 viimeinen viivasto on muita lyhyempi. Virsikoodiin ei kuitenkaan sisällytetä tapaa ilmaista viivastojen pituuksia.

Viivastojen alkuihin piirretyt merkinnät

Kukin viivasto alkaa nuottiavaimella, jota seuraa mahdollinen etumerkintä. Virren ensimmäisen viivaston alussa etumerkintää seuraa mahdollinen tahtiosoitus. Samana pysyvää tahtiosoitusta ei olla ensimmäisen viivaston jälkeen toistettu edes virren 15 neljännen säkeistön nuotissa, vaikka säkeistöjen 1–3 nuotti on päätetty päättävään tahtiviivaan. Virressä 87 tahtiosoitukset ja tahtiosoituksettomat jaksot vaihtelevat keskenään, ja tahtiosoitus on kirjoitettu aina tahtiosoituksellisen jakson alkuun, joka sijaitsee aina viivaston alussa.

Nuottiavaimet

Virsien 1–100 nuottikuvissa jokaisen viivaston alussa on G-avain, eikä nuottiavaimia esiinny muissa kohtaa viivastoja. Tätä voidaan siis pitää oletusarvona, jota ei tarvitse erikseen koodata.

Tahtiosoitukset

Virsissä 1–100 virren alussa oleva tahtiosoitus pysyy voimassa virren loppuun asti kaikissa muissa, paitsi virressä 87. Tahtiosoitus voi kuitenkin olla yhdistetty kahdesta tahtiosoituksesta, jolloin tahdin pituudet voivat vaihdella vapaasti näiden kahden tahtiosoituksen välillä. Virressä 87 vaihtelevat keskenään tahtiosoitukselliset ja tahtiosoituksettomat jaksot.

Virressä 344 (ja virren rinnakkaisvirsissä 417 ja 457) vaihtuu tahtiosoitus kertaalleen.

Taulukossa 5.1 esitetään virsissä 1–100 esiintyvät tahtiosoitukset. Suurin osa virsiä 1–100 tahtiosoituksista ilmaistaan virsikoodissa murtolukuna, joka kirjoitetaan esimerkiksi $3/4$. Virsikirjan nuottikuvissa on kuitenkin myös yhdistettyjä tahtilajiosoituksia, jotka ilmaistaan virsikoodissa kahdella peräkkäisellä murtoluvulla, jotka erotetaan toisistaan viivalla, siis esimerkiksi $3/2-2/2$. Näiden lisäksi nuottikuvissa käytetään kahta merkintää, joissa ei esiinny numeroita lainkaan. Nämä ilmaistaan virsikoodissa muodossa $T:C$ ja $T:C|$. Ilmausten alussa esiintyvä, ABC-notaatiosta [Wal] lainattu T : on tarpeen, jotta kävisi ilmi, että kyse on tahtiosoituksesta. Jos T : jätettäisiin pois, ei tiedettäisi, tarkoittaako C nuottia vai tahtiosoitusta. Merkintä $C|$ olisi tunnistettavissa tahtiosoitukseksi, mutta ihmislukijoiden kannalta se muistuttaisi erehdyttävästi c -nuottia, jonka jälkeen esiintyy tahtiviiva. Murtolukuina ilmaistavat tahtiosoitukset voidaan kirjoittaa ilman T : -merkkejä, koska murtoluku on ilmankin niitä tunnistettavissa tahtiosoitukseksi. Kuitenkin myös murtolukuina ilmaistavat tahtiosoitukset voidaan aloittaa näinnä merkeillä kirjoittaen esimerkiksi $T:3/4$. T -kirjain viittaa englannin ilmaukseen *time signature*.

Etumerkinnot

Virsissä 1–100 virren alussa ilmoitettu etumerkintä pysyy voimassa virren loppuun asti. Taulukossa 5.2 esitetään virsissä 1–100 esiintyvät etumerkinnot. Nuottikuvia selatessa ja taulukkoa tehtäessä oletettiin, että sama määrä ylennys- tai alennusmerkkejä tarkoittaa aina samaa etumerkintää, eli että ensimmäinen ylennys- tai alennusmerkki esiintyy aina samalla paikalla, toinen samoin jne. Taulukon laatimisen jälkeen tarkastettiin oletus käymällä virret

Tahtiosoitus	Koodataan	Esiintyy virsissä
<u>C</u>	T:C	1, 6, 13, 15, 16, 22, 27, 28a, 28b, 31, 34, 56, 57, 63, 64, 66, 67, 68, 78, 93
<u>6</u> <u>4</u>	6/4	2, 5, 10, 11, 18, 19, 20, 38, 54, 62, 72a, 85, 92
<u>3</u> <u>4</u>	3/4	3, 26, 32, 40, 42, 46, 49, 70, 81, 84, 95
<u>C</u>	T:C	4, 7, 8, 9, 17, 23, 25, 29, 36, 41, 43, 47, 52, 58, 60, 65, 69, 71, 82, 96, 98
<u>2</u> <u>2</u>	2/2	12, 24, 30, 35
<u>2</u> <u>4</u>	2/4	14, 21, 50, 55, 75, 76, 79, 89a
<u>3</u> <u>2</u>	3/2	33, 83, 86
<u>32</u> <u>22</u>	3/2-2/2	37, 39, 73
<u>23</u> <u>22</u>	2/2-3/2	44, 51, 53, 74, 80, 88, 91
<u>54</u> <u>44</u>	5/4-4/4	45, 89b
<u>5</u> <u>4</u>	5/4	48
<u>6</u> <u>8</u>	6/8	59, 61
<u>3</u> <u>8</u>	3/8	77
<u>52</u> <u>44</u>	5/4-2/4	94
<u>23</u> <u>44</u>	2/4-3/4	97, 100
<u>32</u> <u>44</u>	3/4-2/4	99
ei tahtiosoitusta		72b, 90

Taulukko 5.1: Virsien 1–100 tahtiosoitukset

1–100 läpi etumerkintä kerrallaan. Ainakin näiden virsien osalta oletus piti paikkansa. Oletuksen pohjalta päädyttiin koodaukseen, jossa ilmoitetaan pelkästään, koostuuko etumerkintä ylennys- vai alennusmerkeistä, ja montako näitä merkkejä on. Sama etumerkintä voi liittyä useampaan eri sävellajiin. Kun etumerkintä ilmaistaan kertomalla vain etumerkkien laatu ja määrä, ei melodiaa koodatessa tarvitse tutkia, onko kyse duurista, mollista vai jostakin muusta säveljärjestelmästä.

Selvyyden vuoksi etumerkintä aloitetaan virsikoodissa merkeillä K:, jossa K-kirjain viittaa ilmaukseen key signature. Esimerkiksi etumerkintä, joka koostuu kahdesta ylennyksestä, ilmaistaan siis K:2#. Etumerkinnän aloittava koodi K: on lainattu abc-notaatiosta.

Etumerkintä	Koodataan	Esiintyy virsissä
	K:1#	1, 2, 6, 11, 12, 15, 19, 25, 32, 36, 37, 44, 49, 61, 62, 67, 69, 71, 82, 90, 97, 100
	K:2#	3, 20, 26, 30, 34, 35, 60, 74, 76, 78, 84, 91
	K:3#	48, 86, 94
	K:4#	7, 52
	K:1b	4, 5, 10, 13, 14, 17, 24, 27, 31, 33, 38, 39, 42, 59, 64, 72b, 73, 75, 80, 88, 89a, 99
	K:2b	8, 9, 22, 29, 41, 46, 47, 63, 66, 68, 77, 79, 96, 98
	K:3b	16, 18, 23, 55, 56, 58, 65, 89b, 95
	K:4b	54, 72a
ei etumerkintää		21, 28a, 28b, 40, 43, 45, 50, 51, 53, 57, 70, 81, 83, 85, 87, 92, 93

Taulukko 5.2: Virsien 1–100 etumerkinnät

Tahtiviivat

Nykyaikaisessa nuottikirjoituksessa nuottikuvaa jaksotetaan yleensä piirtämällä viivaston poikki tahtiviiva, kun melodia on edennyt etumerkinnän osoittaman keston verran. Tällainen tahtiviivojen piirtäminen on automatisoitavissa, jolloin tahtiviivojen paikkaa ei tarvitse koodata erikseen. Tahtiviivojen voidaan olettaa käyttäytyvän tällä tavoin virsissä, jotka on varustettu tahtiviivoilla ja joissa on yksiselitteinen tahtiosoitus. Ensimmäisen tahdin pituus on kuitenkin koodattava jotenkin, koska osa virsistä alkaa kohotahtidilla. Lisäksi on huomattava, että tahtiviivojen koodaamatta jättäminen tekee koodin ihmiselle vaikeaksi lukea. Virsikoodissa tahtiviivat kirjoitetaan aina koodiin pystyviivalla (|). Käytäntö on sama kuin abc-notaatiossa.

Osassa virsiä on jaksottavina merkkeinä tahtiviivojen asemasta tahtiviivaa lyhyempiä pystysuoria viivoja, jotka alkavat nuottiviivasto toiseksi ylimmältä viivalta, risteävät ylimmän viivan kanssa ja jatkuvat viivojen välin verran ylimmän viivan yläpuolelle. Näitä viivoja kutsutaan kesuuroiksi. Virsistä 1–100 on kesuuroin jaksotettu virret 5, 10, 11, 18, 19, 20, 54, 62 ja 92. Virsissä 4, 7, 8, 22, 36, 43, 52, 60, 69, 82, 96 ja 98 esiintyy yksi kesuura virren alussa osoittamassa ensimmäistä painollista nuottia, jonka jälkeen melodiaa ei enää jaksoteta kesuuroin eikä tahtiviivoin. Kesuuraa merkitään virsikoodissa plus-merkillä (+), jonka voidaan ajatella koostuvan nuottiviivaston ylimmästä viivasta ja kesuuramerkistä³.

Virsistä 1–100 kokonaan tahtiviivattomia ja kesuurattomia ovat virret 9, 16, 17, 23, 25, 28b, 29, 41, 47, 58, 65, 71, 72b ja 90. Myöskään virressä 87 ei käytetä edellä lueteltuja merkintöjä, ellei lasketa seurakunnan ja kuoron osuuksien välissä olevaa kaksoistahtiviivaa 7. ja 8. säkeistöjen välissä. Tahtiviivojen puuttumista ei tarvitse virsikoodissa merkitä, koska kaikki tahtiviivat, jotka nuottikuvaan halutaan, on virsikoodissa koodattava erikseen näkyviin.

Esimerkiksi virressä 135 esiintyy kaksoistahtiviiva. Ainakin virressä 20 on kaksoistahtiviiva erottamassa säkeistöä ja kertosäkeistöä toisistaan. Ainakin virressä 87 käytetään kaksoistahtiviivaa erottamaan seurakunnan ja kuoron osuuksia toisistaan. Virsikoodissa kaksoistahtiviiva koodataan kahdella peräkkäisellä pystyviivalla (||).⁴

Jokainen virsi loppuu päättävään tahtiviivaan. Virsikoodissa päättävää tahtiviivaa ei tarvitse koodata. Virressä 15 päättäviä tahtiviivoja esiintyy kaksi kappaletta, koska nuotti on jaettu kahteen osaan, joista edellistä sovel-

³Veisuuvekotin piirtää toistaiseksi kesuurat tahtiviivoina, mutta koodissa kesuurat kannattaa silti koodata omalla merkillään, koska Veisuuvekottimen myöhemmät versiot voivat kyetä piirtämään kesuurat oikein.

⁴Veisuuvekotin ei toistaiseksi osaa tulkita kaksoistahtiviivamerkkiä.

letaan säkeistöihin 1-3 ja jälkimmäistä säkeistöön 4. Myös virressä 87 on kaksi lopettavaa tahtiviivaa, koska virteen liittyy lopuke, jota sovelletaan vain, mikäli virrestä on valittu tietyt säkeistöt. Näitä erityistapauksia Veisuuvekotin ei toistaiseksi osaa käsitellä.

Rivinvaihtokohdat

Riviä vaihdetaan Koskisen nuottikuvissa usein tahtiviivan tai kesuuran kohdalla. Joissakin tahtiviivallisissa ja kesuurallisissa virsissä riviä vaihdetaan kuitenkin myös kesken tahdin. Tällaisia ovat välillä 1–100 virret 1, 2, 5, 10, 11, 14, 18, 19, 20, 24, 27, 28b, 31, 34, 35, 39, 44, 45, 59, 61, 62, 63, 64, 67, 72a, 73, 74, 77, 83, 85, 86, 89b, 93, ja 94.

Rivinvaihto merkitään virsikoodissa kauttaviivalla (/). Virsikirjoissa on joskus painettu sanoitukset siten, että säerajaa on merkitty kauttaviivalla rivin vaihtamisen sijaan. Kauttaviivalla ja rivinvaihdolla on näin ollut aikaisemminkin semanttinen yhteys. Virsikoodissa riviä suositellaan luettavuuden parantamiseksi vaihdettavaksi kauttaviivan jälkeen, vaikkei koodissa esiintyvillä rivinvaihdolla ei ole semanttista merkitystä.

Kertaukset

Virsiä, joista kerrataan maalittomasti alku ja vain alku, ovat virsissä 1–100 virret 5, 7, 8, 9, 10, 11, 17, 25, 28b, 37, 41, 43, 44, 45, 46, 47, 52, 55, 57, 58, 62, 63, 64, 65, 66, 67, 84, 86, 89a, 89b, 93, 96, 97, 98 ja 100. Kerrattavan jakson alkua ei näissä merkitä erikseen aloittavalla kertausmerkillä. Muunlaisia kertausrakenteita, jotka ilmaistaisiin pelkästään aloittavien ja lopettavien kertausmerkkien avulla, esiintyy ainoastaan virressä 1, joka jakautuu kahteen osaan, joista molemmat kerrataan. Kerrattavien osien välissä ei ole merkkiä, joka sekä lopettaisi että aloittaisi kertauksen, koska osien välissä on rivinvaihto, jolloin ensimmäinen osa päättyy päättävään kertausmerkkiin ja toinen osa alkaa aloittavalla kertausmerkillä. Ensimmäisen osan alussa ei ole kertausmerkkiä, kuten ei kerrattavissa aluissa muutenkaan. Maalittomat kertaukset merkitään virsikoodiin kuten taulukossa 5.3 esitetään⁵. Ilmaukset on lainattu abc-notaatiosta [abca, luku 4.8].

Virressä 320 on osia, jotka kerrataan useammin kuin kahdesti. Jos kertaus on tarkoitus laulaa usemmin kuin kahdesti, kirjoitetaan virsikoodiin päättävän kertausmerkin jälkeen kertausten lukumäärä ja asteriski, siis esimerkiksi :|5*.⁶

⁵Veisuuvekotin ei toistaiseksi tue kertausmerkkiä, joka sekä lopettaa että aloittaa kertauksen.

⁶Veisuuvekotin ei toistaiseksi tue useammin kuin kahdesti toistettavia kertauksia.

Kertausmaaleja esiintyy välillä 1–100 virsissä 23, 24, 29 ja 39. Näissä kaikissa kerrattava osa on virren alku. Ensimmäiseen maaliin mennään ensimmäisellä kertauksella ja toiseen toisella. Virret 23 ja 29 ovat tahtiviivattomia, 24 ja 39 tahtiviivallisia. Tahtiviivattomien virsien tapauksessa ensimmäinen kertausmaali alkaa maalia edeltävän ja ensimmäisen maaliin kuuluvan nuotin puolivälistä. Tahtiviivallisten virsien tapauksessa ensimmäinen kertausmaali alkaa tahtiviivan kohdalta. Kummassakin tapauksessa jälkimmäinen kertausmaali alkaa kertausmerkin kohdalta. Kerrattavan jakson alussa ei ole kertausmerkkiä.

Kertausmaalien alut ilmaistaan virsikoodissa aloittavalla hakasululla, jota seuraa maalin numero. Viimeisen kertausmaalin loppu ilmaistaan lopettavalla hakasululla. Käytäntö on lainattu abc-notaatiosta [abca, luku 4.9] sillä erotuksella, että virsikoodin kertausmaalimerkit on erotettava ympäristöstään välilyönnein, vaikka abc-notaatiossa ne voidaan kirjoittaa kiinni tahtiviivaan.⁷

Virsikirjassa on muutamia virsiä, jotka ovat rakenteensa puolesta erikoistapauksia. Näitä ei ole toistaiseksi huomioitu Virsikoodissa eikä Veisuuvekotimessa. Virteen 87 liittyy kerrattavia osia ja lopuke, jonka laulaminen riippuu siitä, mitä säkeistöjä on valittu. Lopukkeen käytöstä annetaan nuotin yhteydessä sanalliset ohjeet. Virren 15 nuotissa ei ole lainkaan kertausmerkkejä. Sen sijaan nuotti on jaettu kahteen osaan, joiden väliin on kirjoitettu 2. ja 3. säkeistön sanat. Tästä nuotin lukija ymmärtää, että ensimmäistä melodiaa sovelletaan säkeistöihin 1-3, toista säkeistöön 4. Virsi 361b jakautuu kahteen osaan, jotka molemmat kerrataan, kuten virressä 1. Virressä 361b esiintyy kertausmerkki, joka sekä päättää että aloittaa kertauksen, toisin kuin virressä 1. Virsissä 501 ja 515 on kerrattava osa poikkeuksellisesti lopussa. Myös virressä 502 on lopussa kerrattava osa, vieläpä kertausmaalein. Virressä 510 on kohotahti, jonka jälkeen seuraa kerrattava osa, joka on varustettu maalein.

Kertausmerkki	Merkintä
aloittava	:
lopettava	:
lopettava ja aloittava	: :

Taulukko 5.3: Kertausten merkitseminen

⁷Veisuuvekotin ei toistaiseksi tue kertausmaaleja.

Hengityspilkut

Hengityspaikat on Koskisen nuottikuvissa merkitty viivaston yläpuolelle piirrettyillä pilkuilla lukuunottamatta virsiä, joissa hengityspaikat ilmenevät tauoista. Hengityspilkuttomia virsiä ovat virret 13, 19, 21, 45, 48, 50, 55, 79, 89a, 90 ja 94. Hengityspilkkuja esiintyy ainakin kahden nuotin välissä (kuten virressä 1), viivaston viimeisenä merkinä (kuten virressä 1), nuotin ja tahtiviivan välissä (kuten virressä 3), nuotin ja kesuuran välissä (kuten virressä 5) sekä tahtiviivan päällä (kuten virressä 24). Hengityspilkut merkitään virsikoodissa heittomerkillä (').

Nuottien korkeudet

Virsikoodissa nuotit koodataan sävelten nimien avulla. Sävelet nimetään saksalaisittain, jolloin c:tä edeltävä nuotti on nimeltään h, jota alennettuna kutsutaan nimellä b. Tämä lienee suomalaisten kirkkomuusikkojen keskuudessa tunnetumpi käytäntö kuin vaihtoehto, jossa c:tä edeltävän nuotin nimi on b, vaikkei sitä olisi alennettu. Virsikoodissa käytetyt sävelten nimet ovat siis c, d, e, f, g, a, h ja b. Kirjainten koolla ei ole merkitystä, mutta pieniä kirjaimia suositellaan. Ylennykset ja alennukset ilmaistaan käyttämällä päätteitä siten, kuin suomen kielessä on tapana. Ylennetty c on cis, kaksoisylennetty ciscis, alennettu c puolestaan ces, kaksoisalennettu ceses. Nuottien nimiä käytetään korkeuksien ilmaisemiseen myös abc- ja LilyPond-notaatioissa. Ylennysten ja alennusten ilmaisemisen osalta abc ja LilyPond poikkeavat toisistaan. Virsikoodin tapa vastaa LilyPondin tapaa.

Virsien melodiat on kirjoitettu korkeuksille, joilla enin osa sävelistä sijoittuu yksiviivaisen oktaavin alueelle. Virsikoodissa nuottien nimet viittaavat oletusarvoisesti yksiviivaiseen oktaavialaan sijoittuviin korkeuksiin. Tätä matalampi oktaaviala merkitään kirjoittamalla nuotin nimen jälkeen pilkku, korkeampi liittämällä heittomerkki. Samoin menetellään LilyPond-notaatiossa käytettäessä absoluuttista korkeuksien ilmaisemista [lila]. Pilkkuja ja heittomerkkejä lisäämällä oktaavialaa voidaan laskea tai nostaa useammalla oktaavilla, vaikkakaan tämä ei ole tarpeen virsiä koodattaessa.

Nuottien kestot

Lyhin nuottiarvo, jota virsissä 1–100 esiintyy, on kuudestoistaosanuotti, pisin kokonuotti, ellei breviksiä lasketa. Pisteellisiä nuotteja esiintyy, mutta pisteitä on korkeintaan yksi. Poikkeusjakoja (triolit, kvintolit, kvartolit, ...) ei esiinny.

Virsistä 1–100 brevikseen päättyviä ovat 4, 7, 8, 9, 16, 17, 22, 23, 25, 29,

36, 41, 43, 47, 52, 58, 60, 65, 69, 71, 82, 96 ja 98. Virressä 87 brevis esiintyy seurakunnan vastausten loppuissa. Nämä kaikki ovat tahtiviivattomia virsiä. Koska pystyviivalla, jota käytetään tahtiviivan ilmaisemiseen, on näissä virsissä niin vähän muuta käyttöä, voidaan sillä merkitä brevis aiheuttamatta sekaannuksia inhimillisille lukijoille. Koska brevisnuotti on aina kokonuotti, ei tätä tarvitse erikseen merkitä. Esimerkiksi yksiviivainen e-brevis merkitään siis |e|. Notaatiota lukeva tietokoneohjelma erottaa pystyviivan tarkoittavan tässä brevistä eikä tahtiviivaa, koska pystyviivat on kirjoitettu kiinni nuotin nimeen, kun taas tahtiviivat on erotettava muusta tekstistä välilyönnein.

Nuottien ja taukojen kestot ilmaistaan kuten taulukossa 5.4. Kuudestoistaosa ilmaistaan kirjoitusvaivan vähentämiseksi pelkällä numerolla 6 numeron 16 sijaan, koska kuudesosanuottia ei ole. Pisteellisten nuottien tapauksessa numeron jälkeen kirjoitetaan piste, kaksoispisteellisten kaksi pistettä (ja niin edelleen). Nuotin tai tauon kesto on oletusarvoisesti sama kuin edellisen nuotin tai tauon, kunpi sitten edeltääkin käsillä olevaa nuottia tai taukoa. Edellisellä nuotilla tai tauolla tarkoitetaan nuottia tai taukoa, joka koodauksessa esiintyy juuri ennen nuottia tai taukoa, jonka kestoä määritetään. Koska koodissa voi olla kertausmerkkejä, ei kyse välttämättä ole nuotista, joka melodiaa toteutettaessa soi ennen nuottia, jonka kestoä määritetään. Ensimmäisen nuotin tai tauon kesto on oletusarvoisesti neljäsosa. Nuottien kestojen ilmaisemisen osalta Virsikoodi muistuttaa LilyPond-notaatiota.

Kesto	Merkitään
kokonuotti	1
puolinuotti	2
neljäsosanuotti	4
kahdeksasosanuotti	8
kuudestoistaosanuotti	6

Taulukko 5.4: Nuottien ja taukojen kestojen merkitseminen

Tauot

Taukoja merkitään kirjaimella r englannin sanan *rest* mukaan. Taukojen kestot ilmaistaan kuten nuottien kestot. Esimerkiksi neljäsosataukoa merkitään r4.

Nuottivarret

Virsissä 1–100 kaikissa nuoteissa on nuottivarret. Ainakin virsikirjan liiteosassa, osiossa Jumalanpalvelusten tekstejä ja lauluja, on kuitenkin myös lauluja, joissa ei ole nuottivarsia nuoteissa. Tällaisia virsiä ovat 718, 719, 755b, 758, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805a, 805b. Virsikoodiin ei toistaiseksi ole määritelty ilmausta nuottivarsien puuttumiselle.

Kaaret

Kaaria käytetään virsien 1–100 nuottikuvissa kahdessa eri tehtävässä. Melisman merkitsemiseen kaarta käytetään virsissä 1, 2, 3, 5, 6, 10, 11, 15, 16, 17, 19, 21, 24, 25, 26, 27, 28a, 29, 30, 31, 32, 35, 38, 40, 42, 44, 45, 46, 48, 49, 50, 51, 54, 55, 56, 58, 61, 62, 64, 65, 66, 69, 70, 72b, 74, 75, 76, 77, 78, 81, 82, 84, 85, 86, 87, 90, 91, 94, 95, 96, 97 ja 100. Melismat ilmaistaan virsikoodissa sanoituksen yhteydessä.

Kaaria käytetään Koskisen nuottikuvissa myös yhdistämään kaksi peräkkäistä, yhtä korkeaa nuottia, kun halutaan ilmaista sävelen kesto yhdistämällä kaksi aika-arvoa yhdeksi. Tällaisia kaaria esiintyy virsissä 5, 10, 15, 18, 30 ja 70. Näistä virsissä 15 ja 30 esiintyy katkoviivalla piirrettyjä kaaria, jotka palvelevat edellä kuvattua tehtävää, kuitenkin niin, että ne tulee säkeistöstä riippuen huomioida tai jättää huomiotta. Kahden peräkkäisen, samankorkuisen nuotin yhdistäminen kaarella yhdeksi pitkäksi nuotiksi säkeistöstä riippumatta ilmaistaan virsikoodissa kirjoittamalla alaviiva (_) viimeiseksi merkiksi sellaisten nuottien koodeihin, joista lähtee tällainen kaari. Jos kaari on riippuvainen säkeistöstä, merkitään kaari melodian asemasta sanoitukseen samoin kuin toimitaan merkittäessä kaaria, jotka ilmaisevat, että sama tavu jatkuu kahden eri korkuisen nuotin aikana.⁸

Palkitus

Koska monissa virsissä esiintyy vain neljäsosanuotteja ja pitempiä aika-arvoja, ei monissa virsissä ole palkkeja lainkaan. Taulukosta 5.5 havaitaan, että virsissä 1–100 on selvästi eniten virsiä, joissa palkkeja ei ole lainkaan. Lisäksi havaitaan, että mikäli palkkeja esiintyy, ne yhdistävät muutamaa poik-

⁸Veisuuvekotin ei toistaiseksi tue alaviivamerkintää. Kaaret, jotka kuuluisi virsikoodissa ilmaista alaviivoin, voitaisiin Veisuuvekottimen puutteen kiertämiseksi ilmaista melisma-kaarten tavoin. Tätä yritettäessä virren 10 kohdalla ei Veisuuvekotin kyennyt piirtämään kaaria oikein. Veisuuvekottimessa on ilmeisesti jokin melismojen merkitsemiseen liittyvä ohjelmointivirhe, joka tuli näin ilmi.

keusta lukuunottamatta vain kaksi nuottia toisiinsa. Koskisen editiossa palkitus ei ole riippuvainen tekstistä. Jo virren 1 nuottikuvassa on palkitettu yhteen kaksi nuottia, joita kumpaakin vastaa oma tavunsa tekstissä. Siispä palkitus on joko koodattava melodiaan tai jätettävä koodin tulkitsijan pääteltäväksi aika-arvoista. Virsikoodissa palkitus kirjoitetaan näkyviin, ja Veisuuvekotin palkittaa ainoastaan nuotit, jotka on virsikoodissa merkitty palkitettaviksi.

Abc-notaatiossa palkitus ilmaistaan kirjoittamalla nuottien nimet yhteen [abca, luku 4.7]. LilyPond-notaatiossa palkit lisätään oletusarvoisesti automaattisesti. Jos palkitus halutaan merkitä manuaalisesti, merkitään palkin alku- ja loppunuotti LilyPond-notaatiossa aloittavalla ja päättävällä hakasululla. [lilb] Virsikoodissa palkitetut nuotit yhdistetään tavuviivalla. Kirjoitettavia merkkejä ei tule yhtään lisää verrattuna palkittamattomiin nuotteihin, koska välilyönti nuottien välistä jää pois. Tavuviivan saa tietokoneen näppäimistöä yleensä yhtä näppäintä painamalla. Voidaan lisäksi ajatella, että merkki, jolla ilmaistaan tavurajoja tekstissä, on luonteva erotin palkitettujen nuottien väliin.

Palkitus	Virret
ei palkkeja	2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19, 20, 23, 25, 27, 28b, 29, 30, 32, 33, 35, 36, 37, 38, 39, 41, 42, 43, 46, 47, 49, 52, 53, 54, 57, 58, 60, 62, 63, 65, 67, 68, 71, 72a, 73, 74, 79, 80, 81, 83, 84, 85, 86, 88, 89a, 89b, 90, 92, 93, 94, 98, 99
ainoastaan palkkeja, jotka yhdistävät kaksi nuottia	6, 15, 16, 17, 21, 22, 24, 26, 28a, 31, 34, 40, 44, 45, 48, 50, 51, 55, 56, 61, 64, 66, 69, 70, 75, 76, 78, 82, 91, 95, 96, 97, 100
palkkeja, jotka yhdistävät vähintään kolme nuottia	1, 59, 72b, 77, 87

Taulukko 5.5: Virsien 1–100 palkitus

Muodoltaan poikkeukselliset virret

Virsikirjassa on muutamia virsiä, joiden nuottikuvissa on poikkeuksellisia piirteitä, jotka vaativat virsikoodiin ja virsikoodia tulkitsevaan järjestel-

mään piirteitä, joita virsikirjan virret yleisesti ottaen eivät vaadi. Tässä projektissa nämä virret jätettiin huomiotta, jotta projektin laajuus saatiin pidettyä kohtuullisena. Seuraavassa käydään läpi poikkeukselliset virret, joita ei huomioitu virsikoodin ominaisuuksia määriteltäessä eikä Veisuuvekotinta toteutettaessa.

Virsikoodi ei tue sanoituksissa esiintyviä kertausmerkkejä. Koska Veisuuvekotin tekstiä nuottikuvaan latoessaan noudattaa melodiakoodissa esiintyviä kertausmerkkejä, on samat sanat kirjoitettava kahdesti, jotta kertauksen jälkeen esiintyväksi tarkoitettu teksti ladottaisiin kertauksen jälkeen eikä kertaukseen. Niinpä esimerkiksi virren 1 nuottikuvaan tulee tarpeettoman paljon tekstiä. Seuraavassa käydään läpi yksittäisten, huomiotta jätettyjen virsien erityispiirteitä.

- Virren 15 nuotissa on nuotinnettu neljäs säkeistö erikseen. Tässä projektissa ei huomioitu virsiä, joissa melodia riippuu säkeistöstä.
- Virsi 87 muodostaa poikkeuksen moniin sääntöihin, joiden mukaan virsikirjan nuottikuvat yleisesti ottaen käyttäytyvät.
- Ainakin virressä 127 esiintyy kertausmerkki, joka samanaikaisesti päättää ja aloittaa kertauksen. Tämä merkintä on määritelty virsikoodiin, mutta Veisuuvekotin ei tue sitä.
- Virren 279 säkeistöihin on merkitty, laulaako ne Syntinen vai Jeesus. Tällaisia mainintoja ei toistaiseksi tueta.
- Virressä 320 on osia, jotka kerrataan useita kertoja. Lisäksi virren tekstissä on I:- ja II:-merkintöjä, joilla oletettavasti tarkoitetaan, että laulajat voivat jakautua kahteen joukkoon, joista toinen laulaa I:-merkityt, toinen II:-merkityt osuudet. Tällaisia merkintöjä ei toistaiseksi tueta. Lisäksi nuottikuvan viimeinen viivasto on muita lyhyempi. Virsikoodissa ei ole tapaa määritellä viivastojen pituuksia.
- Virsikirjan liiteosassa, osiossa Jumalanpalvelusten tekstejä ja lauluja, on runsaasti lauluja, jotka poikkeavat muodoltaan perusvirrestä.
- Virteen 473 on kirjoitettu neliääninen satsi. Virsikoodi ja Veisuuvekotin tulevat vain yksiäänistä nuottikirjoitusta.
- Virteen 549 liittyy aamen, joka lauletaan viimeisen säkeistön jälkeen.

Sointumerkit

Virsistä 1–632 seuraavat on Koskisen nuottikuvissa varustettu reaalisointumerkein: 13, 15, 30, 42, 88, 104, 125, 318, 332a, 338, 341b, 396, 397, 429, 454, 484, 488, 490, 492, 499, 501, 502, 503, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 547, 552, 555, 560, 563, 571, 581. Sointumerkit alkavat soinnun pohjasävelellä. Koskisen nuottikuvissa puolissävelaskel C:n alapuolella sijaitsevaa kantasäveltä merkitään H-kirjaimella, kun taas kokosävelaskelen C:n alapuolella sijaitsevan sävelen nimi on nuottikuvissa Bb. Ylennetyt pohjasävelet merkitään sävelen nimeä seuraavalla ylennysmerkillä (#), alennetut alennusmerkillä (b). Soinnun laadun merkitsemistä varten nuottikuvissa käytetään merkintöjä 7, m, m7, maj7 ja sus sekä pientä ympyrää sointumerkin oikeassa yläkulmassa. Pohjasävelen ja mahdollisen laatumerkinnän jälkeen soinnuissa esiintyy toisinaan kauttaviiva, jonka jälkeen on merkitty soinnun pohjasävelestä poikkeava bassoääni.

Virressä 560 vaihtuu sointu kesken nuotin.

Virsikoodissa sointujen pohjasävelistä käytetään samoja nimiä, joita melodian äänistä käytetään. Jos soinnun bassoääneksi halutaan jokin muu kuin soinnun pohjasävel, merkitään tätä kirjoittamalla soinnun nimen perään kauttaviiva, jonka jälkeen kirjoitetaan bassoääni. Soinnun pohjasävelen ja kauttaviivan väliin voidaan kirjoittaa soinnun laatua kuvaavia merkintöjä. Virsikoodi ei määrittele, millaisia näiden merkintöjen pitäisi olla, ja Veisuuvekotin on toteutettu siten, että nämä merkinnät kirjoitetaan sellaisenaan nuottikuvaan.

Virsien 332a, 502 ja 503 soinnuista kaksi on Koskisen nuottikuvissa ympäröity suluin. Virressä 571 käytetään sulkuja ensin kahden ja myöhemmin kolmen soinnun ympärillä. Virressä 338 on ensimmäiseen kokonaiseen tahtiin kirjoitettu kaksi sointua päällekkäin. Tämä on oletettavasti tarkoitus tulkita siten, että ylempi sointu soitetaan ensin, alempi kerrattaessa. Virsikoodi ei tue sointujen suluttamista eikä erillisten sointujen kirjoittamista kertauksen eri toistoille.

Soinnut erotetaan virsikoodissa toisistaan välilyönnillä. Oletuksena on, että jokaisen nuotin ja tauon päälle ladotaan oma sointunsa. Nuotti tai tauko voidaan ohittaa kirjoittamalla soinnun nimen tilalle tavuviiva. Mikäli halutaan, että soinnut vaihtuvat tietyn nuottiarvon välein, määritellään soinnun kesto kirjoittamalla kestokoodi, jollaisia käytetään nuottien yhteydessä (katso luku 5.4)⁹. Koodi vaikuttaa koodin jälkeen kirjoitettavien sointujen keston. Kestokoodin jälkeen tavuviivalla voidaan siirtyä kestokoodin verran eteenpäin melodiassa. Jos halutaan palata takaisin nuotti nuotilta etenevään

⁹Veisuuvekotin ei toistaiseksi tue kestokoodien käyttämistä.

käytäntöön, kirjoitetaan kysymysmerkki. Riippumatta siitä, edetäänkö nuotteittain vai tietyn aika-arvon välein, voidaan tahtiviivoin varustetun virren tapauksessa merkitä seuraavaan tahtiin siirtymistä pystyviivalla, seuraavan aloittavan kertausmerkin ohi siirtymistä kirjoittamalla aloittavan kertausmerkin koodi (|:) ja seuraavan lopettavan kertausmerkin ohi siirtymistä kirjoittamalla lopettavan kertausmerkin koodi (|:).

Esimerkkejä soinnutusten kirjoittamisesta

Virren 30 soinnutus voidaan ilmaista virsikoodina käyttämättä lainkaan aika-arvoja. Tällöin jokainen melodian nuotti ja tauko saa oletusarvoisesti oman sointunsa. Nuottien ja taukojen ohittamiseen käytetään seuraavaan tahtiin siirtymistä merkitsevää pystyviivaa sekä yksittäisen nuotin tai tauon ohittamista merkitsevää viivaa:

```
D G D | A7 - D |      |
Em A D | - - G | D | Em D/F# |
A | D H7 | Em | A7 |
D | Hm - Em | D/A A7 | D
```

Virsi 13 alkaa kohotahdilla, johon ei Koskisen nuottikuvassa ole merkitty sointua. Ensimmäinen sointu esiintyy ensimmäisen kokonaisen tahdin alussa, ja siitä alkaen jokainen sointu kestää joko puoli- tai kokonuuotin ajan. Soinnutus voidaan virsikoodina kirjoittaa esimerkiksi seuraavasti:

```
| 2 F C7 | F |
B G7 | C | F/A C7/G |
F B | C7 | F
```

Koodi alkaa tahtiviivalla, koska ensimmäiseen tahtiin ei haluta lainkaan sointumerkkiä. Tahtiviivaa edeltävillä välilyönneillä ei ole muuta merkitystä, kuin että ne tekevät koodin ihmislukijalle helpommaksi hahmottaa. Tahtiviivan jälkeen seuraava numero 2 ilmaisee, että soinnut etenevät puolinuotteittain. Numero oltaisiin haluttaessa voitu kirjoittaa myös ennen tahtiviivaa. Huomattakoon, että toisen rivin alussa esiintyvää sointua, jota Koskisen nuottikuvassa merkitään Bb, merkitään pelkällä B-kirjaimella.

Virren 88 soinnutus ilmaistaan esimerkiksi seuraavasti:

```
F | | 2 B C |
F | - C | F |
Gm | C | F B |
F/A Dm | Gm C7 | F
```

Toisen rivin toisen tahdin alussa esiintyvä viiva tarkoittaa, että puolinuotin verran tahdin alusta hypätään yli. Koska ennen viivaa on ollut kestkoodi, viittaa viiva aika-arvon ohittamiseen, ei melodiassa esiintyvän nuotin ohittamiseen.

Virressä 104 on kaksi tahtia, joissa esiintyy kaksi sointua yhden nuotin aikana. Lisäksi virren toiseksi viimeisessä tahdissa soinnut etenevät neljäsosittain, vaikka melodian rytmi on toinen, eikä tahdin viimeinen sointu näin osu kohdakkain minkään nuotin kanssa. Tällöin virsikoodissa on valittava kirjoitustapa, jossa kerrotaan soinnun kesto aika-arvona. Virsi 104 voidaan koodata esimerkiksi seuraavasti, jolloin soinnutus etenee neljäsosittain:

```
| 4 D - - A7 | D G D
| - - A E7 | A D A |
D - H7 - | Em -
A7 | D Hm Em A7 | D G D
```

Sama soinnutus voitaisiin kirjoittaa myös käyttämällä enemmän kesto-koodeja ja vähemmän viivoja:

```
| 2. D A7 | 4 D G D
| 2 - 4 A E7 | A D A |
2 D H7 | Em
A7 | 4 D Hm Em A7 | D G D
```

Sanoitukset

Tavurajat merkitään tavuviivalla. Melismat merkitään lisäämällä tavun perään alaviivoja (_). Jos tavu jakautuu kahdelle nuotille, alaviivoja lisätään yksi, jos kolmelle, lisätään kaksi jne. Järjestelmä, joka tuottaa nuottikuvat, huomioi melismat kaarittamalla melismaan kuuluvat nuotit.

Ainakin virressä 548 on kohta, jossa säkeistöstä riippuen lauletaan joko neljäsosanuotti tai kaksi samankorkuista kahdeksasosanuottia. Koskisen editiossa tämä on ratkaistu kirjoittamalla nuottiin kaksi kahdeksasosaa ja piirtämällä niiden välille pisteviivakaari. Veisuuvekotin piirtää kaaren tai jättää sen piirtämättä riippuen nuottikuvaa seuraamaan ladottavasta säkeistöstä.

Virren numero ja rinnakkaisvirret

Koskisen nuottikuvissa virren numero on piirretty nuottiviivaston yläpuolelle, eikä se vaikuta nuottien asetteluun. Virren numeron jälkeen on maininta mahdollisista rinnakkaisvirsistä. Virsikoodi ei määrittele ilmaisua rinnakkaisvirsien mainitsemiselle.

Huomattakoon, että Koskisen editiossa rinnakkaisvirret voivat olla eri sävellajeissa. Ainakin virret 60 ja 497 ovat eri sävellajeissa, vaikka ovatkin toistensa rinnakkaisvirsiiä. Sama ilmiö toistuu ainakin virren 236 kohdalla.

Tekijätiedot

Taulukossa 5.6 esitetään virsien 1–100 yhteydessä olevat melodian alkuperää kuvaavat ilmaukset. Ilmausten sisältämät tiedot vaihtelevat. Toisinaan kerrotaan säveltäjä, toisinaan ainoastaan alkuperämaa. Osa vuosiluvuista on tarkkoja, osa viittaa vuosisataan, joihinkin on sisällytetty vuoden epätarkkuutta ilmaiseva *noin*. Myös tekstien alkuperää kuvataan eri virsien yhteydessä vaihtelevin tavoin. Virsikoodi ei määrittele, miten tekijätiedot pitäisi ilmaista. Veisuuvekottimessa tekijätietojen rakennetta ei ole määritelty muuten kuin erottamalla sävelmään liittyvät tiedot tekstiin liittyvistä tiedoista. Kummatkin näistä talletetaan järjestelmässä yhtenä merkkijonona. Ratkaisu on yksinkertainen, mutta ei mahdollista esimerkiksi virsien järjestämistä syntyajankohdan mukaan helposti.

Taulukko 5.6: Melodian alkuperämerkinnät

Virsi	Merkintä
1	G. J. Vogler noin 1795
2	Ruotsissa 1697
3	Piae Cantiones 1582 / 1911
4	Jakob Regnart 1574
5	Saksalainen 1500-luvulta
6	James McGranahan 1881
7	Ranskassa 1557
8	Melchior Teschner 1614
9	Saksassa 1653
10	Saksalainen 1500-luvulta
11	Englannissa noin 1540
12	Jaap Geraedts 1973
13	Emmy Köhler 1898
14	Ranskassa 1545
15	Josef Hadar 1956
16	Keskiajalta / Saksassa 1524
17	Keskiajalta / Böömissä 1531
18	Keskiajalta / Saksassa 1533
19	Keskiajalta / Saksassa 1544

Jatkuu seuraavalla sivulla

Taulukko 5.6 – *Jatkoa edelliseltä sivulta*

Virsi	Merkintä
20	Keskiajalta / Saksassa 1543
21	Saksassa 1539
22	Keskiajalta / Saksassa 1524
23	Saksassa 1599
24	Toisinto Pieksämäeltä
25	Saksassa 1524
26	Vilhelm Anttila 1800-luvulla
27	John Francis Wade 1782
28a	Toisinto Pohjois-Savosta
28b	Keskiajalta / Ruotsissa 1697
29	Philipp Nicolai 1599
30	Saksassa 1842
31	Jean Sibelius 1909
32	Armas Maasalo 1925
33	Johann Crüger 1653
34	Seppo Suokunnas 1979
35	Sisilialainen, Pain. 1792
36	Keskiajalta / Saksassa 1543
37	Toisinto Eurajoelta
38	F. P. Krank 1889
39	Toisinto Pieksämäeltä
40	Toisinto Pohjois-Savosta
41	Saksassa 1653
42	Englannissa 1821
43	Philipp Nicolai 1599
44	Saksassa 1719
45	Toisinto Noormarkusta
46	Louis Bourgeois 1551
47	Saksassa 1653
48	Toisinto Kokemäeltä
49	Armas Maasalo 1925
50	Saksassa 1539
55	Toisinto Pohjois-Savosta
64	Toisinto Kalannista
65	Keskiajalta / Saksassa 1542
70	Englannissa noin 1780
72b	Keskiajalta
74	Toisinto Pohjanmaalta

Jatkuu seuraavalla sivulla

Taulukko 5.6 – *Jatkoa edelliseltä sivulta*

Virsi	Merkintä
75	Toisinto Pohjois-Savosta
79	Keskiajalta / Suomessa 1702
84	Toisinto Pohjois-Savosta
87	Keskiajalta
89b	Toisinto Pieksämäeltä
90	Ortodoksinen pääsiäistöropari
92	Keskiajalta / Böömissä 1531
93	Keskiajalta / Ruotsissa 1697
95	Toisinto Eurajoelta
96	Keskiajalta / Saksassa 1524
97	Piae Cantiones 1582 / 1911
100	Piae Cantiones 1582 / 1911

Muut nuottikuvien yhteydessä olevat merkinnät

Edellä käsiteltyjen merkintöjen lisäksi Koskisen nuottikuvien yhteydessä on ainakin seuraavat merkinnät:

- Virren 158 nuotin yhteydessä todetaan, että ”Virsi voidaan laulaa myös sävelmällä 193”. Vastaava maininta on lisäksi ainakin virsien 99, 180, 211, 259, 261, 289, 293, 364, 377 yhteydessä.
- Virren 61 nuotin yhteydessä mainitaan, miten virsi voidaan laulaa vuorolauluna.
- Virren 90 yhteydessä mainitaan, että virsi lauletaan yleensä kolme kertaa.

Näiden kirjaamiseen ei virsikoodissa ole ilmaisia, eikä Veisuuvekotin tue näiden merkintöjen tallettamista verkkopalveluun.

Luku 6

Veisuuvekotin – virsilähtöinen järjestelmä laulunuoottien verkkojulkaisuun

6.1 Toiminnot

Veisuuvekotin mahdollistaa useamman laulukirjan laulujen julkaisemisen samassa verkkopalvelussa. Veisuuvekottimen avulla luotavaan verkkopalveluun voidaan tallettaa laulusta melodia, sanoitus, vapaavalintainen määrä reaalisointumerkein kirjoitettuja soinnutuksia, tiedot sävelmän alkuperästä sekä muita tietoja (esimerkiksi tiedot sanoituksen alkuperästä). Kuvassa 6.1 esitetään virsi 3 Veisuuvekottimessa abcjs-kirjaston [abcb] avulla piirrettynä. Kyseessä on kuvaruutukaappaus Firefox-verkkoselaimesta.

Aineiston löytämiseksi käyttäjä valitsee, haluaako tarkastella kirjaa vai itse muodostamaansa laululistaa. Veisuuvekotin näyttää valitun kirjan tai listan sisällysluettelon taulukon muodossa. Käyttäjä voi rajata taulukkoa kirjoittamalla hakukenttään laulun numeron tai laulun ensimmäisen säkeistön alkusanoja. Käyttäjän kirjoittaessa numeron, karsiutuvat taulukosta laulut, joiden numero ei vastaa käyttäjän kirjoittamaa numeroa. Käyttäjän kirjoittaessa alkusanoja rajautuvat pois laulut, joiden ensimmäinen säkeistö ei ala käyttäjän kirjoittamalla tekstillä. Jo yksittäisen kirjaimen kirjoittaminen hakukenttään rajaa pois kaikki laulut, joiden ensimmäinen säkeistö ei ala tällä kirjaimella.

Taulukossa on kiinteinä sarakkeina laulun numero ja alkusanat. Näiden sarakkeiden tekstit toimivat linkkeinä, joita klikkaamalla kyseisen laulun nuottikuva avautuu. Taulukkoon voidaan valita myös lisätietosarakkeita, jotka kertovat valitut säkeistöt, transponoinnin määrän, valitun soinnutuksen,

3

Piae Cantiones 1582

1 2 3 4 5 6

Benjamin Schmolck 1712. Suom. Elias Lönnrot 1874. Virsikirjaan 1886. Uud. suom. Pekka Kivekäs ja komitea 1984.

Kuva 6.1: Virsi 3 Veisuuvekottimessa

sävelmän alkuperän sekä lauluun talletettujen soinnutusten määrän.

Veisuuvekotin tarjoaa välineet avatun laulun transponoimista sekä soinnutuksen ja säkeistöjen valitsemista ja nuottikuvan koon ja muodon säätämistä varten. (kts. kuva 6.2). Lisäksi tarjotaan painikkeet, joilla siirrytään sisällysluettelossa seuraavana, edellisenä tai ensimmäisenä esiintyvään lauluun. Nämä ovat hyödyllisiä erityisesti silloin, kun sisällysluettelo on piilotettu, koska painikkeet jäävät silloinkin näkyviin.

☒ Transponoi	ylemmäs alemmas	Salli korkeintaan 4 ylennystä (#) tai 2 alennusta (b) etumerkintään.
☒ Sointumerkit	☒ Valitse säkeistöt:	Päivitä
☒ Yhdistä peräkkäiset viivastot	Viivaston leveys 1000 pikseliä	Rinnakkaisia säkeistöjä 6 Päivitä

Kuva 6.2: Välineet nuottikuvan muokkaamiseksi

Transponoimista varten Veisuuvekotin tarjoaa painikkeet *ylemmäs* ja *alemmas*. Näiden yhteydessä on pudotusvalikot, joista käyttäjä voi valita, kuinka monta ylennystä ja kuinka monta alennusta etumerkinnässä saa korkeintaan olla. Kun käyttäjä painaa ylemmäs-painiketta, Veisuuvekotin transponoi laulun lähimpään nykyistä sävellajia ylempään sävellajiin, joka täyttää etumerkintää koskevan ehdon. Jos esimerkiksi lähtösävellaji on D-duuri ja sallitaan korkeintaan kaksi ylennystä ja kaksi alennusta, transponoituu nuottikuva Nosta-painiketta painamalla F-duuriin. Jos käyttäjä painaa samaa painiketta uudelleen, transponoitu nuottikuva edelleen G-duuriin. Alemmas-

painike toimii vastaavasti. Jos kuitenkin käyttäjä poistaa valinnan painikkeiden vasemmalla puolella olevasta transponoi-valintaruudusta (*engl. checkbox*), palaa sävelmä alkuperäiseen sävellajiinsa. Käyttäjän valitessa valintaruudun uudelleen, järjestelmä muistaa sävellajin, johon laulu oli ollut transponoituna, ja palauttaa laulun tähän sävellajiin.

Soinnutus saadaan mukaan nuottikuvaan valitsemalla valintaruutu Sointumerkit. Tämän jälkeen voidaan pudotuslistasta valita haluttu soinnutus. Soinnutukset esiintyvät listassa soinnutuksia kuvailevilla nimillä, jotka soinnuttaja on antanut. Soinnutukset esitetään valikossa nimiensä mukaisessa aakkosjärjestyksessä.

Sanoituksesta voi nuottikuvaan valita koko sanoituksen tai halutut säkeistöt. Käyttäjä voi luetella valitsemansa säkeistöt tekstikentässä pilkulla erotettuna. Mikäli mukaan halutaan useita peräkkäisiä säkeistöjä, voidaan säkeistöväli ilmaista yhdysviivalla. Yhdysviivan avulla ilmaistuja säkeistövälejä ja yksittäisiä säkeistöjä voi kirjoittaa kenttään sekaisin pilkulla erotettuna. Veisuuvekotin ymmärtää esimerkiksi seuraavat ilmaukset:

- 3
- 1-2, 4, 7
- 2, 4-6, 9

Veisuuvekotin lataa yhden säkeistön melodiasa seuraamaan ja loput säkeistöt nuottikuvan alapuolelle erilleen nuottikuvasta. Nuottikuvaa seuraamaan ladotun säkeistön voi valita joko klikkaamalla painiketta, jossa on säkeistön numero (kts. kuva 6.1), tai klikkaamalla halutun säkeistön sanoitusta.¹

Lauluun tehdyt asetukset tallennetaan painamalla painiketta Tallenna asetukset kirjaan.

Lauluista voi muodostaa nimettyjä listoja. Lauluhiin liittyvät asetukset kirjataan palveluun listakohtaisesti, joten sama laulu voi esiintyä eri listoissa esimerkiksi eri sävellajeissa ja eri soinnutuksen kera. Listalla olevat laulut voidaan järjestää vetämällä listan rivit uuteen järjestykseen tarttumalla hiirellä rivien alussa oleviin pallukoihin². Veisuuvekotin näyttää pallukat ainoastaan, kun Listojen muokkaaminen -valintaruutu on valittuna. Listan laulujen asetukset ja laulujen järjestys tallennetaan laulutaulukon yhteydessä olevaa Tallenna lista -painiketta painamalla.

¹Säkeistön latominen nuottikuvaa seuraten edellyttää, että säkeistön sanoitus on tavutettu. Projektin puitteissa kaikkien talletettujen virsien kaikkia säkeistöjä ei tavutettu.

²Toimintoa ei ole kokeiltu kosketusnäytöllisissä laitteissa.

Veisuuvekotin sisältää editorin (kts. kuva 6.3), jonka avulla järjestelmään voidaan syöttää uusia lauluja ja soinnutuksia ja jonka avulla aiemmin syötettyä aineistoa voidaan muokata. Tiedot syötetään tekstin muodossa luvussa 5 esitellyn virsikoodin avulla. Järjestelmä pitää laulujen melodian, soinnutukset ja säkeistöt erillään toisistaan sekä tietokannassa että editorissa. Mitä tahansa näistä kolmesta on mahdollista muokata tuntematta menetelmää, jonka mukaisesti kaksi muuta näistä koodataan. Järjestelmässä myös muokkausoikeudet voidaan kunkin käyttäjän ja laulukirjan osalta määritellä jokaiselle näistä kolmesta lauludatan osatekijästä erikseen. Lisäksi järjestelmä määrittelee oikeuden muokata laulun tekijätietoja, lisätä kirjaan uusia lauluja, lisätä lauluun uusia soinnutuksia ja muokata muokata soinnutuksia, jotka käyttäjä itse on lisännyt, vaikka käyttäjällä ei olisi oikeutta muokata muiden lisäämiä soinnutuksia. Mahdollisuus muokata kutakin informaatiotyyppiä tuntematta muiden informaatiotyyppien koodaamisen tapaa sekä mahdollisuus määritellä muokkausoikeudet informaatiotyyppikohtaisesti ovat tärkeitä ominaisuuksia, kun aineistoa talletetaan järjestelmään useiden henkilöiden voimin. Tällöin esimerkiksi sanoitusten tallettamiseen voi osallistua henkilöitä, jotka eivät ole opetelleet Veisuuvekottimessa käytettävää melodioiden tai soinnutusten koodaamisen tapaa. Tällaisille käyttäjille annetaan oikeudet ainoastaan sanoitusten muokkaamiseen, jolloin välttyään siltä, että nämä käyttäjät muokkaisivat vahingossa melodiakoodia osaamatta palauttaa koodiin tekemiään muutoksia entiselleen.



Kuva 6.3: Editori

Veisuuvekotin käyttöliittymä jakautuu vaakasuunnassa sisällys- ja nuottikuvasarakkeisiin. Sisällyssarakeessa ylimpänä esitetään Sisällön valinta -osio, jossa valitaan tarkasteltava kirja tai laululista ja rajataan valittua aineistoa hakuehdon avulla. Keskellä hakemistosaraketta on Listojen muokkaaminen -osio. Tämä osio tarjoaa välineet laulujen keräämiseen laululistoihin. Alimpana listataan valitut laulut. Nuottikuvasarake puolestaan näyttää nuottikuvan, painikkeet sisällysluettelossa siirtymiseen, kuvassa 6.2 esitetyn

säätöpaneelin ja kuvassa 6.3 esitetyn editorin.

Nuottikuvan esittäminen vaatii paljon tilaa näytöltä, samoin kuin laulutaulukko silloin, kun siinä esitetään paljon valinnaisia sarakkeita. Jotta tietyssä käyttötilanteessa tarpeettomat käyttöliittymän osat eivät turhaan veisi tilaa näytöltä, voidaan käyttöliittymän osia piilottaa ja kutsua esiin tarpeen mukaan. Tarkoitukseen käytetään käyttöliittymän ylälaudassa esitettäviä valintaruutuja. Sisällys- ja nuottikuvasarakkeet voidaan piilottaa kokonaisuudessaan poistamalla valinta vastaavista valintaruuduista. Kun sisällyssarake näytetään, voidaan vielä valita, näytetäänkö siinä valittujen laulujen lisäksi Sisällön valinta- ja/tai Listojen muokkaaminen -osiot. Kun nuottikuvasarake näytetään, voidaan valita, näytetäänkö säätöpaneeli ja/tai editori. Kun editori näytetään, Veisuuvekotin kätkee automaattisesti nuottikuvan alle ladotut säkeistöt. Näin editori saadaan lähemmäs nuottikuvaa, jolloin laulua editoivan käyttäjän on helpompi tarkistaa nuottikuvasta, onko kirjoittanut virsikoodia oikein.

Veisuuvekottimen avulla toteutettua verkkopalvelua voi käyttää kirjautumatta palveluun. Tällöin laululistat eivät kuitenkaan ole käytettävissä, ja palvelu unohtaa lauluihin tehdyt sävellaji-, soinnutus- ja säkeistövalinnat kirjasta toiseen siirryttäessä ja ladattaessa palvelun sivu uudelleen verkkoselaimen.

6.2 Toteutustekniikka

6.2.1 Tietoteknisten käsitteiden esittely

Veisuuvekottimen toteutuksessa on hyödynnetty tietokoneohjelmien suunnittelussa ja toteutuksessa paljon käytettyä olioparadigmaa. Oliotermien valinnassa noudatetaan Kai Koskimiehen Oliokirjan [Kos00] valintoja.

Oliolla tarkoitetaan tietokoneohjelman rakenneyksikköä, joka on yhdistelmä tietoa ja tiedon käsittelemiseen käytettyjä operaatioita. Oliot ovat luokkien ilmentymiä. Luokka vastaa olio-ohjelmoinnissa usein yhtä luonnollisen kielen käsitettä, joka liittyy ohjelmoitavaan sovellukseen. Veisuuvekottimessa esimerkiksi etumerkintää, nuottia ja laulukirjaa vastaavat omat luokkansa. Esimerkiksi etumerkintäluokasta luotu ilmentymä, etumerkintäolio, edustaa Veisuuvekottimessa tiettyä etumerkintää tietyssä kohdassa melodiasa.

Luokka voi määritellä joukon attribuutteja ja operaatioita. Attribuutila tarkoitetaan olioon liittyvää nimettyä tietokenttää. Olion kaikkien attribuuttien arvojen yhdistelmä muodostaa olion tilan. Operaatioihin voi liittyä parametreja. [Kos00, s. 30] Veisuuvekottimessa nuottiluokka määrittelee attribuutteinaan muun muassa nuotin korkeuden ja keston. Operaationaan

luokka määrittelee transponoinnin. Parametrina tällä operaatiolla on intervalli, joka määrittelee, paljonko transponoidaan ja kumpaan suuntaan. Olio-ohjelmoinnissa onkin keskeistä tiedon ja tiedon käsittelemiseen tarvittavien toimintojen niputtuminen yhdeksi kokonaisuudeksi.

Kirjastolla tarkoitetaan tietokoneohjelmoinnin yhteydessä kokoelmaa tietokoneohjelman osia, joita voidaan hyödyntää toteuttaessa muita tietokoneohjelmia. Luokkakirjasto koostuu luokista.

6.2.2 Järjestelmän arkkitehtuuri ja käytetyt teknologiat

Kuvassa 6.4 esitetään Veisuuvekottimen arkkitehtuuri hyvin yleisellä tasolla. Veisuuvekottimen aineisto eli laulujen melodiat, sanoitukset, soinnutukset ja teostiedot talletetaan relaatiotietokantaan verkkopalvelimelle. Samaan tietokantaan talletetaan myös tiedot käyttäjistä, kuten käyttäjätunnukset, salasana ja tiedot käyttäjien laatimista laululistoista ja lauluihin liittyvistä transponointi- ja muista asetuksista. Tietokannanhallintajärjestelmänä toimii MySQL. Relaatiomalli on vanha, koeteltu, paljon käytetty ja yleisesti tunnettu, samoin kuin MySQL, ja molemmat soveltuvat hyvin Veisuuvekottimen toteuttamiseen.

Veisuuvekottimen käyttöliittymä on toteutettu JavaScript-verkkosovelluksena.³ Sovellus koostuu yhdestä ainoasta verkkosivusta, jonka osia näytetään ja piilotetaan tarpeen mukaan, ja jolle ladataan uutta aineistoa tarvittaessa. Kun käyttäjä lataa verkkoselaimeensa Veisuuvekottimen sivun, latautuu selaimeen HTML:llä kirjoitettu palvelun käyttöliittymän perusrakenne, CSS-tiedosto joka määrittelee sivun muotoiluun liittyvät seikat, sekä JavaScript-ohjelmakoodi, jolla toteutetaan sivun toiminnallisuus. Palvelun sisältö ladataan sivulle jälkikäteen JavaScriptin avulla sitä mukaa kun sitä tarvitaan. JavaScriptillä kirjoitettu ohjelmakoodi ei kuitenkaan muodosta suoraa yhteyttä relaatiotietokantaan, vaan tietokantaa luetaan pienillä PHP-kielellä kirjoitetuilla ohjelmilla, jotka sijaitsevat palvelimella ja joita palvelin suorittaa, kun Veisuuvekottimen verkkosivun JavaScriptistä käsin esitetään palvelimelle pyyntöjä. Veisuuvekottimen PHP-ohjelmat tuottavat tietokannasta lukemastaan informaatiosta JSON-muotoisen esityksen, joka siirretään HTTP-yhteyksikäytännön avulla käyttäjän selaimeen, jossa se tulkitaan JavaScriptin avulla. JavaScriptillä informaatiota muokataan tarvittaessa esimerkiksi soinnutuksen yhdistämiseksi melodiaan tai laulun transponoimiseksi. Lopulta käyttäjälle näytettävän informaatio muunnetaan JavaScriptillä

³David Flanagan luokittelee JavaScriptiä käsittelevässä kirjassaan verkkosivut kahteen kategoriaan. Ensimmäiseen kategoriaan kuuluvia sivuja hän nimittää dokumenteiksi, toiseen kuuluvia sovelluksiksi [Fla11, s. 307].

joko grafiikaksi tai HTML-koodiksi, jotka liitetään Veisuuvekkotimen verkkosivuun.

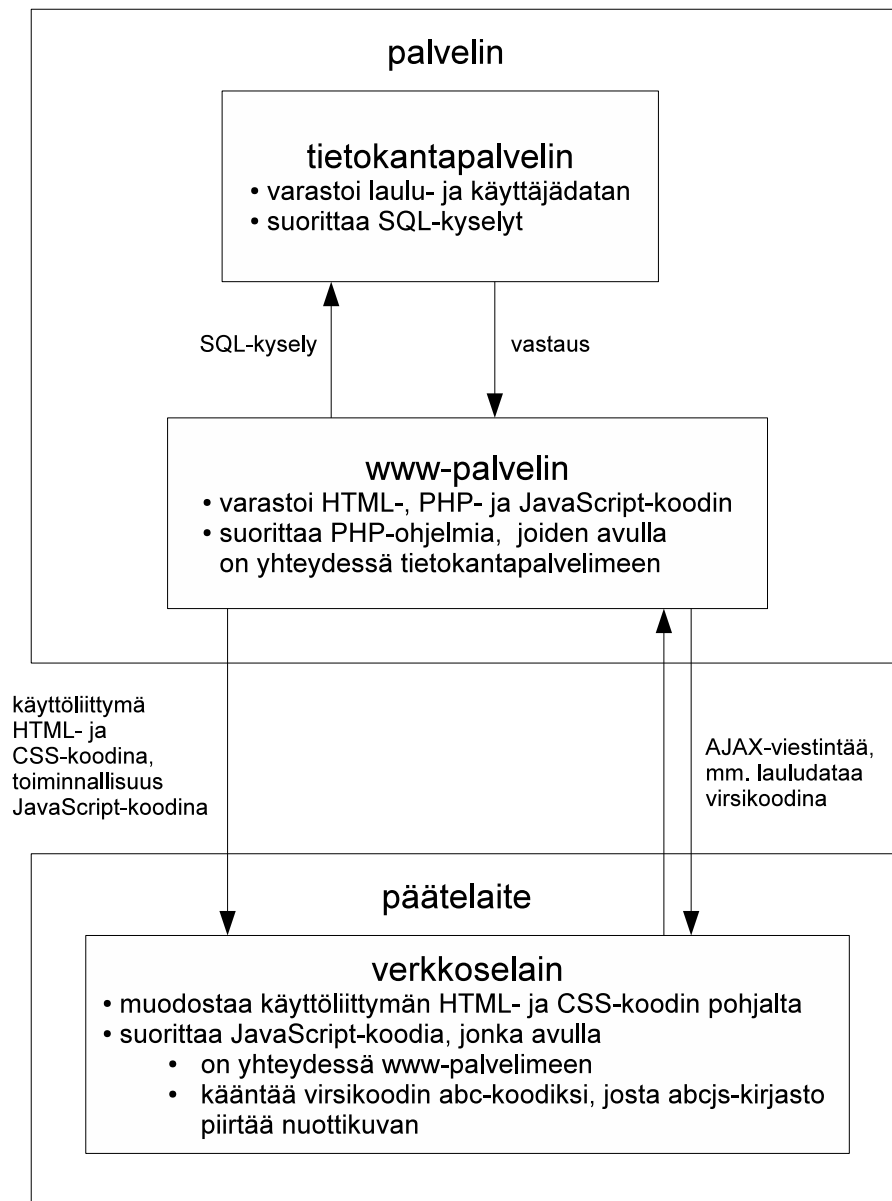
Nuottikuvien tuottamiseen sovelletaan abcjs-kirjastoa [abcb], joka tuottaa abc-muotoisesta musiikki-informaatiosta nuottigrafiikkaa. Abc-muotoon informaatio käännetään JavaScriptillä. Kääntäminen tapahtuu siten, että Veisuuvekkotinta varten suunnitellulla virsinotaatiolla kuvattu informaatio jäsennetään tässä projektissa tuotetulla JavaScript-koodilla, joka perustuu säännöllisiin lausekkeisiin. Jäsennyksen kuluessa informaatiosta luodaan oliomalli. Mallin luomiseen käytettävät luokat on varustettu metodeilla, jotka tuottavat abc-muotoisen esityksen mallista.

Jatkokehittäjän, joka haluaisi parantaa järjestelmän ulkoasua, ei välttämättä tarvitse muokata relaatiotietokantaa, PHP-ohjelmia eikä JavaScript-koodia. Niinpä hänen tarvitsee osata vain HTML- ja CSS-kieliä. Ylläpitäjän, jonka tehtävänä olisi muokata järjestelmän tietosisältöä, ei tarvitse tietää mitään HTML:stä, CSS:stä, PHP:stä tai JavaScriptistä, kunhan hän osaa hallita relaatiotietokantoja. MySQL:n alaisuudessa toimivaa relaatiotietokantaa on mahdollista hallita lukuisilla erilaisilla ohjelmilla. Veisuuvekkotimen tietosisältöä on siis mahdollista muokata käyttämättä lainkaan Veisuuvekkotimen omaa, verkkoselaimella käytettävää käyttöliittymää.

Koska abcjs-kirjasto, jonka avulla nuottikuvat piirretään, on yhteydessä muuhun järjestelmään ainoastaan hyvin kapean rajapinnan kautta, voitaisiin tämä kirjasto helposti korvata toisella JavaScript-kirjastolla, joka kykenisi piirtämään nuottikuvia abc-muotoisesta musiikki-informaation esityksestä. Mikäli haluttaisiin käyttää kirjastoa, joka piirtäisi nuottikuvia muun muotoisesta esityksestä, saattaisi olla suhteellisen helppoa lisätä musiikki-informaatiota esittäviin JavaScript-luokkiin operaatiot tämän muun muotoisen esityksen tuottamiseksi.

MVC-mallin hyödyntäminen käyttöliittymäluokkien kesken tekee MVC-mallin tunteville ohjelmistokehittäjille helpoksi hahmottaa, miten Veisuuvekkotimeen lisättäisiin uusia luokkia, joiden avulla toteutettaisiin uusia toimintoja. MVC-malliin voi tutustua esimerkiksi artikkelista Model–View–Controller [Mica].

Koska musiikki-informaation käsitteleminen ei tapahdu palvelimella vaan käyttäjän verkkoselaimessa, on käyttäjän koneelle ladattua informaatiota mahdollista käsitellä, vaikka yhteys palvelimeen menetettäisiin. Tämän ratkaisun pohjalta Veisuuvekkotimeen olisi melko helppo lisätä ominaisuus, joka mahdollistaisi laululistojen tai kokonaisten kirjojen lataamisen käyttäjän omalle koneelle siten, että niiden sisältämiä lauluja voisi käsitellä transponoimalla ja säkeistöjä ja soinnutuksia valitsemalla. Tämä olisi hyödyllinen ominaisuus ajatellen esimerkiksi leirikeskuksia, joissa ei välttämättä ole kiin-



Kuva 6.4: Veisuuvekottimen arkkitehtuuri

teää nettiyhteyttä, ja jotka saattavat sijaita kaukana matkapuhelinverkkojen tukiasemista.

Koska Veisuuvekottimen toiminnasta enin osa tapahtuu käyttäjän selaimessa, kuormittaa palvelu palvelinkonetta vain vähän, vaikka palvelulla olisi lukuisia yhtäaikaista käyttäjiä. Tämä laskee palvelun ylläpitäjille koituvia kustannuksia ja pienentää riskiä siihen, että palvelu hidastuisi käyttäjämäärien ollessa hetkellisesti poikkeuksellisen korkeita.

Virsikoodi vie palvelimella vain murto-osan siitä tilasta, jonka kuvamuodossa tallennetut nuottikuvat veisivät. Palvelimen ja asiakkaiden välisen tiedonsiirron määrä pysyy myös pienenä, koska palvelimelta ei siirretä asiakkaalle kuvatiedostoja vaan virsikoodia. Nämä Veisuuvekottimen ominaisuudet vähentävät ylläpitäjälle aiheutuvia kustannuksia.

Tekniikat, joita palvelimen on tuettava, jotta Veisuuvekotin voidaan asentaa, ovat hyvin yleisesti käytettyjä. Yrityksiä, jotka tarjoavat Veisuuvekottimen pystyttämiseen soveltuvia palvelimia on paljon, mikä on omiaan pitämään hinnat alhaisina. Tämä oli tärkeä syy valita PHP palvelimella ajettavien ohjelmien kieleksi, vaikka Java, jota myös olisi voitu käyttää, olisi ollut projektin tekijälle ennestään tuttu kieli PHP:n ollessa ennestään vieras. Javaa tukevia palvelinresurssien tarjoajia on vähemmän, jolloin nämä yritykset voivat laskuttaa korkeampia hintoja. PHP:n valintaa palvelimella suoritettavan koodin kieleksi puolsi myös, että PHP:tä käytetään niin paljon ja niin monet osaavat sitä. Eksoottisempien tekniikoiden valinta olisi rajannut potentiaalisten jatkokehittäjien joukkoa turhaan.

Nuotti-informaatiota syötettäessä ja muokattaessa informaatio esitetään Veisuuvekottimessa tekstimuotoisena. Tästä johtuen Veisuuvekottimeen ei tarvinnut toteuttaa editointikäyttöliittymää, vaan voidaan käyttää tekstin editoimiseen tarkoitettuja käyttöliittymäkomponentteja, jotka HTML-standardi vaatii selaimia toteuttamaan. Verkkopalvelut, joissa pitäisi pystyä editoimaan nuotti-informaatiota, ovat niin paljon harvinaisempia kuin palvelut, joissa on voitava editoida tekstiä, että on ymmärrettävää, ettei HTML-standardi vaadi selaimia toteuttamaan nuottieditoreja eivätkä selaimet näitä toteutakaan. Kun harvinaislaatuinen editointitarve, nuottien editointi, palautetaan tavalliseksi editointitarpeeksi, tekstin editoinniksi, tukevat verkkoselaimet tätä. Samaa tekstimuotoa, jossa nuotti-informaatio esitetään editoimista varten, käytetään myös nuottien tallettamiseen relaatiotietokantaan. Tämä tekee helpoksi tietokannan nuottisisällön editoimisen millä tahansa välineellä, joka on suunniteltu tietokannan tekstimuotoisen sisällön editoimiseen.

abcjs:n lisäksi Veisuuvekottimessa hyödynnetään jQuery- [Thea] ja jQuery UI [Theb] -kirjastoja⁴. jQuery:n käyttämisellä pyritään siihen, että Veisuu-

⁴UI on lyhenne englannin ilmaisusta user interface, suom. käyttöliittymä.

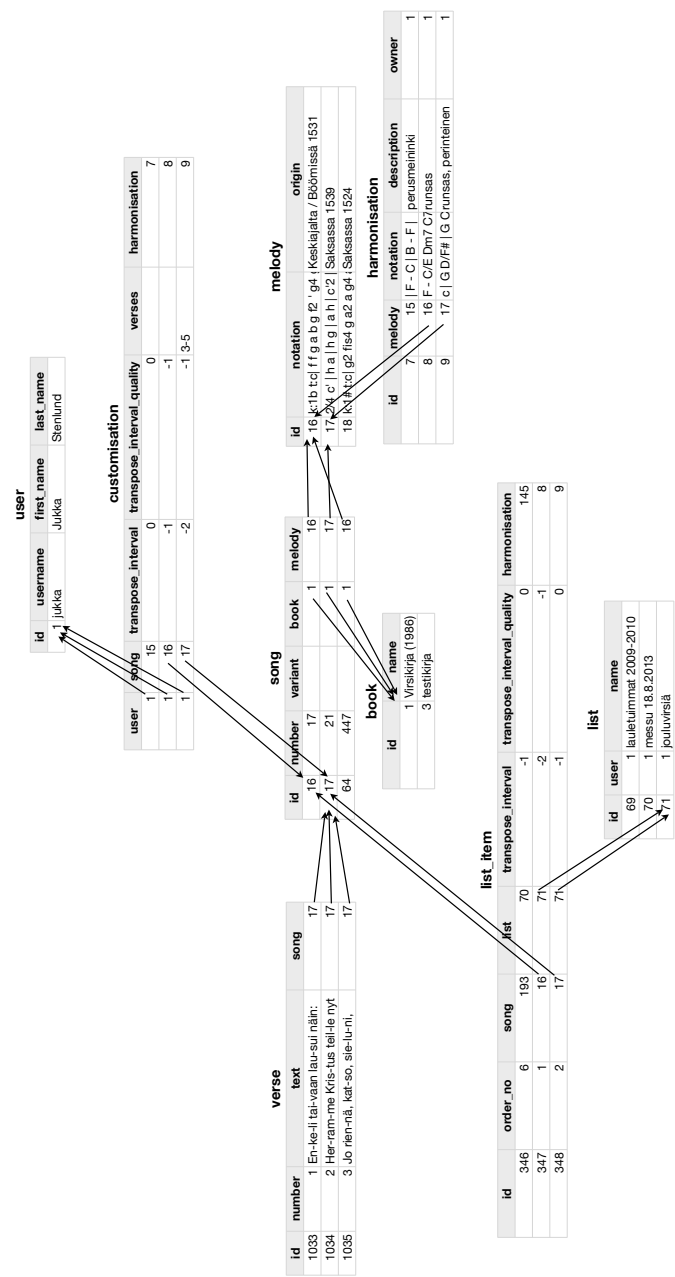
vekotin toimisi mahdollisimman useissa erilaisissa verkkoselaimissa. Tietyt asiat ohjelmakoodissa on kirjoitettava eri verkkoselaimille hieman eri tavoin. jQuery toimii eräänlaisena tulkkina ohjelmoijan ja verkkoselainten välillä: ohjelmoija käyttää jQuery:n välineitä, ja jQuery käyttää kunkin verkkoselaimen kohdalla niitä välineitä, joita kyseisen verkkoselaimen tapauksessa tulee käyttää. jQuery UI:n avulla taas on Veisuuvekottimeen toteutettu laululistojen vedä ja pudota -tyylinen järjestäminen sekä editorin välilehti-sommittelu.

Veisuuvekottimen ohjelmakoodi on kirjoitettu englanniksi, jotta ohjelmaa voisivat jatkokehittää myös henkilöt, jotka eivät osaa suomea. Vaikka Veisuuvekotin on suunniteltu erityisesti Virsikirjan aineiston verkkojulkaisemista silmälläpitäen, tukee järjestelmä useiden kirjojen julkaisemista samassa palvelussa, ja järjestelmän toteutuksessa on vältetty turhaa viittaamista nimen omaisesti Virsikirjaan. Näillä molemmilla ratkaisuilla pyritään siihen, että Veisuuvekotin järjestelmänä kiinnostaisi myös ihmisiä, joita Virsikirja ei kiinnosta. Mitä useampi ihminen käyttää Veisuuvekotinta, sitä todennäköisemmin joku kiinnostuu järjestelmän jatkokehittämisestä, mikä koituu myös Virsikirjan ystävien hyödyksi, vaikka jatkokehittäjää motivoisi mahdollisuus muunlaisen aineiston kuin virsien esillä pitämiseen.

6.2.3 Tietokannan rakenne

Sekä laulu- että käyttäjätieto talletetaan Veisuuvekottimessa relaatiotietokantaan. Relaatiotietokannoissa tieto organisoidaan relaatioiksi eli tauluiksi, joita voidaan havainnollistaa taulukkojen muodossa. Relaation jokainen rivi vastaa yhtä tietuetta ja rivin jokainen sarake tietueen yhtä kenttää. [LHL⁺92, sivu 211]. Veisuuvekottimessa on esimerkiksi melodioiden tallettamista varten oma taulunsa, jossa yksi tietue vastaa yhtä melodiaa. Jokainen melodiataulun tietue jakautuu vielä kenttiin, joista yhteen talletetaan melodiaa vastaava virsikoodimuotoinen esitys, toiseen tieto melodian alkuperästä jne. Relaation rivit yksilöidään nk. avainsarakkeiden avulla [LHL⁺92, sivu 212]. Relaatiosta A voidaan viitata relaatioon B tallettamalla relaation A yhteen sarakkeeseen relaation B avainsarakkeessa esiintyviä arvoja.

Kuvassa 6.5 esitetään nuolilla, miten Veisuuvekottimen taulut viittaavat toisiinsa. Kunkin taulun sisältöä esitetään kuvassa muutaman rivin verran. Todellisuudessa taulut sisältävät paljon enemmän rivejä. Koska kuvan tarkoituksena on havainnollistaa taulujen välisiä yhteyksiä, on tauluista jätetty pois joitakin sarakkeita. Joitakin tauluja on jätetty kokonaan pois. Veisuuvekottimen tietokannan kaikki taulut ja taulujen rakenne käydään läpi liitteessä D.



Kuva 6.5: Tietokannan taulut viittaavat toisiinsa

6.2.4 Virsikoodina esitetyn informaation muuntaminen oliomalliksi

Veisuuvekottimessa melodia, sanoitus ja soinnutus, jotka on tietokannassa ilmaistu luvussa 5 kuvattuna virsikoodina, jäsennetään automaattisesti, ja koodin pohjalta luodaan oliomalli.

Melodian ja soinnutuksen tapauksessa ilmaukset erotetaan koodissa toisistaan välilyönnein tai rivinvaihdoin. Niinpä koodia jäsennettäessä koodi pilkotaan ensin yksittäisiksi ilmauksiksi välilyöntien ja rivinvaihtojen kohdilta. Tämän jälkeen kustakin ilmauksesta tutkitaan, mitä virsikoodin ilmauskategoriaa se edustaa. Tämä tapahtuu säännöllisten lausekkeiden [Fla11, s. 251–258] avulla. Säännöllisillä lausekkeilla voidaan määritellä ehtoja merkkijonoille ilmaisemalla ehto merkkijonon muodossa tiettyjen sääntöjen mukaisesti. Esimerkiksi taukoa edustavat ilmaukset tunnistetaan Veisuuvekottimessa tauoiksi määrittelemällä merkkijonolle ehto säännöllisenä lausekkeena muodossa $\sim r\d? \backslash.*\$$. Tämä ehto määrittelee seuraavat asiat:

1. Merkkijonon on alettava ($\sim$ viittaa alkuun) r-kirjaimella.
2. r-kirjaimen jälkeen merkkijonossa voi esiintyä yksi numeromerkki eli jokin merkeistä 1,2,3,4,5,6,7,8,9 ja 0, mutta tämä voi jäädä poisinkin. ($\d$ tarkoittaa numeromerkkiä, $?$ taas ”yksi tai ei yhtään”)
3. Riippumatta siitä, seurasiko r-kirjainta numeromerkki vai ei, voi merkkijonon lopussa olla vielä mikä tahansa lukumäärä pisteitä (myös nolla on lukumäärä). ($\backslash.$ tarkoittaa pistettä, $*$ taas vapaavalintaista lukumäärää.)
4. Mitään muuta ei merkkijonossa voi olla. ($\$$ viittaa merkkijonon loppuun.)

Ilmaisujen kategorian tunnistamisen lisäksi on selvitettävä ilmaisujen tarkempi sisältö. Tauon tapauksessa on tiedettävä tauon kesto. Lisäämällä säännöllisiin lausekkeisiin sulkua voidaan määritellä, mistä tutkittavan merkkijonon osista ollaan kiinnostuneita. Esimerkiksi JavaScript-kieli tarjoaa väli-
neet, joilla nämä osat voidaan lukea muuttujien arvoiksi. Kun edellä esitetty säännöllinen lauseke muutetaan muotoon $\sim r(\d?)(\backslash.*)\$$, voidaan lausekkeen avulla erottaa tutkittavasta merkkijonosta numero, joka kertoo, onko kyseessä esimerkiksi neljäsosa- vai kahdeksasosatauko, sekä pisteistä koostuva osa.

Kun on selvillä, mitä kategoriaa ilmaus edustaa, tiedetään, mistä luokasta luodaan ilmentymä, olio, vastaamaan ilmausta. Kun on lisäksi selvitetty,

millaisesta ilmaisukategorian edustajasta tarkemmin ottaen on kyse (tauon tapauksessa kestoä kuvaava numero ja pisteiden lukumäärä) voidaan luoda olio, joka edustaa kyseistä luokkaa ja alustaa olion kentät vastaamaan ilmaisun attribuutteja.

Säännöllisten lausekkeiden syntaksia ei käydä tässä tarkemmin läpi, kuten ei myöskään muiden virsikoodin ilmaisujen tunnistamista. Taukojen tunnistamista käsiteltiin ainoastaan esimerkkinä, jotta lukija saisi yleiskäsityksen Veisuuvekottimessa käytetystä virsikoodin ilmaisujen tunnistamisen tavasta. Soinnutuskoodi käsitellään samaan tapaan kuin melodiakoodi. Sanoituskuodin jäsentäminen eroaa hieman soinnutus- ja melodiakoodin jäsentämisestä, mutta sitä ei käsitellä tässä dokumentissa.

6.2.5 Lauluinformaation esittämiseen käytetty oliomalli

Yleiskatsaus oliomalliin

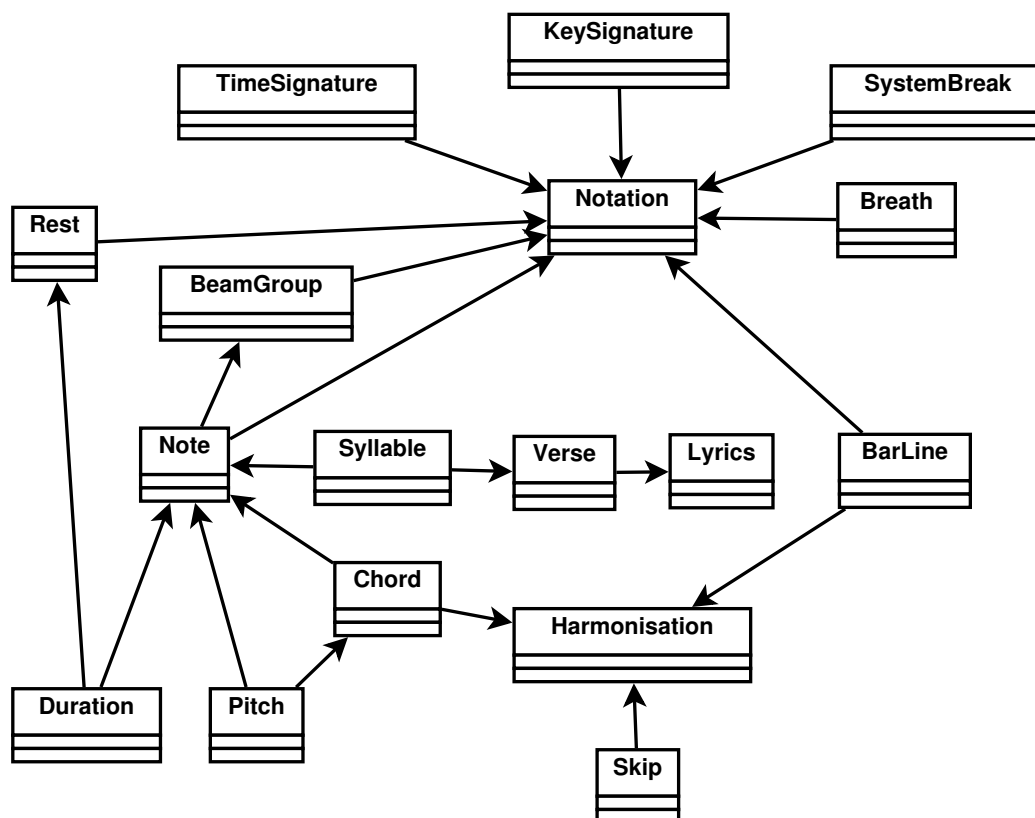
Laulun melodia-, soinnutus- ja sanoitusinformaatio, jota tarvitaan nuottikuvan piirtämiseen, esitetään Veisuuvekottimessa oliomallina⁵. Mallissa nuottikuvien tuottamisen kannalta relevantteja musiikillisia käsitteitä vastaamaan on määritelty luokkia, joiden ilmentyminä Veisuuvekotin luo olioita, kun Virsikoodia jäsennetään.

Oliomallin katto-olio on Notation, johon voidaan tallettaa nuottikuvan yksittäisiä osasia, nuotteja, taukoja ym. edustavia oliota. Seuraavassa käydään lyhyesti läpi kaikki luokat, joiden ilmentymiä käytetään laulun esittämiseen Veisuuvekottimessa. Luokat käydään läpi aakkosjärjestyksessä, koska luokkien kuvauksissa viitataan toisten luokkien kuvauksiin, jolloin on tärkeää, että minkä tahansa luokan kuvaus on helppo löytää. Lukeminen kannattanee kuitenkin aloittaa Notation-luokan kuvauksesta.

Kuvassa 6.6 esitetään, mitkä luokat liittyvät kiinteästi mihinkin toisiin luokkiin. Mikäli luokan A symbolista on piirretty nuoli osoittamaan luokan B symbolia, tarkoittaa tämä, että luokan B ilmentymät voivat sisältää viitteitä luokan A ilmentymiin. Kuvasta on nähtävissä, että viiteketjut johtavat lopulta Notation-, Harmonisation- tai Lyrics-luokkaan. Jotkut luokat liittyvät viittein suoraan tai välillisesti useampaan kuin yhteen näistä kolmesta luokasta.

⁵Tekijä silmäili projektin aikana Kai Lassfolkin väitöskirjaa *Music Notation as Objects — An Object-Oriented Analysis of the Common Western Music Notation System* [Las04]. Lassfolkin lähestymistapa oli graafisempi kuin tässä projektissa tavoiteltu (s. 6 "Here, music notation is approached primarily as a graphical system, which contrasts with some recently developed representations of music notation."), eikä teokseen paneuduttu projektissa syvällisemmin. Luvun 7 luokkakaavioista oli kuitenkin apua englanninkielisten musiikkitermien lähteenä.

Harmonisation-olioilla kuvatut soinnnutukset ja Verse-olioilla kuvatut sanotussäkeistöt voidaan liittää Notation-olioon. Kaavioon ei kuitenkaan ole piirretty nuolia Harmonisation- ja Verse-luokista Notation-luokkaan, koska Notation-olioon ei edes välillisesti talleteta viitettä Harmonisation- eikä Verse-olioihin, vaan ainoastaan näiden sisältämiin olioihin.



Kuva 6.6: Notaatioluokkien keskinäiset viittaussuhteet

BarLine

BarLine-olio edustaa tahtiviivaa, joka voi olla tavallinen tahtiviiva tai kertauksen aloittava tai lopettava tahtiviiva. Tahtiviivan tyyppi ilmaistaan olion attributtina.

BeamGroup

BeamGroup-olio edustaa palkilla yhteen sidottujen nuottien ryhmää. Palkitetut nuotit esitetään BeamGroup-olion sisällä Note-olioina.

Breath

Breath-olio edustaa hengityspaikkaa.

Chord

Chord-olio edustaa reaalisointumerkkiä. Olio tallettaa soinnun eriteltynä soinnun kanta- ja bassosäveliin, jotka esitetään Pitch-olioina, sekä soinnun laatuun, joka esitetään merkkijonona. Sointu määrittelee transponointiopeeraation, joka toteutetaan transponoimalla soinnun kanta- ja bassosävel Pitch-luokan transponointiopeeraation avulla.

Duration

Duration-olioita käytetään ilmaisemaan nuotin tai tauon kestoa. Kesto määritellään kahden attribuutin avulla. Denominator-attribuutti määrittelee, kuinka moneen osaan kokonuotti pitää jakaa, jotta saadaan haluttu, pistetön nuottiarvo. Dots-attribuutti kertoo, montako pistettä nuottiin on liitetty. Taulukossa 6.1 esitetään muutamia esimerkkejä siitä, kuinka näiden attributtien yhdistelmillä saadaan ilmaistua nuottien kestot.

Taulukko 6.1: Kestojen ilmaiseminen

denominator	dots	merkitys
1	0	kokonuotti
2	0	puolinuotti
4	0	neljäsosanuotti
4	1	pisteellinen neljäsosanuotti
4	2	kaksoispisteellinen neljäsosanuotti

Harmonisation

Harmonisation-olio edustaa reaalisointumerkein kirjoitettua harmonisaatiota lauluun. Teknisesti kyseessä on lista Chord-, Skip- ja BarLine-olioita.

Notation-luokka määrittelee operaation setHarmonisation. Tällä operaatilla on Harmonisation-tyyppinen parametri. Kun melodiaan liitetään harmonisaatio setHarmonisation-operaation avulla, liitetään harmonisaation sisältämät soinnut melodian nuotteihin ja taukoihin oletusarvoisesti siten, että jokainen nuotti ja tauko saa oman sointumerkkinsä. Harmonisation-olio ei

siis määrittele sointujen ajallista kestoja, vaan sointujen kesto määräytyy melodian nuottien ja taukojen kestoista. Koska usein halutaan kuitenkin, että sama sointu jatkuu useamman nuotin tai tauon ajan, voidaan Harmonisation-olion sointulistaan tallettaa sointua edustavien Chord-olioiden lisäksi nuotin tai tauon ohittamista edustavia Skip-olioita. Menettely ei mahdollista sointumerkin sijoittamista nuottien ja taukojen väleihin, mitä voidaan pitää melko vakavana puutteena ja tärkeänä jatkokehityskohteena Veisuuvekottimessa.

Harmonisaation-olion sointulistaan voidaan tallettaa sointujen väliin myös tahtiviivoja edustavia BarLine-oliota. Kun sointuja melodiaan yhdistettäessä kohdataan sointulistalla BarLine-olio, joka edustaa tavallista tahtiviivaa, ladotaan sitä seuraava sointu vasta seuraavaan tahtiin melodiassa. Mikäli sointulistalla kohdattu BarLine-olio edustaa kerrattavan jakson aloitettavaa tahtiviivaa, yhdistetään sitä seuraava sointu siihen nuottiin tai taukoon, joka esiintyy melodiassa seuraavan tahtiviivan, joka aloittaa kerrattavan jakson, jälkeen. Vastaavasti menetellään, mikäli sointulistalla kohdattu BarLine-olio edustaa kerrattavan jakson lopettavaa tahtiviivaa.

Interval

Interval-olio kuvaa intervallin laajuutta ja suuntaa. Intervalli esitetään Interval-oliossa kahden attribuutin avulla. Interval-attribuutti kertoo, onko kyseessä priimi, sekunti tai terssi ja niin edelleen, quality-attribuutti taas, mikä on intervallin laatu (suuri, pieni, vähennetty, ...). Mikäli interval-attribuutin arvo on positiivinen, kuvaa olio ylöspäin suuntautuvaa intervallia, kun taas negatiiviset interval-attribuutin arvot tarkoittavat intervallin suuntautumista alaspäin. Interval-attribuutin arvojen merkitykset esitetään taulukossa 6.2, quality-attribuutin arvojen merkitykset taulukossa 6.3. Attribuuttien yhdistelmien merkityksistä annetaan esimerkkejä taulukossa 6.4.

KeySignature

KeySignature-olio edustaa etumerkintää. Olio määrittelee ainoastaan etumerkinnän ottamatta kantaa, onko kyse duurista, mollista vai kenties josta-kin muusta säveljärjestelmästä. Oliolla on yksi kokonaislukutyypinen attribuutti, joka ilmaisee etumerkkien laadun ja määrän. Positiivinen kokonaisluku tarkoittaa ylennysten määrää, negatiivinen alennusten.

Etumerkinnällä ei ole sävelkorkeutta, eikä näin ollen ole välttämättä aivan oikeellista puhua etumerkinnän transponoimisesta. KeySignature-luokka määrittelee kuitenkin transpose-operaation, koska laulua transponoitaessa joudutaan etumerkintääkin muuttamaan. KeySignature-luokan transpose-operaation toiminta voidaan ymmärtää seuraavasti.

Taulukko 6.2: Interval-olion interval-attribuutin merkitys

arvo	merkitys
-7	alaspäinen oktaavi
-6	alaspäinen septimi
-5	alaspäinen seksti
-4	alaspäinen kvintti
-3	alaspäinen kvartti
-2	alaspäinen terssi
-1	alaspäinen sekunti
0	priimi
1	ylöspäinen sekunti
2	ylöspäinen terssi
3	ylöspäinen kvartti
4	ylöspäinen kvintti
5	ylöspäinen seksti
6	ylöspäinen septimi
7	ylöspäinen oktaavi

Taulukko 6.3: Interval-olion quality-attribuutin merkitys

arvo	merkitys
-2	vähennetty tai kaksoisvähennetty
-1	pieni tai vähennetty
0	suuri tai puhdas
1	ylinouseva
2	kaksoisylinouseva

Taulukko 6.4: Interval-olion attributtien yhdistetty merkitys (esimerkkejä)

interval-arvo	quality-arvo	yhteismerkitys
0	0	puhdas priimi
1	0	suuri sekunti
1	-1	pieni sekunti
1	1	ylinouseva sekunti
2	0	suuri terssi
2	1	ylinouseva terssi
2	-1	pieni terssi
2	-2	vähennetty terssi

Kuvitellaan, että etumerkinnällä, josta tiedetään vain etumerkkien laatu ja määrä, viitataan duurisävellajiin. Tällöin etumerkkien laadusta ja määräs-
tä voidaan päätellä, mikä on kyseisen duurisävellajin kantasävel. Tämä kan-
tasävel voidaan transponoida samoin kuin nuotitkin transponoidaan. Etu-
merkinnän transponointi onkin SongLibrary-järjestelmässä teknisesti toteu-
tettu siten, että luodaan oletetun duurisävellajin kantasävelen sävelkorkeutta
edustava Pitch-olio ja käytetään tämän olion transpose-operaatiota. Pitch-
olioilla kuvataan myös nuottien sävelkorkeuksia, joten myös nuotit transpo-
noidaan Pitch-olion transpose-operaation avulla. Kun kantasävel on trans-
ponoitu, selvitetään, millainen etumerkintä vastaa duuria, jonka kantasävel
on sävel, johon transponoinnilla päädyttiin. KeySignature-olioon talletetaan
tämä etumerkintä.

Lyrics

Lyrics-olio edustaa laulun sanoitusta. Laulun säkeistöt talletetaan Lyrics-
olioon yksittäisiä säkeistöjä edustavina Verse-olioina siten, että säkeistöä tal-
letettaessa talletetaan myös tieto säkeistön numerosta.

Notation

Notation-olio edustaa laulun melodiaa, johon voi liittyä sanoitus ja soinnu-
tus. Melodia talletetaan notation-olioon listana, jossa voi esiintyä seuraa-
via olioita: KeySignature, TimeSignature, Note, Rest, BeamGroup, BarLine,
Breath ja SystemBreak. Notation-luokka määrittelee setLyrics-operaation,
jolla on Verse-tyyppinen parametri. Tämän operaation avulla melodiaan
voidaan liittää yksittäisen säkeistön sanat. Tällöin säkeistön tavut liite-

tään melodian nuotteja kuvaaviin Note-oloihin Syllable-olioiden muodossa. Notation-luokka määrittelee myös setHarmonisation-operaation, jolla on Harmonisation-tyyppinen parametri. Tämän operaation avulla voidaan Harmonisation-olion kuvaama reaalisoitumerkein kirjoitettu soinnutus liittää Notation-olion nuotteja ja taukoja kuvaaviin Note- ja Rest-oloihin Chord-olioiden muodossa. Menettely ei mahdollista sointumerkkien sijoittamista nuottien tai taukojen väleihin, mitä voidaan pitää melko pahana puutteena ja tärkeänä jatkokehityksen kohteena SongLibrary-järjestelmässä. Operaatiota on kuvattu tarkemmin Harmonisation-luokan kuvauksessa.

Notation-luokka määrittelee transponointioperaation, jonka avulla Notation-olion edustama laulu soinnutuksineen saadaan transponoitua uuteen sävellajiin. Operaatio toteutetaan siten, että olion sisältämä aliolioiden lista käydään läpi, ja jokaisen olion, jonka luokka määrittelee transponointioperaation, transponointioperaatiota sovelletaan. Lopulta kaikki transponointi tapahtuu sävelkorkeutta edustavan Pitch-luokan transponointioperaation avulla.

Note

Note-olio edustaa yksittäistä nuottia. Nuotin kesto ilmaistaan Duration-olion muodossa, korkeus Pitch-olion muodossa. Nuottiin voi liittyä sointumerkki, joka ilmaistaan Chord-oliona ja sanoituksen tavu, joka ilmaistaan Syllable-oliona.⁶ Note-luokalla on siis Duration-, Pitch-, Chord- ja Syllable-tyyppiset attribuutit⁷. Note-luokka määrittelee transponointioperaation, joka toteutetaan nuotin korkeutta edustavan Pitch-luokan transponointioperaation avulla.

Pitch

Pitch-luokka edustaa sävelkorkeutta. Luokkaa käytetään nuotteja edustavan Note-luokan lisäksi myös sointumerkkiä edustavan Chord-luokan ja etumerkintää edustavan KeySignature-luokan yhteydessä. Chord-olioiden yhteydessä sointujen kanta- ja bassosävel ilmaistaan Pitch-olion avulla. KeySignature-luokan yhteydessä Pitch-oliota käytetään apuna, jotta etumerkintä saadaan transponoitua (kts. KeySignature-luokan kuvaus).

⁶Note-luokka tukee useammankin tavun liittämistä samaan nuottiin, mutta Notation-luokka ei tarjoa välineitä useamman kuin yhden tavun liittämiseksi yksittäiseen nuottiin. Käytännössä Note-luokka siis tukee useamman säkeistön latomista nuottikuvaa seuraten, Notation-luokka ei.

⁷Tarkkaan ottaen tavuattribuutin tyyppi on Syllable-tilukko, koska Note-luokka tukee useammankin tavun liittämistä samaan nuottiin.

Pitch-luokka määrittelee transponointioperaation, jolla on Interval-tyyppinen parametri. Melodian intervallisuhteiden säilymisen kannalta on oleellisen tärkeää, että transponoinnin määrä ilmaistaan intervallina eikä puolisävelaskelina. Kun laulua transponoitaessa jokaisen yksittäisen transponoinnin laajuus, kohdistuipa se sitten nuottiin, etumerkintään tai reaalisointumerkkiin, ilmaistaan intervallina, säilyvät laulun rakenneosasten keskinäiset intervallisuhteet transponoinnissa.

Rest

Rest-oliot edustavat taukoja. Tauon pituus ilmaistaan Duration-olion muodossa. Taukoon voi liittyä reaalisointumerkki, joka liitetään taukoon Chord-oliona.

Skip

Skip-olioiden avulla merkitään harmonisaatioon, että sointuja melodian nuotteihin ja taukoihin yhdistettäessä halutaan hypätä nuotin tai tauon ylitse liittämättä siihen sointua. Skip-olioiden käyttämisestä kerrotaan Harmonisation-luokan kuvauksessa.

Syllable

Syllable-olio edustaa sanoituksen yksittäistä tavua. Olio sisältää tiedon siitä, kuinka monen nuotin ajan tavu kestää sekä siitä, onko kyse sanan viimeisestä tavusta. Jälkimmäistä tietoa tarvitaan, jotta tavuja nuottien yhteyteen ladottaessa tiedetään, tuleeko tavun jälkeen piirtää tavuviiva.

SystemBreak

SystemBreak-olioilla merkitään kohdat, joissa nuottien ladonnassa tulee siirtyä seuraavalle viivastolle.

TimeSignature

TimeSignature-oliot edustavat tahtiosoituksia. Tahtiosoitukset, jotka voidaan ilmaista murtolukuna, talletetaan TimeSignature-olioon kahdena kokonaislukuna (osoittajana ja nimittäjänä). Erikoistapauksina TimeSignature-olio mahdollistaa C- ja alla breve -tahtiosoitusten tallettamisen. TimeSignature-olio ei varsinaisesti tue virsissä esiintyviä yhdistettyjä tahtilajeja. Tämä puute kierretään Veisuuvekottimessa luomalla yhdistettyä tahtilajia edustamaan TimeSignature-olio, jonka osoittaja-attribuutti on

yhdistetyn tahtilajien osoittajien yhdistelmä ja jonka nimittäjä-attribuutti on yhdistetyn tahtilajin nimittäjien yhdistelmä. Kun virsikoodimuotoisesta esityksestä luodaan oliomalli, luodaan esimerkiksi tahtiosoituksesta, joka virsikoodina ilmaistaan 3/2-2/2 KeySignature-olio, jonka osoittaja-attribuutti saa arvon 32 ja nimittäjä-attribuutti arvon 22. Näin voidaan toimia, koska yhdistettyihin tahtiosoituksiin tottuneet ihmiset tulkitsevat nuottikuvassa esiintyvän tahtiosoituksen 32/22 yhdistettynä tahtiosoitukseksi, jonka osat ovat 3/2 ja 2/2.

UnknownMarking

UnknownMarking-olioita käytetään edustamaan tekstimuotoisen esityksen kohtia, joita Veisuuvekkottimen jäsentäjä ei kyennyt tulkitsemaan.

6.2.6 ABC-muotoisen esityksen tuottaminen oliomallista

Veisuuvekkotin tuottaa nuottikuvat abcjs-nimisen JavaScript-kirjaston [abcb] avulla⁸. Laulun data on annettava abcjs-kirjastolle abc-muodossa. Niinpä Veisuuvekkottimen oliomalli on varustettu välineillä abc-muotoisen esityksen tuottamiseksi oliomallin esittämästä laulusta.

Koska Notation-olio edustaa laulun kokonaisuutta, on operaatio abc-muotoisen esityksen tuottamiseksi laulusta sijoitettu Notation-luokkaan. Tämä metodi tuottaa abc-muotoisen esityksen hyödyntämällä muiden notaatioluokkien⁹ operaatioita, jotka tuottavat abc-muotoisen esityksen edustamaan laulun osatekijästä. Esimerkiksi Note-luokassa on operaatio, joka palauttaa yksittäisen nuotin abc-muotoisen esityksen. Menetelmää voidaan verrata Notation-olion transponoimiseen, joka toteutetaan muiden notaatioluokkien transponointioperaatioiden avulla.

Yksittäisen olion edustaman asian abc-muotoinen esitys on joissakin tapauksissa riippuvainen olion edeltävistä olioista. Esimerkiksi Note-olion edustaman nuotin abc-esitys on riippuvainen edeltävästä KeySignature-oliosta. Niinpä abc-esitystä tuotettaessa luodaan olio, joka edustaa kontekstia, jossa abc-muotoisen esityksen tuottaminen tapahtuu. Kullekin oliolle, jolta abc-muotoista esitystä pyydetään, annetaan viite tähän kontekstiolioon, jolloin oliot voivat muokata kontekstietoa. Esimerkiksi KeySignature-olion operaatio, jolla tuotetaan etumerkinnän abc-muotoinen esitys, merkitsee etu-

⁸Kirjasto on tässä ohjelmointitekniinen termi, joka selitettiin luvussa 6.2.1.

⁹Notaatioluokalla tarkoitetaan tässä luokkaa, josta luotuja olioita voidaan tallettaa Notation-olioon. Notaatioluokkia ovat näin ollen KeySignature, TimeSignature, Note, Rest, BeamGroup, BarLine, Breath ja SystemBreak.

merkinnän muistiin kontekstiolioon. Myöhemmin hyödynnettävät Note-olion operaatiot, jotka tuottavat nuotin abc-muotoisen esityksen, saavat viitteen tähän kontekstiolioon, jolloin operaatiot voivat huomioda voimassa olevan etumerkinnän.

6.2.7 Käyttöliittymäluokat

Yleiskatsaus

Käyttöliittymäluokiksi on tässä laskettu Veisuuvekottimen luokat, jotka eivät osallistu nuotti-informaation esittämiseen tarvittavan oliomallin muodostamiseen, vaan joiden avulla toteutetaan palvelun muu toiminnallisuus, kuten tarkasteltavan kirjan tai laululistan valinta. Nämä luokat käydään myöhemmin läpi yksitellen aakkosjärjestyksessä.

Käyttöliittymäluokkien välinen kommunikaatio on järjestetty suurelta osin Tarkkailijaksi kutsutun suunnittelumallin (Observer design pattern) mukaisesti¹⁰. Kuvassa 6.7 esitetään olioiden väliset tarkkailusuhteet. Nuolet on piirretty tarkkailijasta tarkkailtavaan päin. StepTranspose-luokkaan ei liity lainkaan nuolta, koska se ei ole kumpanakaan osapuolena tarkkailusuhteessa.

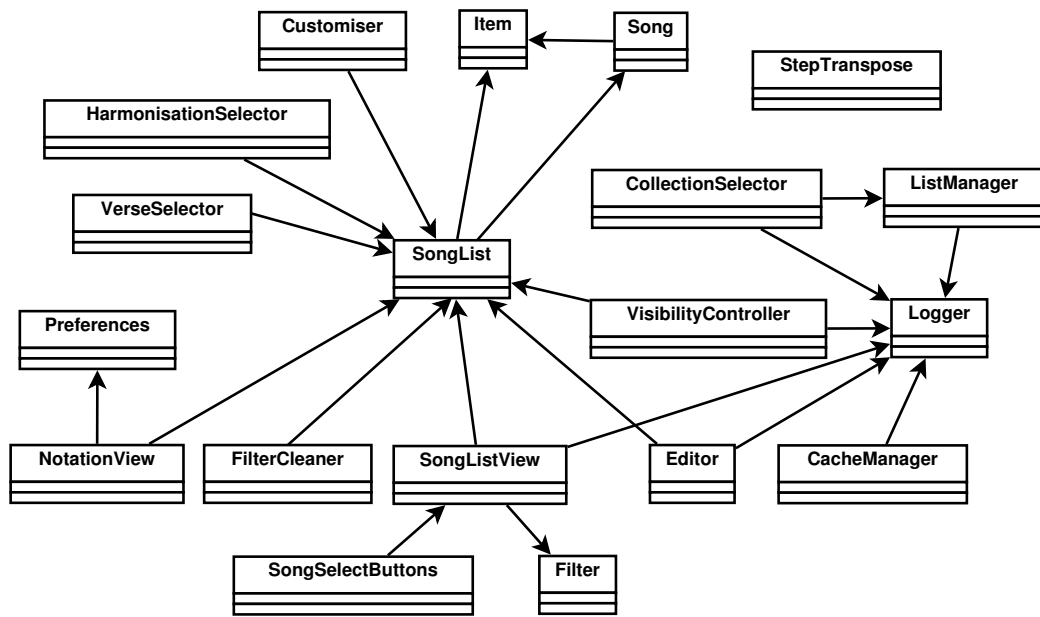
Tarkkailtavan ei tarvitse tietää tarkkailijoistaan muuta kuin että tarkkailija voi kiinnittää niiden huomion itseensä tarkkailijoiden notify-operaatioiden avulla. Tarkkailijoiden on kuitenkin tiedettävä, mitä attribuutteja tai operaatioita tarkkailtava tarjoaa tilansa selvittämiseksi.

Kuvassa 6.8 esitetään puolestaan, mitkä oliot muuttavat minkäkin olioiden tilaa. Oliosta, joka muuttaa toisen olion tilaa, on piirretty yhtenäisellä viivalla nuoli oloon, jonka tilaa muutetaan. SongList-oliosta on piirretty nuoli katkoviivalla Item- ja Song-olioihin, koska SongList luo nämä oliot mutta ei muuta luomiensa olioiden tilaa. Song-oliosta Item-olioon kulkeva nuoli on piirretty pisteviivalla, koska Song-olio muuttaa Item-olion tilaa ainoastaan luontivaiheessaan. Oliot, joihin ei kiinnity lainkaan nuolta on otettu kuvaan mukaan, jotta helposti nähdään, mitkä oliot eivät osallistu kuvattaviin suhteisiin kumpanakaan osapuolena.

Olioiden välisen vuorovaikutuksen hahmottamiseksi molempia kuvia on tutkittava yhdessä. Olettakaamme esimerkiksi, että StepTranspose-olio muuttaa Item-olin tilaa, kuten kuvan 6.8 mukaisesti voi tapahtua. Kuvasta 6.7 nähdään, että SongList-olio tarkkailee Item-olioita, joten SongListille välittyy tieto siitä, että Item-olion tila on saattanut muuttua. SongList-oliota puolestaan tarkkailee muun muassa NotationView-olio. Niinpä NotationView-olio saa tiedon siitä, että SongList-olion tila on saattanut

¹⁰Observer-suunnittelumalliin voi tutustua tarkemmin esimerkiksi artikkelista Observer [Micb].

muuttua. NotationView-olio tietää tällöin hakea ajantasaisen datan, jota se tarvitsee nuottikuvan piirtämiseen. Näin StepTranspose -olio pääsee välillisesti vaikuttamaan siihen, millaisesta datasta NotationView-olio piirtää nuottikuvan. StepTranspose-olion ei kuitenkaan tarvitse tuntea NotationView-oliota. Riittää, kun se tuntee Item-olion. Item-olion ei myöskään tarvitse tuntea NotationView-oliota. Niinpä NotationView-luokka voitaisiin korvata uudella, johonkin tarkoitukseen paremmin soveltuvalla, nuottikuvia piirtävällä luokalla tekemättä muutoksia StepTranspose- ja Item-luokkien ohjelmakoodiin.



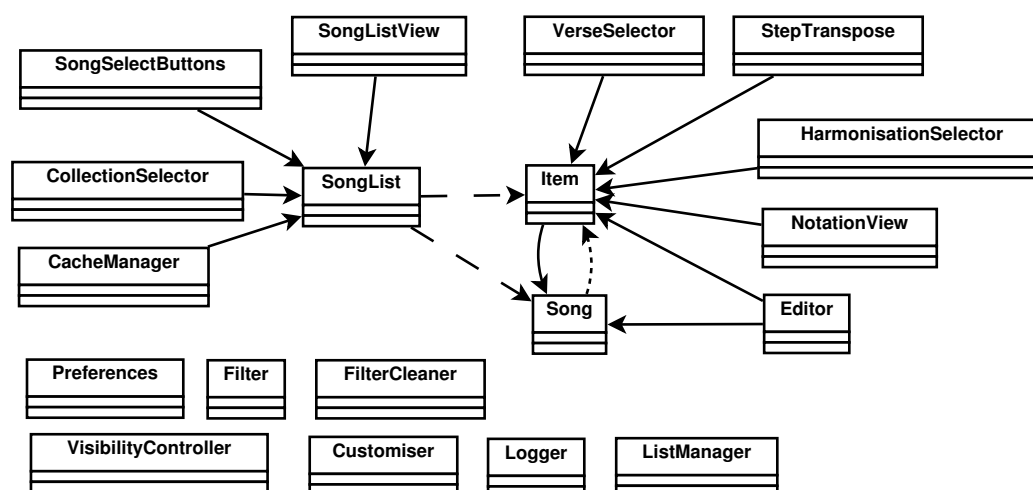
Kuva 6.7: Käyttöliittymäolioiden keskinäiset tarkkailusuhteet

CacheManager

CacheManager toimii linkkinä Logger-olion ja SongList-olion välissä kuuntelemalla Logger-olion tilan muutoksia ja välittämällä näistä tiedon SongList-oliolle. Tällainen välittäjä mahdollistaa sen, ettei Logger-olion tarvitse tuntea SongList-oliota eikä SongList-olion tarvitse tuntea Logger-oliota.

CollectionSelector

CollectionSelector lataa SongList-olioon käyttäjän valitseman laulukirjan tai -listan.



Kuva 6.8: Mitkä käyttöliittymäoliot muuttavat minkäkin olioiden tilaa?

Customiser

Customiser-luokan avulla voidaan tallettaa laulun transponointi- ym. asetukset siten, että asetuksia sovelletaan lauluun, kun laulu avataan kirjan sisällysluettelosta käsin. Customiser tallettaa siis listoista riippumattomat oletusarvot tietylle käyttäjän ja laulun yhdistelmälle.

Editor

Editor-luokka tarjoaa verkkopalvelun käyttäjälle mahdollisuuden muokata laulujen dataa: teostietoja, melodiaa, sanoituksia ja soinnutuksia. Editor-luokka huolehtii myös siitä, että käyttäjän annetaan muokata vain niitä lauluun liittyviä tietoja, joiden muokkaamiseen hänellä on oikeus.

Filter

Filter-luokka suodattaa SongListView-luokan näkymää siten, että vain ne laulut, jotka täyttävät käyttäjän määrittelemän hakuehdon, näytetään laulujen listassa.

FilterCleaner

FilterCleaner-olio tyhjentää Filter-olioon liittyvän hakukentän, kun sisällysluettelon lähde (kirja tai laululista) vaihtuu.

HarmonisationSelector

HarmonisationSelector tarjoaa käyttäjälle välineet halutun soinnutuksen valitsemiseksi.

Item

Item edustaa yksittäistä laulua kirjan tai listan sisällysluettelossa. Ennen laulun lataamista Item sisältää ainoastaan sisällysluettelossa tarvittavat tiedot laulusta. Kun laulu ladataan, laulun koko data liitetään Item-olioon Song-olion muodossa.

Item-oliolla on seuraavat attribuutit:

songId vastaa laulun yksikäsitteistä tunnistetta tietokannassa.

transposeInterval kertoo Interval-olion muodossa, paljonko laulua tulee transponoida.

selectedVerses kertoo RangeSet-olion muodossa, mitkä säkeistöt laulusta on valittu mukaan.

selectedVerse ilmoittaa kokonaislukuna, mikä säkeistö parhaillaan on "menossa".

harmonisationId on yksikäsitteinen tunniste, joka yksilöi lauluun harmonisoinnin.

songNumber on laulun numero. Virsikirjan tapauksessa tämä kenttä sisältää myös mahdollisen laulun varianttiin viittaavan kirjaimen, jolloin kentän sisältö voi olla esimerkiksi "28b".

firstWords sisältää laulun alkusanat, jotta nämä voitaisiin listata sisällysluettelossa.

song sisältää viitteen laulun koko dataan Song-olion muodossa, mikäli laulu ollaan ladattu.

Item-luokassa on lisäksi operaatiot valittujen säkeistöjen, parhaillaan meilläään olevan säkeistön, transponoinnin ja harmonisoinnin valitsemiseen. Nämä operaatiot asettavat vastaavien attributtien arvon, käyttävät Itemiin liittyvän Song-olion updateNotation-operaatiota, jotta Song-olio tietäisi päivittää notaatiomallinsa vastaamaan uusia asetuksia ja huomauttavat Item-olion tarkkailijoille, että olion tilaan on tullut muutoksia. Kun Item-olion tilaa muutetaan näitä metodeita kutsumalla, pysyvät sekä Itemiin liittyvä Song-olio että Item-olion tarkkailijat ajan tasalla muutoksista.

ListManager

Huolehtii laulujen lisäämisestä listoihin ja laulujen poistamisesta listoista.

Logger

Logger-luokka huolehtii käyttäjän kirjautumisesta palveluun ja ulos palvelusta.

Preferences

Preferences-olio huolehtii kaikkien laulujen näyttämiseen liittyvistä asetuksista. Näillä asetuksilla voidaan valita, huomioidaanko laulukohdaiset transponointiasetukset ja säkeistövalinnat, vai näytetäänkö laulut alkuperäisessä sävellajissaan kaikkien säkeistöjensä kera.

Song

Song-olio edustaa yksittäistä laulua. Laulun data pidetään Song-oliossa tallessa kahdessa eri muodossa: merkkijonoina ja oliomallina. Oliomallia tarvitaan, jotta dataa voitaisiin helposti käsitellä eri soinnutusten ja säkeistöjen yhdistämiseksi melodiaan ja laulun transponoimiseksi. Merkkijonomuotoista esitystä tarvitaan, kun laulua muokataan tekstieditorilla.

Veisuuvekottimen tietokannassa lauludata on merkkijonoina, virsikoodina. Kun laulu on ladattu ja oliomalli luotu, voitaisiin merkkijonomuotoinen esitys unohtaa ja rekonstruoida se oliomallin pohjalta, mikäli käyttäjä avaa editorin. Esitys, johon näin päädyttäisiin, ei kuitenkaan olisi välttämättä sama kuin tietokannasta ladattu esitys. Tämä johtuu siitä, että virsikoodi sallii kirjoittaa saman asian useammalla eri tavalla: esimerkiksi rivinvaihtokohdilla ei ole merkitystä luotavan oliomallin kannalta. Veisuuvekotin on suunniteltu siten, että järjestelmä ei koskaan muokkaa virsikoodimuotoisia esityksiä, vaan ne säilyvät siinä muodossa, johon käyttäjä on ne alun perin kirjoittanut tai johon käyttäjä on ne itse muokannut.

Sanoitusta kuvataan Song-oliossa Lyrics-oliolla, jossa jokaista säkeistöä kuvataan Verse-oliolla, jossa taas jokaista tavua kuvataan Syllable-oliolla. Jokaista vaihtoehtoista harmonisaatioita kuvataan Harmonisation-oliolla, jossa jokaista sointua edustaa Chord-olio. Melodian, valitun harmonisaation ja menossa olevan säkeistön yhdistelmää kuvataan Notation-olion avulla. Notation-olioita luodaan Song-olioon kaksi, joista toinen kuvaa laulua transponoimattomana, toinen transponoituna siten kuin Item-olio, johon Song-olio liittyy, määrittelee. Notation-olio sisältää kaiken datan, jota tarvitaan

laulun nuottikuvan piirtämiseen. Song-olio asetetaan tarkkailemaan Item-oliota, johon Song-olio liittyy, ja Song-olioon talletetaan viite tähän Item-olioon. Kun Item-olion tila muuttuu, Item-olio huomauttaa tästä Song-oliolle, jolloin Song-olio lukee Item-oliosta tiedon siitä, mikä sanoitussäkeistö ja harmonisointi pitäisi liittää Song-olion sisältämiin Notation-oloihin, ja paljonko transponoitavaa Notation-oliota pitäisi transponoida. Tiedot saatuaan Song-olio päivittää Notation-olionsa vastaamaan Item-olion tilaa.

SongList

SongList edustaa yksittäisen laulukirjan tai laululistan sisältöä. SongList sisältää listan Item-olioita.

SongSelectButtons

SongSelectButtons-luokka huolehtii painikkeista, joilla siirrytään seuraavaan, edelliseen tai ensimmäiseen laululistan lauluun.

StepTranspose

StepTranspose-luokka toteuttaa transponointikäyttöliittymän, jonka avulla laulu voidaan transponoida seuraavaan ylempään tai alempaan sävellajiin, jossa ylennyksiä ja alennuksia on korkeintaan käyttäjän sallima määrä. Käyttäjä voi määritellä suurimman sallitun ylennysten ja alennusten määrän erikseen.

VerseSelector

VerseSelector toteuttaa toiminnon, jonka avulla käyttäjä voi valita, mitkä säkeistöt hän haluaa mukaan laulun nuottiin.

VisibilityController

Hoitaa käyttöliittymän osien näyttämistä ja piilottamista.

6.2.8 Tietoturva

Palveluun kirjautuminen

Veisuuvekotinta voi käyttää kirjautumatta palveluun. Kirjautumaton käyttäjä ei kuitenkaan voi lisätä uusia lauluja eikä soinnutuksia, eikä saa käyttöönsä Veisuuvekottimen laululistaominaisuutta. Veisuuvekottimessa ei toistaiseksi

ole toimintoa uusien käyttäjätunnuksien luomiseksi tai aiemmin luodun käyttäjän tietojen (salasana, nimi) päivittämiseksi. Nämä toimenpiteet on tehtävä käsittelemällä Veisuuvekottimen tietokantaa jollakin muulla välineellä.

Palvelu pitää kirjaa siitä, kuinka monta kertaa tietyllä käyttäjätunnuksella on yritetty kirjautua väärällä salasanalla a) viimeisimmän onnistuneen kirjautumisen jälkeen ja b) sen jälkeen, kun salasana on viimeksi vaihdettu. Mikäli samalla käyttäjätunnuksella yritetään kirjautua yli kolmekymmentä kertaa peräkkäin ilman yhtään onnistunutta kirjautumista, palvelu sulkee käyttäjätunnuksen. Käyttäjätunnus suljetaan myös, jos samalla käyttäjätunnuksella yritetään kirjautua yli tuhat kertaa epäonnistuneesti ilman että salasanaa vaihdetaan välillä. Onnistunut kirjautuminen siis nolaa toisen laskurin, salasanan vaihtaminen toisen. Kirjautumisyrityksiä rajoitetaan, jotta murtautuja ei voisi arvata salasanaa loputtomasti uudelleen käyttämällä tarkoitukseen laadittua tietokoneohjelmaa. Vääriä kirjautumisia sallitaan melko paljon kahdesta syystä. Ensiksikin käyttäjät unohtavat salasanojaan, jolloin rehellinenkin kirjautuja saattaa yrittää väärää salasanaa toistuvasti. Toisekseen halutaan tehdä käyttäjätunnusten sulkeminen ilkeävaltamielessä hieman työläemmäksi. Jos väärällä salasanalla kirjautumisia sallittaisiin vain muutama, olisi helppoa sulkea kenen tahansa käyttäjän käyttäjätunnus yksinkertaisesti kirjautumalla muutaman kerran väärällä salasanalla. Kolmekymmentä kirjautumisyritystä vaatii jo hieman vaivannäköä. Ratkaisu tekee käyttäjätunnusten sulkemisen toki edelleen varsin helpoksi, joten kirjautumiseen liittyvien uhkien torjumiseksi tarvittaisiin muitakin menettelytapoja kuin kirjautumisyritysten lukumäärän rajaaminen.

SQL-injektiot

SQL-injektiot torjutaan Veisuuvekottimessa käyttämällä prepared statements -mekanismia aina, kun tietokantaan ollaan kirjoittamassa käyttäjältä saatua dataa.

HTML-injektiot

HTML-injektiot torjutaan Veisuuvekottimessa seuraavasti. Aina, kun sivun HTML-koodiin ollaan lisäämässä käyttäjän kirjoittama merkkijono, korvataan tässä merkkijonossa esiintyvät, taulukossa 6.5 esitetyt erikoismerkit näiden merkkien HTML-koodeilla. Näin merkkien erityismerkitys poistuu. Tarkoitukseen käytetään Veisuuvekotinta varten kirjoitettua funktiota `substitute_html_entities`. Valitussa toimintatavassa käyttäjän syöttämä data talletetaan tietokantaan sellaisenaan. Näin toimitaan, jotta käyttäjän tekstikenttiin syöttämä data saataisiin tietokannasta takaisin editoitavaksi

käyttäjän syöttämässä muodossa tarvitsematta dekodata tallennusvaiheessa koodattuja erikoismerkkejä.

Taulukko 6.5: Korvattavat HTML-merkit

Merkki	Korvaava koodi
"	"
'	&apos
&	&
<	<
>	>

6.2.9 Kehitysympäristö ja testaaminen

Veisuuvekottimen palvelinympäristönä on projektin loppuvaiheessa toiminut `nettihotelli.fi` -webhotelli. Nettihotellin palvelimessa, jolla `veisuuvekotin.net`-sivusto toimii, on käyttöjärjestelmänä Debian 6 Squeeze GNU/Linux-käyttöjärjestelmä. Veisuuvekottimen tietokantaa hallitaan nettihotellissa MySQL 5.5 -tietokannanhallintajärjestelmällä ja asennettu PHP on 5.3-sarjaa.¹¹

Veisuuvekotinta on kehitystyön aikana käytetty lähinnä Firefox-selaimella tietokoneessa, jonka käyttöjärjestelmänä on Mac OS X¹². Laitteistona on toiminut MacBook, jossa on 2 GHz Intel Core 2 Duo -prosessori, 4 Gt 667 MHz DDR2 SDRAM -muistia ja 13 tuuman näyttö. Työskentelyergonomian parantamiseksi hyödynnettiin myös erillistä, isompaa näyttöä.

6.3 Jakelu ja lisensointi

Veisuuvekotin hyödyntää kolmea JavaScript-kirjastoa, jotka toimitetaan Veisuuvekottimen mukana. Nämä kirjastot ovat jQuery, jQueryUI ja abcjs. jQuery ja jQueryUI on julkaistu MIT-lisenssin [mit] alaisuudessa. Lisenssin perusajatus on antaa vapaat oikeudet käyttää ja jakaa eteenpäin lisenssin alaisuudessa julkaistua koodia, kunhan koodin yhteydessä säilytetään maininta siitä, että se on julkaistu MIT-lisenssin alaisuudessa. abcjs-kirjasto on julkaistu tämän dokumentin liitteenä E olevan GNU GPL v3 -lisenssin [Fou]

¹¹21.8.2013

¹²Tämän kirjoittamishetkellä, 21.8.2013, käytetyn Firefox-selaimen versio on 23.0.1 ja Mac OS X:n 10.6.8

alaisuudessa [abcb]. abcjs-kirjasto sisältää Raphael-vektorigrafiikkakirjaston [rap], joka on julkaistu MIT-lisenssin alaisuudessa.

Kirjoittaja pitää itsellään Veisuuvekottimen lähdekoodin tekijänoikeudet, mutta julkaisee koodin osoitteessa `veisuuvekotin.net` ja sallii sen käyttämisen GPL-lisenssin ehtojen mukaisesti. Lisenssi antaa kenelle tahansa luvan käyttää ohjelmakoodia, jaella sitä eteenpäin ja jatkokehittää sitä, kunhan samat oikeudet luovutetaan eteenpäin ja sovelletaan kaikkiin niihin tietokoneohjelmiin, joiden osaksi GPL-lisensoitua ohjelmakoodia otetaan. Myös abcjs-kirjasto on lisensoitu GPL:n alaisuudessa, ja niinpä kirjaston käyttäminen osana Veisuuvekotinta edellyttikin Veisuuvekottimen lisensoimista samalla lisenssillä.

Luku 7

Projektin kulku

Aloitin tietokoneella käytettävän virsikirjan kaavailun keväällä 2008 havaittuani, että suomen ev.-lut. kirkon sivuilla oleva Verkkovirsikirja ei hyödynnä kaikkia mahdollisuuksia, joita verkko julkaisualustana tarjoaisi. Ensimmäinen versioni teemasta perustui ilmaiseen LilyPond-nuotinnusohjelmaan, joka tuottaa kauniita, haluttaessa hyvinkin monimutkaisia nuottikuvia tekstimuodossa annetusta informaatiosta. Ohjelmani oli komentorivipohjainen, eli sitä käytettiin kirjoittamalla tietokoneelle komentoja. Myöhemmin päätin laatia ohjelmasta graafisella käyttöliittymällä varustetun, käyttäjäystävällisemmän version. Virsien tekstien lähteen käytin kirkon sivujen Verkkovirsikirjaa, sävelmien lähteenä taas Kotimaa-yhtiöiden `virret.net`-palvelua. Sekä tekstejä että sävelmiä jouduin kuitenkin editoimaan käsin. Esittelin Laulukirjastoksi nimeämäni tietokoneohjelmaa monille ihmisille ja sain paljon myönteistä palautetta. Kukaan ei huomauttanut minulle, että vastaava tuote olisi jo ennestään olemassa. Myönteistä palautetta tässä vaiheessa antaneista mainittakoon kirkkomusiikin edustajista Juhani Haapasalo ja Timo Kiiskinen Sibelius-Akatemian kirkkomusiikin osastolta, Hanna Koskinen kirkkohallituksesta sekä rippikouluharjoitteluni ohjaajaa Kati Pirttimaa. Sibelius-Akatemian musiikkiteknologian osastolla vaikuttava Otto Romanowski kannusti minua myös.

Elämäntilanteeni muuttuessa päädyin toteamaan, että vapaa-aikani ei riitä Laulukirjaston saattamiseen julkaisukuntoon. Vuoden 2011 keväällä kysyin Timo Kiiskiseltä, soveltaisiko nuottikuvien muokkaamiseen kykenevä, tietokonepohjainen virsikirja kirkkomusiikin opintojen lopputyöprojektiksi. Lähetin Kiiskiselle sähköpostin liitteenä Laulukirjasto-ohjelmasta laatimani esittelyn, jossa selvitin ohjelmaan suunnittelemani piirteitä (kts. liite A). Kiiskinen hyväksyi aiheen ja seminaarini alettua seminaarin vetäjä Jorma Hannikainen lupautui ohjaamaan projektini kirjallista osaa.

Jo ennen opintoprojektin alkamista, kehittäessäni Laulukirjastoa harras-

tuksenani, esittelin hankettani ystävälleni, tietojenkäsittelytieteen opiskelijalle Martin Pärtelille, jolta olen saanut arvokasta apua hankkeeni eri vaiheissa. Pärtel ei ole muusikko, mutta arvostan hänen asiantuntemustaan tietokoneohjelmointiin liittyvissä kysymyksissä. Pärtel vahvisti epäilyni, että Java-appletteja ei kannata käyttää verkkopalveluni toteuttamiseen, mutta piti Java-servlettien käyttämistä täysin mahdollisena valintana¹. Pärtel mainitsi minulle myös sittemmin oleelliseksi projektissani nousseen jQuery-kirjaston sekä AJAX- ja JSON-tekniikat. Myöhemmin keskustelin hänen kanssaan lyhyesti virsikoodin jäsentäjän toteuttamisesta. Ylipäätään olen hankkeeni varrella esitellyt Pärtelille teknologiaan liittyviä valintojani ja kyselty, onko niissä hänen mielestään järkeä.

Lokakuun alussa 2011 olin jo päättänyt, että minun pitäisi toteuttaa virsikirjani verkkopalvelun muodossa. Avoimia kysymyksiä oli paljon. Missä muodossa data, josta nuottikuvat tuotetaan, pitäisi tallettaa? Käytettäisiinkö relaatiotietokantaa, vai mihin data talletettaisiin? Miten nuottikuvat tuotettaisiin datan pohjalta? Mikä osa ohjelmakoodista pitää kirjoittaa itse, mihin tarkoituksiin olisi tarjolla valmista ohjelmakoodia?

Ennen lokakuun loppua löytyi projektin käytännöllisen osan ohjaajaksi diplomi-insinööri ja musiikkiteknologian opiskelija Risto Ranta-aho, jolla oli kokemusta nuottikuvia tuottavan tietokoneohjelman kirjoittamisesta. Ranta-ahon ohjauksesta olen kuullut erityisesti viestin: tee asiat kunnolla. Ranta-ahon ohjaus vahvisti näkemystäni, että minun kannattaa käyttää datan tallettamiseen valmista tietokannanhallintajärjestelmää sen sijaan, että tallettaisin datan itse määrittelemälläni tavalla, ja että minun kannattaa luoda musiikkidatasta oliomalli ja käsitellä tätä mallia sen sijaan, että käsittelisin dataa tekstimuodossa etsi ja korvaa -tyyppisesti.

Harrastuspohjalta laatimani virsikirjaohjelman olin kirjoittanut Javanimisellä ohjelmointikielellä. Javalla on mahdollista toteuttaa dynaamisia verkkosivustoja nk. servlettien avulla, ja alun perin ajattelin, että en tässä projektissa opettelisi uusia ohjelmointikieliä. Olin tietojenkäsittelytieteen opinnoissani Helsingin yliopistossa ollut mukana projektissa, jossa kehitettiin Java-kielellä verkkopalvelua, jonka ajoalustana käytettiin Tomcat-palvelinta. Tässä projektissa en ollut joutunut pystyttämään Tomcat-palvelinta, joten aloin selvittää, miten se tapahtuu.

Lokakuussa 2011 olin löytänyt JavaScript-kielellä toteutetun abcjs-kirjaston, jonka avulla voidaan tuottaa nuottikuvia verkkoselaimessa musiikkidatan abc-muotoisesta esityksestä. Lokakuun lopulla olin päätenyt rat-

¹Appletit ovat käyttäjän verkkoselaimen sisällä ajettavia Java-ohjelmia, servletit taas palvelimella ajettavia Java-ohjelmia, joiden avulla voidaan dynaamisesti tuottaa verkkosivuja. Servlet-tekniikkaa opiskelin muun muassa Servlet & JSP -tutoriaalista [Kur12].

kaisuun, että nuottikuvien tuottamisen pitää tapahtua käyttäjän päätelaitteessa. Samassa kuussa Soitin Pekka Lähteenarolle, Kotimaanetin johtavalle www-suunnittelijalle. Kysyin, saanko käyttää heidän **virret.net**-sivustonsa tarjoamia, midi-muotoisia virsisävelmiä projektissani. Hän antoi luvan. Lopulta en kuitenkaan käyttänyt näitä tiedostoja vaan kirjoitin melodiat käsin. Helsingin messukeskuksessa järjesteyillä Kirjamessuilla tapasin kirjamessuilla Gracevoice Oy:n Joni Nisulan, joka esitteli yhtiönsä tuottamaa eVirsi kirjaa. Nisula mainitsi minulle, että ePub-formaatin uusimmassa versiossa on tuki JavaScriptille, ja ainut vektorigrafikkaformaatti, jota ePub tukee, on SVG, jota abcjs-kirjasto tuottaa. Tämä oli kiintoisaa: JavaScriptillä toteutettu nuottikuvien tuottaminen saattaisi siis olla mahdollista upottaa ePub-muotoiseen tiedostoon siten, että ePub-formaattia tukevat sähköisten kirjojen lukulaitteet kykenisivät tuottamaan nuottikuvia dynaamisesti. Tämäkin puolsi osaltaan nuottikuvien tuottamista JavaScriptillä ja abcjs:llä.

Marraskuussa 2011 Testasin abcjs-kirjaston toimivuutta Applen sormitietokoneessa, iPad2:ssa. Kirjasto vaikutti toimivan laitteessa. Veisuvekotinta en ole iPadissa kokeillut, vaikka Nisula tarjosi ystävällisesti mahdollisuutta lainata Gracevoice Oy:n iPadiä. Osallistuin teologian ja kirkkomusiikin opiskelijoiden yhteiseen seminaarikokoontumiseen. Laadin seminaarilaisille täytettäväksi kyselylomakkeen (kts. liite B), jossa luetteloin tietokonepohjaisen virsikirjan mahdollisia piirteitä ja pyysin ympäröimään piirteet, joita kirjassa pitäisi olla, jotta vastaaja arvelisi itse käyttävänsä sitä. Lomakkeen alussa oli lisäksi kysymyksiä, joilla kartoitin vastaajan musiikillista taustaa.

Kevätlukukauden 2012 alussa aloin kokeilemaan JavaScriptin kirjoittamista Flanaganin JavaScript-kirjan [Fla11] avulla. Tammikuun lopulla vierailin kirkkohallituksessa kertomassa projektistani. Kirkon sivuilla olevan Verkkovirsikirjan toteuttaja Timo Lehtonen totesi, että hankkeeni vastaa mooneen odotukseen². Kirkon tiedotuskeskuksen verkkoviestintäpäällikkö Sami Kallioinen sanoi olevansa vaikuttunut. Osan ominaisuuksista, joita tuolloin esitin tulevaksi palveluuni, jouduin kuitenkin jättämään pois, jotta projekti valmistuisi aikataulussa. Kalvoissa, joita kirkkohallituksessa näytin, kaavailin palveluni osaavan latoa useammankin säkeistön sanat nuottikuvaa seuraamaan. Projektin puitteissa ehdin toteuttaa vain yhden vapaavalintaisen säkeistön ladonnan nuottia seuraten. Sointumerkkien yhteyteen piti olla saatavissa otekaaviot kitaristeja varten. Kaikkien virsien nuotit piti voida ladata yhtenä pdf-tiedostona tulostamista varten. Tällä hetkellä virret pitää tulostaa yksi kerrallaan eikä palvelu tuota pdf-tiedostoja. Ympäristöissä, joissa käyttöjärjestelmä mahdollistaa pdf-tiedoston luomisen mistä tahansa tu-

²Lehtonen osallistui palaveriin etäyhteyden päästä, eikä kuvaa tietokoneeltani saatu siirrettyä hänelle. Lehtonen kommentoi hankettani siis ainoastaan kuulemansa perusteella.

lostettavasta sisällöstä, voidaan pdf-tiedostoja kuitenkin luoda verkkoselaimen tulostustoiminnon avulla. Yksittäinen virsi piti voida ladata MusicXML-tiedostona, joka voitaisiin edelleen ladata esim. Finale- tai Sibelius-ohjelmaan jatkokäsittelyä varten. MusicXML-tiedostojen tuottamista ei toteutettu projektissa aikataulusyistä. Lupailin palvelun toimivan perinteisten pöytä- ja sylimikrojen lisäksi sormitietokoneissa ja älypuhelimissa. Veisuuvekottimen toimivuutta sormitietokoneissa ja älypuhelimissa ei testattu projektin aikana, vaikka onkin perusteita olettaa, että se toimii monissa näistä ainakin jossain määrin. Lisäksi kalvoissa esitettiin, että Veisuuvekottimeen voitaisiin tallettaa soinnutukset ainakin Kirjapajan Soinnutetusta virsikirjasta ja Kitaravirsiä-kirjasta, mikäli lupa saataisiin. Projektissa ei kysytty lupia näiden soinnutusten käyttämiseen eikä niitä siis talletettu.

Maaliskuun 2012 alussa suunnittelin jo oliomallia melodioiden esittämiseen, ja puolella välissä kuuta Virsikoodi alkoi hahmottua. Samassa kuussa osallistuin teologien ja kirkkomuusikoiden yhteiseen symposiumiin Vaiviossa. Teija Pitkänen esitteli seminaarissa keräämäänsä aineistoa, josta ilmenee, mitä virsiä seurakunnissa on laulettu kirkkovuonna 2009–2010. Jenni Urponen puolestaan muistutti minua kouluvirsilistasta ja ilmaisi näkemyksensä, että Veisuuvekotin voisi palvella koulumaailmaa. Päätin käyttää sekä Pitkäsen aineistoa että kouluvirsilistaa valitessani virsiä, jotka projektin puitteissa talletan palveluuni. Omaan esitykseeni symposiumissa en ollut täysin tyytyväinen: minun olisi pitänyt luottaa enemmän siihen, että ihmiset olivat perehtyneet tekstiini etukäteen ja jättää työni esittely vähemmälle, jolloin keskustelulle olisi jäänyt enemmän aikaa. Palatessani symposiumista tunsin silti saaneeni uutta pontta projektiini.

Huhtikuussa 2012 Vertailin eri virsikirjaeditioita ja valitsin Hanna Koskisen nuottikuvat Veisuuvekottimeen talletettavien virsien lähteeksi. Aloin käymään nuottikuvia läpi systemaattisesti selvittääkseni, millaisia merkintöjä niissä on. Selitin Martin Pärtelille, että minun pitäisi kirjoittaa jäsennin, joka luo tekstipohjaisesta esityksestä oliomallin. Kysyin, millaista lähestymistapaa hän suosittelisi. Pärtel kyseli hieman kielestäni, suositteli säännöllisten lausekkeiden käyttämistä ja selitti lyhyesti säännöllisten lausekkeiden piirteen, jonka avulla merkkijonoista voi poimia haluttuja osia muuttujien arvoiksi. Toukokuussa 2012 tutkin Koskisen nuottikuvia.

Sekä Tomcat-palvelimen että MySQL-tietokannanhallintajärjestelmän pystyttäminen oli minulle melko työlästä. Olin käyttänyt kumpaakin tietojenkäsittelytieteen opinnoissani, mutta minun ei ollut tarvinut koskaan asentaa näitä järjestelmiä. Verkosta löytyvien ohjeiden seuraaminen osoittautui hankalaksi varsinkin Tomcatin tapauksessa. Pääsin Tomcatin kanssa eteenpäin hankittuani aiheesta painetun kirjan [VG11]. Myös MySQL:n hallintaa

opiskelin kirjasta [DuB09], vaikka verkkoaineistoa olisi ollut saatavilla.

Ensimmäisen projektivuoden (lukuvuosi 2011–2012) lopulla verkkopalveluni oli Java-kielinen ja verkkosivujen tuottaminen tapahtui Tomcat-palvelimen päällä ajettavien Java-servlettien avulla. Lauludata oli MySQL-relaatiotietokannassa ja Java-ohjelmani luki ja kirjoitti tietokantaa. Virsikoodi oli pitkälti määritelty ja olin päättänyt kirjoittaa JavaScriptillä oliomallin virsikoodin käsittelemiseen käyttäjän verkkoselaimessa, mutta en ollut vielä aloittanut oliomallin toteuttamista. Olin kuitenkin jo kirjoittanut JavaScriptillä jäsennintä virsikoodin tulkitsemiseen. Kirjoitin projektipäiväkirjaani, ettei minun kannattaisi huolehtia liikaa siitä, onko Java paras kieli Veisuuvekottimen palvelinpuolen toteuttamiseen. Huonokin päätös olisi parempi kuin ei päätöstä ollenkaan.

Toisen projektivuoden (lukuvuoden 2012–2013) aloitin kirjoittamalla JavaScriptillä oliomallia lauludatan esittämiseksi. Tämän jälkeen vaihdoin palvelimen puolella ajettavien ohjelmien kielen Javasta PHP:ksi ja luovuin samalla Java-kieleen liittyneen Tomcat-palvelimen käyttämisestä. Minua oli pitkään vaivannut epäily, että Java ei välttämättä olisi järkevä valinta palvelinpuolen toteutuskieleksi, vaikka osasinkin sitä entuudestaan toisin kuin PHP:tä. Valinnan ratkaisivat lähinnä taloudelliset kysymykset: halusin, että yksityishenkilö (kuten minä itse) voisi perustaa Veisuuvekottimen avulla verkkopalvelun webhotelliin, vaikkei saisi webhotellin laskuille maksajaa ja joutuisi näin ollen maksamaan laskut itse. Veisuuvekottimen piti siis toimia myös edullisissa webhotelleissa, ja pienen selvityksen pohjalta vaikutti siltä, että halvimmat PHP:tä tukevat webhotellit ovat halvempia kuin Javaa tukevat.

Syyskuun 2012 puolen välin paikkeilla olin saanut toteutettua yksinkertaisen laulukirjasovelluksen MySQL-tietokannanhallintajärjestelmän ja PHP- ja JavaScript-kielten avulla. Lokakuussa 2012 aloin tutustumaan JavaScript-ohjelmointia helpottavaan jQuery-kirjastoon. Marraskuusta vietin kolme viikkoa työharjoittelussa, jona aikana projektini ei juurikaan edennyt.

Kevätlukukauden 2013 aloitin kokeilemalla AJAX- ja JSON-tekniikoiden soveltamista verkkopalveluuni. Palveluni arkkitehtuuri alkoi kehittymään kohti yhden verkkosivun mallia, jossa käyttäjän toimenpiteet eivät johda kokonaan uuden verkkosivun lataamiseen palvelimelta, vaan aiemmin ladattua sivua muokataan.

Tammikuun 2013 lopulla aloin jättämään palvelusta pois ominaisuuksia. Tammi–helmikuun vaihteessa 2013 aloin muokkaamaan JavaScriptillä kirjoittamaani käyttöliittymäkoodia olio- ja MVC-pohjaiseksi. Koodi oli muodostunut jo niin monimutkaiseksi, että se piti jäsennellä uudelleen, jotta sitä voisi helpommin hallita. Helmikuussa päätin jättää pois toiminnon, jonka avul-

la voitaisiin luoda laululistoja antamalla palvelulle kerralla useiden listalle haluttujen laulujen numerot.

Maaliskuussa 2013 keksin palvelulle nimeksi Veisuuvekotin, tilasin webhotellin `nettihotellit.fi`-palvelusta ja rekisteröin domainnimen `veisuuvekotin.net`. Huhtikuussa 2013 siirsin Veisuuvekottimen tietokannan nettihotelliin. Samassa kuussa hain abcjs-kirjaston lähdekoodin omalle tietokoneelleni tarkoitukseni tutustua koodiin lähemmin. Tarkoitukseni oli muun muassa tutkia, onko koodissa tietoturvan kannalta ongelmallisia piirteitä. Jouduin luopumaan abcjs:n koodin lähemmästä tarkastelusta aikataulusyistä.

Toukokuussa 2013 tutkin virsikirjan melodiodien alkuperämerkintöjä. Totesin nämä niin monimutkaisiksi, että niiden rakenteellistaminen jäi projektin ulkopuolelle. Niinpä luovuin säveltäjä- ja sävellysvuosikentistä ja korvasin ne sävelmän alkuperää kuvaavalla kentällä. Muokkasin käyttöliittymän kolmisarakkeisesta kaksisarakkeiseksi ja poistin otsikoita.

Kesä- ja heinäkuun 2013 vietin kausikanttorina Espoonlahden seurakunnassa. Heinä- elokuun vaihteessa muutin perheineni Mikkeliin, jossa minulla oli alkamassa tilapäinen kanttorin virkasuhde syyskuun alussa. Elokuussa 2013 järjestin kotia, kävin autokoulua, koska työni Mikkeliissä vaati oman auton käyttöä sekä viimeistelin projektia jättökuntoon. Tässä vaiheessa keskityin projektin kirjallisen osuuden viimeistelyyn. Käytännöllistä osuutta viimeistelin poistamalla tietokannasta tekijänoikeussuojattua aineistoa ja poistamalla tietokannan tauluista kenttiä, joita Veisuuvekotin ei enää tai vielä käyttänyt.

Luku 8

Loppukatsaus

Veisuuvekotin-projektissa havaittiin, että valittuja tekniikoita käyttäen on mahdollista toteuttaa verkkopalvelu, joka tuottaa käyttäjälle virsien nuot tikuvia dynaamisesti. Projektin aikataulu ei tosin mahdollistanut Veisuuvekottimen kunnollista testaamista, joten epäselväksi jäi, kuinka hyvin palvelu teknisesti toimii ja kuinka miellyttäväksi käyttäjät sen kokevat. Myöskään tietoturvaan ei ehditty kovin perusteellisesti paneutua. Lähinnä näistä syistä johtuen `veisuuvekotin.net`-palveluun pääsyä rajoitetaan toistaiseksi käyttäjätunnuksilla, joita annetaan vain harvoille.

Vaikka palveluun suunnitelluista ominaisuuksista merkittävä osa jätettiin lopulta pois, osoittautui palvelun toteuttaminen ja dokumentoiminen melko isoksi urakaksi yhdelle henkilölle kahden vuoden aikana tehtäväksi. Kysymys, jota jo projektin alussa pohdittiin, jäi vaille vastausta: olisiko vastaava palvelu mahdollista toteuttaa siten, että hyödynnettäisiin enemmän valmista ja tehtäisiin vähemmän itse?

Projektissa havaittiin, että näinkin laajan järjestelmän toteuttamisessa on syytä käyttää välineitä, joiden avulla järjestelmää jäsennetään. Tällaisiksi välineiksi projektissa otettiin olioperustainen ohjelmistosuunnittelu sekä observer- ja MVC -suunnittelumallit. Myös tietokannanhallintajärjestelmän käyttäminen alkoi projektin edetessä vaikuttaa yhä paremmalta ratkaisulta.

Veisuuvekottimeen liittyy useita aihealueita, joiden suhteen järjestelmää voidaan jatkokehittää. Järjestelmä voisi tarjota kehittyneemmät ominaisuudet laulukirjojen, laululistojen, soinnutusten ja sävellajien luokitteluun ja valintaan. Laululistojen luomiseen voisi olla tehokkaammat toiminnot. Järjestelmän yhteensopivuutta muiden musiikkiohjelmien kanssa voitaisiin parantaa esimerkiksi lisäämällä ominaisuus, jonka avulla aineisto saataisiin ladattua palvelusta MusicXML-muodossa. Palvelusta tulostamista voitaisiin helpottaa toiminnolla, joka tulostaisi laululistan kaikki laulut yhdellä komennolla. Nuotinluvun avuksi voitaisiin lisätä otediagrammien ja tabulatuu-

rien tuottaminen. Neliäänisten satsien tuki tekisi palvelusta kiinnostavamman klassisesti orientoituneiden muusikoiden kannalta.

Jo ennen projektin alkua ja edelleen projektin aikana hankkeesta saatu palaute antaa ymmärtää, että jollekin Veisuuekottimen kaltaiselle olisi tilausta. On mahdollista, että tässä projektissa toteutettua järjestelmää jatkokehitetään todelliseen käyttöön soveltuvaksi palveluksi asti. Koska Veisuuekottimen ohjelmakoodi on lisensoitu GPL v. 3 -lisenssin alaisuudessa, kuka tahansa voi kehittää järjestelmää edelleen. On myös mahdollista, että tämä projekti toimii jollekulle innoittajana toteuttaa jotakin vastaavaa käyttämällä osittain tai kokonaan eri tekniikoita.

Lähteet

- [abca] The abc music standard 2.1. <http://abcnotation.com/wiki/abc:standard:v2.1>, viitattu 15.8.2013.
- [abcb] abcjs – ABC parser, renderer and editor in javascript. <http://code.google.com/p/abcjs/>.
- [App] Apple Inc. iTunes. <http://www.apple.com/fi/itunes/what-is/>, viitattu 6.3.2013.
- [DuB09] Paul DuBois. *MySQL*. Addison-Wesley, 4. edition, 2009.
- [dyn] Dynamic web page. http://en.wikipedia.org/wiki/Dynamic_web_page, viitattu 9.8.2013.
- [EKU09] Katarina Engström, Jani Kairavuo, and Jenni Urponen. *Virsilöytöretki*. LK-kirjat – Nuorten keskus, Helsinki, 2009.
- [fla] Adobe flash player 11. <http://www.adobe.com/fi/products/flashplayer.html>.
- [Fla11] David Flanagan. *JavaScript: The Definitive Guide*. O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 6. edition, 2011.
- [Fou] Free Software Foundation. The GNU general public license v3.0. <http://www.gnu.org/licenses/gpl.html>.
- [Graa] Gracevoice Oy. Suomalainen eVirsiKirja — löydä virret uudelleen. <http://www.evirsikirja.fi/>, viitattu 14.8.2013.
- [Grab] Gracevoice Oy. Suomalainen raamattukauppa. <http://www.raamattukauppa.fi/>, viitattu 14.8.2013.
- [HEt02] Hannu Hurme, Katarina Engström, and Mika Mäntyranta (toim.). *Kitaravirsiä*. LK-KIRJAT, Lasten Keskus, Helsinki, 2002.

- [ima] Comparison of web browsers. http://en.wikipedia.org/wiki/Comparison_of_web_browsers#Image_format_support.
- [kor02] *Koraalikirja*. F-Kustannus Oy, Helsinki, 2002. Ensimmäinen julkaisu: Fazer Musiikki Oy 1987.
- [Kos00] Kai Koskimies. *Oliokirja*. Satku, 2000.
- [Kur12] Budi Kurniawan. *Servlet and JSP : A Tutorial*. BrainySoftware.com, 1. edition, 2012.
- [Ky] Kotimaa-yhtiöt. Virret.net – löydä hakemasti virsi helposti. <http://www.virret.net/>, viitattu 14.8.2013.
- [Las04] Kai Lassfolk. *Music notation as objects : : an object-oriented analysis of the common western music notation system*. University of Helsinki, 2004.
- [leg] Legato sheet music viewer. <http://www.musicrain.com/products-sheetmusicviewer.php>.
- [LHL⁺92] Heikki Lokki, Ilkka Haikala, Seppo Linnainmaa, Kari Rönkä, and Outi Susiluoto. *Tietotekniikka*. Tietotekniikan liitto ry, 1992.
- [lila] Lilypond notation reference: Absolute octave entry. <http://www.lilypond.org/doc/v2.16/Documentation/notation/writing-pitches#absolute-octave-entry>, viitattu 14.8.2013.
- [lilb] Lilypond notation reference: Beams. <http://www.lilypond.org/doc/v2.16/Documentation/notation/beams>, viitattu 15.8.2013.
- [lilc] Lilypond...music notation for everyone. <http://www.lilypond.org/>, viitattu 14.8.2013.
- [Mica] Microsoft. Model-View-Controller. <http://msdn.microsoft.com/en-us/library/ff649643.aspx>, viitattu 22.8.2013.
- [Micb] Microsoft. Observer. <http://msdn.microsoft.com/en-us/library/ff649896.aspx>, viitattu 22.8.2013.
- [mit] The MIT license (MIT). <http://opensource.org/licenses/mit-license.php>.
- [musa] MusicXML for exchanging digital sheet music. <http://www.musicxml.com/>.

- [musb] MusiXTeX. <http://en.wikipedia.org/wiki/MusiXTeX>.
- [Musc] MusicEase Software. Musicease software products. <http://www.musicease.com/products.htm>, viitattu 14.8.2013.
- [Nik] Niko Antin. Kysy Macmaalta: miten katson Flash-sisältöä iPadilla? <http://macmaa.com/2012/05/29/kysy-macmaalta-miten-katson-flash-sisalto-a-ipadilla/>, viitattu 21.8.2013.
- [nota] Noteflight. <http://www.noteflight.com/login>.
- [notb] Noteflight embed API developers' guide. http://www.noteflight.com/info/api/client_doc.
- [peta] IMSLP:File formats. http://imslp.org/wiki/IMSLP:File_formats, viitattu 22.2.2013.
- [petb] IMSLP/Petrucci music library: Free public domain sheet music. <http://imslp.org/>.
- [Pit] Teija Pitkänen. Kirkkovuoden 2009–2010 jumalanpalvelusten virret. Esitys virsitutkimushankkeen symposiumissa Vaiviossa 23.3.2012.
- [PN02] Pekka and Ilona Nyman. *Soittajan virsikirja*. Kirkkopalvelut, 2002.
- [PT04] Reijo Pajamo and Erkki Tuppurainen. *Kirkkomusiikki (osa sarjaa Suomen musiikin historia)*. WSOY, 2004.
- [rap] Raphaël-javascript library. <http://raphaeljs.com/>.
- [rkr] Suomen rauhanyhdistysten keskusyhdistys ry. Srk-verkkokauppa. <http://kauppa.srk.fi/>, viitattu 28.2.2013.
- [run] Running scorch plug-in in a 64-bit capable browser. http://avid.force.com/pkb/articles/en_US/FAQ/en362435, viitattu 22.2.2013.
- [scoa] Scorch. <http://www.sibelius.com/products/scorch/index.html>, viitattu 9.8.2013.
- [scob] Score exchange. <http://www.scoreexchange.com/>.

- [scoc] Sibelius scorch plugin has been blocked for your protection. <https://addons.mozilla.org/fi/firefox/blocked/p248>, viitattu 22.2.2013.
- [Siba] Sibelius-Akatemia. Suomen vanhat virsisävelmät. <http://www2.siba.fi/virtuaalikatedraali/vanhatvirret/index.html>, viitattu 14.8.2013.
- [Sibb] Sibelius-Akatemia. Virtuaalikatedraali. <http://www2.siba.fi/virtuaalikatedraali/>, viitattu 14.8.2013.
- [soi07] *Soinnutettu kirkkovirssikirja*. Kotimaa-Yhtiöt Oy, Kirjapaja, 2007.
- [Spo] Spotify Ltd. Music for every moment — Spotify. <https://www.spotify.com/>, viitattu 6.3.2013.
- [sta] Static web page. http://en.wikipedia.org/wiki/Static_web_page, viitattu 9.8.2013.
- [Tec] Technimo LLC. ireal b. <http://www.irealb.com/>, viitattu 14.8.2013.
- [Thea] The jQuery Foundation. jquery. <http://jquery.com/>, viitattu 19.8.2013.
- [Theb] The jQuery Foundation. jQuery UI. <http://jqueryui.com/>, viitattu 19.8.2013.
- [vec] Spotting the difference – vector and raster PDF. <http://visual-integrity.com/support/faq/spotting-difference-vector-raster-pdf/>, viitattu 9.8.2013.
- [ver] Verkkovirssikirja. <http://evl.fi/Virssikirja.nsf>, viitattu 14.8.2013.
- [vex] VexFlow – HTML5 music engraving. <http://www.vexflow.com/>.
- [VG11] Aleksa Vukotic and James Goodwill. *Apache Tomcat 7*. Apress, 2011.
- [vir] Peruskoulun virsisuosituslista. <http://www.evl.fi/kkh/to/KJM/virsiaineistoa/kouluvirret.html>, viitattu 14.8.2013.
- [vir87] *Virssikirja*. Kustannus-osakeyhtiö Kotimaa, Kirjapaja, 1987.
- [vir04] *Virssikirja*. Raamattutalo, 2004.

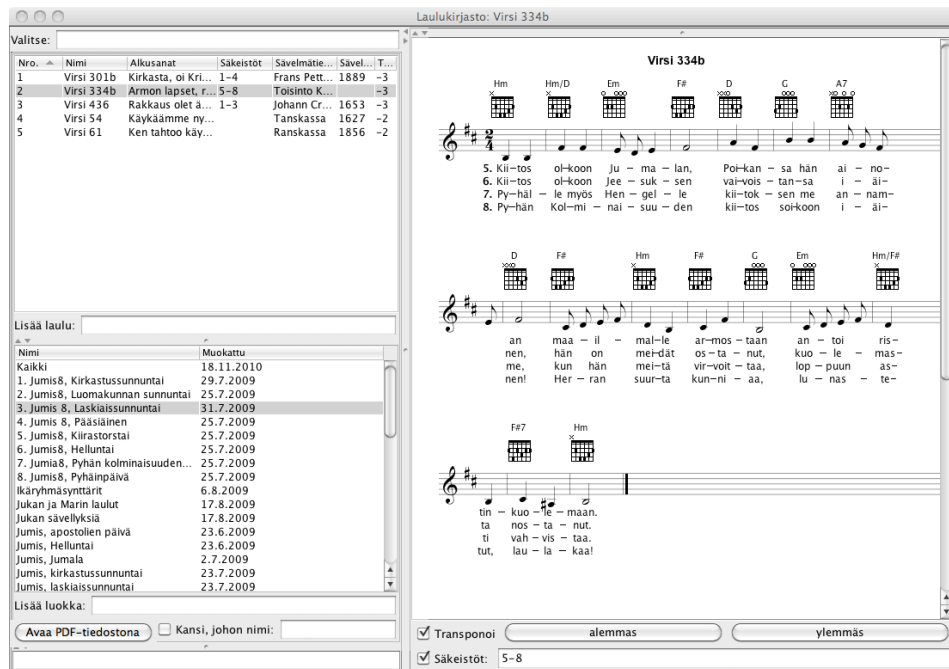
- [VLRN05] Marjasisko Varha-Lankinen, Marianne Ropponen, and Pekka Nyman. *Virsién kirja*. Kirkkopalvelut, Raamattutalo, 2005.
- [Wal] Chris Walshaw. abc notation home page. <http://abcnotation.com/>, viitattu 14.8.2013.

Liite A

Laulukirjastoeseite

Laulukirjasto – oiva apu laulattajille

Laulukirjasto on älykkäänä laulukirjana toimiva tietokoneohjelma, joka on suunnattu erityisesti kanttoreille, kirkon nuorisotyön ohjaajille sekä muille, jotka säestävät virsiä ja muita lauluja esimerkiksi kosketinsoittimilla tai kitaralla tai ohjaavat tällaista toimintaa. Ohjelmaa kutsutaan Laulukirjastoksi, koska ohjelmaan voidaan tallettaa useiden laulukirjojen materiaali. Kyseessä ei ole valmis tuote, ohjelmasta on olemassa vasta osittain toimiva prototyyppi. Tässä dokumentissa kuvataan tärkeimmät Laulukirjastoon suunnitellut ominaisuudet.



Ohjelman prototyyppi näyttää tällä hetkellä (19.1.2011) tältä. Ohjelman piirtämässä nuottikuvassa on vielä puutteita, esimerkiksi rivinvaihdot tapahtuvat milloin tahansa, nuottivarret ovat aina ylöspäin eikä nuottien palkitus ole käytössä.

Laulukirjasto-ohjelma tarjoaa useita etuja painettuihin laulukirjoihin verrattuna:

- Läppäriin kovalevylle tai muistitikulle mahtuu valtava määrä nuotteja. Lukuisien laulukirjojen anti saadaan näin ikään kuin yhteen kirjaan, pieneen tilaan, kevyesti kannettavaksi.
- Etsittäessä laulua ei tarvitse selata useita erillisiä laulukirjoja, vaikkei muistaisikaan, mistä kirjasta laulu löytyy. Ohjelman hakutoiminnot etsivät laulua kaikista tallennetuista kirjoista.
- Ohjelma osaa transponoida laulut mille tahansa korkeudelle.
- Ohjelmaan voidaan tallettaa kuhunkin lauluun niin monta vaihtoehtoista reaalisointumerkein kirjoitettua soinnutusta tai neliäänistä satsia kuin halutaan. ¹

¹ On julkaistu useita kirkoja, jotka tarjoavat toisistaan poikkeavia reaalisointumerkein kirjoitettuja soinnutuksia virsikirjan virsiin: on Soinnutettu virsikirja, Soittajan virsikirja, Kitaravirsiä-kirja.... Jos kaikkien näiden kirjojen soinnutukset talletetaan Laulukirjastoon, voi käyttäjä etsimänsä virren löydettyään kätevästi valita soinnutuksen, joka sopii parhaiten käyttötilanteeseen: haluttuun soittotyyliin, instrumenttivalintoihin ja soittajien taitotasoon.

- Ohjelma lataa haluttujen säkeistöjen sanat melodian alle.
- Reaalisointumerkkien alle saadaan haluttaessa diagrammit, jotka kertovat, miten soinnut otetaan kitarasta.
- Kitaristeja varten laulukirjasto antaa mahdollisuuden valita, kirjoitetaanko esim. Ab-duurissa kulkevan laulun soinnut normaaliin tapaan Ab-duuriin, vai merkitäänkö nuotin alkuun "kapo 1" ja kirjoitetaan soinnut G-duuriin.²
- Neliäänisen satsin tapauksessa ohjelma osaa haluttaessa kirjoittaa bassostemman omalle viivastolleen urkureita varten. Käyttäjä saa lisäksi päättää, kirjoitetaanko alttostemma samalle viivastolle sopraanon vai tenorin kanssa. Alttostemman kirjoittaminen samalle viivastolle tenorin kanssa tukee triosoittoa. Käyttäjä saa valita, minkä viivastojen väliin tekstit kirjoitetaan. Näin tekstit voidaan sijoittaa lähelle niitä viivastoja, joita käyttäjä ensisijaisesti lukee.
- Lauluja voi hakea alkusanojen, laulun nimen, säveltäjän, sanoittajan, laulussa esiintyvän tekstin tai muun lauluun hakuja varten liitetyn tiedon perusteella. Hakutulokset päivittyvät sitä mukaa kun käyttäjä kirjoittaa hakuetta. Lauluun voidaan liittää seuraavaa tietoa:
 - Mihin kirkkovuoden pyhään laulu liittyy?
 - Mihin virsikirjan osastoon laulu teemansa puolesta kuuluisi, jos se olisi virsikirjassa? ³
 - Mitä ovat laulun tematiikan avainkäsitteet?
 - Mihin Raamatunkohtaan laulu liittyy?
- Lauluista voi muodostaa listoja, joita on helppo selata. Esim. tulevan messun virret ja mahdolliset muut laulut voi kerätä laululistaan oikeassa järjestyksessä, sopiville korkeuksille transponoituna, varustettuna valituilla soinnutuksilla tai neliäänisillä satseilla ja sillä kertaa laulettavien säkeistöjen sanoilla. Listalle kerättyjen laulujen nuotit voi halutessaan tulostaa, ellei halua soittaa suoraan tietokoneen näytöltä. Listoista on hyötyä myös, jos nuotit heijastetaan datatykillä seurakunnalle.
- Ohjelma osaa soittaa laulun soinnutuksineen näyttäen koko ajan nuotista, missä ollaan menossa.
- Laulun nuotti voidaan helposti kopioida tekstinkäsittelyohjelmaan vaikkapa käsiohjelmaan liitettäväksi.
- Laulu voidaan tallettaa MIDI-tiedostona, joka voidaan sitten avata jollakin nuotinnusohjelmalla (Sibelius, Finale, ...) jatkokäsittelyä varten.
- Käyttäjä voi kirjoittaa lauluihin myös omia soinnutuksiaan, tai lisätä Laulukirjastoon kokonaan uusia lauluja.
 - Omat soinnutukset ja muu ohjelmaan lisätty aineisto voidaan jakaa netin välityksellä muiden Laulukirjasto-käyttäjien kanssa.
- Ohjelman näkemät aineistot voivat sijaita käyttäjän omalla tietokoneella tai verkkopalvelimilla. Ohjelma tarjoaa mahdollisuuden imuroida palvelimella tarjottu aineisto käyttäjän omalle koneelle, jolloin se on käytettävissä myös ilman Internet-yhteyttä.

² Kitaristit käyttävät toisinaan kapotastoa, joka on kitaran otelaudalle kiinnitettävä pieni laite, joka nostaa kitaran sävelkorkeutta. Esimerkiksi jos kapotasto asetetaan 1. nauhalle kitaran kaulalla, soi kitara Ab-duurista, kun soitetaan helpommilla G-duurin sointuotteilla. Toisinaan lauluja, jotka ovat kitaristien kannalta hankalassa sävellajissa, kirjoitetaan siten, että merkitään nuotin alkuun, mille nauhalle kapotasto asetetaan, ja sitten kirjoitetaan sointuotteiden mukaiset reaalisoinnut.

³ Erityisesti pitkään virassa olleet kanttorit ovat oletettavasti tottuneet ajattelemaan virsikirjan kategorioiden kautta.

Ohjelman rajoitteista

Ohjelmaan voidaan tallettaa vain yksinkertaisia laulunuootteja, joissa on melodia, sanat ja reaalisoinnut tai neliääninen satsi. Ohjelmaan ei siis voida tallettaa esimerkiksi kokonaista orkesteripartituuria.

Ohjelman toimintaperiaatteesta

Ohjelmaan ei talleteta valmiita nuottikuvia, vaan informaatiota, josta ohjelma osaa tuottaa erilaisia, kuhunkin tilanteeseen sopivia nuottikuvia.

Suurin haaste: lauluaineistojen tekijänoikeudet

Toistaiseksi on ratkaisematta, kuinka ohjelman mukana voitaisiin julkaista tekijänoikeuksin suojattua lauluaineistoa. Varsinainen ongelma ei ole, kuinka esimerkiksi Virsikirja eri opuksissa julkaistuine soinnutuksineen, Nuoren seurakunnan veisukirjat, Tuomaslaulukirjat, Siionin virret jne. saataisiin ohjelman ymmärtämään muotoon. Ongelmana on, kuinka tätä aineistoa voitaisiin jaella siten, että tekijänoikeuksien haltijat saisivat heille kuuluvan korvauksen. Pitäisikö Laulukirjaston ymmärtämään muotoon käännettyjä laulukirjoja ja yksittäisiä lauluja myydä netin välityksellä samaan tapaan kuin esim. iTunes Storessa myydään mp3-muotoista musiikkia? Ongelmana on myös, että kaikki tekijänoikeuksien haltijat eivät välttämättä anna lupaa aineiston sähköiseen jakeluun, vaikka korvaukset maksettaisiin.

Ohjelman toteutustekniikasta ja laitteistovaatimuksista

Ohjelma on toteutettu Java-kielellä, joten se toimii esimerkiksi Windows-, OS X- ja Linux-tietokoneissa. Ohjelma siis toimii sekä PC- että Mac-tietokoneissa.

Ohjelman toteuttajasta

Laulukirjaston on suunnitellut ja toteuttanut Jukka Stenlund. Jukka opiskelee kirkkomusiikkia Sibelius-Akatemiassa. Aiemmin hän opiskeli tietojenkäsittelytiedettä Helsingin yliopistossa, ja hän on myös vetänyt harjoitusryhmiä samaisen yliopiston ohjelmointikursseilla. Jukka soittaa itse lähinnä kosketinsoittimia ja kitaraa, mutta on tutustunut hieman myös bassokitaraan, rumpujen ja perkussoiden soittoon. Kanttorin työhön Jukka on saanut kosketusta kesätöissä Kymen ja Savonlinna-Säämingin seurakunnissa, joissa hän on toiminut erityisesti rippikouluissa.

Jukan tavoittaa sähköpostitse osoitteesta jukka.stenlund@siba.fi tai puhelimella numerosta 040 4158629. Jukka asuu Helsingin Kannelmäessä vaimonsa ja marraskuussa 2009 syntyneen tyttärensä kanssa.

Liite B

Kysely sähköisen virsikirjan potentiaalisille käyttäjille

Kysely sähköisen virsikirjan potentiaalisille käyttäjille

Jukka Stenlund 11.11.2011

Tämän kyselyn tarkoituksena on kartoittaa, mitä ominaisuuksia sähköiseen virsikirjaan kannattaisi toteuttaa, jotta se palvelisi henkilöitä, jotka ovat tekemisissä virsien kanssa, mutta eivät ole ammattimuusikkoja. Kyselyssä alkupuolella selvitetään vastaajan musiikillista taustaa.

Vastaa ympäröimällä oikea vaihtoehto/oikeat vaihtoehdot.

Vastaajan musiikilliseen taustaan liittyvät kysymykset

1. Osaatko laulaa itsellesi ennestään tuntemattoman virren nuotista ilman soittimen apua tutkittuasi nuottia hetken? kyllä en
2. Soitatko joitakin soittimia? kyllä en Mitä soittimia? _____
3. Osaatko soittaa itsellesi ennestään tuntemattoman virren melodian nuotista jollakin soittimella? kyllä en
4. Mitkä seuraavista soinnuista osaat ottaa kitarasta (ympyröi osaamasi soinnut):
E Am C G D F Cm Eb D9
5. Osaisitko soittaa kitaralla seuraavan tabulatuurin: kyllä en

5. Mitkä seuraavista soinnuista löydät pianosta:
C G Eb Dm Fm Gsus4 Em7-5 A13

6. Millä sanoilla alkavat seuraavat virret:



Alkusanat: _____



Alkusanat: _____



Alkusanat: _____

7. Yhdistä viivalla vasemman sarakkeen säveltäjät/artistit oikean sarakkeen teoksiin/tyylilajeihin/soittimiin

Dietrich Buxtehude	jazz
Jimi Hendrix	urut
Arnold Schönberg	12-sävelisyys
Antonio Vivaldi	latin
Ringo Starr	Neljä vuodenaikaa
Charlie Parker	kitara
Antônio Carlos Jobim	rummut

8. Mitkä seuraavista aikakausista ovat tuottaneet musiikkia, jonka koet itsellesi merkitykselliseksi?

Ympyröi itsellesi tärkeimmät ajanjaksot.

1500-luku ja sitä vanhemmat ajat

1600-luku 1700-luku 1800-luku 1900-luku kokonaisuudessaan

1910-luku 1920-luku 1930-luku 1940-luku 1950-luku 1960-luku 1970-luku 1980-luku

1990-luku 2000-luku 2010-luku

9. Mitä seuraavista arvelet tekeväsi tulevaisuudessa:

- valitset virsiä tilaisuuksiin yhdessä kanttorin kanssa
- valitset virsiä tilaisuuksiin itsenäisesti
- toimit esilaulajana, kun virsiä lauletaan yhdessä
- säestät virsilaulantaa
 - Millä soittimilla? _____

Sähköisen virsikirjan ominaisuuksien tarpeellisuus vastaajan kannalta

Mitkä seuraavista kriteereistä sähköisen virsikirjan pitäisi täyttää, jotta arvelisit itse käyttäväsi sitä?

Ympyröi pallukat näiden ominaisuuksien vasemmalta puolelta.

Laite- ja ohjelmistoalusta:

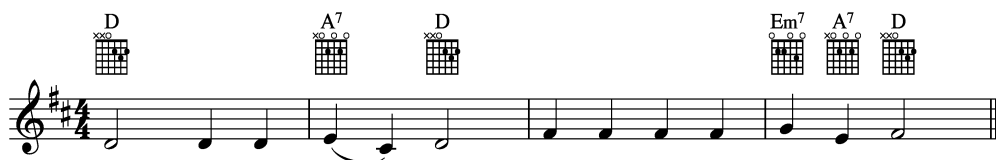
- Käytettävissä nettipalveluna
- Käytettävissä ilman Internet-yhteyttä
- Käytettävissä perinteisellä pöytä- tai sylimikrolla
- Käytettävissä sormitietokoneella
- Käytettävissä kännykällä
- Käytettävissä erityisillä sähköisen kirjan lukulaitteilla (engl. eBook reader)

Sisältö:

- Sisältää kaikki virsikirjan virret
- Sisältää melodiat
- Sisältää sanoitukset
- Sisältää sointumerkein kirjoitetut soinnutukset
- Sisältää neliääniset satsit (altto-, tenori- ja bassoääni melodian lisäksi)
- Sisältää kuunneltavia levytyksiä virsistä
- Käyttäjä voi itse lisätä virsiin
 - omia soinnutuksia
 - omia neliäänisiä satseja
- Käyttäjä voi jakaa itse kirjoittamansa sovitukset muiden käyttäjien kanssa

Nuottikuvien räätälöinti

- Osaa varustaa sointumerkit kaavioilla, jotka kertovat, miten soinnut otetaan kitarasta (katso kuva alla)



- Osaa latoa neliääniset satsit eri tavoin: kahdelle tai kolmelle viivastolle, erillisinä

stemmoina, ...

- Osaa transponoida virren halutulle korkeudelle
- Antaa käyttäjän valita, minkä säkeistöjen sanat otetaan mukaan

Mistä nuottikuvat luetaan?

- Tarjoaa nuotit helposti tulostettavassa muodossa
- Tarjoaa nuotit helposti ruudulta luettavassa muodossa

Virsiä kuuntelu tietokoneen tuottamana äänenä

- Osaa soittaa virren melodian
- Osaa soittaa virrestä melodian lisäksi soinnut
- Osaa soittaa virrestä melodian lisäksi muut stemmat (altto, tenori, basso)

Virren hakeminen

- Hakuperusteena voidaan virren numeron lisäksi käyttää seuraavia (voit ympyröidä yksittäisiä hakuehtoja):
 - alkusanat
 - katkelma mistä tahansa kohtaa tekstiä
 - melodiakatkelma (hakuehto annettaisiin kirjoittamalla sävelten nimet)
 - säveltäjä
 - sanoittaja
 - sävellys- tai sanoitusvuosi

Virsiä kokoaminen listaksi

- Virsistä voi muodostaa järjestettyjä listoja, jotka muistavat, millaisiksi nuottikuvat räätälöitiin
- Listoihin voi liittää tietoa siitä, millaiseen käyttöön lista luotiin (mille kirkkovuoden pyhälle tms.)

Lisäkysymyksiä

Minkä painettujen kirjojen virsisovitukset haluaisit nähdä sähköisessä virsikirjassa?

- Soinnutettu virsikirja
- Koraalikirja
- Hannu Hurme, Katarina Engström ja Mika Mäntyranta (toim.): Kitaravirsiä, LK-KIRJAT, Lasten Keskus, Helsinki 2002. Teos sisältää 140 virttä varustettuna kitarasäestyskäyttöön tarkoitetuilla sointumerkinnöillä.
- Pekka ja Ilona Nyman: Soittajan virsikirja, Kirkkopalvelut. Kirjaan on valittu 70 virttä, ja virsiin on sointumerkkien lisäksi kirjoitettu kuorosatseja ja säestysideoita bändisoittimille.
- Marjasisko Varha-Lankinen, Marianne Ropponen ja Pekka Nyman: Virsiä Kirja,

Kirkkopalvelut, RAAMATTUTALO. Kirjaan on valittu 46 virttä, joihin on soinnutusten lisäksi liitetty muutamia yksinkertaisia kuoro- ja bändisovituksia sekä taustatietoja.

- Katarina Engström, Jani Kairavuo ja Jenni Urponen: Virsilöytöretki, LK-kirjat – Nuorten keskus. Kirja ”on täynnä tietoa, vinkkejä, tehtäviä ja virikkeitä yli 20 virsikirjamme virrestä”. Virsiä on kirjassa sovitettu eri tavoin, ja virsiin on kirjoitettu myös sointumerkit.

Kuvaile vapaasti, millainen olisi juuri sinun tarpeisiisi optimoitu sähköinen virsikirja. Voit käyttää apuna seuraavia kysymyksiä:

- Millaista aineistoa kirja sisältäisi?
- Missä muodossa? (tekstinä, nuottina, äänenä, valokuvina, videokuvana...)
- Miten sähköinen virsikirja osaisi käsitellä aineistoa?

Liite C

Esimerkkejä koodatuista virsistä

Virsi 2, Avaja porttis, ovesi

Virsi 2 on helppoa koodattavaa: ei monimutkaisia rytmejä, ei palkkeja, ei kertauksia, tahtiosoitus murtolukuna, tahtiviivat on.

```
k:1# 6/4 e4 g a | h2 e4 e2 dis4 | e2. ' g4 a h | /  
c2 h4 a2 g4 | fis2. ' fis4 g fis | e2 a4 a2 gis4 | /  
a2. ' fis4 g a | h2 h4 h2 gis4 | h2. ' d'4 g c | /  
a2 g4 g2 fis4 | g2. ' fis4 g a | h2 e4 /  
e2 dis4 | e2 ' fis4 g2 g4 | a2 a4 h a g | /  
fis2 ' h4 e2 a4 | a4 b g fis2 fis4 | e2.
```

Virsi 23, On ruusu Iisain juuren

Ei tahtiviivoja, alla breve-tahtiosoitus, kertausmaalit, päättyy brevikseen.

```
k:3b t:C| b2 b4 b c b b2 g ' as g4 f /  
es2 d [1 es1 :| [2 e2 r4 ] g f d es c b,2 r4 b4 /  
b b c' b b2 g ' as g4 f es2 d |es|
```

Virsi 24, Kas, yöllä paimenille

Tahtiviivat, kertausmaalit.

```
k:1b 2/2 d4 | g d g e | f4. e8 d2 ' | a4 b8-a /  
g4 fis | [1 g2 r4 :| [2 g1 ' | ] a2 b | c4 b a2 ' | fis4 g /  
a4 b | g f8-e d4 ' fis4 | g b8-a g4 fis | g2 r4
```

Virsi 26, Oi armon lähde autuas

Palkkeja, myös ylennettyjen nuottien yhteydessä. Kohotahti.

```
k:2# 3/4 a,4 | d2 a,4 | d2 e4 | fis2 e4 | d2 ' fis8-g | /  
a4 d' cis' | h4. a8 g8-fis | e-fis g4 h | a2 ' fis4 | /  
h2 g4 | a2 fis8-e | d4. e8 fis4 | e2 ' a4 | /  
g4 fis e | d4 cis h, | a,4. h,8 c4 | d2
```

Virsi 622, Minä vaivainen, vain mato, matkamies maan

Paljon palkkeja.

```
k:1# 2/4 g8-g | d8-e fis-fis | g-h h-a | g-fis e-g | /  
fis4 ' d8-d | g4 a8-h | c4 h8-g | a4 g | r4 :| /  
h8-a | h4 h8-fis | g8-h h-a | g-fis e-g | /  
fis4 ' d8-d | g4 a8-h | c4 h8-a | g4 a8-a | /  
h4 ' g8-g | d4 g8-a | h-a g-g- | fis4 e | r4
```

Liite D

Veisuuvekottimen tietokannan taulujen rakenne

Tässä liitteessä käydään taulukoiden muodossa läpi Veisuuvekottimen tietokannan relaatioiden eli taulujen rakenne pääpiirteittäin. Kenttien tyypit on ilmaistu termein, jotka ovat teknisessä mielessä epätasällisiä mutta jotka ovat teknisesti tarkkoja termejä helpommat ymmärtää lukijoille, joille tietokantojen tekniikka on vieras aihe. Kentän merkityksen yhteydessä on suluisa mainittu, mikäli kyseessä on viite johonkin toiseen tauluun. Viittaamisen kohteena on aina viitattavan taulun id-kenttä, joka yksilöi taulun rivin.

Tauluihin book, harmonisation, list, list_item, melody, rights, song, user ja verse liittyy taulukoissa mainittujen kenttien lisäksi aikaleimatyyppinen added-kenttä, joka kertoo, milloin rivi on lisätty tauluun.

Taulukko D.1: Book-aulun rakenne

Kenttä	Tyyppi	Merkitys
id	kokonaisluku	yksilöi kirjan
name	merkkijono	kirjan nimi

Taulukko D.2: Customisation-taulun rakenne

Kenttä	Tyyppi	Merkitys
user	kokonaisluku	käyttäjä, jonka oletusarvoista on kyse (viittaa user-tilaan)
song	kokonaisluku	laulu, jonka oletusarvoista on kyse (viittaa song-tilaan)
transpose_interval	kokonaisluku	transponoitava intervalli
transpose_interval_quality	kokonaisluku	transponoitavan intervallin laatu
verses	merkkijono	säkeistöt, jotka laulusta on valittu
harmonisation	kokonaisluku	lauluun valittu soinnutus (viittaa harmonisation-tilaan)

Taulukko D.3: Harmonisation-taulun rakenne

Kenttä	Tyyppi	Merkitys
id	kokonaisluku	yksilöi soinnutuksen
notation	merkkijono	soinnutus virsikoodina
melody	kokonaisluku	melodia, jonka soinnutuksesta on kyse (viittaa melody-tilaan)
description	merkkijono	lyhyt, soinnutusta kuvaava teksti
owner	kokonaisluku	soinnutuksen omistaja (viittaa user-tilaan)
public	joko N tai Y (No tai Yes)	N jos soinnutus on tarkoitettu näytettäväksi vain owner-kentän määrittelemälle käyttäjälle, Y jos soinnutuksen saa näyttää kaikille käyttäjille

Taulukko D.4: List-taulun rakenne

Kenttä	Tyyppi	Merkitys
id	kokonaisluku	yksilöi listan
user	kokonaisluku	käyttäjä, jonka listasta on kyse (viittaa user-tauluun)
name	merkkijono	listan nimi

Taulukko D.5: List_item -taulun rakenne

Kenttä	Tyyppi	Merkitys
int	kokonaisluku	yksilöi nimikkeen
order_no	kokonaisluku	nimikkeen järjestysnumero listalla
song	kokonaisluku	nimikkeen laulu (viittaa song-tauluun)
list	kokonaisluku	lista, johon tämä nimike kuuluu (viittaa list-tauluun)
verses	merkkijono	säkeistöt, jotka laulusta on valittu
transpose_interval	kokonaisluku	transponoitava intervalli
transpose_interval_quality	kokonaisluku	transponoitavan intervallin laatu
harmonisation	kokonaisluku	lauluun valittu soinnutus (viittaa harmonisation-tauluun)

Taulukko D.6: Melody-taulun rakenne

Kenttä	Tyyppi	Merkitys
id	kokonaisluku	yksilöi melodian
notation	merkkijono	melodia virsikoodina
origin	merkkijono	sävelmän alkuperä (esim. säveltäjä tai maa)

Taulukko D.7: Rights-taulun rakenne

Kenttä	Tyyppi	Merkitys
user	kokonaisluku	käyttäjä, jonka oikeuksista on kysymys (viittaa user-tauluun)
book	kokonaisluku	kirja, johon liittyvistä oikeuksista on kyse (viittaa book-tauluun)
add_song	joko N tai Y (No tai Yes)	Y, jos user-kentän määrittelemällä käyttäjällä on oikeus lisätä uusi laulu book-kentän määrittelemään kirjaan
add_harmonisation	joko N tai Y (No tai Yes)	Y, jos user-kentän määrittelemällä käyttäjällä on oikeus lisätä uusi soinnutus lauluun, joka kuuluu book-kentän määrittelemään kirjaan
edit_metadata	joko N tai Y (No tai Yes)	Y, jos user-kentän määrittelemällä käyttäjällä on oikeus muokata book-kentän määrittelemään kirjaan kuuluvien laulujen metatietoja (sävelmän alkuperä yms.)
edit_melody	joko N tai Y (No tai Yes)	Y, jos user-kentän määrittelemällä käyttäjällä on oikeus muokata book-kentän määrittelemään kirjaan kuuluvien laulujen melodioita
edit_any_harmonisation	joko N tai Y (No tai Yes)	Y, jos user-kentän määrittelemällä käyttäjällä on oikeus muokata mitä tahansa soinnutusta, joka liittyy book-kentän määrittelemään kirjaan kuuluvaan melodiaan
edit_own_harmonisation	joko N tai Y (No tai Yes)	Y, jos user-kentän määrittelemällä käyttäjällä on oikeus muokata omia soinnutuksiaan, jotka liittyvät book-kentän määrittelemään kirjaan kuuluvaan melodiaan
edit_lyrics	joko N tai Y (No tai Yes)	Y, jos user-kentän määrittelemällä käyttäjällä on oikeus muokata book-kentän määrittelemään kirjaan kuuluvien laulujen sanoituksia

Taulukko D.8: Song-aulun rakenne

Kenttä	Tyyppi	Merkitys
id	kokonaisluku	yksilöi laulun
number	kokonaisluku	laulun numero laulukirjassa
variant	yksi merkki	erottaa virsissä a- ja b-version toisistaan
book	kokonaisluku	kirja, johon laulu kuuluu (viittaa book-tauluun)
name	merkkijono	laulun nimi (ei käytetä virsien yhteydessä)
melody	kokonaisluku	viittaa melodiatauluun
information	merkkijono	Lauluun liittyvää tietoa. Virsikirjan tapauksessa tähän talletetaan tiedot, jotka virsikirjoissa painetaan tekstin jälkeen, eli muun muassa tekstin tekijätiedot.

Taulukko D.9: User-taulun rakenne

Kenttä	Tyyppi	Merkitys
id	kokonaisluku	yksilöi käyttäjän
username	merkkijono	käyttäjätunnus
password	merkkijono	salasana muunnettuna
first_name	merkkijono	etunimi
last_name	merkkijono	sukunimi
admin	joko N tai Y (No tai Yes)	Y, jos käyttäjä on järjestelmän ylläpitäjä, N muuten
wrong_passwd_after_last_login	kokonaisluku	väärällä salasanalla tehtyjen kirjautumisyritysten määrä edellisen onnistuneen kirjautumisen jälkeen
wrong_guesses_for_current_passwd	kokonaisluku	väärällä salasanalla tehtyjen kirjautumisyritysten määrä sen jälkeen, kun salasana on viimeksi vaihdettu
enabled	joko N tai Y (No tai Yes)	Y, jos tällä tunnuksella saa kirjautua sisään, N jos tunnus on suljettu

Taulukko D.10: Verse-taulun rakenne

Kenttä	Tyyppi	Merkitys
id	kokonaisluku	yksilöi säkeistön
number	kokonaisluku	säkeistön numero
text	merkkijono	säkeistön teksti
song	kokonaisluku	laulu, johon säkeistö kuuluu (viittaa song-tilaan)

Liite E

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

PREAMBLE

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain

responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component

(kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the

conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- (a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- (b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- (c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will

therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

- (d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation’s users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- (a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- (b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

- (c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- (d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- (e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product

from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal

in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- (a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- (b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- (c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- (d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- (e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- (f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate

that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license

to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
```

```
Copyright (C) <textyear> <name of author>
```

```
This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
```

```
This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.
```

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, your program’s

commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.