

# **Tapaustutkimus: Ohjelmistolisenssien ominaisuuksien vertailu**

Jaana Partanen

Tampereen yliopisto  
Tietojenkäsittelytieteiden laitos  
Tietojenkäsittelyoppi /Ohjelmistokehitys  
Pro gradu -tutkielma  
Ohjaaja: Pirkko Nykänen  
Kesäkuu 2009

Tampereen yliopisto  
Tietojenkäsittelytieteiden laitos  
Tietojenkäsittelyoppi /Ohjelmistokehitys  
Jaana Partanen: Tapaustutkimus: Ohjelmistolisenssien ominaisuuksien vertailu  
Pro gradu -tutkielma, 82 sivua  
Kesäkuu 2009

---

Lisenssin valitseminen tuotteistettavalle ohjelmistolle on tarkkuutta vaativaa työtä. Ensin tulee perehtyä mahdollisimman moniin erilaisiin lisensointimalleihin tai -tyyppeihin. Pitää pohtia erilaisten mallien vaikutusta liiketoiminnalle sekä niiden vaatimia resursseja ja velvoitteita sekä myyjän että asiakkaan näkökulmista. Kun tilanteen kannalta parhaiksi koettut vaihtoehdot on rajattu muutamaaan, voidaan suorittaa lopullinen vertailu näiden kesken.

Hyvä apu valittaessa parasta vaihtoehtoa on käyttää helposti omaksuttavaa vertailumenetelmää, jonka tulokset ovat selkeästi luettavia. Tällöin väärinymmärrysten määrä ja virheellisen tulkinnan mahdollisuudet ovat mahdollisimman vähäisiä ja tulosten perusteella tehdyt ratkaisut mahdollisimman luotettavia ja oikeita.

Yllämainitun kaltainen arviointimenetelmä on tässä työssä käytettävä The Composite Features Diagram -menetelmää, jonka perusajatuksena on pilkkoa mitattavat ominaisuudet sellaisiksi osiksi, että niitä pystytään mittaamaan määrällisesti. Kaikki valitut kohteet testataan näiden perusominaisuuksien osalta ja näin saatujen arvojen perusteella voidaan summata yhteen lopulliset arvosanat kultakin verrattavalta kohteelta Kiviatiagrammiin. Saadut tulokset ovat täysin vertailtavissa keskenään ja helposti ymmärrettävässä kaaviomuodossa, jolloin päätelmien tekeminen on helppoa, varmaa ja luotettavaa.

Tässä tutkielmassa vertaillaan erilaisia sekä toimittajalle että asiakkaalle tärkeitä lisensointiin liittyviä ominaisuuksia. Testaaminen tapahtui vertaamalla kahta eri ääripään ratkaisuvaihtoehtoa keskivertovaihtoehtoon ja tutkimalla, miten näiden saamat tulokset eroavat toisistaan. Keskivälin vaihtoehdoksi pyrittiin valitsemaan kohtuullisin ja suositeltavin toimintamalli kahden ääripään väliltä.

Vertailussa saatiin selville, että vaikka ääripään vaihtoehdot saivatkin loistavia tuloksia joistakin ominaisuuksista, niin niiden saamat tulokset eivät olleet lainkaan hyväksyttävissä joidenkin ominaisuuksien kohdalla. Ainoa vertailusta selvinnyt kohde oli keskivälin vaihtoehto, joka myös sai erittäin tyydyttävät loppupisteet.

Avainsanat ja -sanonnat: lisenssi, arviointi, The Composite Features Diagram, Kiviatiagrammi

## Sisällys

1.	Johdanto.....	1
2.	Yleisesti lisensoinnista.....	2
2.1.	Lisenssimaksut.....	2
2.2.	Lisenssiterminologiaa.....	3
2.3.	Lisensoinnin hyödyt ja haitat.....	3
2.4.	Tekijänoikeus.....	5
2.5.	Lisenssisopimus.....	6
3.	Tutkimuksia lisensointiin liittyen.....	8
3.1.	Kahden periodin -malli ohjelmistojen jakamiseen.....	8
3.2.	Lisensoinnin vaikutus yrityksiin ja verkostoihin.....	8
3.3.	Ohjelmistojen muuttuva hinnoittelu.....	9
3.4.	Ohjelmistojen jakamiseen liittyvää ongelmatiikkaa.....	10
4.	Lisensointimalleja.....	14
4.1.	Perinteiset myyjä-asiakas -mallit.....	17
4.1.1.	Yhden käyttäjän lisenssi.....	17
4.1.2.	Monen käyttäjän jaettu lisenssi.....	17
4.1.3.	Aikarajoitteinen lisenssi.....	18
4.2.	Luokittelumallit.....	18
4.3.	Sekoitetut lisensointimallit.....	19
4.4.	Teknologiakumppanuus.....	20
4.5.	Avoimen lähdekoodin lisenssejä.....	20
5.	Tapaustutkimuksen lähtökohdat.....	23
5.1.	Myyjäyrittäjä.....	23
5.2.	Asiakasprofiili.....	23
5.3.	Ohjelmiston luonnehdinta.....	24
6.	Lisenssiltä vaadittavia ominaisuuksia.....	26
6.1.	Toimittajan taloudellisia kysymyksiä.....	26
6.1.1.	Miten varmistetaan maksimituotot?.....	26
6.1.2.	Miten minimoidaan kulut?.....	27
6.1.3.	Miten varmistetaan asiakassuhteen pysyvyys?.....	27
6.1.4.	Miten järjestetään myyntikanava tai useat myyntikanavat?.....	28
6.1.5.	Miten varmistetaan lisenssiehtojen joustavuus?.....	28
6.1.6.	Miten tehdään kokoonpanonhallinta kustannustehokkaasti?.....	29
6.2.	Ylläpitoon ja kokoonpanonhallintaan liittyviä kysymyksiä.....	29
6.2.1.	Miten käytännössä hoidetaan kokoonpanonhallinta?.....	29
6.2.2.	Miten hallitaan eri asiakkaiden ostamat lisenssit?.....	30

6.2.3.	Miten lisensoitu ohjelmisto poistetaan käytöstä? .....	31
6.2.4.	Miten hoidetaan versionnostojen aiheuttama työ?.....	31
6.2.5.	Miten saadaan henkilöresurssit vastamaan tarvetta?.....	32
6.3.	Lisensointikysymyksiä asiakkaan näkökulmasta .....	32
6.3.1.	Miten hinta jakaantuu ohjelmiston elinkaaren ajalle?.....	33
6.3.2.	Miten ennakoitavissa henkilöresurssien tarve on?.....	33
6.3.3.	Miten hoidetaan versionnostojen aiheuttama työ?.....	34
6.3.4.	Miten toimivat ylläpito ja tukipalvelut käytännössä?.....	35
7.	The Composite Features Diagram -arviointimenetelmä .....	36
7.1.	Menetelmän kuvaus.....	36
8.	Vertailuun valittavat ominaisuudet ja niiden ositus .....	42
8.1.	Kustannuksiin liittyvät pääominaisuudet ja niiden ositus .....	42
8.2.	Ylläpitoon liittyvät pääominaisuudet ja niiden ositus.....	44
8.3.	Pääominaisuudet asiakkaan näkökulmasta.....	47
9.	Vertailun toteutus .....	49
9.1.	Testitapaukset.....	49
9.2.	Asteikot .....	63
9.3.	Raja-arvot.....	69
10.	Testaustulokset, niiden arviointi ja vertailu .....	70
10.1.	Vaihtoehto A.....	70
10.2.	Vaihtoehto B .....	73
10.3.	Vaihtoehto C .....	75
11.	Johtopäätökset .....	79
	Viiteluettelo .....	81

## 1. Johdanto

Tässäpro gradu -tutkielmassa kuvataan lisenssien tärkeimpiä ominaisuuksia, perehdytään aiheesta tehtyihin tutkimuksiin ja vertaillaan lisenssiehtojen erilaisten muunnosten vaikutuksia muun muassa lisenssistä saataviin tuottoihin.

Vertailu on erittäin hyödyllistä, koska lisenssiehtoja sovittaessa on osattava tehdä kannattavimmat ja molempia osapuolia hyödyttävät ratkaisut. Ehtojen on taattava kattavat tuotot myös pitkällä aikavälillä. Sopimusehtojen määrittelemillä toimenpiteillä on suuria vaikutuksia sekä syntyviin kuluihin että saataviin tuottoihin. Myös sopimusehtojen muuttaminen myöhemmin on resursseja kuluttava ja pitkä prosessi. Siksi on erittäin tärkeää, että tehdään oikeita valintoja heti prosessin alusta lähtien.

Ensimmäisessä luvussa kuvataan tutkielman aihe, rakenne ja työn tekemisprosessi. Luvussa kaksi kuvataan lisenssejä ja niihin vaikuttavia seikkoja yleisesti. Kappaleessa kolme kerrotaan muutamaan mielenkiintoisimpaan aiheesta tehtyyn tutkimukseen liittyviä keskeisiä seikkoja. Luvussa neljä on lyhyesti kuvattu erilaisia lisensointimalleja ja niiden evoluutiota yhden käyttäjän lisenssistä kohti teknologiakumppanuutta. Viidennessä luvussa kuvataan tämän tapaustutkimuksen tausta ja tilanne eli kuvataan myyjää, asiakasta sekä ohjelmistoa, jonka tarpeisiin lisenssiä ollaan valitsemassa. Luvussa kuusi on eritelty lisenssiltä vaadittavia ominaisuuksia ja kuvattu sekä perusteltu näitä. Luvussa seitsemän esitellään tutkielmassa käytettävä vertailumetodi, The Composite Features Diagram -menetelmä (CFD). Kahdeksannessa luvussa kerrotaan vertailuun käytettävistä ominaisuuksista sekä jaotellaan ne vertailun kannalta tarkoituksenmukaisiin alueisiin. Luvussa yhdeksän kuvataan vertailussa käytettävät testitapaukset, niiden arvosteluasteikko sekä käytettävät raja-arvot. Kymmenennessä luvussa kuvataan testauksen tuloksia, ja lopuksi luvussa yksitoista kerrotaan vertailun johtopäätökset.

Luvuissa 2, 3 ja 4 käytetyt termit olen suurimmaksi osaksi suomentanut itse. Lisäksi olen käyttänyt termejä toimittaja ja myyjä toistensa synonyymeinä.

## 2. Yleisesti lisensoinnista

Lisensointi on menettely, jolla keksintö, ohjelmisto, teos tai muu henkisen työn tulos voidaan tuotteistaa. Lisenssin myöntämisellä annetaan jollekin toiselle yritykselle oikeus valmistaa, käyttää ja/tai myydä lisensoitua tuotetta. Lisenssin kohteena on useimmiten käyttöoikeus, kuten patenti tai tavaramerkki, mutta myös liikesalaisuus ja tietotaito voidaan lisensoida [Pelkonen, 2003].

Teollisen kehityksen mukanaan tuoma massatuotanto on tehnyt tarpeelliseksi sen, että tekijänoikeutta suojataan eli kyse on eräänlaisesta palkitsemismenettelystä, jolla luovan työn tekijä palkitaan yksinoikeudella, joka kattaa sekä immateriaali- että taloudelliset oikeudet, tuotokseensa tiettyssä maassa ja tietyn mittaisen ajanjakson verran. [Mansala, 2001]

### 2.1. Lisenssimaksut

Lisensoinnissa lisenssinostajan maksamaa korvausta kutsutaan lisenssimaksuksi (royalty), joka yleisesti ottaen voidaan maksaa neljällä eri tavalla. Joskus lisenssin koko hinta maksetaan etukäteen joko kerralla tai osissa, joskus maksetaan aluksi nk. kynnysraha, joka kattaa lisenssinantajalle jo aiheutuneet kustannukset ja lisäksi maksetaan myös vuosittaisia maksuja. Yleisimmin maksetaan vuosittaisia maksuja joko kynnysrahalla tai ilman. [Korkiakangas, 1998]

Lisenssimaksun suuruuteen vaikuttaa yleensä lisensoitavan kohteen käytön laajuus, esimerkiksi yhtäaikaisten käyttäjien lukumäärä. Kynnysmaksu taas on toisaalta maksu, joka on maksettava riippumatta esimerkiksi yhtäaikaisten käyttäjien määrästä, mutta toisaalta se voi myös näin sovittaessa antaa oikeuden tiettyyn käyttäjien määrään. [Pelkonen, 2003]

Etukäteen maksettavia lisenssimaksuja käytetään yleensä vain sellaisissa tilanteissa, joissa ei voida olla varmoja lopullisesta lisenssinostajasta tai ostajan maksukyvyistä ja halutaan varmistaa voitot. Lisenssisopimus solmitaan usein pidemmäksi aikaa, jopa 10 – 20 vuodeksi kerrallaan. [Korkiakangas, 1998]

Jos lisenssimaksu maksetaan vain kerran ja se oikeuttaa käyttämään ohjelmaa ikuisesti, puhutaan jatkuvasta lisenssistä (perpetual license). Tällöin ohjelman päivittäminen ja tukipalvelut maksavat erikseen. Jos taas lisenssisopimuksessa rajoitetaan ohjelman käyttöä esimerkiksi vain arviointia tai koulutusta varten, puhutaan rajoitetusta lisenssistä (restricted license). Lisenssisopimusten katoavaa mallia ovat paikallislisenssit (site license), jotka sallivat ohjelmiston rajoittamattoman käytön yhdessä tiettyssä paikassa. [Tynan, 2004]

## 2.2. Lisenssiterminologiaa

Kausilisenssi (term license) (tai ohjelmiston vuokraus (software subscription)) on sopimus, joka solmitaan yleensä 1 – 5 vuodeksi kerrallaan. Sopimus sisältää käytönaikaisen tuen, ohjelmistopäivitykset ja ylläpidon. Kauden päättyessä asiakkaan on joko uusittava sopimus tai poistettava ohjelmisto käytöstä. [Cusumano, 2007; Hämäläinen, 2003; Tynan, 2004]

Kaikkein laajin lisenssityyppi on rajoittamaton lisenssi (unrestricted license), joka sallii lisenssinostajan edelleen kehittää ohjelmistoa, ottaa siitä kopioita ja muokata sitä yhteensopivaksi myös muiden ohjelmistojen kanssa. [Tynan, 2004]

Ohjelmistolisenssien kohdalla puhutaan yleisesti tuotelisenssisopimuksista. Tällaisessa sopimuksessa sovitaan yleensä tuotteen osan, jonkin komponentin, koko tuotteen tai jopa kokonaisen tuoteperheen käyttöoikeudesta. Useimmiten tuotelisenssit myönnetään nk. eksklusiivisina eli yksinomaisena tai valikoivana (exclusive licence), jolloin lisenssin saajalla on oikeus käyttää tuotetta jollakin tietyllä maantieteellisellä alueella. Tällöin lisenssisopimukseen lisätään usein myös kohta, ettei lisenssinantajakaan myy enempää lisenssejä kyseiselle maantieteelliselle alueelle [Classen, 2007; Korkiakangas, 1998; Pelkonen, 2003].

On tärkeää huomata, että jos lisenssinantaja voi itsekään käyttää tuotetta kyseisellä maantieteellisellä alueella, puhutaan yksinkertaisesta lisenssistä (sole license). Jos maantieteellistä rajoitusta ei lisenssiehdoissa mainita ja lisenssinantaja voi näin myydä useampia lisenssejä samalle alueelle, on kyseessä valikoimaton lisenssi (non-exclusive license) [Classen, 2007; Pelkonen, 2003]. Molemmipuolisessa eli nk. ristiinlisensoinnissa sekä lisenssin ostaja että myyjä saavat ja antavat käyttöoikeuksia toisiltaan ja toisilleen [Pelkonen, 2003].

Jos yrityksen työntekijälle halutaan antaa mahdollisuus käyttää lisensoitua tuotetta myös kotonaan, siitä pitää mainita lisenssiehdoissa. Tästä käytetään termiä koti- tai rinnakkaislisenssi (home license tai dual-use license) [Tynan, 2004].

## 2.3. Lisensoinnin hyödyt ja haitat

Asiakasyritykselle lisenssin ostamisesta tuleva pääasiallinen hyöty on, että lisenssin hankkiminen mahdollistaa yritykselle lisätuloja esim. monipuolistuvampana tarjontana ja korkeampana teknologian tasona ilman tuotekehityksen ja tutkimuksen aiheuttamia kuluja. [Lainkari, 2005] Yrityksen ei siis tarvitse käyttää omia resurssejaan kehitelläkseen tuotetta johonkin tiettyyn tarkoitukseen, jos sellaisen voi ostaa valmiina. Usein sopimukseen kuuluvat myös tuotteen ylläpito ja päivitykset, jotka siis myös ovat pois lisenssinostajan työmäärästä. Varsin usein lisenssinostajalla on myös se tilanne, että ohjelmiston tekeminen ei ole yrityksen varsinaista liiketoiminta-alueetta, jolloin yritykseltä saattaa puuttua

tarvittava tietotaito. Tällöinkin yrityksellä on mahdollisuus saada käyttöönsä tarvitseman ohjelmisto, kun se ostaa lisenssin ohjelmistoon.

Ostajan kannattaa myös miettiä, haluaako hän ostaa vai vuokrata lisenssin. Jos halutaan vain käyttöoikeutta johonkin ohjelmistoon, on vuokraus hyvä vaihtoehto. Jos taas halutaan käyttöoikeuden lisäksi omistusoikeus ohjelmistoon, se pitää ostaa. Ostamisen hyötyä lisää myös, että kun käyttötarve loppuu, voi ohjelmiston myydä eteenpäin. [Hämäläinen, 2003; Hämäläinen, 2008]

Ohjelmiston vuokraan taas yleensä sisällytetään erilaisia houkuttimia, kuten uusien ohjelmaversioiden päivityksiä. Lisäksi budjetointi helpottuu, kun ei tarvitse varautua ohjelmistojen epäsäännöllisesti ilmaantuviin päivityksiin muuttuvine hintoineen. [Hämäläinen, 2008]

Riippumatta siitä, vuokrataanko ohjelmisto vai ostetaanko, on sopimusehdoissa hyvä mainita edelleen luovutusosoikeudesta. Tällä tarkoitetaan sitä, että jos yritys esimerkiksi myy vanhoja työkoneita henkilöstölleen kotikoneiksi, saako niihin asennettuja ohjelmistoja vielä käyttää vai täytyykö ne poistaa. [Hämäläinen, 2003; Hämäläinen, 2008]

Lisenssinantajalle lisenssit ovat nopea ja helppo keino siirtää korkeaa teknologiaa uusille alueille ja yrityksille. Jos esimerkiksi ohjelmistotalo haluaa kansainvälistyä, on sille pienempi riski myydä lisenssejä kohdemaahan kuin sijoittaa rahaa investointeihin tai yhteistyöyrityksiin esimerkiksi sellaisissa maissa, joiden poliittiset olot ovat epävakait. Ainoa, mitä lisenssinmyyjä voi tällaisessa tilanteessa menettää, ovat lisenssitulot. Toisaalta yleisesti voidaan ajatella, että lisenssinmyöntäjä menettää osan omaa kilpailuetuaan myymällä pois omia liikesalaisuuksiaan ja tietotaitoaan lisenssimaksua vastaan. [Lainkari, 2005]

Lisenssinantaja on mm. seuraavia hyötyjä lisenssinantajalle:

- Lisensseistä saatavat tuotot, joita ovat esimerkiksi etumaksut ja vuosittaiset maksut.
- Menojen väheneminen, jos tuotteen jatkokehittäminen annetaan myyntiorganisaatiolle. Tällöin lisenssinantaja voi siirtää työntekijänsä uusiin projekteihin.
- Laadukkaammat ohjelmistot, sillä myyntiorganisaatio todennäköisesti parantelee ohjelmistoa ennen sen tuomista markkinoille. Tästä on kahtalainen positiivinen vaikutus lisenssinantajalle; toisaalta heidän yrityskuvansa paranee, sillä alkuperäinen ohjelmisto on heidän ja varmasti mainitaan myös uudistetun tuotteen yhteydessä, ja toisaalta he yleensä saavat käyttöönsä ilmaisen lisenssin paranneltuihin ohjelmistoversioihin.
- Psykologiset hyödyt yrityksen työntekijöille, jotka kokevat työpanoksensa olevan arvokas yrityksen tuloksen tekijä. Tämä motivoi työntekijöitä. [Allemann, 2007]



## 2.4. Tekijänoikeus

Suomen tekijänoikeuslain [L404/1961] ensimmäisen pykälän mukaan tietokoneohjelma luokitellaan kirjalliseksi teokseksi. Se tarkoittaa sitä, että ohjelmistoihin sovelletaan kirjallisia teoksia koskevia erityissäädöksiä. Tekijänoikeuslain 19 pykälän mukaisesti ohjelmakappaleen valmistamiseen tai kopiointiin pitää olla tekijän lupa.

Lisenssinantajan kannattaa suojata lähdekoodinsa tekijänoikeudesta kertovalla copy right -merkinnällä jokaisen lähdekoodi-tiedostonsa alussa tai vähintäänkin jokaisen erillisen komponentin tai ohjelmapaketin yhteydessä [Lainkari, 2005]. Copy right -tekstin sijaan voidaan käyttää myös © -merkintää. Bernin yleissopimuksen artiklan 5, kohdan 2 mukaan [Bernin yleissopimus, 1986] copy right -merkintä ei ole pakollinen ja tekijänoikeus on silti voimassa, vaikka merkintää ei olisikaan. Toisaalta tekijänoikeus ohjelmistoon ei synny vaikka merkinnän lisäksi, jos ohjelmiston teoskynnys ei ylitä.

Mikä tahansa tietokoneohjelma ei ylitä teoskynnystä, vaan yleisenä sääntönä pidetään, että ohjelmassa pitää olla jotain omaperäisyyttä. Perusajatus on, että jos joku muu päättäisi kirjoittaa ohjelman samalla idealla, niin siitä ei tulisi täysin vastaavaa. Näin on esimerkiksi algoritmien laita, joissa on kuvattu vaiheet, joilla tietystä syötteestä saadaan haluttu tulos. Riippumatta siitä, kuka algoritmin kirjoittaa ohjelmaksi, lopputulos on suurin piirtein sama. Ohjelma, joka on tehty ainoastaan toteuttamaan jokin algoritmi, ei siis yksinään riitä ylittämään teoskynnystä.

Silti copy right -merkintä on hyvä olla olemassa siltä varalta, että ohjelmistoa myydään joskus maahan, joka ei ole liittynyt Bernin sopimukseen, mutta on allekirjoittanut yleismaailmallisen tekijänoikeussopimuksen (Universal Copyright Convention) [UCC, 1971]. Tästä mainitaan sopimuksen artiklassa 3 kohdassa 1.

Copy right -merkintä olisi hyvä lisätä kaikkiin yrityksen ohjelmiin riippumatta siitä, aiotaanko tuotetta lisensoida tai luovutetaanko lisenssin ostajalle lähdekoodi vai pelkkä ajettava ohjelma. Yleisesti suositellaankin, että yrityksen tulisi aina tehdä suunnitelma keinoista, joilla se aikoo suojata uuden ohjelmistotuotteensa. Tällaisella menettelyllä annetaan yrityksestä ulospäin kuva, että se pitää huolta immateriaalioikeuksistaan. [Lainkari, 2005]

Ohjelmistolisenssien perimmäinen tarkoitus on, että ohjelmiston julkaisija myöntää lisenssin yhteen tai useampaan kopioon ohjelmistosta, mutta pitää ohjelmiston omistusoikeuden itsellään. Tästä syystä käytännössä lähes kaikki ohjelmistoa koskevat oikeudet pysyvät julkaisijalla itsellään ja vain tietyt, hyvin rajatut oikeudet luovutetaan lisenssin ostajalle. Tällaiset ohjelmistolisenssit sisältävät myös useita sellaisia rajoituksia ohjelmiston käyttöön, jotka pelkkä tekijänoikeuslaki sallisi.

Ohjelmisto voidaan oikeudellisessa mielessä nähdä immateriaaliomaisuutena, kaupallisenä hyödykkeenä tai yleisenä hyödykkeenä. Immateriaaliomaisuuden näkökulmasta ajateltuna yrityksellä on lähes rajattomat oikeudet tehdä aineettomalle omaisuudelleen mitä haluaa. Immateriaaliomaisuutta suojaa tekijänoikeuslaki (immaterial property right).

Jos ohjelmiston tekijä päättää antaa osan tai kaikki näistä oikeuksistaan pois, siitä kirjoitetaan lisenssi- tai myyntisopimus, jossa kerrotaan tarkasti, mistä oikeuksista tekijä luopuu ostajan hyväksi. [Kim, 2008]

Toinen näkökulma on mieltää tuote kaupalliseksi hyödykkeeksi. Näin ajatellaan usein varsinkin valmisohjelmistojen kohdalla. Tällöin myyntitilanteessa ei välitetä niinkään lisenssiehdoista vaan kuluttajansuojalaista. [Kim, 2008]

Joskus ohjelmisto halutaan nähdä yleisenä omaisuutena, johon kaikilla on yhtäläiset oikeudet. Tätä näkökantaa edustaa avoin lähdekoodi. [Kim, 2008]

## 2.5. Lisenssisopimus

Tietokoneohjelman lisenssisopimusta suunniteltaessa on hyvä muistaa lain asettamat rajoitukset, joiden mukaan lisenssin ostajalla on oikeus ottaa ohjelmasta varmuuskopio, tutkia ohjelman toimintaperiaatteita normaalikäytön yhteydessä sekä kääntää ohjelman koodi, jos se on ohjelmien yhteensopivuuden kannalta välttämätöntä. Näitä oikeuksia ei voida lisenssisopimuksessa rajoittaa, mutta sen sijaan esimerkiksi ohjelmakoodin muuttaminen, kuten löydetyn virheen korjaaminen, voidaan sopimuksella estää. [Mansala, 2001]

Lisenssisopimuksessa on myös mainittava, ollaanko luovuttamassa tekijänoikeutta vai omistusoikeutta. Omistusoikeuden luovuttaminen on verrattavissa esimerkiksi musiikkiteoksen luovutukseen. Tällöin ostaja saa haltuunsa sekä ohjelmiston että oikeuden siihen. Sen sijaan ohjelman lähdekoodia ostaja ei saa haltuunsa, eikä hänellä myöskään ole oikeutta lisensoida ohjelmaa tai sen osia kolmannelle osapuolelle. [Mansala, 2001]

Kun yritys ensimmäisen kerran alkaa tuotteistaa ohjelmistojaan, sen on syytä perehtyä tarkoin muutamaankin tärkeään asiaan eli avoimen lähdekoodin käyttöön ja kilpailuun. Jos lisensoitava ohjelmisto pohjautuu avoimeen lähdekoodiin, on syytä tutkia huolellisesti, minkä avoimen lähdekoodin lisenssien alla lainatut koodit on julkaistu, et-tei niistä koidu ongelmia myöhemmin, kun ohjelmistoa lisensoidaan. On myös hyvä pohtia, onko ohjelmisto tai sen sisältämä teknologia liian tärkeää lisensoitavaksi. Jos ohjelmisto päätyisikin kilpailijalle, menetettäisiinkö siinä liiaksi kilpailuetua. [Allemann, 2007]

Lisenssisopimus voi syntyä joko neuvotteluiden tuloksena tai ilman neuvotteluja. Klassisen mallin neuvotteluissa käydään läpi normaalit sopimuksen solmimisen vaiheet tarjouspyynnöstä aina yhteisymmärryksen syntymiseen ja sopimuksen muodostumiseen. [Kim, 2008]

Ilman neuvotteluja syntyvät lisenssit voidaan jakaa neljään pääryhmään: toimeenpannut kirjoitetut sopimukset (executed written agreements), clickwrap-lisenssit (clickwrap agreements), shrinkwrap-lisenssit (shrinkwrap agreements) ja browsewrap-lisenssit (browsewrap agreements). [Kim, 2008]

Normaalisti lisenssisopimukseen tarvitaan molempien osapuolten allekirjoitukset, vaikka sopimusteksti olisikin kirjoitettu ilman osapuolten välisiä neuvotteluja. Tällaisella

menettelyllä voimaan astuvaa lisenssiä kutsutaan toimeenpannuksi kirjoitetuksi sopimukseksi. Yleensä tällaista järjestelyä käyttävät isot yritykset, joilla on monopoli kyseisillä markkinoilla ja jokin pienempi yritys haluaa ostaa tuotteen lisenssin. Tällöin isomman yrityksen ei ole taloudellisesti kannattavaa aloittaa neuvotteluja, vaan se antaa kiinteät ehdot ja hinnan tuotteestaan, jonka pienempi yritys joko hyväksyy tai hylkää. [Kim, 2008]

Clickwrap-sopimuksiksi luokitellaan sellaiset sopimukset, jotka tulevat esille elektronisesti esimerkiksi ohjelman lataamisen tai asentamisen yhteydessä tai ennen käyttöi-keuden sallimista tietyille internet-sivustolle. Käyttäjä hyväksyy lisenssiehdot klikkaamalla painiketta, jossa lukee ”suostun”, ”ok” tai jokin vastaava myöntymistä merkitsevä sanamuoto. [Kim, 2008; Oksanen, 2006] Samantapaisia lisenssejä ovat myös browsewrap-lisenssit, joissa käyttäjän on ensin ostettava lisenssi, jotta hänellä olisi pääsy jollekin tietyille internet-sivustolle [Kim, 2008].

Vyöryviksi sopimuksiksi (rolling contracts) sanotaan sellaisia sopimustyyppisiä, joissa ostajan pitää hyväksyä sopimusehdot ennen niiden lukemista. Tällainen sopimustyyppi on mm. shrinkwrap-lisenssi, jossa käyttäjän täytyy ensin ostaa valmisohjelmisto ja hyväksyä lisenssiehdot avaamalla ohjelmiston asennuslevykettä, ohjeistusta ja lisenssisopimusta suojaava muovikääre tai myyntipakkaus. [Kim, 2008]

Clickwrap- ja shrinkwrap-sopimusten sitovuudesta on kiistelty paljon, sillä näissä järjestelyissä ostaja ja myyjä eivät vahvista sopimusta suullisesti eivätkä kirjallisesti. Todellisuudessa vain ani harva kuluttaja lukee esimerkiksi loppukäyttäjän sopimustekstiä EULAA (end-user license agreement) tai ainakaan sisäistää täydellisesti sen merkitystä. Ongelmaksi koetaan myös se, että käyttäjän on avattava pakkaus – näin ollen antaa suostumuksensa noudattaa lisenssisopimusta – edes nähdäkseen, mitä sopimustekstissä sanotaan [Hämäläinen, 2008; Oksanen, 2006]. Yleensä sopimustekstissä onkin maininta, että sopimus tulee voimaan vasta sitten, kun kyseinen ohjelmisto asennetaan ja otetaan käyttöön. Yleensä ostaja voi myös palauttaa tuotteen ja saada rahansa takaisin, jos ei suostu sopimusehtoihin.

### 3. Tutkimuksia lisensointiin liittyen

Seuraavaksi esittelen muutamia lisensointia käsitteleviä mielenkiintoisimpia tutkimuksia.

#### 3.1. Kahden periodin -malli ohjelmistojen jakamiseen

Zhang ja Seidmann [2003] esittelevät optimaalista lisensointistrategiaa sellaiselle yritykselle, joka vuokraa ja myy monopolin kaltaisesti ohjelmistoja ja palveluita verkon yli.

Mallin kehittämiseen on vaikuttanut mm. yritysten muuttuva myyntistrategia, jossa pysyvän lisenssin myynnin sijaan vuokrataan ohjelmistoja ja erilaisia palveluita, kuten ylläpitoa, päivityksiä ja koulutusta, verkon yli. Verkon yli tulevista päivityskehotuksista ja päivitysten hinnoittelusta on herännyt kysymys, onko asiakkaan pakko ostaa vuosittainen ohjelmistopäivitys ja tuleeko se lopulta kalliimmaksi kuin kerralla ostettu lisenssi tuotteeseen.

Mallissa tuotteen myynti ja vuokraus jaetaan kahteen periodiin. Periodin I alussa tuotteesta julkaistaan versio I sekä myyntiin että vuokrattavaksi. Myyjä lupaa pitää tuotteen vuokran samana kahden periodin ajan. Kun periodin II alussa ohjelmistosta julkaistaan versio II, voi lisenssin ostanut asiakas päättää, haluaako hän päivityksen vai ei; vuokraajalle päivitys asennetaan automaattisesti. Periodilla II uudet asiakkaat saavat ostaa ohjelmasta version II. Molemmissa tapauksissa periodilla I tuotteen hankkineille hinta jää samaksi riippumatta siitä, ostetaanko vai vuokrataanko se.

Ero kahden vaihtoehdon, ostamisen ja vuokraamisen, välillä syntyy tuotteen laadussa, sillä jos versio II on laadullisesti huonompi kuin versio I, vuokraaja ei pysty kieltäytymään päivityksestä, mutta jos tuotteen ostanut henkilö ei huoli päivitystä, hän saa käyttää laadukkaampaa ohjelmaversiota pidempään.

#### 3.2. Lisensoinnin vaikutus yrityksiin ja verkostoihin

Persson ja Steinby [2006] tutkivat, miten sääntely ja etenkin lisensointi vaikuttaa valtion valvomiin yrityksiin ja niiden muodostamiin yhteistyöverkostoihin.

Tutkimusta tehtiin kahdesta liiketoimialasta, lääke- ja vakuutusalaista, Suomessa ja Ruotsissa. Tutkimuksen lähtökohtana oli selvittää, miten yrityksiin vaikuttaa, että niiden pitää ensin hakea oikeutusta olemassaololleen ja todistaa pätevyytensä anomalla lisenssiä alalla toimimiseen. Tämän oletettiin rajoittavan usealla tavalla yrityksen rakennetta ja verkostoitumista.

Verkostoitumisen tutkimisessa käytetään kahta lähestymistapaa. Ensinnä oletetaan, että suhteet syntyvät asianosaisten yritysten vapaaehtoisesta kanssakäymisestä. Toinen oletus on, että ympäristön muutokset vaikuttavat kanssakäymiseen.

Ykkösoletuksen mukaisesti verkostot syntyvät luonnollisesti sen mukaan, kenen kanssa yritys tekee yhteistyötä. Mutta usein yhteiskunta pakottaa yrityksen tekemään tiettyjä valintoja (esimerkiksi lisenssisopimukseen merkittyjen rajoitusten muodossa) ja tällä perusteella se pystyy vaikuttamaan muodostuvaan verkostoon.

Toisessa näkökannassa lähdetään siitä, että ympäristö lasketaan mukaan yhdeksi toimijaksi pelikentässä, jolloin sillä siis on yhtäläinen mahdollisuus vaikuttaa syntyvään rakenteeseen kuin muillakin osapuolilla.

Tutkimuksesta käy ilmi, että valtiolla on montakin eri tapaa puuttua verkostoitumiseen. Ensimmäinen tutkimusoletus verkostoitumisen vapaaehtoisuudesta kumotaan, sillä molemmilta aloilta oli löydettävissä useitakin esimerkkejä ei-vapaaehtoisesta kanssakäymisestä.

Toisessa oletuksessa pohditaan ympäristön ja etenkin lisenssien ja määräysten kautta tapahtuvaa väliintuloa verkostojen kehityksessä. Tällainen yhteys on löydettävissä, mutta edelleen halutaan tutkia, ovatko vaikutukset pelkästään rajoittavia vai löytyykö myös yhteistyötä helpottavia tekijöitä.

Tutkimuksessa löydettiin kolme eri aluetta, joilla rajoituksista on sekä hyötyä, että haittaa. Tuotteiden osalta lisenssit rajoittavat mm. varastointia ja valikoimaa, mutta toisaalta ne myös selkeyttävät yritys-konseptia, helpottavat kommunikointia asiakkaiden kanssa sekä lisäävät yrityksen luotettavuutta.

Lisenssit ja säädökset rajoittavat myös yhteistyökumppaneiden valintaa sekä yhteistyön muotoja. Toisaalta rajoitukset myös helpottavat ja suojaavat yhteistyökumppaneiden välille muodostuvia suhteita sekä poistavat epävarmuutta näiden väliltä.

Lisenssit lukitsevat ja säätelevät myös yrityksen omistusrakennetta, mikä osaltaan vähentää kilpailua. Toisaalta, hyvin rakennettu omistus pohja ja selkeästi määritellyt roolit takaavat tehokkuuden, pääsyn sisään muihin vastaavankaltaisiin verkostoihin sekä hyvät markkinointimahdollisuudet.

### **3.3. Ohjelmistojen muuttuva hinnoittelu**

Cusumano [2007] kirjoittaa lisenssien hinnoittelusta ja hinnoitteluun vaikuttavista muutoksista markkinoilla.

Alun perin lisenssit myytiin lähinnä siten, että asiakas maksoi lisenssin etukäteen ja sai tällä tavoin käyttää tuotetta rajoittamattoman ajan esimerkiksi jossakin tietokoneessa.

Suuri muutos markkinoissa tuli avoimen lähdekoodin ohjelmistojen myötä. Kun asiakkaat huomasivat saavansa vastaavia ohjelmistoja ilmaiseksi, he alkoivat kritisoida ohjelmistolisenssien kalliita hintoja. Tästä syystä yritysten on ollut pakko käyttää mielikuvitustaan ja keksiä uusia tapoja saada rahaa kehittämästään ohjelmistosta.

Eräs ratkaisusta on kausilisenssit, jolloin asiakas ostaa ohjelman esimerkiksi kolmeksi vuodeksi käyttöönsä. Kun kausi päättyy, asiakas saa päättää, jatkaako hän sopimusta

vai lopettaako hän ohjelmiston käytön. Lisenssi kattaa vain ohjelman käytön, tuesta ja muista palveluista joutuu maksamaan erikseen. Alussa maksettava iso korvaus siis pilkootaan pienemmäksi eli lisenssin hinnaksi ja tukipalveluiden hinnoiksi; lopputulos pysyy kuitenkin samana. Tämä ulkoapäin joustavalta vaikuttava malli on kuitenkin käytännössä usein lähellä pysyvää järjestelyä, sillä harva asiakas on oikeasti valmis poistamaan ohjelmistoa käytöstä ja ostaa siis mieluummin uuden lisenssin.

Toinen kausilisenssimalli on sellainen, jossa kaudet ovat todella lyhyitä, esimerkiksi vain kuukauden mittaisia. Kun tällä lisenssillä saa myös täyden palvelun, sitä kutsutaan nimellä ”ohjelmisto palveluna” (software as a service). Myös tässä mallissa pyritään sitomaan lisenssin alkuhinta ja palveluiden hinnat yhteen siten, että ohjelmistosta saataisiin haluttu hinta.

Muitakin ratkaisuvaihtoehtoja on, kuten esimerkiksi eri versiot ”ilmainen, mutta ei ilmainen” (free, but not free) -mallista. Yksi tällainen on nk. partaveitsi-malli (razor and blade model), jossa asiakkaalle annetaan ohjelmistosta perustoiminnallisuus, mutta laskutetaan muista osista, päivityksistä tai kehittyneemmistä ohjelmiston osista. Tästä hyvä esimerkki on Adoben Acrobat -tuoteperhe, josta käyttäjille annetaan ilmaiseksi Adobe Acrobat reader -tiedostonlukuohjelma, mutta laskutetaan muokkaustyökalusta.

Toinen vastaava malli on palvelumalli (service model), jossa käyttäjälle annetaan ohjelmisto ilmaiseksi, mutta laskutetaan erikseen palveluista, kuten ylläpidosta, päivityksistä, koulutuksesta, asennuksesta ja tukipalveluista. Useat vapaan lähdekoodin ohjelmistoista, kuten esimerkiksi Red Hat, toimivat tämän mallin perusteella.

Kolmas ”ilmainen, mutta ei ilmainen” -mallin muunnelma on mainostusmalli (advertising model). Tällaisessa mallissa asiakas saa käyttää tuotetta vapaasti, mutta joutuu samalla katselemaan mainostajien mainoksia. Tunnetuin esimerkki tämän mallin kaltaisesta ohjelmiston jakamisesta on internet-hakukone Google.

Artikkelin lopussa kerrotaan, että vuoden 1997 jälkeen markkinoilla ovat oikeastaan selvinneet vain sellaiset ohjelmistotalot, jotka ovat omaksuneet uusia lisensointimalleja. Tulevaisuudessa luultavasti suosituimmaksi malliksi nousee kuukausittainen merkintä (monthly subscription), joka sisältää myös isännöintipalvelut (hosting). Asiakkaan näkökulmasta tämä selkiyttää laskutusta ja pidemmällä aikavälillä hyödyttää myös yrityksiä tasaamalla tulojen kertymistä.

### **3.4. Ohjelmistojen jakamiseen liittyvää ongelmatiikkaa**

Kim [2008] pohtii oikeuden näkökulmasta ongelmaa, johon törmätään, kun ohjelmistojen lisenssiehtojen rikkomisesta käydään oikeutta. Noudatettavat lait riippuvat siitä, tulkittaanko myyjän ja ostajan välille syntynyt sopimus myynniksi vai lisenssiksi.

Kun tarkastellaan ohjelmistojen myynnin ja lisensoinnin välistä erottelua, on löydettävissä kaksi erilaista näkökulmaa. Ensimmäinen keskittyy ohjelmistolisenssien linjauksen merkitykseen, lähinnä kysymyksiin, pitäisikö ohjelmistot mieluummin lisensoida kuin

myydä ja kumpi sopisi paremmin ohjelmistojen luonteeseen sekä miten ylipäänsä ohjelmistoja pitäisi määritellä.

Toinen näkökulma keskittyy ohjelmistojen jakamisen (distribution) muotoihin. Tällöin keskitytään yleensä vain lisensseihin, joista osapuolet eivät ole neuvotelleet keskenään. Kiinnostava kysymys kuuluu, ovatko molemmat osapuolet suostuneet sopimusehtoihin?

Molemmat näkökulmat sopivat yhteen ohjelmistolisensoinnin ainutlaatuisen luonteen kanssa. Lisenssejä tutkittaessa punnitaan sopimuksen, immateriaalioikeuksien, kuluttajan ja digitaalisen tiedon välistä vuorovaikutusta. Ohjelmistojen lisensointia ei voida tulkita yksinomaan tekijänoikeuden kannalta, sillä mukaan on huomioitava myös teknologia, jolla ohjelmisto on tehty sekä itse ohjelma eli tuote.

Yleensä sopimukset sisältävät molempien, sekä lisensoinnin että myynnin ehtoja. Tämä johtuu siitä, että yleensä henkisen työn tuloksesta myydään vain kopio, mutta ohjelmistossa mukaan tulevat myös konekieliset ohjelmakoodit. Lisensointi nostaa esiin monia mutkikkaita kysymyksiä liittyen sekä ohjelmistojen luonteeseen että tapaan, jolla niitä jaetaan kuluttajille.

Ohjelmistot eivät oikeusmielessä istu suoraan mihinkään olemassa olevaan luokitukseen, sillä ne ovat sekä aineellista että aineetonta, yksityisesti omistettua että yleisesti jaettavissa olevaa omaisuutta. Vaikka ohjelmisto itsessään olisikin tekijänsä omaisuutta, niin välinettä, jossa sitä säilytetään, voidaan kuitenkin siirrellä toisten osapuolten välillä. Toisin kuin aineellista omaisuutta, ohjelmakoodia voidaan helposti muokata ja jakaa edelleen ilman, että tekijä tietää siitä tai on suostunut siihen. Toisaalta, toisin kuin aineeton omaisuus, kuten liikesalaisuus tai tavaramerkki, ohjelmisto on usein helposti saatavilla sekä kenen tahansa myytävissä ja ostettavissa.

Ohjelmistojen jakamiseen liittyvä luonnehdinta vaikuttaa myös muihin markkinoihin. Esimerkiksi, jos sopimusta käsitellään myyntinä, niin laki (teksti ei mainitse, minkä maan laista on kysymys, mutta koska artikkelin kirjoittaja on Kaliforniasta, niin voidaan olettaa, että kyseessä on Kalifornian laki) sanoo, että tekijänoikeus suojaa omistajaa siten, että vain hänellä on oikeus uudelleen koota sekä jatkokehittää ja kopioida työnsä tuloksia. Mutta myyntilain mukaan kopion omistajalla on myös oikeus edelleen myydä tai edelleen luovuttaa kappaleensa kysymättä lupaa tekijänoikeuden omistajalta. Myös tietokoneiden yhteydessä käytettävä nk. reilun käytön -provisio (fair use -provision) sanoo, että kopion omistajalla on oikeus ottaa toinen kopio ohjelmasta, vaikka tekijänoikeuden pitäisi estää tämä.

Ongelmia tulee myös siitä, ymmärretäänkö ohjelmisto omaisuudeksi (property) tai aineettomaksi omaisuudeksi (immaterial/intangible property), tuotteeksi (good/product) vai yleiseksi omaisuudeksi (public property). Jos ohjelmisto katsotaan omaisuudeksi, sen myyntiä voidaan säädellä sopimuksin. Jos kyseessä on tuote, sen myyntiin pätevät samat kaupankäynnin säännöt kuin minkä tahansa tavaran myymiseen, mm. kuluttajansuojalaki.

Jos taas ohjelmisto on yleistä omaisuutta, kuten avoimen lähdekoodin ohjelmistoja, sen saatavuutta ei saisi tarpeettomasti rajoittaa. Tällainen jaottelu ei kuitenkaan käytännössä toimi, koska ohjelmiston luonteen ratkaisee lähinnä asiayhteys ja se, kuinka ohjelmistoa käytetään.

Seuraavankaltainen ohjeistus auttaa miettimään, oliko ohjelmiston siirto luokiteltava lisenssin ostamiseksi/myynniksi vai kaupaksi:

- Syntyivätkö sopimusehdot neuvottelun tuloksena (negotiated or non-negotiated): Jos sopimusehdoista ei ole neuvoteltu, niin osapuolet voivat ymmärtää sopimustekstin eri lailla. Tästä ei voida vielä päätellä, onko ostaja tehnyt hankintapäätöksen tietoisesti vai ei.
- Hinnoittelun rakenne (structure of license fee/timing and method of payment): Jos kauppahinta maksetaan kertakorvauksena ja etukäteen, on kyse luultavimmin kaupasta ainakin ostajan näkökulmasta. Jos taas hinnoittelu perustuu lisenssimaksuihin tai maksu jaotellaan ajan tai käytön mukaan, on luultavasti tarkoitettu lisenssisopimusta.
- Oikeuksien laajuus (scope of license grant): Jos sopimuksessa on tiukat rajoitukset ohjelmiston käytölle, niin luultavasti on tarkoitettu tehdä lisenssisopimus. Väljät rajoitukset viittaavat siihen, että oli tarkoitus tehdä kauppasopimus. Oikeuksien myöntäminen ei oikeastaan ole sopimus käytön laajuudesta, vaan tekijänoikeuden omistajan lupaus olla haastamatta ostajaa oikeuteen, mikäli tämä pysyy sopimuksen rajoissa ja maksaa laskunsa.
- Verotus ja kirjanpito (tax and accounting issues): Sekä myyjä että ostaja merkitsevät kirjanpitoonsa ostotapahtuman eri tavalla kuin lisenssin hankinnan. Kirjanpitoa tarkastelemalla nähdään, ovatko ostaja ja myyjä merkinneet ne samalla tavoin.
- Sopimusehtojen yksilöinti (mass consumer or individualized license terms): Yleensä yksilöidyt ehdot viittaavat lisenssisopimukseen, sillä silloin sekä myyjän että ostajan intresseissä on sopia tarkemmin ehtojen yksityiskohdista ja laajuudesta.
- Jakelukanava (method of distribution): Jos myyntitapahtumassa käytetään kolmatta osapuolta, niin tämän rooli ratkaisee, onko kyse kaupasta vai lisenssistä. Jos kolmas osapuoli ei ole valtuutettu toimimaan ohjelmiston omistajan puolesta, on kyse enemmän kaupasta kuin lisenssistä. Jos taas kolmas osapuoli on valtuutettu jälleenmyyjä (sublicensor/agent), niin kyseessä on ehkä ennemminkin lisenssi.
- Yhteistyön jatkuvuus (existence and nature of ongoing relationship): Yleensä osapuolet erkanevat kaupanteon jälkeen toisistaan ja ovat yhteydessä vain tuotteen takuuseen liittyvissä asioissa. Jos taas yhteistyö jatkuu myös muilta kuin takuun osalta sopimuksen teon jälkeenkin, on tapahtuma käsitettävä lisenssiksi.



- Sopimuksen voimassaoloaika (term of license): Jos sopimus rajoittaa ohjelmiston käyttöä määräajaksi, on kyseessä lisenssi. Jos käyttöaikaa ei ole rajoitettu päättyväksi minään tiettyä aikana, on kyseessä kauppa.
- Ohjelmiston hävittäminen (destruction or return of product): Ohjelmistojen helpon kopioinnin vuoksi ohjelmalevykkeen palauttamista tai tuhoamista ei voida pitää sopimuksen päättymisen todellisena merkinä, mutta jos päättymisaika on merkitty sopimukseen, se tarkoittaa, että sopimuksella on tarkoitettu nimenomaan lisenssisopimusta, ei kauppaa.

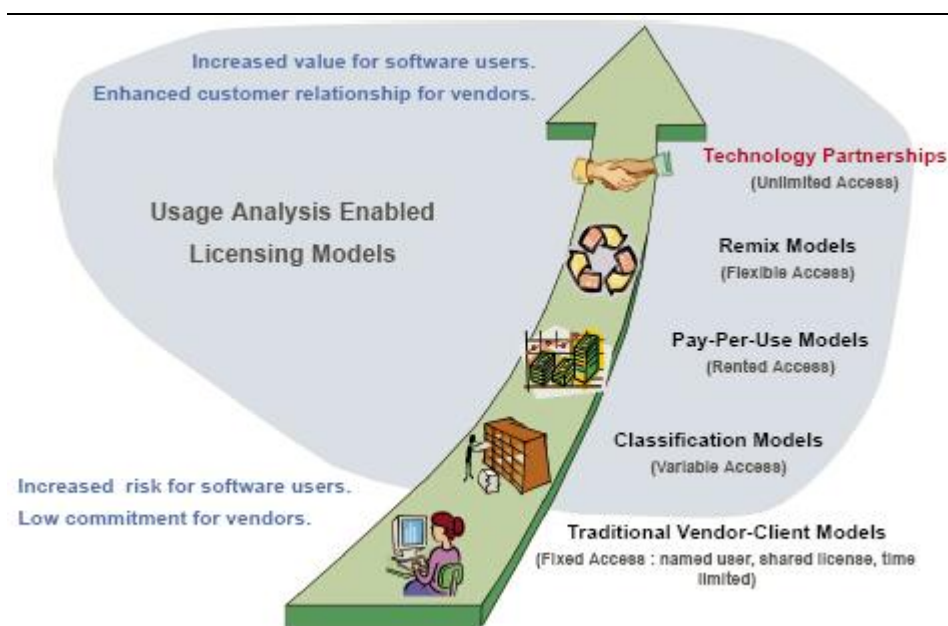
Kim [2008] ehdottaa, että lisensointia ja myyntiä ei pitäisi erotella toisistaan vaan tehdä malli, joka soveltuu yleisesti ohjelmistojen jakamiseen. Tällainen joustava sopimuslaki (dynamic contract law) perustuisi jatkuvaan ketjuun, jossa tapahtumat edeltäisivät ja seuraisivat toisiaan tai muuten vaikuttaisivat sopimuksen syntyyn. Tämän tarkoituksena olisi kartoittaa osapuolten näkökulmaa siitä, onko kyseessä kauppa- vai lisenssisopimus, sekä tasata osapuolten välisiä linjanvetoja. Tällaiset teknologian luomat ongelmat pitäisi ratkaista innovatiivisilla yritysstrategioilla eikä uusilla ja kyseenalaisilla lakikiemuroilla.

## 4. Lisensointimalleja

Samaan aikaan, kun teknologia kehittyy, kehittyvät mukana myös perinteiset ostajan ja myyjän väliset lisensointimallit, kuten ”yksi käyttäjä – yksi lisenssi” (single user – single license), ”monta käyttäjää – jaettu lisenssi” (multiple users – shared license) sekä ajallisesti joko rajoitettu tai rajoittamaton lisenssi. Tämä johtuu siitä, että perinteiset lisensointimallit eivät enää tyydytä asiakkaan muuttuvia markkinatarpeita, kuten tasapainoilua tuottavuuden ja tehokkuuden välillä tai mukautumista muuttuviin taloudellisiin tilanteisiin. [Ingram, 2006]

Nykypäivän kasvava teknologiantaso ja monipuolistuvammat markkinat tarvitsevat joustavampia ratkaisuita, ja siksi suuntaus onkin kohti kumppanuussopimuksia ja yhteistyömalleja. [Ingram, 2006]

Kuten kuva 1 kertoo, perinteiset mallit ovat olleet ostajalle riskiksi, että myyjällä ei ole lisenssin myynnin jälkeen enää kovinkaan suurta intressiä myytyä tuotetta kohtaan. Sen sijaan teknologiakumppanuussuhteilla on suotuista vaikutus molemmille osapuolille. Lisenssinantajan asiakassuhteet paranevat ja tätä kautta koko liiketoiminta kasvaa. Lisenssinostaja taas saa täyden tuen ja hyödyn ostamastaan tuotteesta. [Ingram, 2006]



Kuva 1. Lisensointimallien kehitys kulkee kohti kumppanuussopimuksia [Ingram, 2006, s. 1].

Mikä sitten on motivaationa uusien lisensointimallien kehittämiseksi? Ostaja joutuu tasapainottelemaan tuottavuuden ja tehokkuuden välillä. Tuottavuus kärsii, jos työntekijät eivät pääse käyttämään kriittistä ohjelmistoa. Samoin käy, jos joudutaan ostamaan ylimääräisiä lisenssejä esimerkiksi sesonkihuippujen tarpeita varten [Ingram, 2006]. Jos

ostaja joutuu ostamaan ohjelmistolisenssin kokonaan itselleen, hänen pitää tehdä se maksikäyttötarpeensa mukaan, jotta hänellä olisi tarvittavat resurssit käytössään silloin, kun niitä tarvitaan. Lisenssien kertaostoshinta on korkea, ja jos ohjelmiston normaali käyttö on huomattavasti alhaisempi kuin sesonkihuippuina, asiakas menettää rahaa. Tällainen herättää asiakkaassa negatiivisia tunteita, mikä ymmärrettävästi vaikeuttaa yhteistyötä jatkossa.

Edullisemmat väliaikaiset lisenssit voivat helpottaa tilannetta, mutta eivät ratkaise sitä kokonaan. Tällöin voi käydä niin, ettei lisenssiä saadakaan silloin, kun ohjelmistolle olisi tarvetta ja tällöin taas tuottavuus alenee. Tulevaisuuden ennustaminen lisenssin ostohtekellä on myös vaikeaa varsinkin silloin, jos ollaan ostamassa uudelta, ennestään tuntemattomalta myyjältä tai ennestään tuntematonta teknologiaa. Mitä joustavammiksi järjestelyt tehdään, sitä laajemmat arviointimahdollisuudet ostaja saa ennen tuotteen hankkimista ja ilman turhia kopioita tuotteesta. [Ingram, 2006]

Yllä mainitut asiat tuovat etua myös lisenssinantajalle. Ohjelmistontuottajien keskinäinen kilpailu yllyttää parantamaan asiakassuhteita ja etsimään keinoja saavuttaa suurempia markkinaosuuksia esimerkiksi vakiinnuttamalla asemansa asiakkaan ykkösohjelmistotoimittajana. Uusien innovatiivisten lisensointimallien omaksuminen voi toimia houkuttimena potentiaalisille asiakkaille, kun taas niiden käyttöönotto myöhemmässä vaiheessa voi antaa yrityksestä joustamattoman kuvan [Ingram, 2006]. Tietotekniikka-alalla pelataan suurelta osin mielikuvien varassa – mikä yritys on trendikkäin tai teknologinen edelläkävijä – ja jokainen yritys pyrkii antamaan itsestään mahdollisimman nuorekkaan, innovatiivisen ja asiakasorientoituneen kuvan. Jos yrityksen palvelutarjonta ei täysin vastaa asiakkaan tarpeita, löytyy markkinoilta aina jokin yritys, joka kyseisen palvelun tarjoaa. Vaikka yrityksen tarjoama tuote olisi kuinka hyvä tahansa, niin kankeat lisensointimallit voivat saada asiakkaan epäröimään tuotteen hankintaa.

Karkealla tasolla lisenssit voidaan jakaa kolmeen pääryhmään: loppukäyttäjän lisenssiin (end user licenses), yhteishankkeisiin (spin-out/joint ventures) sekä jälleenmyyntisopimuksiin (distribution agreements).

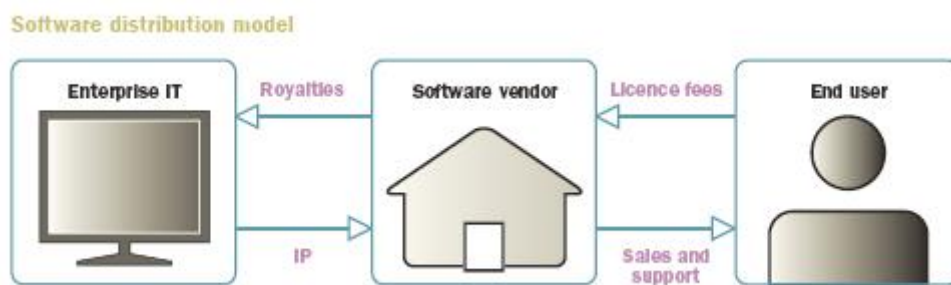
Loppukäyttäjän lisenssi -mallissa yritys itse myy ohjelmistojansa tai teknologiaansa suoraan toiselle yritykselle tai yksityiselle henkilölle ilman välikäsiä. Tällaisen järjestelyn etu lisenssinantajalle on siinä, että sillä säilyy täydellinen kontrolli lisenssin hinnasta sekä siitä, kenelle lisenssejä myönnetään. Tämä voi olla erittäin tärkeää tilanteissa, joissa yritys ei halua tuotteensa päätyvän mahdollisesti tulevalle kilpailijalle. Haittapuoleksi tällaisessa mallissa voidaan katsoa, että lisenssinantaja on itse vastuussa markkinoinnista, asiakastuesta ja ylläpidosta. Joskus lisenssinantaja ei halua itselleen tällaista taakkaa, varsinkaan jos kyseessä ei ole hänen keskeinen liiketoimintansa [Allemann, 2007]. Yritys on voinut luoda ohjelmiston omiin tarpeisiinsa ja tukemaan omaa perusliiketoimintaansa, joka voi olla toiselta alalta, ja siksi yrityksellä ei riitä resursseja ja tietotaitoa ohjelmiston jakamiseen edelleen, vaikka itse tuote olisikin sellainen, joka voisi hyödyttää myös muita alalla

toimivia yrityksiä tai henkilöitä. Tällaisessa tilanteessa yrityksellä on käytännössä kolme vaihtoehtoa. Se voi joko tyytyä pitämään ohjelmiston vain itsellään tai julkaista ohjelmisto avoimen lähdekoodin lisenssin alla, mutta näin myös menettää mahdolliset lisenssitulot, tai se voi yrittää löytää kolmas osapuoli, joka alkaa jälleenmyydä tuotetta.

Yhteishankkeissa luodaan aluksi uusi yritys, joka vastaa lisensoitavan tuotteen myynnistä ja markkinoinnista. Pitkällä aikavälillä tällainen malli voi olla hyödyttävä, jos uusi yritys pärjää markkinoilla hyvin. Järjestelyn ansiosta ohjelmiston luonut yritys myös välttyy markkinointi-, asiakastuki- ja ylläpitotöiltä. Uuden yrityksen luominen tosin vaatii paljon alkupääomaa eli rahaa, aikaa ja henkilökuntaa. Jos uusi yritys ei menestykään markkinoilla, se tarkoittaa pääoman menetyksen lisäksi myös yrityskuvan kärsimistä yhteistyöyritykselle [Allemann, 2007]. Myyntiä varten luotu yritys perii suuren osan yrityskuvastaan perustajayritykseltään. Maine karttuu ajan kuluessa myös myytyjen tuotteiden saamasta palautteesta. Tämä maine kulkeutuu osittain takaisin tuotteen luoneelle yritykselle, koska markkinoilla ne rinnastetaan usein toisiinsa. Yritysten maineet siis vaikuttavat vääjäämättä toisiinsa niin hyvässä kuin pahassakin.

Jälleenmyyntisopimus toimii siten, että ohjelmiston tehnyt yritys myy lisenssin yhdelle tai useammalle myyjäyritykselle, jotka sitten kaupallistavat tuotteen ja myyvät sen eteenpäin loppukäyttäjille. Myyntiyritysten vastuulla on myös tuotteiden ylläpito ja asiakastuki. Ohjelmiston tehnyt yritys saa yleensä myyntiyritykseltä etukäteismaksun sekä myös tuloja sitä mukaa, kun tuotteistettua ohjelmistoa myydään eteenpäin. [Allemann, 2007]

Tällaisen tietotaidon, vastuiden ja rahan kiertokulun voi nähdä myös kuvasta 2. Etuna tällaisessa järjestelyssä on, että ohjelmistoa saadaan levitettyä suuremmalle ostajajoukolle, mikä tietysti tarkoittaa suurempia tuottoja sekä myyjälle että tuotteen kehittäjälle. Toisaalta tuotteen kehittäjä joutuu yleensä luopumaan osasta kontrollistaan hinnoittelun ja ohjelmiston ostajien valinnan suhteen. [Allemann, 2007]



Kuva 2. Jälleenmyynti-mallin mukainen vastuujako [Allemann, 2007, s. 10]

## 4.1. Perinteiset myyjä-asiakas -mallit

### 4.1.1. Yhden käyttäjän lisenssi

Perinteisistä myyjä-asiakas -malleista (traditional vendor-client models) yksinkertaisimmassa, perinteisessä yhden käyttäjän lisenssi -järjestelyssä (single user license), asiakas ostaa yhden lisenssin jokaiselle ohjelmistoa käyttävälle työntekijälle. Tällainen lisenssi voidaan ostaa joko jollekin nimetylle henkilölle (seat license) tai vaihtoehtoisesti johonkin tiettyyn laitteeseen (cpu license) [Hämäläinen, 2003; Ingram, 2006; Tynan, 2004]. Tämänkaltaiset yhteen toiminnalliseen solmuun, joko henkilöön tai laitteeseen, sidotut lisenssit ovat helposti hallittavissa sekä ostajan että myyjän näkökulmasta. [Ingram, 2006]

Kaikkein perustavanlaatuisin yhden käyttäjän lisenssityyppi on OEM-lisenssi (original equipment manufacturer -license). Tällä lisenssillä myytävä ohjelmisto asennetaan yleensä ennen myyntiä suoraan tietokoneeseen, jossa sitä saa käyttää niin kauan kuin tietokone on käytössä, mutta sitä ei saa siirtää toiseen tietokoneeseen, vaikka alkuperäinen laite hajoaisi tai esimerkiksi poistettaisiin käytöstä vanhentuneen tekniikan vuoksi. Tällaisella lisenssillä myydään mm. tietokoneen mukana tulevat käyttöjärjestelmät. [Hämäläinen, 2008]

Tilanne on helpoin ostajan kannalta silloin, kun kyseessä on edullinen ohjelmistolisenssi, sillä tällöin voidaan ostaa oma lisenssi jokaiselle työntekijälle. Tilanne monimutkaistuu, jos lisenssi on kallis [Ingram, 2006]. Tällöin asiakkaan tulee miettiä, kannattaako jokaiselle työntekijälle ostaa oma lisenssi vai voisiko lisenssin ostaa esimerkiksi vain osaan laitteista ja työnkiertoa lisäämällä taata kaikille työntekijöille käyttömahdollisuus ohjelmistoon. Tämäkään ei auta tilannetta, jossa joskus ohjelmiston tulisi olla kaikkien työntekijöiden saatavilla.

### 4.1.2. Monen käyttäjän jaettu lisenssi

Kun tietokoneita alettiin verkottaa 1980-luvun loppupuolella, mahdollistui uudenlainen lisensointitekniologia, joka salli lisenssin jakamisen useamman työntekijän kesken [Ingram, 2006]. Tuolloin otettiin myös käyttöön nk. suuriasiakas- ja oppilaitoshinnoittelut [Hämäläinen, 2003]. Tällaisesta lisensointimallista käytetään joko termiä monen käyttäjän jaettu lisenssi (multi-user shared license), yhtäaikaisten käyttäjien lisenssi (concurrent user license) tai kelluva lisenssi.

Tällaisessa mallissa asiakas ostaa esimerkiksi 10 lisenssiä, jolloin 10 työntekijää saa käyttää ohjelmaa yhtä aikaa. Käyttäjien ei tarvitse olla erikseen nimettyjä henkilöitä, vaan rajoitus on se, että yhtä aikaa ohjelmaan saa olla auki 10 eri yhteyttä. Yhtäaikaisten yhteysien määrää valvoo joko ohjelmisto itse tai verkon yli käytettävissä ohjelmissa tarkoitusta varten luotu apuohjelma, joka sallii 10 yhteyttä, mutta estää yhdennentoista yhtey-

den muodostamisen ohjelmistoon. Kun mikä tahansa aikaisemmasta 10 yhteydestä puretaan, mahdollistaa apuohjelma taas yhden uuden yhteyden luomisen.

Tällaista lisensointimallia käyttäen lisenssinostaja saattaa ratkaista yhden käyttäjän lisenssin aiheuttaman ongelman kalliimpien lisenssien kohdalla siten, ettei työntekijöiden enää tarvitse vaihtaa tietokoneita keskenään päästäkseen käyttämään lisensoitua ohjelmistoa [Ingram, 2006]. Silti tämäkään lisensointimalli ei ratkaise sesonkiaikojen kasvavaa käyttötarvetta koskevaa ongelmatiikkaa.

#### **4.1.3. Aikarajoitteinen lisenssi**

Usein ostaja haluaa myös kokeilla ohjelmistoa ennen sen ostamista. Tätä tarkoitusta varten on kehitetty aikarajoitteinen lisenssi (time limited license). Se on eräänlainen ”demolenssi”, jossa asiakas saa käyttöönsä täysin toimivan ohjelmiston tietyn ajanjaksoksi, yleensä yhdeksi, kolmeksi tai kahdeksitoista kuukaudeksi. [Ingram, 2006]

Lyhytaikaisen lisenssin ostaminen toimii hyvin silloin, kun asiakas haluaa ohjelmiston käyttöönsä vain tietyssä vuodenaikana esimerkiksi joulusesongin ajaksi tai jos asiakas haluaa testata ohjelmaa ennen varsinaisen ostopäätöksen tekemistä. Ostaja voi myös haluta jakaa ohjelmiston käytöstä aiheutuvat kulut pienempiin osiin yhden suuren kertaostoksen sijaan. [Ingram, 2006]

Tämä malli ratkaisee yhden käyttäjän lisenssimallin sekä monen käyttäjän jaetun lisenssimallin sesonkeihin liittyvän perusongelman, mutta jättää vielä toisen ongelman jäljelle. Nimittäin asiakkaan kaikki työntekijät eivät välttämättä ole keskenään tasaväkisessä asemassa ohjelmistoon nähden. Jotkut henkilöt tarvitsevat ohjelmaa useammin tai monipuolisemmin kuin toiset. Aikaan sidotussa lisensointimallissa kaikki ostajan lisenssit koskevat tuotetta samanlaajuisena, vaikka ohjelmiston käyttötarve vaihtelee myös ohjelmiston sisältämän toiminnallisuuden näkökulmasta.

#### **4.2. Luokittelumallit**

Käyttäjien luokitteluun perustuva malli (classification model) on kehitetty yhden nimetyn käyttäjän mallista. Tällaisessa mallissa ei nimetä ketään tiettyä käyttäjää vaan käyttäjät jaetaan luokkiin tai ryhmiin, joilla on keskenään tasa-arvoinen oikeus (tai ei ole oikeutta) johonkin tiettyyn ohjelmistoon, komponenttiin tai toimintoon. Rajoitus voi koskea myös kestoa, jonka ajan jokin tietty toiminnallisuus on käytettävissä tai kuinka monta yhteyttä tiettyyn toiminnallisuuteen voi olla aktiivisena jonkin ajanjakson, esimerkiksi päivän, viikon tai kuukauden aikana. [Ingram, 2006]

Ohjelmiston käyttäjät ostavat erilliset lisenssit jokaiselle eri käyttäjäryhmälle. Usein myyjä määrittelee eri kategoriat sopimusneuvotteluiden aikana. Tämä taas vaatii käyttäjien toimintojen analysointia sen varmistamiseksi, että kaikki käyttäjät saadaan sijoitettua sopivaan luokkaan. [Ingram, 2006]

Maksa käytöstä -vuokrausmallissa (pay-per-use rental model) kyse on siitä, että myyjä antaa asiakkaan käyttöön eli vuokraa tälle enemmän lisenssejä kuin mitä tämä on ostanut. Kun asiakas käyttää ohjelmistoa, päivitetään samalla lokitiedostoa, johon kerätään tietoa ohjelmiston käyttöasteesta, joko käytetystä ajasta tai vaihtoehtoisesti käytettyjen tapahtumien eli transaktioiden määrästä. Tämä tehdään sekä ostetuista että vuokratuista ohjelmista. Kerran jokaisen laskutusyökin aikana asiakas toimittaa lokitiedoston lisenssinantajalle, joka analysoi tiedot. Jos lisenssinostaja on käyttänyt ohjelmistoa enemmän kuin ostamiensa lisenssien kattaman määrän verran, hän joutuu maksamaan lisämaksun. [Ingram, 2006]

Lisenssinostajalle tällaisesta lisensointimallista on se etu, että hänen ei tarvitse ostaa lisenssejä maksimikäyttötarpeensa mukaan, vaikka hänellä onkin heti saatavillaan lisää ohjelmistokapasiteettia, kun suurempi tarve ilmenee. Tällöin ostaja maksaa lisämaksun vain käyttämästään ylimääräisestä ohjelmistokapasiteetista.

Lisenssinantajalle tästä on se etu, että joustava ratkaisu houkuttaa useampia asiakkaita ostamaan omakseen edes osan lisensseistä, sillä heille tuo turvaa tietoa siitä, että lisätarpeen tullen heillä on mahdollisuus käyttää ohjelmistoa enemmän kuin ostamiensa lisenssin määrän verran. Lisäksi lisäkäytöstä aiheutuvat maksut on myös yleensä hinnoiteltu korkeammiksi kuin peruslisenssien hinta, joten lisenssinantaja saa silti maksun tuotteestaan, vaikka ei saisikaan myytyä yhtä monta kappaletta lisenssiä kuin asiakas joutuisi ostamaan, jos käytettäisiin jotakin jäykemmistä lisensointimalleista.

Aikaan sidotussa pay-per-use -mallissa (time based pay-per-use model) lokitiedostoon tallennetaan tietoa siitä, kuinka kauan aikaa ostaja on käyttänyt niitä ohjelmistolisenssejä, joita hän ei ole ostanut itselleen [Ingram, 2006]. Laskutus siis perustuu ohjelmiston käyttöaikaan eikä niinkään ohjelmistokomponenttien tai transaktioiden määrään.

Transaktioihin perustuvassa pay-per-use -mallissa (transaction based pay-per-use model) tallennetaan tietoa käytettyjen ohjelmistokomponenttien tai toimintojen lukumäärästä. Tässä mallissa käytön kesto aika ei ole oleellinen. Siksi tätä lisensointimallia suositellaan käytettäväksi silloin, kun eri komponenttien tai toimintojen käyttöaika on aina suurin piirtein sama. [Ingram, 2006]

### **4.3. Sekoitettut lisensointimallit**

Kaikkien edellä mainittujen mallien ongelma on, että ne eivät reagoi tarpeeksi nopeasti muuttuviin markkinatilanteisiin. Sekoitetuissa lisensointimalleissa (remix license models) luodaan läheisempi kumppanuussuhde lisenssinantajan ja lisenssinostajan välille. Idea on se, että lisenssinostaja tekee investoinnin lisenssinantajan omistamaan teknologiaan yleisesti sen sijaan, että ostaisi jonkin tietyn ohjelmiston lisenssin. [Ingram, 2006]

Tällaisessa lisensointimallissa ostaja saa käyttöönsä laajan valikoiman lisenssinantajan ohjelmistotuotteita sekä mahdollisuuden muuttaa lisenssien määrää omien tarpeidensa

mukaan. Järjestely toimii erityisen hyvin silloin, kun lisenssinostaja haluaa pitkän aikavälin lisäarvoa halpojen alkukustannusten sijaan. [Ingram, 2006]

Sekoitetuissa malleissa on käytössä kaksi eri lähestymistapaa. Ensimmäinen on staattinen sekoittaminen (static remixing), jossa tarvittavien lisenssien määrä ja laajuus tarkistetaan säännöllisesti esimerkiksi 2 – 4 kertaa vuodessa. Tällöin jaksojen aikana kerätään tilastoa kunkin ohjelmiston käytöstä, ja tätä tietoa käytetään perustana mietittäessä seuraavan jakson lisenssitarvetta. [Ingram, 2006]

Toinen malli on jatkuva käytönaikainen sekoittaminen (continuous real-time remixing), josta käytetään myös termiä merkkeihin perustuva lisensointi (token based licensing). Sen sijaan, että asiakas ostaisi lisenssin johonkin tiettyyn ominaisuuteen tai toiminnallisuuteen, hän ostaakin lisenssin tiettyyn määrään merkkejä. Jokainen yksittäinen ohjelmiston ominaisuus vastaa tiettyä määrää merkkejä. Tällöin asiakas voi oman tarpeensa mukaan valita myyjän valikoimasta niin monta ominaisuutta kuin hänellä on ostettuja merkkejä. [Ingram, 2006]

#### **4.4. Teknologiaikumppanuus**

Täydellinen teknologiaikumppanuus (technology partnerships) edellyttää, että asiakkaalla on rajoittamaton käyttöoikeus kaikkiin myyjän tarjoamiin tuotteisiin tai sopimuksessa erikseen määritelyyn osaan tuotteista. Joskus tällaisista sopimuksista käytetään myös nimitystä ”buffetti” (buffet) tai ”syö-niin-paljon-kuin-jaksat” (all-you-can-eat). [Ingram, 2006]

Tällainen malli on erittäin palkitseva sekä myyjälle että ostajalle, sillä asiakkaan markkinatilanteen muuttuessa myyjä voi heti tarjota sopivaa teknologiaa entisten ratkaisujen tilalle. [Ingram, 2006]

Asiakas saa hyvää palvelua myyjältä ja myyjä saa takuuvarmat tuotot. Sopimus on monivuotinen, ja asiakas maksaa sovitun vuosittaisen maksun siitä hyvästä, että saa käyttää myyjän ohjelmistoja rajoittamattomasti sopimuksessa määritellyin osin. Maksettavien lisenssimaksun määrä tarkistetaan vuosittain pay-per-use -analyysien pohjalta. [Ingram, 2006]

#### **4.5. Avoimen lähdekoodin lisenssejä**

Saastamoinen [2006] kuvaa avoimen lähdekoodin ohjelmistoja sekä näiden lisenssejä.

Avoimen lähdekoodin ohjelmistoja voidaan levittää usealla eri tavalla. Ilmaisohjelma (freeware) on yleisnimitys ohjelmalle, jonka on saatavilla ilmaiseksi. Silti ohjelman käytössä, muuntelussa ja edelleenlevityksessä voi olla rajoituksia. Tästä erona ovat vapaat ohjelmistot (free software), joilla näitä rajoituksia ei ole.

Osuusohjelmien (shareware) idea perustuu siihen, että ohjelman tilapäisestä käytöstä ei tarvitse maksaa, mutta pysyvästä käytöstä kyllä. Julkisohjelmat (public domain) ovat



kaikille vapaasti saatavissa, käytettävissä ja osittain jopa myytävissä edelleen. Tällaisia teoksia ovat mm. musiikkikappaleet, joiden tekijänoikeus on vanhentunut. Taiteellisille teoksille kuten musiikki-, elokuva- ja kirjallisuusteoksille on myös oma levitystapansa Creative Commons.

Avoimen lähdekoodin lisensseillä on kaksi ominaisuutta, joita ei ilmene suljetun lähdekoodin lisensseissä. Nämä ovat pysyvyys (persistence) ja tarttuvuus (inheritance). Pysyvyys tarkoittaa sitä, että jos ohjelmistoa kehitetään edelleen, niin myös nämä uudet ohjelmat on lisensoitava saman lisenssin alaisuudessa kuin alkuperäinenkin ohjelma. Tarttuvuus tarkoittaa sitä, että jos tarttuvan lisenssin ohjelmakoodia yhdistellään toiseen avoimen lähdekoodin ohjelmaan, niin tarttuva lisenssi syrjäyttää toisen ohjelman lisenssin, jolloin yhdistelmä saa lisenssiksensä tarttuvan lisenssin. Näiden rinnalla on myös käytössä sallivia lisenssejä (permissive), jotka eivät rajoita ohjelmakoodin käyttöä ja uusien ohjelmaversioiden lisenssointia juuri lainkaan.

Kuten taulukosta 1 näkyy, lisääntyvät käyttäjän oikeudet mentäessä yksinoikeuslisensseistä jakelutyyppien kautta avoimen lähdekoodin lisensseihin.

Lisenssi tai levitystapa	Vapaa levitys	Vapaa käyttö	Vapaa muuntelu	Avoin koodi	Pysyvä	Tarttuva
Yksinoikeuslisenssi	-	-	-	-	-	-
Shareware	X	-	-	-	-	-
Freeware	X	- / X <sup>1</sup>	-	-	-	-
Public domain	X	X	X <sup>2</sup>	-	-	-
BSD, MIT, Apache, Artistic	X	X	X	X	-	-
LGPL, MPL	X	X	X	X	X	-
GPL	X	X	X	X	X	X

<sup>1</sup> Freeware -ohjelma voi olla vapaa kaikkeen käyttöön, mutta se voi sisältää myös ehtoja, jotka esimerkiksi rajoittavat käytön vain ei-kaupalliseen toimintaan.

<sup>2</sup> Public domain -ohjelman muuntelu vaikeutuu suuresti, jos ohjelman lähdekoodi ei ole saatavilla  
Taulukko 1. Tekijänoikeuden laajuus eri lisenssi- ja jakelutyypeissä. [Saastamoinen, 2006, s. 39]

GPL (GNU General Public License) on suosituin avoimen lähdekoodin lisenssi. Varmaankin tunnetuin ohjelmisto, joka on julkaistu tämän lisenssin alla, on GNU/Linux-käyttöjärjestelmä. GPL-lisenssin lähtöajatus on, että tämän lisenssi alla julkaistavat ohjelmistot ovat vapaasti kaikkien käytettävissä ja myös näistä edelleenkehitetty ohjelmistot säilyvät vapaana. GPL ei aseta rajoituksia ohjelman käytölle, mutta ohjelman kopioiminen, muokkaaminen tai levittäminen edellyttää, että lisenssiehdot hyväksytään. Koska GPL-lisenssi on sekä tarttuva että pysyvä, sitä on kritisoitu mm. siitä, että se ”saastuttaa” koko ohjelmiston määräämällä, että uusi ohjelmisto lisensoidaan myös GPL:ksi.

LGPL (GNU Lesser General Public License) oli alkuperäiseltä nimeltään library general public license, sillä alun perin lisenssi oli tarkoitettu ohjelmakirjastojen levitykseen. LGPL:n suurin ero GPL:ään on se, että LGPL ei ole tarttuva vaan ainoastaan pysyvä. Ehkä tunnetuin LGPL-lisenssin alla julkaistu ohjelmisto on OpenOffice.org:n julkaisema toimisto-ohjelmisto.

Netscape Communications loi MPL-lisenssin (Mozilla Public Lidence) vuonna 1998, kun se tarvitsi lisenssin uudelle Communicator-WWW -selaimelleen. MPL-lisenssejä saa myöntää vain voittoa tavoittelematon Mozilla-säätiö, jonka tunnetuimpia tuotteita ovat mm. Mozilla Suite -internet-ohjelmisto, Firefox -WWW-selain sekä Thunderbird-sähköpostiohjelmisto. MPL-lisenssi on pysyvä, mutta ei tarttuva.

BSD (Berkeley Software Distribution) on vanhin ja myös kolmanneksi suosituin avoimen lähdekoodin lisenssi heti GPL:n ja LGPL:n jälkeen. Lisenssin ehdot ovat varsin lyhyet ja ne on yleisesti tulkittu niin, että ne antavat lähes rajattoman vapauden käyttää, muunnella, myydä ja kopioida sekä alkuperäistä teosta että siitä muunneltuja teoksia sillä ehdolla, että binäärikopioissa on maininta tekijästä ja BSD-lisenssistä. BSD-lisensoidut ohjelmat ovat houkuttelevia kohteita sellaisille yrityksille, jotka haluavat kehittää yksinoikeuslisensoitavia tuotteita, sillä BSD ei ole tarttuva eikä pysyvä. MIT-lisenssi (Massachusetts Institute of Technology) on kehitetty BSD:n pohjalta. Tavoitteena on ollut tehdä BSD:n kaltainen lisenssi, joka olisi helpommin luettava ja ymmärrettävä.

Apache-lisenssit myöntää voittoa tavoittelematon Apache Software Foundation, jonka tuotteet ovat erityisen suosittuja ohjelmistokehittäjien ja WWW-palveluntarjoajien keskuudessa. Tunnetuin Apache-lisenssillä lisensoitu ohjelmisto on maailman suosituin HTTP-palvelinohjelmisto Apache HTTP Server. Apache-lisenssi on hyvin samankaltainen salliva lisenssi kuin BSD, mutta se on kirjoitettu huolellisemmin ja yksityiskohtaisemmin ja se myös sisältää BSD:hen verrattuna joitakin lisäehtoja ja tarkennuksia.

Myös PHP-lisenssejä ylläpitää voittoa tavoittelematon järjestö, The PHP Group. Lisenssi luotiin alun perin PHP-ohjelmointikielen levittämistä varten ja se on myös BSD:n tyyliin salliva. Lisenssin ehkä näkyvin ehto edellyttää, että PHP-lisenssin alla julkaistulla WWW-sivustolla pitää olla PHP:n mainos.

Artistic-lisenssin nimi viittaa sen kehittäjän tavoitteeseen säilyttää teoksen taiteellinen kontrolli sen alkuperäisellä kehittäjällä. Lisenssi luotiin Perl 4.0 -ohjelmistoa varten, kun tekijät halusivat säilyttää Perl-projektin hallinnan itsellään, mutta rohkaista samalla uusia jäseniä liittymään projektiin tai muuten vain kehittämään ohjelmaa. Lisenssiä ei siis alun perin ole tarkoitettu avoimen lähdekoodin lisenssiksi eikä edes yleiseen käyttöön.

## 5. Tapaustutkimuksen lähtökohdat

### 5.1. Myyjäyritys

Ohjelmistojensa tuotteistusta suunnitteleva yritys on pieni, kotimainen, alle sadan henkilön yritys, jonka omistavat kaksi suurta konsernia. Yrityksen historia on varsin pitkä, mutta itsenäisenä yrityksenä se on toiminut vasta vuodesta 2006 lähtien. Omistajiltaan yritys saa käyttöönsä laajan kirjon palveluita, kuten esimerkiksi myyntikoneiston sekä yhteyshenkilöt, joilla on suhteita niin kotimaisiin kuin kansainvälisiin asiakasehdokkaisiin.

Yrityksellä itsellään ei ole aikaisempaa kokemusta ohjelmistojensa tuotteistuksesta, sillä tähän asti se on tehnyt räätälöityjä ohjelmistoja suoraan asiakkailleen. Sen sijaan yrityksen omistajilla on valikoimissaan myös erilaisia tuotteistettuja ohjelmistoja, joten tätä tietotaitoa on mahdollista myyjäyrityksenkin käyttää.

### 5.2. Asiakasprofiili

Ohjelmiston tulevat asiakkaat toimivat pääosin valtion turvallisuudesta vastaavissa organisaatioissa. Ohjelmisto tarjoaa heille kriittisen infrastruktuurin turvaamisessa tarvittavaa informaatiota. Tästä lähtökohdasta on mahdollista tehdä asiakasprofiili mahdollisista tulevista asiakkaista. Asiakasprofiili auttaa hahmottamaan asiakkaan tilannetta ja näkökulmaa sekä helpottaa tulevan lisenssin ehtojen muotoilemista siten, että se olisi molempia osapuolia hyödyttävä.

Tulevan ostajan asiakasprofiili sisältää mm. seuraavankaltaisia asioita:

- organisaatioiden suuruus
- byrokratian määrä
- päätöksenteon hitaus
- valtiosalaisuudet ja muu salassapito
- budjetoinnin luonne ja tuotteen hinnoittelu
- henkilöresurssit
- teknologiaosaaminen.

Käytännössä profiilin kuvaamat asiat tarkoittavat sitä, että lisenssiehtojen pitää taipua asiakkaan erikoislaatuiseen asemaan markkinoilla sekä organisaatioiden ehkä raskaaseenkin hierarkiaan. Todennäköiset asiakkaat ovat luultavasti valtioita tai niiden puolustuselimiä tai valtion valvonnassa toimivia suuryrityksiä. Tällaisissa organisaatioissa on usein paljon byrokratiaa, jolloin päätöksenteko ja toiminta voivat hidastua. Tuotteen myyntiprosessissa tähän on varauduttava siten, että tarjouspyynnöstä kaupan syntymiseen ja tuotteen käyttöönottoon voi kulua hyvin jopa useita vuosia ja lisenssin on oltava koko

tämän prosessin ajan lähestulkoon muuttumaton, mutta myös vastattava sekä sen hetkistä että myös tulevaa markkinatilannetta ja tietenkin myös mahdollisesti muuttuvaa lainsäädäntöä sekä normatiivisia ohjeita.

Byrokratian määrä sekä organisaation suuruus ja hierarkkisuus heijastuvat myös asiakkaan budjettiin. Budjetti suunnitellaan usein pitkälle aikavälille, ja esimerkiksi valtion tietyille organisaatioille myöntämät vuosittaiset määrärahat ovat rajalliset. Lisenssin on siis katettava sellainen toiminnallisuuden ja palvelujen osa ohjelmistosta, että lopullinen hinta on sekä ajoissa ennakoitavissa, kohtuullisen muuttumaton ja että sen sovittaminen budjettiin onnistuu mahdollisimman sulavasti.

Tällaisilla asiakkailla on myös suuri tarve pitää huolta siitä, etteivät valtiosalaisuudet tai muu salaiseksi luokiteltu tieto pääse vuotamaan ulkopuolisten käsiin. Tämän vuoksi esimerkiksi ohjelmiston käyttömäärien tai käyttöasteen seuranta lokitiedostojen avulla ei onnistu, sillä yksikään asiakas ei suostuisi luovuttamaan arkaluonteista materiaalia myyjäyritykselle analysoitavaksi. Lisenssin hinnoittelun pitää siis perustua muunlaisiin laskutusperusteisiin.

Valtion omistamat organisaatiot ovat myös yleensä tarkkoja henkilöresursseistaan. Henkilöstöön kuuluu vain organisaation oman toimialan ammattilaisia, eikä ohjelmiston käyttöönottoon tai ylläpitoon voida välttämättä osoittaa sellaista henkilöä, jolla olisi tarvittavaa ammattitaitoa tällaisiin töihin. Lisenssissä on siis huomioitava tarkoin, mikä kaikki käyttöönottoon ja ylläpitoon kuuluva työ tehdään asiakkaan ja mikä myyjäyrityksen toimesta. Myöskään tuotteen toiminnallisuus tai käyttöliittymä ei saa muuttua esimerkiksi ohjelmistopäivitysten myötä liian paljon, sillä käyttäjien koulutukseen ei välttämättä löydy tarvittavia resursseja.

Vielä eräs asiakkaasta huomioitava seikka on, että myyjäyrityksellä ei ole tietoa asiakkaan teknologisesta osaamisesta. Myös tämä vaikuttaa siihen, millaiseksi yhteistyö myyjän ja asiakkaan välillä muodostuu.

### **5.3. Ohjelmiston luonnehdinta**

Yrityksellä on vahva näkemys, jonka mukaan se voisi tuotteistaa osan tekemistään räätälöidyistä ohjelmistoista ja myydä niitä myös muiden valtioiden puolustusorganisaatioille tai mahdollisesti tulevaisuudessa myös siviiliturvapuolelle, kuten satamien, lentokenttien tai ydinvoimaloiden valvontaan.

Ohjelmisto on alun perin valtion turvallisuuteen liittyvälle organisaatiolle suunniteltu ja räätälöity kokonaisuus, joka koostuu useista pienistä komponenteista. Alkuperäinen idea ja ohjelmiston vanhimmat osat ovat jo noin kaksikymmentä vuotta vanhoja, joten ne on aika toteuttaa uudemmalla teknologialla. Samalla kun ohjelmistoa uudelleen kirjoitetaan, tehdään sellaisia muutoksia, jotka sallivat ohjelmiston tuotteistuksen ja myynnin myös komponenteittain.

Ohjelmisto on pitkälle räätälöity ja komponenttien integrointi eli yhteensovittaminen niin hankalaa, että tuotteesta ei käytännön syistä voida tehdä niin kutsuttua off the shelf -tuotetta vaan joka tapauksessa kullekin asiakkaalle on tehtävä omia muutoksia ohjelmistoon. Tulevaa tuotelisenssiä valittaessa tämä seikka on otettava huomioon, sillä muutosten tekeminen ja hallinta vievät resursseja sekä myyjältä että ostajalta, ja tämän vuoksi vastuunjaosta on sovittava selkeästi.

Ohjelmisto koostuu ohjelmistoverkostosta, jossa osa toimipaikoista on kiinteitä ja osa liikkuvia yksiköitä. Ohjelmisto on suunniteltu siten, että kiinteitä toimipaikkoja ja liikkuvia yksiköitä voi olla lähes rajaton määrä ja ne kaikki yhdessä muodostavat verkon, jonka solmut synkronoituvat keskenään, keskustelevat toistensa kanssa ja jakavat tietonsa. Tarpeen tullen verkosta voidaan erottaa yksi tai useampia liikkuvia solmuja joskus pitkäksikin aikaa. Irrotetut solmut pysyvät täysin toimintakelpoisina ja ne voidaan myös tarvittaessa liittää takaisin verkkoon.

Lisensoinnille ohjelmiston rakenne tuo sekä rajoituksia että selkeitä päätöksiä. Yksi selvä asia on se, että tuote on lisensoitava komponentti kerrallaan. Todennäköisin ratkaisu on myydä sekä tukiasemaan että liikkuvaan yksikköön peruspakettia sekä näiden lisäksi vielä muita komponentteja, jotka asiakas saa halutessaan ostaa perustoiminnallisuutta monipuolistamaan.

Haaste lisensoinnille muodostuu siitä, voiko myyjäyrittäjä määrätä, kuinka monta toimipaikkaa yhdellä lisenssillä saa muodostaa. Pohditaan kuvitteellista tilannetta, jossa jokin toinen valtio haluaa ostaa ohjelmistolisenssin tuotteeseen: Todennäköistä on, että tällöin ostajan intresseissä ei ole kertoa myyjäyrittäjälle, montako toimipaikkaa se on perustamassa. Tämähän on valtiosalaisuus ja tärkeä tieto maan puolustuksen kannalta. Sama pätee myös ohjelmiston käyttöä dokumentoivaan lokitiedostoon. On epärealistista kuvitella, että mikään valtio tai muu asiakas suostuisi paljastamaan näin arkaluonteista materiaalia myyjäyrittäjälle. Lisenssiehdoissa siis laskutus on tapahduttava perustuen johonkin muuhun kuin asennettujen ohjelmistojen lukumääriin tai ohjelmiston käyttöön. Todennäköisin ratkaisuvaihtoehto on, että tuote maksetaan etukäteen ja sen jälkeen asiakas saa täydet oikeudet asentaa ohjelmisto niin usealle alustalle kuin haluaa. Myyjäyrittäjälle tämä tarjoaa jatkomyyntimahdollisuuden versiopäivitysten, ylläpidon, koulutuksen ja muiden vastaavien palveluiden myynnistä.

Tämänkaltaiset ohjelmistot ovat myös elinkaareltaan varsin pitkiä, usein jopa 10 – 20 vuotta. Myös ohjelmiston täysimittainen käyttöönotto voi venyä useiden vuosien mittaiseksi. Tämä asettaa haasteita niin myyjäyrittäjän kokoonpanonhallinnalle kuten myös valittavalle lisenssille. Lisenssiehtojen tulee tukea myyjäyrittäjän versionhallintaa mm. siten, että jossain vaiheessa vanhojen ohjelmaversioiden tuki lakkaa ja asiakkaan on siirryttävä käyttämään uudempia ohjelmaversioita. Lisenssiehdoissa on myös selkeästi sovittava, kuinka monta ohjelmaversiota taaksepäin ohjelmiston osien on oltava yhteensopivia.

## 6. Lisenssiltä vaadittavia ominaisuuksia

Pohdittaessa lisenssiltä vaadittavia ominaisuuksia kaikki tiedot myyjärytyksestä, asiakasprofiilista ja ohjelmistosta itsestään on otettava huomioon. Lisenssin perimmäinen tarkoitus on selkeyttää vastuunjakoa ja varmistaa sekä myyjälle kuuluvat tuotot että asiakkaalle hänen rahojensa vastineeksi saamansa hyödyt. Vain sekä myyjän että asiakkaan molempien näkökulmat huomioon ottaen on mahdollista solmia ohjelmiston elinkaaren mukaisia, pitkäaikaisia asiakassuhteita.

Toimittajalla on kaksi erilaista lähtökohtaa lisensointiin. Toisaalta hän on kiinnostunut ohjelmiston myynnin aiheuttamista taloudellisista vaikutuksista ja toisaalta ylläpidon muodostamista tarpeista. Toimittajan perimmäinen motivaatio tuotteistaa ja myydä ohjelmistoaan on liiketalouden perusarvojen mukaan tuottojen saaminen ja hyvän taloudellisen tuloksen varmistaminen myös pitkällä aikavälillä. Toimittajan kannalta lisenssiehtojen tulee siis tukea tasaisia ja pitkäaikaisia tulovirtoja samalla, kun sellaisten rasitteiden, kuten versionhallinnan, ylläpidon ynnä muun hoitaminen, pitäisi pystyä tekemään mahdollisimman vähillä resursseilla ja pienin kuluin. Asiakas taas haluaa lisenssiehtojen varmistavan, että hänellä on rahojaan vastaava, laadukas tuote käytössään ja että ongelmien tullen hän saa yritykseltä apua tukipalveluiden muodossa.

### 6.1. Toimittajan taloudellisia kysymyksiä

Seuraavaksi kuvataan toimittajalle tärkeitä, lisenssin ominaisuuksiin liittyviä kysymyksiä kustannustehokkuuden näkökulmasta.

#### 6.1.1. Miten varmistetaan maksimituotot?

Jos toimittaja saisi muista tekijöistä riippumattomasti päättää tuotteensa lisenssiehdoista, hän todennäköisesti pyytäisi tuotteesta mahdollisimman korkean hinnan kalliilla etumaksulla sekä suurilla kausimaksuilla eikä suostuisi minkäänlaiseen ylläpitotoiminnallisuuteen tai ainakin laskuttaisi myös siitä ankarasti.

Tämä ei kuitenkaan käytännössä voi toteutua, sillä asiakas ei varmasti suostuisi hyväksymään tämänkaltaisia lisenssiehtoja. Varmistaakseen maksimituotot on toimittajan tehtävä myönnytyksiä niin hinnoittelussa kuin tuotteen elinkaaren eri vaiheissa tarvittavissa huoltotöissäkin.

Nykypäivän markkinoilla paras tapa varmistaa tasaiset tuotot onkin tarjota varsinaisen tuotteen lisäksi palveluita, kuten esimerkiksi asennusta, koulutusta ja versiopäivityksiä, ja periä näistä vain kohtuullista hintaa. Tämä luo positiivista ja yhteistyökykyistä kuvaa toimittajasta, jolloin asiakas haluaa sitoutua pidemmäksi aikaa yritykseen ja ostamaansa lisensoituun tuotteeseen, mikä puolestaan varmistaa myös toimittajalle tasaisen tulovirran.

### 6.1.2. Miten minimoidaan kulut?

Maksimituottojen lisäksi yrityksen tuloksen kannalta on tärkeää myös minimoida kulut. Kaikki ylimääräiset toiminnot ja palvelut varsinaisen tuotteen päälle lisäävät toimittajan menoja ja syövät myynnistä saatuja tuottoja. Esimerkiksi versiopäivitykset vaativat sekä henkilöresursseja että aikaa ja rahaa. Jonkunhan pitää ensin suunnitella, mitä uutta seuraava versio sisältää, jonkun pitää tehdä tarvittavat tekniset muutokset, tämän jälkeen ohjelmaversio pitää toimittaa asiakkaalle, joka mahdollisesti tarvitsee vielä lisäksi apua asennuksessa, datakonversioissa ja koulutuksessa.

Lisenssiehdoissa voidaan myös päättää, saako yhdellä asiakasyrityksellä olla monta eri ohjelmaversiota käytössään vai pitääkö hänen pitäytyä vain yhdessä versiossa. Mitä enemmän erilaisia versioita ohjelmistosta on käytössä, sitä hankalampaa niiden hallinta on toimittajalle. Joka tapauksessa on varsin epätodennäköistä, että eri asiakkailla olisi käytössään keskenään samat ohjelmaversiot, joten ylläpitotyötä varmasti tulee paljon siltikin, vaikka yhdellä asiakkaalla olisi vain yksi ohjelmaversio.

Lisenssiehtojen kannalta on erityisen tärkeää miettiä, miten kuluja aiheuttavien toimintojen vastuunjako määritellään asiakkaan ja toimittajan välillä. Lisenssiehdoissa voidaan esimerkiksi sopia, että asiakkaan on suostuttava version nostoon aina, kun toimittaja sellaisen tekee. Tämä vähentää niin datakonversioiden tarvetta kuin myös toimittajan kokoonpanonhallintaa, koska asiakkaalla olevaa ohjelmaversiota ei koskaan tarvitse päivittää kuin yhden pykälän ylöspäin. Samoin lisenssiehtoihin voidaan kirjoittaa sääntö, että toimittaja esimerkiksi lupautuu antamaan tietyn tuntimäärään koulutusta jokaisen versionoston yhteydessä, ja jos asiakas haluaa sitä enemmän, hän joutuu maksamaan siitä erikseen.

### 6.1.3. Miten varmistetaan asiakassuhteen pysyvyys?

Molemmille sopimuksen osapuolille on hyötyä siitä, että kumppanuussuhde jatkuu pitkään. Molemmat yritykset, toimittaja ja asiakas, oppivat tuntemaan toistensa toimintatavat ja yrityskulttuurin.

Toimittajalle pitkä asiakassuhde takaa tasaisen tulovirran, asiakas taas puolestaan saa aina vain paremman ja toimivamman ohjelmiston käyttöönsä versiopäivitysten ja ylläpitotuen muodossa. Koska myös tuotteen elinkaari on pitkä, joustava sopiminen hinnoittelusta ja oheispalveluista on tarpeen. Tämä onnistuu, kun osapuolten välillä vallitsee luottamus, vastavuoroisuus ja hyvä yhteistyöilmapiiri.

Siispä kannattaakin jo lisenssiehtojen sopimisvaiheessa valita sellaisia ratkaisuja, jota tukevat teknologiakumppanuutta ja mahdollistavat joustavan yhteistyön yritysten välillä.

#### **6.1.4. Miten järjestetään myyntikanava tai useat myyntikanavat?**

Toimittajan täytyy myös miettiä, miten tuotteistettu ohjelmisto saadaan jaettua tuleville asiakkaille. Jos yrityksellä itsellään on myyntikoneisto, niin sen ylläpito tuo myös lisäkustannuksia yritykselle. Toisaalta, oma myyntiorganisaatio on helposti ja nopeasti tavoitettavissa ja asioista voidaan yleensä sopia yrityksen taloudellinen hyvinvointi sekä asiakastyytyväisyys mielessä pitäen.

Jos taas yrityksellä ei ole tietotaitoa tai resursseja ylläpitää omaa myyntitoimintaa, sen on hyväksyttävä palvelun ostamisesta kolmannelta osapuolelta aiheutuvat kulut. Jos kolmas osapuoli on jälleenmyyntiyritys, toimittaja tekee erillisen lisenssisopimuksen tämän kanssa. Tässä sopimuksessa sovitaan, saako jälleenmyyjä tehdä omia muutoksiaan ohjelmistoon ja kuinka paljon jälleenmyyjä maksaa alkuperäiselle yritykselle myymistään ohjelmakopioista.

Yrityksen on myös hyvä miettiä, riittääkö yksi myyntikanava sen tarpeisiin vai tarvittaisiinko niitä useita. Yritys voi esimerkiksi sekä myydä tuotetta itse että antaa sen kolmannelle osapuolelle myytäväksi. Tai jälleenmyyjä voi tuotteella olla useita esimerkiksi eri asiakasryhmille kohdistettuina.

#### **6.1.5. Miten varmistetaan lisenssiehtojen joustavuus?**

Tuotteen elinkaari on pitkä, ehkä jopa 20 vuotta. Elinkaarta pidentää myös se seikka, että tarjouspyynnöstä kaupan syntymiseen voi viedä vuosia ja senkin jälkeen ohjelmiston varsinaiseen käyttöönottoon voi lisäksi viedä vielä useita vuosia lisää.

Lisenssiehtojen kannalta tämä on merkittävä seikka, sillä molemmat osapuolet muuttuvat varmasti tänä aikana. Myös markkinat ehtivät muuttua huomattavasti näin pitkän ajan kuluessa.

Lisenssiehtojen on siis oltava niin joustavat, että ne soveltuvat paitsi nykyiseen markkinatilanteeseen niin myös tulevaisuuden mukanaan tuomiin muutoksiin. Ehtojen kirjoittajalta vaaditaan siis myös kykyä ennakoida tulevaisuutta siten, että ehdot ovat validit kaupan suunnittelusta ohjelmiston elinkaaren loppuun asti.

Yleensä tällaisia muutostarpeita aiheuttavia tilanteita varten lisenssiehtoihin lisätään lause, että toimittaja voi tarkastaa lisenssiehtoja tietyin määräajoin tai tietyin perustein. Toimittajalle tämä tietysti tuo turvaa, koska näin hän voi valmistautua tiettyihin vaikeisiin tilanteisiin ennalta ja muuttaa tarvittaessa lisenssiehtoja omaksi hyväkseen. Asiakkaalle tämä taas aiheuttaa jatkuvaa epävarmuutta siitä, ovatko tämänhetkiset lisenssiehdot voimassa vielä ensi vuonna ja jos ehdot muuttuvat, niin huonontavatko ne sen asmaan niin paljon, ettei sopimus ole enää kannattava.



### **6.1.6. Miten tehdään kokoonpanonhallinta kustannustehokkaasti?**

Ohjelmiston pitäminen toimintakunnossa eri asiakkailta 20 vuoden ajan on yritykselle suuri haaste. Toimittajan on pystyttävä pitämään kirjaa siitä, mikä versio kenelläkin ohjelmistosta on sekä ylläpidettävä kaikkia näitä versioita.

Ylläpitoon kuuluu virheiden korjausta, osittain myös uusien ominaisuuksien kehittämistä sekä datakonversioita versionnostojen aikana. Myös asennuspalvelut, integroinnit muiden ohjelmistojen tai uusien laitteiden kanssa sekä mahdollisesti annettava koulutus kuuluvat niin ikään ohjelmiston ylläpitoon.

Kokoonpanonhallintaan sen sijaan kuuluu asiakkaiden ostamien lisenssien hallinta. Toimittajan pitää tietää, kenellä asiakkaalla on mikäkin versio ohjelmistosta ja mitä lisäkomponentteja asiakas on tilannut perustoiminnallisuuden päälle. Tämän lisäksi toimittajan on myös huolehdittava siitä, että kaikki ostetut lisäpalvelut tulevat tehdyiksi ajallaan.

Koska kaikki nämä aiheuttavat toimittajalle resurssipaineita niin henkilöstön kuin ajan ja kustannustenkin muodossa, näitä asioita on pohdittava hyvissä ajoin tuotteistusprosessin alusta lähtien. On tärkeää miettiä, mikä olisi kustannustehokkain keino järjestää ylläpito ja kokoonpanonhallinta.

Lisenssiehdot vaikuttavat epäsuorasti myös ylläpidon ja kokoonpanonhallinnan kustannuksiin. Jos näiden töiden koordinoitua varten joudutaan palkkaamaan täysipäiväinen henkilö esimerkiksi sen vuoksi, että lisenssiehdoissa ei ole pakotettu asiakasta pysyttämään vain yhdessä versiossa tai tekemään versionnostoa joka kerta, kun uusi ohjelma-versio tuodaan markkinoille, ja näiden seikkojen vuoksi yleiskuvan hallinta vaikeutuu, on se yritykselle suuri kulujen tuoja. Toisaalta, jos lisenssiehdot rajaavat ylläpitoa ja tukevat kokoonpanonhallintaa, voidaan saada aikaan suuria säästöjä tuotteen elinkaaren aikana.

## **6.2. Ylläpitoon ja kokoonpanonhallintaan liittyviä kysymyksiä**

Toimittajalla on myös sellaisia ylläpitoon ja kokoonpanonhallintaan liittyviä kysymyksiä lisenssin ominaisuuksista, jotka eivät ole suoranaisesti sidoksissa kustannuksiin. Seuraavaksi tutustumme näihin seikkoihin.

### **6.2.1. Miten käytännössä hoidetaan kokoonpanonhallinta?**

Tuotteen elinkaari on normaalia tietokoneohjelmistoa pidempi, mikä aiheuttaa käytännön ongelmia sekä ylläpidolle että kokoonpanonhallinnalle.

Jos ohjelmistosta tulee uusia versioita vaikkapa vuoden välein, tämä tarkoittaa jo noin pariakymmentä versiota yhden tuotteen elinkaaren aikana. Lisäksi ohjelmistoon liitettäviä lisäkomponentteja kehitetään myös, joten tämäkin lisää ylläpidettävien versioiden lukumäärää. Jos tähän vielä lisätään, että yrityksellä on useita asiakkaita ja että tuotteis-

tettujen ohjelmistojen lukumäärä kasvaa, alkaa ylläpidettävien ohjelmaversioiden määrä olla jo varsin suuri.

Sen lisäksi, että tiedetään kenellä asiakkaalla on mitäkin ohjelman osia käytössään, tarvitaan vielä paljon muunkinlaista kokoonpanonhallintaa. Esimerkiksi versionnostojen väliset datakonversiot täytyy suunnitella tarkoin, ja usein näihinkin vaikuttavat eri asiakkaille räätälöidyt ominaisuudet. Myös eri asiakkaiden ympäristöjen luomat laite- ja integrointiongelmat pitää pystyä hallitsemaan.

Yritys siis joutuu ylläpitämään tietoa niin asiakkaistaan kuin asiakkailla käytössä olevista tuoterakenteista, lisensseistä ja asiakkaan laiteympäristöstäkin. Tämä kaikki vie yrityksen resursseja henkilöiden, ajan ja ohjelmistojen muodossa. Jonkun henkilön on perehdyttävä soveliaisiin kokoonpanonhallintaohjelmistoihin sekä tehtävä käytännön suunnitelmat muuttuvien tilanteiden vaatimusten mukaan.

Lisensoinnin tulee tukea tätä prosessia helpottamalla hallittavaa tietomäärää. Keinoja tähän on esimerkiksi määrätä asiakas hyväksymään versionnosto silloin, kun uusi ohjelmaversio julkaistaan tai pakottaa asiakas pitämään samaa ohjelmaversiota kaikissa asennamissaan solmuissa. Vähintäänkin voidaan sopia, kuinka pitkään taaksepäin ohjelmaversioiden on oltava keskenään yhteensopivia, sillä silloin toimittaja voi jossain vaiheessa lopettaa vanhojen ohjelmaversioiden ylläpidon. Hankalammin määräiltävissä on asiakkaan laiteympäristö, mutta lisenssisopimuksella voidaan sentään rajoittaa ohjelmiston käyttövarmuus ja tuki koskemaan vain tiettyjä laitealustoja.

### **6.2.2. Miten hallitaan eri asiakkaiden ostamat lisenssit?**

Oman haasteensa asiakastietojen hallinnalle muodostaa lisenssien hallinta. Kysymys herää muun muassa siitä, miten ohjelmisto pilkotaan osiin ja mitä osia mikäkin lisenssi kattaa. Myös lisenssiehtojen kausittainen tarkistaminen ja lisenssien uusiminen lisenssikauden loputtua tuottaa omat haasteensa.

Ohjelmisto voidaan lisensoida joko kokonaisuudessaan tai jakaa komponentteihin. Jos ohjelmisto myydään yhtenä isona pakettina, on lisenssien hallinta helpompaa, kuin jos se myydään osissa. Jos ohjelmisto lisensoidaan komponentti kerrallaan, on varmistettava, että jokainen komponentti toimii itsenäisesti ja minkä tahansa muun komponenttiyhdistelmän kanssa. Taloudelliselta näkökannalta katsottuna edullisin ratkaisu olisi myydä tuotteesta perustoiminnallisuus ja sen lisäksi irrallisia komponentteja. Tämä on hyvä ratkaisu myös ylläpidon kannalta, sillä silloin hallittavien lisenssien määrä pienenee. Tällainen ositus takaa myös sen, että asiakkaalle jää joustavat mahdollisuudet täydentää tuotetaan vain tarvitsemillaan lisäkomponenteilla.

Lisenssien ositusten lisäksi pitää hallita myös tuotteen pitkän elinkaaren aikana tapahtuvia muutoksia. Lisenssiehtoihin onkin hyvä kirjoittaa pykälä, jossa sovitaan lisenssiehtojen ja hinnoittelun tarkastaminen tietyin aikavälein tai tiettyjen ehtojen, esimerkiksi markkinatilanteen muutosten, täytyttyä. Toisaalta tämä lisää resurssitarvetta, sillä jonkun

pitää käydä uudet neuvottelut, mutta toisaalta tämä säästää pitkällä aikavälillä paljon rahaa. Säästöt tulevat esimerkiksi hinnan korotuksista, jotka seuraavat inflaatiota, nykyiseen markkinatilanteeseen soveliaammiksi muokatuista lisenssiehdoista sekä asiakastytyväisyyden nousun kiihdyttämästä lisäkaupasta.

### **6.2.3. Miten lisensoitu ohjelmisto poistetaan käytöstä?**

Asiakkaalla voi olla käytössään joitakin voimassa olevia lisenssejä samaan aikaan, kun hän haluaa ostaa lisenssin johonkin uuteen ohjelmistoon tai komponenttiin. Tämä voi johtaa siihen, että asiakkaan eri lisenssien voimassaoloajat loppuvat eri aikoina. Kun lisenssintikauti alkaa olla loppullaan, kohdataan haasteita muun muassa lisenssin uusimisen tai ohjelmiston poiston kanssa.

Lisenssintiehdossa on hyvä olla tarkkaan kerrottuna, mitä tehdään, kun lisenssi päättyy. Suuria ongelmia ei tule, jos päättyvä lisenssi koskee ainoastaan lisäkomponenttia, koska sen uusimispaineet eivät ole kovinkaan suuret. Sen sijaan lisenssiehdossa on sovittava erityisesti, mitä tehdään, kun perustoiminnallisuuden lisenssi päättyy, ja miten tämä mahdollisesti vaikuttaa sellaisten lisäkomponenttien lisensseihin, jotka vielä olisivat jatkumassa. Lisäkomponenttien lisenssien loppumisajankohdat kannattaakin jo alun perin sopia samaksi kuin perustoiminnallisuuden lisenssin loppumisaika.

Lisenssintiehdossa on hyvä myös sopia, miten konkreettisesti hoidetaan ohjelmiston poisto käytöstä sitten, kun lisenssiä ei enää haluta uusia tai ohjelmiston elinkaari on loppussa. Käytännössä on varsin vaikeaa todeta, onko ohjelmiston ajettavat versiot todella poistettu kaikilta asiakkaan tietokoneilta tai onko palautetusta asennuslevykkeestä tehty kopioita. Siksi ehtojen kirjoituksessa kannattaakin mieluummin sopia laitonta käyttöä seuraavasta rangaistuksesta kuin käytännön poistotoimista.

### **6.2.4. Miten hoidetaan versionnostojen aiheuttama työ?**

Lisenssintiehdossa pitäisi sopia käytännön järjestelyistä, jotka seuraavat uusien ohjelmistoversioiden julkaisemista. Jos uusi versio päätetään päivittää asiakkaan ympäristöön, pitää tietää, kuinka ohjelmakoodi viedään asiakkaalle, kuinka se asennetaan käyttöön ja kuinka hoidetaan muuttuvan ohjelman mahdollisesti vaatimat datakonversiot. Lisäksi uusi versio pitää integroida toimimaan yhteen myös asiakkaan ympäristön muun teknologian ja laitteiston kanssa.

Jos toimittaja on vastuussa yllämainituista toimenpiteistä, pitää sopia, kuka voi mennä asiakkaan toimitiloihin asennusta tekemään ja mitä hän siellä saa tehdä ja mitä ei. Koska tämän tutkielman tapaustutkimuksen ohjelmisto on luokiteltava erittäin salaiseksi aineistoksi ja koska sitä todennäköisesti tullaan myymään toisille valtioille, on hyvin epätodennäköistä, että toimittaja saa osallistua ohjelmiston päivitystöihin.

Ohjelmiston luonteen vuoksi lisenssiehtojen tulisikin selvittää muun muassa, miten ohjelmistopäivitykset toimitetaan asiakkaalle ja tekeekö asiakas tarvittavat datakonversiot itsenäisesti toimittajan antamien ohjeiden mukaan vai toimitetaanko asiakkaalle tarvittavat muutosskriptit ohjelmiston mukana.

Toimittajalle tuskin kerrotaan myöskään kovin tarkasti asiakkaan laitealustasta ja muista käytössä olevista ohjelmistoista. Tämä vaikeuttaa integrointisuunnitelman tekemistä samoin kuin varsinaisen integroinnin toteuttamista. Lisenssiehdoissa on siksi myös sovittava, kuuluvatko integrointityöt toimittajalle, ja jos kuuluvat, niin millaista informaatiota asiakas silloin antaa toimittajalle omista ympäristöistään. Toisaalta, jos toimittaja pääsee käsiksi asiakkaan laitetietoihin, hän voi lisätä omaa liiketoimintaansa tarjoamalla integrointipalveluja lisätyönä ja ehkä myös kehittää uusia, asiakasta hyödyttäviä ohjelmistoja tai komponentteja. Tätä kautta laitetietojen luovuttaminen toimittajalle olisi myös asiakkaan kannalta hyödyllistä.

#### **6.2.5. Miten henkilöresurssit saadaan vastamaan tarvetta?**

Sekä lisenssien hallinta että niitä koskettavien ohjelmistojen kehittäminen vaativat henkilöresursseja. Kaikkia edellä olevissa luvuissa käsiteltyjä toimenpiteitä tekemään tarvitaan osaavaa henkilöstöä.

Yrityksellä on siis oltava tarpeeksi paljon henkilökuntaa, mutta pelkästään se ei riitä. Henkilöstöllä on myös oltava tarpeellinen tietotaito tehtäviensä suorittamiseen. Tämä tarkoittaa henkilöstön kouluttamista muun muassa kokoonpanonhallintatyökalujen käyttöön sekä asiakkaan ympäristön laitteiden ja muiden ohjelmistojen käyttöön ainakin pintapuolisesti. Jos lisäksi on sovittu koulutuspalvelusta asiakkaan kanssa, on yrityksen varmistettava, että siltä löytyy sopiva ja osaava henkilö, esimerkiksi kielitaidon puolesta, toimimaan tässä tehtävässä. On myös huomioitava, että lisensoituja ohjelmistoja on ylläpidettävä ja edelleen kehitettävä, ja myös tähän työhön on löydettävä asiansa osaavaa henkilöstöä.

Kaikki tämä maksaa yritykselle rahaa niin henkilökunnan palkkojen, toimitilakustannusten kuin koulutuksenkin vuoksi. Yrityksen on siis hyvä miettiä jo lisenssiehdoista sopiessaan, minkälaisia palveluja se pystyy asiakkaalle lupaamaan. Pitää miettiä, ovatko lisensseistä saatavat tulot tarpeeksi suuret kattamaan henkilöresurssien varmistamisen aiheuttamat kulut.

### **6.3. Lisensointikysymyksiä asiakkaan näkökulmasta**

Koska lisenssisopimuksen allekirjoittamiseen tarvitaan aina kaksi osapuolta, on lisenssiehtoja sovittaessa tarkasteltava asiaa myös asiakkaan näkökulmasta. Sopimusehtojen on palveltava myös asiakkaan etuja, sillä muuten hän ei allekirjoita sopimusta. Jos sopimusehdot myöhemmin paljastuvat asiakkaalle epäoikeudenmukaisiksi, se estää asiakas-

suhdetta muodostumasta pitkäaikaiseksi kumppanuudeksi. Seuraavaksi käsitellään kysymyksiä, joihin pohditaan vastausta asiakkaan näkökulmasta.

### **6.3.1. Miten hinta jakaantuu ohjelmiston elinkaaren ajalle?**

Myös asiakkaalla on huoli omasta taloudestaan ja toimintansa kannattavuudesta. Asiakkaat ovat isoja organisaatioita tai vastaavat toiminnastaan valtionhallinnolle, joten niiden byrokratian määrä on suurta ja päätöksenteko hidasta. Asiakkaan talous ja lisenssien ostamiseen käytettävät varat määräytyvät pitkälti niille myönnetystä määrärahoista. Byrokratia ja vuosittaiset määrärahat vaikuttavat siihen, että myös budjetointi on hidasta ja syklimäistä.

Asiakkaan kannalta siis lisenssin hinnoittelun tulisi olla mahdollisimman helposti ennustettavaa, eikä se myöskään saisi sisältää suuria kertasummia. Mitä kohtuullisempi on lisenssin alkuhinta, sitä helpommin sen hankintapäätös menee läpi ja ohjelmisto saa ostoluvan. Toisaalta vuosittaiset lisenssimaksut tai muu kausittainen laskutus helpottaa tulossa olevien kustannusten ennakointia ja budjetointia.

Lisenssiehtojen kannattaa siis tukea mallia, jossa tuotteen ostohinta on kohtuullinen ja jossa vuotuiset maksut pysyvät suurin piirtein samanlaisina vuodesta toiseen.

Toinen vaihtoehto on sopia laskutettavista summista, esimerkiksi versionnostopalkkioista tai asennus- ja koulutuspalkkioista, tarpeeksi paljon etukäteen eli käytännössä jo ennen sen vuoden budjetin tekemistä, jolle laskutus lankeaa. Käytännössä tällainen on kuitenkin hankalaa, sillä voi olla vaikeaa ennakoita etukäteen, koska versionnosto on ajankohtaista tai koska jotakin tiettyä palvelua toimittajalta kaivataan. Joskus tällaiset tarpeet voivat ilmaantua hyvinkin nopeasti, ja silloin niille olisi jo oltava rahoitus valmiina asiakkaan budjetissa.

Asiakkaan ja myös toimittajan kannalta suotuisin ratkaisu olisikin vuosittainen korvaus, joka kattaisi myös osan mahdollisista äkillisistä kuluista. Lisenssimaksun hinta tarkistetaan esimerkiksi vuoden tai kahden vuoden välein samalla, kun lisenssin muitakin ehtoja tarkennetaan. Tällöin asiakas voisi budjettia tehdessään varautua täsmälliseen, helposti ennakoitavaan summaan.

Myös toimittajalle tällainen malli on kannattava siinä mielessä, että vaikka tällöin joudutaankin ottamaan se riski, että joinakin vuosina työtä tehdään enemmän kuin laskutettavan summan kattamat kulut, niin joinakin vuosina työtä on vastaavasti vähemmän. Silti toimittajan saamat tulovirrat pysyvät tasaisina ja hyvä, joustava kumppanuus asiakkaan kanssa säilyy.

### **6.3.2. Miten ennakoitavissa henkilöresurssien tarve on?**

Asiakkaan tietoturvaso on hyvin korkea, ja usein suojeltavana on niin valtiosalaisuuksia kuin myös valtion maanpuolustukseen liittyviä asioita. Asiakas siis ei voi päästää toi-

mittajaa näkemään tietokantojensa sisältämää dataa eikä ole myöskään aivan itsestään selvää, että toimittajaa päästetään edes asiakkaan toimitiloihin. On myös toimittajan edun mukaista valvoa omia liikesalaisuuksiaan ja varmistaa, että esimerkiksi eri asiakkaat eivät saa vahingollisia tietoja toisistaan.

Näiden syiden vuoksi kaikki tarvittavat toimenpiteet tulevat mitä todennäköisimmin vaatimaan henkilöresursseja myös asiakkaan puolelta. Asiakkaan on siis voitava valmistautua ennakolta muun muassa versionnostojen aiheuttamiin toimenpiteisiin kuten datakonversioihin ja koulutuksen tarjoamiseen. Aina asiakkaalla ei välttämättä ole sopivaa, vapaata ja osaavaa henkilökuntaa saatavilla näihin tehtäviin, joten osa vastuusta siirtyy välttämättä takaisin toimittajalle.

Lisenssiehdoissa on siis hyvä olla selkeästi kuvattuna, mitä tehdään muutostilanteissa ja kuka vastaa mistäkin tilanteen vaatimasta työstä. Sopimuksessa voidaan esimerkiksi sopia toimittajan puolelta joku tietty luottohenkilö, jolla on lupa toimia asiakkaan ympäristössä tai voidaan sopia, että toimittaja kouluttaa asiakkaan henkilökunnasta joitakin henkilöitä toimimaan välimiehinä toimittajan ja asiakkaan välillä.

Riippumatta siitä, miten vuorovaikutus toimittajan ja asiakkaan välillä hoidetaan, on henkilöresurssien tarpeen ennakoitavuus tärkeää toimittajan lisäksi myös asiakkaalle ja tästä johtuen molemmilta osapuolista vaadittavien asioiden listaus lisenssisopimuksessa on erityisen tärkeää.

### **6.3.3. Miten hoidetaan versionnostojen aiheuttama työ?**

Myös asiakkaan, kuten toimittajankin, tärkein varjelu kohde on liiketoiminnan salaisuuksien säilyttäminen. Tämä tuo käytännön ongelmia niin versiopäivitysten kuin datakonversioidenkin asennukseen ja käyttöönottoon.

Ensin on mietittävä, miten ohjelmisto toimitetaan asiakkaalle eli lähetetäänkö se sähköisessä muodossa esimerkiksi internetin välityksellä, lähetetäänkö ohjelmat levykkeelle poltettuna postitse tai muun kuriirin välityksellä vai viekö joku asennuslevykkeet asiakkaalle henkilökohtaisesti.

Välityspalvelujen käytössä on aina omat riskinsä, että tieto joutuu väärin käsiin. Jos ohjelmisto esimerkiksi lähetetään postin kautta, on aina se vaara, että kirjekuori hukkuu tai päätyy sellaisille tahoille, jotka käyttävät sitä joko toimittajaa tai asiakasta vastaan.

Henkilökohtainen toimitus on ehkä turvallisinta tapa toimittaa levykke perille, vaikka toki sekään ei ole täysin varma. Henkilön käyttö välittäjänä on myös kallista, sillä esimerkiksi lentoliput luovutuspaikoihin ovat kalliit.

Sen sijaan tietojen välitys internetin kautta on nykyaikana varsin turvallista erilaisten salausmenetelmien ansiosta. Lisäksi voidaan käyttää asiakkaan ja toimittajan välille tehtyjä suljettuja verkkoja, joista ei ole pääsyä yleiseen verkkoliikenteeseen.

Eri lähetysvaihtoehtojen pohdinta on siis tärkeä vaihe lisenssineuvotteluja ja erilaisten vaihtoehtojen hyvin, ja huonojen puolien pohdinta johtaa molempia osapuolia tyydyttävään ratkaisuun.

#### **6.3.4. Miten toimivat ylläpito ja tukipalvelut käytännössä?**

Yksi eniten asiakastyytyväisyyteen vaikuttavista osatekijöistä on toimivat ylläpito- ja tukipalvelut. Jos asiakas huomaa ongelmia ohjelmistossaan, hän haluaa saada toimittajalta pikaisen vastauksen ja mahdollisesti korjauksen havaitsemaansa virheeseen. Myös myönteinen ilmapiiri kehitysehdotusten jakamiseen puolin ja toisin lisää tyytyväisyyttä.

Lisenssiehdoissa kannattaakin sopia myös, miten mahdollisissa virhetilanteissa toimitaan ja millä keinoilla asiakas esimerkiksi ilmoittaa löytämänsä virheen ohjelmistosta. Samoin voidaan sopia, miten asiakas ilmoittaa keksimänsä kehitysehdotukset ohjelmistoon. Asiakastyytyväisyyden nimissä toimittajan kannattaisi myös asettaa takaraja, jolloin asiakkaan viestiin viimeistään vastataan.

## 7. The Composite Features Diagram -arviointimenetelmä

The Composite Features Diagram (CFD) on arviointimenetelmä, joka tuo käytettävät arviointikriteerit, raja-arvot sekä tulokset näkyvään, helposti luettavaan ja ymmärrettävään muotoon. Sitä on helppo käyttää, ja siksi se on myös erittäin tehokas väline arviota tehtäessä.

Uskon, että monipuolisten ominaisuuksiensa ja tulosten helpon luettavuutensa vuoksi CFD sopii erittäin hyvin tapaustutkimukseni arviointimenetelmäksi. Valitsin tämän menetelmän myös siksi, että en ole ennen nähnyt sitä käytettävän tämänkaltaisessa työssä yrityksessäni ja haluan osoittaa, että se todella toimii.

CFD on varsin uusi arviointimenetelmä, eikä sen käytöstä vastaavankaltaisissa tutkimuksissa löytynyt viitteitä. Uskon kuitenkin, että menetelmä sopii rakenteensa puolesta hyvin tähän tutkielmaan. Haluan myös osoittaa, että tämä menetelmä todellakin toimii halutulla tavalla ja että koko tämä tutkielma antaa esimerkin myös muille vastaavankaltaisten vertailujen tekijöille.

CFD-menetelmä yhdistää sekä kvalitatiivisen eli laatuun painottuvan, että kvantitatiivisen eli määrään perustuvan tutkimuksen tekotavan, jonka vuoksi se on kattavampi kuin monet muut yleisesti käytetyt menetelmät. [Georgiadou, 2003]

Laadullinen lähestymistapa tulee ilmi siitä, että jo arvioinnin alkuvaiheessa todetaan kunkin arvioitavan kohteen kohdalta, onko sillä halutut ominaisuudet ja täyttävätkö ne annetut kriteerit. Tällä perusteella tehdään karsintaa läpi koko arviointiprosessin [Georgiadou, 2003].

Määrällinen näkökulma tulee määrällisesti mitattavien ominaisuuksien testaamisesta. Testit toistetaan jokaiselle arvioinnin kohteelle samanlaisina ja saman laajuisina [Georgiadou, 2003]. Testien perusteella arvotetaan vertailtavien vaihtoehtojen paremmuus.

### 7.1. Menetelmän kuvaus

Arviointiprosessi käyttäen CFD:a aloitetaan etsimällä kaikkein tärkeimmät ja mielenkiintoisimmat ominaisuudet arvioitavasta kohteesta. Joskus tällaiset ominaisuudet ovat hyvin yleisellä tasolla määriteltyjä, kuten ”helppokäyttöisyys”. Tällaisia yleisen tason ominaisuuksia on kuitenkin hankala mitata numeerisesti. [Georgiadou, 2003]

Usein on myös vaikeaa tehdä suoraa vertailua kahden kohteen välillä esimerkiksi siksi, että niiden myyntiselosteessa asiat on kuvattu eri tavoin, tai siksi, että sellaisia lauseita kuin ”meidän tuotteemme parantaa tuottavuuttanne suuresti” on vaikea tulkita ja verrata käytännössä [Georgiadou, 2003].



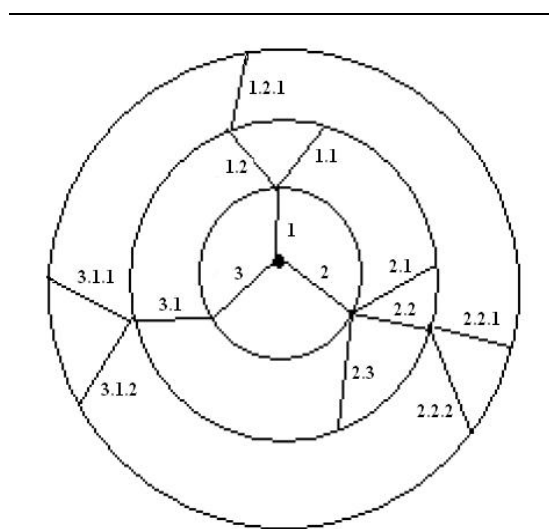
CFD-menetelmässä arviointiprosessi alkaa siitä, että valitaan ne ominaisuudet arvioitavasta kohteesta, jotka siinä on välttämättä oltava ja joiden on laadultaan oltava tietyllä tasolla, jotta kohde voitaisiin hyväksyä käytettäväksi suunniteltuun tarkoitukseen.

Ylemmän tason tarkasteltavia ominaisuuksia kannattaa valita 3 - 5 kappaletta. Tällaisen määrän ominaisuuksia pystyy piirtämään kaavioon ilman ongelmia ja tulosten lukeminen on vielä helppoa. Jos ominaisuuksia on enemmän, niin kaaviosta voi tulla sekava ja tällöin myös tulosten lukeminen vaikeutuu.

Kun ylemmän tason ominaisuudet on valittu, niitä aletaan jakaa pienempiin osiin. Näin saadut ominaisuudet pilkotaan jälleen pienempiin osiin ja näin jatketaan, kunnes löydyt sellainen ominaisuus, jonka arvo pystytään mittaamaan numeerisesti. [Georgiadou, 2003]

Esimerkkinä jakamisesta voidaan ajatella ylemmän tason ominaisuutta ”helppokäyttöisyys”, joka itsessään on liian yleinen käsite mitattavaksi. Tämä ominaisuus voidaan jakaa pienempiin osiin, joista yksi voisi olla ”nopea oppia”. Tämä taas puolestaan voidaan jakaa osiin, joista yksi voisi olla ”uusi toiminto x opitaan alle 5 minuutissa”. Tämä alimman tason ominaisuus voidaan nyt mitata minuuteissa eli jakaminen voidaan osin lopettaa. Suurin hyöty tämänkaltaisessa lähestymistavassa on, että jos arvioitava kohde ei omaa arvioitavaa ominaisuutta, niin se voidaan hylätä saman tien.

CFD-kaavio on piirros, joka koostuu sisäkkäisistä renkaista, kuten näkyy kuvasta 3. Kuvassa korkeimman tason ominaisuudet on numeroitu välillä 1 – 3. Alemman tason ominaisuudet piirretään kuvaajan seuraaville tasoille ikään kuin haarautuviksi puunoksiksi.

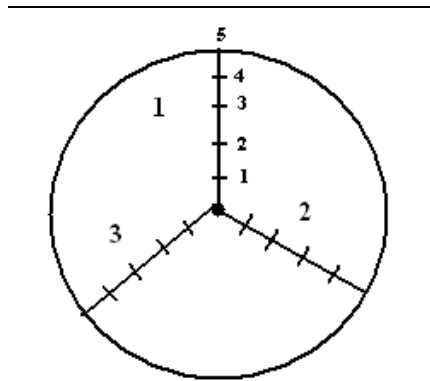


Kuva 3. Esimerkki The Composite Features Diagram -kaaviosta.

Kaaviota selkeyttää se, että ominaisuudet on numeroitu ja vain numerot on piirretty diagrammiin näkyviin. Jos kuvassa olisi auki kirjoitettuna jokaisen ominaisuuden selite, niin se muuttuisi epäselväksi ja olisi vaikeampi tulkita.

Seuraavaksi voidaan päättää kunkin ominaisuuden arvosteluasteikosta. Tämä vaihe vaatii kaikkien asianosaisten välistä keskustelua, jotta saavutettaisiin yksimielisyys siitä, kuinka testitulokset merkitään kaavioon ja kuinka kaavion tuloksia luetaan.

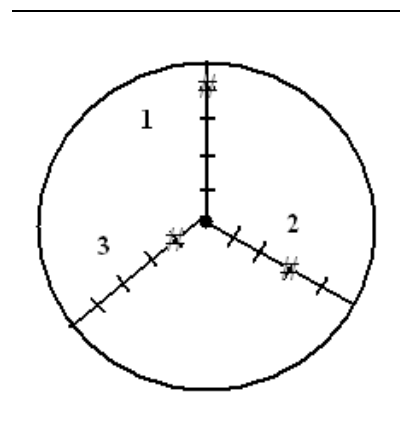
Yleensä arvosteluasteikkona käytetään numeroväliä 1 – 5 siten, että huonoin arvona on 1 ja paras on 5. Jos mitattava ominaisuus puuttuu tai se saa testissä niin huonon tuloksen, ettei sitä voida pitää hyväksyttävänä, arvosanaksi tulee nolla, joka automaattisesti tarkoittaa, että kohde hylätään [Georgiadou, 2003]. Asteikon luvut 2 – 4 sijoitetaan tasavälein ylimmän ja alimman arvon väliin. Arvosteluasteikko näkyy kuvasta 4.



Kuva 4. Esimerkki asteikkojen merkitsemisestä CFD-kaavioon.

Kun arvosteluasteikko on hyväksytty, on aika päättää, mikä on kunkin ylimmän tason ominaisuuden kohdalta se raja-arvo, jonka tulee täyttyä, että kohde voidaan hyväksyä. Raja-arvojen päättämiseen tarvitaan kaikkien asianosaisten yhteinen päätös.

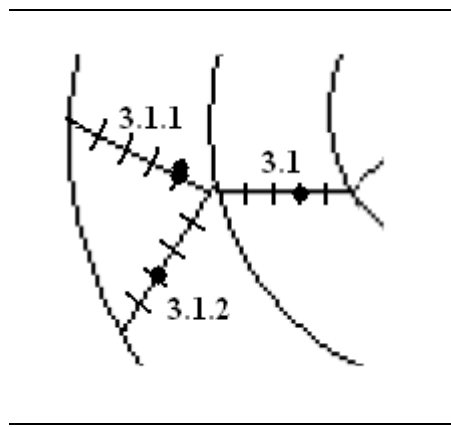
On erittäin tärkeää, että raja-arvoja ja niiden välisiä suhteita mietitään huolellisesti, sillä jos kaikille ominaisuuksille laitetaan raja-arvoksi 5, olisi arvioitavan kohteen oltava joka suhteessa täydellinen. On kuitenkin erittäin harvinaista, että mikään arvioitava kohde saa täydet pisteet jokaisesta arvioitavasta ominaisuudesta, joten tällaiset raja-arvot aiheuttavat sen, että mahdollisesti yksikään arvioitava kohde ei täyttäisi kriteereitä. Raja-arvot on piirretty kuvaan 5 asteikkoviivojen päälle käyttäen #-merkkejä.



Kuva 5. Esimerkki raja-arvojen merkitsemisestä CFD-kaavioon.

Seuraavaksi suoritetaan varsinainen testaaminen. Testitapahtumassa voidaan esimerkiksi pyytää valittuja henkilöitä suorittamaan testattavia tehtäviä ja havainnoida esimerkiksi käytettyä aikaa. Kaikkien testattavien asioiden ei tarvitse olla näin konkreettisia, vaan jotkin arvot voidaan saada esimerkiksi matemaattisesti laskemalla tai tutkimalla tilastoja.

Kun kaikki tarvittavan testit on tehty ja testitulokset muutettu arvosteluasteikkoja vastaaviksi arvosanoiksi, on aika siirtää tulokset kaavioon. Arvosanojen vieminen kuvaajaan aloitetaan uloimmalta kehältä, jossa alimman tason mitattavat ominaisuudet ovat. Kuten kuvassa 6 näkyy, kunkin ominaisuuden kohdalle tehdään merkintä saadun arvosanan mukaisesti.

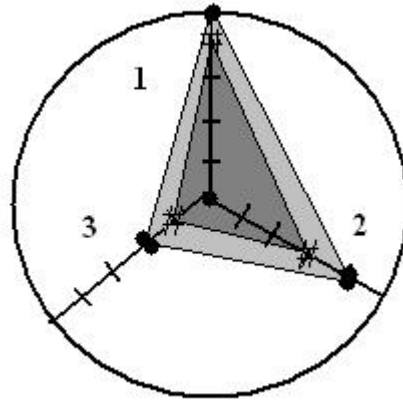


Kuva 6. Esimerkki tulosten merkitsemisestä CFD -kaavioon.

Esimerkin ominaisuuden numero 3 raja-arvo on 1, kuten näkyy kuvasta 5. Koska sekä 3.1.1 että 3.1.2 saavuttivat tämän raja-arvon, lasketaan näiden kahden keskiarvo, josta tulee ominaisuuden 3.1 arvosana. [Georgiadou, 2003]. Samalla periaatteella lasketaan kaikkien ominaisuuksien arvosanat, kunnes lopulta saadaan arvosanat myös ylimmän tason pääominaisuuksille.

Tässä vaiheessa, kun diagrammin sisintä rinkiä täytetään, voi kaaviosta käyttää myös nimitystä Kiviati-diagrammi. Se on ulkonäöltään täysin vastaava kuin CFD:n sisimmäinen rengas. Kiviati-diagrammin suurin etu on sen mahdollistamat kolmion pinta-alaan perustuvat laskentakaavat, joita voidaan käyttää apuna arvioitaessa eri vaihtoehtojen keskinäistä paremmuutta. Laskentakaavoja voi käyttää myös jo CFD:n uloimmilla kehillä näin halutessaan [Georgiadou, 2003]. Tässä tutkielmassa näitä kaavoja ei kuitenkaan ole tarkoitus käyttää.

Kiviati-diagrammissa tulosten arviointia varten piirretään viivat sekä raja-arvojen että arvosanojen välille siten, että niistä muodostuu kaksi umpinaista kuviota. Kaavion lukemista helpottaa, jos molempien kuvioiden pohjavärit vaihdetaan. Tämä näkyy selkeästi kuvassa 7.

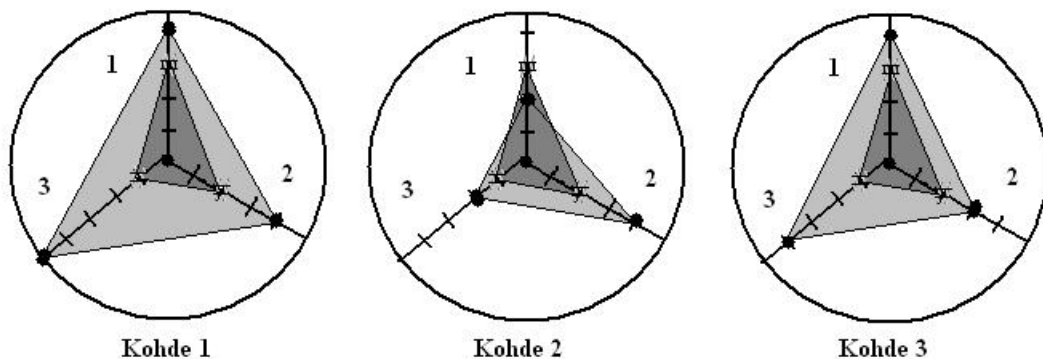


Kuva 7. Esimerkki raja-arvojen ja arvosanojen peittämien alueiden merkitsemisestä Kiviat-diagrammiin.

Arviointi tapahtuu vertailemalla piirrettyjen kuvioiden peittämää aluetta. Jotta tulos olisi hyväksyttävä, jokaisen ominaisuuden saama arvosana pitää olla yhtä suuri tai suurempi kuin ominaisuudelle ennen testausta määritelty raja-arvo.

Jos arvioitavia kohteita on useampia, voidaan niistä kaikista tehdä samanlaiset CFD-kaaviot. Eri kohteiden keskinäisen paremmuuden toteaminen on luotettavaa, koska jokaisen kohteen vertailu perustuu samoihin testeihin, joiden tulokset on myös arvioitu samalla asteikolla ja samoilla raja-arvoilla. Vertailussa suurimman alueen omaava kohde on vertailun paras ja sen pitäisi tulla valituksi.

Kuvassa 8 on esimerkki tilanteesta, jossa vertaillaan kolmea kohdetta, joista kustakin on testattu kolme eri ominaisuutta.



Kuva 8. Kolme esimerkkikaaviota.

Aivan ensimmäiseksi nähdään, että kohde numero 2 ei täytä vaadittuja ehtoja, sillä se ei yllä yhteen vaadituista raja-arvoista. Tällä perusteella kohde numero 2 voidaan hylätä.

Kohteista 1 ja 3 on helppo katsoa, kuinka suuret ovat niiden arvosanojen peittämät alueet, ja verrata näitä kahta alaa toisiinsa. Alue on suurempi kohteella numero 1, joten se voittaa vertailun ja tulee valituksi.

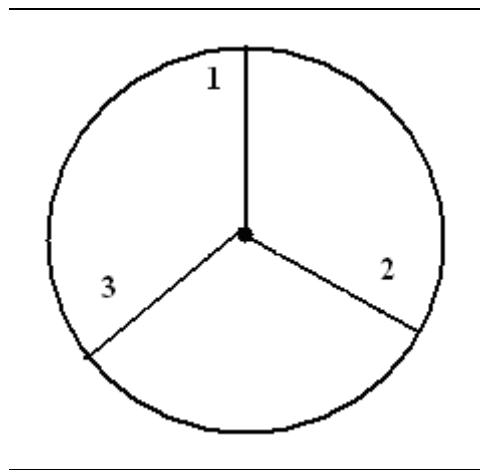
Tilanteessa, jossa vertailtavilla kohteilla on täysin yhteneväiset tulokset, voidaan periaatteessa valita mikä vain, sillä silloin kaikki kohteet ovat laadultaan tasavertaisia. Parhaimman vaihtoehdon valintaa voi kuitenkin helpottaa noudattamalla seuraavia valintakriteereitä:

1. Katsotaan, minkä ominaisuuden raja-arvo on asetettu korkeimmaksi eli mikä ominaisuus on arvioitu tärkeimmäksi ja valitaan se kohde, joka on saanut parhaimmat arvostamat tästä ominaisuudesta.
2. Katsotaan mitattuja testituloksia ja valitaan se kohde, jolla on tasaisimmat testiarvostamat. Näin siksi, että tällainen kohde on luultavimmin tasaisen laadukas jokin mitattavan ominaisuuden osa-alueella sen sijaan, että olisi todella hyvä jossakin ja vastaavasti todella huono jossakin.
3. Testejä tehtäessä on saatu kokemusta eri kohteista. Tällainen kokemus voi johtaa siihen, että jokin arvioitava kohde tuntuu luontevimmalta ja miellyttävimmältä käyttää ja tulisi tästä syystä tulla valituksi.
4. Voidaan myös valita lisää ominaisuuksia ja tehdä näitä käyttäen uusi arviointi, jonka perusteella valinta tehdään.
5. Jos mikään yllä olevista ei tuo ratkaisua, valitaan se, joka on visuaalisesti miellyttävimmän näköinen. Yleensä kaunis ulkoasu tuo positiivisen vaikutelman tuotteesta ja lieventää muutosvastarintaa.

## 8. Vertailuun valittavat ominaisuudet ja niiden ositus

Vertailuun valitaan kolme lisenssiltä vaadittavaa ominaisuutta. Ominaisuuksien painopiste on toimittajalle tärkeissä asioissa siksi, että tätä tutkielmaa tehdään ohjelmiston omistajan näkökulmasta. Toimittajan ominaisuudet jakaantuvat kahteen osaan, kustannusten hallintaan ja ylläpidon hallintaan. Kolmanneksi pääominaisuudeksi tulee mukaan asiakkaan näkökulmasta tärkeitä asioita. Kaikista vertailtavista ominaisuuksista on sovittu yhdessä tapaustutkimuksen myyjäryityksen kanssa neuvottelemalla.

Vertailuun valitut pääominaisuudet näkyvät kuvassa 9. Kuvan numero 1 tarkoittaa toimittajan kustannusten hallintaan liittyviä ominaisuuksia, numero 2 toimittajan ylläpidon hallintaan kuuluvia ominaisuuksia ja numero 3 asiakkaan näkökulmasta kiinnostavia kysymyksiä.



Kuva 9. Vertailuun valitut pääominaisuudet CFD-kaaviossa.

### 8.1. Kustannuksiin liittyvät pääominaisuudet ja niiden ositus

Kustannuskysymykset on jaettu suoraan viiteen eri mitattavaan ominaisuuteen. Nämä ominaisuudet kuvaavat tuotteen hintaan ja sen jakautumiseen liittyviä vaihtoehtoja, versionnostojen pakollisuutta ja koulutuksen hinnoittelua, myyntikoneiston valinnan vaikutuksia, ohjelmaversioiden asiakaskohtaisen lukumäärän vaikutusta sekä markkinatilanteen muutoksiin joustavasti suhtautumisen vaikutusta tuotteen lopulliseen hintaan sen elinkaarren loputtua.

Seuraavaksi kuvataan kutakin mitattavaa kustannuksiin vaikuttavaa ominaisuutta.

#### 1.1 Lisenssin alkuhinta ja laskutus lisäpalveluista

Lisenssin kokonaishinta koostuu sen alkuhinnasta ja lisäpalveluiden, kuten versionnostojen, asennuspalveluiden, koulutuksen ynnä muiden vastaavien hinnoista. Toimittaja haluaa saada maksimaalisen tuoton ohjelmistostaan pitkällä

aikavälillä. Ohjelmiston korkea alkuhinta ei välttämättä takaa kaikkein suurimpia tuottoja juuri pitkällä aikavälillä.

Vertailussa tutkitaan tuotteen lopullista tuottoa sen 20 vuoden elinkaaren aikana erilaisilla hinnoittelun jaoilla.

### **1.2 Versionnoston pakollisuus ja koulutuspalvelun hinnoittelu**

Lisenssisopimuksesta kirjoitettaessa pohditaan usein sitä, velvoitetaanko asiakas tekemään versionnosto aina ja jos näin tapahtuu, niin liittykö siihen toimittajan antamaa koulutusta vai pitääkö koulutuspalvelu ostaa.

Vertailussa tutkitaan, miten lisenssistä saataviin lopullisiin tuloihin vaikuttaa se, onko versionnosto pakko tehdä ja kuuluuko siihen mukaan koulutusta vai ei.

### **1.3 Myyntikoneiston valinnan vaikutus lopulliseen tuottoon**

Yrityksen valitsema myyntitapa vaikuttaa myös siihen, kuinka paljon lisenssi tuottaa yritykselle. Jos omalla yrityksellä on resursseja myydä tuotettaan, kaikki lisenssitulot tulevat myös sille itselleen.

Jos tuotteen sen sijaan joutuu antamaan kolmannelle osapuolelle jaettavaksi, vähentää se oman osuutensa saaduista tuotoista. Kolmannen osapuolen osuus lisenssin myyntituloista riippuu siitä, saako se tehdä omia muutoksiaan ohjelmistoon vai ei.

Vertailussa tutkitaan, miten myyntikoneiston valinta vaikuttaa hinnoittelun lopputulokseen.

### **1.4 Asiakaskohtaisten versioiden lukumäärän vaikutus tuottoon**

Asiakkaalla käytössä olevien ohjelmaversioiden lukumäärä vaikuttaa negatiivisesti yrityksen tuottoon. Jos yhdelle asiakkaalle sallitaan useita ohjelmaversioita, se lisää ylläpitokustannuksia. Lisäksi versionnostot aiheuttavat tällöin suuremman työmäärän.

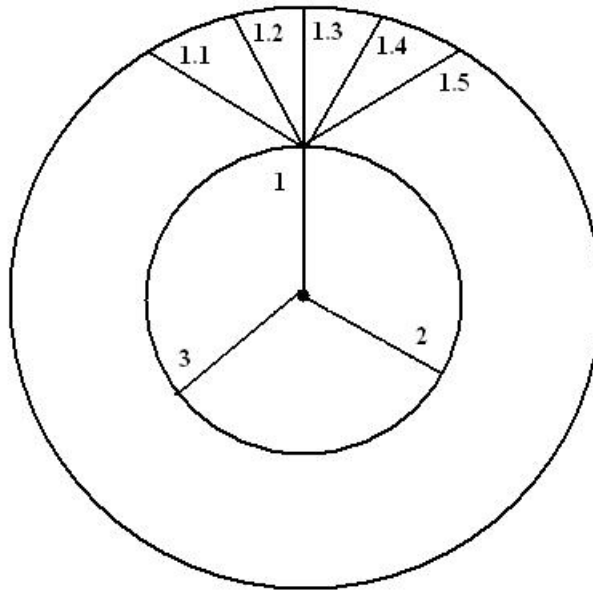
Vertailussa tutkitaan ohjelmaversioiden lukumäärän vaikutusta tuotteen lopulliseen hintaan.

### **1.5 Joustava reagointi markkinatilanteen muutoksiin**

Markkinat muuttuvat tuotteen pitkän elinkaaren aikana varmasti moneen kertaan. Tuotteen loppuhintaan vaikuttaa siis myös, kuinka joustavasti lisenssiehdot näihin muutoksiin reagoivat. Jos lisenssiehtoja ei voida muuttaa markkinatilanteen muuttuessa, voidaan menettää paljonkin tuloja. Esimerkiksi voimakas inflaatio voi syödä rahan arvoa niin paljon, että jos tuotteen hintaa ei korjata, niin toimittaja voi jäädä tappiolle. Siksi joustava lisenssiehtojen uudistaminen myös kesken sopimuskauden on suotavaa.

Vertailussa tutkitaan joustava ja joustamaton sopimusmalli vaikuttaa lisenssituottoihin.

Kuvassa 10 on CFD-kaavioon sijoitettuna kustannustehokkuuteen ja tuotteen loppuhintaan vaikuttavat mitattavat ominaisuudet.



Kuva 10. Kustannustehokkuuteen vaikuttavat vertailtavat ominaisuudet CFD-kaaviossa.

## 8.2. Ylläpitoon liittyvät pääominaisuudet ja niiden ositus

Ylläpitoon ja kokoonpanonhallintaan liittyvät kysymykset on jaettu kolmeen eri alaryhmään. Ensimmäinen ryhmä kokoaa yhteen henkilöresursseihin liittyviä seikkoja, toinen ryhmä tekniset haasteet ja kolmanneksi ryhmäksi on valittu lisenssiin liittyviä kysymyksiä.

Henkilöresurssiominaisuudet kattavat kysymykset toimintatavan vaikutuksesta datakonversion virheellisyyteen sekä eri toimintojen vaatimista henkilötyömääristä.

Tekniset kysymykset käsittelevät versionnostojen pakollisuuden ja laiteympäristöjen moninaisuuden vaikutusta ylläpitoon.

Lisenssiehdoista riippuvista asioista tutkitaan lisenssin jaottelun vaikutusta ylläpidettävien lisenssien määrään, lisenssiehtojen tarkistuksen vaikutusta henkilöresursseihin sekä lisenssiajan loppumisen eri toimenpiteiden vaikutuksia.

Seuraavassa on kuvattuna kaavion osion 2.1 mitattavat, henkilöresursseihin vaikuttavat ominaisuudet:

### 2.1.1 Virhemahdollisuus datakonversioiden tekemisessä

Datakonversioiden tekeminen eri ohjelmistoversioiden välillä on aina riskialtista työtä. Toimittaja on parhaiten selvillä ohjelmistoon tehdyistä muutoksista, joten osaltaan hän myös tietää eniten muutostarpeista. Toisaalta asiakas on omien tietojensa paras asiantuntija, joten on helppo kuvitella, että myös hänellä olisi datakonversioiden tekemisessä tarvittavaa tietoa. Käytännössä siis konversion voi siis tehdä joko asiakas tai toimittaja, mutta yhteistyössä keskenään.



Vertailussa tutkitaan virheprosentin suuruutta riippuen siitä, kumpi data-konversion tekee, asiakas vai toimittaja, ja kuinka paljon yhteistyötä prosessi sisältää.

### **2.1.2 Henkilöressurssien tarve**

Henkilöressurssien ylläpito on kallista. Kaikkia toimittajan lupaamia toimintoja ja palveluita varten on löydettävä henkilökuntaa tehtäviä suorittamaan. Usein ohjelmiston ylläpitoon liittyvät tehtävät ovat luonteeltaan sellaisia, että yhden ihmisen ei ole mahdollista tehdä niistä useaa. Tämä johtuu esimerkiksi siitä, että jotkin tehtävät vievät niin paljon aikaa, ettei henkilö ehdi tekemään muuta tai tehtävä vaatii sellaista erikoisosaamista, jota kaikilla ei ole. Siksi toimittajan kannattaa miettiä tarkoin, millaisia palveluita se on valmis tarjoamaan.

Vertailussa pohditaan toimittajan tarjoamien toimintojen ja palveluiden vaikutusta henkilöressurssien tarpeeseen.

Seuraavaksi kuvataan kaavion osion 2.2 teknisiin kysymyksiin liittyvät mitattavat tutkimusongelmat.

### **2.2.1 Versionnostojen pakollisuus**

Versioiden ylläpito vaatii paljon resursseja. Mitä pidempi on ohjelmiston elinkaari, sitä useampia versioita siitä ehditään tekemään. Myös asiakkaiden lukumäärä lisää versioiden määrää, sillä hyvin todennäköisesti jokaisesta komponentista joudutaan muokkaamaan kullekin asiakkaalle oma versio. Ylläpityötä lisää, että eri ohjelmaversioiden pitää olla taaksepäin yhteensopivia edellisten versioiden kanssa.

Ylläpityö vähenee merkittävästi, jos lisenssiehdoissa saadaan sovittua asiakkaan kanssa versioiden rajoittamisesta. Tämä voidaan tehdä esimerkiksi siten, että asiakas pakotetaan huolimaan versionnosto aina kun sellaista tarjotaan tai sovitaan, että yhteensopivuus edellisiin versioihin rajoitetaan esimerkiksi kahteen edelliseen versioon.

Vertailussa tutkitaan, miten erilaiset versioiden määrien rajoittamiskeinot vaikuttavat ylläpidettävien ohjelmistoversioiden määriin.

### **2.2.2 Laiteympäristöjen lukumäärä per asiakas**

Ohjelmiston pitää myös toimia useilla eri laitealustoilla, sillä etukäteen ei voida olettaa, että kaikilla tulevilla asiakkailla on samanlaiset laitteistot käytössään. Tämä lisää myös osaltaan ohjelmaversioiden lukumäärää sekä versionhallintaan liittyvää ylläpitoa.

Ylläpidettävän versiomäärän vähentämiseksi voidaan myös lisenssiehdoissa sopia rajoituksia asiakkaan laitteistoihin. Rajoitukset voisivat esimerkiksi määrätä, että yhdellä asiakkaalla saa olla vain tietty määrä erilaisia laitteita käytössään.

Toinen keino tehdä rajoitusta tuettavien laitteistojen määrään on ilmoittaa jo etukäteen, että ohjelmisto on testattu toimivaksi vain tietyillä, erikseen luetelluilla laitealustoilla. Asiakas voi toki käyttää ohjelmistoa muissakin laitteissa, mutta toimittaja ei ole tällöin vastuussa ohjelmiston oikeasta toiminnasta.

Vertailussa tutkitaan muutamien eri laitteistorajoitusten vaikutusta ylläpidettävien ohjelmistoversioiden lukumääriin.

Seuraavaksi listataan kaavion osion 2.3 lisenssiehtoihin liittyvät mitattavat ominaisuudet.

### **2.3.1 Ohjelmiston jakaminen erikseen lisensoituihin komponentteihin**

Myös lisenssejä pitää hallita ja tämä vaatii toimittajalta resursseja. Koska ohjelmisto koostuu pääosin komponenteista, pitää toimittajan miettiä tarkoin, miten nämä komponentit lisensoidaan, jotta ylläpityö olisi mahdollisimman helppoa.

Vertailussa tutkitaan, miten ylläpidettävien lisenssien lukumäärään vaikuttaa se, kuinka komponentit on jaettu lisensoitaviin osiin.

### **2.3.2 Lisenssiehtojen tarkistamisen**

Ohjelmiston elinikä on pitkä ja sinä aikana markkinatilanteet muuttuvat varmasti paljon. Poliittisten olojen vaihtelut samoin kuin inflaatiokin vaikuttavat siihen, ovatko lisenssiehdot ja lisenssin hinta enää kannattavia toimittajalle. Käytännössä on epäedullista molempia osapuolia kohtaan tehdä sellainen sopimus, jota ei tarkisteta missään vaiheessa lisenssikautta. Tarkistamisen käänköpuoli on, että muutokset vaativat neuvotteluja asiakkaan ja toimittajan välillä.

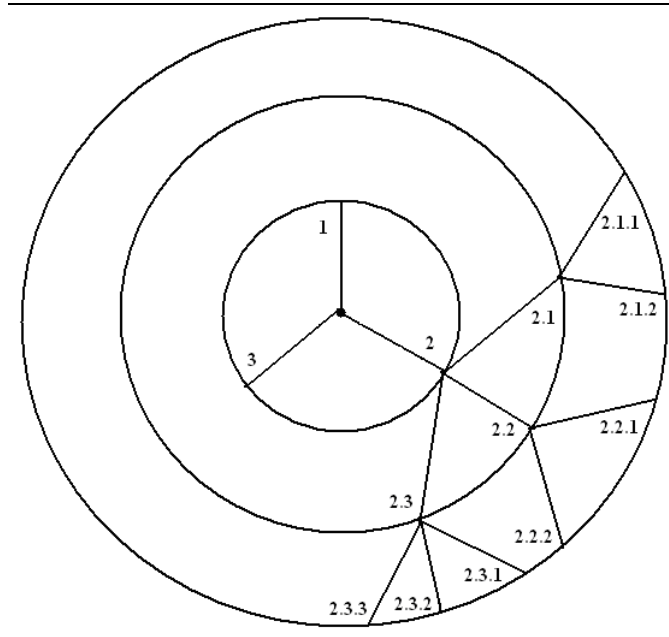
Vertailussa tutkitaan, miten lisenssiehtojen tarkistaminen vaikuttaa henkilöresursseihin pitkällä aikavälillä.

### **2.3.3 Lisenssikauden loppumisen aiheuttamat toimenpiteet**

Lisenssikauden loppuminen luo myös haasteita lisenssiehdoille. Jos asiakas päättyy sellaiseen ratkaisuun, että lisenssiä ei uusita, on jollakin tapaa hoidettava käytössä olevan ohjelmiston poistaminen. Käytännössä toimittaja ei voi olla varma, että ohjelmisto on todella hävitetty asiakkaan koneista, mutta lisenssiehdoissa voidaan määritellä, milloin ohjelman käyttämisestä tulee laitonta.

Vertailussa tutkitaan erilaisia keinoja poistaa ohjelmisto käytöstä.

Kuvassa 11 on CFD-kaavioon sijoitettu ylläpitoon vaikuttavat mitattavat ominaisuudet.



Kuva 11. Ylläpitoon liittyvät vertailtavat ominaisuudet CFD-kaaviossa.

### 8.3. Pääominaisuudet asiakkaan näkökulmasta

Asiakkaan näkökulmasta kaavion pääominaisuus 3 voidaan löytää neljä merkittävää asiaa. Näitä ovat ohjelmiston jakelureitin valinnan aiheuttamat riskit, hintojen jakaantuminen ohjelmiston elinkaarelle, asiakkaan henkilöresurssien tarve sekä tukipalveluiden vasteajat.

#### 3.1 Jakelureitin valinta

Riippumatta siitä, kuinka ohjelmisto toimitetaan asiakkaalle, kuljetukseen liittyy aina riskejä siitä, että arkaluontoista tietoa joutuu väärin käsiin. Yhtälailla posti tai kuriiripalvelu voi hukata kirjeen, tietoliikenneverkosta voidaan kaapata arkaluontoista tietoa tai ohjelmalevykettä kantavan ihmisen kimppuun voidaan hyökätä. Toiset näistä tavoista ovat kuitenkin luotettavampia kuin toiset. Esimerkiksi nykyisten salaustekniikoiden myötä internetin kautta voidaan välittää tietoa suhteellisen turvallisesti.

Vertailussa tutkitaan miten erilaiset jakelureitit vaikuttavat tietoturvariskeihin.

#### 3.2 Hintojen jakaantuminen ohjelmiston elinkaarelle

Hintojen ennakoitavuus on asiakkaalle tärkeää. Myös tuotteen kokonaishinta on helpompi määritellä, jos hinta on jo mahdollisimman aikaisessa vaiheessa tiedossa. Ohjelmiston kokonaishintaan vaikuttavat sekä alkuhinta että vuosimaksut, mutta myös yksittäisistä palveluista tai toiminnoista kertyvät maksut. Myös toimittajan on helpompi laskea lisenssistä saatavat tuotot, kun hinnoittelu on mahdollisimman selkeää.

Vertailussa tutkitaan millaiseksi tuotteen loppuhinta muodostuu erilaisilla laskutusmalleilla.

### 3.3 Asiakkaan tarvitsemat henkilöresurssit

Asiakkaalle on tärkeää pystyä ennakoimaan esimerkiksi versionnostoissa tarvittava henkilöstön määrä. Ylimääräisen henkilöstön pitäminen työsuhteessa vie turhaan resursseja, mutta yhtä hankalaa on myös irrottaa muuta henkilöstöä omista työtehtävistään. Asiakkaan ja toimittajan välinen sopimus siitä, kuka hoitaa minkäkin työvaiheen, auttaa sopivien ratkaisumallien etsimisessä myös tässä ongelmassa.

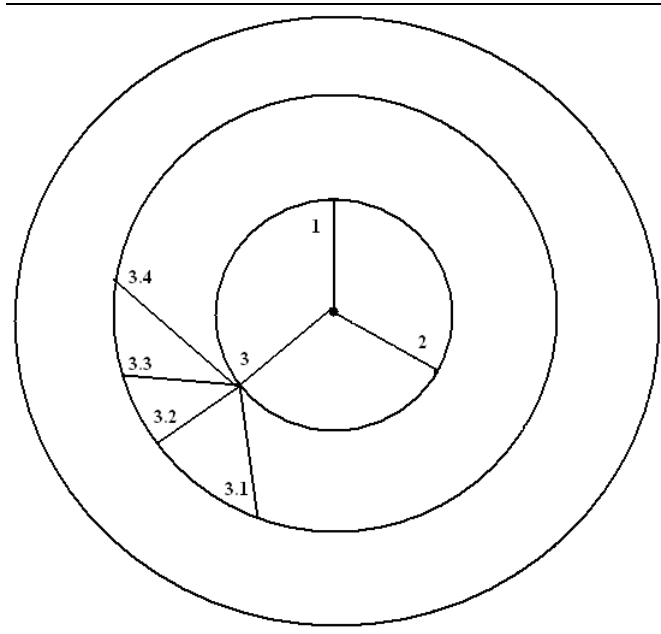
Vertailussa tutkitaan, miten erilaiset toimittajan ja asiakkaan väliset sopimukset työvaiheiden vastuuhenkilöistä vaikuttaa asiakkaan henkilöstöresursseihin.

### 3.4 Tukipalveluiden vasteajat

Ongelmatilanteiden ilmaantuessa asiakkaan on tärkeää saada yhteys toimittajaan mahdollisimman pian. Nopea vastaus nostaa myös asiakastyytyväisyyttä. Lisenssisopimuksessa voidaan sopia, että toimittaja järjestää tukipalveluita. Usein sopimuksessa sovitaan myös vasteaika, jonka sisällä asiakkaan viesteihin vastataan.

Vertailussa tutkitaan vasteajan vaikutusta asiakkaan ja toimittajan välisessä vuorovaikutuksessa.

Kuvaan 12 on piirrettyä asiakkaalle tärkeitä ominaisuuksia.



Kuva 12. Asiakkaan näkökulmaan liittyvät vertailtavat ominaisuudet CFD-kaaviossa.

## 9. Vertailun toteutus

Lisenssien vertailu määriteltyjen ominaisuuksien perusteella tehdään tutkimalla jokaisen yksittäisen ominaisuuden vaihtelun merkitystä lopputulokseen. Tämän vuoksi vertailussa on mukana kolme vaihtoehtoista lähestymistapaa. Vaihdot A ja B kuvaavat jonkin tietyn ominaisuuden suhteen kahta eri ääripäätä ja vaihtoehto C niiden välille sijoittuvaa arvoa.

Lähtökohtaisesti vaihtoehto A kuvaa ankarinta ääripäätä, siis sitä ominaisuuden arvoa, joka on ehdottomin, rajaavin ja joustamattomin. Vaihtoehto B taas kuvaa sitä tilannetta, jossa lisenssiehdoista on yritetty tehdä mahdollisimman joustavat ja vähän rajoittavat. Vaihtoehto C on suositeltavin ominaisuuden arvo tai ratkaisuvaihtoehto.

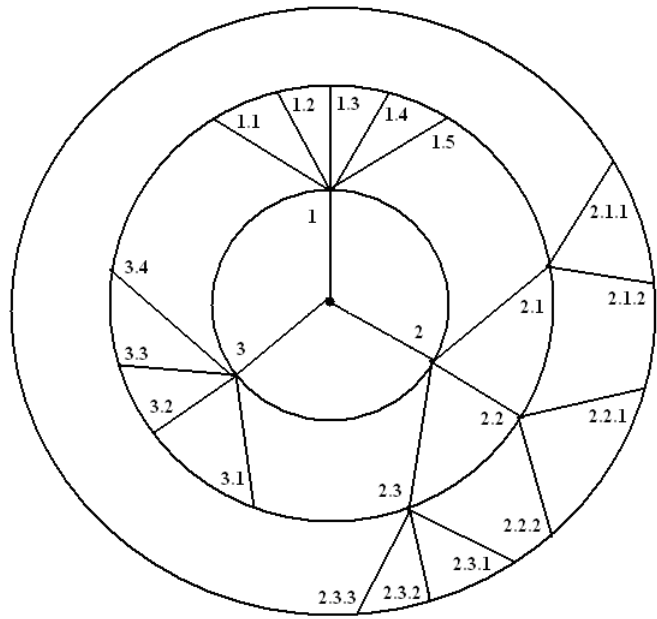
Testaaminen tapahtuu siten, että mietitään kutakin ominaisuutta erikseen sekä kuvataan sen ilmenemistä vaihtoehtoisissa A, B ja C. Lopuksi verrataan, minkälaisia tuloksia kukin kolmesta vaihtoehdosta on saanut.

### 9.1. Testitapaukset

Seuraavaksi kuvataan tarvittavat testitapaukset sekä vaihtoehtojen A, B ja C ratkaisumallit.

Esimerkeissä käytetyt hinnat ovat täysin kuvitteelliset ja ne on pyöristetty mahdollisuuksien mukaan tasaluvuiksi, sillä se helpottaa laskemista. Kaikki vertailussa käytetyt numerot ovat muutenkin vain viitteellisiä ja niiden on tarkoitus kuvata ainoastaan asioiden välisiä suhteita. Testitapauksissa oletusarvoisesti tuotteen eliniäksi on arvioitu 20 vuotta. Kaikki laskennoissa käytetyt numerot on myös pyöristetty kahteen desimaaliin.

Testitapaukset on numeroitu samoin kuin edellisessä kappaleessa esitetyt ominaisuuksien kuvaukset. Kaikki nämä ominaisuudet on myös piirretty näkyviin kuvaan 13.



Kuva 13. Kaikki testitapaukset CFD-kaaviossa.

### 1.1 Lisenssin alkuhinta ja laskutus lisäpalveluista

Vertaillaan korkeaa, matalaa ja kohtuullista tuotteen alkuhintaa sekä lisäpalveluiden korkean, matalan ja kohtuullisen hinnoittelun vaikutusta tuotteen loppuhintaan sen elinkaaren loputtua. Oletetaan, että lisäpalveluita ostetaan tasaisesti 2 kappaletta per vuosi.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Tuotteen alkuhinta	100 000 €	10 000 €	50 000 €
Yhden lisäpalvelun hinta	1 000 €	5 000 €	2 500 €
Tuotteen loppulinen hinta elinkaaren loputtua	(1 000 €* 2 kpl * 20 vuotta = 40 000 €) 140 000 €	(5 000 €* 2 kpl * 20 vuotta = 200 000 €) 210 000 €	(2 500 €* 2 kpl * 20 vuotta = 100 000 €) 150 000 €

## 1.2 Versionnoston pakollisuus ja koulutuspalvelun hinnoittelu

Vertaillaan versionnoston aiheuttamaan koulutustyöhön kuluvaan palkkasummaa tuotteen koko eliniän aikana siten, että tutkitaan, kuinka paljon versionnostosta saatavat kulut vähenevät koulutuskulujen vuoksi. Oletetaan, että versioita tulee yksi per vuosi ja versionnoston hinta on 1 000 € per kerta. Jokaisen versionnoston yhteydessä tarvitaan 2 päivää koulutusta eli 15 henkilötyötuntia. Kouluttajan palkaksi lasketaan 100 € per tunti.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Versionnoston pakollisuus ja versionnostojen lukumäärä	Ei pakollista versionnostoa (asiakas ottaa vain puolet versioista) 10 kpl	Pakollinen versionnosto 20 kpl	Pakollinen versionnosto 20 kpl
Versionnostoista saadut tuotot	10 000 €	20 000 €	20 000 €
Kouluttajan palkkakulut	Ei sovittua koulutuskiintiötä (toimittaja tarjoaa 2 pv koulutusta puolelle versionnostoista eli 100 €/h, 15 h ja 5 kertaa) 7 500 €	Sovittu täysi koulutus (100 €/h, 15 h/v ja 20 vuotta) 30 000 €	Sovittu 1 pv koulutuskiintiö (100 €/h, 7,5 h/v ja 20 vuotta) 15 000 €
Tuotot koulutuskulujen jälkeen	2 500 €	-10 000 €	5 000 €

### 1.3 Myyntikoneiston valinnan vaikutus lopulliseen tuottoon

Vertaillaan myyntikoneiston valinnan vaikutusta siihen, kuinka suuri osuus lisenssimyynnin tuotoista saadaan oman yrityksen käyttöön.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Valittu myyntikanava	Kolmannen osapuolen omistama jälleenmyyjä, joka saa tehdä ohjelmistoon omia muutoksiaan	Kolmannen osapuolen omistama jälleenmyyjä, joka ei saa tehdä ohjelmistoon omia muutoksia	Yrityksen oma myyntiosasto hoitaa ohjelmiston myynnin
Lisenssin myynnistä yritykselle palautuva %-osuus	40 %	80 %	100 %



#### 1.4 Asiakaskohtaisten versioiden lukumäärän vaikutus tuottoon

Vertaillaan, miten asiakkaan ohjelmaversioiden lukumäärä vaikuttaa lisenssistä saatuun tuottoon koko ohjelmiston elinkaaren aikana. Oletetaan, että ilman lisäkuluja ohjelmistosta saataisiin tuottoa sen elinkaaren aikana 100 000 €/per lisenssi. Tästä summasta vähennetään versioiden ylläpidon aiheuttamat kulut, joiden arvioidaan olevan 100 €/per vuosi per ohjelmaversio. Oletetaan myös, että ohjelmistosta tulee uusi versio kerran vuodessa.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Asiakkaan omistamat ohjelmistoversiot	Ei rajoituksia versioiden lukumäärään (asiakas haluaa kaikki versiot käyttöönsä) 20 kpl	Asiakkaan versioiden lukumäärää on rajoitettu kahteen kappaleeseen 2 kpl	Asiakas pakotetaan yhteen versioon 1 kpl
Versionhallinnan kustannukset tuotteen elinkaaren aikana	1. v 100 € 2. v 200 € 3. v 300 € jne... Yhteensä 21 000 €	1. v 100 € 2. v 200 € 3. v 200 € jne... Yhteensä 3 900 €	1. v 100 € 2. v 100 € 3. v 100 € jne... Yhteensä 2 000 €
Lisenssistä saatu hinta versionhallintakulujen jälkeen tuotteen elinkaaren aikana	79 000 €	96 100 €	98 000 €

### 1.5 Joustava reagointi markkinatilanteen muutoksiin

Tutkitaan, miten lisenssiehtojen muutokset vaikuttavat lisenssistä saatavaan tuottoon sen koko elinkaaren aikana. Oletetaan, että ilmanlisäkuluja lisenssistä saataisiin sen elinkaaren aikana 100 000 € Oletetaan myös, että markkinatilanteiden heilahteluiden ynnä muiden vastaavien muutosten vaikutus lisenssin hintaan on -5 % per vuosi.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Lisenssiehtojen muutosten vaikutus kokonais hintaan prosentteina	Ei mitään muutoksia lisenssiehtoihin, -5 % per vuosi, 20 vuotta	Lisenssiehtojen tarkastus 5 vuoden välein eli vuosina 1, 6, 11 ja 16, muina vuosina -5 % per vuosi	Lisenssiehtojen tarkastus vuoden välein
Lisenssikauden lopulla saatava hinta vähennettynä muutostannuksilla	1. v -5 000 € 2. v -4 750 € 3. v -4 512,50 € jne... Yhteensä 37 735,36 €	16. v -5 000 € 17. v 4 750 € 18. v -4 512,50 € € jne... Yhteensä 81 450,62 €	Kaikki vuodet -0 € Yhteensä 100 000 €
Muutos alkuperäisestä summasta	-62,26 %	-18,55 %	+/-0 %

### 2.1.1 Virhemahdollisuus datakonversioiden tekemisessä

Vertaillaan virheprosentin suuruutta riippuen siitä, millä menetelmällä datakonversio tehdään ja kuinka paljon yhteistyötä konversion tekemisessä käytetään.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Virheiden mahdollisuus prosentteina	Asiakas tekee konversion itse seuraten toimittajan ohjelistaa 80 %	Asiakas ajaa toimittajan lähettämän korjauskriptin 50 %	Asiakas ajaa toimittajan lähettämän korjauskriptin seuraten toimittajan ohjelistaa 10 %

### 2.1.2 Henkilöresurssien tarve

Vertaillaan, montako henkilöä toimittaja tarvitsee lupaamiensa ohjelmiston ylläpitoon liittyvien palveluiden tuottamiseen. Oletetaan, että yksi henkilö tekee vain yhtä työtehtävää.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Palveluiden tarjoamiseen vaadittavien henkilöiden lukumäärä	Täydet tuki-, ylläpito- ja koulutuspalvelut (eli tarvitaan kouluttaja, ohjelmoija, määrittelijä, joku hallinnollinen henkilö, tietokannan ylläpitäjä, tekninen tukihenkilö) 6 henkilöä	Toimittaja ei tarjoa mitään palveluita (eli tarvitaan vain yksi hallinnollinen henkilö) 1 henkilö	Toimittaja tarjoaa koulutus- ja asennuspalveluja lukuun ottamatta muut palvelut (eli kouluttajaa ja teknistä tukihenkilöä ei tarvita) 4 henkilöä

### 2.2.1 Versionnostojen pakollisuus ja versioiden lukumäärä per asiakas

Vertaillaan, montako erilaista versiota on ylläpidossa ohjelmiston elinkaaren loppuessa riippuen siitä, sallitaanko asiakkaan käyttää samasta ohjelmistosta useita versioita yhtä aikaa. Oletetaan, että ohjelmistosta tulee uusi versio kerran vuodessa. Oletetaan myös, että asiakkaalla on käytössään 2 lisäkomponenttia koko ohjelmiston elinkaaren ajan eli 20 vuotta. Oletetaan, että myös lisäkomponentteihin tulee päivitys kerran vuodessa.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Ylläpidettävien versioiden lukumäärä	Ei pakollista versionnostoa, ei rajoituksia eri versioiden määrään (asiakkaalla on kaikkia versioita käytössään sekä ohjelmistosta, että lisäkomponenteista) $20 + 20 + 20 = 60$ versiota	Joka toinen versionnosto pakollinen sekä ohjelmistolle, että lisäkomponenteille $2 + 2 + 2 = 6$ versiota	Versionnosto on pakko tehdä joka kerta sekä ohjelmistolle, että lisäkomponenteille $1 + 1 + 1 = 3$ versiota

### 2.2.2 Laiteympäristöjen lukumäärä per asiakas

Vertaillaan, kuinka laitealustojen lukumäärä vaikuttaa ylläpidettävien ohjelmistoversioiden lukumääriin. Oletetaan, että asiakkaita on 10 kappaletta.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Ylläpidettävien ohjelmistoversioiden lukumäärä	Ei rajoitusta - laitealustojen lukumäärästä (joka asiakkaalla on 6 eri laitealustaa) $10$ asiakasta * $6$ laitteistoa = $60$ versiota	Toimittaja lupaa alustatuen $4$ laitteistolle $10$ asiakasta * $4$ laitteistoa = $40$ versiota	Toimittaja lupaa alustatuen $2$ laitteistolle $10$ asiakasta * $2$ laitteistoa = $20$ versiota

### 2.3.1 Ohjelmiston jakaminen erikseen lisensoituihin komponentteihin

Vertaillaan, miten ohjelmiston jakaminen lisensoitaviin osiin vaikuttaa siihen, montako lisenssiä joudutaan ylläpitämään. Oletetaan, että ohjelmisto koostuu kaikkiaan 100 komponentista.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Ylläpidettävien lisenssien lukumäärä	Kaikki komponentit myydään erikseen omilla lisensseillä 100 kpl	Kaikki komponentit myydään yhden ison lisenssin alla 1 kpl	Ohjelmiston perustoiminnallisuus 50 komponenttia myydään yhden lisenssin alla ja loput erikseen $1 + 50 =$ 51 kpl

### 2.3.2 Lisenssiehtojen tarkistamisen

Tutkitaan lisenssiehtojen tarkistamisen vaikutusta lisenssin hintaan ohjelmiston elinkaaren lopussa. Oletetaan, että ehtojen tarkistamiseen menee aikaa yksi työviikko eli 37,5 henkilötyötuntia ja henkilön tuntipalkka on 100 €. Oletetaan myös, että lisenssisopimuksen hinta on ensimmäisenä vuonna 100 000 € ja jokainen lisenssiehtojen tarkastuskerta nostaa vuosituottoja 5 %.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Tarkistusten lukumäärä	Ei tarkistuksia	Tarkistukset 5 vuoden välein eli vuosina 6, 11 ja 16 = 3 kpl	Tarkistukset 2 vuoden välein eli vuosina 3, 5, 7, 9, 11, 13, 15, 17 ja 19 = 9 kpl
Tarkistajan palkka	0 €	$37,5 * 100 * 3$ = 11 250 €	$37,5 * 100 * 9$ = 33 750 €
Lisenssin tuotot	$100\ 000 * 20 =$ 2 000 000 €	1. v 100 000 € 6. v 105 000 € 11. v 110 250 € jne... = 2 155 062,50 €	1. v 100 000 € 3. v 105 000 € 5. v 110 250 € jne... = 2 515 578,58 €
Tuotot tarkastuskulujen jälkeen	2 000 000 €	2 143 812,50 €	2 481 828,58 €

### 2.3.3 Lisenssikauden loppumisen aiheuttamat toimenpiteet

Tutkitaan, kuinka pitkän toimintakatkon asiakkaan aiheuttaa erilaiset lisenssikauden loppuun määritellyt toimintamallit ohjelmiston poistamiseksi käytöstä. Oletetaan, että asiakas tekee kirjaimellisesti sen, mitä lisenssiehdoissa sanotaan, mutta voi myös tehdä näiden toimenpiteiden jälkeen jatkaa toimintaansa lain rajoissa miten haluaa.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Lisenssiehdoissa määritelty toiminta	Asennuslevykkeet tulee palauttaa lisenssikauden lopussa.	Lisenssikauden loputtua ohjelmisto tulee poistaa asiakkaan koneilta.	Lisenssiehdoissa sanotaan, että ohjelmiston käyttö muuttuu laittomaksi lisenssikauden loputtua, ellei uutta sopimusta ole syntynyt.
Asiakkaan reagointi tilanteeseen	Asiakas on kopioinut asennuslevykkeet ne saatuaan. Lisenssikauden lopussa asiakas palauttaa alkuperäiset levykkeet, mutta jatkaa ohjelmiston käyttöä kopioisista.	Asiakas kyllä poistaa ohjelmistot, mutta asentaa ne uudelleen ja palauttaa tiedot varmuuskopioisista.	Asiakkaan toiminta muuttuu laittomaksi sopimuskauden päätyttyä, ellei uutta sopimusta ole tehty. Jos asiakas jatkaa ohjelmiston käyttöä, hän joutuu lain edessä vastaamaan tekemisistään.
Toimenpiteen aiheuttama toimintakatko	0 pv	31 pv	Toimittajan näkökannalta pysyvä

### 3.1 Jakelureitin valinta

Vertaillaan jakelureitin valinnan aiheuttamaa riskiprosenttia. Turvallisuusalan yleisten periaatteiden mukaan tässäkin vertailussa lähdetään siitä, että yksittäiseen ihmiseen (tai ihmisryhmään) ei voi luottaa. Tämän vuoksi on tehtävä kaikki mahdolliset varmistukset siitä, että mikään työvaihe ei jää pelkästään yhden ihmisen (tai ihmisryhmän) tehtäväksi ja että kaikki työvaiheet ja niiden tekijät on jäljitettävissä ja varmistettavissa.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Valittu jakelureitti	Toimittajan valitsema henkilö kuljettaa levykkeen asiakkaalle.	Asennuslevyke toimitetaan postin kautta kirjatuna kirjeenä	Tiedot toimitetaan asiakkaalle internetin välityksellä salattuna.
Riski	Henkilö ei ole luotettava tai hänet ryöstetään	Kirje hukkuu tai joutuu sivullisten käsiin.	Sivulliset purkavat salauksen ja kaappaavat tiedot.
Riskin toteutumisen todennäköisyys	(Valittu henkilö joko on luotettava tai sitten ei ja hänet joko voidaan ryöstää tai sitten ei.) 50 %	(Postissa seurataan kirjattujen kirjeiden kulkua ja työvaiheisiin osallistuu aina useita henkilöitä kerralla. Aina joskus kirjeitä kuitenkin katoaa.) 10 %	(Työvaiheeseen ei osallistu ulkopuolisia henkilöitä ja salaustekniikat ovat nykyään varsin luotettavia. Myöskään siirron aikataulut eivät helposti leviä ulkopuolisten tietoon.) 1 %



### 3.2 Hintojen jakaantuminen ohjelmiston elinkaarelle

Tutkitaan, miten asiakas pystyy vastaamaan ohjelmiston elinkaaren aikana eteentuleviin kuluihin. Oletetaan, että asiakkaan vuosibudjetti ohjelmistoa varten on 5 000 € eli 100 000 € koko tuotteen elinkaaren aikana. Sovitaan myös, että tasan 100 000 € tulee käytetyksi kaikissa vertailtavissa vaihtoehtoissa. Oletetaan myös, että joka vuosi asiakas joutuu ostamaan kaksi maksullista lisäpalvelua ohjelmistoon.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Hinnoitteluperiaate	Täysin epäsäännölliset vuosittaiset kulut	Epäsäännölliset vuosittaiset kulut, vaihteluraja +/- 1 000 € (ylimenevät kulut siirretään seuraavalle vuodelle)	Säännöllinen 5 000 € vuosimaksu
Vuosikulut tuotteen elinkaaren aikana euroina	1. v 5 000 + 1 000 2. v 2 000 + 2 000 3. v 5 000 + 3 000 4. v 2 000 + 1 000 5. v 3 000 + 2 000 6. v 7 000 + 1 000 7. v 1 000 + 3 000 8. v 2 000 + 3 000 9. v 5 000 + 2 000 10. v 1 000 + 1 000 11. v 2 000 + 4 000	1. v 5 000 + 1 000 2. v 2 000 + 2 000 3. v 5 000 + 1 000 4. v 2 000 + 3 000 5. v 3 000 + 2 000 6. v 5 000 + 1 000 7. v 3 000 + 3 000 8. v 2 000 + 3 000 9. v 5 000 + 1 000 10. v 1 000 + 2 000 11. v 2 000 + 4 000	1. – 20. v 2 palvelua, hinta yhteensä 5 000

	12. v 3 000 + 2 000	12. v 3 000 + 2 000	
	13. v 2 000 + 3 000	13. v 2 000 + 3 000	
	14. v 1 000 + 4 000	14. v 1 000 + 4 000	
	15. v 2 000 + 2 000	15. v 2 000 + 2 000	
	16. v 4 000 + 4 000	16. v 4 000 + 2 000	
	17. v 1 000 + 3 000	17. v 1 000 + 5 000	
	18. v 2 000 + 2 000	18. v 2 000 + 2 000	
	19. v 3 000 + 2 000	19. v 3 000 + 2 000	
	20. v 1 000 + 1 000	20. v 1 000 + 1 000	
Kuinka monena vuonna kulut menevät yli bud- jetin	6	8	0

### 3.3 Asiakkaan tarvitsemat henkilöresurssit

Tutkitaan, miten tulevien toimenpiteiden tehtäväjako ja ennakoitavuus vaikuttaa asiakkaan henkilöresurssien tarpeeseen. Oletetaan, että asiakkaalla on kahden henkilön resurssit käytössään ohjelmiston vaatimia toimenpiteitä varten.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Tapahtuma ja henkilöiden tarve	Äkillinen versionnosto, asennus ja datakonversiot sekä koulutus = 3 henkilöä	Ennalta sovittu versionnosto. Asiakas tekee asennuksen, datakonversiot ja koulutuksen = 3 henkilöä (asiakkaalla on aikaa etsiä yksi henkilö lisää)	Ennalta sovittu versionnosto. Asiakas asentaa ja tekee datakonversiot. Toimittaja kouluttaa. = 2 henkilöä
Henkilöresurssien vastaaminen tarpeeseen	-1 henkilö	0 henkilöä	0 henkilöä

### 3.4 Tukipalveluiden vasteajat

Tutkitaan, kuinka nopeasti asiakas tavoittaa toimittajan tukihenkilön ilmoitettuaan virheestä.

Ominaisuus	Vaihtoehto A	Vaihtoehto B	Vaihtoehto C
Sopimus ja vasteaika	Ei sovittua tukipalvelua. = ei vastausta	Sovittu tuki, ei vasteaikaa. = arviolta vastauksen saaminen kestää 2 vkoa	Sovittu tuki ja 2 pv vasteaika. = max 2 pv

## 9.2. Asteikot

Asteikot on kaikilla ominaisuuksilla numeroitu välille 1 – 5 siten, että 1 on pienin hyväksyttävä pistearvo ja 5 suurin saavutettavissa oleva arvosana.

Seuraavaksi luetellaan kunkin ominaisuuden kohdalla, millaisilla tuloksilla saavutetaan minimiarvo ja minkälaiset tulokset vaaditaan maksimiarvoon. Muut arvot maksimin ja minimin väliltä jakautuvat asteikon keskivälille tasavälein jaetuille pistearvoille 2 - 4.

Merkit "→" ja "←" on luettava siten, että esimerkiksi "5 henkilöä →" tarkoittaa "viisi henkilöä tai enemmän" ja "← 5 kpl" tarkoittaa "5 kappaletta tai vähemmän". Alla olevan listan kohdat on numeroitu samoin kuin kuvan 13 tehtävät.

### 1.1 Lisenssin alkuhinta ja laskutus lisäpalveluista

Minimituotto, joka hyväksytään, on 100 000 € Maksimiarvosanan saa, jos tuotto on yli 145 000 €

Arvo- sana 1	2	3	4	Arvo- sana 5
100 000 - 114 999,99 €	115 000 - 124 999,99 €	125 000 - 134 999,99 €	135 000 - 144 999,99 €	145 000 € →

### 1.2 Versionnoston pakollisuus ja koulutuspalvelun hinnoittelu

Minimituotto, joka versionnostoista halutaan, on 1 000 € Maksimipistemäärän saa, jos tuotot ovat yli 9 000 €

Arvo- sana 1	2	3	4	Arvo- sana 5
1 000 - 2 999,99 €	3 000 - 4 999,99 €	5 000 - 6 999,99 €	7 000 - 8 999,99 €	9 000 € →

### 1.3 Myyntikoneiston valinnan vaikutus lopulliseen tuottoon

Myydyistä lisensseistä halutaan mahdollisimman suuri osa tuotoista omalle yritykselle. Pienin prosenttiosuus, joka vielä hyväksytään, on 10 %, ihannetilanteessa tuotoista saadaan itselle yli 80 %.

Arvo- sana 1	2	3	4	Arvo- sana 5
10 – 19 %	20 – 39 %	40 – 59 %	60 - 79 %	80 % →

#### 1.4 Asiakaskohtaisten versioiden lukumäärän vaikutus tuottoon

Lisenssien myynnistä halutaan tuottoja vielä versionhallintakustannusten jälkeenkin. Minimisumma, joka hyväksytään, on 50 000 € Maksimipistemäärän saa, jos summa on 90 000 €tai yli.

Arvo- sana 1	2	3	4	Arvo- sana 5
50 000 - 59 999,99 €	60 000 - 69 999,99 €	70 000 - 79 999,99 €	80 000 - 89 999,99 €	90 000 €→

#### 1.5 Joustava reagointi markkinatilanteen muutoksiin

Jos yritys ei saa alun perin suunnittelemaansa hintaa tuotteesta, sen talous kärsii. Siksi pienin sallittavissa oleva heitto haluttuun loppusummaan on -20%. Sen sijaan positiivinen muutos hinnassa on toivottavaa, joten jo 11 % hinnannousulla saa täydet pisteet.

Arvo- sana 1	2	3	4	Arvo- sana 5
-20 - -11 %	-10 - -6 %	-5 - +5 %	+6 - +10 %	+11 % →

#### 2.1.1 Virhemahdollisuus datakonversioiden tekemisessä

Datakonversioiden merkitys on suuri, ja virheitä pitää pyrkiä välttämään kaikin mahdollisin keinoin. Siksi suurin hyväksyttävissä oleva riskiprosentti on enintään 80 % ja hyvän tuloksen saa sillä, että riskiprosentti on 10 % tai vähemmän.

Arvo- sana 1	2	3	4	Arvo- sana 5
80 - 70 %	50 - 69 %	30 - 49 %	11 - 29 %	← 10 %

### 2.1.2 Henkilöresurssien tarve

Henkilöresurssit ovat kalliita, joten henkilöstön lukumäärään on kiinnitettävä huomiota jo siinä vaiheessa, kun toimittaja miettii, mitä palveluita se pystyy asiakkaalle tarjoamaan. Halvimmalla toimittaja pääsee, jos töistä vastaa vain yksi henkilö. Siksi maksimiarvosanan saa, jos tarve on vain yhdelle henkilölle. Tässä yhteydessä on laskettu niin, että jokainen työntekijä vähentää maksimiarvosanaa yhdellä.

Arvo- sana 1	2	3	4	Arvo- sana 5
5 henkilöä	4 henkilöä	3 henkilöä	2 henkilöä	1 henkilö

### 2.2.1 Versionnostojen pakollisuus ja versioiden lukumäärä per asiakas

Versioiden ylläpito on kallista, siksi toimenpiteitä vaativien versioiden lukumäärää pyritään vähentämään minimiin. Maksimiarvosanan saa, jos ylläpidossa on 5 tai alle versiota. Minimiarvosanan saa, jos versioiden lukumäärä on 21 - 25 kappaletta.

Arvo- sana 1	2	3	4	Arvo- sana 5
21 – 25 kpl	16 – 20 kpl	11 – 15 kpl	6 – 10 kpl	← 5 kpl

### 2.2.2 Laiteympäristöjen lukumäärä per asiakas

Jokaista laiteympäristöä, jossa ohjelmiston tulee toimia, varten pitää ohjelmistosta olla oma versio. Tämän vuoksi myös asiakkaalle sallittujen ja tuettujen laitteiden määrää on hyödyllistä rajoittaa. Tämän vuoksi maksimiarvosanan saa, jos ylläpitoa tarvitaan 5 laitealustalle tai alle. Minimiarvosana tulee, jos laitetukea joudutaan antamaan 21 – 25 laitteelle.

Arvo- sana 1	2	3	4	Arvo- sana 5
21 – 25 kpl	16 – 20 kpl	11 – 15 kpl	6 – 10 kpl	← 5 kpl

### 2.3.1 Ohjelmiston jakaminen erikseen lisensoituihin komponentteihin

Ylläpidettävien lisenssien lukumäärä kannattaa pitää mahdollisimman alhaisena, sillä niiden ylläpidosta seuraa kustannuksia. Siksi maksimiarvosanan saa, jos ylläpidossa on 20 tai alle lisenssiä. Minimiarvosanan saa, jos ylläpidossa on 100 tai yli lisenssiä.

Arvo- sana 1	2	3	4	Arvo- sana 5
100 kpl →	99 – 71 kpl	70 – 46 kpl	45 – 21 kpl	← 20 kpl

### 2.3.2 Lisenssiehtojen tarkistamisen

Tuottojen halutaan nousevan mahdollisimman paljon ohjelmiston elinkaaren aikana. Minimiarvosanan saa sillä, että tuotot pysyvät alkuhintaan nähden +/- nollana eli 2 000 000 €, maksimi arvosanan saa sillä, että hinta nousee maksimimäärän ohjelmiston elinkaaren aikana eli 2 707 074,02 €

Arvo- sana 1	2	3	4	Arvo- sana 5
← 2 000 000 €	2 000 000,0 1 – 2 250 000 €	2 250 000,0 1 – 2 500 000 €	2 500 000,0 1 – 2 700 000 €	2 700 000,0 1 €→

### 2.3.3 Lisenssikauden loppumisen aiheuttamat toimenpiteet

Käytännössä toimittaja haluaa, että ohjelmisto poistetaan heti lisenssikauden loputtua ja asiakas lakkaa saamasta ohjelmiston tarjoamaa hyötyä joko lopullisesti tai vähintään siihen asti, kunnes lisenssisopimus jälleen uusitaan. Tämän vuoksi maksimiarvosanan saa vain pysyvällä toimintakatkolla, jollaiseksi luetaan tässä yhteydessä yli 4 kk kestävä tauko. Miniarvosanaksi hyväksytään 1 kk kestävä toiminnan jatkuminen lisenssisopimuksen päättymisen jälkeen.

Arvo- sana 1	2	3	4	Arvo- sana 5
31 pv	62 pv	93 pv	124 pv	125 pv →

### 3.1 Jakelureitin valinta

Jakelureitin valintaan vaikuttaa sen turvallisuus. Jos asennuslevykkeet tai datakonversiotiedot leviävät ulkopuolisten tahojen käsiin, siinä saatetaan menettää kallisarvoista turvallisuuteen vaikuttavaa tietoa. Siksi jakelutavan tuoman riskiprosentin pitää olla 1 tai alle, että se voi saada maksimiarvosanan. Minimiarvosanan voi saada, jos riskiprosentti on 8 - 10 %.

Arvosana 1	2	3	4	Arvosana 5
8 - 10 %	6 - 7,99 %	4 - 5,99 %	1,01 - 3,99 %	← 1 %

### 3.2 Hintojen jakaantuminen ohjelmiston elinkaarelle

Jos ohjelmiston lisäkustannukset ylittävät budjetin salliman määrän, voi olla, ettei niihin löydy lisärahoitusta ja ne jäävät kokonaan hankkimatta. Tästä syystä maksimiarvosanan voi saada vain, mikäli kulut eivät minään vuonna yllä yli budjetin. Minimiarvosanan saa, jos ylitysvuodet jäävät 5.

Arvosana 1	2	3	4	Arvosana 5
4 - 5 kpl	3 kpl	2 kpl	1 kpl	0 kpl

### 3.3 Asiakkaan tarvitsemat henkilöresurssit

Henkilöresurssit ovat yritykselle suuri menoerä ja siksi työt halutaan suunnitella siten, että niistä selvitään mahdollisimman tehokkaasti. Maksimipistemäärän saa siitä, että olemassa olevat henkilöresurssit riittävät ja lisähenkilöiden tarve on 0. Minimipisteet saa, mikäli ylimääräisten henkilöiden lukumäärä jää 5.

Arvosana 1	2	3	4	Arvosana 5
-4 - -5 henkilöä	-3 henkilöä	-2 henkilöä	-1 henkilöä	0 henkilöä



### 3.4 Tukipalveluiden vasteajat

Asiakas haluaa vastauksen esittämiinsä kysymyksiin mahdollisimman pian. Siksi maksimipisteet saa, jos vasteaika on alle 2 päivää. Minimipisteet saa, jos vastaus tulee 14 päivän aikana.

Arvo- sana 1	2	3	4	Arvo- sana 5
14 pv	11 – 13 pv	7 – 10 pv	3 – 6 pv	← 2 pv

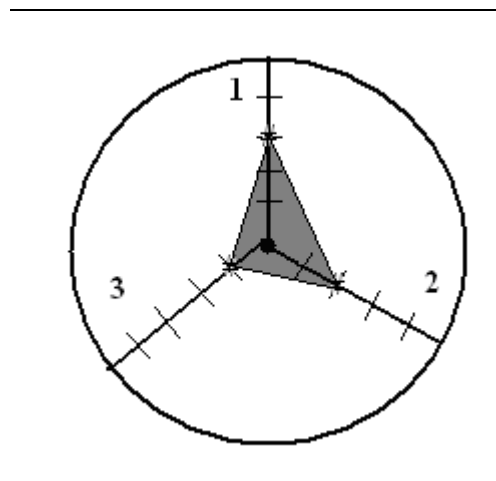
### 9.3. Raja-arvot

Kun vertailua tehdään toimittajan näkökulmasta, toimittajan ominaisuudet ovat tärkeämpiä kuin asiakkaan näkökulmasta esitetyt ominaisuudet. Siksi toimittajalle tärkeät asiat ovat arvostettuna korkeammalle kuin asiakkaalle tärkeät ominaisuudet. Toimittajan pääominaisuuksista kustannusseikat ovat tärkeämpiä kuin ylläpitoon liittyvät, joten ne on myös arvoitettu korkeammalle.

Vertailussa halutaan välttää liian kriittisiä raja-arvoja, sillä kaikki kolme vertailun kohdetta halutaan pitää mukana mahdollisimman kauan.

Tästä seuraa, että asiakkaan näkökulman ominaisuuksien raja-arvoksi valitaan 1, toimittajan ylläpito-ominaisuuksille valitaan raja-arvoksi 2 ja tärkeimmille eli toimittajan kustannusominaisuuksille valitaan raja-arvoksi 3.

Kuvaan 14 on piirrettyä Kiviati-diagrammikuvaan kaikkien kolmen pääominaisuuden osalta asteikot ja raja-arvot. Myös raja-arvojen peittämä alue on väritetty valmiiksi näkyviin.



Kuva 14. Kiviati-diagrammiin sijoitetut asteikot sekä raja-arvot ja niiden peittämä alue.

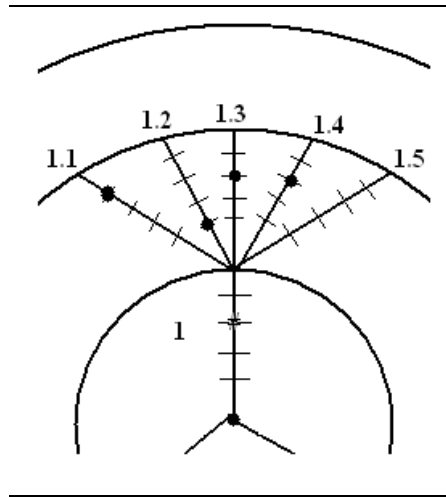
## 10. Testaustulokset, niiden arviointi ja vertailu

### 10.1. Vaihtoehto A

Vaihtoehdon A saamat tulokset:

Tehtävän numero	Testauksen tulos	Arvosana
1.1	140 000 €	4
1.2	2 500 €	1
1.3	40 %	3
1.4	79 000 €	3
1.5	-62,26 %	-
2.1.1	80 %	1
2.1.2	6 henkilöä	-
2.2.1	60 kpl	-
2.2.2	60 kpl	-
2.3.1	100 kpl	1
2.3.2	2 000 000 €	1
2.3.3	0 pv	-
3.1	50 %	-
3.2	6 kpl	-
3.3	-1 henkilöä	4
3.4	Ei vastausta	-

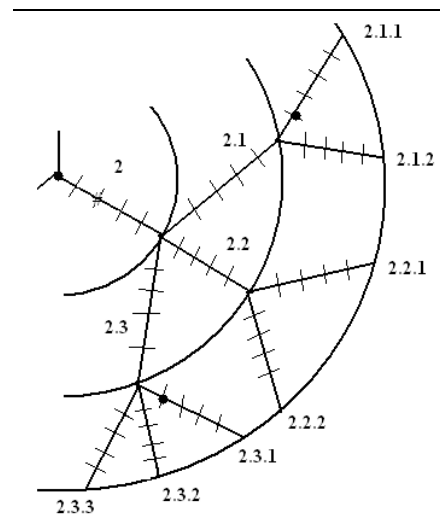
Vaihtoehto A ei saanut tarvittavia pisteitä toimittajan kustannusominaisuuksista selvittääkseen vertailussa jatkoon. Tämä johtuu kahdesta seikasta. Ensimmäkin arvosana tehtävästä 1.2 jää alle raja-arvon. Lisäksi A ei saanut lainkaan arvosanaan oikeuttavaa tulosta tehtävästä 1.5. Kuvassa 15 on eritelty vaihtoehdon A kustannusominaisuuksista saamat pisteet.



Kuva 15. Vaihtoehto A:n saamat kustannusominaisuuksien pisteet sijoitettuna CFD-kaavioon.

Pääominaisuuden yksi, toimittajan kustannusominaisuudet, tulosten mukaan todellisuudessa vaihtoehto A:n tulosten vertailua ei jatkettaisi, mutta että saataisiin vertailua eri vaihtoehtojen välille myös muiden pääominaisuuksien osalta, viedään nyt muutkin vaihtoehto A:n saamat tulokset CFD-kaavioon.

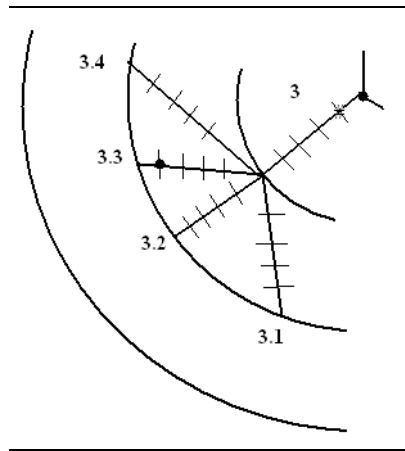
Kuva 16 osoittaa, että vaihtoehto A ei saanut tarvittavia tuloksia myöskään pääominaisuuden kaksi, toimittajan ylläpito-ominaisuudet, kohdalta. Siellä seitsemästä mitatusta ominaisuudesta vain kahteen saatiin tulos ja nämäkin jäivät molemmat alle raja-arvon.



Kuva 16. Vaihtoehto A:n saamat ylläpito-ominaisuuksien pisteet sijoitettuna CFD-kaavioon.

Kuvassa 17 näkyy vaihtoehto A:n saamat tulokset kolmannen pääominaisuuden, asiakkaalle tärkeiden ominaisuuksien, osalta. Tulokset ovat heikot myös tällä osa-alueella,

sillä neljästä vertailuun valitusta mitattavasta ominaisuudesta vain yksi on saanut arvostuksen.



Kuva 17. Vaihtoehto A:n saamat asiakkaan ominaisuuksien pisteet sijoitettuna CFD-kaavioon.

Vaihtoehto A putoaa pois vertailusta siis jokaisen yksittäisen pääominaisuutensa osalta. Tästä voimme päätellä, että ankara ja joustamaton näkökulma lisenssiehtoihin ja hinnoitteluun ei ole kannattava.

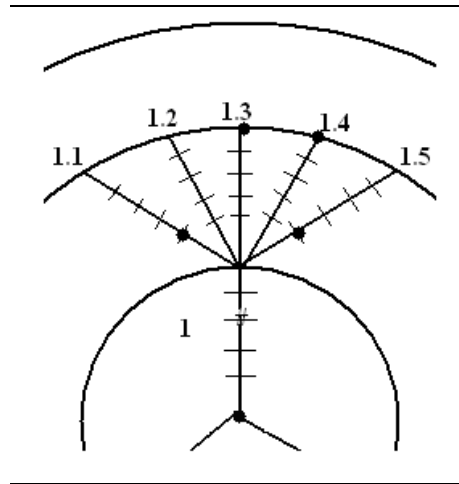
## 10.2. Vaihtoehto B

Seuraavaksi listataan vaihtoehdon B saamat tulokset vertailussa:

Tehtävän numero	Testauksen tulos	Arvosana
1.1	210 000 €	1
1.2	-10 000 €	-
1.3	80 %	5
1.4	96 100 €	5
1.5	-18,55 %	1
2.1.1	50 %	2
2.1.2	1 henkilö	5
2.2.1	6 kpl	4
2.2.2	40 kpl	-
2.3.1	1 kpl	5
2.3.2	2 143 812,50 €	2
2.3.3	31 pv	1
3.1	10 %	1
3.2	8 kpl	-
3.3	0 henkilöä	5
3.4	14 pv	1

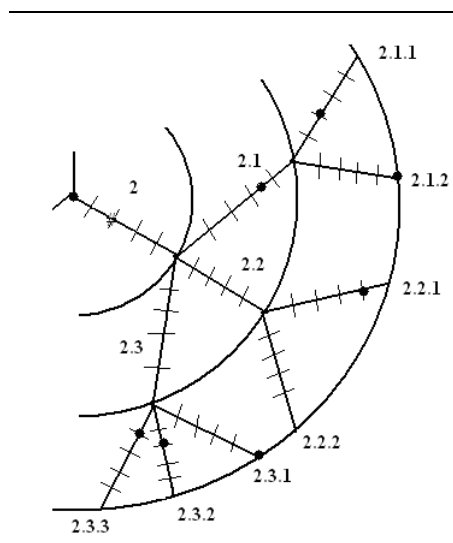
Kuten kuvasta 18 näkyy, niin myöskään vaihtoehto B ei saanut kelvollisia tuloksia vertailusta toimittajan kustannusominaisuuksista. Vaikka B sai tehtävistä 1.3 ja 1.4 täydet pisteet, niin pisteet ominaisuuksista 1.1 ja 1.5 eivät yllä raja-arvoon. Lisäksi B ei saanut ollenkaan tulosta kohdasta 1.2.

Näistä syistä johtuen vaihtoehto B ei myöskään voi jatkaa vertailussa eteenpäin. Silti vaihtoehdon B loputkin tulokset on sijoitettu CFD-kaavioon, että vertailua voitaisiin tehdä eri vaihtoehtojen välillä myös muista kahdesta pääominaisuudesta.



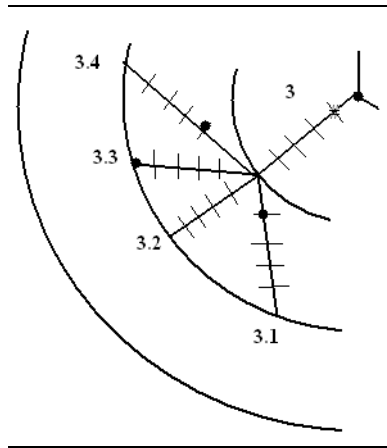
Kuva 18. Vaihtoehto B:n saamat kustannusominaisuuksien pisteet sijoitettuna CFD-kaavioon.

Kuva 19 osoittaa, että B ei ole saanut lopullista arvosanaa pääominaisuudelle kaksi, toimittajan ylläpito-ominaisuudet, sillä se ei ole saanut arvosanoille yltävää tulosta mitattavasta ominaisuudesta 2.2.2. Lisäksi ominaisuuden 2.3.3 saama arvosana ei yllä pääominaisuudelle määriteltyyn raja-arvoon 2.



Kuva 19. Vaihtoehto B:n saamat ylläpito-ominaisuuksien pisteet sijoitettuna CFD-kaavioon.

Vaihtoehto B ei saa lopullista arvosanaa myöskään kolmannelle pääominaisuudelle eli asiakkaan näkökulmasta tärkeistä asioista. Tämä johtuu siitä, että se ei ole saanut ollenkaan arvosanaa ominaisuudesta 3.2 kuten näkyy kuvasta 20.



Kuva 20. Vaihtoehto B:n saamat asiakkaan ominaisuuksien pisteet sijoitettuna CFD -kaavioon.

Kuten vaihtoehto A:llekin kävi, jää myös vaihtoehto B täysin vaille vertailukelpoisia tuloksia kaikista määritellyistä pääominaisuuksista. Tästä voimme päätellä, että myöskään kaiken salliva ja kaikkeen suostuva näkökulma lisenssiehtoihin ei ole kannattava.

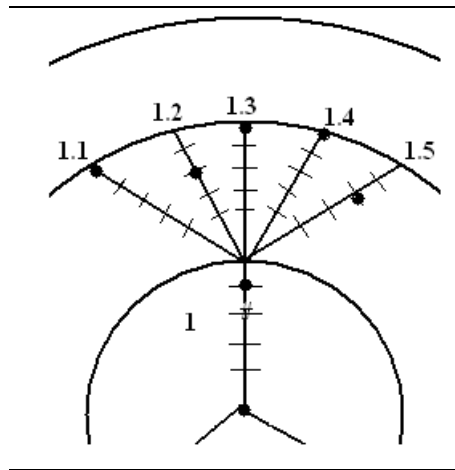
### 10.3. Vaihtoehto C

Tässä on vaihtoehto C:n saamat tulokset:

Tehtävän numero	Testauksen tulos	Arvosana
1.1	150 000 €	5
1.2	5 000 €	3
1.3	100 %	5
1.4	98 000 €	5
1.5	0 %	3
2.1.1	10 %	5
2.1.2	4 henkilöä	2
2.2.1	3 kpl	5
2.2.2	20 kpl	2
2.3.1	51 kpl	3
2.3.2	2 481 828,58 €	3
2.3.3	Pysyvä	5
3.1	1 %	5
3.2	0 kpl	5
3.3	0 henkilöä	5
3.4	2 pv	5

Kuten kuva 21 kertoo, on vaihtoehto C saanut hyväksyttävät pisteet toimittajan kustannusominaisuuksista jatkoon pääsyä varten. C on saanut arvosanan kaikista viidestä vertailussa olleesta mitattavasta ominaisuudesta, ja kaikki arvosanat ovat samat tai korkeammat kuin vaadittu raja-arvo 3.

Vertailtujen, mitattavien viiden ominaisuuden keskiarvo on 4, joten pääominaisuus yksi saa arvosanakseen 4. Tämä ylittää vaaditun raja-arvon, joten vaihtoehto C:n tulosten vertailua voidaan jatkaa.



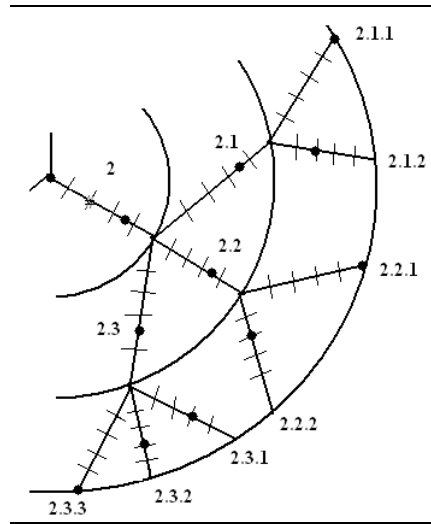
Kuva 21. Vaihtoehto C:n saamat kustannusominaisuuksien pisteet sijoitettuna CFD -kaavioon.

Kuten kuva 22 osoittaa, vaihtoehto C on läpäissyt vaaditut testit myös toisen pääominaisuuden eli toimittajan ylläpito-ominaisuuksien, mitattavat ominaisuudet.

Ominaisuuksien 2.1.1 ja 2.1.2 keskiarvo on 3.5, joka siirtyy suoraan ominaisuuden 2.1 arvosanaksi. Samoin ominaisuuksien 2.2.1 ja 2.2.2 sekä 2.3.1, 2.3.2 ja 2.3.3 keskiarvot ovat 3.5, jotka siis siirtyvät myös ominaisuuksille 2.2 ja 2.3.

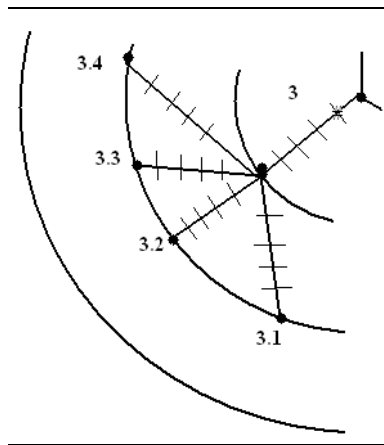
Koska 3.5 ylittää pääominaisuuden 2 raja-arvon, joka on 2, lasketaan keskiarvo ominaisuuksille 2.1, 2.2 ja 2.3. Tämä keskiarvo on 3.5, josta tulee pääominaisuuden 2 arvosana. Tämä arvosana on hyväksyttävä, joten vertailua voidaan jatkaa kolmanteen pääominaisuuteen.





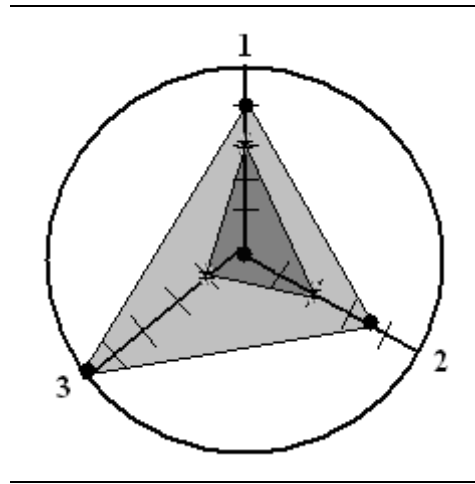
Kuva 22. Vaihtoehto C:n saamat ylläpito-ominaisuuksien pisteet sijoitettuna CFD-kaavioon.

Koska vaihtoehto C on saanut täydet pisteet jokaisesta pääominaisuuden 3, asiakkaan ominaisuudet, mitattavasta ominaisuudesta, myös pääominaisuus 3 saa arvosanakseen 5. Tämä ylittää reilusti pääominaisuudelle merkityn raja-arvon 1, joten vaihtoehto C sai kaikista kolmesta pääominaisuudesta vertailukelpoisen loppusumman. Tämä näkyy myös kuvassa 23.



Kuva 23. Vaihtoehto C:n saamat asiakkaan ominaisuuksien pisteet sijoitettuna CFD-kaavioon.

Nyt vaihtoehto C:n saamat tulokset voidaan sijoittaa Kiviati –diagrammiin. Tämä kaavio nähdään kuvassa 24. Kuten kuvasta voidaan havaita, on raja-arvojen peittämä, tummempi alue selvästi pienempi kuin saatujen arvosanojen muodostama, vaaleampi alue. Tämä tarkoittaa sitä, että vaihtoehto C on selvinnyt kaikista mitatuista ominaisuuksista paremmin, kuin mikä olisi ollut pienin vielä hyväksyttävä arvosana.



Kuva 24. Vaihtoehto C Kiviat -diagrammissa.

Koska vaihtoehto C oli ainoa vertailussa ollut kohde, joka sai hyväksyttävän arvosanan kaikista mitattavista ominaisuuksista ja koska kaikkien pääominaisuuksien osalta se sai myös korkeammat arvosanat kuin niille määritellyt raja-arvot, se voitti vertailun. Jos muut vaihtoehdot olisivat saaneet raja-arvoja suuremmat arvosanat kaikista pääominaisuuksista, olisi voittaja ollut se vaihtoehto, jonka peittämä alue olisi ollut suurin.

Tästä voimme päätellä, että joustava ja pitkänäköinen, molempia osapuolia tasapuolisesti kunnioittava ja kuormittava näkökanta lisenssiehtoihin ja hinnoitteluun on paras ratkaisu.

## 11. Johtopäätökset

Tapaustutkimuksen myyjäyrityksen osalta ohjelmiston tuotteistusprosessi on vielä kesken, eikä lisenssin valinta ole siksi vielä ajankohtainen silloin, kun tämä tutkielma valmistuu. Vertailun tulokset onkin esitelty sen mukaan, mikä olisi suositeltavin ja pitkällä aikavälillä edullisin toimintamalli.

Vertailuun oli tarkoituksella valittu vaihtoehdoksi kaksi toisistaan täysin poikkeavaa näkökulmaa eli vaihtoehdot A ja B. Tämä siksi, että voitaisiin selkeästi todentaa, kuinka äärimmäiset vaihtoehdot eivät pitkällä aikavälillä ole yhtä suotuisia toimittajan eivätkä asiakkaan kannalta kuin tasapainoinen ja järkevä keskitie.

Vaihtoehdot A ja B saivat joistakin vertailussa käytetyistä mitatuista ominaisuuksista varsin hyviä tuloksia, mutta se, missä ne loistivat jonkin ominaisuuden osalta, kostautui jonkin toisen ominaisuuden kohdalla. Tämä heikkous johti siihen, että kumpikaan ääripään vaihtoehto ei selvinnyt vertailun loppuun asti. Johtopäätös on siis, että yletön voitontavoittelu ja toisen osapuolen oikeuksien sortaminen ei ole yhtään sen parempi ratkaisu kuin kaiken lupaaminen maan ja taivaan väliltäkään.

Jos vertailussa olisi ollut mukana vain vaihtoehdot A ja B, niin näistä kahdesta B olisi ollut lähempänä haluttuja kriteereitä. B sai arvosanoille yltävän tuloksen useammista vertailtavista ominaisuuksista kuin A. Myös B:n saamat tulokset olivat useilta osin parempia kuin A:n vastaavat arvosanat. Jos raja-arvoja olisi helpotettu ja asteikkoa lievennetty, niin vaihtoehto B olisi voinut saada hyväksyttävät tulokset kaikista ominaisuuksista ja näin päästä mukaan lopulliseen vertailuun. Jos tällöin mukana olisivat olleet vain vaihtoehdot A ja B, niin B olisi voittanut vertailun. Tällä kertaa kuitenkin vaihtoehto C oli niin ylivoimainen, että asteikon ja raja-arvojen muuttamiseen ei ollut tarvetta ja C tuli valituksi voittajaksi.

Tämän vertailun lopputulos ja suositus onkin, että kannattaa suosia sellaisia lisenssiehtoja, jotka tasaavat vastuuta molemmille osapuolille. Toimittajan kannattaa rehellisesti miettiä, millaisia palveluita sillä on mahdollisuus tarjota vaarantamatta omaa talouttaan. Myös hinnoittelun tulee olla järkevää ja perustua ennalta sovittuihin kriteereihin, jolloin toiminnan suunnitteleminen ja töiden resursoiminen onnistuu parhaiten molemmilta osapuolilta.

CFD-menetelmä osoittautui hyväksi valinnaksi tähän vertailututkimukseen, sillä se suosii niin yhden kuin myös useiden vaihtoehtojen tasapuolista ja luotettavaa vertailua. Tällaisen vertailun tekeminen perinteisin menetelmin olisi ollut hankalaa, sillä esimerkiksi tuotteista kuultavat myyntipuheet ovat usein varsin erilaisia keskenään ja myös eri tuotteiden kokeileminen ilman kunnon arviointimenetelmän apua antaa vain summittaisen käsityksen tuotteiden keskinäisestä paremmuudesta. CFD-menetelmää käyttämällä saatiin

suoraan keskenään vertailukelpoista informaatiota valituista kohteista, jolloin myös saadut tulokset olivat suoraan, helposti ja luotettavasti vertailtavissa keskenään.

CFD-menetelmä toimii erityisen hyvin tämän tutkielman pitkälle aikavälille sijoittuvan vertailun kaltaisessa tilanteessa myös siksi, että siinä voi ottaa huomioon muun muassa rahanarvon alenemisen. Tämä johtuu siitä, että saatuja mittaustuloksia ei käytetä suoraan, vaan ne muunnetaan arvosteluasteikon arvosanoiksi. Arvosanojen keskinäinen suhde vaikuttaa vertailun tulokseen, ei niinkään testauksessa saatu alkuperäinen arvo. Tällä tavalla saadut tulokset säilyttävät arvonsa pitkään, toisin kuin testauksesta saadut mittaustulokset.

## Viiteluettelo

- [Allemann, 2007] Allemann, Anders, Life and money beyond patents. *Intellectual Asset Management*, volume 24, 2007, ss. 9–12.
- [Bernin yleissopimus, 1986] Bernin yleissopimus kirjallisten ja taiteellisten teosten suojaamisesta, Pariisin sopimuskirja, valtiosopimus numero 79/1986.
- [Cusumano, 2007] Cusumano, Michael A., *The Changing Labyrinth of Software Pricing*. *Communications of the ACM*, volume 50, issue 7, 2007, ss. 19-22. 28
- [Georgiadou, 2003] Georgiadou, Elli, GEQUAMO– A Generic, Multilayered, Customisable, Software Quality Model. *Software Quality Journal*, volume 11, issue 4, 2003, ss. 313-323.
- [Hämäläinen, 2003] Hämäläinen, Pertti, Lisenssit hallintaan. *Tietokone*, nro 4b, 2003.
- [Hämäläinen, 2008] Hämäläinen, Pertti, Ohjelmalisenssit hallintaan. *Tietokone*, nro 2, 2008.
- [Ingram, 2006] Ingram, Rick, The Evolution of Software Licensing Models. Lisence Tracker, <http://www.oilit.com/papers/licenstracker.pdf> (haettu 7.5.2009), 2006, 8 sivua.
- [Kim, 2008] Kim, Nancy S., The Software Licensing Dilemma, *Brigham Young University Law Review*, volume 2008, issue 4, 2008, s. 1103-1164.
- [Korkiakangas, 1998] Korkiakangas, Jukka-Pekka, Lisenssin osto ja hankintapäätökseen vaikuttavat tekijät. Taloustieteen pro gradu -tutkielma, Jyväskylän yliopisto, Taloustieteellinen osasto, 1998, 85 sivua, 2 liitettä.
- [Lainkari, 2005] Lainkari, Jarmo, Teknologian lisensointi yritysten välillä; edut ja haitat. Tietojärjestelmätieteen kandidaatintutkielma, Jyväskylän yliopisto, Tietojenkäsittelytieteiden laitos, 2005, 25 sivua.

- [Mansala, 2001] Mansala, Marja-Leena, Immateriaalioikeuden lisensointi, teoksessa: Mylly, Tuomas (toim.), *Immateriaalioikeudet kansainvälisessä kaupassa*. KAUPPAKAARI Lakimiesliiton kustannus, Talentum Media Oy, Helsinki, 2001, ss. 1-29.
- [Oksanen, 2006] Oksanen, Ville, *Aktiivinen kuluttajaliike – yksi avain turvallisempaan verkkoympäristöön*, Asiantuntija-artikkeli, Liikenne- ja viestintäministeriön Luoti-ohjelma, nro 4, 2006, 5 sivua.
- [Pelkonen, 2003] Pelkonen, Antti, Lisensointi, mitä se on ja miten se näkyy Metso Oyj:ssä? *Kolster Info*, volume 2, 2003, ss. 2.
- [Persson & Steinby, 2006] Persson, Sabine G. & Steinby, Camilla, Networks in protected business context: Licenses as restraints and facilitators. *Industrial Marketing Management*, volume 35, issue 7, 2006, ss. 870-880.
- [Saastamoinen, 2006] Saastamoinen, Matti, *Avoimen lähdekoodin lisenssit kaupallisessa liiketoiminnassa*. Tietojenkäsittelyopin pro gradu –tutkielma, Tampereen yliopisto, Tietojenkäsittelytieteiden laitos, 2006, 95 sivua, 4 liitettä.
- [L404/1961] Tekijänoikeuslaki L404/1961
- [Tynan, 2004] Tynan, Dan, Software Licensing 101: Types and Terms. *InfoWorld*, volume 11, 2004, ss. 36-37.
- [UCC, 1971] Universal Copyright Convention (Yleismaailmallinen tekijänoikeussopimus), Tarkistettu Pariisissa 24.7.1971, valtiosopimus numero 25/1963.
- [Zhang & Seidmann, 2003] Zhang, Jie & Seidmann, Abraham, The Optimal Software Licensing Policy under Quality Uncertainty. *ACM International Conference Proceeding Series*; volume 50, ACM, Pennsylvania, 2003, ss. 276 – 286.