

Opinnäytetyö YAMK

Taiteen uudet kontekstit

2021

Atte Häkkinen

Se on vähän semmoinen oma saarekkeensa

– SuperCollider-ohjelmointikielen käyttö
suomalaisessa kokeellisessa musiikissa ja
äänitaiteessa



TURKU AMK

TURKU UNIVERSITY OF
APPLIED SCIENCES

Opinnäytetyö (YAMK) | tiivistelmä

Turun ammattikorkeakoulu

Taiteen uudet kontekstit

2021 | 40 sivua

Atte Häkkinen

Se on vähän semmoinen oma saarekkeensa

- SuperCollider-ohjelmointikielen käyttö suomalaisessa kokeellisessa musiikissa ja äänitaiteessa

Opinnäytetyössäni tutkin SuperCollider-ohjelmointikielen käyttöä suomalaisessa kokeellisessa musiikissa ja äänitaiteessa. Olen työssäni haastatellut neljää taiteilijaa, jotka käyttävät tai ovat käyttäneet SuperCollideria taiteellisessa työssään. Haastattelujen pohjalta olen koostanut opinnäytetyöhöni kolme teemaa, jotka ovat: SuperCollider omana saarekkeenaan, SuperColliderin käyttäjäkokemukset ja SuperColliderin tulevaisuus.

SuperCollider-kielen käyttö rajautuu havaintojeni mukaan lähinnä taidekonteksteihin. Ohjelmointikielellä on omat käyttäjäyhteisönsä, jotka sitä vaalivat, mutta kieli ei ole levinnyt kaupalliseen käyttöön. SuperColliderin yhdistäminen muihin ohjelmointikieliin ja esimerkiksi pelimoottoreihin on hankalaa ja tämä rajoittaa kielen leviämistä. Tästä huolimatta SuperColliderin voidaan katsoa edistäneen ihmisten urapolkuja ja kielellä on oma vakiintunut paikkansa kokeellisen musiikin ja äänitaiteen kentällä.

Asiasanat:

ohjelmointikielet, ohjelmointiympäristö, äänitaide, kokeellinen musiikki

Master's Thesis | Abstract

Turku University of Applied Sciences

Master of Culture and Arts

2021 | 40

Atte Häkkinen

It's a bit like its own island

- The use of the SuperCollider programming language in Finnish experimental music and sound art

In my thesis, I study the use of the SuperCollider programming language in Finnish experimental music and sound art. In my work, I have interviewed four artists who use or have used SuperCollider in their artistic work. Based on the interviews, I have compiled three themes for my thesis, which are: the SuperCollider as its own island, the user experiences of the SuperCollider programming language, and the future of SuperCollider.

According to my observations, the use of the SuperCollider language is mainly limited to art contexts. The language has its own user communities that cherish it, but the language has not spread to commercial use. Integrating SuperCollider with other programming languages and game engines, for example, is not easy and this limits the spread of the language. Despite of this, SuperCollider can be considered to have promoted people's career paths and the language has its own well-established place in the field of experimental music and sound art.

Keywords:

programming languages, programming environment, sound art, experimental music

Sisältö

Sanasto	7
1 Johdanto	9
1.1 Tutkimusaineisto	11
1.2 Haastateltavat	12
1.3 Tutkijapositio	13
2 Oma saarekkeensa	15
2.1 Livekoodaus	19
2.2 SuperColliderin opettaminen	21
2.3 Niche-alue	23
3 Käyttäjäkokemukset	25
3.1 SuperCollider, Max/MSP ja Pure Data	26
3.2 Käyttöliittymän puute	27
3.3 SuperCollider ja DAW	29
3.4 Synteesi vai samplemateriaali	30
4 SuperCollider-kielen tulevaisuus	32
4.1 SuperColliderin käyttäjäyhteisö	33
4.2 SuperColliderin integrointi muihin järjestelmiin	34
5 Lopuksi	36
Lähteet	39

Kuvat

Kuva 1. Facebook-postaus koskien Algorave työpajaa Algorave Helsingin Facebook-sivuilta	20
Kuva 2 Stenon kuvaus GitHub-sivulla.....	21
Kuva 3. Kuvassa SuperColliderin ohjelmointiympäristö, jossa näkyy Huoneiden kirja -suurteoksen Huoneeseen 26 tekemääni äänisuunnittelun koodia.	28
Kuva 4. Esimerkki SuperCollider-tutoriaalin koodista.....	30

Sanasto

DAW Käytän termiä DAW (eng. digital audio workstation) kuvaamaan musiikinteko-ohjelmia, joissa yhdistyvät esimerkiksi sekvensseri, nauhuri, mikseri ja joihin saa liitännäisiä (engl. plugin). Tässä työssä tärkeäksi DAWiksi nousee Ableton Live. DAWit eroavat ohjelmointikielistä esimerkiksi siinä, että niissä on valmiina graafinen käyttöliittymä ja musiikinteko-ominaisuuksia.

Niche Käsitteellä niche tarkoitetaan erityisen pientä markkina-aluetta, jolla on kuitenkin oma kuluttajakuntansa. Yleensä nichetuotteiden kuluttajat vaativat tuotteiltaan erikoistumista tai harvinaisuutta. Itse käytän termiä kuvaamaan SuperColliderin suhteellisen pientä, mutta omistautunutta käyttäjäkuntaa.

Ohjelmointi ja koodaus Käytän käsitteitä ohjelmointi (engl. programming) ja koodaus (engl. coding) synonyymeinä. Kyseessä on jollakin ohjelmointikielellä tehty ohjelma.

Virtuaali-instrumentti Virtuaali-instrumentilla tarkoitetaan tietokoneohjelmaa, joka käyttäytyy soittimen kaltaisesti. Virtuaali-instrumentteja voi soittaa esimerkiksi midikoskettimiston avulla. Toisaalta myös algoritmi voi soittaa näitä instrumentteja ohjelmoitsijan syöttämien parametrien mukaisesti.

Ääniohjelmointi Käsitteellä ääniohjelmointi (engl. audio coding) tarkoitan ääneen keskittyvää ohjelmointia, jota varten on kehitetty ohjelmointikieliä, joista SuperCollider on yksi. Ääntä voidaan ohjelmoida lausekielisesti tai niin sanottuja visuaalisia ohjelmointikieliä käyttäen. Visuaalisia ohjelmointikieliä on esimerkiksi MAX / MSP ja Pure

Data. Lausekielisiä ohjelmointikieliä ovat esimerkiksi C Sound, Chuck ja SuperCollider.

Ääniohjelmoija Käytän tässä opinnäytetyössä käsitettä ääniohjelmoija (engl. audio coder) kuvaamaan ketä tahansa, joka ohjelmoi ääntä. Opinnäyte keskittyy koodaamisen taiteelliseen puoleen, mutta yhtä hyvin ääniohjelmoija voisi työskennellä esimerkiksi pelien parissa tai suunnitella uutta DAWia (äänityöohjelmisto, engl. digital audio workstation).

1 Johdanto

Tutkimukseni käsittelee SuperCollider-ohjelmointikielen käyttöä äänitaiteessa ja kokeellisessa musiikissa 2020-luvun Suomessa. Ohjelmointi on koko kulttuurimme läpäisevä ilmiö, joten ei ole ihme, että se on saanut jalansijaa myös nykytaiteen kentällä.

Kotimainen äänitaiteen ja kokeellisen musiikin digitaalisuuden historia ulottuu elektronisen musiikin varhaisvaiheisiin, sillä alan pioneeri Erkki Kurenniemi rakensi digitaalitekniikkaa sisältäviä sähkösoittimia jo 1970-luvulla. Kurenniemi oli mukana 1960-luvun underground-kulttuurissa ja avantgarde-liikkeessä.

Kokeellisen musiikin kentän voidaan nähdä syntyneen avantgarden ja undergroundin risteymänä. Diskursiivisesti voidaan ajatella, että termit eroavat toisistaan. Avantgarde viittaa korkeakulttuurin kenttään ja kokeellisen musiikin kohdalla sitä on käytetty, kun on haluttu puhua koulutettujen säveltäjien musiikillisista kokeiluista. Underground on taas viittaa alakulttuuriin ja vaihtoehtokulttuuriin. Undergroundille on leimallista, että musiikin tekijät eivät ole saaneet musiikillista koulutusta, vaan he voivat tulla musiikin kentälle muilta esittävien taiteiden aloilta. Käytännössä erojen tekeminen on kuitenkin mahdotonta, sillä esimerkiksi avantgardemanifesti *Hälyjen taiteen* kirjoittaja Luigi Russolo (2018 [1916]) oli koulutukseltaan taidemaalari. Ilmiöt eivät myöskään noudata jatkuvuutta siinä mielessä, että ne kehittyisivät toisistaan jotenkin evoluution kaltaisesti, vaan saman tyyppisiä ilmiöitä nousee historian saatossa ja taiteilijat ovat aina ottaneet vaikutteita myös aiemmista taideilmiöistä. Myös haastatteleman henkilöt tulevat monipuolisesti taiteen eri kentiltä (mediataide, äänitaide, valo- ja äänisuunnittelu). Dick Higgins (2001 [1965]) käyttää englanninkielistä termiä intermedia puhuessaan siitä, kuinka taide tapahtuu usein medioiden välissä. Termi on edelleen käyttökelpoinen kuvaamaan tämänhetkistä taidekulttuuria, joka on lähtökohtaisesti monimediaista.

Kokeellinen musiikki ei rajaudu mihinkään tiettyyn genreen. Musiikintutkija Tanja Tiekso (2013) onkin kutsunut kokeellista musiikkia epägenreksi.

Kyseessä on tällöin eräänlainen kaatoluokka musiikille, joka ei mahdu tavallisten genrekäsitteiden sisään. Äänitaidetta taas käytän käsitteenä äänellisille taideilmiöille, jotka eivät mene musiikin viitekehukseen. Käytännössä raja musiikin ja äänitaiteen välillä on kuitenkin häilyvä. Riippuu pitkälti kontekstista haluammeko puhua kokeellisesta musiikista vai äänitaiteesta. Kontekstiin voi vaikuttaa niinkin arkinen asia kuin apurahahaut. Usein taiteilijat kokevat, että äänitaiteella on helpompi saada apurahoja kuin kokeellisella musiikilla. Lähtökohtaisesti käytän kuitenkin niitä käsitteitä, joita taiteilijat itse työstään käyttävät. Tutkimuksessa kokeellinen musiikki ja äänitaide ovat diskursiivisia kattokäsitteitä. Ne ovat viitekehys, jonka puitteissa tutkimushaastattelut on tehty.

Tutkimuskysymykseni on seuraava: ”Miten äänitaiteilijat ja kokeellisen musiikin tekijät käyttävät SuperCollideria taiteellisen työnsä tukena?” Haastattelen tutkimuksessani sellaisia kokeellista musiikkia tai äänitaidetta tekeviä henkilöitä, jotka käyttävät tai ovat käyttäneet taiteellisessa työssään SuperCollider-ohjelmointikieltä. Työni tavoitteena on tutkia SuperCollider-kielen käytettävyyttä taiteellisessa työskentelyssä. Tarkastelemalla ääniohjelmoinnin työskentelyprosesseja ja haasteita, sekä pohtimalla SuperCollider-kielen tulevaisuutta.

Tutkimukseni etenee seuraavasti. Olen jakanut työni kolmeen eri teemaan. Luvun kaksi teema on SuperCollider omana saarekkeenaan. Käsittelen luvussa livekoodausta, SuperColliderin opettamista ja SuperCollideria niche-alueena. Toinen teema on käyttäjäkokemukset SuperColliderista. Sitä käsittelen luvussa kolme tarkastelemalla SuperColliderin suhdetta muihin ääniohjelmointikieliin, käyttöjärjestelmän puutetta, SuperColliderin suhdetta DAWiin, sekä synteessin ja samplemateriaalin käyttöä SuperColliderissa. Tutkimukseni kolmantena teemana on SuperCollider-kielen tulevaisuus. Tätä aihetta käsittelen luvussa neljä tarkastelemalla SuperColliderin käyttäjäyhteisöä, sekä SuperColliderin integroimista muihin järjestelmiin. Viimeisessä luvussa käsittelen tutkimuksen tuloksia.

1.1 Tutkimusaineisto

Keräsin aineistoni teemahaastattelemalla ihmisiä, jotka käyttävät tai ovat käyttäneet SuperCollider ohjelmointikieltä taiteellisessa työssään. Minulla oli käytössäni kymmenen kysymyksen kysymyslista, jota käytin runkona haastatteluissa. Haastattelukysymykset painottuivat kokemuksiin SuperColliderin käytöstä. Käytännössä haastattelukeskustelut etenivät teemasta toiseen niin, että kaikkiin kysymyksiini sain vastauksen, mutta saatoin vaihtaa kysymysten järjestystä. Osaan kysymyksistä sain vastauksen keskustelun lomassa, niin että kysymystä ei tarvinnut esittää. Esitin haastattelun yhteydessä myös mieleeni juolahtaneita lisäkysymyksiä tarvittaessa. Haastatteluissa keskityin kokemuksiin SuperCollider-ohjelmointikielen käytöstä, koska tunnen sen itse parhaiten käytössä olevista ääneen keskittyivistä ohjelmointikielistä. Haastattelut rönsyilivät aiheen ulkopuolelle minun ja haastateltavien kollegiaaliseen ajatuksen vaihtoon, mutta pääpaino oli kuitenkin SuperColliderin käsittelyssä.

SuperCollider-ohjelmointikielen käyttö liittyy kansainvälisesti vallalla olevaan livekoodaustrendiin ja yleisempään keskusteluun ohjelmoinnista. Aineiston kautta pääsin käsiksi yleisempään digitaalisuutta koskevaan keskusteluun ja eri työvälineiden käyttökelpoisuuteen ääniohjelmoinnissa.

Tutkimuskirjallisuuteni koostui digitaalista äänenkäsittelyä koskevasta kirjallisuudesta. SuperCollider on usein tässä kirjallisuudessa maininnan tasolla lukuunottamatta *The SuperCollider Book* -kirjaa, joka sekin koostuu lähinnä koodausohjeista. Tutkimuskirjallisuutta lukiessa ei voi välttyä huomiolta, että ääniohjelmointia ei ole juuri tutkittu kulttuurillisesta näkökulmasta. Kirjallisuudessa SuperCollider usein niputetaan yhteen muiden ääniohjelmointiympäristöjen, kuten Max/MSP:n kanssa ja ääniohjelmointiympäristöistä puhutaan yhtenä nippuna ohjelmia.

1.2 Haastateltavat

Haastateltavana minulla on opinnäytteessäni neljä taiteilijaa, jotka ovat käyttäneet taiteellisessa työssään SuperCollider-ohjelmointikieltä. Taiteilijat Kirsi Ihalainen, Joonas Siren, Jusu Vehviläinen ja Ellen Virman valikoituivat haastateltaviksi, koska tiesin heidän käyttäneen SuperCollidieria taiteellisissa töissään. Suomessa on muitakin SuperColliderin käyttäjiä, mutta nämä neljä haastateltavaa tulevat taiteen kentän eri puolilta ja tekevät taidetta ammattimaisesti ja kansainvälisesti.

SuperColliderin käyttö on haastateltavilla erityyppistä. Ihalainen käyttää SuperCollidieria lähinnä installaatioissaan, mutta hän on luopunut kielen käytöstä asiakkaille myytävien tuotteiden osalta. Jusu Vehviläinen käyttää SuperCollidieria Pink Twins -duossa ja hänellä on haastateltavista pisin kokemus kielen käyttäjänä. Joonas Siren käyttää SuperCollidieria Forces musiikkiprojektissaan, jossa hän käyttää generatiivisia elementtejä. Siren on myös ollut tuomassa Algorave-liikettä Suomeen. Ellen Virman taas on ottanut SuperColliderin sävellystyökalukseen ja tekee sillä kappaleita. Taiteellinen työskentely on ajanut kaikki haastateltavat SuperColliderin pariin eli he ovat alkaneet koodata sillä nimenomaan taiteen takia. Seuraavaksi esittelen tarkemmin haastateltavat.

Äänitaiteilija Kirsi Ihalainen on Kelosound Oy:n toimitusjohtaja. Kelosound Oy tekee kelopuusta valmistettavia KELOSOUND®-tuotteita, jotka soittavat luonnonääniä ja toimivat esimerkiksi sisustuselementteinä. Kelosound Oy:n tuotteisiin kuuluu myös SAUNASOUND®, eli soivat saunat. Tämä äänijärjestelmä asennetaan saunan rakenteisiin siten, että saunan rakenteet soittavat luonnonääniä ja resonoivat. Ihalainen on tehnyt pääsääntöisesti isompia ääni-installatioita luonnonäänillä. Tämän lisäksi hän on tehnyt yksittäisiä teoksia suomalaisten perinne-eläinten äänistä. Teokset muistuttavat veistoksia, jotka voi laittaa seinälle taulumaisesti, mutta erona perinteiseen veistostaiteeseen on se, että Ihalaisen teokset soivat. Ihalainen on myös tehnyt

Suomi 100-teoksen nimeltään Luontoäidin kutsu (2017), joka sijaitsee Suomen luontokeskus Haltiassa. (Ihalainen 2021.)

Ääni- ja mediataiteilija Joonas Siren on tehnyt visuaalista taidetta ja ääniinstallaatioita. Hän myös tekee musiikkia nimellä Forces käyttäen SuperCollider-ohjelmointikieltä. Viime aikoina Siren on perehtynyt Unreal Engine 3D-pelimoottorin käyttöön ja tekee taidetta sillä. Sireniä kiinnostaa töissään teknologian väärinkäyttö ja generatiivisuus. (Siren 2021.)

Jusu Vehviläinen on muusikko ja koodari. Hän muodostaa veljensä Vesa Vehviläisen kanssa Pink Twins -duon, joka yhdistelee kokeellista kollaasityyppistä musiikkia videotaiteeseen. Pink Twins on toiminut 1997 lähtien ja se on esiintynyt Euroopassa, Amerikassa, Aasiassa ja Australiassa. (Vehviläinen 2021.)

Ellen Virman on valo- ja äänisuunnittelija, joka tekee musiikkia SuperCollider-ohjelmointikieltä käyttäen nimellä Eilien. Eilien on tunnettu live-esiintymistään ja hänen *Digital Lovers* -albumi (2021) on kerännyt positiivista huomiota osakseen. Virman on esiintynyt Suomen lisäksi Ukrainassa. (Virman 2021.)

1.3 Tutkijapositio

Lingvisti Kenneth Pike (1912–2000) käyttää käsitteitä emic ja etic. Emicillä tarkoitetaan merkityksiä, joita tieteellisesti koulutetut asiantuntijat antavat kulttuureille ja kielelle. Eticillä taas tarkoitetaan merkityksiä, jotka nousevat kulttuurissa sisällä olevien ja esimerkiksi paikallista kieltä puhuvien keskuudesta. (Rice 2008, 53.) Myös oma tutkimukseni ääniohjelmoinnista asettuu näiden kysymysten keskiöön.

Olen itse käyttänyt SuperCollideria vuodesta 2012, mutta vasta viime vuosina olen ottanut sen osaksi taiteellista työkalupakkiani. Viimeisin SuperColliderilla tehty työ oli äänisuunnittelu Maija Westerhomin immerssiiviseen nukketeatteriinstallaatioon, joka hyödynsi AR-tekniikkaa. Huone 26 oli osa *Huoneiden kirja* -suurteosta, joka esitettiin Turussa kesällä 2021. Teoksessa SuperColliderilla

tehtyjä ääniä ohjasi Unity-pelimoottorilla tehty ohjelma. Olen myös rakentanut SuperColliderilla syntetisaattorin, jota olen käyttänyt livekeikoillani.

Ääniohjelmoijana olen ristiriitaisessa tilanteessa. Missä mielessä minun on käsitettävä itseni koulutetuksi tarkkailijaksi (emic) ja missä mielessä kulttuurin sisäpiiriläiseksi (etic). Onneksi minun ei ole pakko valita puoliani, vaan liikun tutkimuksessa mainittujen positioiden välissä.

2 Oma saarekkeensa

SuperCollider muodostaa oman saarekkeensa ohjelmointikielten joukossa. Kyseessä on äänen käsittelyyn kehitetty ohjelmointikieli, joka löytänyt paikkansa lähinnä kokeellisen musiikin ja äänitaiteen saralta. Tässä luvussa käsittelen SuperColliderin ympärille kehittynyttä livekoodauskulttuuria, SuperCollider-ohjelmointikielen opettamista, sekä SuperCollideria omana niche-alueenaan. Näitä alueita käsittelemällä pyrin luomaan kuvan siitä millainen ohjelmointikieli SuperCollider on.

SuperCollider on James McCartneyn vuonna 1996 luoma ympäristö ja ohjelmointikieli reaaliaikaiseen äänisynteesiin ja algoritmisäveltämiseen. McCartney suunnitteli SuperColliderin interaktiiviseksi järjestelmäksi, joka noudattaa asiakaspalvelinparadigmaa: palvelin (server) toimii käyttöjärjestelmän taustalla ja tuottaa ääntä, kun se vastaanottaa komentoja asiakkaalta (client). Palvelinta ohjataan tiedonsiirtoprotokollalla, joka tunnetaan nimellä Open Sound Control (OSC). Ohjaukoodi voidaan kirjoittaa koodausikkunaan ja suorittaa se. (Elsea 2013, 363–364.)

James McCartney (2011) kertoo vuonna 2011 ilmestyneessä *The SuperCollider Book* -kirjassa, että SuperColliderin kehittäminen liittyi tiiviisti Applen tietokoneiden kehitykseen. McCartney oli opetellut C-kieli pohjaista CSoundohjelmointikieltä, mutta hän ei pitänyt sen monimutkaisuudesta. Niinpä vuonna 1990 McCartney kehitti Synth-O-Matic nimisen ohjelman, joka sisälsi graafisen käyttöliittymän. Ohjelma oli kuitenkin hidas ja hän lakkasi käyttämästä sitä. Tilanne muuttui, kun McCartney osti vuonna 1994 Power-PC-suorittimella toimivan ajankohtaan nähden nopean tietokoneen. Uudella tietokoneellaan hän pystyi ajamaan Synth-O-Matic -ohjelmaansa 32 kertaa nopeammin kuin aiemmin ja tämä loi edellytykset reaaliaikaiselle äänisynteessille. Samoihin aikoihin granulaarisynteesin kehittäjänä tunnettu Curtis Roads kehotti jatkamaan SynthO-Maticin kehittämistä. Syntyi SuperCollider-ohjelman ensimmäinen versio, joka julkaistiin maaliskuussa 1996. (McCartney 2011, x–xi.)

SuperCollider oli alkujaan maksullinen ohjelma. Kauppa ei kuitenkaan käynyt ja McCartney oli ratkaisujen edessä. Vuonna 2002 hän aloitti työnsä Applella ja SuperColliderin versio 3 päätettiin laittaa vapaaseen levitykseen. SuperCollider-ohjelmointikielestä tuli ilmainen vapaan lähdekoodin ohjelmointikieli ja ohjelmointiympäristö. Applella McCartney on kehittänyt esimerkiksi CoreAudio arkkitehtuuria OSX:lle ja iOS:lle.

SuperCollider-ohjelmointikieli on ilmainen ja sillä on vapaa lähdekoodi, mikä tarkoittaa, että ohjelmointikieltä voi käyttää ja muokata kuka tahansa. Ohjelmointikieli on jaettu kahteen osaan. Siinä on ääniä tuottava palvelin scsynth, joka siis käytännössä tarkoittaa esimerkiksi digitaalisia äänigeneraattoreita. Toinen osa on itse ohjelmointikieli eli slang, jolta lähtevät käskyt ohjaavat ääniä tuottavaa scsynthiä. Käskyjen välisenä protokollana toimii UDP (User Datagram Protocol). Kyseessä on tietoliikenneprotokolla, joka muistuttaa internetissä käytettyä TCP/IP-protokollaa, mutta eroaa esimerkiksi siinä, että datapakettien perillemeno ei varmisteta. UDP:n yleisrasite on pienempi kuin TCP/IP:n ja tämän vuoksi data kulkee tavallaan paikasta toiseen suoremmin. Sclangin ja scsynthin välisessä datasiirrossa käytetään Open Sound Controlia (OSC), joka käyttää UDP:tä. Käytännössä Open Sound Control mahdollistaa myös esimerkiksi älylaitteiden käytön SuperColliderin kanssa verkon välityksellä. Ohjelmointiympäristössä ei ole valmista graafista käyttöliittymää (GUI, graphical user interface), mutta SuperCollider tarjoaa työkaluja käyttöliittymän tekoon. SuperCollider tukee MIDIä. Siihen voi kytkeä MIDI-kontrollerin eli esimerkiksi koskettimiston kiinni.

Ohjelmointikieli sisältää lukuisia valmiita äänigeneraattoreita (unit generator, Ugen), jotka voivat tuottaa erilaisia ääniä kohinasta erilaisiin aaltomuotoihin. Lisäksi joidenkin äänigeneraattoreiden puskureihin (engl. buffer) voi ladata ääninäytteitä (engl. sample). Äänigeneraattoreita voi myös koodata itse C ja C++ -ohjelmointikielillä. Itse SuperCollider on koodattu C/C++:lla. Äänigeneraattoreiden käyttö muistuttaa modulaarisyntetisaattoreiden toimintaa. Erona on se, että SuperColliderissa asiat tehdään koodaamalla, eikä johtoja kytkemällä. Yksi kiinnostava ominaisuus on myös äänimateriaalin

monistettavuus ja niin kutsuttu granulaarisynteesitekniikka, jossa samplet eli ääninäytteet pilkotaan pieniksi millisekuntien mittaisiksi rakeiksi ja niitä päällekkäin soittamalla saadaan aikaan ilmiö, jota kutsutaan granulaaripilveksi. Ilmiö muistuttaa hiukan esimerkiksi sateen ääntä eli sadat pienet samplerakeet voivat soida samaan aikaan.

SuperCollider on suunniteltu algoritmisäveltämiseen ja generatiiviseen musiikkiin eli sille voi syöttää parametrit, joiden perusteella ohjelma luo ääntä tai musiikkia. Parametrit voivat olla nuottidataa tai oikeastaan mitä tahansa musiikin digitaalisia parametreja liittyen äänten keston, korkeuteen tai siihen miltä puolelta stereokuvaa ääni tulee. SuperColliderilla tehdyn ohjelman voi laittaa soittamaan kokonaisen kappaleen tai sille voi syöttää todennäköisyyksiä, joiden mukaan ohjelma ääniä soittaa. Äänet voi myös laittaa reagoimaan ulkopuolisiin parametreihin eli liikkeestä tai valosta saatuun dataan eli tietoon, joka on muunnettu analogisesta digitaalisignaaliiksi. Esimerkiksi Arduino-mikrokontrolleria voidaan käyttää tähän tarkoitukseen.

Omimmillaan SuperCollider on silloin, kun sillä tehdään generatiivista musiikkia tai algoritmisäveltämistä. Ajatus generatiivisesta musiikista esitettiin jo 1950-luvun lopulla, jolloin Markovin ketjua käytettiin analysoimaan nuottidataa. Tätä analyysiä käytettiin luomaan uusia sävellyksiä Moogin ja Buchlan valmistamilla syntetisaattoreilla. Syntetisaattorit laitettiin soittamaan itseään. Ajatus on luonut kokonaisen algoritmisäveltämisen kulttuurin, johon kuuluvat esimerkiksi ohjelmat: MUSIC 5, Csound, Max ja SuperCollider (Galanter 2016, 148). Nykyisin generatiivinen esitys voi tarkoittaa esimerkiksi sitä, että tietokone laitetaan soittamaan jotakin äänimateriaalia itsekseen esimerkiksi galleriatilaan ja äänimateriaali muuttaa muotoaan koko ajan. Algoritmisäveltäminen taas tarkoittaa metodia, jossa tietokone laitetaan soittamaan melodioita tai rytmejä valmiiksi syötetyn algoritmin perusteella. Algoritmisäveltäminen ja generatiivisuus perustuvat todennäköisyyslaskentaan ja siihen mitä parametreja ääniohjelmoija syöttää ohjelmaan.

SuperCollidieria pidetään haastattelujen perusteella omana saarekkeenaan.

Tämä tarkoittaa sitä, että SuperCollider ei helposti vertaudu muihin ohjelmointikieliin. SuperCollider on myös jäänyt lähinnä taidekäyttöön, eikä sitä ole juuri hyödynnetty kaupallisissa tuotteissa. Ohjelmointikieli on myös muodostanut oman käyttäjäyhteisönsä, joka on keskittynyt kielten kehittämiseen ja käyttöön.

Lähimpänä SuperCollider-kielen syntaksia eli ohjelmointikielen lauseoppia, on 1970-luvun alussa kehitetty ohjelmointikieli nimeltään SmallTalk. Apple on käyttänyt SmallTalkia esimerkiksi 1980-luvulla MacApp-ohjelmien käyttöliittymien tekoon (Kaisler 2005).

Kyllähän siinä on aika vahva se SmallTalk vaikutte tai tausta. Sullla on tavallaan se yksi semmoinen kokonainen ympäristö missä on se kehityspuoli, mutta myös se runtime ja se on aika erilainen malli kuin normaalisti tai tavallaan normaalit ohjelmat ei toimi silleen. (Vehviläinen 2021.)

Vehviläisen mukaan normaalit ohjelmat tehdään kehitysalustalle ja sitten ohjelmakoodi käännetään ajettavaan muotoon. Tämän jälkeen ohjelma toimii käyttöjärjestelmän päällä. SuperColliderissa kaikki on tavallaan samassa. (Vehviläinen 2021.)

Sinänsä kiinnostava malli ja ehkä se just soveltuu tavallaan tämmöiselle tämmöisen musiikin synteesispuoleen, koska se on tavallaan hyvin semmoinen reaaliaikainen. Sä kirjoitat sinne rimpun, joka määrittelee jonkun synteessin ja laitat sen soimaan ja pystyt muokkaamaan sitä siinä ajon aikana koko ajan. (Vehviläinen 2021.)

SuperCollider toimii siten, että siinä lähetään viestejä oliolta toiselle (Vehviläinen 2021). Malli ei ole yleistynyt ohjelmointikielten parissa, mutta mahdollistaa reaaliaikaisen äänisynteessin tekemisen niin, että SuperCollidieria voidaan käyttää esimerkiksi livekoodaamiseen. SuperColliderin vahvuus on sen reaaliaikaisuus, nopeus ja tehokkuus. Kieli toimii tehokkaammin kuin sen visuaaliset kilpakumppanit ja se on mahdollista saada toimimaan esimerkiksi Raspberry Pi -tietokoneissa.

2.1 Livekoodaus

Livekoodauksella tarkoitetaan musiikissa käytettyä tekniikkaa, jossa ääniohjelmoija koodaa ääntä livenä ja luo siten musiikkia. Yleensä livekoodaustapahtumissa koodi heijastetaan seinälle, jolloin yleisö pystyy seuraamaan koodaustapahtumaa reaaliaikaisesti. Livekoodausta tehdään usein SuperColliderin lisäosalla, joka on nimeltään Tidal Cycles.

Livekoodaus käyttää koodin modulaarisia osia, joita voidaan muokata erikseen. Tämä mahdollistaa nopeidenkin muutosten tekemisen reaaliaikaisesti, mutta livekoodattu musiikki on samalla synkronisoitua. SuperCollideria käyttävä livekoodari Nick Collins vertaa livekoodaustekniikkaa 1900-luvun säveltäjäpianistien tekniikkaan (Hugill 2012). Livekoodausta on popularisoinut erityisesti 2010-luvun alussa Lontoosta lähtenyt Algorave-liike, jonka kantavia voimia edellä mainittu Nick Collins on. Usein tietokoneen näyttö heijastetaan yleisölle visuaaliseksi elementiksi algorave-keikoilla, jotta yleisö pystyy näkemään mitä ruudulla tapahtuu. Tämä toimintamalli ei kuitenkaan kiinnosta kaikkia ja Jusu Vehviläinen kertookin, ettei Pink Twins käyttänyt livekoodausta, vaikka Vehviläinen selkeästi tuntee myös tämän puolen SuperColliderista.

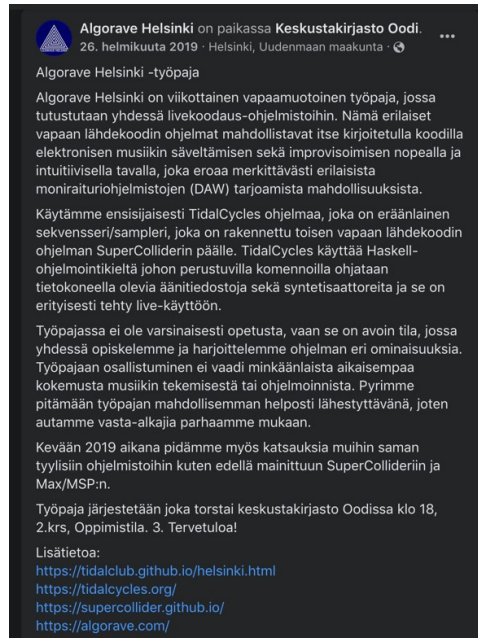
Meillä on se visuaalinen puoli myös siinä mukana, niin siinä ei ole mun mielestä tavallaan mitään syytä tuoda siihen semmoista teknisempää puolta jotenkin selvittämään tai avaamaan sitä. Se konsepti juttu on keskittynyt kuitenkin siihen lopputulokseen eikä niihin prosesseihin. (Vehviläinen 2021.)

Suomeen livekoodausta ja algorave-kulttuuria on tuonut etupäässä Algorave Helsinki, jossa ovat mukana Joonas Siren ja Viktor Toikkanen. Myös Ellen Virman on osallistunut Sirenin ja Toikkasen järjestämiin livekoodaustyöpajoihin, joita järjestettiin Helsingin Keskustakirjasto Oodissa.

Pidettiin vähän sellaisia viikoittaisiin meettejä ja missä opettelin itse ehkä enemmän käyttää Tidal Cyclesiä siinä, mutta sinne sai tulla kaikilla taidoilla ja ilman mitään ennakkokokemusta. (Virman 2021.)

Algorave Helsingin järjestämä työpajoja järjestettiin vuonna 2019

Keskustakirjasto Oodissa ja niissä keskityttiin TidalCycles lisäosan käyttöön. Tidal Cyclesissä erilaisia sekvenssityyppisiä rakenteita pystytään nopeasti luomaan kirjoittamaan esimerkiksi Atom-editorilla.



Kuva 1. Facebook-postaus koskien Algorave työpajaa Algorave Helsingin Facebook-sivuilta.

Algorave-työpajojen suosio yllätti Joonas Sirenin ja Viktor Toikkasen ja Oodin sali täyttyi ääriään myöten (Siren 2021).

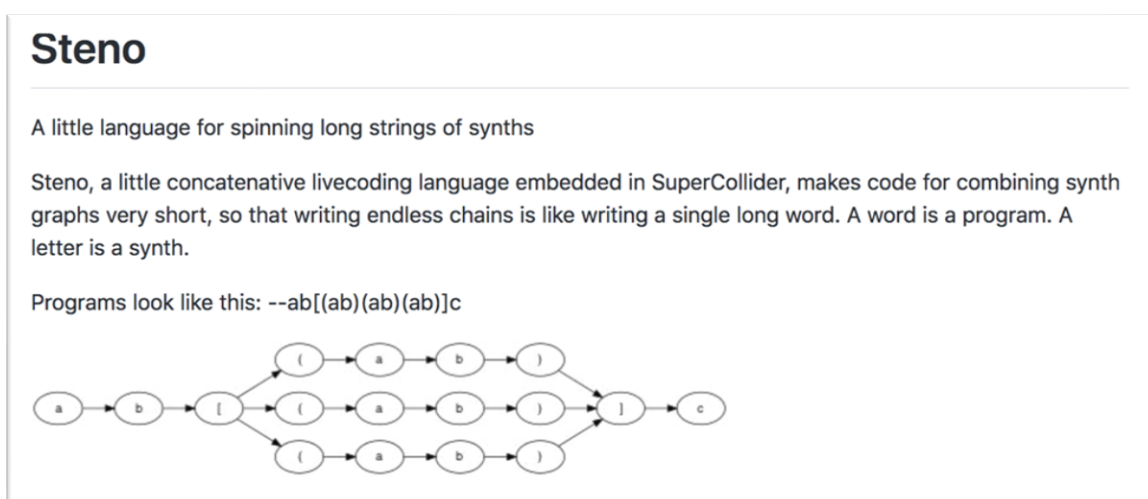
Me tosiaan pidettiin vuoden verran sitä algoritmi Helsinki hommaa. Se oli enemmän Tidal Cycles -juttu ja alussa meillä oli heti ihan täynnä se sali siellä Oodissa. Mä en tiedä, että miten me oltiin tavoitettu niitä ihmisiä, koska mä en tuntenut niistä ketään, mut ilmeisesti Guardianilla oli ollut joku artikkeli et mitä se suunnilleen vois olla se Algorave tai livekoodaus. (Siren 2021.)

Siren valittelee haastattelussaan, että ihmisten valmiudet audiokoodaamiseen eivät olleet hyvät ja varsinaisen SuperColliderin opettaminen oli mahdotonta. Algorave Helsinki on kuitenkin jatkanut SuperColliderin ilosanoman viemistä eteenpäin ja audiokoodaukselle on ilmiselvää kiinnostus olemassa.

2.2 SuperColliderin opettaminen

Do Cheaper (2013, 102) esittää, että SuperCollider ei sovellu aloittelijatasen opetuskieleksi. Tidal Cycles on helpottanut tilannetta Joonas Sirenin ja Viktor Toikkasen vetämissä algorave-työpajoissa, koska Tidal Cyclesillä pystyy tekemään helposti tehtyä valmiilla sampleilla esimerkiksi rumpukompeja. Tidal Cyclesin kanssa ilmeni kuitenkin opetuksen yhteydessä teknisiä ongelmia, sillä Tidal Cycles vaati Haskell-kirjaston, jota ei helposti saatu asennettua tiettyihin Mac-tietokoneisiin. Toikkanen ja Siren eivät saaneet työpajoista palkkaa, joten heistä alkoi tuntua, että he olivat opettamassa ilmaiseksi. Toikkanen pääsi vuoden 2019 aikana Sibelius Akatemian musiikkiteknologian oppiaineeseen, eikä Siren jaksanut pyörittää työpajoja yksin, joten työpajat loppuivat siihen. (Siren 2021.)

Algorave Helsinki kutsui myös SuperColliderin kehittäjä saksalaisen Till Bovermannin esittelemään SuperCollideriin sisällytettävästä metakieltä Stenoa (Siren 2021). SuperCollideriin on saataville erilaisia lisäominaisuuksia, jota käyttäjäyhteisö kehittää. Steno on yksi tällainen metakieli ja lisäominaisuus, joka mahdollistaa SuperColliderin käskyjen ketjuttamisen nopeasti.



Oodissa ainoastaan Siren ja Toikkanen pystyivät seuraamaan Bowermannin esittelyä Stenosta. Yleisö ei pysynyt mukana. Bowermann on itse asiassa opettanut Ihalaista Aalto Yliopistossa (Ihalainen 2021).

Till Bowermann, joka sitä silloin alkuun opetti, niin mulla meni sen kanssa sukset ristiin, kun se veti sen kurssin niin huonosti ja mua ärsytti, kun mä en oppinut sitä kieltä sen opettamana, vaikka se on itse ollut kehittämässä sitä kieltä. Se, että vaikka sä kehität jotain kieltä, niin se ei tee susta hyvää opettajaa. Sehän on vaan näin. Sitten mä lähdin ovet paukkuen sieltä kurssilta sen jälkeen, kun se oli sellaista, että se ei itse asiassa mennyt siihen koodaamisen ollenkaan. Siinä kohtaa, kun rupesi olemaan, että mennään tonne äänikävelylle ja sitten mä olin silleen, et nyt riittää. Mä en tullut tänne äänikävelee. Mä tulin tänne koodaa. (Ihalainen 2021.)

Bowermannin lisäksi Ihalaisella ja Sirenillä on myös toinen yhteinen nimittäjä eli Alejandro Olarte, joka opettaa SuperCollideria Sibelius Akatemiassa. Ihalainen kävi Olarten kurssin ja oppi kieltä siellä (Ihalainen 2021).

Mut onneks sitten Alejandro oli paljon ystävällisempi siinä mielessä, että se ymmärsi mun kysymykset ja sitten mä pystyin kysyä siltä ihan mitä vaan ja se ei hermostunut ja opin sen kielen sitten sitä kautta. (Ihalainen 2021.)

Siren sanoo myös oppineensa SuperColliderin käytön Alejandro Olarten kurssilla. Siren ei ollut sisällä Sibelius Akatemiassa, mutta Olarte järjesti vuoden mittaisen kurssin, johon oli mahdollisuus päästä myös avoimen yliopiston kautta. Loppujen lopuksi kurssille jäivät lähinnä avoimen puolen opiskelijat (Siren 2021).

Oma arvioni on, että Olarten kursseilla on merkittävä rooli SuperColliderin leviämässä Suomessa. Tämä myös tarkoittaa sitä, että SuperColliderin opetus on painottunut Helsinkiin ja nimenomaan Sibelius Akatemiaan. Ellen Virman tosin kertoo, että Taideyliopiston Teatterikorkeakoulun Valo- ja äänisuunnittelun koulutusohjelmassa SuperColliderista tuli pienoinen muoti-ilmiö vuonna 2017 (Virman 2021).

Joku oli jossain vaihdossa törmännyt siihen (SuperCollideriin) ja vähän opetellut sitä. Sitten yks toinen mun luokkatoveri oli silleen, että ei vitsi mä aion nyt opetella sitä vähän ja sitten sekin opetteli sitä. Sitten kun kaikki mun ympärillä alkoivat opettelemaan sitä, niin mäkin olin silleen, että en mä halua jäädä ulkopuoliseksi. Mä haluan tietää, että mitä nämä kaikki muut tekevät ja osaavat. Sitten mäkin aloin opettelemaan sitä, mutta sit siinä kävi silleen, että mä vähän jäin siihen koukkuun ja vaan jatkoin ja jatkoin ja sitten ilmeisesti sitten monilla muilla se kiinnostus vähän lopahti. Muistan, että jonkun puolen vuoden jälkeen tästä hittibuumista mulle tuli jotain ongelmia siinä SuperColliderissa ja sitten

menin niiltä mun kurssitovereilta kysymään, että minulle tuli tällainen ongelma, että mitä mun pitäis tehdä? Sitten ne olivat ihan silmät pyöreinä, että ei mitään hajua, että mitä sä olet oikein tehnyt? Ne oli lopettanut sen jo kauan sitten, ja mä en ollut tajunnut sitä. (Virman 2021.)

Omaa tarinaansa tosin kertoo se, että ainoastaan Virman jäi käyttämään kieltä ja muiden mielenkiinnon kohteet siirtyivät muualle. Kiinnostavaa onkin, että miksi SuperCollider ohjelmointikielenä karkottaa osan ihmisiä ja osa jollain tavalla haltioituu siitä niin, että SuperColliderista tulee heidän pääasiallinen musiikintekovälineensä.

2.3 Niche-alue

SuperCollider ei ole helpoimpia kieliä oppia, eikä opettaa. Se ei muistuta yleisimpiä kieliä, kuten JavaScriptia, Pythonia tai Javaa. Jusu Vehviläisen mukaan SuperCollider ei ole ohjelmointikieli, jonka niin sanotusti kantaisi mukanaan muualle (Vehviläinen 2021). Myös Ihalainen pohtii, miksi SuperCollider ei ole levinnyt (Ihalainen 2021).

Tota SuperCollideria kun tuppaa vaan käyttämään äänikoodarit ja kokeellisen musiikin tekijät. Se ei ole päässyt tavallaan sieltä autotallista pois, jos niin voisi sanoa. Se ei ole levinnyt ja mä en oikein osaa sanoa siihen mitään muuta syytä kuin, että tavallaan se ei ole tarpeeksi kiinnostava koodikieli isojen firmojen ja kaupallisten tahojen mielestä. (Ihalainen 2021.)

SuperColliderin integroiminen muihin ohjelmointiympäristöihin on hankalaa, eikä sitä myöskään välttämättä osata käyttää ohjelmointitaustasta huolimatta (Ihalainen 2021). SuperCollider on jäänyt omaksi niche-alueekseen, jolla on oma vankka kannattajakuntansa, mutta sitä eivät ole kaupalliset toimijat ottaneet omakseen.

Liiketaloudessa niche-termiä käytetään kuvaamaan pienten yritysten toimintaa, joiden tuotteet on suunnattu rajatulle kohdeyleisölle, jotka haluavat erikoistuneempia tuotteita. SuperCollideria on yritetty integroida esimerkiksi pelimoottoreihin kotimaisen tietokonepelijätin Remedyn toimesta (Ihalainen 2021). Siitä ei ole kuitenkaan ole tullut teollisuusstandardia alalle, vaan se on jäänyt omaksi saarekkeekseen. Philips (2014) esittää, että generatiivisen musiikin lisääminen peleihin ei välttämättä tuottaisi kiinnostavia lopputuloksia,

koska generatiivisuus tuottaa sattumanvaraisia melodioita. En kuitenkaan usko, että SuperColliderin etu olisi perinteisen säveltäjyyden korvaamisessa, vaan kokonaisvaltaisessa äänisuunnittelussa. Esimerkiksi generoimalla erilaisia äänitehosteita pelaajalle voidaan tuottaa erilaisia äänikokemuksia pelikerrasta riippuen.

Vehviläinen kiteyttää hyvin sen miksi SuperCollider on olemassa. Puhtaasti ääneen keskittyviä ohjelmointikieliä on suhteellisen vähän ja SuperCollider ajaa asiansa.

Se on niin tavallaan fokusoitunut siihen ääniaihepiiriin, mutta muistaakseni tämä SuperCollider tekijäkin jossain vaiheessa mainitsi, että itse asiassa SuperCollider on turhaa, että kaiken pystyy tekemään C++:lla. Jos tällöistä systeemiä ei olisi, niin sitten se pitäisi vaan ensin tehdä. Tavallaan se tällöinen reaaliaikainen ääneen fokusoitunut ohjelmointikieli, niin totta kai semmoinen pitää olla. Mitä vaan työkaluja sinänsä voi käyttää, mutta kyllä tämä on hyvin ytimekäs siinä mitä se tekee. (Vehviläinen 2021.)

SuperCollider mahdollistaa ääniohjelmoijien keskittymisen taiteelliseen työhön sen sijaan, että heidän pitäisi rakentaa ohjelma, jolla rakennetaan virtuaalisia instrumentteja.

3 Käyttäjäkokeemukset

Tässä luvussa käsittelen SuperCollideriin liittyviä käyttäjäkokeuksia. Luvussa käsitellään SuperCollider-ohjelmointikielen suhdetta muihin vastaaviin kieliin tai ohjelmistoihin. Pohditaan käyttöliittymän puutetta, sekä käyttökokeuksia synteisillä ja sampleilla.

Syy SuperCollider-kielen opettelemiseen ei välttämättä löydy käytännöllisyydestä. Kyseessä on kokemuksellinen asia, jota Ellen Virman kuvaa seuraavasti.

Mä aloin sitä (SuperCollideria) opettelee, niin se vaan tuota tuntui musta tosi silleen rauhoittavalta ja siltä että minun oli helppo keskittyä sen kanssa. Siis ne ovat oikeastaan varmaan ne syyt. Sitten myös se, että sillä oli musta vaan tosi kiehtovaa tuottaa ääntä sillä tavalla. (Virman 2021.)

Koodilla äänen tuottaminen on erilaista kuin äänen tuottaminen perinteisillä instrumenteilla. Se on myös erilaista kuin äänen tuottaminen esimerkiksi valmista DAWia (digital audio workstation) käyttämällä. Vaikka haastateltavat kuvaavat SuperCollideria instrumentin rakennustyökaluksi, niin kielelle asetetut päämäärät eroavat toisistaan. Virman kertoo tekevänsä SuperColliderilla kappaleita (Virman 2021). Vehviläinen kuvaa Pink Twinsin musiikkia vapaamuotoisena ja kollaasityyppisenä hifi noisena (Vehviläinen 2021). Sireniä kiinnostaa musiikissaan ja ääni-installaatiossa generatiivisuus (Siren 2021). Ihalaisen äänitaide painottuu luonnon ja eläinten äänillä tehtyihin teoksiin (Ihalainen 2021).

Edellä luetellusta lähtökohdista päätellen voisi ajatella, että SuperColliderin käyttötarkoitukset ovat moninaiset. Niin varmasti osittain onkin, mutta toisaalta SuperColliderin käyttö on rajautunut pääosin taidekontekstiin. Kielen integroiminen muihin ohjelmointikieliin ja ympäristöihin on hankalaa. Tästä syystä sitä ei juuri käytetä muualla kuin kokeellisen äänen teossa.

Ensinnäkään se ei ole kauhean kaupallisessa käytössä juurikaan ja sen yhteensopivuus monen eri platformin kanssa on aika epävarmaa. Koska se on tuommoinen ilmaiskehitetty koodi, niin sen käyttö on aika vähäistä. (Ihalainen 2021.)

SuperCollider-kielen leviämisessä Suomessa Sibelius Akatemiolla on merkittävä rooli ja Ihalainen ja Siren mainitsevatkin Alejandro Olarten SuperCollider-kurssin merkittäväksi sen suhteen, että siellä kieltä oppi käyttämään (Ihalainen 2021, Siren 2021). Nykyisin tilanne on toki muuttunut jonkin verran ja kielen oppimisen kynnyksiä madaltavat esimerkiksi Eli Fieldsteelin YouTube-tutoriaalit, jotka Siren ja Virman mainitsevat haastattelussa (Virman, Siren 2021). Virman kertoo katsoneensa Fieldsteelin tutoriaalit ja opetelleensa kieltä niiden avulla. Myös omassa SuperColliderin oppimisprosessissani Fieldsteelin tutoriaaleilla on merkittävä rooli.

3.1 SuperCollider, Max/MSP ja Pure Data

SuperCollider on tietysti vain yksi monista ohjelmointikielistä, joita äänen käsittelyyn on saatavilla. Tavallisimmissa ohjelmointikielissä on rajapinta äänelle usein olemassa, mutta SuperColliderin erikoisuus on se, että se on niminomaan ääniohjelmointikieli. Sitä ei ole suunniteltu varsinaisesti mihinkään muuhun. Haastatteluissa SuperColliderin rinnalla mainitaan usein Max/MSP ohjelmointiympäristö, josta kaikilla haastateltavilla on jonkin asteista kokemusta. Max/MSP on IRCAM instituutissa kehitetty ohjelmointiympäristö, joka on vakiinnuttanut asemansa esimerkiksi taidemusiikin kentällä. Esimerkiksi säveltäjä Kaija Saariaho on käyttänyt Max/MSP:tä töissään.

Max/MSP:n lisäksi keskustelu kääntyy myös Miller Puckettin kehittämään Pure Dataan, jota voidaan pitää Max/MSP:n ilmaisversiona. Max/MSP:ssä ja Pure Dataan on saman tyyppinen visuaalisen ohjelmoinnin rakenne eli ohjelmointi tapahtuu vetämällä viivoja laatikoiden välille. Laatikot voidaan ajatella funktioksi, jotka suorittavat jonkun pienen asian eli esimerkiksi laskevat lukuja yhteen tai laatikon avulla luku ohjaa ääntä. Pink Twins käytti Pure Dataa uransa alkuajankana, kunnes kokoonpano siirtyi kokonaan SuperColliderin käyttöön (Vehviläinen 2021).

Käytettiin tuota Pure Dataa ja Max/MSP:tä SuperColliderin rinnalla, mutta sitten niistä olen siirtynyt pois juuri sen takia, että sitten kun sulla on joku systeemi, joka on ollut vuosikausia käytössä ja se on vähän laajentunut, niin siihen palaaminen

uudestaan, niin että sitä kehittää alkaa olla aika hankalaa, kun visuaalinen ohjelmointi on vähän mitä on. (Vehviläinen 2021.)

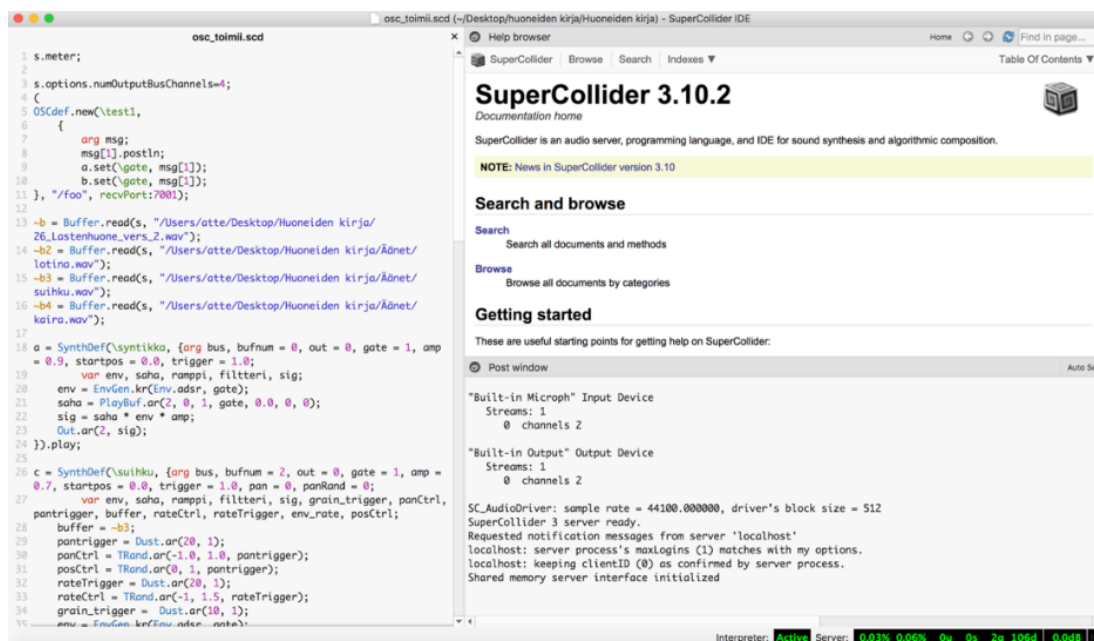
Max/MSP ja Pure Data tuntuvat usein toimivan porttiteorian tavoin matkalla äänikoodaamisen saloihin.

Jossain vaiheessa Max tuntu vähän silleen, että se on aika hankalaa välillä operoida sen kanssa. Se on aika hidasta ottaa se käyttöön ja tota mä sit mä olin kyllä kuullut SuperColliderista, mutta mä olin pari kertaa buutannut sen, mutta mä en tajunnut siitä vielä mitään. (Siren 2021.)

SuperCollider ei ole missään nimessä helppo kieli ja tästäkin syystä Max/MSP:n ja Pure Datan suosion voi ymmärtää hyvin. Jusu Vehviläinen on haastateltavista ainoa, joka on oppinut kielen itseopiskelun kautta. Vehviläinen kertoo ohjelmoineensa lapsesta saakka. SuperCollideriin tuntuu liittyvän vahvasti yhteisö ja tietynlainen mestari ja kisälli -ajattelu, jossa Sibelius Akatemiassa toimivalla Alejandro Olartella vaikuttaa olevan Suomessa mestarin rooli. Toki kielen oppimista ovat demokratisoineet esimerkiksi Eli Fieldsteelin nettitutoriaalit ja striimaukset, sekä SuperCollider-yhteisön järjestämät Zoom-tapaamiset. Tästä huolimatta SuperColliderin kotimainen käyttäjäyhteisö vaikuttaa muodostuneen lähinnä pääkaupunkiseudulle Olarten kurssien vaikutuksen ansiosta.

3.2 Käyttöliittymän puute

SuperColliderissa ei ole valmista graafista käyttöliittymää eli Graphical User Interfacea (GUI). SuperColliderin ohjelmointinäkyvä koostuu koodausikkunasta, Help Browserista ja Post Windowsta, mutta kyseessä on tekstipohjainen ohjelmointikieli, jossa ei ole valmiina graafista käyttöliittymää.



Kuva 3. Kuvassa SuperColliderin ohjelmointiympäristö, jossa näkyy Huoneiden kirja -suurteoksen Huoneeseen 26 tekemääni äänisuunnittelun koodia.

SuperCollideriin voi rakentaa käyttöliittymän. Tällöin käyttöliittymä on ohjelmitava itse. SuperCollideriin on itse asiassa mahdollisuus rakentaa erilaisia käyttöliittymiä, sillä ohjelmointiyhteisöt ovat kehittäneet niitä. Koutsomichalis (2013) esittää, että GUI-objektit ovat helppokäyttöisiä, mutta ne ovat kuitenkin erittäin muokattavissa ja erittäin tehokkaita. Haastateltavista kuitenkin kukaan ei ole innostunut graafisen käyttöliittymän suunnittelusta SuperColliderin sisään.

Tavallaan sen (SuperColliderin) käyttäminen on jäänyt vähemmälle, mutta kaikkea ihan mahtavaahan sillä pystyisi tekemään ja siinä on potentiaalia tehdä monta eri asiaa, mutta se käytäntö vaan... Ja sitten kun monesti, jos tekee jotain, niin siinä ne GUI:t ja muut vaihtoehdot on aika köpöjä, niin ei ainakaan asiakastarkoitukseen pysty tekemään. (Ihalainen 2021.)

Kirsi Ihalainen on tehnyt graafisen käyttöliittymän SuperColliderilla, mutta hän ei pidä ratkaisua toimivana. Jusu Vehviläinen on ratkaissut SuperColliderin käyttöliittymäongelman niin, että hän on Javalla kirjoittanut oman käyttöliittymän, jolla hän käyttää SuperCollideria.

Mä en käytä sen (SuperColliderin) käyttöliittymää, vaan mulla on erikseen sitten tehty käyttöliittymä, jolla ohjaan näitä kohtia, joten tavallaan se on tehokas

siinäkin kannalta, että se käyttää vaan sitä synteesipuolta, eikä ole tavallaan mitään graafista juttuja mikä hidastaisi konetta. (Vehviläinen 2021.)

Toisaalta ehkä juuri käyttöliittymän puuttuminen mahdollistaa SuperColliderin käytön tehokkuuden. Itse olen esimerkiksi käyttänyt SuperCollideria pienten Raspberry Pi Linux -tietokoneiden kanssa ja huomannut, että ohjelmat pyörivät vähilläkin tietokoneen tehoilla juuri siitä syystä, että graafista käyttöliittymää ei ole hidastamassa konetta.

3.3 SuperCollider ja DAW

Miten SuperCollideria sitten käytetään ilman graafista käyttöliittymää? Mahdollisuuksia on monia. SuperCollider on esimerkiksi Joonas Sirenin töissä käytössä ääni-installaatioissa, joissa SuperCollider generoi musiikkia tai ääntä.

Mulla vähän sekoittuu, että miten mä käytän SuperCollideria ja sitten mä nauhoitan koko ajan Abletonille. Mulla on välillä ainakin raitoja siellä, mutta livekeikkoja koetan rakentaa silleen, että mulla ei ole mitään muuta (kuin SuperCollider). (Siren 2021.)

Myös Ellen Virmanilla on Ableton Live käytössä SuperColliderin kanssa. Yleensä tekniikka toimii siten, että äänisynteesi ja samplemateriaali soitetaan SuperColliderilta, mutta Ableton Liveä tai jotain muuta DAWia (digital audio workstation) käytetään miksaukseen ja masterointiin. Äänikorteissa on myös mahdollisuus käyttää loopback-ominaisuutta, jossa äänidata kierrätetään ohjelmasta toiseen.

SuperCollideria haastatteluissa kuvataan nimenomaan audio toolkitiksi (Vehviläinen 2021) tai instrumentin rakennustyökaluksi (Virman 2021). Ohjelmointiympäristöstä puuttuvat perinteiset DAWille tyypilliset piirteet, joten ääniohjelmoijan täytyy suunnitella millä tavalla hän haluaa musiikkia tai äänitaidetta tehdä. Tämä mahdollistaa musiikin kokeellisuuden idean. Tanja Tiekso (2013, 71) esittää, että termi kokeellisuus oli 2000-luvun alussa etupäässä mediataiteilijoiden käyttämä termi, jolla viitattiin etenkin uusien ja vanhojen ääni- ja kuvatekniikoiden luovaan käyttöön osana taiteentekemisen

prosessia tai teoksia. Käytän termiä kokeellinen juuri edellä mainitulla tavalla. SuperCollider mahdollistaa kokeellisen musiikin ja äänitaiteen tekemisen siten, että kokeellisuus tarkoittaa uusien instrumenttien luomista taiteelliseen käyttötarkoitukseen. Koko prosessi on itsessään kokeellinen Tanja Tiekson kuvaamalla tavalla (2013, 71). Kokeellisuuden käsite on toki alati muutoksessa, mutta yksi helpoimpia tapoja hahmottaa kokeellisuutta on käsitellä uusia tekniikoita ja teknologioita osana taiteellista prosessia.

3.4 Synteesi vai samplermateriaali

Millaisia sitten ovat nämä instrumentit? SuperColliderin keinovalikoima voidaan jakaa karkeasti kahtia. Sillä voi tehdä äänisynteesiä eli yhdistelmällä ugeneiksi kutsuttuja äänigeneraattoreita voidaan saada aikaan esimerkiksi syntetisaattorilta kuulostavia ääniä. Tekniikka muistuttaa modulaarisyntetisaattoreita, joita käytetään niin, että erilaisia ääntä ohjaavia tai ääntä synnyttäviä moduuleja yhdistetään johtojen avulla. SuperColliderissa ei ole johtoja, eikä muutenkaan mitään muuta kuin koodia.

```
(
SynthDef("sine", { arg freq=800;
  var osc;
  osc = SinOsc.ar(freq, 0, 0.1); // 800 Hz sine oscillator
  Out.ar(0, osc); // send output to audio bus zero.
}).add;
)
```

Kuva 4. Esimerkki SuperCollider-tutoriaalin koodista.

Esimerkkinä olevassa koodissa luodaan 800 hertzin sin-aalto, joka lähetään stereona äänikortille. SuperCollider luo Out.ar:n arvolla nolla stereoparin automaattisesti.

Vastaavanlaisia rivejä yhdistelemällä saadaan aikaan monimutkaisia äänikudoksia. SuperColliderin etuna on äänidatan monistettavuus. Sillä pystyy

vähäisellä koodimäärällä luomaan tekstuureja, joissa satoja ääniä soi päällekkäin muodostaen äänellisen kudoksen.

Toinen tapa käyttää SuperCollideria on käyttää sampleja eli ääninäytteitä. Samplet voivat olla mitä tahansa nauhoitettuja ääniä instrumenteista erilaisiin äänilähteisiin. Yksi SuperColliderilla suosittu tekniikka on käyttää niin sanottua granulaarisynteesiä. Granulaarisynteesissä samplesta luetaan pieniä pätkiä, jotka muodostavat grain-objekteja eli eräänlaisia äänirakeita. Yksi äänirae toimii ääniobjektien rakennuspalikkana. Yhdistämällä tuhansia rakeita pystytään saamaan aikaan hälyefekti, joka muistuttaa hieman esimerkiksi sateen ääntä. Yksittäinen rae sisältää aika-alueen tiedot (alkamisaika, kesto ja verhokäyrän muoto), sekä taajuusalueen tiedot (rakeen sisällä olevan aaltomuodon korkeus ja rakeen spektri). (Roads 2002, 87.)

Sampleja pystytään tietenkin myös pitchaamaan eli sen sävelkorkeutta pystytään muuttamaan. Lukuisia sampleja voidaan myös soittaa päällekkäin, jolloin saadaan aikaan monipuolisia äänitekstuureja. Kysyttäessä Ellen Virmanilta, että käyttääkö hän enemmän synteesi- vai samplemateriaalia, hän sanoo käyttävänsä musiikissaan molempia.

Tosi paljon just synteesin keinoilla oon tehnyt, mutta välillä on jotain samplehommia kyllä joo... (Virman 2021).

Synteesi ja samplepuolen erottaminen toisistaan ei vaikuta olevan haastateltavista merkittävää, koska äänen ohjelmointiprosessi on melko samantyyppinen kummassakin tapauksessa. SuperColliderissa ugenejä voidaan myös kytkeä kontrollisignaaleiksi, joten äänen ohjelmalliset kytkentämahdollisuudet ovat monipuoliset.

4 SuperCollider-kielen tulevaisuus

SuperColliderin tulevaisuudesta haastateltavilla on erilaisia näkemyksiä. Jusu Vehviläinen povaa SuperColliderille pitkää tulevaisuutta, koska SuperColliderin yhteensopivuus taaksepäin on suhteellisen vahva verrattuna esimerkiksi Javaan, jonka Vehviläinen kertoo vertailukohdaksi (Vehviläinen 2021). Vastakkaista näkemystä taas edustaa Kirsi Ihalainen, joka ei usko SuperCollider kielen menestykseen.

Minä luulen, että SuperCollider ei enää pysy tavallaan kehityksen kärjessä mukana, koska kaupalliset instanssit alkaa jo implementoimaan omia ja tavallaan tuommoisia juttuja sitten siihen niiden omaan systeemiin kiinni. Ihmiset tekee sitä (SuperCollideria) omalla vapaa ajallaan ja ilmaiseksi, niin se voi olla, että se jossain kohtaa jopa katoaa. Varsinkin sitten, jos yliopistot luopuu sen opettamisesta. (Ihalainen 2021.)

Ihalaisen pelko on aiheellinen, sillä ainakin Joonas Sirenin (2021) mukaan SuperColliderin kehitystiimin ongelmana on se, että kehitystiimiläisiä palkataan isoihin yrityksiin, kuten Native Instrumentsille ja vapaa-ajalla tehty SuperColliderin kehittäminen jää vähemmälle.

Mä vaan toivon, että ei käy silleen, että yhtäkkiä ei olisi ketään, joka olisi kehittämässä sitä. Et se vaan kuihtuisi se development skene, koska jengihän tekee sitä ilman mitään palkkaa. Sitä tekee ihmiset silleen vaan omaa hyvyttään. (Siren 2021.)

Tätä kehitystä on tapahtunut SuperColliderin alkumetreiltä asti, kun kielen alkuperäinen kehittäjä James McCartney palkattiin Appllelle ja SuperCollider muutettiin ilmaiseksi ohjelmointikieleksi. Toisaalta kielen kehitystyö on pysynyt elinvoimaisena ja SuperCollideriin on saatu viime vuosina merkittäviä uusia ominaisuuksia, kuten esimerkiksi tuki VST:lle (Virtual Studio Technology). Siren kertoo keskustelleensa VST-tuen kehittäjän kanssa ja SuperColliderin nettiyhteisöllä on ollut Zoom-tapaamisia, jossa kehittäjät ja kielen käyttäjät ovat keskustelleet keskenään. SuperColliderin kehittäjä- ja käyttäjäyhteisö vaikuttaa Siren haastattelun perusteelta olevan aktiivinen ja toisaalta melko vapaamuotoinen. Tämä tarkoittaa, että kehitysideat liikkuvat käyttäjiltä kehittäjille ja haluttuja muutoksia on mahdollisuus tehdä. Toisaalta taas

SuperColliderin tuki on puhtaasti vapaamuotoisen käyttäjä- ja kehittäjäyhteisön varassa. Jos ohjelmointiympäristön kanssa on ongelmia, niin ei ole mitään kaupallista tai institutionaalista instanssia, jonka puoleen voi kääntyä. Ominaisuuden puuttuessa käyttäjillä on mahdollista ryhtyä toimiin ja koodata haluttu ominaisuus itse tai odotella, että joku tekee ominaisuuden.

4.1 SuperColliderin käyttäjäyhteisö

SuperColliderilla on aktiivinen kansainvälinen käyttäjäyhteisö. Siren ja Virman kuuluvat Pan-Nordic Computer Music Networkiin, jonka tarkoitus oli järjestää jäsenilleen keikkoja eri Pohjoismaissa. Sitten tuli korona ja sotki suunnitelmat, mutta yhteisö elää ainakin netissä. (Virman 2021.)

Joonas Siren (2021) puhuu haastattelussaan ”DSP-guruista” (digital signal processing). Siren on aktiivinen eri sosiaalisen median kanavissa, kuten Telegramissa, Discordissa ja Facebookissa. ”DSP-guruilla” Siren tarkoittaa ihmisiä, jotka hallitsevat digitaalisen signaalikäsittelyn niin pitkälle, että he pystyvät kehittämään ääniohjelmointiympäristöjä. Tämän rinnalla Siren (2021) puhuu SuperColliderin käyttäjistä, jotka osaavat koodata SuperCollidieria, mutta eivät kehitä sitä. Paras kehitystyö toki tapahtuu, jos nämä kaksi osapuolta ovat vuorovaikutuksessa keskenään. Usein ”DSP-gurutkin” ovat taiteilijoita ja Siren (2021) kertoikin hauskan anekdootin Miller Puckettesta, Max/MSP:n ja Pure Datan kehittäjästä, jonka keikan Siren oli nähnyt. Keikalla Puckette soitti midikitaraa, johon hän oli loihtinut DSP-taidoillaan täydellisen blues-kitarasoundin.

Ellen Virmanin musiikillinen työ SuperColliderin parissa Eilien-nimellä antaa uskoa siihen, että SuperCollider saa myös uusia käyttäjiä. Eli Fieldsteelin YouTube-tutoriaalit ovat madaltaneet kynnyistä tarttua kieleen ja helpottanut sen opettelua suunnattomasti. Se tieto, joka ennen oli lähinnä korkeakoulujen kursseilla, on nykyisin kenen tahansa tavoitettavissa. Ilmeistä kuitenkin on, että SuperCollider tarvitsee opettajia. Ilman vankkaa koodaustaustaa ohjelmointikieltä on vaikea opetella itse.

4.2 SuperColliderin integrointi muihin järjestelmiin

Yksi Kirsi Ihalaisen (2021) esittämistä uhkakuvista koskien SuperCollider-kielen tulevaisuutta liittyy siihen, että kaupalliset instanssit alkavat implementoimaan omia SuperColliderin kaltaisia liitännäisiä (engl. plugin). Hän kertoo esimerkiksi pelimoottoreissa käytettävästä Firelight Technologiesin kehittämästä FMOD Studio -työkalusta, jossa on käytössä SuperColliderin kaltaisia generatiivisia ääniratkaisuja (Ihalainen 2021). Myös Vehviläinen (2021) mainitsee, että hän on törmännyt ohjelmointityökaluihin, joissa näkee SuperColliderin vaikutusta. SuperColliderin vahvuus on vuosia ollut algoritmisäveltäminen ja generatiiviset äänijärjestelmät. Nyt nämä samat ominaisuudet tulevat osaksi erilaisia digitaalisia äänijärjestelmiä.

Yksi ääniohjelmoinnin alueita ovat pelilliset ratkaisut. Itse opetan peliohjelmointia ja käytän Unity-pelimoottoria. Joonas Siren (2021) kertoo käyttävänsä taiteellisessa työssään Unreal Engineä. Myös Kirsi Ihalaisella (2021) on kokemusta pelimusiikin tekemisestä esimerkiksi Remedy-yhtiössä. SuperCollider olisi ihanteellinen työkalu peliäänen tekemiseen, jos se olisi sisällytettävissä suoraan pelimoottoreihin, kuten FMOD Studio. Remedyllä on ollut tavoitteenaan hyödyntää SuperCollidieria omassa pelimoottorissaan (Ihalainen 2021), mutta SuperColliderin käytöstä Remedyllä tällä hetkellä ei ole varmaa tietoa.

Itse olen käyttänyt SuperCollidieria *Huoneiden kirja* -suurteoksessa Huoneessa 26, jonka suunnitteli Maija Westerholm. CTRL-yhtiö teki huoneeseen ARjärjestelmän (augmented reality), jonka äänet toteutin SuperColliderilla. Huoneessa oli langaton verkko, jonka kautta Unity-pelimoottori lähetti käskyjä OSC-protokollan (open sound control) kautta Raspberry PI -tietokoneella toimivaan SuperCollideriin, joka tuotti äänen kaiuttimiin äänikortin kautta.

Immersiiviset taideprojektit ja pelikehitys ovat tulevaisuuden oivia käyttöalustoja SuperColliderille, mutta SuperColliderin integroiminen eri järjestelmiin hidastaa kehitystä. Kirsi Ihalainen (2021) suositteleekin SuperColliderin käytön sijaan peliäänen tekemistä FMOD Studiolla.

Näen SuperColliderin käyttöalueen tulevaisuudessakin kokeellisen musiikissa ja erilaisissa installaatioissa. Yrityskontekstissa tilanne on toinen. SuperColliderilla on vaikea saada aikaan myytäviä tuotteita, vaikka esimerkiksi Monomen kehittämä kaupallinen Norns-syntetisaattori käyttääkin SuperCollideria. Kirsi Ihalainen (2021) kertoo, että hän on yrittänyt tehdä valmiita äänituotteita SuperColliderilla siinä onnistumatta. Jos SuperCollideria haluaa käyttää yrityskontekstissa, niin törmää väistämättä ongelmiin.

SuperCollider ei taivu kaupallisen tuotteen toteutukseen, jolloin se ei myöskään näy rahassa ja se on vain kylmä realiteetti (Ihalainen 2021.)

Toisaalta eipä SuperColliderin käyttö mitään maksakaan. Ilmaisella avoimen lähdekoodin ääniohjelmointikielellä on oma käyttäjäkuntansa, joka mahdollistaa esimerkiksi livekoodauksen kaltaisten ilmiöiden muodostumisen. Kieli on edelleen hyödyllinen taideinstallaatioiden teossa sekä kokeellisen musiikin instrumentintekovälineenä. Paremmilla integrointimahdollisuuksilla SuperCollider avasi peliänten tekoon aivan uudenlaisia mahdollisuuksia. Nyt ohjelmointiympäristö jää puhtaasti taidekäyttöön, eikä laajenemismahdollisuuksia kontekstin ulkopuolelle juuri ole. Vehviläinen (2021) tosin kertoo projektista, jossa robotteja on ohjattu SuperColliderin avulla, joten ainakin periaatteessa SuperCollider toimii myös muuhun kuin ääniohjelmointiin. Ihminen on luova laji myös teknologian väärinkäyttäjänä ja kuka tietää tullaanko SuperColliderkin näkemään jossain muussa kuin taidekäytössä.

5 Lopuksi

Olen tutkimuksessani kysynyt miten äänitaiteilijat ja kokeellisen musiikin tekijät käyttävät SuperCollideria taiteellisen työnsä tukena. Haastatteluissa on tullut ilmi, että käyttötapoja on ollut useita ja että SuperCollider nähdään etupäässä digitaalisten instrumenttien rakennustyökaluna sekä soittimena. SuperCollideria voi käyttää lukuisilla tavoilla. Sillä voi tehdä valmiita kappaleita tai esimerkiksi generoituvaa musiikkia. SuperCollider on livekäytössä tehokas työkalu ja se mahdollistaa esimerkiksi livekoodauksen.

Aineistoni perusteella SuperCollider on niche-kieli, jolla on oma tiukasti rajautunut käyttäjäkuntansa. Tämä käyttäjäkunta kokoontuu sosiaalisen median kanavissa ja tapaa toisiaan keikoilla. Taiteen kokija taas voi törmätä SuperColliderin käyttöön erilaisissa installaatioissa ja elektronisen musiikin keikoilla. SuperCollider on ilmainen ja vapaan lähdekoodin kieli, joten sen pystyy asentamaan useille erilaiselle alustoille. Se toimii Windowsissa, Mac-koneissa ja Linux-tietokoneissa. SuperCollideria on vaikea integroida muihin kieliin tai järjestelmiin ilman välissä olevaa protokollaa, kuten OSCia (Open Sound Control).

Kulttuurihistoriallisesti kotimainen SuperColliderin käyttö jatkaa kehitystyötä, joka on ollut käynnissä suomalaisen elektronisen musiikin varhaisvaiheista lähtien. SuperColliderin käytöllä on selkeää hengenheimolaisuutta esimerkiksi Erkki Kurenniemen pioneerityöhön digitaalisten instrumenttien saralla. Toisaalta, koska SuperCollider-kielen käyttö rajautuu kokeelliseen musiikkiin, niin kielellä on roolinsa tämänhetkisessä avantgarde- ja undergroundkulttuurissa.

SuperCollider ei ole kieli, jolla pääsee käsiksi kaupalliseen menestykseen. Se on enemmänkin jossain katvealueella toimiva oma järjestelmänsä, jota pyöritetään vapaaehtoisvoimin ja vapaa-aikana. Se ei ole myöskään kieli, jonka rakenteita voisi hyödyntää muissa ohjelmointikielissä. SuperColliderin käyttö rajautuu tiukasti taidekontekstiin, mutta toisaalta juuri taideinstallaatioiden tekemiseen ja kokeelliseen musiikkiin SuperCollider erityisesti sopii.

Mihin sitten SuperColliderin käyttö johtaa? SuperColliderin kehittäjä James McCartney päätyi Applelle tekemään CoreAudio-arkkitehtuuria Mac OSX:lle ja iOS:lle. SuperColliderin kehittäjätiimistä on palkattu ihmisiä esimerkiksi Native Instrumentsille. Kirsi Ihalainen yritti käyttää SuperCollidieria äänituotteiden tekoon, mutta päätyi kehittämään omat äänijärjestelmänsä. Vaikuttaa siltä, että SuperCollider on merkittävä työkalu ihmisten urapolulla. Se ei palvele kaikkia tarkoituksia, mutta sitä opetelleet ihmiset ovat päässeet urallaan eteenpäin. SuperColliderin opettelu on kivinen tie, eikä tarjolla ole pikavoittoja tai taloudellista hyötyä, mutta ehkä sen voima on siinä, että sen voi jokainen asentaa sen ilmaiseksi koneelleen ja se tarjoaa reitin lausekielisen ääniohjelmoinnin maailmaan.

SuperCollidieria on tutkittu varsin vähän ja se niputetaan usein muiden ääniohjelmointikielten joukkoon. Akateeminen kiinnostus suomalaista elektronista musiikkia kohtaan on ollut viime vuosina nousussa, mutta se on keskittynyt pääasiassa elektronisen musiikin varhaisvaiheisiin. Ohjelmointikieli instrumentin tekovälineenä ja soittimena tarjoaa oivat jatkotutkimusmahdollisuudet. Itse toivoisin, että varsinkin ohjelmoinnin kokemuksellista puolta tutkittaisiin enemmän. Haastatteluja läpikäydessäni kiinnitin erityisesti huomiota Ellen Virmanin kuvaukseen ääniohjelmoinnin meditatiivisesta voimasta.

Taiteellisessa käytössä SuperCollider tuottaa tietynlaista estetiikkaa. Pink Twinsin, Eilienin ja Forcesin tarjoama musiikki on ääninväriltään moniulotteista ja generatiivista. Ihalaisen äänitaideteokset hyödyntävät luonnon ääniä teknologisin keinoin. Eri DAWit (digital audio workstation), ääniohjelmointikieliet ja instrumentit tuottavat erilaista estetiikkaa ja SuperColliderista on muodostunut yksi kokeellisen musiikin tärkeimmistä työkaluista. Haastateltavat kutsuvat SuperCollidieria instrumentin rakennustyökaluksi ja sitä se totisesti on. SuperColliderilla on mahdollisuus päästä taiteellisesti omaperäisiin lopputuloksiin, sillä jokainen sillä tehty ohjelma on tehty ääniohjelmoitsijan omista lähtökohdista ja on siten erilainen. SuperCollider ei myöskään tarvitse toimiakseen valtavia ohjelmointitiimejä, vaan se mahdollistaa tietynlaisen

auteur-tekijyyden, jossa tekijä koodaa itse instrumenttinsa, esittää instrumentilla tuotoksensa ja usein vielä miksaa teoksensa. Masteroinnin usein tekee joku ulkopuolinen.

SuperColliderilla on oma käyttäjäyhteisönsä ja yleisönsä, joka sitä vaalii, mutta siitä tuskin tulee kaupallisessa käytössä olevaa teollisuusstandardia. Se on kokeelliseen musiikkiin ja äänitaitteeseen suunnattu ohjelmointikieli ja sellaisenaan arvokas.

Lähteet

Algorave Helsinki: <https://www.facebook.com/algoravehelsinki/> (Viitattu 31.10.2021)

Greher, GR, & Heines. JM 2014. Computational Thinking in Sound: Teaching the Art and Science of Music and Technology, Oxford University Press, Incorporated, New York. Available from: ProQuest Ebook Central. [Viitattu 31.10.2021].

Elsea, P. 2013. The Art and Technique of Electroacoustic Music, A-R Editions, Inc., Middleton (WI). Available from: ProQuest Ebook Central. [Viitattu 6.4.2021].

Galanter P. 2016. Generative Art Theory, Paul, C (ed.), A Companion to Digital Art, John Wiley & Sons, Incorporated, Hoboken. Available from: ProQuest Ebook Central. [Viitattu 6.4.2021].

Higgins, D. 2001 [1965]. Intermedia. Leonardo 34 (1): 49–54.
<https://doi.org/10.1162/002409401300052514> Viitattu 31.10.2021.

Hugill, A. 2012. The Digital Musician, Taylor & Francis Group. ProQuest Ebook Central [Viitattu 29.10.2021].

Koutsomichalis, M. 2013. Mapping and Visualization with SuperCollider, Packt Publishing, Limited, Birmingham. Available from: ProQuest Ebook Central. [Viitattu 31.10.2021].

Phillips, W. 2014. A Composer's Guide to Game Music, MIT Press, Cambridge. Available from: ProQuest Ebook Central. [Viitattu 6.4.2021].

Rice, T. 2008. Towards a Mediation of Field Methods and Field Experience. Shadows in the Field – New Perspectives for Fieldwork in Ethnomusicology 2. painos. Toim. Gregory Barz ja Timothy Cooley. Oxford: Oxford University Press.

Roads, C. 2002. Microsound, MIT Press, Cambridge. Available from: ProQuest Ebook Central. [Viitattu 31.10.2021].

Russolo, L. 2018 [1916]. Hälyjen taide. Suom. Tanja Tiekso. Helsinki: Tutkijaliitto.

Steno: <https://github.com/musikinformatik/Steno> (Viitattu 31.10.2021)

SuperCollider 3 Server Tutorial: <http://doc.sccode.org/Tutorials/Tutorial.html>
(Viitattu 31.10.2021)

Kaisler, S. 2005. Software Paradigms, John Wiley & Sons, Incorporated.
ProQuest Ebook Central. [Viitattu 31.10.2021].

Tiekso, T. 2013. Todellista musiikkia: Kokeellisuuden idea musiikin
avantgardemanifesteissa. Helsinki: Osuuskunta Poesia.

HAASTATTELUT:

Ihalainen, Kirsi 2021. Haastattelu. Äänitaiteilija ja yrittäjä. Haastatteli Zoom-
etäyhteydellä 9.9.2021 Atte Häkkinen.

Siren, Joonas, 2021. Haastattelu. Mediataiteilija. Haastatteli Zoom-
etäyhteydellä 17.6.2021 Atte Häkkinen.

Vehviläinen, Jusu 2021. Haastattelu. Mediataiteilija. Haastatteli Zoom-
etäyhteydellä 3.9.2021 Atte Häkkinen.

Virman, Ellen, 2021. Haastattelu. Valo- ja äänisuunnittelija. Haastatteli Zoom-
etäyhteydellä 20.8.2021 Atte Häkkinen.