

AVOIN ELÄMÄ

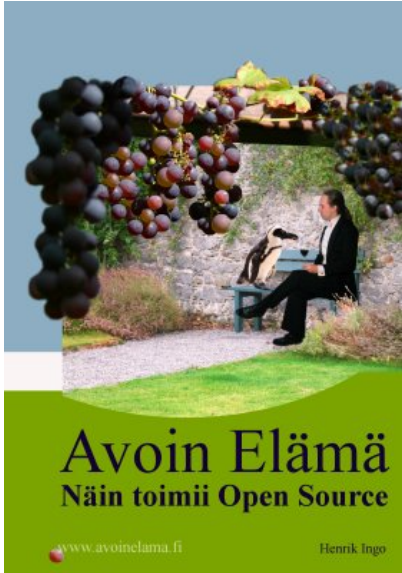
Published : 2012-06-01
License : GPLv2

AVOIN ELÄMÄ

1. AVOIN ELÄMÄ -KIRJASTA

1. AVOIN ELÄMÄ -KIRJASTA

Tämä on uusi kollaboratiivisesti muokattava versio [Henrik Ingon](#) kirjasta Avoin Elämä. Kirjan alkuperäinen versio löytyy osoitteesta <http://avoinelama.fi/>. Englanniksi kirjan voi lukea osoitteesta <http://openlife.cc/>.



LISENSSI

Teksti

(cc) Henrik Ingo 2005

Kirja "Avoin Elämä: Näin toimii Open Source" on vapautettu yhteisomistukseen, mikä tunnetaan englanniksi termillä Public Domain. Tämä tarkoittaa, että tekstiä voi kuka tahansa vapaasti kopioida, muokata ja julkaista edelleen. Tällaisen teoksen yhteyteen voi liittää Creative Commonsin "No Rights Reserved" -logon. Logoa klikkaamalla lukija päätyy Creative Commonsin sivuille osoitteeseen <http://creativecommons.org/licenses/publicdomain/>, jossa Public Domain -käsitettä selvitetään selvällä kielellä, jota ymmärtävät muutkin kuin lakimiehet.



Kansikuva

(cc) Henrik Ingo 2005

Kannen kuva taas on julkaistu Creative Commonsin "Attribution-ShareAlike" lisenssillä. Se tarkoittaa, että kuvaa saa vapaasti kopioida, muokata ja julkaista edelleen mutta ehtona on, että myös kaikki omat tuotokset, missä tällaista kuvaa on käytetty, tulee jakaa edelleen samoin ehdoin. "Attribution-ShareAlike" lisenssistä on samanlainen tiivis ja ymmärrettävä versio ja se on myös suomennettu "Nimi mainittava - Sama lisenssi" nimiseksi: <http://creativecommons.org/licenses/by-sa/2.0/deed.fi>.



Kannen kuva on kollaasi seuraavista kuvista:

- (cc) Osmo Penna 2004 (Frakkipukuinen mies)
- (cc) Adrian Pingstone 2004
(<http://en.wikipedia.org/wiki/Image:Penguin.jackass.arp.500pix.jpg>)
- (cc) Kees de Vos 2004
(<http://www.flickr.com/photos/devos/416019/>)
- (cc) Bernie Goldbach 2004
(<http://www.flickr.com/photos/19994833@N00/291527/>)

Epilogi

© Linus Torvalds 2004

Käännös: Henrik Ingo, 2005

Epilogi on alunperin julkaistu englanninkielisenä lokakuussa 2004, osana Linuxin dokumentaatiota. Sen kopiointi ja julkaiseminen on sallittu GPL lisenssin määräämin ehdoin, ks. <http://www.gnu.org/copyleft/gpl.html> (suomeksi http://www.turre.com/licenses/gpl_fi.php). Tämä kirja on kokonaisuudessaan saatavana elektronisessa muodossa ("lähdekoodi") osoitteessa <http://www.avoinelama.fi>. Epilogi on myös saatavana [Linuxin lähdekoodin](#) tiedostossa Documentation/ManagementStyle.

GPL-lisenssin ehdot ovat tässä tapauksessa käytännössä yhtenevät edellä selitetyn Creative Commons Attribution-ShareAlike lisenssin kanssa: Alkuperäiset tekijät on mainittava, "lähdekoodi" eli tässä tapauksessa alkuperäiset tekstitiedostot on oltava saatavilla ja myös kaikki uudet teokset on pakko julkaista GPL-lisenssin ehdoin.

ALKUSANAT

2. ALKUSANAT

2. ALKUSANAT

En ole koskaan jaksanut ymmärtää, miksi kirjoihin kirjoitetaan alkusanoja, joita kukaan ei kuitenkaan jaksa lukea. Turhaa lässytystä, jonka yleensä itse hyppään yli, että päästään itse asiaan. Kun aloin kirjoittamaan tätä kirjaa päätinkin, ettei siihen tulisi minkäänlaista johdantoa. Nyt kun kirja on lähdössä julkaistavaksi, tuntuu kuitenkin siltä, että jotain voisin kirjan syntyprosessistakin tässä kertoa. Mutta se ei kuitenkaan tarkoita, että kenenkään tarvitsisi tuhlata aikaansa sen lukemiseen. Joten suosittelen, että hyppäät suoraan lukuun yksi, niin minä tekisin, jos olisin sinä.

Idea tähän kirjaan syntyi vuoden 2003 alussa, työskennellessäni ohjelmointikouluttajana Tieturilla. Järjestimme seminaaria aiheesta "Linux ja Open Source" ja minun oli määrä puhua Open Source -kulttuurin erityisluonteesta. Tämä oli aikaa jolloin Turun kaupunki oli ilmoittanut siirtyvänsä Linuxiin ja ajattelimme, että sellaiselle seminaarille olisi kysyntää. Seminaari peruuntui vähäisen osanottajamäärän takia, mutta olin jo ehtinyt valmistella puhettani keräämällä ranskalaisia viivoja työkoneeni tekstitiedostoon "Open Sourcen filosofia.txt".

Vaikka seminaari peruuntuikin, tekstitiedosto jäi ja alkoi lopulta elää omaa elämäänsä. Ajan mittaan sinne kertyi lisää ranskalaisia viivoja. Minua kiehtoivat ja kiinnostivat Open Source -yhteisön kulttuuri ja toimintatavat, jotka poikkesivat perinteisestä ohjelmistotuotannosta ja oikeastaan mistä tahansa perinteisestä työkalukulttuurista.

Ei niin, etteikö Open Sourcesta olisi jo paljon kirjoitettukin. Lukemani kirjat kuitenkin edustavat yleensä jompaa kumpaa kahdesta leiristä. Jos kirja on jonkun Open Source -yhteisön sisäpiiriläisen kirjoittama, se harvemmin aukeaa tavalliselle kadunihmiselle, jota ei välttämättä kiehdo Linuxin historian yksityiskohdat tai eri ohjelmointikielten erikoisominaisuudet. Jos taas kirja on jonkun "ulkopuolisen" kirjoittama yleistajuinen teos, se yleensä kadottaa osan siitä rehellisen suorasta elämänasenteesta, tuiasta, mikä Open Source -yhteisön ihmisissä minua on aina kiehtonut. Halusin siksi tehdä kirjan, jossa yleistajuisesti käsitellään Open Sourcea laajemmin kuin "miten Linux toimii tietokoneessa" – pikemminkin otsikolla "miten Open Source toimii tietokoneen ulkopuolella" – mutta käyttämällä oikeita esimerkkejä ja tarinoita Linuxin ja Open Sourcen, vallankumouksen historiasta. Tapaustutkimuksen kaltaisia tarinoita siinä on yli 60, kattaen mielestäni aika hyvin ne pelurit, jotka ovat Linuxin ensimmäisen 13 vuoden aikana tavalla tai toisella ottaneet osaa vallankumoukseen. Luotan siihen, että poimimani esimerkit tulevat kiehtomaan lukijaa yhtä paljon kuin minuakin.

Kirjan kirjoittaminen varsinaisen työnsä ohella on iso urakka, johon en välttämättä aio ihan heti ryhtyä uudestaan. Ensimmäisen luvun valmistumisesta on kulunut jo puolitoista vuotta. Olen yrittänyt hokea itselleni Linuxin periaatetta: "Se on valmis sitten kun se on valmis". Täytyy tunnustaa, että hienojen periaatteiden lausuminen on hiempää kuin niiden toteuttaminen. Joskus on kärsimättömyys ollut lähellä viedä voiton.

Pitkäksi venynyt kirjoittamisprosessi oli myös haaste ajankohtaisen aiheen käsittelylle. Tänä aikana Linux, Openoffice, Firefox ja muut Open Source -ohjelmistot aloittivat lopullisen läpimurtonsa ja kehitys on tällä hetkellä uskomattoman nopeaa. Kun pääsin vihdoin kirjan viimeiselle sivulle, sain palata takaisin toteamaan, että työni aikana kaikkien tuntema, Red Hatin päätuote, Red Hat Linux oli lakannut olemasta, toiseksi suurin Linux-yhtiö SUSE joutunut Novellin ostamaksi, Mandrake päässyt selvitystilasta ja teki reippaasti voittoa. Nyt kun kirjoitan tätä johdantoa saan todeta, että juuri julkaistu Mozilla Firefox on aivan uskomattoman näppärä selain, joka lyö laudalta Internet Explorerin kaikissa kategorioissa. Vielä eilen Mozillaa haukuttiin tässä kirjassa "vähän hitaaksi".

Yksi motivaationi kirjan kirjoittamiseen oli markkinoida Open Source -ajattelua myös IT-maailman ulkopuolelle – tätä aihetta käsittelee neljäs luku. Nyt huomaan, että Wired-lehti on juuri julkaissut musiikki CD:n, jonka kaikki musiikki on Creative Commons -lisenssillä vapaasti kopioitavissa. Kuten neljännessä luvussa kerrotaan, tämä ei suinkaan ole ensimmäinen Open Source musiikkilevy, mutta nyt mukaan on jo saatu suhteellisen tunnettuja nimiä, kuten Beastie Boys ja David Byrne, omilta levy-yhtiöiltään saamallaan poikkeusluvulla. Tältäkin osin siis avoimuuden vallankumous edistyy.

Monet ystävät ovat kannustaneet ja auttaneet kirjoitusprosessissa. Haluan erityisesti mainita heistä yhden, Pertti Vehkavuoren. Pertti onkin rohkaisemisen ja kannustamisen mestari, jokaisella pitäisi olla vähintään yksi Pertti ystävänä. Kun Pertti oli lukenut kirjan ensimmäisen luvun, hän soitti ja puhui tunnin pihtaamisesta omassa työssään fysioterapeuttina ja kouluttajana. Luettuani toisen luvun hän soitti ja kysyi, onko kolmas luku kohta valmis. Puhelut fysioterapeutin kanssa, joka ei koskaan itse liene Linuxia käyttänyt, antoivat varmuuden, että kirjasta oli tulossa sitä mitä olin tavoitellutkin.

Tätä kirjaa ei olisi olemassa ilman rakasta vaimoani Sannaa. Monet ajatukset, mukaan lukien kirjan perustan luova ensimmäinen luku, ovat peräisin keskusteluista hänen kanssaan. Naisena hän teki selväksi, että olisi hyvin mielellään kirjailijan vaimo, mikä vasta sai insinöörin ottamaan ajatuksen kirjan kirjoittamisesta vakavasti. Eikä kirjaa olisi syntynyt ilman niitä lukemattomia sunnuntaipäiviä, joina hän on hoitanut tiskit, pyykkiä ja siivouksen, jotta saisin paneutua kirjoittamiseen. Voi kuinka rakastan sinua Sanna.

Espoon Matinkylässä, itsenäisyyspäivän aattona 2004
henrik ingo

LUKU 1

3. LUKU 1

4. RANSKALAISTEN KRIISI

5. KYSYNNÄN JA TARJONNAN LAKI

6. VIINIÄ VIEMÄRIIN

7. VIINILASISINFONIA

8. MAANVILJELYN MORAALI

9. PIHTAAMINEN JA EU:N

MAATALOUSTUET

10. AVOIMUUS

11. LINUX JA OPEN SOURCE

3. LUKU 1

jossa ajatus on kovin korkealentoista, eikä Linuxista vielä puhuta ollenkaan. Vai puhutaanko?

4. RANSKALAISTEN KRIISI

"Beaujolais in crisis. The secret is out!"
(Beaujolais kriisissä. Salaisuus julki!)

Näin otsikoi etusivullaan "The News",¹ englanninkielinen lehti Ranskassa asuville ulkomaalaisille ja muille Ranskasta kiinnostuneille. Olin ostanut lehden junalukemiseksi, sillä olimme vaimoni kanssa reppumatkalla Länsi-Euroopan halki. Olin ottanut tavakseni lukea International Herald Tribune lisäksi kunkin maan paikallisia lehtiä, mutta koska olin lukenut lukiossa fysiikkaa enkä ranskaa, jouduin nyt Ranskassa turvautumaan englanninkieliseen ranskalaiseen lehteen. Ja mikäs siinä.

Minkälainen kriisi oli nyt koitumassa Beaujolais-viinien kasvattajien tuhoksi? Oliko halla vienyt sadon vaiko varikset tai tappajaheinäsirkat olleet asialla? Ei sinnepäinkään.

Kriisi oli saanut alkunsa jo vuonna 2001. Beaujolais-sato oli silloin ollut *ennätysellisen hyvä*. Kyllä, se oli siis ollut *ennätysellisen hyvä*. Nyt oli kuitenkin katastrofi valmis, sillä vuoden 2002 sadosta oli tulossa *ihan yhtä hyvä!* Tämä oli onnistuttu jonkin aikaa pitämään vain viljelijöiden ja muun sisäpiirin tiedossa, mutta nyt salaisuus oli siis tullut julki ja perikato odotti viljelijäparkoja.

¹"Beaujolais in Crisis". The News, France's English Language Newspaper For Residents and Lovers of France. No 159/August 2002, sivu 1.

5. KYSYNNÄN JA TARJONNAN LAKI

Jokainen meistä on varmasti kuullut, jos ei muuten niin lukemalla Asterixia ja Obelixia,² että kysynnän ja tarjonnan laki on se, joka määrää hinnat markkinoilla ja yleisemminkin tuntuu pyörittävän talouden rattaita.

Kysynnän ja tarjonnan laki perustuu matemaatikko Daniel Bernoullin oivallukseen, että tuotteen hinnan ei niinkään tulisi määräytyä tuotteen valmistuskustannuksista, vaan siitä, paljonko hyötyä se tuottaa omistajalleen. Halpaa tuotetta kannattaa myydä kalliilla silloin, kun ostaja on valmis maksamaan siitä paljon. Suomessahan sanotaan, että ei se ole tyhmä joka pyytää, vaan se joka maksaa.

Toisaalta taas jos samaa tuotetta voi ostaa useammalta eri myyjältä, kannattaa ostajan tietenkin tehdä kauppaa sen myyjän kanssa, jolla on edullisimmat hinnat. Näin siis kysyntä nostaa hintaa ja samalla tarjonta laskee sitä ja jossakin löytyy sitten jonkinlainen tasapainotila, jossa kauppaa käydään.

Kysynnän ja tarjonnan laki voi johtaa tilanteisiin, jotka maalaisjärjellä tarkasteltuna saattavat vaikuttaa oudoilta. OPEC on järjestö, johon kuuluu suurin osa maailman öljyä tuottavista maista. Järjestön jäsenmaat sopivat keskenään siitä, kuinka paljon öljyä kukin milloinkin tuottaa. Jos uutisissa kuultaviin lyhyisin uutissähkeisiin on uskominen, järjestön kokouksissa tehdään lähinnä kahdenlaisia päätöksiä. Joko arabiherrat päättävät lisätä öljyntuotantoa tai sitten vähentää sitä.

Jos he nyt sitten jossain kokouksessa päättävät vähentää öljyntuotantoa, niin takuuvarmasti seuraavan päivän uutisissa kerrotaan siitä, että öljyn hinta maailmanmarkkinoilla on noussut. Tämä on itse asiassa niin normaalia, että se ei ehkä hätkähdytä sinua ollenkaan. Mietitäänpä kuitenkin seuraavaa. Kyseessä on täsmälleen sama öljy, jota vielä eilen myytiin huomattavasti halvemmalla. Se ei ole parempi- tai huonompilaatuista, sitä myydään samankokoisissa tynnyreissä, se on täsmälleen samaa öljyä, mutta kalliimpaa. Tämä on esimerkki kysynnän ja tarjonnan laista. Kun tarjonta laskee, hinta nousee vaikka kaikki muu pysyisi samana.

2ks. "Obelix ja kumpp."

6. VIINIÄ VIEMÄRIIN

Ranskalaisten Beaujolais-kasvattajien kannalta oli nyt päässyt käymään niin ikävästi, että tarjonta ylitti kysynnän. Viiniä oli liikaa. Jos salaisuus tulisi julki, niin hinnat saattaisivat pudota dramaattisesti. Ja kun salaisuus lopulta tuli julki, oltiin kriisissä.

Ranskalaiset eivät kuitenkaan jääneet neuvottomiksi. Ammattimaiset viinimaistelijat olivat jo aloittaneet viinien luokittelun kolmeen kategoriaan. Vain parhaimmat viinit pääsisivät myyntiin. Keskikastin viineistä tehtäisiin viinietikkaa ja huonoimmat viinit kaadettaisiin viemäristä alas. Viininviljelijät tunsivat kysynnän ja tarjonnan lain ja laskeskelivat, että on taloudellisesti kannattavampaa myydä vähemmän viiniä kalliimmalla kuin enemmän viiniä halvemmalla.

7. VIINILASISINFONIA

Pieninkin kyläkauppa, oli se sitten ranskalainen tai suomalainen, toimii kysynnän ja tarjonnan lain asettamilla ehdoilla, eikä siitä kirjoittaminen ole mikään järisyttävä uutinen.

Sen sijaan on järkyttävää, että kysynnän ja tarjonnan laki on salakavalasti ujuttautunut oman reviirinsä ulkopuolelle, alueille joille sen ei kuuluisi mennä. Kysynnän ja tarjonnan mekanismit on tiedostettu kohta kolme vuosisataa ja siitä on tullut niin olennainen osa ajatusmaailmaamme, että sen lonkerot ovat sekoittuneet elämämme sellaisillekin alueille, joilla ei ole mitään tekemistä kaupankäynnin kanssa. Koska niillä ei ole mitään tekemistä kaupankäynnin kanssa, saattaa tuloksena syntyvä kieroutunut ajatusmaailmamme tuottaa joskus hyvinvointia jälkeä. Ja mikä pahinta, koska kyse ei enää ole kaupankäynnistä, emme tiedosta kysynnän ja tarjonnan lakia silloin kun itse asiassa sovellamme sitä.

Kun ensimmäistä kertaa olin joulupäivällisellä vaimoni kotona, paljastui että appeni on sitä tyyppiä, jolla on aina jokin temppu takatuskussaan, millä viihdyttää kanssaihmissään. Juhlan kunniaksi oli pöytä katettu viinilaseilla, joista ei kylläkään juotu Beaujolais-viiniä. Jälkiruokaa odotellessa appeni kostutti etusormensa ja ryhtyi esittämään, miten viinilasilla soittaminen luonnistuu.

Jos olette koskaan olleet vastaanlaisessa tilanteessa, niin tiedätte varsin hyvin mitä seuraavaksi tapahtui. Kaikki muutkin alkoivat kokeilemaan viinilasilla soittamista. Muistaakseni minä olin viimeinen, joka onnistui taikomaan lasistaan kohtuullisen puhtaan vinkunan. Siinä sitä sitten yhdessä ujellettiin. Tämän jälkeen appi siirtyi illan bravuurinumeroon, joka oli haarukan ja veitsen tasapainottelu tulitikun nokassa viinilasilla. Jos ette ole kyseistä temppua koskaan juhlapäivällisillänne nähneet, niin tulkaa joskus tervehtimään meitä kun olemme anoppilassa Pietarsaareissa. Se on todella näkemisen arvoinen. Se näytti myös sen verran vaikealta, että meiltä muilta jäi se siltä erää opettelematta. Viinilasin vinguttamisen ja tulitikutemppujen jälkeen joululounaamme sitten jatkui entistäkin hilpeämmissä merkeissä.

Kävipä sitten niin, että uudenvuodenpäivänä oli vaimoni tehtävä järjestää polttarit kaasolleen, jonka kaasona hän vuorostaan oli. Polttareihin liittyi myös illallinen, joilla – näin olen kuullut, minuahan ei nimittäin kutsuttu – myös juotiin viinilaseista. Jossain vaiheessa illallista vaimoni innostui näyttämään muille viinilasilla soittamisen taitoa. Ja kuten arvata saattaa, myös näillä kutsuilla kaikki innostuivat vinguttamaan lasiaan.

Kun vaimoni jälkeenpäin kertoi tästä illasta, hän oli tässä kohden hieman vaivautunut. Kun rupesin utelemaan asiasta, hän lopulta paljasti, hyvin hiljaisella äänellä, että oikeastaan häntä vähän harmitti, että meni näyttämään tempun kaikille kavereilleen. Nimittäin nyt kun kaikki muutkin osasivat tempun, se ei enää tuntunutkaan kovin ihmeelliselle.

Kuinka tarkkanäköisen havainnon hän tulikaan tehneeksi! Tempu ei ole arvokas, jos kaikki osaavat sen. Siinä ei silloin olekaan mitään ihmeellistä. Näinhän me ajattelemme. Mutta toisaalta on täysin kestänytöntä väittää, että musiikki on kauniimpaa, jos sitä ei soiteta, kuin jos sitä soitetaan. Tai että vitsi on hausempi, jos sitä ei kerro kenellekään, kuin jos sen kertoo. Samaten ei voi olla niin, että viinilasilla soittaminen on hienompaa, jos sitä ei tee muiden nähden. Mutta näin me kylläkin useasti ajattelemme.

Tämä tarina viinilasilla soittamisesta on erinomainen esimerkki kysynnän ja tarjonnan lain ajatusmaailmasta silloin, kun se on eksynyt talouselämän ulkopuolelle. Se tunkeutuu arkiajatteluamme ja *käytökseemme* niin salakavalasti, että tyhmempi ei sitä usein huomaakaan. Enkä fiksumpikaan.

Me olemme sisäistäneet tämän ajatusmaailman jo lapsena. Se vaikuttaa meissä silloin kun kaksi lasta ilmoittaa kolmannelle: "Me ei nyt leikitä sun kaa!" Heidän kaveruutensa ikäänkuin vahvistuu siitä, että he *eivät leiki* muiden kanssa.

Tämän lisäksi "meillä" saattaa olla joku "salaisuus jota me ei kerrota sulle!". Lasten salaisuudet sinänsä saattavat koskea jotakin naurettavan merkityksetöntä asiaa, mutta se ei nyt olekaan tärkeää. Tärkeää on se, että sitä *ei kerrota* muille! Tämä on täysin sama toimintamalli, mikä aikuisilla jatkuu liikesalaisuuksien muodossa. Liikesalaisuudeksi määritellyt asiatkin saattavat useasti olla naurettavia ja merkityksettömiä, mutta tärkeää on kuitenkin olla paljastamatta niitä muille. Tämän päivän yritykset ovat vähintäänkin yhtä kiinnostuneita siitä mitä he eivät tee, kuin siitä mitä he ovat tekevinään ja tuottavinaan. Eivätkä näiden yritysten tekemiset aina ole lasten hiekkalaatikollekkejä korkeammalla tasolla. Tietoviikko-lehden kolumnisti kirjoitti kerran kaverinsa kokemuksista nyt jo konkurssiin menneessä uusmediayhtiössä. Siellä kuulemma salassapitosopimusten tekemiseen kulutettiin yleensä enemmän aikaa kuin varsinaiseen kaupantekoon. Näin jälkikäteen ajatellen kaveri oli siihen myös ihan tyytyväinen. Eihän niistä uusmediabuumin touhuista enää kehtaisikaan puhua julkisesti!

Näiden tarkastelujen valossa näyttää uhkaavasti siltä, että lähes kaikki mitä me teemme perustuu *pihtaamisen logiikalle*, niin talouselämässä kuin arjessammekin! Me hoidamme ihmissuhteitamme siinä missä arabit öljyntuotantoa. "OPEC ilmoitti eilisen kokouksen jälkeen, että se ei enää leiki sun kaa."

8. MAANVILJELYN MORAALI

Ranskalaisen Beaujolais-artikkelin pohjalta stereotyyppinen käsitykseni ranskalaisista ja etenkin ranskalaisista maanviljelijöistä vahvistui. He ovat yksiä moraalittomia hyypiöitä! Enkä nyt tarkoita sitä, että mielestäni hyvän alkoholin kaataminen viemäriin olisi moraalitonta. Ajatus josta kylläkin suuri osa suomalaisista varmaan olisi kanssani samaa mieltä.

Eriyisen silmiinpistävää artikkelissa nimittäin oli juuri se, että siinä ei mitenkään käsitelty sitä, onko hyvän viinin – enkä nyt siis tarkoita viiniä alkoholi juomana vaan pikemminkin elintarvikkeena – kaataminen viemäristä alas mahdollisesti, edes hieman, kyseenalaista.

Maanviljelijän tehtävä on tuottaa ruokaa. Ilman ruokaa kuolemme, emme siis pohdi nyt mitään täysin merkityksetöntä kysymystä. Ja kysymys pätee yleisemminkin, sillä yleisesti ottaen voidaan sanoa, että työnteon tarkoitus on tuottaa hyvinvointia. Ruokaa, terveyttä, turvallisuutta, viihdettä ja niin edelleen.

"Kyllä kyllä", saatat nyt ajatella, "mutta viini ei ole ruokaa. Viinien idea onkin olla harvinaista ja varsinkin kallista herkkua. Eihän niillä muuten voisi snobbailla! Ja snobbailu on viininjuonnissa olennaista. Eri asia olisi jos heittäisivät viemäriin maitoa ja perunoita."

Ehkä näin. Vaikka minä olenkin jo menettänyt uskoni ranskalaisiin maanviljelijöihin, niin saatat olla oikeassa. Eihän kukaan sentään oikeasti heittäisi pois ruokaa? Eihän?

Väitän, että heittäisivät. Väitän, että ihan niinkuin kaikki muukin toimintamme nykyään, myös maanviljelijöiden toiminta perustuu pihtaamisen logiikalle.

9. PIHTAAMINEN JA EU:N MAATALOUSTUET

Kuten kaikki tiedämme, koko Euroopan maatalous perustuu massiiviselle tukijärjestelmälle. Jostain syystä maatalous nyt vaan on niin kannattamatonta toimintaa, että maanviljelijöille on pakko maksaa tukea jos ylipäättään aiomme saada jotain suuhunpantavaa.

Tietääköhän kukaan, mitä kaikkia poliittisia syitä maataloustukien taustoihin liittyy? Vähäisin ei liene kansallinen turvallisuus. Koska me täällä Pohjolan perukoilla olemme säiden suhteen huomattavasti heikommassa asemassa kuin etelä-eurooppalaiset ystävämme, on suorastaan kysymys kansallisesta turvallisuudesta, että maanviljelijöitä tuetaan. Jokainen maa kokee, että on tärkeää säilyttää omassa maassa tietty omavaraisuustaso ruuan tuotannon suhteen. Ja koska näillä arktisilla leveysasteilla emme pysty kilpailemaan eteläeurooppalaisten maanviljelijöiden kanssa, maksamme tukea maanviljelijöillemme.

Edellä olevan minäkin ymmärrän. Sen sijaan en ymmärrä sitä, miksi myös ranskalaisille maksetaan tukia ja erityisesti en ymmärrä sitä, miksi heille maksetaan enemmän kuin pohjoisessa työskenteleville kollegoilleen!

No, minä en ymmärräkään EU:n maatalouspolitiikkaa, ei se mitään. Mutta tiedän kuitenkin vielä sen, että EU:ssa puhutaan paljon *kiintiöistä*. On maitokiintiöitä, munakiintiöitä ja viljakiintiöitä. Samalla kun maanviljelijä saa työnsä tietyn tuen, hän sitoutuu siihen että hän *ei ylitä kiintiötään*. Sillä jos maanviljelijät ylittäisivät kiintiönsä, meillä olisi enemmän ruokaa kuin EU:n kansalaiset jaksaisivat syödä ja olisimme kohta samanlaisissa kriisissä kuin Beaujolaisin viininviljelijät.

Sanalla sanoen, Euroopan maatalouspolitiikkakin perustuu *pihtaamiselle*. Tukipolitiikka perustuu siihen, että maanviljelijä sitoutuu olemaan tuottamatta ruokaa yli kiintiönsä. Jos otetaan huomioon, että suurin osa maapallon väestöstä näkee nälkää ja moni kuolee ruuan puutteeseen, tämä on mielestäni hyvinkin kyseenalaista politiikkaa.³ Mutta siitä ei kirjoiteta lehdissä, sillä meistä pihtaaminen on normaalia ja järkevää. Ironista kyllä, niin maataloudessa kuin talouselämässä, me puhumme usein tuottamisesta silloin kun oikeastaan on kyse pihtaamisesta.

3T ässä ei paneuduta siihen problematiikkaan, että myös ylituotannon dumppaaminen kehitysmaiin olisi ongelma, sillä se vie kannattavuuden heidän kotimaiselta maataloudeltaan. Kuitenkin myös tämä kysymys liittyy kysynnän ja tarjonnan lakiin.

Vuonna 2003 käydyissä USA:n, Euroopan ja monen kehitysmaan välisissä vapaan kaupan WTO-neuvotteluissa nähtiin sellainenkin ihme, että lääkeyhtiöt olivat valmiita tekemään myönnytyksiä patenteissaan, koska kehitysmaat edellyttivät sitä. Loppujen lopuksi sopimukseen ei kuitenkaan päästy, koska varsinkaan EU-maat eivät olleet valmiita luopumaan maataloustuistaan. Jostain syystä se vaan on niin vaikeaa.

10. AVOIMUUS

Kysynnän ja tarjonnan laista seuraava pihtaamisen logiikka tulee kaikista räikeimmin vastaan informaatioteknologian alalla. Se tulee vastaan kaikilla informaatioaloilla, myös siis musiikin, elokuvan ja muun viihteen yhteydessä, mutta kaikkein räikeimmin juuri tietokonealalla.

Informaatioalalla on mielenkiintoista se, että "tuotteen" luovuttaminen ei ole koskaan pois sen antajalta. Digitaalista informaatiota voi kopioida loputtomasti. Einsteinin kuuluisa lause pätee: "Jos minä annan sinulle pennin, olet sinä pennin rikkaampi ja minä taas pennin köyhempi. Mutta jos annan sinulle idean, on sinulla uusi idea, mutta minulla on edelleen omani."

Valitettavasti Einsteinin periaate ei ole ollut IT -alan kantavana ajatuksena. Sen sijaan, että digitaalista informaatiota käsiteltäisiin tavalla, joka olisi sille kaikkein luonnollisinta, sitä käsitellään ikään kuin se olisi ehtyvä luonnonvara. Sitä myydään pahvilaatikoissa siinä missä öljyä myydään tynnyreissä.

Tyypillinen tietokoneohjelma tänä päivänä maksaa – sanotaanko nyt vaikka – vajaat 1000 euroa. Tietenkin kalliimpiakin on, suuret yritysjärjestelmät voivat maksaa jopa 50 000 euroa, mutta on tonninkin paljon rahaa näin euroaikana. Tietokoneohjelma ostetaan yleensä fyysisen CD-levyn muodossa, joka lisäksi voi olla paketoitu kauniiseen pahvilaatikkoon. Yhden CD-levyn valmistuskustannus kansineen on noin 1€. Pahvilaatikko sen ympärillä ei maksa sitäkään. Lisäksi tuotteeseen liittyy tietenkin kuljetus- ja varastointikustannuksia, ja pitäähän myyjänkin joku provisio saada. Mutta kuitenkin ostaessamme tietokoneohjelman maksamme 99%:sti tyhjästä.

Tietenkään me emme ostaneet mitä tahansa CD-levyä, vaan nimenomaan sen, jossa oli joku tietty tietokoneohjelma. On selvää, että tietokoneohjelmat eivät synny itsestään, vaan myös ohjelmoijille pitää maksaa palkkaa. Mutta siitä huolimatta IT -alalla vallitseva kaupanteon mekaniikka on digitaalisen aineiston luonteelle epäluonnollinen. Ja tästä onkin seurannut tiettyä jännitettä.

Tämä jännite ilmenee muun muassa *piratismina*. Piratismi tarkoittaa sitä, että tietokoneohjelma (tai muu informaatioalan tuote) kopioidaan, vaikka ei saisi. Kopiointi kun ei maksa mitään, eikä ole keneltäkään pois. Tämä siis pitää paikkansa siinä mielessä, että kopioinnin tuloksena ei kenelläkään toisella ole aiempaa vähemmän tietokoneohjelmia. Piratismi kuitenkin on laitonta ja piratismiin kitkemiseen onkin sekä IT -alalla että viihdealalla kulutettu paljon verta, hikeä ja kyyneleitä. Melkein yhtä paljon kuin amerikkalaisten ja arabien taistelussa Irakin öljystä.

Kuitenkin juuri IT -alalla meillä on paljon opittavaa myös pihtaamiskierteen rikkomisessa. Yhä kasvava joukko ohjelmoijia tekee ohjelmiaan *Open Source* -periaatteen pohjalta. *Open Source* -liikkeessä tietokoneohjelmien kopiointia ei kielletä. Itse asiassa siihen pikemminkin kannustetaan. Ohjelmien toimintaperiaatteita ei salailla, vaan nimensä mukaisesti ohjelmien koodi on *Open Sourcessa* avointa.

Open Source -ohjelmistojen olemassaolo ei sinänsä ole ihmeellistä. Internetin myötä ohjelmiston levityskustannuksia ei ole, joten olisi ihmeellistä jos internetissä ei levitettäisi ilmaisohjelmia. Sen sijaan Open Sourcessa on ihmeellistä se, että viime vuosina on yhä selkeämmin näyttänyt siltä, että Open Source -ohjelmat ovat monessa suhteessa parempia kuin vastaavat suljetut ohjelmat. Näyttäisi siltä, että toimimalla digitaalisen informaation luonteen myötäisesti, sen sijaan että tekisi kaikkensa tämän luonteen hillitsemiseksi, Open Source -yhteisö on päästänyt valloilleen suunnattoman voimavaran, joka on aiemmin jäänyt IT -alalla ymmärtämättä ja hyödyntämättä.

Selkeimmin tämä voima korostuu, jos esitämme kysymyksen: Jos kerran Open Source -ohjelmat, sellaiset kuin Linux, ovat parempia tai edes yhtä hyviä kuin vastaavat ei-avoimet ohjelmat, mistä ne sitten tulevat? Kuka ne on tehnyt ja kuka on työn rahoittaja?

Ensi näkemältä näyttää siltä, että Open Source -ohjelmistot syntyvät itsestään! Tämä ei tietenkään voi olla totta, eikä se olekaan. Mutta kun kysytään, kuka ne on tehnyt, niin vastaus on: Ei kukaan erityisesti! Esimerkiksi Linus Torvalds on työskennellyt Linuxinsa parissa jo yli kymmenen vuotta, mutta hän on kuitenkin kirjoittanut vain pienen osan Linux-kernelin koodista. Suurimman osan on kirjoittanut joku muu. Kuka muu? Lukuisat muut samanlaiset ihmiset kuin Linus. Jotkut tekevät sitä huvikseen – Linus on yksi heistä⁴ – toiset tekevät sitä työkseen. Ahaa, mikä yritys siis on heidän työnantajansa? Vastaus on: Lukuisat eri yritykset. Miten vain asiaa käännelläänkin, lopputulos on, että Linuxia ei ole suunnitellut kukaan, sitä ei ole rahoittanut kukaan erityinen taho ja voisi melkein sanoa että se vaan syntyi ihan itsestään. Linuxin takana on suuri joukko vapaaehtoisvoimin toimivia ohjelmoijia, jotka eivät kannata pihtaamista!

Open Source -yhteisö on kääntämässä IT-alan käytäntöjä pääläelleen. Mikä rohkaisevinta – näyttää siltä, että avoin koodi on todistamassa meille käytännössä, että pihtaaminen onkin huonompi vaihtoehto kuin avoimuus. Mutta tarkoittaako se myös sitä, että musiikki on sittenkin kauniimpaa esitettynä, ja että vitsit ovat hauskoja vain kerrottuina? Voisimmeko oppia tunnistamaan pihtaamisen logiikan muilla elämänalueilla, niin että meillä avoimen koodin lisäksi voisi olla avoin mieli ja avoin elämä?

4Itse asiassa juuri tätä kirjaa kirjoitettaessa Linus siirtyi työskentelemään Open Source Development Labs -nimiseen säätiöön, jossa hän ensimmäistä kertaa Linuxin kaksitoistavuotisen historian aikana tuli työskentelemään päätoimisesti Linuxinsa parissa.

11. LINUX JA OPEN SOURCE

Tämän kirjan tavoitteena ei ole todistaa, että Linux on parempi kuin Windows, tai että Open Source on ainoa oikea ratkaisu. Varsinkaan en yritä kiistää sitä tosiasiaa, että öljy on ehtyvä luonnonvara. Mutta tarkoitus olisi kuitenkin hieman haastaa vallalla olevaa ajattelua. Tarkoituksena on seuraavaksi tarkastella Linux-maailman periaatteita, arvoja ja toimintatapoja ja mahdollisesti ottaa niistä jotain opiksikin.

Tämän johdantona toimineen luvun loppuun meidän on kuitenkin otettava huomioon se tosiasia, että monet lukijoista eivät ole ehkä koskaan kuulleetkaan Linuxista tai Open Source -käsitteestä. Ennen kuin jatkamme eteenpäin, tarjoamme siksi tässä hyvin lyhyen ja ytimekkään tietoisuuden Open Source -ohjelmistojen taustoista.

Keskiverto tietokoneen käyttäjä vuonna 2003 uskoo – tavallaan aivan oikein – että ei ole koskaan käyttänyt Linuxia. Todennäköisimmin hän käyttää Microsoftin valmistamaa Windows-nimistä käyttöjärjestelmää ja lisäksi muita Microsoftin ja joidenkin muiden yhtiöiden valmistamia ohjelmia päivittäisiin askareihinsa. Välillisesti kuitenkin jokainen meistä on jo hyödyntänyt Open Source -yhteisön tuottamia ohjelmia. Tämäkin kirja, jota parhaillaan luet, on kirjoitettu Linux-tietokoneella *OpenOffice*-nimisellä tekstinkäsittelyohjelmalla.

Jos olet käyttänyt internetiä, olet aivan varmasti ollut yhteydessä verkkopalvelimeen, jossa on *Linux*-käyttöjärjestelmä. Yli 60% kaikista internetin *www*-sivuista välitetään verkkoon *Apache*-nimisen avoimen palvelinohjelman avulla. Samalla lailla suurin osa internetin sähköposteista välitetään vastaanottajalleen avoimen *Sendmail*-nimisen sähköpostipalvelinohjelman avulla. Nämä kaikki ovat tunnettuja Open Source -ohjelmistoja. Tietämättäsi käytät siis todennäköisesti Open Source -ohjelmia päivittäin!

Kuten edellä on selostettu, on ohjelmistoyrityksissä vallalla oleva käytäntö sellainen, että tietokoneohjelmien lähdekoodi, se teksti mitä ohjelmoija työkseen kirjoittaa, on salaista. Sitä ei yleensä näytetä kenellekään yrityksen ulkopuolella, vaan valmiit ohjelmat jaetaan ainoastaan konekielisinä, eli siinä muodossa missä tietokone niitä suorittaa. Konekielisen ohjelman lukeminen on ihmiselle käytännössä mahdotonta. Lisäksi myös konekielisen ohjelman käyttämistä on rajoitettu erilaisilla ehdoilla joihin ohjelman ostaja pakotetaan suostumaan. Tyypillinen ehto on esimerkiksi se, että yhdeltä CD-levyltä ohjelman saa asentaa vain yhdelle koneelle. Jos jollakin on kaksi tietokonetta, hänen pitää ostaa myös kaksi kopiota haluamastaan ohjelmasta, vaikka mitään teknistä syytä tähän ei olekaan.

Open Source -yhteisö tuottaa ohjelmia täysin erilaisella periaatteella. Ohjelman käyttämistä ei keinotekoisesti rajoiteta mitenkään. Lähdekoodi on kaikkien saatavilla, se julkaistaan esimerkiksi internetissä. Lähdekoodi ei ole ainoastaan luettavissa, vaan kenellä tahansa on myös oikeus käyttää toisten koodia omista ohjelmissaan.

Nämä yksinkertaiset periaatteet ovat viime vuosina voimakkaasti haastaneet perinteisen ohjelmistoteollisuuden. Vaikka perinteinen ohjelmistoteollisuus on vielä voimissaan, on alkanut näyttää siltä, että Linuxiin ja muihin Open Source -ohjelmiin pohjautuvat järjestelmät ovat monissa tapauksissa sekä halvempia että laadukkaampia. Viime vuosina monet isot tietojärjestelmäprojektit onkin jo toteutettu Linuxin varaan. Ja kuten sanottu, internet-palvelimissa Linuxin käyttö on pikemminkin sääntö kuin poikkeus.

Tämä kirja ei tämän enempää esitele Open Source -ohjelmistojen historiaa tai teknistä taustaa. Lopetan kuitenkin tämän luvun pariin kirja- ja linkkivinkkiin niille, joita aihe saattaisi kiinnostaa.

Linuxin tarina

Tuula Nikkanen. Suomen ATK-kustannus, 2000.

Kirja käy kattavasti ja ymmärrettävällä tavalla läpi Linuxin historian, mukaan lukien avoimen lähdekoodin periaatteen syntyminen.

The Cathedral and the Bazaar

Eric S. Raymond

Saatavana kirjana: O'Reilly, 2001

Verkossa: <http://catb.org/~esr/writings/cathedral-bazaar/>

Open Source -liikkeen merkkijulkaisu, joka ruotii yhteisön toimintakulttuuria ja dynamiikkaa ja oli aikanaan ratkaisevassa asemassa kun Open Source -ohjelmistojen liiketaloudellista uskottavuutta piti vielä erikseen puolustaa.

The GNU Project

Richard Stallman

Julkaistu kirjassa: Open Sources

Verkossa: <http://www.gnu.org/gnu/thegnuproject.html>

Free Software (Vapaiden ohjelmistojen) -liikkeen perustajan ja johtajan Richard Stallmanin katsaus liikkeen historiaan.

The Open Source definition

Bruce Perens

Verkossa: <http://www.opensource.org/docs/definition.php>

Kymmenen periaatetta, jotka Open Source -ohjelman lisenssiehtojen tulee täyttää.

Just for fun – Menestystarina

Linus Torvalds & David Diamond. Suomentanut Sara Torvalds. Schildts förlags AB, 2001.

Linus Torvaldsin omaelämäkerrallinen tarina, joka Linuxin historian lisäksi sisältää jopa Linuksen ajatuksia elämän tarkoituksesta.

Hakkerietiikka

Pekka Himanen. WSOY, 2001.

Suomalaisen tähtifilosofin eettinen näkökulma Open Source -yhteisön toimintaperiaatteisiin.

LUKU 2

12. LUKU 2

13. OPETUKSIA OPEN SOURCE -
MAAILMASTA

14. KUOLEMA DEADLINEILLE

15. TEKEMÄTTÖMÄT TYÖT

16. ÄLÄ SUUNNITTELE MITÄÄN

17. TEE SITÄ MITÄ HALUAT

18. LAISKUUS ON HYVE

19. VALISTUNUT DIKTAATTORI

20. SUVAITSEVAISUUS

21. MONIMUOTOISUUS

22. ROHKEUS JA UTELIAISUUS

23. NIMET JA IDENTITEETTI

24. MITÄ ON ETIIKKA?

12. LUKU 2

jossa Linus on diktaattori,
laisuus on hyve ja hakkerit taistelevat identiteetistään.

13. OPETUKSIA OPEN SOURCE - MAAILMASTA

Tämän kirjan tarkoituksena ei ole olla Linux-historiikki. Myöskään en aio kirjoittaa mitään teknistä selostusta siitä, miten Linux toimii. Koska olen insinööri, en myöskään aio kirjoittaa etiikasta niinkuin Himanen tekee.

Open Source -liike on ohjelmistoalalla menestyksekkäästi haastanut perinteisen ajattelutavan, jota edellä kutsuimme niinkin ikävällä sanalla kuin pihtaaminen. Mutta ohjelmistoala on vain pieni osa maailmaa. Henkilökohtaisesti uskon, että muilla liiketoiminnan aloilla olisi Open Source -liikkeestä paljon opittavaa. Ja yksityisinä henkilöinä meillä kaikilla on jotain opittavaa siitä, miten nämä ihmiset ovat onnistuneet haastamaan pihtaamisen kulttuurin. Suurin osa meistä ei ehkä edes huomaa elävänsä sellaisessa ajatusmaailmassa, jossa emme halua opettaa kaverillemme viinilasilla soittamista. Mutta vaikka huomaisimmekin, niin meillä ei ehkä kuitenkaan olisi mitään vaihtoehtoisia toimintamalleja.

Tämän kirjan tavoitteena on tutkia Open Source -yhteisön tapoihin, työntekoon ja liiketoimintaan liittyvää kulttuuria ja siihen liittyviä arvoja. Lähestymistapa on hyvin käytännönläheinen, paikoin jopa hieman tapaustutkimuksia muistuttava. Tässä luvussa tarkastelemme – opiksi ja opetuksiksi – joitakin Open Source -yhteisön työssään arvostamia periaatteita ja käytäntöjä. Aloitetaan aiheesta, josta on takuuvarmasti kaikille hyötyä: stressinhallintaa a la Linus Torvalds.

14. KUOLEMA DEADLINEILLE

Linus Torvalds on siis Linux-käyttöjärjestelmän luoja ja johtaa edelleen sen kehitystä. Kun Linus helmikuussa 1997 valmistui Helsingin yliopiston tietojenkäsittelytieteen laitokselta hän siirtyi Kalifornialaiseen Transmeta-yhtiöön töihin. Koska Linux alkoi samaan aikaan olla jo suhteellisen kypsä käyttöjärjestelmä, se sai vähitellen myös lehdistön huomiota. Linuxin siirtyminen Amerikan mantereelle teki hänestä amerikkalaiselle lehdistölle helpommin tavoitettavan ja niin ujoista nörtistä tuli yhtäkkiä koko Open Source -liikkeen kansikuvapoju.

Yksi yleisimmistä kysymyksistä siihen aikaan oli: "Milloin julkaistaan Linuxin seuraava versio?". Tähän Linuxella oli aina vakiovastaus: "Sitten kun se on valmis."

Yksi Open Source -ohjelmistokehityksen kantavia periaatteita jo ennen Linuxin mukaantuloa on ollut, että ohjelma julkaistaan sitten kun se on valmis. Mitään deadlineja ei tunneta, eivätkä tekijät suostu antamaan edes viitteellisiä lausuntoja mahdollisesta valmistumisajankohdasta.

Legendaarisessa The Cathedral and the Bazaar⁵ -kirjoituksessaan Eric Raymond ruotii tätä periaatetta. Hän tulee siihen tulokseen, että ohjelmistoprojektissa, kuten missä tahansa muussa projektissa, asetetaan tyypillisesti kolmenlaisia tavoitteita: 1) Valmiissa tuotteessa tulee olla tietyt ominaisuudet, 2) tuotteen tulee täyttää tietyt laatukriteerit ja 3) sen tulee valmistua tietyssä aikataulussa.

Näiden kolmen vaatimuksen – ominaisuuksien määrä, laatu ja aikataulu – välillä on tietty jännite. Jos esimerkiksi aikataulu on liian kireä, ei ehkä kaikkia ominaisuuksia ehditäkään toteuttamaan. Jos projektipäällikkö kuitenkin on tiukka ja pitää kiinni asetetuista tavoitteista sekä ominaisuuksien määrän että aikataulun suhteen, joutuvat työntekijät hosumaan ja laatu kärsii. Raymond tulee kuitenkin siihen tulokseen, että jos tietoisesti suostutaan luopumaan yhdestä tavoitteesta, voidaan kahden muun osalta yleensä päästä hyvin tuloksiin.

Open Source -yhteisön periaate deadlineja vastaan on tätä taustaa vasten järvevä valinta. Se on todennäköisesti yksi syy Open Source -ohjelmistojen menestyksen takana. Toisaalta se on räikeä vastakohta muussa yritysmaailmassa vallalla olevalle projektikulttuurille, jossa ei ainoastaan ole yhtä deadlinea, vaan myös monta väli-deadlinea. Joskus tuntuu jopa siltä, että suurin osa projektipäällikön tehtävistä liittyy aikataulujen suunnitteluun, aikataulujen seurantaan ja aikataulujen uudelleenarviointiin. Itse tehtävä työ onkin sitten toisarvoista.

Sain itse kokea tämän jokin aika sitten, kun osallistuin erääseen projektiin, jossa asiakkaana oli eräs suomalainen tietotekniikka-alan yritys.⁶ Ensimmäisissä palaverissa asiakas ei vielä tarkalleen edes tiennyt projektin sisältöä, vaan tarkempi vaatimusmäärittely (ominaisuuksien määrä) oli osa projektia. Heti alusta yksi asia kuitenkin tiedettiin varmasti: aikataulu. Ennen kuin edes tiedettiin mitä haluttiin tehdä, oltiin lyöty lukkoon deadline. Aikataululle oli myös "hyvin perustellut" syyt. Projekti oli pakko toteuttaa seuraavan kahden kuukauden sisään, sillä siihen käytettävät varat tulivat meneillään olevan vuosineljänneksen budjetista. En tiedä kertooko tämä jotain asioiden tärkeysjärjestyksestä kyseisen yrityksen sisällä, mutta minua se ainakin huvitti.

Projektin deadline lähestyessä huvitus olikin tiessään, kun myöhään sunnuntai-iltana viimeistelimme tuotettamme. Projektin päätöspalaveri saatiin kuin saatiinkin pidettyä aikataulun puitteissa, ja lasku lähetettiin vuosineljänneksen viimeisenä päivänä ja kaikki olivat tyytyväisiä. Tähänkin liittyi tosin jonkin verran itsepetosta, sillä parin kuukauden päähän sovittiin vielä yksi palaveri, jossa käytäisiin läpi kaikki asiat joita ei ehditty tehdä päätöspalaveriin mennessä, ja niitä oli kyllä aika paljon. Hyvä niin. Parempi tinkiä sitten vaikka omasta rehellisyydestään kuin laadusta. Projekti todettiin siis onnistuneeksi ja ajallaan valmistuneeksi.

Laatu ja ominaisuuksien määrä ovat yritysmaailmaan ja tuotantoon liittyviä asioita. Ehkäpä joku projektipäällikkö voi hyödyntää Eric Raymondin ajatuksia seuraavassa projektissaan. Mutta vielä tärkeämpää on ymmärtää deadlinein vaikutus meidän jokaisen omaan elämään. Deadline on monille sananmukaisesti kuolemanvakava asia. Hosumme kuin hullut ja teemme yötä myöten töitä, jotta ehtisimme saada työn valmiiksi aikataulussa. Kuitenkin hyvin usein tämä kuolemanvakava aikataulu on täysin hatusta vedetty tai – kuten omassa tapauksessani – jonkun täysin turhan byrokraattisen pikkuseikan seuraus. Tämähkö takia me olemme niin stressaantuneita?

Linus on pyrkinyt selvittämään omaa suhtautumistaan työntekoon periaatteella, joka tunnetaan Linuksen lakina. Linuksen laki vastaa kysymykseen, miksi ihminen tekee jotain? Ensinnäkin ihminen tekee jotain säilyäkseen hengissä. Toiseksi ihminen tekee asioita sosiaalisista syistä. Ja kolmanneksi ihminen tekee asioita hivin vuoksi. Ja nimenomaan siis tässä järjestyksessä.

Linuksen laki pätee siis esimerkiksi juuri työntekoon. Ihmiset tekevät työtä ensisijaisesti siksi, että saisivat palkkaroilla ruokaa ja katon päälle. Mutta jos tämä olisi ainoa tavoite, riittäisi nykyistä huomattavasti vähäisempikin työn määrä. Joku on sanonut, että jos kysymys olisi vain välttämättömän ravinnon hankkimisesta, riittäisi länsimaiselle ihmiselle tehdä pari tuntia työtä viikossa. Olemme siis tienanneet välttämättömimmän osan palkastamme jo maanantai-iltapäivään mennessä. Sitä paitsi Suomen kaltaisessa hyvinvointivaltiossa olisi todennäköisesti täysin mahdollista säilyä hengissä koko elämänsä⁷ tekemättä työtä ollenkaan. Kuitenkin ihmiset haluavat tehdä työtä. Miksi?

Ihmiset käyvät töissä sosiaalisista syistä ja hivin vuoksi. Parhaat kaverimme löydämme yleensä koulusta ja työpaikalta. Tämä on siis sosiaalinen syy. Ja yleensä ihmiset pyrkivät pääsemään sellaiseen työhön, josta he nauttivat. He tekevät siis työtään hivin vuoksi.

Mutta jos teemme työtä hivin vuoksi ja nauttiaksemme siitä, miksi sitten raadamme itsemme näännäksi teennäisten aikataulupaineiden alla? Miksi raadamme kuin kysymys olisi edelleen elämästä ja kuolemasta? Linusin mielestä mitään järkevää syytä ei ole. Työstä kuuluu nauttia, koska sen takiahan sitä tehdään. Deadlinet ovat haitaksi sekä työn tekijälle, että itse työlle.

Linux 2.2 julkaistiin lopulta 25 tammikuuta 1999. Tämän jälkeen alkoikin arvuuttelu siitä, milloin julkaistaan seuraava versio 2.4.⁸ Nyt ollaan jo päästy versioon 2.6. Sen julkaisua lehdistö jaksoi jo odottaa rauhassa, eikä kukaan enää kysynyt Linukselta siihen liittyviä päivämääriä.

Lehdistö on oppinut jotain. Ota sinäkin oppia Linuxista. Ota rennosti ja nauti siitä mitä teet.

5<http://catb.org/~esr/writings/cathedral-bazaar/>

6En valitettavasti voi tässä paljastaa yrityksen nimeä, enkä myöskään projektin sisältöä, sillä projektiin liittyi tietekin salassapitosopimus. Kuinkas muuten.

7Tietekin on totta, että kaikki ihmiset säilyvät hengissä koko elämänsä :-). Mutta tarkoitan siis, että on mahdollista saavuttaa eläkeikä tekemättä päivääkään työtä. Useimmat eivät kuitenkaan halua tätä tehdä.

8Linuxin versionumerot ovat aina parillisia. Parittomat ovat ohjelmoijille suunnattuja kehitysversioita.

15. TEKEMÄTTÖMÄT TYÖT

Toinen kysymys, jonka Linus ja monet muut Open Source -ohjelmoijat kuulevat usein on: "Miksi ohjelmassasi ei ole sitäjasitä ominaisuutta?". Tämän kysymyksen esittäjät eivät niinkään usein ole lehtimiehiä kuin esimerkiksi kyseisen ohjelman käyttäjiä, eli tavallaan tuotteen asiakkaita. Myös tähän kysymykseen on olemassa vakiovastaus: "Koska kukaan ei ole vielä tehnyt sitä."

Vastaus on päällisin puolin tarkasteltuna vähän töykeä. Varsinkin jos siihen vielä lisätään: "Ohjelman lähdekoodi on verkossa vapaasti saatavana. Jos tarvitset jotain ominaisuutta voit vapaasti tehdä sen itse." Ottaen huomioon, että kysymyksen esittäjä ei välttämättä edes osaa ohjelmoida, vastaus todellakin on vähän töykeä.

Vastaukseen sisältyy kuitenkin aimo annos tervettä itsesuojeluvaistoa. Suurin osa Open Source -ohjelmistoista on tehty vapaaehtoisvoimin. Esimerkiksi Linus aloitti Linuxin teon opiskellessaan käyttöjärjestelmien ohjelmointia. Se oli hänen harrastuksensa, ei mitään muuta. Kun Linuxista sitten tuli joten kuten toimiva versio, Linus edelleen kehitti sitä vain ja ainoastaan itseään varten, toteuttaen sellaisia ominaisuuksia, jotka kiinnostivat häntä itseään. Halusipa Linus sitä tai ei, Linuxista tuli kuitenkin suosittu. Muutkin kuin Linus itse innostuivat käyttämään hänen käyttöjärjestelmäänsä. Palautteen saaminen omasta työstään on varmasti ollut imartelevaa. Mutta palautteen mukana alkaa tulla myös erilaisia pyyntöjä: "Yksi juttu ei toimikaan minun koneellani, voisitko korjata sen?" tai "Olisi todella siistiä jos Linuxissa voisi tehdä myös..." Toisten miellyttämisen halu on inhimillinen taipumus. Varsinkin kun kyse on elämäntyöstämme, haluamme mielellämme, että ihmiset pitäisivät siitä. Mutta kun pyyntöjä tulee kohta tuhansia, niihin hukkuu.

Lisäksi täytyy ottaa huomioon, että maailmassa on valitettavasti paljon ihmisiä, joiden elämäntehtävänä on valittaa. Kun Linus on innoissaan tehnyt itselleen hienon käyttöjärjestelmän, kuuluu suosionosoitusten seasta myös: "Linuxissa ei ole sitä..." ja "Linux ei osaa tätä..." Nämä ovat niitä ihmisiä, jotka eivät ikinä ole tyytyväisiä mihinkään. Samaan kastiin kuuluvat myös ne propellipäät, jotka tulevat innoissaan esittelemään omaa ideaansa: "Olen ajatellut, että Linuxiin voisi lisätä sellaisen ominaisuuden..." Mutta jos heidän ehdottamansa ominaisuus ikinä lisättäisiin, he eivät ikinä tulisi käyttäneeksi sitä, sillä heillä olisi jo kymmenen uutta ideaa, joiden toteuttaminen olisi todella siistiä. Ohjelmoijaparka haluaisi mielellään miellyttää kaikkia näitä valopäitä, mutta jos hän sitä yrittää, hän kohta näantyy kaikkien pyyntöjen alle. Mikä pahempaa, liiallinen suosio voi sitä kautta koitua jopa hyvin alkaneen projektin kohtaloksi.

Tätä taustaa vasten meidän tulee ymmärtää, että ohjelmoijan töykeähkö vastaus onkin tulkittavissa kohteliaaksi eiksi. "Koska teen tätä ohjelmaa harrastusluontoisesti ja omaksi ilokseni, minulla ei valitettavasti ole aikaa toteuttaa tuollaista ominaisuutta, jota en itse tarvitse. Mielestäni ideasi on kuitenkin mainio, ja jos haluat voin neuvoa miten voit itse toteuttaa sen, sillä koodi on vapaasti verkossa saatavana. Yhdessä tekeminen on muutenkin kivaa."

Vastaukseen sisältyy itsesuojeluvaiston lisäksi toinenkin pieni totuuden siemen. Onhan nimittäin totta, että jos kysyjä todella tarvitsisi kysymäänsä ominaisuutta, hän voisi tehdä sen itse tai ainakin palkata jonkun, joka osaa tehdä sen. Kuten juuri totesimme, monet näistä kysyjistä ovat kyllä innoissaan hyvästä ideastaan, mutta kun asia esitetäänkin heille tällä lailla, innostus laantuu. He eivät oikeasti usko loistavaan ideaansa niin paljoa, että olisivat valmiit panostamaan ideaansa sekuntiakaan – tai penniäkään.

Hyvä ystäväni on ammatiltaan pastori. Voin kertoa, että pastorin työ on paljon monipuolisempaa kuin voisi kuvitella. Jumalanpalvelusten ja häiden lisäksi pitää järjestää kaikenlaista ohjelmaa niin nuorille kuin vanhuksillekin. Tämän lisäksi pitää ymmärtää äänentoiston päälle ja olla mikrotukihenkilönä. Ja paljon muuta mielenkiintoista projektia hänellä on aina meneillään. Onpa hän saarnannut myös internetissä suorassa lähetyksessä.

Puhuimme kerran siitä, miten hän suunnittelee omia töitään. Hän kertoi, että yleensä kesä- ja joululoman jälkeen hän kirjoittaa paperille listan asioista ja projekteista, joita pitäisi tehdä. Lista voi myös tarpeen vaatiessa myöhemmin lisätä rivejä, jos uusia hyviä ideoita syntyy. Sitten vaan valitaan listalta yksi tai useampi asia työn alle. Kun joku projekti saadaan päätökseen, voidaan sen yli vetää viiva.

Kun puoli vuotta on kulunut, on taas uuden listan vuoro. Moni vanhan listan projekti on saatu onnistuneesti päätökseen, mutta suurin osa on yleensä tekemättä. Kuten ohjelmoijillakin, myös pastoreilla tuntuu olevan enemmän ideoita kuin aikaa. Edelleen tekemättä olevat projektit voidaan ottaa mukaan uudelle listalle.

Joskus voi käydä niin, että joku loistoidea sitkeästi keikkuu vuodesta toiseen tekemättömien töiden listalla. Tällöin se poistetaan listalta, eikä muistella sitä sen enempää.

Tietämättään tämä ystäväni noudattaa töidensä priorisoinnissaan samaa periaatetta kuin Linus ja kumppanit. Heilläkin on lista ominaisuuksia, jotka he haluavat toteuttaa ohjelmassaan. Osa itse keksittyjä ja osa ehkä käyttäjiltä tulleita toivomuksia. Sitten kaikki tekevät mitä huvittaa.

Kun kaikki tekevät mitä huvittaa, niin usein tietenkin käykin niin, että joku tekemätön työ jää tekemättä pidemmäksi aikaa, kun ketään ei huvita tehdä juuri sitä. Linus ei ota tästä mitään paineita. Jos joku ominaisuus jää vuosiksi toteuttamatta, se ei voi olla kovin tärkeä, kun ilmankin on selvitty! Linuksen oppi tässä kohden siis on, että tärkeät työt valikoituvat automaattisesti tehtäväksi, eikä asiasta kannata sen vuoksi stressata.

Tänä päivänä Linux on miljardien dollarien bisnes. Pelissä mukana on sellaisia yhtiöitä kuin IBM ja HP. Jos jotain ominaisuutta oikeasti tarvitaan, he todella voivat tehdä sen itse – ja tekevätkin. Koodihan on vapaasti verkossa saatavana.

Vaikka sinulla ei olisi miljardeja dollareja, voit kuitenkin toteuttaa tätä Open Source -yhteisön periaatetta myös omassa elämässäsi. Kun seuraavan kerran pomosi tarjoaa kiireellistä uutta projektia, joka on todella tärkeä ja jonka pitäisi olla valmis heti, voit ajatella: *Jos tämä projekti on firmalle todella näin tärkeä, siihen voidaan varmasti palkata joku, jonka ei tarvitse tehdä sitä ylitöinä.*

16. ÄLÄ SUUNNITTELE MITÄÄN

Linuxin syntymäpäivä lasketaan viestistä, jonka Linus lähetti Usenet-järjestelmän keskusteluryhmään comp.os.minix, 25. elokuuta 1991. Huomaa, että mitään Linux-aiheisia keskusteluryhmiä ei luonnollisestikaan ollut olemassa!

```
From: Linus Benedict Torvalds (torvalds@klaava.Helsinki.FI)
Subject: Mitä ominaisuuksia haluaisit nähdä minixissä?
Newsgroups: comp.os.minix
Date: 1991-08-25 23:12:08 PST
```

Moi kaikki minixin käyttäjät -

Olen tekemässä (vapaata) käyttöjärjestelmää (se on vain harrastus, siitä ei tule isoa ja ammattimaista niinkuin gnu:sta) 386(486) AT klooneille. Olen keiteltyt tätä kokoon huhtikuusta asti ja se alkaa olla jo valmis. Haluaisin kuulla mistä asioista ihmiset pitävät/eivät pidä minixissä, sillä minun käyttöjärjestelmäni muistuttaa sitä jonkin verran (tiedostojärjestelmässä on sama fyysinen järjestys (johtuen käytännön syistä) ja pari muuta juttua).

Olen jo portannut bash(1.08) ja gcc(1.40), ja homma tuntuu toimivan. Tämä tarkoittaa, että jo parin kuukauden päästä minulla pitäisi olla jotain toimivaa ja minua kiinnostaisi kuulla mitä ominaisuuksia ihmiset haluaisivat. Kaikki ehdotukset ovat tervetulleita, mutta en lupaa mitään :-)

Linus (torvalds@kruuna.helsinki.fi)

PS. Kyllä – siinä ei ole yhtään minix-koodia, ja siinä on monisäikeinen tiedostojärjestelmä. Se EI ole siirrettävä (käyttää 386 prosessinvaihtoa jne), eikä se varmaan ikinä tule tukemaan muuta kuin AT-kovalevyjä, koska muita minulla ei ole :-(⁹

Viestissä on tiettyä yleveyttä ja historian siipien havinaa. Mutta näin kolmentoista vuoden perspektiivissä se on näiviudessaan myös huomittava. Linux on iso ja kehitystyö sen ympärillä ammattimaista. Päälle päätteeksi se on todennäköisesti historian siirrettävin käyttöjärjestelmä. Nykyään Linux tukee ainakin kuuttatoista eri prosessoriarkkitehtuuria, IBM:n s390 suurkoneista pieniin kulutuselektronikan prosessoreihin asti. Itse asiassa Linuxin koodin kehutaan olevan esimerkillisen modulaarista ja siksi helposti siirrettävää.

Ironiaa lisää se, että Linuksen mainitsema GNU-projekti ei ole vielä saanut valmiiksi omaa "isoa ja ammattimaista" Hurd-kerneliään. Kaikki kunnia kuitenkin heille siitä, että eivät ole vielä luovuttaneet. Olen itse seurannut Linuxin kehitystä läheltä yli puoli vuosikymmentä, ja koko sen ajan Hurd on ollut melkein valmis. On jotenkin huvittavaa nähdä, että tilanne oli sama jo kolmentoista vuotta sitten Linuxin syntyessä.

Käytämme yleensä ns. historian suurmiehiä esimerkkinä, kun yritämme itse löytää jonkinlaista menestystä omassa elämässämme. On jopa olemassa ihan oma psykologeista, filosofoista, tanssijoista ja ties mistä tyypeistä koostuva värikäs ammattikunta, joka ravaa yritystilaisuuksissa luennoimassa menestymisestä ja hakee inspiraatiota ja esimerkkiä muun muassa tällaisista lähteistä. Yleensä näissä luennoissa puhutaan suurista visioista, määrätietoisuudesta ja kovasta työnteosta.

Linus – kiistämätön suurmielinen tietotekniikan historiassa – on kuitenkin näiden hengennostatusluentojen sisältöön nähden hyvin poikkeava esimerkki. Näyttäisi siltä, että Linuxin kehittymistä ovat ohjanneet pikemminkin vision puute ja päämäärättömyys, tai ehkä kutsuisimme sitä mieluummin *avoimuudeksi*. Avoimuudeksi toisten ihmisten ehdotuksille. Avoimuudeksi sille, että suunnitelmat muuttuvat.

On mielestäni huvittavaa kuvitella, miten olisi käynyt, jos Linus olisi ollut esimerkiksi "määrätietoinen". Esimerkiksi kun joku Open Source -periaatteen mukaisesti ensimmäisen kerran lähetti Linusille koodia, jonka avulla Linuxia voi käyttää myös SCSI-tyyppisten kovalevyjen kanssa, hän olisi voinut vastata: "Kiitos, mutta ei kiinnosta. Tämähän on vain minun pieni ja epäammattimainen harrastukseni ja kuten kirjoitin, minulla ei ole SCSI-levyjä, joten en tarvitse tätä koodia. Sitä paitsi minulla on nyt vähän kiire, sillä teen kovasti töitä."

Vaikka ajatus naurattaakin, niin olen varma, että maailma on itse asiassa täynnä juuri vastaavanlaisia tarinoita. Tarinoita ihmisistä, jotka sanovat "kiitos, ei kiinnosta" elämänsä tilaisuudelle. Ja mahdollisesti juuri siksi, että joku luennoitsija on käskenyt heitä olemaan määrätietoinen ja tekemään kovasti töitä. Näitä tarinoita ei vain ikinä kuule, sillä näistä ihmisistä ei koskaan tule historian suuria miehiä tai naisia.

Mikä sitten on Linuxin suuruuden salaisuus? Mielestäni Linuxin tarinasta voimme löytää ainakin seuraavat tärkeät periaatteet:

- Linus tekee jotain mistä on innoissaan.
- Sen lisäksi, että Linus on kiinnostunut käyttöjärjestelmäohjelmoinnista itsessään, Linux on alusta asti vastannut myös konkreettiseen tarpeeseen: 1991 PC-koneiden käyttöjärjestelmät olivat joko surkeita (dos) tai kalliita (unix) ja siksi Linus päätti tehdä itse paremmin omia tarpeitaan vastaavan käyttöjärjestelmän.
- Voidaan myös sanoa, että Linus oli oikeassa paikassa oikeaan aikaan. Hän on itsekin todennut, että jos hän ei olisi tehnyt Linuxia, joku muu olisi.
- Linus oli kuitenkin se henkilö, joka oli oikeaan aikaan avoin tälle mahdollisuudelle silloin kun se tuli hänen eteensä, vaikka se ei kuulunutkaan hänen suunnitelmiinsa.
- Hän ei myöskään jäänyt työnsä kanssa omaan koppiinsa piilottelemaan, vaan jakoi työnsä tuloksia avoimesti ystävilleen.
- Ja vielä yksi tärkeä kohta: Linus ei toiminut yksin. Hän oli avoin muiden tarjoamille ideoille ja yhteistyölle. Sen lisäksi hän perusti työnsä muiden aikaisemmin tekemälle työlle (minix) ja käytti muiden tekemiä työkaluja (bash, gcc).

Tietenkin listaan voitaisiin lisätä muita, innostuskonsulttienkin hehkuttamia ominaisuuksia. On selvää, että Linus on lahjakkaimpia alallaan. Hän on myös tehnyt Linuxin parissa kovasti töitä, mutta olisi väärin kutsua tätä "kovaksi työkse". Pikemminkin on kysymys *innostuneesta työstä*, mikä on täysin eri asia. On siis väärin neuvota ketään tekemään kovasti töitä, pitäisi neuvota ihmisiä tekemään sitä mistä ovat innostuneita.

Tiivistäen tämän historian suurhenkilön opetus olisi siis: Älä suunnittele mitään, sillä et voi tietää mitä elämä tuo tullessaan.

Olisi epäilemättä lopettaa tämä kappale tähän, ilman pientä huomionosoitusta GNU-projektia kohtaan. GNU-projekti on Richard Stallmanin 1984 aloittama projekti, ja hän on se henkilö, joka ensimmäisenä pyhitti elämänsä avoimien ohjelmistojen tekemiselle. Vaikka Hurd-kerneli jäikin Linuxin jalkoihin, ovat lähes kaikki Linux-käyttäjärjestelmän perustyökalut peräisin GNU-projektista, mukaan lukien bash ja gcc, jotka Linuskin maintsi Linuxin syntymäviestissä. Vaikka siis hieman naurammekin Hurd-kernelille (johon Linus viittaa vain epäsuorasti), on reilua todeta, että Open Source -liikkeen menestystarinan takana on myös paljon GNU-liikkeen tekemää työtä. Ehkä onkin sopiva lopettaa tämä kappale Richard Stallmanin sanoihin, jossa hän kommentoi väitettä, että suurin osa Open Source -ohjelmista ovat syntyneet (vain) tekijän henkilökohtaisen kiinnostuksen tyydyttämiseksi: "Ehkä näin tapahtuukin joskus, mutta monet keskeiset GNU-ohjelmat on tehty nimenomaan vapaan käyttäjärjestelmän toteuttamiseksi. Ne eivät syntyneet hetken mielijohteesta, vaan niiden takana on visio ja suunnitelma."

9Alkuperäisen englanninkielisen version voit lukea internetistä osoitteesta: <http://groups.google.com/groups?selm=1991Aug25.205708.9541%40klaava.Helsinki.FI>

17. TEE SITÄ MITÄ HALUAT

Nyt olemmekin päässeet tässä inspiroivassa Linus-analyysissämme niin hyvään vauhtiin, että olemme edellisessä kappaleessa jo melkein tulleet käsitelleeksi tämänkin kappaleen. Ei se mitään, innostuneisuus on hieno asia!

Open Source -liikkeen ideologiaan, jota esimerkiksi Pekka Himanen on kutsunut *hakkerietiikaksi*, kuuluu olennaisesti ohje: Tee sitä mitä haluat.

Open Source -ohjelmia on tähän saakka tehty paljolti vapaaehtoistyönä. Suomeksi voisimme sanoa että *talkoovoimin*. Vaikka tilanne onkin nyt muuttumassa ja mukana on yhä enemmän yrityksiä ja sitä kautta palkattua työvoimaa, elää tämä perinne kuitenkin edelleen vahvana. Vapaaehtoistyövoimasta seuraa yleensä suoraan, että mukana oleva porukka on innostunut. Ketään ei ole pakotettu mukaan ja kukapa haluaisikaan vapaaehtoisesti tehdä jotain, mistä ei ole kiinnostunut. Tästä seuraa valtava innostuneisuuden ja kiinnostuneisuuden ilmapiiri. Tämän ilmapiirin onkin sanottu olevan toinen tärkeä syy Open Source -ohjelmien korkeaan laatuun. Kun ihminen tekee kahdeksasta-neljään periaatteella jotain tehtävää, johon hänet on käsketty, on jälkikin erilaista kuin jos ihminen innostuneesti tekee jotain, mistä todella nauttii ja minkä ehkä kokee myös elämäntyökseen.

Tee mitä haluat -periaatteessa on kuitenkin kysymys enemmän kuin vain jostain talkoohengen seurauksista. Ainakin Linus itse on sisäistänyt tämän periaatteen paljon syvemmin ja osannut pukea sen sanoiksi. Sen lisäksi, että lehdistö alkoi 90-luvun lopulla kiinnostumaan Linuxista, samoin siitä kiinnostuivat lukuisat ohjelmoijat. Lukuisat tietotekniikan opiskelijat ja myös työelämässä olevat koodarit halusivat tutustua Linuxiin ja myös osallistua sen kehitykseen. Joillekin se tarjosi mielenkiintoisen haasteen ja toiset kiinnostuivat varmaan vain siksi, että se oli niin siistiä, eli coolia.

Tällaisen muotivillityksen seurauksena myös Linus sai henkilökohtaisia yhteydenottoja ihmisiltä, jotka halusivat tulla mukaan Linuxin kehitykseen ja kysyivät Linukselta jonkinlaista neuvoa siitä, mitä heidän kannattaisi ruveta tekemään. Taas kerran Linuxen vastaus oli kuiva ja mitään sanomaton: Sinun pitää itse miettiä mikä sinua kiinnostaa, ja mennä siihen mukaan.

Linuxella oli vastaukseensa myös selkeät motiivit. Hän selitti asian niin, että jos hän mielivaltaisesti neuvoisi ihmisiä menemään mukaan johonkin projektiin, ehkä johonkin sellaiseen joka häntä itseään sillä hetkellä kiehtoi, niin hänen neuvonsa varmastikin osoittautuisivat huonoiksi. Eihän kysyjää todennäköisesti kiinnosta samat asiat kuin häntä itseään. Samaten kysyjän lahjakkuus ei ehkä ole samalla alueella kuin Linuxen. Jos kysyjä sitten tekisi niinkuin hän neuvoo, kysyjä ennen pitkää turhautuisi koko projektiin ja pahimmassa tapauksessa olisi sen jälkeen vihainen sekä Linusille että Linuxille. Sitä paitsi, eipä uudesta vapaaehtoistyövoimasta olisi ehdotetulle projektillekaan paljoa hyötyä, jos tekijä ei ole innostuneena mukana.

Yritysmailmalle tämä periaate on taas kerran haaste. Ja samoin se on haaste Linuxille, nyt kun yritysmailma on tulossa mukaan Linuxiin. Kuinka moni työelämässä tekee juuri sitä mitä haluaa? Kuinka moni tekee sitä mitä pomo käskee? Onko tällaista periaatetta edes mahdollista toteuttaa yritysmailmassa?

Jos uskomme, että "tee mitä haluat" on yksi Open Source -yhteisön kulmakiviä, meidän on loogista uskoa myös, että sama periaate voi tuottaa yritykselle samanlaista menestystä kuin se on tuottanut Linuxille.

Yleensä yrityksen visiot ja strategiat sorvataan kokoon jossakin johtoportaassa, jonka jälkeen työntekijän tehtäväksi jää "tehdä mitä pomo haluaa". Joissakin yrityksissä tämä perinne on vahvempi, joissakin heikompi, mutta yleensä hyväksymme, että näin kuuluu olla. Kuitenkin "tee mitä haluat" periaatteen mukaan yrityksen johdon pitäisi kokonaan lopettaa visioiden ja strategioiden tuottaminen ("älä suunnittele mitään" -periaatetta noudattaen) ja sen sijaan keskittyä siihen, että työntekijät voivat mahdollisimman hyvin toteuttaa omaa visiotansa. Monelle johtajalle tämä on varmasti täysin vieras ajatus.

Niille meistä, jotka emme ole yritysmaailman johtajia, jää jäljelle kysymys: Teetkö juuri nyt sitä mitä todella haluat? Kun valitsit nykyisen työpaikkasi, valitsitko sen missä oli korkein palkka vai valitsitko sen mikä kiinnosti sinua kaikkein eniten? Tämä kysymys ei ole olennainen vain sinulle oman mielenterveytesi kannalta, vaan sen pitäisi kiinnostaa myös työnantajaasi. Miksi yritysmaailman pitäisi tyytyä kahdeksasta-neljään suorittajiin, kun Linux-maailmassa asiat ovat paremmin?

Noudattavatko Linus ja kumppanit sitten itse omaa esimerkkiään? Kyllä noudattavat. Kaikki tärkeimmät Linux-ohjelmoijat ovat nykyään töissä yrityksissä, joissa heidän pääasiallinen työtehtävänsä on Linuxin ohjelmointi. Harrastuksesta ja elämäntehtävästä on tullut myös päivätyö. Voisiko parempaa tilannetta enää kuvitella? Älä sinäkään tyydy vähempään!

18. LAISKUUS ON HYVE

Saatat ehkä yllättyä, kun kerron, että ohjelmoijan tärkein hyve on laiskuus. Linux-yhteisössäkin tämä on jopa innostuneisuutta tärkeämpi ja tunnustetumpi hyve. No, jos olet naimisissa ohjelmoijan kanssa, olet ehkä todennutkin, että laiskoja ronttejahan nuo ovat, ei se ikinä tiskaa eikä pese pyykkiä, eikä aina muista syödäkään. Mutta monille saattaa olla vaikea käsittää miten esimerkiksi Linusta voisi väittää laiskaksi – poikaa joka sulkeutui huoneeseensa vuodeksi ohjelmoimaan yötä myöten Linuxin ensimmäistä versiota.

Väitteen logiikka menee näin: Mitä laiskempi ohjelmoija, sitä enemmän hän ohjelmoi. Kun ohjelmoija kohtaa tylsän, aikaa vievän ja paljon rutiineja sisältävän tehtävän, häntä alkaa laiskottaa. Laiska ohjelmoija ei mitään pahempaa tiedäkään, kuin tylsät rutiinitehtävät. Sen takia hän päättääkin tehdä tietokoneohjelman, joka voisi tehdä rutiinitehtävät hänen puolestaan. Tätä tietokoneohjelmaa hän sitten tekee yötä myöten, näennäisen ahkerasti. Hän ohjelmoi tekstinkäsittelyohjelman, koska kirjoituskoneella kirjoittaminen on niin vaivalloista. Ja koska kirjeen tulostaminen ja postiin vieminen, postimerkin nuolemisesta nyt puhumattakaan, on niin vaivalloista, hän ohjelmoi sähköpostin. Näetkö nyt, että ohjelmoijat tosiaan ovat laiskoja?

Laiskuus on siis ohjelmoijan hyve. Tämä ristiriitaisuudessaan vähän huvittavakin väite on hyvä muistaa. Usein nimittäin ihmiset saattavat unohtaa, minkä takia tietokoneet ja niiden ohjelmat – ja kaikki muukin teknologia – on alun perin keksitty.

Suomessa rakennetaan parhaillaan uutta ydinvoimalaa. Ydinvoimalan rakentamispäätöstä edelsi vilkas keskustelu, jossa kantaa otettiin sekä puolesta että vastaan. Eräs keskusteluun osallistuneista kansanedustajista oli sitä mieltä, että ydinvoimalan rakentaminen on Suomelle huono vaihtoehto, koska muut energiantuotantomuodot työllistävät paremmin. Haluan nyt korostaa, että mielestäni on monia hyviä syitä, miksi maailmaan ei lisää ydinvoimaa tarvitsisi rakentaa. Mutta tämä oli kyllä naurettavin perustelu, mitä olen koskaan kuullut missään väittelyssä käytetyn ja siksi kyllä pidätän itselläni oikeuden nauraa sille päin naamaa.¹⁰

Onhan tietenkin totta, että sähköä voitaisiin tuottaa esimerkiksi siten, että pistetään kaikki Suomen työttömät polkemaan kuntopyörillä, jotka on kiinnitetty dynamoihin, joista syötetään sähköverkkoon sähköä. Varmaan sillä ihan kohtuukokoisen kylän saisi valaistuakin ja ennen kaikkea saataisiin ihmiset työllistettyä. Mutta hölmöläisten hommaahan se olisi.

Sähkön koko olemassaolon tarkoitushan on, että ihmisten ei tarvitsisi tehdä töitä. Olemme keksineet sähköä käyttäviä koneita, jotka tekevät työn puolestamme samalla kun me makaamme sohvalla tuijottamassa sähköä käyttävää televisiota. Tämä yksinkertainen tosiasia kuitenkin helposti unohtuu monilta poliitikoilta, varsinkin kun puhutaan työttömyydestä.

Älkää ymmärtäkö minua väärin. Työttömäksi joutuminen on suurimpia vastoinkäymisiä mitä ihmisen elämässä voi sattua. Monet varmaan ottaisivat mieluummin sekä sairauden että avioeron, kunhan vaan saisivat pitää työnsä. Ja osittainhan työttömyys seuraakin aina teknologian edistysaskeleita. Kun traktori keksittiin, jäivät rengit ilman töitä ja ihmiset muuttivat kaupunkeihin. Onneksi saivat kaupungissa sitten tehtaan tuotantolinjoilta töitä. Mutta nyt kun tehdas uudistaa tuotantolinjaansa, lopputuloksena monesti on, että uudempi ja hienompi tuotantokoneisto vaatii toimiakseen aiempaa vähemmän työntekijöitä. Ja taas ihmisiä joutuu työttömäksi.

Mutta älkää syyttäkö insinöörejä. Insinöörien ansiosta meidän ei enää tarvitse tuhota selkäämme peltoa kuokkimalla. Insinöörien ansiosta koneet tekevät tehtaan raskaat työt, ja ihmiset katsovat vierestä. Insinöörien ansiosta voimme illalla maata sohvalla katsomassa televisiota. Tätä me kaikki halusimmekin, silloin kun traktoria vasta keksittiin. Koneiden, tietokoneohjelmien ja muiden keksintöjen tarkoitus on, että työtä ei tarvitsisi tehdä – tai ainakaan sitä ei tarvitsisi tehdä niin paljon.

Nyky-yhteiskunnan ongelma sitten on, että jäljelle jääneet työt jaetaan niin, että toiset jäävät tehtaaseen tekemään ihan yhtä paljon työtä kuin ennenkin ja toiset jäävät kokonaan ilman. Ehkäpä tässäkin olisi syytä ottaa oppia Open Source -ajattelusta, eli avoimuuden ja jakamisen periaatteista. Ehkäpä insinööri oli sen niin tarkoittanutkin, silloin kun traktoria keksi, että työtä olisi jokaiselle, ei liikaa, mutta sopivasti ja laiskottelukin jaettaisiin kaikille tasan.

10Hihh. Hahhah.

19. VALISTUNUT DIKTAATTORI

Nyt jätämme hetkeksi Linuksen ja hänen kavereidensa elämänarvojen tutkimisen ja otamme seuraavaksi oppia Open Source -yhteisön organisaatioista ja niiden hierarkioista. Miten hallitaan ohjelmistoprojektia, jonka kymmenet – tai Linuxin tapauksessa sadat – ohjelmoijat etätöskentelevät eri puolilta maapalloa ja vieläpä vapaaehtoisvoimin, ilman mitään virallista työtekijäsuhdetta projektiin?

Linus on Linux-projektin *valistunut diktaattori*. Hän ei itse kylläkään käytä itsestään tätä nimitystä vaan se on peräisin Eric Raymondin *The Cathedral and the Bazaar* -kirjasta, jossa kirjoittaja tutkii Open Source -projektien eri organisaatiomuotoja. Vaikka diktaattorimalli ei ole ainoa tapa, joilla Open Source -projekteja johdetaan, se on selvästi yleisin ja samalla myös epämuodollisin. Python-nimisen ohjelmointikielen luoja Guido van Rossumia kutsutaan python-piireissä ihan julkisesti tittelillä BDFL – Benevolent Dictator For Life (elinikäinen valistunut diktaattori). Ilmeisesti python-ohjelmoijat eivät aio suoda Guido-paralle eläkepäiviä lainkaan!

Valistunut diktaattori on projektin johtaja, jolla on yksin kaikki päätäntävalta.¹¹ Usein tämä auktoriteetti on luonnollinen seuraus siitä, että johtaja on myös projektin alullepanija, kuten Linus on Linuxin tapauksessa.

Meille länsimaisissa demokratioissa eläville ihmisille puheet diktatuurista saattavat kuulostaa epäilyttäviltä. Vaikka diktatuuriin liittyvä suoraviivaisuus on varmasti kustannustehokasta ja organisaatorakenne kevyttä, on historia opettanut meille myös diktaattoreihin liittyvistä ongelmista. Monet monarkeistamme ja diktaattoreistamme eivät ole olleet kovinkaan valistuneita. Joten vaikka demokratiaan liittyvät neuvottelut, kompromissit ja äänestykset tuntuvatkin usein kalliilta, tehottomilta ja ennen kaikkea tuskastuttavan turhauttavilta, olemme kuitenkin historian opettamina päätyneet juuri siihen. Jos Linus onkin meistä ehkä keskimääräistä valistuneempi, niin voimmeko luottaa siihen, että hänen seuraajansa – sitten kun se päivä koittaa – ei ole täydellinen katastrofi?

Voimme, sillä ei Linuskaan ole Linuxin johtaja, joka sattumalta ja onneksi sattuu olemaan valistunut. Asia on ihan päin vastoin. Hän on johtaja vain ja ainoastaan siksi, että on niin fiksu kuin on.

Open Sourcen periaatteista seuraa mielenkiintoinen dynamiikka, jolla on suoria seurauksia myös projektiorganisaation hierarkiaan ja organisaation jäsenten välisiin suhteisiin. Mitä tapahtuisi jos Linus jostain syystä päättäisi lyödä homman läskiksi ja alkaisi ihan ilkeyttään tekemään Linuxin typeriä ratkaisuja? Heti seuraavana päivänä muut Linuxin ohjelmoijat jättäisivät Linuksen pelleilemään yksinään, kopioisivat Linuxin lähdekoodit johonkin, missä Linus ei pääse niihin käsiksi, ja jatkaisivat ilman häntä. Todennäköisesti hyvin pian projektille myös valittaisiin – enemmän tai vähemmän tietoisesti ja enemmän tai vähemmän demokraattisesti – uusi valistunut diktaattori.

Tämä kaikki on mahdollista siksi, että koodi on avointa ja vapaasti kenen tahansa käytettävissä. Linuxella on diktaattorina kaikki päätäntävalta, mutta toisaalta hänellä ei ole mitään valtaa. Muut pitävät häntä johtajanaan ainoastaan siksi, että hän on niin lahjakas, eli valistunut. Valta ja vapaus ovat mielenkiintoisessa tasapainossa. Diktaattorilla on kaikki valta, ja muilla on vapaus äänestää jaloillaan.

Samasta syystä diktaattorimalli ei toimi hyvin joissain muissa ympäristöissä. Jos pomosi ei ole tehtäviensä tasalla, se on ikävä juttu, mutta minkä teet? Jos armeija tottelee sokeasti ylipäällikköään, joka osoittautuu sairaaksi tyranniksi, minne pakenet? Näissä tilanteissa ei ole läsnä sellaista avoimuutta, mikä Open Source -prosessiin kuuluu erottamattomasti. Ilman avoimuutta emme myöskään pysty täysin luottamaan toisiin organisaation jäseniin, vaan joudumme turvautumaan raskaisiin demokraattisiin prosesseihin suojautuaksemme valtataisteluilta ja erilaisilta muilta pihtaamisen muodoilta. Open Source -projekteissa avoimuus on niin kiinteä osa järjestelmää, että tällaisia varoimenpiteitä ei tarvita.

Linus diktaattorina on hyvin konkreettinen esimerkki avoimuuden periaatteen hyödyistä. Avoimuudesta seuraa suoraan, että Open Source -projektit eivät tarvitse raskasta projektiorganisaatiota, vaan voivat toimia hyvinkin kevyesti, ketterästi ja kustannustehokkaasti. Voisiko diktaattorimalli sitten toimia perinteisemmässä organisaatiossa? Tästä on olemassa ainakin yksi esimerkki.

WWW-konsortio, lyhyemmin W3C, on standardointielin, joka on luonut lähes kaikki webiin liittyvät tekniikat. W3C:n julkaisemia standardeja ovat muun muassa HTML, CSS-tyylisivut, XML, DOM – joka on osa JavaScriptiä – PNG-kuvaformaatti, SVG-kuvaformaatti ja monta muuta tekniikkaa, joita me internetin käyttäjät hyödynämme päivittäin. W3C:n jäsenenä on noin 450 IT-alan yritystä ja organisaatiota Microsoftista ja IBM:stä alkaen.

W3C:n päätöksentekoprosessi standardeja valmistellessa muistuttaa mielenkiintoisella tavalla Open Source -diktaattorimallia. Standardien valmistelutyö tapahtuu aiheeseen perehtyneessä työryhmässä, jonka työ on hyvin avointa. Standardeista julkaistaan useita esiversioita, joita kuka tahansa voi kommentoida. Kun työryhmä saa työnsä valmiiksi, tuodaan ehdotus äänestykseen. Äänestykseen saavat osallistua kaikki jäsenorganisaatiot ja jokaisella on siinä yksi ääni.

Äänestyksen jälkeen W3C:n johtaja yksin joko hyväksyy tai hylkää ehdotuksen. Johtaja voi siis täysin oman päänsä mukaan joko noudattaa tai olla noudattamatta äänestystulosta. Käytännössä W3C:n toiminta on hyvin konsensushakuista, mutta järjestelmä muistuttaa kuitenkin valistuneen diktaattorin mallia.

W3C:n johtajana on sen perustamisesta asti toiminut Tim Berners-Lee. Hän on sama mies, joka 90-luvun alussa CERN:ssä työskennellessään kehitti HTML:n ja muut webiin liittyvät tekniikat. Hän on siis webin Linus – luoja ja diktaattori.

Kuten W3C:ssä, myös Linuxin kehitystyössä päätökset tehdään yleensä perusteellisten keskustelujen pohjalta. Erityisesti Linus kuuntelee neuvoja lähimmiltä ja pitkäaikaisimmilta työtovereiltaan, joita yhteisö kutsuu hänen luotnanteikseen. He ovat siis eräänlaisia pikkudiktaattoreita, joilla on projektissa oma vastuualue. Aivan kuten Linuxenkin, myös heidän auktoriteettinsa perustuu vuosien aikana osoitettuun lahjakkuuteen ja sitä kautta saavutettuun luottamukseen. Tällaista järjestelmää kutsutaankin osuvasti *meritokratiaksi*.

Sellaistakin on tapahtunut, että diktaattori-Linus on lopulta joutunut murjottaen hyväksymään ratkaisun, jota enemmistö vaatii, vaikka hän itse on kovasti vastustanut sitä. Jos hän ei toimisi näin, hän ei ehkä joku päivä enää olisikaan diktaattori. Avoin järjestelmä toimii siis kevyesti mutta myös demokraattisesti.

11T ässä vaiheessa varmasti kaikki lukijakunnan projektipäälliköt heräsivät ja terästävät katsettaan!

20. SUVAITSEVAISUUS

Edellä olemme tutustuneet eräisiin mielenkiintoisiin piirteisiin, jotka ovat Open Source -yhteisölle erittäin tunnusomaisia. Vaikka jatkammekin vielä tämän mielenkiintoisen yhteisön tutkimista, on tässä vaiheessa todettava, että suvaitsevaisuus ei ole mikään tunnusomainen Open Source -yhteisön piirre, eikä myöskään sille kuuluva yksinoikeus. Eri keskustelupalstoilta ja sähköpostilistoilta on helppo löytää lukuisia esimerkkejä Open Source -yhteisön jäsenistä, jotka ovat kaikkea muuta kuin suvaitsevia. Kuitenkin yhteisöstä on myös mahdollista löytää opettavaisia tarinoita suvaitsevaisuudesta, joista voi olla inspiraatioksi meille kaikille.

Palaamme tarinassamme taas kerran Linuksen henkilöön ja niihin vuosiin kun IT -alan lehdistö oli löytänyt mielenkiintoisen ja tuoreen Linux-käyttöjärjestelmän. Tämä tapahtui aikaan, jolloin Microsoft oli onnistunut valtaamaan monopoliaseman sekä käyttöjärjestelmissä että toimisto-ohjelmissa ja oli koko painollaan parasta aikaa syöksemässä Netscapea, kilpailijaansa www-selainmarkkinoilla, kohti lopullista tuhoa. Tähän aikaan käyttäjät ihan syystä tuskailivat vaihtoehtojen puutteesta ja olivat ärsyntyneet sekä Microsoftin monopoliasemasta ja siihen liittyvistä väärinkäytöksistä, että toisaalta sen ohjelmistojen puutteista ja yllättävistä oikuista, mukaan lukien kiemurtelevat paperiliitinanimaatiot. Microsoftiin oli tietokonepiireissä alettu suhtautua suorastaan kaiken pahan alkuna ja siitä käytettiin esimerkiksi nimitystä "pahan valtakunta". Uusi haastaja Linux luonnollisesti kiinnosti toimittajia myös tästä näkökulmasta. Tulisiko Linux syrjäyttämään Microsoftin?

Taas kerran Linuxella oli lehdistölle yllättävä mutta viisautta henkivä vastaus. Hän totesi, että Microsoft ja sen tietokoneohjelmat eivät juurikaan kiinnosta häntä, sillä hän ei itse käytä Windowsia. Linuxia hän on tehnyt omaksi huvikseen, ei siksi että hänellä olisi jotain hampaankolossa Microsoftia tai ketään muutakaan kohtaan.

Vaikka silloin ja vielä enemmän tänä päivänä monet Microsoftin kilpailijat näkevät Linuxin nimenomaan Microsoftin haastajana, on hyvä ymmärtää, että Linuxin kehittäjät eivät ensisijaisesti ajattele tilannetta ylipääntään tästä näkökulmasta. Heillä todellakaan ei pääsääntöisesti ole mitään mielipidettä koko Windows-maailmasta, sillä he itse käyttävät Linuxia.

Vaikka itsekkin Microsoftin ohjelmia käyttäneenä ymmärrän täysin toimittajien näkökulman, on kuitenkin vaikea kuvitella, että Linux olisi menestynyt, jos sen pääasiallinen tarkoitus olisi ollut vain Windowsin korvaaminen. Sellaisen projektin osallistujat tuskin olisivat olleet niin innovatiivisia kuin Linuxin ohjelmoijat, jotka tekevät työtään rakkaudesta käyttöjärjestelmäohjelmointiin. Sellainen projekti olisi jo lähtöasemistaan tuomittu vain Microsoftia apinoivaksi räksyttäjäksi, joka ehkä joiltakin osin kyllä haastaisi Microsoftin tuotteet, mutta toisilla alueilla olisi ihan yhtä surkea ja mielikuviukseton kuin esikuvansakin.

Ennen kaikkea Linuxen suvaitseva asenne on kuitenkin järkevä hänen omaa mielenterveyttään ajatellen. Ei ole tervettä tehdä jotain, jonka pääasiallisena motivaattorina on viha ja pelko. Sitä paitsi, mitä tapahtuisi sitten kun Linux mahdollisesti jonain päivänä koituisi Microsoftin tuhoksi? Katoaako elämältä silloin tarkoitus? Täytyykö silloin löytää joku uusi pelottava uhka, jonka vastustamisesta voi hakea lisää energiaa?

Monet, joiden toiminta perustuu uhkakuva-ajatteluun, joutuvat tällaiseen kierteseen. Esimerkiksi sopii vaikka USA:n armeija. Olisi voinut kuvitella, että Neuvostoliiton kaatuminen olisi ollut amerikkalaisille sotilaille ja CIA:n vakoojille onnen päivä. Mutta mitä vielä, kaikkea muuta! Sehän tarkoitti heille vain ja ainoastaan uhkaavaa työttömyyttä. Jostain oli äkkiä löydettävä uusi uhka. Ensin yritettiin maalata taivaalle tummia pilviä pelottavan kansainvälisen huumekaupan muodossa. Kun se ei kuitenkaan oikein tuntunut ydinasevallan veroiselta vastustajalta, piti keksiä muita vastustajia. Nyt on lopulta löytynyt ihannevastustaja, nimittäin terroristit, joita on kaikkialla, mutta joita ei kuitenkaan löydy mistään. Uhan torjumisessa on taas työtä tarpeeksi, rahahanat ovat auki ja motivaatio huipussaan. USA on taas voimissaan.

Vaikka USA:n ulkomaanpolitiikan haukkuminen on jopa kivempaa kuin ranskalaisten maanviljelijöiden, niin on hyvä tässä vaiheessa muistaa, että ennen kaikkea puhumme nyt omasta itsestämme ja omista motiiveistamme. Monet ihmiset perustavat elämänsä ihan samanlaiselle uhkakuva-ajattelulle ja löytävät dramatiikkaa milloin mistäkin. Se on heidän elinvoimansa lähde. Mutta todellisuudessa he ovat itse luoneet kaiken dramatiikan. Jos he haluaisivat, he voisivat onnellisina ja rauhassa puuhastella oman "Linuxinsa" parissa, ja nauttia siitä ilman teennäisesti luotua dramatiikkaa.

Linuksen esimerkki suvaitsevaisuudesta on mielestäni erittäin oivaltava. Vaikka hän ei lahjakkaana ohjelmoijana voikaan hyväksyä bugisia ohjelmia omassa tietokoneessaan, niiden olemassaolo ei sinänsä haittaa häntä. Hän osaa itse ohjelmoida paremmin ja on tyytyväinen omaan työhönsä. Mutta toisaalta häntä ei haittaisi, vaikka kaikki muut ihmiset haluaisivat Linuxin sijasta käyttää Windowsia. Eihän se ole hänen ongelmansa. Erilaisten valtaisteluiden ja uhkakuva-ajatteluiden maailmassa, Linuksen suvaitseva asenne on kuin virkistävä tuulahdus, josta jokainen voi ammentaa rauhallisuutta omaankin ajatusmaailmaansa.

21. MONIMUOTOISUUS

Advogato-nimiseen Open Source -yhteisön keskustelufoorumiin kirjoitti kesällä henkilö, jonka mielestä Open Source -ohjelmistojen markkina oli liian pirstoutunut ja sekava. Jo pelkästään tekstieditoreita oli yli 100 erilaista. Linuxiin juuri siirtyneenä henkilönä tämä sai hänet tuntemaan itsensä lähinnä eksyneeltä. Olisi ollut parempi jos voitaisiin sopia mitä editoria käytetään, kehittää se huippuunsa ja opettaa kaikille sen käyttöä.

Keskustelunavaus ei saanut juurikaan vastakaikua. Lähinnä tyydyttiin toteamaan, että kirjoittaja ei ilmeisesti tunne Open Sourcea vielä kovin hyvin. Ne 100 editoriahan ovat esimerkki Open Source -maailman rikkaudesta. Tietenkin on totta, että suurin osa niistä saattaa olla monella lailla puutteellisia ja ovat syntyneet esimerkiksi jonkun tietotekniikkaopiskelijan harjoitustyön tuloksena. Mutta mitä se meitä haittaa? Hienoa, että hänkin on julkaissut työnsä avoimesti käytettäväksi, vaikka juuri minä en hänen editoriaan käytäkään.

Toisekseen niinkin yksinkertainen asia kuin puhtaiden tekstitiedostojen käsittely voi todellisuudessa tarvita hyvinkin erilaisia työkaluja. Esimerkiksi vi-niminen editori on peräisin 70-luvulta ja siksi varmaan yksinkertaisimpia ja kömpelöimpiä työkaluja mitä tietokonemaailma on ikinä nähnyt. Siitä huolimatta, tai oikeastaan juuri siksi, se kuuluu Unix-standardiin ja löytyy sen takia kaikista maailman Unix-koneista, mukaan lukien kaikki Linux-tietokoneet. Jos opettelet vi:n käytön, voit siis aina luottaa siihen, että saat työsi tehtyä. Se toimii kaikissa ympäristöissä, käytitpä sitä sitten etäyhteyden yli "terminaalissa" tai tavallisen PC:n "graafisessa ympäristössä". Toisaalta haluat ehkä joskus käyttää monipuolisempaa tekstinkirjoitustyökalua, esimerkiksi graafista sellaista. Silloin valitset jonkun niistä. Ja yllättävää kyllä, monet meistä valitsevat ohjelmansa paljolti myös henkilökohtaisten mieltymysten – esimerkiksi värin – perusteella. Vaikka siis ajatus siitä, että karsittaisiin päällekkäistä työtä ja kehitettäisiin Linuxia varten vain yksi tai korkeintaan pari tekstieditoria, on kaunis, niin käytännössä olisi ongelmallista päättää mikä editori tämän kunnian saisi. Kuka pääsee makutuomariksi?

Toinen asia missä samanlaista pirstaloituneisuutta näkyy on se, että Linux-maailmassa on tällä hetkellä kaksi eri graafista ikkunointiympäristöä:¹² KDE ja GNOME. Tämä on aina välillä ollut huolenaihe Linux-maailmassa. Vaikka GNOME:a varten tehdyt ohjelmat toimivatkin KDE:ssä ja päinvastoin, ne saattavat kuitenkin näyttää erilaisilta ja noudattaa hieman eri toimintaperiaatteita. Kaikki tämä voi varsinkin aloittelijalle olla sekavaa. Tilanne voidaan myös kyseenalaistaa tehokkuusnäkökulmasta. Käytetäänkö ohjelmoijien resursseja epäoptimaalisesti, kun työvoima on hajautettu kahden kilpailevan projektin ympärille?

Näistä kaikista ongelmista huolimatta Open Source -yhteisössä ollaan yhtämielisiä siitä, että monimuotoisuus – olkoonkin, että se vaikuttaa joskus epäoptimaaliselta – on hyvä asia. Ensinnäkin meidän täytyy muistaa Open Sourcen peruseriaatteen, joita on tässäkin kirjassa jo käsitelty. Ohjelmoijathan tekevät sitä mitä haluavat. Jos joku haluaa tehdä GNOME-ohjelmia, niin kuka häntä voi estää? Ja ketä se haittaa jos joku tekee GNOME-ohjelmia, vaikka kaikki muut haluaisivat käyttäjä KDE-ohjelmia? Nämä peruseriaatteen johtavat automaattisesti monimuotoisuuteen, sillä Open Source -yhteisö ei ole mikään Neuvostoliitto, jossa kaikkien pitäisi toimia kulloinkin voimassaolevan viisivuotissuunnitelman mukaan.

Kahden erillisen projektin kilpailutilanteesta on myös monia hyötyjä. Graafisten käyttöliittymien tekeminen on aika nuorta tiedettä, eikä kukaan oikeastaan tiedä mikä on paras ja ainoa oikea tapa tehdä asioita.¹³ Kilpailevat projektit saattavat joissakin asioissa päätyä erilaisiin teknisiin ratkaisuihin ja vasta jälkeenpäin voidaan nähdä kumpi oli viisaampi ja toimivampi ratkaisu. Näin kahden erillisen projektin olemassaolo vähentää riskiä, että graafinen Linux-maailma päätyisi teknologiseen umpikujaan.

Kilpailevat projektit myös ovat toisilleen aidosti hyödyksi. Vaikka KDE on alusta asti ollut teknisesti kehittyneempi, niin GNOME tunnettiin alunperin hyvin taiteellisenä ympäristönä. Erilaisia teemoja asentamalla omasta ympäristöstään pystyi saamaan hyvin eri näköisen ja ennen kaikkea oman näköisen. Yllättävän monet ihmiset muuten pitävät juuri tästä ominaisuudesta tietokoneessaan. Taustakuva ja ohjelman värit tuntuvat olevan paljon tärkeämpi asia kuin varsinainen työnteko. Niinpä KDE-ohjelmoijat ovatkin panostaneet vastaavien ominaisuuksien kehittämiseen omassa projektissaan ja nykyään KDE on vähintään yhtä värikäs ja taiteellinen kuin GNOME. Samanaikaisesti GNOME-ohjelmoijat taas ovat tehneet töitä ympäristönsä vakauden ja teknisten ominaisuuksien parantamiseen, jotta KDE:n etumatkaa tällä puolella saataisiin kiinni.

Kilpailutilanne siis hyödyntää molempia osapuolia. Mikä parasta, osapuolet myös itse ymmärtävät tämän. Vaikka internetissä usein törmääkin keskusteluihin, joissa väitellään KDE:n ja GNOME:n paremmuudesta – ja joskus jopa Advogato-kirjoittajan tapaan ehdotetaan yksinkertaisuuden nimissä toisen projektin lopettamista – niin nämä keskustelijat eivät koskaan ole KDE:n tai GNOME:n ohjelmoijia. Heillä ei ensinnäkään ole aikaa osallistua näin joutaviin keskusteluihin, sillä heillä on parempaa tekemistä. Mutta ennen kaikkea he eivät edes koe toisiaan kilpailijoiksi sanan perinteisessä merkityksessä, vaan ymmärtävät olevansa pikemminkin yhteistyökumppaneita – vaikka muu maailma ei sitä aina ymmärtäisikään.

Kysytäänpä siis lopuksi, taas kerran, miten muu maailma sitten asian ymmärtää?

Vaikka länsimaiseen markkinatalouteen nimenomaan kuuluu vapaa kilpailu, niin todellisuudessa yritykset eivät aina jaksa nähdä kilpailua positiivisena asiana. Kilpailu yleensä tarkoittaa vähemmän rahaa minun pussiini. Kilpailijoista kannattaa siis pyrkiä eroon keinolla millä hyvänsä. Microsoft on hyvä esimerkki siitä, että tällainen strategia on taloudellisesti kannattava – onnistuessaan erittäin kannattava. Tästä johtuen länsimaiset yritykset, joiden luulisi uskovan vapaaseen markkinatalouteen, ja jotka oikeastikin varmaan *uskovat uskovansa* vapaaseen markkinatalouteen, käytännössä kuitenkin tekevät kaikkensa estääkseen kilpailun syntymisen. Mikäli Linuxiin on uskominen, he samalla kaivavat myös omaa hautaansa.

12Ikkunointiympäristöllä tarkoitetaan sitä tietokoneohjelmaa, joka luo ruudulle nk. *graafisen käyttöliittymän*, jossa muita ohjelmia käytetään, osoitetaan hiirellä ja niin edelleen. Vastakohtana on terminaali- tai komentoriviympäristö, jossa yleensä käytetään vain näppäimistöä ja kirjaimet näkyvät vaaleina tietokone-ruudun tummaa taustaa vasten ja käytössä on vain yksi ohjelma kerrallaan.

13Vaikka Apple ja Microsoft ovatkin tehneet graafisia käyttöliittymiä jo yli kymmenen vuotta, on kyseessä kuitenkin lyhyt aika ja vain kaksi erilaista näkemystä. Windows-käyttöliittymäkin on kehittynyt ja kehittänyt edelleen – juuri siirryttiin XP:ssä uuden näköiseen ympäristöön ja Microsoft on ilmoittanut työskentelevänsä "Longhorn" nimisen projektin parissa, joka taas tuo uusia tekniikoita ja käytäntöjä. Tämän lisäksi Unixeissa käytetty X-arkkitehtuuri on hyvin erilainen kuin Windowsin tai Applen arkkitehtuuri, joten Unix-maailma ei voi suoraan ottaa kaikkea Windowsista, vaan KDE ja GNOME joutuvat joka tapauksessa tekemään paljon ihan aitoa pioneerityötä.

22. ROHKEUS JA UTELIAISUUS

Suosituksen Debian Linux -jakelun asennusoppaassa käytetään käyttöjärjestelmän asennuksen jälkeisestä ensimmäisestä koneen käynnistyskerrasta sähköinsinööreiltä lainattua ja hieman pelottavalta kuulostavaa termiä "smoke test", suomeksi savutesti. Savutesti liittyy sähköinsinöörien rakenteluprojekteihin, esimerkiksi komponenttien juottamiseen piirilevyille. Kun piirilevy on valmis, on aika kytkeä siihen sähköä. Jos kaikki on tehty oikein, piirilevy toimii suunnitellusti. Jos taas jossain on oikosulku, levyllä juotetut komponentit tyypillisesti pokahtelevat ja päästävät ilmoille pienen savupilven. Savu on siis merkki siitä, että piirilevy ei toiminutkaan.

Vaikka aineeton tietokoneohjelma ei savua pystykään tuottamaan, on vasta-asennetun tietokoneen ensimmäinen käynnistyskerta samalla lailla totuuden hetki. Se joko toimii tai sitten ei. Vastauksen saa tietää vain kokeilemalla.

Kuten insinöörejä yleisemminkin, myös Linux-ohjelmoijia vaivaa suunnaton uteliaisuuden palo. Heitä kiinnostaa esimerkiksi tietää, mitä kaikkea uutta ja hienoa Linusin uusimmassa Linux-kernelissä on, ennenkuin kerneli on edes valmis. Näitä ei-valmiita Linuxeja kutsutaan kehitysversioiksi, koska niitä siis vielä kehitetään ennen lopullista julkistusta. Vaikka niiden käyttöä ei suositellakaan, ne ovat samalla lailla saatavilla internetistä kuin virallisetkin versiot ja moni utelias Linux-käyttäjä päätyykin käyttämään niitä jo ennen lopullista julkistamista.

Tämä uteliaisuus on eräänlaista rohkeutta. Uuden tietokoneohjelman epävakaa kehitysversion käyttöönotto on aina riski. Se ei ehkä toimikaan. Varsinkin kun puhutaan kokonaisesta käyttöjärjestelmästä, on vaarana, että virheellisesti toimiva ohjelma saa koko tietokoneesi niin pahasti solmuun, että menetät peruuttamattomasti kaikki tietokoneellesi tallentamasi tiedostot. Tästäkin huolimatta uteliaisuus monesti voittaa ja innokas kokeilija ei välitä riskeistä ja työläistä varotoimenpiteistä. On pakko päästä kokeilemaan.

Minulle annettiin kerran työkoneeksi kannettava tietokone, joka toimi muuten hyvin, mutta akku oli rikki. Halusin selvittää oliko akku täysin kuollut, vai sellaista varsin yleistä mallia, joka jaksaa pitää koneen käynnissä noin viisi minuuttia ennenkuin sammuu. Tämän voisi tietenkin selvittää monella tavalla. Esimerkiksi lähettämällä akun huoltoon tai lainaamalla sähköinsinööriltä yleismittaria tai jollain muulla vastaavalla tavalla. Mutta yksi tapa oli kuitenkin kaikkia muita nopeampi ja siten paras tapa tyydyttää uteliaisuuteni. Kokeileminen.

Niinpä vedin sähköjohdon irti seinästä ja koneeni sammui välittömästi. Tämä oli sikäli ikävää, että en ollut muistanut tallentaa juuri kirjoittamaani tekstiä, joten jouduin kirjoittamaan sen uudestaan. Tästäkin huolimatta uteliaisuuteni oli vienyt voiton ja kokeileminen toi selkeän vastauksen kysymykseeni. Tiesin nyt, että kyseessä oli 100%:sti rikkiäinen akku.

Linux-ohjelmoijien kutsuminen rohkeiksi saattaa kuulostaa yllättävältä. Kun ajattelet keskiverto-ohjelmoijaa, mieleesi tulee ehkä suorattukainen, huonosti kammattu ja epämuodikkaita silmälaseja käyttävä, ujo, varovainen ja hiljainen nörtti. Vaikka kaikki ohjelmoijat eivät tätä stereotypiaa noudatakaan, niin Linux-ohjelmoijat ehkä jossain määrin ovat juuri tämän stereotypian henkilöitymiä. Voi siis olla lioiteltua kutsua tätä ujoa ja hiljaista nörttilaumaa rohkeaksi, mutta edellä oleva tarkastelu antaa aiheen uskoa, että yleisestä varovaisuudestaan ja ujoudestaan poiketen nörtit ovat tietokoneiden parissa rohkeita. Tietokoneisiin kohdistuva uteliaisuus voittaa muuten vallitsevan varovaisuuden ja nörtti muuttuukin rohkeaksi.

Kun tämä kirja on tähän asti tarjonnut Linux-yhteisöä esikuvaksi meille lukijoille, niin voi olla, että juuri ujut nörtit itse voisivat tällä kertaa kaikkein eniten ottaa oppia itsestään rohkeuden alalla.

Kuvitellaanpa esimerkiksi, mitä todennäköisesti tapahtuu, kun ujo ja varovainen nörtti istuu yksin koulunsa tai työpaikkansa ruokalassa ja samaan pöytään istuu tuntematon kaunotar. Todennäköisestihän ei tapahdu yhtään mitään! Nörtti tuijottaa alas omaan ruokaansa, syö sen kaikessa hiljaisuudessa ja lähtee. Jos vastapäätä istuva kaunotar on myös ujo ja hiljainen nörtti – tämä ominaisuushan ei ole pelkästään miesten yksinoikeus – niin hän todennäköisesti tekee samoin. Mitään ei siis vahingossakaan pääse tapahtumaan, ja tämä siitä huolimatta, että he voisivat olla kiinnostuneita toisistaan.

Minkälainen ajatuskulku nörtin päässä johtaa siihen, että mitään ei tapahdu? Häntä kiinnostaisi luoda kontaktia vastapäiseen kaunottaren. Pitäisi siis sanoa jotain. Mitään järkevää ei tule mieleen. Mutta silloin ei synny kontaktiakaan. Pitäisi siis sanoa jotain, ihan mitä vaan. Mutta silloin kaunotar voi pitää nörttiä tyhmänä. Lisäksi noin kaunis nainen on todennäköisesti jo varattu tai vaikka ei olisikaan, olisi tuskin nörtistä kiinnostunut kuitenkään. Nörtti ei uskalla ottaa riskiä tulla torjutuksi. Jos nörtti sanoisi jotain, voisi vastaus olla tyyli. Sitäkään riskiä nörtti ei halua ottaa. Ja näin siis päädyimme siihen, että nörtti ei sano mitään ja mitään ei tapahdu.

Jos nörtti käsittelee kaunotarta kuin tietokonetta, hän toimisi toisin – rohkeammin. Kaunotar istuu vastapäätä, tämä on nyt totuuden hetki. On savutestin paikka! Nörtti päättää sanoa jotain, siitähän huolimatta, että savupilvi voi poksahuttaa naamalle tylyn vastauksen muodossa. Onhan savutesti epäonnistunut monesti myös piirilevyjen tai tietokoneohjelmien yhteydessä, eikä se ole nörttiä haitannut. Oikeastaan nörtti ei näitä epäonnistumisia edes koe epäonnistumisina. Ne olivat vain erilaisia kokeiluja ja yhtä hauskoja ja opettavaisia kuin onnistuneetkin kokeet. Ja epäonnistuneiden savutestien kautta lopulta päädytään myös onnistumisiin.

Jos nörtti lähestyisi naista tällä logiikalla, hän heti paikalla avaisi keskustelun. Kaunottaren reaktio voi olla positiivinen, tai se voi olla negatiivinen. Molemmat vaihtoehdot ovat nörtin kannalta hyviä, sillä ne tyydyttävät hänen uteliaisuuttaan ja kokeilunhaluaan. Edes torjuva vastaus ei ole nörtin kannalta epäonnistuminen, se on vain savutesti, jonka jälkeen voi yrittää uudelleen.

23. NIMET JA IDENTITEETTI

On mielenkiintoista panna merkille, että yhteisössä, jossa kovasti vannotaan avoimuuden ja jakamisen nimeen, yhdestä asiasta pidetään kovasti kiinni: nimistä. Jopa niin paljon, että yhteisö, joka on kääntänyt päälaelleen omistamiseen ja tekijänoikeuteen liittyviä käytäntöjä, näkee tavaramerkit – eli laillistetun nimisuojan – suhteellisen positiivisena asiana. Esimerkiksi "Linux" on Linus Torvaldsin rekisteröimä tavaramerkki, eikä mitä tahansa käyttöjärjestelmää tai muuta ohjelmaa sen takia saa kutsua Linuxiksi. Linus sallii Linuxiksi kutsuttavan vain hänen julkaiseman Linux-kernelin ja sen muunnelmien päälle tehtyjä käyttöjärjestelmiä.

Ihmisen tai projektin oman nimen kunnioittaminen on itse asiassa myös yksi avoimuuden puolustusmekanismeista. Open Source -termi itsessäänkin on suojattu, ja termiä "OSI certified Open Source" saa käyttää vain ohjelmista, jotka täyttävät tietyt avoimuuden kriteerit.

Open Source -yhteisössä jokainen on avoimesti sitä mitä on – oma itsensä. Julkaistessaan uusia Linuxin versioita Linus esiintyy omalla nimellään. Voit luottaa siihen, että ohjelma jota kutsutaan Linuxiksi on juuri Linuxen tekemä, ei jotain muuta. Nimi toimii laadun takeena. Tekijä on itse henkilökohtaisesti työnsä laadun takeena. Mutta toisaalta taas työ tuo kunniaa tekijälleen. Ohjelman tekijä tulee yleensä tunnetuksi juuri työnsä kautta. Vaikka työn tuloksia jaetaan avoimesti, kunniaa työstä ei jaeta. Ja miten voitaisikaan, eihän sellainen olisi edes rehellistä. Kunnia työstä kuuluu yksinomaan työn tekijöille. Yhteisössä, jossa kaikki on avointa ja jaettava, tästä pidetään kiinni. Johonkin on voitava luottaa.

Mutta Open Source -yhteisön suhde nimiin on paljon vahvempi kuin pelkät kysymykset tavaramerkeistä tai brändeistä. Jotkut sanat liittyvät vahvasti yhteisöä koskeviin identiteettikysymyksiin. Niistä pidetään kiinni kysin hampain, vaikka muu maailma ei aina ymmärtäisi mistä on kysymys.

Yksi tällainen identiteettiin liittyvä sana on *hakkeri*. Suurin osa tämänkin kirjan lukijoista varmaan mieltää hakkerin synonyymiksi sanalle "tietokonekrikollinen". Eli sellaiseksi ihmiseksi, joka esimerkiksi internetiä hyödyntäen murtautuu toisen tietokoneeseen, lukee sieltä salaisia tiedostoja ja mahdollisesti tekee jotain ilkeältäakin. Jopa IT-alan lehdistö käyttää sanaa tässä merkityksessä.

Open Source -yhteisössä sana tarkoittaa kuitenkin jotain muuta. Tietokoneiden syntyäikoihin 60- ja 70-luvulla MIT:n ja muiden amerikkalaisten yliopistojen ohjelmoijat kutsuivat itseään *hakkereiksi*. Unix-käyttöjärjestelmän kehittyessä sana siirtyi sitä ympäröivään yhteisöön. Unixin ohjelmoijat olivat hakkereita tai Unix-hakkereita ja he olivat edelleen pääosin yliopistoväkeä. Sanaan ei liittynyt mitään rikollista, vaan ennemminkin siihen liittyi ajatus erityisen lahjakkaasta ohjelmoijasta, gurusta ja Unix-yhteisön jäsenestä, joka on intohimoisen innostunut ohjelmoinnista ja tekniikasta.

Alkuaikojen hakkerikulttuuri oli ideologiaaltaan hyvin samanlainen nykyisen Open Source -kulttuurin kanssa. Tosin Open Sourcea ei tunnettu erillisenä käsitteenä, vaan pikemminkin oli niin, että kaikki tietokoneohjelmat olivat avoimia. Yliopistoympäristössä syntyneelle kulttuurille tämä oli luonnollista. Muutos kohti sitä suljettujen ohjelmistojen kulttuuria, joka meidän mielestämme on normaalia, tuli vasta 80-luvulle siirryttäessä. Samalla hakkerikulttuuri jäi vähän sivuraiteelle. Pihtaamismalli, jolla suljettuja ohjelmistoja alettiin valmistamaan ja myymään, ei kuulunut hakkerikulttuuriin.

Jossain vaiheessa lehdistö alkoi raportoimaan tietomurroista ja syystä tai toisesta päätyi kutsumaan tekijää hakkeriksi. Nimitys ei ehkä alun perin ole ollut täysin huonokaan, sillä siihen aikaan – päin vastoin kuin nyt – on tietomurron tekijältä vaadittu paljon tietotaitoa ja lahjakkuutta, eli hän on siinä mielessä ehkä ollutkin hakkeri. Mutta tästä eteenpäin sana hakkeri tuli tarkoittamaan tietokonerikollista.

Olen monesti pohtinut, minkä takia Open Source -yhteisön jäsenet niin itsepäisesti haluavat edelleen kutsua itseään hakkeriksi? Sanan alkuperäinen merkitys on niin täydellisesti jäänyt uudemman merkityksen varjoon, että tuntuu suorastaan kummalliselta, että joku lainakuuliainen kansalainen haluaa enää kutsua itseään sillä nimellä. Jopa tietokonealan ammattilaiset menevät yleensä hämilleen, sillä hakkeri-sanan alkuperäistä merkitystä ei todellakaan hakkeripiirien ulkopuolella juuri tunneta. Eräskin lehtimies-raukka oli onnistunut kirjoittamaan pitkän artikkelin, jossa hän kuvasi miten pahamaaiseiset tietokonerikolliset olivat luoneet uuden Linux-nimisen käyttöjärjestelmän! Kukaan ei ilmeisesti ollut päässyt kertomaan hänelle hakkeri-sanan kahdesta eri merkityksestä. Artikkelia lukiessani en tiennyt olisiko pitänyt itkeä vai nauraa.

Mutta ymmärrän kyllä hakkereita. Miten kukaan voisi luopua nimestä, joka määrittelee oman identiteetin? Nimestä, johon liittyy niin pitkä ja kunniaikas historia, aina ensimmäisiin tietokoneisiin ja tekoälylaboratorioihin¹⁴ asti. Nimestä, johon kiteytyvät koko yhteisön ideologiset peruspilarit, joista Stallman jaksaa saarnata yhä uudelleen ja joista filosofi Himanen niin ylevästi kirjoittaa kirjassaan Hakkerietiikka. Kaikesta sekaannuksesta huolimatta hakkerit ovat ylpeitä hakkeruudestaan. Ja täytyy myöntää, että koko sydämestäni ymmärrän heitä. Emmehän me suomalaisetkaan ole vaihtaneet maamme nimeä, vaikka monissa kielissä finni varsinaisesti tarkoittaakin mätäpaisetta!¹⁵

On olemassa toinenkin sanoihin liittyvä tarina, johon hakkerit suhtautuvat yhtä voimakkaasti kuin hakkeri-sanaan. Mutta päinvastoin kuin hakkeruus, nämä sanat eivät ole hakkereita yhdistäviä, vaan ne ovat olleet monen kitkerän väittelyn syynä.

Kun kaupalliset suljetut ohjelmistot yhä enemmän alkoivat vallata alaa pääosin yliopistoihin majoittuneelta hakkerikulttuurilta, jäljelle jäi yksi mies, joka päätti pyhittää elämänsä taistelulle pihtaamista vastaan. Vuonna 1984 Richard Stallman ilmoitti perustaneensa GNU-projektin. Projektin tavoitteena oli tuottaa vapaita ohjelmistoja – *Free Software* – ja jonain päivänä julkaista täysin vapaa käyttöjärjestelmä.¹⁶ Richard Stallman ei ollut mikä tahansa propellipää. Hän todella antautui projektilleen sataprosenttisesti. Hän irtisanoi itsensä tutkijan työpaikastaan yliopistolla, sillä hänen tuli varmistua siitä, ettei yliopisto voisi väittää omistavansa hänen tekemiä ohjelmia, eikä hän siksi voinut olla työsuhteessa yliopistoon. Stallmanin esimieheltä MIT:ssä ei puuttunut visionääriin kykyjä häneltäkään. Ymmärrettyään Stallmanin eron syyt, hän määräsi, että Stallmanin entinen työhuone ja yliopiston muut palvelut – erityisesti tietokoneet – olisivat edelleen hänen käytössään. Ilman tätä projektista tuskin olisi mitään tullutkaan. Rahapulaan joutunut Stallman jopa asui jonkin aikaa työhuoneessaan.

Näin alkoi vapaiden ohjelmistojen, eli Free Softwaren, historia. Projekti oli ilmeinen menestys, sillä jo vuoteen 1991 mennessä, kun Linus alkoi puuhastelemaan oman projektinsa parissa, lähes kaikki käyttöjärjestelmään kuuluvat työkaluohjelmat olivat valmiina ja puuttuva Hurd-kernelikin oli "melkein valmis". Kun Linux sitten oikeaan aikaan täytti tämän aukon, oli Stallmanin unelma vihdoinkin toteutuksessa.

Suurin osa 90-luvusta kului internet-huuman vallassa, eikä vapaasta käyttöjärjestelmästä vielä paljoa tiedetty. Mutta pinnan alla internetin leviäminen kiihdytti myös vapaiden ohjelmistojen kehitystä ja leviämistä entisestään, sillä internetin ansiosta yhä useammat hakkerit pystyivät osallistumaan maailmanlaajuisiin Open Source -talkoisiin. Ja vaikka lehdistö ei Linuxista paljoa vielä kirjoitellut, niin todellisuudessa esimerkiksi Suomessa suurin osa internetin palvelimista toimi jo Linuxin varassa. Vuosikymmenen lopulle tultaessa sekä lehdistö että yritysmaailma alkoi tosissaan kiinnostumaan sekä Linuxista, että sen syntyyn liittyvistä avoimista kehitysmalleista.

Tässä vaiheessa kokoontui neuvonpitoon joukko hakkereiden johtajahahmoja. (Ei siis tietokonekollisia, vaan Free Software -yhteisön johtohenkilöitä.) He näkivät tilaisuutensa vihdoinkin koittaneen. Jo lähes unhoon jäänyt hakkerikulttuuri oli uudelleen haastamassa kaupalliset ohjelmistotalot. Yritysmaailma oli aidosti kiinnostunut vapaista ohjelmistoista. Mutta miten tilanne saataisiin hyödynnettyä parhaalla mahdollisella tavalla? Miten saataisiin yritykset panostamaan Linux-järjestelmiin? Miten saataisiin ohjelmistotalot siirtymään avoimeen kehitysmalliin? Ja miten tulisi varautua Microsoftin väistämättömään vastahyökkäykseen?

Hakkeri-isät tulivat siihen tulokseen, että vapaat ohjelmistot tarvitsivat uuden brändin, jonka eteen tehtäisiin paljon pr-työtä. Näin syntyi vuoden 1998 alussa termi Open Source – avoimen lähdekoodin brändi.

Miten nimeen Open Source päädyttiin? Stallmanin Free Software -termiä oli kritisoitu epäselväksi. Englannin sana Free, joka tarkoittaa vapautta, tarkoittaa myös muun muassa ilmaisuutta. Ilmaista ohjelmistoista puhuminen taas ei kuulostanut hyvältä idealta, kun haluttiin saada ohjelmistotalot kiinnostumaan asiasta. Ja Free-sanan semantiikasta keskusteleminen taas on hämmentävää ja johtaa väistämättä sivuraiteille ja varsinainen asia saattaa unohtua. Brändinhän tulee olla lyhyt ja ytimekäs, ei epämääräinen.

Sinänsä Stallman oli jo aikoja sitten joutunut selvittämään samaa semantiikkaongelmaa, ja keksinyt aika nasevan muistisäännön: "Free as in Speech, not Free as in Beer."¹⁷ Vaikka ilmainen olut on hieno juttu, on sananvapaus kuitenkin tärkeämpää. Ja tästä vapaudesta Stallman mielellään puhuikin – ja juuri siinä olikin koko ongelman ydin. Free-sanan moniselitteisyyden kanssa olisi ehkä vielä pystytty elämään, mutta kun tietokoneohjelmien yhteydessä joku alkaa puhumaan sananvapaudesta, niin siinä on jo uskottavuus vaakalaudalla. Eikä siinä vielä kaikki, Stallman puhui myös mielellään mielipiteistään, joiden mukaan suljettujen ohjelmien malli oli suorastaan epäeettinen!

Juuri nämä puheet olivat todellisuudessa Open Source -termin synnyn takana. Muut hakkerivanhimmat olivat päättäneet ottaa pesäeroa sinänsä kunniaakkaan GNU-projektin liian ideologisesta retoriikasta. Sellainen ei sopinut Wall Street Journalin tapaisten lehtien haastatteluihin. Tarvittiin uusi identiteetti, joka olisi salonkikelpoinen sekä Wall Streetillä että ohjelmistoyrityksissä. Takkutukkainen Stallman haluttiin syrjään ja tilalle tarjottiin hymyilevää Linus-kansikuvapojua.

Mutta uuden identiteetin luomiseksi tarvittiin ensin uusi nimi. Open Source kertoi selkeästi mistä oli kysymys: koodi oli avointa. Avoinuudesta seurasi laatu, kuten Eric Raymond The Cathedral and the Bazaar -kirjassaan selvitti. Open Source = laatu! Jos haluat parhaita tietokoneohjelmia, luovu Windowsista, käytä Linuxia! Jos haluat halvimmän vaihtoehdon, käytä Linuxia! Me emme tiedä etiikasta mitään, mutta tiedämme mikä menetelmä toimii tietokoneiden ohjelmoinnissa parhaiten: Open Source!

Open Source -brändi onnistui loistavasti. Linus sai hymyillä lehtien kannessa oikein olan takaa. Linux-yhtiö Red Hatin pörssikurssi nousi ensimmäisenä pörssi päivänään nelinkertaisesti. Yritykset alkoivat vähitellen käyttää Linuxia yhä enemmän. Netscapesta tuli ensimmäinen suljettu ohjelmisto, jonka koodi avattiin ja kehitystyö siirtyi Open Source -pohjaiseksi Mozilla-projektiksi. Myöhemmin muut ovat seuranneet perästä, mukaanlukien eräs aiemmin StarOfficena tunnettu OpenOffice-tekstinkäsittelyohjelma, jolla tämäkin kirja on kirjoitettu. Ennen kaikkea lehdistö ja tavallinen kansa oppi tuntemaan Open Source -käsitteen.

Mutta kaikki eivät olleet tyytyväisiä. Stallman ei hyväksynyt Open Source -termin käyttöä. Hänen mielestään oli tärkeää ymmärtää vapaiden ohjelmien taustaan liittyvää ideologiaa, eikä tyytyä "käyttämään sitä mikä toimii parhaiten". Ja vaikka Open Source -leiri saavuttikin enemmän vastakaikua hakkeriyhteisön ulkopuolella, niin hakkeriyhteisön sisällä moni pysyi uskollisena Stallmanille ja alkuperäiselle Free Software -ideologialle.

Vaikka katkerimmat taistelut Free Software ja Open Source -leirien välillä ovat onneksi jo takana päin, voi yhä tänään päätellä kumpaan leiriin henkilö kuuluu, siitä kumpaa termiä hän käyttää Linuxista puhuessaan. Kysymys ei ole vain mistä tahansa sanaparista. Kysymys on identiteetistä ja siihen liittyvästä ideologiasta. Ja niistä hakkerit pitävät kiinni.

14Siinäpä toinen hauska nimi. Vaikka tietotekniikan kehitys sen jälkeen onkin ollut huimaa, odottaa todellinen tekoäly edelleen keksijänsä. 60-luvun tietokoneet olivat meidän näkökulmastanne pelkkiä sähkösyöppöjä lämpövastuksia, jotka eivät laskentatehossa pärjäisi yksinkertaisimmallekaan taskulaskimelle. Tästä huolimatta puhuttiin jo silloin tekoälystä.

15Kun seuraavan kerran puhut tietokonerikollisista, voit tehdä hakkereille mieliksi ja kutsua tekijää samalla nimellä kuin hekin: krakkeri (englanniksi cracker). Ainakin minä haluan varata hakkeri-nimen niille ihmisille ja heidän yleisille periaatteilleen, joista tämäkin kirja kertoo.

16Jollainen Linuxista noin kymmenen vuotta myöhemmin tuli. Suureksi osaksi saamme kiittää siitä juuri GNU-projektia.

17Eli vapaasti suomennettuna: Kyse on samasta asiassa kuin sananvapaudessa, ei samasta asiassa kuin ilmaisessa oluessa.

24. MITÄ ON ETIIKKA?

Kamppailu Free Software ja Open Source -leirien välillä johtaa meidät lopuksi yllättäen erään mielenkiintoisen kysymyksen äärelle. Mitä on etiikka?

Open Source -leirin tyypillisin edustaja on juuri Linus. Hänen asennettaan voisi kutsua insinöörimäisen käytännönläheiseksi. Tämä johtuu hänen luontaisen asennoitumisensa lisäksi myös osittain siitä, että Linus katsoo, että hänen asemansa Linux-projektin johtajana velvoittaa hänet olemaan mahdollisimman neutraali. Hän ei siksi halua sekaantua poliittisiin kysymyksiin. Hänen johtolauseenaan on suurin piirtein: Me teemme sen, mikä toimii parhaiten ja ohjelmoinnissa Open Source toimii parhaiten.

Free Software -puolen oppi-isä Stallman taas pitää rajoittumista noin yksinkertaiseen ajatteluun hyvin vaarallisena. Hänen mielestään avoimen ja suljetun ohjelmistokehityksen välillä on huomattava ero, ja tällä erolla on vaikutus niinkin tärkeisiin asioihin kuin esimerkiksi tasa-arvo ja valtionhallinnon läpinäkyvyys, ja toisaalta moneen tärkeään tekniseen kysymykseen kuten tietoturvaan ja luottamussuhteisiin. Näistä kaikista syistä seuraa, että kysymys parhaasta ohjelmistojen kehitysmallista on ennen kaikkea eettinen.

Mitä siis on etiikka?

Hullun lehmän taudinhan todettiin johtuvan siitä, että lehmien rehuihin oli alettu sekoittamaan lampaiden ja lehmien aivoista ja luista jauhettua sekoitusta. En tiedä mikä tässä eurooppalaisessa maataloudessa oikein on vikana, mutta tässä sitä taas ollaan! Hyvät aivot kun muuten olisivat menneet ihan hukkaan, niin päätettiin syöttää ne lehmille itselleen ja säästää siten oikean rehun kuluissa. Ja parin vuoden kuluttua siitä meillä on hullun lehmän tauti, ja kymmeniä eurooppalaisia kuolee syötyään näitä lehmiä.

Vaikka Suomi säästyikin tältä lähinnä brittejä vaivanneelta epidemialta, niin kriisiä puitiin myös täällä. Eräässä television keskusteluohjelmassa pohjoissuomalainen maanviljelijä kertoi, että aivorehua oli 90-luvun alussa tarjottu myös Suomeen. Suomessa maanviljelijät pitivät kuitenkin ajatusta lehmien lihottamisesta toisten lehmien teurasjätteellä täysin epäeettisenä ja kieltäytyivät ostamasta rehua.

Pohjois-Suomesta siis löysimme yllättäen maanviljelijän, joka puhui *eettisyydestä*. Nythän surullisen kuuluisa "lihaluujauho" on jo kielletty koko Euroopassa, kun kävi ilmi, että se oli hullun lehmän taudin synnä. Mutta Suomessa maanviljelijät olivat kieltäytyneet sen käytöstä jo kymmenen vuotta aiemmin, koska se ei heidän mielestään ollut *eettistä!* Ja koska lihaluujauhoa ei täällä oltu käytetty, Suomi myös säästyi hullun lehmän taudilta ja traagisilta ihmiskohtaloilta, joita siihen muualla Euroopassa liittyi.

Voisi sanoa, että lihaluujauhon kieltäminen on järkevää, koska se toimii. Kuka tahansa ymmärtää, että lehmille ei voida syöttää sellaista rehua, joka saa lehmät sairastumaan ja joka on hengenvaarallista myös lehmänlihan syöjälle. Mutta tämä on valitettavasti jälkiviisautta. Olisi tietenkin ollut parempi, jos lihaluujauhoa ei alunperinkään olisi syötetty lehmille ja siten oltaisiin säästytyt niiltäkin turhilta kuolemilta, jotka epidemia aiheutti. Mutta miten kukaan olisi voinut arvata, että näin tulee käymään?

Aiheellinen kysymys! Ja kuitenkin oli myös niitä maanviljelijöitä, jotka eivät käyttäneet lihaluujuuhoja. He eivät vedonneet mihinkään käytännön syihin, vaan eettisyyteen. Jälkikäteen katsottuna tämä ratkaisu kuitenkin säästi ihmishenkiä. Ja jälkikäteen katsottuna tämä ratkaisu oli se, joka toimi.

Kun Open Source -leirin kannattajat puhuvat epäpoliittisesti "vain siitä mikä toimii", he ovat tavallaan oikeassa. Mutta olisiko Stallman voinut käyttää samaa argumenttia vuonna 1984, kun suljettujen ohjelmistojen suuntaan oltiin vasta siirtymässä? Koska ohjelmistoala kaiken kaikkiaan oli niin nuori, ei voitu vedota faktoihin ja käytännön kokemuksiin kummankaan mallin puolesta. Siksi Stallman joutuikin puhumaan eettisyydestä. Ja hän oli oikeassa, mutta se tultaisiin näkemään vasta myöhemmin.

Myös Stallman päätyi Free Software -ristiretkelleen tavallaan juuri siksi, että hän havaitsi etteivät suljetut ohjelmistot toimineet yhtä hyvin kuin vapaat. Esseessään "The GNU Project" hän kertoo, kuinka heidän laboratorioissaan oli tulostin, jonka hallintaohjelma oli kömpelö ja siksi sen käyttäminen oli erittäin turhauttavaa. Lahjakkaana ohjelmoijana Stallman tiesi, että hän voisi korjata ongelman helpostikin, mutta laitteiston myyjäpä ei suostunutkaan antamaan tulostinohjelmiston lähdekoodia! Tämä episodi osaltaan vaikutti Stallmanin myöhemmin tekemiin johtopäätöksiin.

Ympyrä siis sulkeutuu. Kun Linus pitäytyy omassa "vain se mikä toimii" linjassaan, hän itse asiassa puhuu etiikasta! Eettiset ratkaisut ovat eettisiä juuri siksi, että ne ovat oikeita. Ja oikeat ratkaisut ovat oikeita siksi, että ne toimivat.

LUKU 3

25. LUKU 3

26. HAKKERIN BISNESMALLIT

27. ONHAN VESIKIN ILMAISTA

28. ANNA OHJELMAT ILMAISEKSI, MYY
PALVELUJA

29. MESENAATTIMALLI (O'REILLY & LARRY
WALL, TRANSMETA & LINUS, KDE)

30. MAKSETAAN TYÖNTEOSTA (SAKSA &
KROUPWARE)

31. STEPHEN KING, JUOMARAHAMALLI JA
TYÖNVÄLITYSPÖRSSI (STEPHEN KING,
ROGER WILLIAMS, SOURCEFORGE, KOLAB,
JBOSS)

32. KAHDET LISENSSIT (MYSQL,
TROLLTECH QT)

33. VÄHÄN MOLEMPIA (XIMIAN
EVOLUTION, CODEWEAVERS,
TRANSGAMING)

34. EPÄONNISTUMISEN TARINA (COREL)

35. ÄLÄ OLE LIIAN AHNE (JAVA,
GHOSTSCRIPT)

36. OPEN SOURCE JA KOUKKU (RED HAT
NETWORK JA SUSE YAST)

37. VAPAUTETUT I: NETSCAPE/MOZILLA

38. VAPAUTETUT II: INTERBASE/FIREBIRD

39. VAPAUTETUT III: QUAKE

40. VAPAUTETUT IV:
STAROFFICE/OPENOFFICE.ORG

41. VAPAUTETUT V: ECLIPSE

42. VAPAUTETUT VI: BLENDER

43. TIE PÄÄTTYY, MATKA ALKAA (LINUX
WEEKLY NEWS)

44. KIITOS, MUTTA ME HALUAMME
MAKSAA (MANDRAKE CLUB)

- 45. TALKOOT JA PYYKKINARUPARADOKSI
(DEBIAN)**
- 46. LASITALO, TÄYSIN LÄPINÄKYVÄ
YRITYS (FIKTIIVINEN)**
- 47. YHTEENVETO**

25. LUKU 3

jossa tasapainoillaan periaatteiden ja rahanhimon välillä,
Netscape vapautetaan, Stephen King jättää romaanin viimeisen sivun
kirjoittamatta ja työnteosta maksetaan palkkaa.

26. HAKKERIN BISNESMALLIT

Hakkerien kovin erilainen ideologia verrattuna ohjelmistoalan valtavirtaan on tietenkin haaste myös hyvin konkreettisessa mielessä. Jättääkö Open Source mitään tilaa kestäväälle yritystoiminnalle, jos ohjelmistoja pitäisi jakaa ilmaiseksi?¹⁸ Loppujen lopuksi puheet etiikasta ovat kuitenkin tyhjiä, jos ideologia ei toimi käytännössä. Silloin kysymyksessä olisi utopia, joka kuulostaa hienolta, mutta on pohjimmiltaan kestävätkö. Linuksen "vain se mikä toimii" periaate puoltaa siis paikkaansa ja yritysmaailma on hyvä mittari sille, ovatko edellä esitetyt ajatukset toimivia myös käytännössä. Jotta kysymyksessä ei olisi vain utopistinen haave, meillä tulisi olla näyttöä siitä, että Linux-yritykset, jotka toimivat hakkeruuden periaatteita noudattaen, pärjäävät myös taloudellisesti yhtä hyvin tai jopa paremmin kuin jotkut muut yritykset.

Open Source on jopa tietotekniikan alan mittakaavassakin vielä uusi asia ja alan yritykset ovat poikkeuksetta nuoria. On vielä liian varhaista lyödä lopullisesti lukkoon, mitkä ovat olleet menestyksiä ja mitkä tulevat hyvästä alusta huolimatta päätymään umpikujaan. Tästä huolimatta tulemme kuitenkin tässä luvussa käymään läpi todellisia esimerkkejä sekä menestyneistä että huonommin onnistuneista Linux-maailman liiketoimintaideoista. Linux-markkinoiden täysin uudenlaiset pelisäännöt näkyvät monen idean innovatiivisuudessa. Uusi tilanne on vaatinut uudenlaista ajattelua.

¹⁸Todellisuudessa Open Source ei tarkoita sitä, että ohjelmat ovat ilmaisia, vaan ainoastaan sitä, että asiakkaalla eli käyttäjällä on oikeus ohjelman lähdekoodiin, jatkokehitykseen ja jatkolevitykseen. Nämä ehdot kylläkin estävät esimerkiksi Microsoftmaisen monopoliasetelman syntyminen ja pitävät sitä kautta hinnat kohtuullisina, mutta ei kuitenkaan välttämättä nollassa. Käytännössä suurin osa Open Source -ohjelmista kuitenkin on ladattavissa ilmaiseksi internetistä, mutta ilmaisuus ei millään lailla ole pakollista.

27. ONHAN VESIKIN ILMAISTA

Ottaen huomioon, että käytännössä kaikki Open Source -ohjelmat ovat ladattavissa ilmaiseksi internetistä, on yllättävän moni yhtiö kuitenkin rakentanut liiketoimintansa juuri Linux-käyttöjärjestelmän ja sen oheisohjelmien myynnille. Näitä yrityksiä kutsutaan ns. *Linux-jakeluiksi* eli -distributioiksi. Tunnetuimpia niistä lienevät Red Hat Linux, SUSE Linux, Mandrake Linux ja esimerkiksi suomalainen SOT Linux. Vielä pari vuotta sitten kaikki nämä yritykset saivat valtaosan tuloistaan myymällä Linux CD-levyjä pakattuina värikkäisiin pahvilaatikoihin, ihan niinkuin perinteiset ohjelmistotalotkin tekevät.

Tätä liiketoimintamallia voidaan perustella analogialla vedestä. Vesi on ilmaista, mutta tästä huolimatta veden myyminen voi olla hyvä bisnes. Siis muutenkin kuin vain veden myyminen aavikolla asuville arabeille. Esimerkiksi me kaikki ostamme vettä päivittäin kun avaamme hanan. Vaikka vesijohtovesi ei kovin kallista olekaan, niin kyllä vesilaskumme aina yhden vedenpuhdistamon verran väkeä työllistää ja päälle vielä pari putkimiestä. Vesijohtoveden lisäksi useimmat meistä ovat myös ostaneet Eviania tai jotain muuta kallista lähdevettä pullotettuna. Veden myynti voi siis monellakin tavalla olla kannattavaa liiketoimintaa.

Vaikka internet on täynnä ilmaisia Open Source -ohjelmia, niin monille käyttäjille on kuitenkin paljon käytännöllisempää ostaa nippu CD-levyjä, joille kaikki tärkeimmät ohjelmat on koottu valmiiksi, sen sijaan että joutuisi viettämään tuntikausia hitaan internet-yhteydensä ääressä keräten niitä ilmaiseksi verkosta. Analogia veteen on sikäli mielenkiintoinen, että myös CD-levymarkkinoilla on myynnissä kahdenlaisia kokoelmalevyjä. Jotkut yritykset ovat erikoistuneet tekemään levyjä, joille on koottu mahdollisimman paljon erilaisia ohjelmia ja myyvät näitä levyjä mahdollisimman halvalla. Tämä on ikäänkuin vesijohtovettä. Toisaalta taas Red Hatin kaltaiset Linux-jakelut myyvät levyjä oman tuotemerkkinsä suojassa ja omilla logoillaan varustettuina ja kummallista kyllä, ihmiset mielellään maksavat vähän enemmän aidosta Red Hat Linuxista sen sijaan, että ostaisivat merkittömän CD:n, joka voi olla sisällöltään identtinen.

Tämä "vesikin on ilmaista" malli muistuttaa monilta osin perinteistä ohjelmistoalaa. Yritysten liiketoiminta perustuu molemmissa tapauksissa ohjelmia sisältävien CD-levyjien myymiseen. Samanlaisuus on kuitenkin enimmäkseen näennäistä. Kun ostat tietokoneohjelmia esimerkiksi Microsoftilta, et osta CD-levyä siksi, että tarvitset itse levyä, vaan ostat levyn siksi, että sinun on lain mukaan pakko ostaa se, jos aiot ylipäättään käyttää ohjelmaa. Jos hankit kotiisi toisen tietokoneen ja haluat käyttää siinä samaa Microsoft Office -ohjelmaa kuin ensimmäisessäkin koneessa, niin sinun on pakko ostaa myös uusi CD. Et luonnollisestikaan tarvitse tuota CD:tä mihinkään, sillä sinullahan on kotonasi jo yksi aivan samanlainen, mutta sinun on kuitenkin pakko ostaa se.

"Vedenmyyntimallin" mukaan toimivat Linux-yritykset taas eivät pakota ketään ostamaan mitään. Jos CD:stä on sinulle apua, voit ostaa sen, mutta jos et tarvitse CD:tä, niin älä sitten osta sitä. Voit käyttää itse ohjelmaa joka tapauksessa, omistitpa CD:n tai et. CD-levyjä myyvät yritykset hyväksyvät tämän ja kaikki ovat tyytyväisiä. Pihtaamismalli taas tuntuu kovin keinotekoiselta ja itse asiassa monia tavallisia tietokoneenkäyttäjiä – kutsutaan heitä vaikka maallikoiksi – on yleensä vaikea saada ymmärtämään, miksi heidän on pakko ostaa kallis CD, jota he eivät tarvitse mihinkään. CD-levyjen myyminen CD-levyinä siis ainakin tuntuu monella lailla luonnollisemmalta kuin erilaiset pihtaamiseen perustuvat teennäiset pakotteet.

28. ANNA OHJELMAT

ILMAISEKSI, MYY PALVELUJA

Jo 90-luvun puolivälistä on peräisin ajatus "give away software for free, sell services for a fee". Siis suomeksi: annetaan ohjelmat ilmaiseksi, myydään palveluja. Myös "vedenmyyntimalli" on tavallaan yksi tämän periaatteen toteutus.

"Myy palveluja" mallin peruseriaate on yksinkertainen. Vaikka tietokoneohjelmat olisivatkin ilmaisia, se ei useimpia ihmisiä vielä auta. Keskiverto tietokoneenkäyttäjä ei osaa edes asentaa ohjelmia, eli hän tarvitsee apua jo alkuun pääsemisessä. Moni voisi asennusavun lisäksi myös hyötyä pienestä käyttöönottokoulutuksesta. Yksinkertaisimmillaan palvelujen myynti voi olla sitä, että ohjelmat sisältävän CD-levyn ohessa on helppolukuinen ja auttava käyttöopas. Tällöin tietokoneohjelman CD:tä ei niinkään osteta itse CD:n takia vaan mukana seuraavan oppaan.

On siis helppo nähdä, että ilmaisen tietokoneohjelman ympärille voi hyvinkin rakentaa kannattavaa liiketoimintaa. Edelliset esimerkit olivat vasta alkua. Todella isoa liiketoimintaa palvelujen myynti on silloin, kun puhutaan isojen yritysten tai yhteisöjen tietokonejärjestelmistä. Ilmaisten ohjelmien asentaminen esimerkiksi 14 000 Münchenin kaupungin tietokoneeseen on iso urakka. Kaikki näiden ohjelmien käyttäjät pitää kouluttaa. Siirto pitää suunnitella tarkoin ja toteuttaa mahdollisimman sujuvasti, niin että se mahdollisimman vähän haittaa varsinaista työntekoa. Joitakin ohjelmia pitää ehkä muokata – alan termistön mukaan "räätälöidä" – jotta ne toimisivat sujuvasti jonkun Münchenin kaupungin sisäisen tietojärjestelmän kanssa. Käyttäjille pitää tarjota mikrotukea ja koneita pitää huoltaa ja päivittää. Tietoturva pitää jatkuvasti pitää ajan tasalla. Ja niin edelleen, ja niin edelleen...

Tällaisen urakan Münchenin kaupunki on tilannut IBM:ltä ja SUSE:ltä ja kaupan arvo on arvioitu noin 35 miljoonan euron suuruiseksi. Mistään ilmaisuudesta lienee siis turha puhua!

Palvelumalli ei sinänsä ole kovin eksoottinen ajatus. Kaikki isot tietotekniikkatalot, jotka ennen ovat keskittyneet myymään tietokoneita, niiden ohjelmia tai molempia, ovat nykyään profiloitumassa entistä enemmän juuri konsultointia myyviksi palveluyrityksiksi.

Palvelujen myyminen on kuitenkin erityisen kiinnostava tapaus silloin, kun se rakentuu Open Source -ohjelmien ympärille. Suljetun ohjelmiston tai laitteiston ympärille rakennettuja tukipalveluja nimittäin uhkaa sama monopolisoitumisen vaara kuin suljettuja ohjelmistoja itseään. Jos esimerkiksi ostat IBM:n suurkoneen¹⁹ ja siihen kuuluvan IBM AIX -käyttöjärjestelmän, niin tarvitset erittäin todennäköisesti hieman apua käyntiin pääsemisessä. Tätä apua joudut käytännössä ostamaan myös IBM:ltä, sillä kovin moni muu ei noista koneista mitään ymmärrä – ei ainakaan niin hyvin kuin IBM. Mitä tulee räätälöintiin ja ohjelmiston päivityksiin, olet täysin sidottu IBM:ään, sillä ohjelmien koodi ei ole avointa. Ainoastaan IBM:n työntekijät pääsevät siihen käsiksi ja voivat tehdä siihen muutoksia.

Avoimen järjestelmän omistaja taas voi tilata palvelunsa mistä haluaa. Jos esimerkiksi päädyt käyttämään tietokonetta, jossa on Red Hat Linux, niin todennäköisesti haluat ostaa myös tukipalvelut Red Hatilta, sillä heidän luonnollisesti ovat oman tuotteen asiantuntijoita. Jos heidän palvelunsa ei kuitenkaan miellytä tai pidät sitä vaikka vain liian kalliina, niin voit kuitenkin halutessasi ostaa palvelun miltä tahansa muulta yritykseltä. Tarjokkaita kyllä löytyy – kuka tahansa viimeisen viiden vuoden aikana tietotekniikkaa opiskellut teekkari pystyy todennäköisesti auttamaan sinua. Räätelöinnit ja päivityksetkin onnistuvat, koska kaikilla on pääsy Linuxin lähdekoodiin.

Tällainen kilpailuasetelma on tietenkin asiakkaalle hyödyllinen. Mutta samalla se tuntuu myös järkevältä ja jollakin lailla luonnolliselta. On järkevää ostaa palvelua sieltä, mistä saa parasta palvelua. Kilpailutilanne myös pitää palvelun myyjän sopivasti varpaillaan. Asiakas ei ole sidottu samaan palveluntarjoajaan ikuisiksi ajoiksi vain siksi, että joskus on päädytty heidän järjestelmäänsä. Jos palvelun laatu laskee, voidaan palveluntarjoajaa aina vaihtaa. Myyjän on siis joka päivä ansaittava asiakkaansa uudelleen.

Palvelujen myynnissä on myös toinen ominaisuus, joka tuntuu järkevälle ja luonnolliselle. Kun asiakas ostaa palvelua, hän ostaa työtä. Sen sijaan kun asiakas ostaa Microsoftin CD-levyn – jota hän kaiken lisäksi ei tarvitse – Microsoft ei joudu tekemään yhtään sen enempää työtä kuin ennenkään. Microsoftille se on vain ilmaista rahaa. Palvelua voidaan laskuttaa ainakin jossain suhteessa siihen, kuinka paljon työntekeä toimeksianto aiheuttaa. Sen sijaan suljettu tietokoneohjelma hinnoitellaan täysin mielivaltaisesti ja yleensä asiakas on siinä pelissä heikoilla.

Mutta palvelujen myynti on myös myyjälle mielekäs vaihtoehto. Jos vertaillaan palvelujen myyntiä CD-levyjen myyntiin – oli kyseessä sitten suljetun ohjelmiston myyminen tai avoimen ohjelmiston myyminen "vedenmyyntimallilla" – niin palvelujen myynti tuottaa paljon pysyvämpiä asiakassuhteita ja sitä kautta pysyvämpiä rahavirtoja. Siinä missä tietokoneohjelman ostaminen on kertaluontoinen tapahtuma, on tukipalveluiden tarve tyypillisesti pysyvää. Palvelusopimukset saatetaan jopa tehdä suoraan kuukausiveloitteisiksi, jolloin rahavirta myyjäyrityksen kassaan on erittäin tasaista ja turvattua. Tällöin yrityksen ainoana veloitteena on asiakkaan pitäminen kaikin puolin tyytyväisenä.

Kertaluontoisiin myyntitapahtumiin toimintansa perustava yritys taas joutuu väistämättä hieman outoon suhteeseen asiakkaisiinsa nähden. Toisaalta yrityksen pitäisi luonnollisesti pyrkiä tarjoamaan asiakkailleen mahdollisimman hyvää tuotetta, mutta toisaalta tuotteen ei pitäisi olla niin täydellinen, etteikö vuoden kuluttua voitaisi myydä asiakkaille uutta ja vielä parempaa tuotetta. Tähän ristiriitaisuuteen törmäämme itse asiassa monilla muillakin aloilla, aina repeilevistä naisten sukkahousuista alkaen. Linux-yrityksille tämä ei toistaiseksi ole vielä ollut ongelma. Linux-maailma on viime vuosina kehittynyt niin nopeasti, että esimerkiksi Red Hat on kiireisimpinä vuosina ehtinyt julkaisemaan jopa kaksi uutta versiota Linux-jakelustaan. Sen sijaan Windows-maailma on sen verran kypsempi, että Microsoftin julkaisemat päivitykset – varsinkin Office-tuoteperheessä – ovat jo pitkään tuntuneet teennäiseltä rahanlypsämiseltä. Uusi versio pitäisi ostaa, mutta kukaan ei tiedä miksi. Ongelman tiedostaneena myös Microsoft on viime vuosina pyrkinyt siirtymään yhä enemmän kohti palvelumaisempaa mallia ja siihen liittyvää kiinteää vuosilaskutusta.

Jos vertaillaan "vedenmyyntimallia" ja "palvelumallia" toisiinsa, niin kumpaa mallia Linux-yritykset sitten suosivat? Tilanne on sikäli mielenkiintoinen, että jo 90-luvulta asti kaikki varteenotettavat Linux-yhtiöt ovat pyrkineet pääsemään palveluyrityksiksi, mutta aivan viime vuosiin asti suurin osa niiden tuloista on tullut värikkäisiin pahvilaatikkoihin pakattujen CD-levyjen myynnistä. Vuoden 2003 aikana tilanne kuitenkin selkeästi muuttui. Münchenin kaupungin kaltaiset isot urakat ovat tosissaan avaamassa palvelumarkkinoita. Samoin monien yritysten konehuoneissa pyöriviä vanhoja Unix-koneita päivitetään kovalla tohinalla Linuxille. Toisaalta yhä useammalla internetin käyttäjällä on nopea verkkoyhteys ja lisäksi käytössään polttava CD-asema, joten Linuxin lataaminen internetistä ja CD-levyjen tekeminen itse on yhä yleisempää, eikä CD-levyjen myynti siksi ole enää niin tärkeä bisnes kuin aiemmin.

Suurista Linux-yrityksistä Red Hat päätti kesällä 2003 yllättäen lopettaa halvan perus-Linuxinsa valmistamisen kokonaan, ja keskittyy tästedes pelkästään hyvin maksavien yritysasiakkaiden palvelemiseen, joita sillä hyvien tilinpäätösnumeroiden perusteella ilmeisesti löytyykin tarpeeksi. Red Hatin Enterprise Linux -tuotteita myydäänkin nykyään täysin vuosittaiseen tilausmaksuun perustuvalla mallilla, pahvilaatikkojen myynnistä on siis luovuttu kokonaan. SUSE taas päätyi tammikuussa 2004 vanhan ja kokeneen ohjelmistotalon Novellin ostamaksi, mitä kautta sillä on nyt tarjota huomattavasti laajempi ja kokenempi palveluverkosto kuin Red Hatilla. SUSE onkin tehnyt useita kauppvoja Saksan kaupunkien kanssa. Monen kotikäyttäjän suosima Mandrake taas on kannustanut käyttäjiään unohtamaan CD-levyjen ostamisen ja lataamaan ne ennemmin itse netistä, edellyttäen että käyttävät samalla vastaavan summan rahaa Mandrake Club -palveluun.

19Ja miksi et ostaisi? Halvimmathan maksavat vain \$21 645 dollaria, mutta todennäköisesti haluaisit kuitenkin ennemmin niitä kunnan puolen miljoonan dollarin tietokoneita. Mukaan kuuluu käyttöjärjestelmä ja siihen vuoden päivitykset, eli hieman palvelua.

29. MESENAATTIMALLI

(O'REILLY & LARRY WALL, TRANSMETA & LINUS, KDE)

Eräs tapa, jolla monet Open Source hakkerit ovat onnistuneet elättämään itsensä, muistuttaa monella lailla keskiajan ja renesanssiajan suhdetta taiteilijan ja hänen rikkaan suojelijansa – mesenaatin – välillä. Nykyajan vastaavuus olisi ehkä sponsorointi, mutta mesenaatista ja taiteilijasta puhuminen on kuvaavampaa. Ehkä paras esimerkki tällaisesta *mesenaattimallista* on suosituksen Perl ohjelmointikielen kehittäjän *Larry Wallin* suhde pitkäaikaiseen, mutta nyt jo entiseen työnantajaansa *O'Reilly & Associates* kirjakustantamoon. Voidaan oikeutetusti kysyä mitä Open Source -ohjelmointikielen luoja ja kehittäjä tekee täysipäiväisenä työntekijänä kirjakustantamossa?

O'Reilly on tullut tunnetuksi huipputason tietokoneoppaiden julkaisijana. Yhtiön tavaramerkkinä on käyttää kirjojensa kansien kuvituksena erilaisia eksoottisia eläimiä. Esimerkiksi Perliä opettavan "Programming Perl" -kirjan maskottina on kameli. Kamelikirja – The Camel Book – tunnetaan suorastaan Perl-ohjelmoijien raamattuna.

Kamelikirjasta tuli osa jokaisen Perl-ohjelmoijan pakkolukemistoa jo varhain 90-luvulla. Tässä vaiheessa O'Reilly-kustantamon pääjohtaja Tim O'Reilly ajatteli jokseenkin seuraavasti: Mitä enemmän Perl-ohjelmoijia, sitä enemmän myytyjä kamelikirjoja. Eli kaikki mikä on hyväksi Perille, on hyväksi O'Reillylle. Sanoista tekoihin, ja niinpä Tim otti Larry Wallin palkkalistoilleen työtehtävänä kehittää Perliä täyspäiväisesti ja siinä sivussa tietenkin uuden kamelikirjan painoksen kirjoittaminen. Molempia osapuolia hyödyttävän symbioottisen suhteen lisäksi Open Source -yhteisön tunnetun johtajahahmon palkkaaminen oli myös kustantamolle hyvää mainosta. O'Reillystä onkin tullut ei vain Perl-ohjelmoijien, vaan koko Open Source -yhteisön hovihankkija ohjelmointioppaiden ja muun tietokonekirjallisuuden markkinoilla.

Hieman samanlainen suhde oli myös *Linus Torvaldsilla* aiempaan työnantajaansa *Transmetaan*. Transmeta ei ole Linux- eikä Open Source -yhtiö, vaan valmistaa vähän virtaa kuluttavia tietokoneprosessoreita. Kun Linus valmistui Helsingin yliopistosta vuonna 1997, Transmeta palkkasi tämän Linux-maailman kiistattoman johtajan ja antoi Linusille samalla vapaat kädet käyttää työaikaansa Linuxin parissa puuhasteluun.

Vaikka Transmetalla ei ollut itse Linuxille mitään käyttöä, oli maailman tunnetuimman ohjelmoijan palkkaaminen kannattava veto jo pelkästään markkinointimielessä. Millään rahamäärällä ei olisi saanut ostettua yhtä tehokasta mainoskampanjaa. Lehdistön mielenkiintoa vielä lisäksi se, että Transmeta ei tuolloin ollut vielä julkaissut ensimmäistäkään tuotetta ja piti tulevaisuudensuunnitelmiaan visusti salassa. Varmasti ei voitu edes tietää, tekikö yhtiö ylipäättään prosessoreja vaiko jotain aivan muuta. Kun heidän ensimmäiset prosessorinsa sitten vihdoinkin valmistuivat, kaikki olivat jo niin haljeta uteliaisuudesta, että pelkän lehdistötöiläisyyden pitäminen oli etusivun uutinen kaikissa IT-alan lehdissä, eikä sen kummempaa mainoskampanjaa juuri tarvittu.

Linusin palkkaaminen ei kuitenkaan ollut Transmetalille pelkästään hyväntekeväisyyttä tai edes mainoskikka. Transmetan uudet prosessorit olivat toiminnallisilta periaatteiltaan yhteensopivia Intelin valmistamien x86-arkkitehtuurin prosessorien kanssa. Ja koska Linusin Linux-järjestelmä oli tehty juuri näille Intelin prosessoreille, hän oli luonnollisesti yksi parhaita x86-arkkitehtuurin asiantuntijoita maailmassa. Niinpä hänestä on varmaan ollut Transmetalille korvaamatonta apua monien kiperien pulmien ratkojana, vaikka hän omien sanojensa mukaan viettääkin yli puolet tyypillisestä työpäivästään harrastuksensa Linuxin parissa.

Maailman parhaalle ohjelmoijalle ei ehkä ole vaikeaa löytää halukkaita mesenaattiehdoikkaita. Mutta miten on tavallisen keskiverto tuntemattoman hakkerin kohdalla? Palkkaako joku hänetkin? Vaikka mesenaattimalli ehkä toimiikin parhaiten juuri tähtihakkerien kohdalla, niin Open Source -yhteisöstä löytyy myös rivijäseniä, jotka ovat onnistuneet elättämään itsensä vastaavalla tavalla.

Quanta Plus on KDE-ikkunointiympäristöön kuuluva HTML-editori, eli suomeksi sanottuna kotisivunteko-ohjelma. Ohjelman kaksi pääkehittäjää ovat amerikkalainen *Eric Laffoon* ja unkarilainen *Andras Mantia*. Eric kasvattaa ja myy päivätyönään kissanminttua – ilmeisesti kissat todella pitävät siitä niin paljon, että silläkin lailla voi joku itsensä elättää. Oman perheen elättämisen lisäksi Eric lähettää osan kissanminttutuloista Andrakselle, joka puolestaan ahkeroi täysipäiväisesti Quanta Plussan ja muiden KDE-ohjelmien parissa. Kaksikko on laskelmoinut, että tämä järjestely on tehokkaampi kuin jos molemmat yrittäisivät kehittää Quanta Plussaa päivätyönsä ohessa. Yhteistyö tietenkin perustuu myös osittain siihen elintasoeroon mikä Itä-Euroopan ja USA:n välillä vallitsee. Mutta maailmantaloudesta viis, Eric on todellinen mesenaatti, joka vapaa-aikaansa sijaan osallistuu KDE-projektiin rahallisesti ja käyttää sitten säästyneen ajan rahan tekemiseen kissanminttubisneksessään.

KDE on saanut paljon myönteistä julkisuutta myös Adopt-a-Geek²¹ nimisen ohjelmansa ansiosta. Adopt-a-Geek on KDE-aktivisti Scott Wheelerin aloittama projekti. Projekti sai alkunsa huomiosta, että monet ahkerat KDE-ohjelmoijat ovat opiskelijoita, itä-eurooppalaisia tai sitten muuten vaan köyhiä ja paremman puutteessa tekevät arvokasta vapaaehtoistyötä vanhoilla ja hitailla tietokoneilla. Adopt-a-Geek ohjelma pyrkii tukemaan näitä köyhimpiä KDE-ohjelmoijia hankkimalla heille tehokkaampia koneita, jotta he niiden avulla voisivat myös työskennellä tehokkaammin. Vaikka ohjelma välittää vain tietokoneita ja niiden osia eikä rahaa, on tämäkin yksi esimerkki siitä, miten KDE-ohjelmien käyttäjät voivat tukea ohjelmiansa tekijöitä heidän materialististen tarpeidensa tyydyttämisessä.

Tuomio: Mesenaattimalli, tai modernimmin *sponsorointi*, on osoittautunut käytännössä toimivaksi tavaksi hankkia leipää Open Source -hakkerin pöytään, mutta soveltuu ehkä kuitenkin paremmin kuuluisille hakkeritähdille kuin keskiverto rivikoodareille. Eettisesti malli saa luonnollisesti arvosanan 10+. Se ei ainoastaan läpäise eettistä tarkastelua, vaan suorastaan huokuu hakkerietiikkaa ja jakamisen iloa.

21<http://devel-home.kde.org/~wheeler/adopt-a-geek/>
Suomeksi: Adoptoi nörtti

30. MAKSETAAN TYÖNTEOSTA (SAKSA & KROUPWARE)

Mistäpä muusta sitä palkkaa maksettaisinkaan, jos ei työnteosta? Markkinatalouden koukerot ovat ihmeelliset, mutta kaikkien 2000-luvun taitteen teknobuumin aikana syntyneiden loistavien liikeideoiden keskeltä löytyy vieläkin ihmisiä, jotka elättävät itsensä ihan vaan rehellistä työtä tekemällä! Tämä itsestäänselvyydessään suorastaan sarkastiselta kuulostava idea on yksi esimerkki palvelumallista. Kun perinteisessä ohjelmistoteollisuudessa ensin tehdään tietokoneohjelma ja sitten myydään sitä värikkäisiin pahlavilaatikoihin pakattuna, niin "maksetaan työnteosta" -mallissa ohjelmia tehdäänkin tilaustöinä. Argumentointi on jokseenkin seuraavanlainen: Internet on täynnä loistavia Open Source -ohjelmia, joita voit vapaasti käyttää ihan ilmaiseksi. Jos kuitenkin tarvitset jotain mitä ei ole vielä olemassa, me voimme tehdä sen sinulle – maksua vastaan luonnollisesti.

Esimerkki tämän mallin onnistuneesta soveltamisesta oli Saksan valtion tilaama *Kroupware*-projekti. Kesällä 2002 Saksan valtio tilasi kolmelta KDE-projektilta läheisesti liittyvältä yritykseltä (saksalaiset Efrakon ja Intevation sekä ruotsalainen Klarälvdalens datakonsult) Linuxilla toimivan ryhmätyöpalvelimen, sekä siihen liittyvät asiakasohjelmat sekä Linuxille että Windowsille.

Ryhmätyöohjelmistolla tarkoitetaan sellaista sähköpostiohjelmaa, jossa sähköpostin lisäksi on muita toimistossa tarpeellisia toimintoja kuten kalenteri, osoitteisto ja mahdollisesti myös keskusteluryhmät. Tunnetuimpia yritysten käyttämiä ryhmätyöohjelmistoja ovat Microsoftin Exchange ja IBM:n Lotus Notes. Vaikka Linuxista löytyy monia sähköpostipalvelinohjelmia, ei tällaista monipuolisempaa ryhmätyöpalvelintä ollut kukaan vielä tullut tehneeksi. Mutta nyt Saksan valtio halusi siirtää ryhmätyötoimintonsa Open Source -maailmaan.

Tarjouskilpailun voittaneet kolme yritystä nimesivät projektin *Kroupware*ksi.²² Palvelinohjelmisto päätettiin rakentaa tunnettujen Open Source -ohjelmistojen varaan – sellaisien kuin Apachen *www*-palvelin, Postfix sähköpostipalvelin, OpenLDAP hakemistopalvelin ja IMP Webmail sähköpostin *www*-käyttöliittymä. Asiakasohjelma taas kasattiin kokoon olemassaolevista ja jo hyvin toimivista KDE-projektin sähköpostiohjelmasta, kalenterista ja osoitteistosta, joista pienellä lisäpanostuksella saatiin Microsoft Outlookia muistuttava kokonaisuus. Luonnollisesti joitain toimintoja piti myös tehdä ihan itse, mutta paljon jo tehtyä työtä pystyttiin tällä tavalla hyödyntämään suoraan. *Kroupware*-projekti valmistuikin tästä syystä ennätysajassa. *Kolab*-nimen lopulta saanut ryhmätyöohjelma valmistui tiukan aikataulun mukaisesti kesällä 2003, eli vain vuosi projektin aloittamisesta.

Näin monimutkaisen ohjelmiston teettäminen vuodessa on IT-maailmassa jokseenkin ennenkuulumatonta. Vielä ihmeellisempää on se, että työn tekivät kolme suhteellisen pientä ja tuntematonta konsulttiyhtiötä. Nopea työtahti oli mahdollista ainoastaan siksi, että projekti pystyi rakentamaan niin paljon jo aiemmin tehdyn laadukkaan Open Source -työn varaan.²³

Kroupware-projekti rikkoo mielenkiintoisella tavalla totuttuja markkinatalouden mekanismeja. Saksan valtio sai haluamansa ryhmätyötuotteen, mutta koska Kolab on Open Sourcea, kuka tahansa joka vastaisuudessa tarvitsee Linux-alustalle hyvää ryhmätyöpalvelintä, voi kopioida sen internetistä ilmaiseksi! Onko reilua, että Saksan valtio joutui yksin maksamaan ohjelmistosta, joita muut sitten käyttävät ilmaiseksi?

Äkkiä ajateltuna ajatus ei kuulosta reilulta. Mutta Open Source -maailmassa tämä malli on täysin luonnollinen. Open Source hakkerit ovat kautta aikain tehneet ohjelmia *omiin tarpeisiinsa*. Ei heitä ole kiinnostanut se, kuka muu heidän työstään mahdollisesti hyötyy tai ei hyödy, he ovat ratkoneet ensisijaisesti omia ongelmiaan. Saksan valtio toteutti tätä samaa logiikkaa asiakkaana. Ja tuskinpa heillä onkaan mitään syytä valittaa. Hehän saivat sen mitä halusivat. Saivat vieläpä laadukkaan ratkaisun halvalla ja ennätysmäisen nopeasti. Mikä tässä siis olikaan mielestäsi epäreilua?

Toisekseen täytyy muistaa, ettei Saksan valtio todellisuudessa maksanut Kolabia kokonaan itse. Suurin osa Kolabistahan koostuu jo olemassaolevista Open Source -ohjelmista, jotka sisältävät monen tuhannen miestyövuoden arvosta verta, hikeä ja kyyneleitä – tai ainakin suuret määrät innostunutta hakkerihenkeä. Tästä miljoonien, jos ei miljardien arvoista työstä Saksa ei maksa pfenniäkään – tai siis senttiäkään.

Tämän tosiasian muistaminen on tärkeää, jotta ymmärtäisimme miten Open Source parhaimmillaan toimii. Jos mustasukkaisesti tarkkailemme koko ajan vain sitä, miten muut mahdollisesti saavat jotakin meitä enemmän, tulee elämästämme hidasta ja kallista riitojen ja pattiitilanteiden selvittelyä. Jos sen sijaan jokainen keskitymme omien ongelmiamme ratkomiseen, kaikki voittavat.

Tuomio: Erittäin hyvin käytännössä toimivaksi osoittautunut malli, joka on täysin hakkerietiikan kanssa linjassa ja hyödyntää Open Source -mallin vahvuuksia. Mainitsemisen arvoista on myös, että kaikki aiemmin tarkastelemamme mallit nojaavat taloudellisesti johonkin muuhun kuin ohjelmointiin. Siis esimerkiksi niin, että päätoimena on konsultointi ja ohjelmointi on jonkinlainen sivuharrastus. Kroupware-projektissa sen sijaan ohjelmoijille maksettiin ohjelmoinnista, mikä tuntuu erittäin terveeltä ajatukselta! "Maksetaan työstä" -periaate vetoaa jokaisen ihmisen maalaisjärkeen, varsinkin teknobuumin synnyttämien paperimiljonäärien jälkeen.

Kroupware-projekti jättää kuitenkin avoimeksi joitakin kysymyksiä. Saksan valtio on tarpeeksi suuri asiakas, jotta se pystyy yksin rahoittamaan tällaisen tilaustyön. Mutta miten käy pienempien asiakkaiden? Jäävätkö pienyritykset ja kuluttajat vain nuolemaan suuryritysten ja valtioiden pöydiltä tippuvia murusia vai voisiko sama malli toimia myös pienemmässä mittakaavassa? Näitä kysymyksiä pohdimme seuraavaksi.

22Englanniksi ryhmätyöohjelmistoja kutsutaan nimellä *groupware*. KDE-projektissa taas on tapana nimetä kaikki ohjelmat K-kirjaimella alkaviksi. Alun perin hauska juttu on alkanut vuosien mittaan tuntumaan yhä mielikuvituksettomammalta.

23Hakkerit siteeraavatkin usein mielellään Isaac Newtonia, joka kirjoitti:
"If I see further, it is because I stand on the shoulders of giants." (Jos minä olen ollut kaukonäköinen, se johtuu siitä, että olen seissyt jättiläisten olkapäillä.)

31. STEPHEN KING, JUOMARAHAMALLI JA TYÖNVÄLITYSPÖRSSI (STEPHEN KING, ROGER WILLIAMS, SOURCEFORGE, KOLAB, JBOSS)

Maailmankuulu kauhukirjailija *Stephen King* teki vuonna 2000 joitakin mielenkiintoisia kokeiluja internet-julkaisemisen saralla. Vaikka nämä eivät mitään Open Source -projekteja olleetkaan, niin King selvästi osoitti ymmärtävänsä aineettoman työn ja internetin välistä dynamiikkaa niin hyvin, että mekin voimme ottaa hänestä oppia.

King teki historiaa julkaisemalla vuoden 2000 keväällä romaaninsa "Riding the Bullet" ainoastaan digitaalisesti. Tämä oli ensimmäinen kerta kun tunnettu kirjailija oli kirjoittanut kirjan, jota ei tultais lainkaan julkaisemaan paperilla. (Tosin kolme vuotta myöhemmin tarina julkaistiin suurista puheista huolimatta paperille painettuna kokoelmassa *Everything's eventual*.) Erialaisten eBook-laitteiden ja -ohjelmien myyjät ottivat tilaisuudesta kaiken mainosarvon irti, ja vanhoista uutisartikkeleista voi vieläkin lukea tarinoita "painetun kirjan kuolemasta".

Digitaalisen kirjan tuotantokustannukset ovat luonnollisesti pienemmät kuin paperisen kirjan. Tästä ja myös markkinoinnillisista syistä kirjaa myytiin niinkin halvalla kuin 2,50\$. Ensimmäisen vuorokauden aikana kirjaa myytiin huomattavasti enemmän kuin 400 000 kappaletta. En tiedä minkälaisiin myyntimääriin King on tottunut, mutta ainakin lehdistö piti tätä suurena menestyksenä ja eBook-valmistajat todisteena – niin tosiaan – "painetun kirjan kuolemasta".

King ei ilmeisesti kuitenkaan ollut täysin tyytyväinen kokeiluun. Siitä huolimatta, että julkaisun käyttämissä eBook-ohjelmissa oli käytetty lapsellisen yksinkertaisia kopionestomenetelmiä, krakkerit onnistuivat nopeasti murtaamaan suojaukset ja verkossa levisi pian piraattiversioita kirjasta. Vaikka myös verkkokaupat Amazon.com ja Barnes&Noble olivat siinä vaiheessa siirtyneet mainostarkoituksessa jakamaan kirjaa ilmaiseksi omilta sivuiltaan, piraattiversioiden ilmestyminen tuntui harmittavan Kingiä suunnattomasti.

Myöskään asiakkaat eivät olleet eri eBook-formaatteihin kovinkaan tyytyväisiä. Samaiseen kopioinninestojätteluun liittyi, että kirjan tulostaminen paperille ei ollut eBook-ohjelmista mahdollista. Kirjaa oli siis pakko lukea tietokoneen ruudulta ja tämän takia jännittävästä tarinasta ei pystynyt nauttimaan lempiasennossaan sohvalta tai piilossa peiton alla. Kopioinninestotekniikkaan liittyikin tältä osin jonkin verran ironiaa. Piraattiversioita nimittäin luonnollisesti pystyi tulostamaan, eli ne olivat siinä mielessä asiakasystävällisempiä kuin viralliset eBook-versiot.

eBook-hype on sittemmin laantunut ja ennustuksista huolimatta painettu kirja ei kuollut. Romaaneja kun on sittenkin kivempi lukea paperilta kuin tietokoneen ruudulta. Monet puuhastelevat kuitenkin edelleen toimivampien kopioinninestojärjestelmien keksimiseksi. He ilmeisesti uskovat edelleen, että joku päivä ihmiset suostuvat ostamaan kirjoja, joita eivät voi lukea lempiasennossaan sohvalta.

Pettymyksestä huolimatta Kingiltä löytyi kuitenkin vielä sen verran pioneerihenkisyttä, että hän jo samana kesänä toteutti toisen internet-kokeilun. Tällä kertaa hän julkaisikin tuotoksensa suoraan omalla kotisivullaan ohittaen täysin kaikki kustantamot. Sattumalta "The Plant"-niminen tarina enteellisesti kertoi tappavasta vampyyrikasvista, joka piti tarinan kustantamon väkeä kauhun vallassa. Stephen Kingin kokeilut taas herättivät kauhua ihan oikeissa kustantamoissa. Tulisiko koe osoittamaan, että kustantamoja ei enää tarvita?

eBook-seikkailuihin pettyneenä King oli päättänyt kääntää marssijärjestyksen pääläelleen. Hän julkaisi tarinan kaksi ensimmäistä lukua internetissä. Tekstiä ei nyt oltu sotkettu monimutkaisten kopiosuojauksen taakse, vaan kyseessä oli ihan tavallinen verkossa julkaistava teksti. Teksti ei kuitenkaan ollut ilmaista, vaan lukijoita kehoitettiin maksamaan yksi dollari per luku. Jos tarpeeksi moni lukija maksaisi, King lupasi kirjoittaa tarinan loppuun luku kerrallaan.

Kingin idea noudattaa aika tarkkaan edellisessä kappaleessa kuvattua Kroupware-projektin mallia. Hän hyväksyi sen tosiasian, että monet tulisivat lukemaan tarinan ilmaiseksi. Sillä ei ollut hänen kannaltaan väliä, niin kauan kuin löytyisi myös tarpeeksi maksajia. Jos joku maksaa, niin Stephen tekee työtä.

Kokeilu ei saanut yhtä hyvää vastaanottoa kuin keväinen eBook-kokeilu, mutta ensimmäisen viikon aikana 152 132 innokasta fania oli kuitenkin käynyt lukemassa ensimmäisen luvun ja heistä 116 200 oli dollarinsa maksanut. Tämä vastasi Kingin ennakkoon asettamia ehtoja, joten hän omaksi yllätykseksensä joutuikin jatkamaan kustantamo-kauhutarinaa.

Kuudenteen lukuun tultaessa innostus oli kuitenkin laskenut. Edellistä lukua oli käyty lukemassa vain 40 000 kertaa ja näistäkin yli puolet oli jättänyt laskunsa maksamatta. Niinpä King päätti yllättäen luopua hyvin alkaneesta projektista.

The Plantin epäonnistuminen on luonnollisesti pienoinen pettymys avointa verkkoyhteiskuntaa havitteleville. Jos Kingin kaltainen suosikkikirjailija ei pysty elättämään itseään internet-julkaisulla, niin kuka sitten?

Pienoista kritiikkiä voi kuitenkin kohdistaa myös Kingiä kohtaan. Projekti olisi mahdollisesti voinut menestyä paremminkin, jos King olisi ymmärtänyt Open Source -mekanismeja. Nythän hänen kokeilunsa vain sattumalta muistutti Open Source -yhteisön käytäntöjä, hän ei todennäköisesti itse tuntenut eikä ollut ottanut oppia Linuxista ja sen ympärillä kukoistavasta liiketoiminnasta. Esimerkiksi Kingin sivuilla kutsuttiin varkaiksi niitä, jotka lukivat tarinaa maksamatta. Tällainen omien lukijoiden haukkuminen ei ehkä markkinointimielessä ollut kovinkaan järkevää. Kokeilusta saa myös sellaisen kuvan, että Kingiä ei niinkään haitannut se, että maksavien lukijoiden määrä pieneni, vaan pikemminkin se, että ilmaiseksi lukevien osuus kasvoi. Jostain käsittämättömästä syystä kirjan ensimmäinen luku jossain vaiheessa myös poistettiin verkosta, mikä käytännössä takasi sen, että uusia lukijoita ei ainakaan enää kirjalle tulisi. Tietty avoimuuden puute siis on projektista aistittavissa, ja on luonnollista, että tämä on saattanut puolestaan vaikuttaa lukijoihin siten, että hekin suhtautuivat kokeiluun varauksella.

Kaikin puolin tuntuu siltä, että King ei ihan täysin ollut alun perinkään sitoutunut koko projektiin. Itse asiassa projektin aloittaneet kaksi ensimmäistä lukua hän oli kirjoittanut jo parikymmentä vuotta aiemmin. Hän ilmeisesti kaivoi tämän julkaisematta ja viimeistelemättä jääneen tekeleen jostain ullakolta ja laittoi sen internetiin juuri kokeena. Kun lukijat yllättäen sitten maksoivatkin tekstistä, hän jaksoi kirjoittaa neljä lukua lisää, mutta siihen kiinnostus sitten lopahtikin.

Jos vielä hetken jatkamme tätä jälkiviisastelua, niin projekti olisi ehkä onnistunut paremmin, jos kirjan julkaisemista ei olisi pilkottu niin moneen pieneen erään. On tietenkin selvää, että alkuinnostuksen jälkeen maksavien lukijoiden määrä on pelkästään laskeva, varsinkin kun kirjan alku oli poistettu verkosta. Olisikin varmaan ollut onnistuneempi ratkaisu pyytää asiakkaita maksamaan heti alkuun hieman suurempia summia kahdessa tai korkeintaan kolmessa erässä.

Oli miten oli, ainakin yhden asian voimme Kingiltä oppia. Jos ikinä kirjoitat kauhukirjallisuutta, älä ikinä myy lukijoillesi puolikasta kirjaa. Lopettamis päätöksestä suivaantuneiden lukijoiden kritiikki nimittäin oli murskaavaa. Jostain syystä King ei edes osannut ennakoida, että kymmenet tuhannet lukijat, jotka siihen mennessä olivat maksaneet jo seitsemän dollaria kirjanpuolikkaastaan, suuttuisivat kun eivät saakaan lukea jännittävää tarinaa loppuun. Varsinkin kun King oli kesällä nimenomaan luvannut, että jos kaksi ensimmäistä lukua saavat jatkoa, hän ei sen jälkeen jätä kirjaa kesken. Jos The Plant -projektin kokemukset eivät muuten kovin rohkaisevia olekaan, niin opimme ainakin jotain: Jos petät asiakkaitasi, he yleensä suuttuvat.

Toinen mielenkiintoinen internet-julkaisemisen kokeilu on *Roger Williamsin* kirjoittama kahdeksanlukuinen "The Metamorphosis of Prime Intellect". Päinvastoin kuin King, Williams on tuntematon esikoiskirjailija, joka ei yrityksistä huolimatta onnistunut saamaan romaanilleen kustannussopimusta. Niinpä hän päätti julkaista romaaninsa internetissä. Vaikka myöskään tämä romaani ei teknisesti ottaen täytä Open Source -ideologian vaatimuksia, oli Williamsin lähestymistapa internet-julkaisemiseen paljon myönteisempi ja avoimempi kuin Kingin. Romaania sai vapaasti lukea ja kopioida ja se siis julkaistiin heti kokonaisuudessaan. Niille, jotka pitivät romaanista, Williams tarjosi mahdollisuutta maksaa "tippiä" eli juomarahaa PayPal-palvelun kautta.

Williams teki myöhemmin tiliä kokeilun onnistumisesta artikkelissa "The Tip Jar as Revenue Model: A Real-World Experiment"²⁴ (Juomarajat tulonlähteenä: Tosielämän kokeilu). Hän ei ainoastaan tutki asian taloudellisia puolia, vaan toteaa myös, että hänen kaltaiselleen tuntemattomalle kirjailijalle jo pelkästään lukijoiden saaminen on suuri asia sinänsä. Mutta miten siis romaanille kävi?

Williamsin tilastojen mukaan romaanin luki loppuun noin 5000 – 10 000 ihmistä. Ei siis ollenkaan huonompi tulos tuntemattomalle esikoiskirjailijalle! Juomarahojakin kertyi ihan mukavasti, 760\$ puhtaana käteen. Vaikka Williams olikin tyytyväinen menestykseen, hän totesi lopuksi aivan oikein: "Tällä ei kuitenkaan makseta vuokraa".

Hieman vastaavia kokeiluja tehtiin myös Open Source -maailmassa, siis tietokoneohjelmiin liittyen, 2000-luvun taitteessa. Mutta mitäpä silloin ei olisi kokeiltu! Parikin yritystä yritti tuoda yhteen Open Source -ohjelmien tekijöitä ja käyttäjiä eräänlaiseen *työnvälityspörssiin*. Ohjelmoijat saivat sivuilla kertoa, minkälaista ohjelmaa olivat parhaillaan tekemässä, ja projektista kiinnostuneet kuluttajat puolestaan pystyivät sivuston kautta tukemaan ohjelmoijaa rahallisesti. Jos taas kuluttaja oli kiinnostunut "tilaamaan" jonkun ohjelman, hän pystyisi sivuston kautta tarjoamaan siitä mielestään sopivan summan rahaa. Myös muut samanlaisesta ohjelmasta kiinnostuneet "ostajat" voisivat tämän jälkeen sitoutua projektiin omasta puolestaan sopivalla rahasummalla. Ja kun kaikki tarjoukset ynnättäisiin yhteen, saataisiinkin jo sievoinen summa rahaa, joka oli tarjolla työtä hakevalle ohjelmoijalle sitten kun haluttu ohjelma jonain päivänä valmistuu.

Yksikään näistä projekteista ei päässyt alkua pidemmälle.²⁵ Voi olla, että ne olivat myöskin aikaansa edellä, mutta todennäköisesti ajatusmalli yksinkertaisesti on väärä. Työnvälityspörssit eivät ensinkään tuntuneet kiinnostavan hakkeriyhteisöä. Hakkereilla oli nimittäin kiire työskennellä niiden ohjelmien parissa, joita he jo olivat tekemässä. He tekivät mieluummin korvauksetta jotain, mikä heitä kiinnosti, kuin rahasta jotain, mikä ei kiinnostanut.

Vuoden 2004 alussa maailman ylivoimaisesti suosituin Open Source -projektien isännöintipalvelu *SourceForge*²⁶ ryhtyi tarjoamaan palvelua käyttäville ohjelmoijille mahdollisuutta vastaanottaa rahalahjoituksia PayPal-palvelun kautta. Ideana tietenkin on, että osa kaikista juomarahoista menee provisiona SourceForgelle itselleen.

SourceForgen tapa rahoittaa Open Source -projektien ohjelmoijia kuitenkin eroaa epäonnistuneista työnvälityspörssistä. Sopivampi nimitys onkin juuri *juomarahamalli*. Ero on siinä, että SourceForgen ensisijainen tarkoitus ei ole toimia rahanvälityspörssinä, vaan sen varsinainen tehtävä on tarjota ohjelmoijille heidän työssään tarvitsemia työkaluja sekä levityskanavan heidän ohjelmilleen. Mahdollisuus tukea lempiohjelmansa tekijää pienellä lahjoituksella on tämän päälle vain lisä, ei toiminnan keskipiste.

SourceForgen menestys verrattuna työnvälityspörssien epäonnistumiseen on hyvä opetus meille kaikille, varsinkin tuleville IT-alan yrittäjille. Työnvälityspörssissä tärkein kysymys oli raha, jonka jälkeen vasta päästiin puhumaan ohjelmoinnista. SourceForgessa taas uusia projekteja aloitetaan joka päivä, eikä kukaan puhu rahasta mitään. Menestyneille projekteille juomarahoja sitten kertyy vähitellen, kenelle enemmän kenelle vähemmän. Sama kuvio on tuttu IT-alalta muutenkin, on niitä joilla on hienot liiketoimintasuunnitelmat ja sitten on niitä jotka tekee työtä.

SourceForgen juomarahoille kuitenkin pätee varmasti sama sääntö kuin Williamsin juomarahakokeilulle. Rahaa tulee ihan kivasti, mutta yksittäisillä juomarahoilla ei kuitenkaan elätä itseään eikä ainakaan perhettä. Onko siis todellakin niin, että "maksetaan työstä" ei toimi pienessä mittakaavassa, jos vaatimuksena on, että työstä saatava maksu vastaisi ihan oikeaa palkkaa?

Joitakin esimerkkejä on, missä "maksetaan työstä" mallia on onnistuneesti sovellettu myös Saksan valtiota pienempien asiakkaiden kanssa. Jotkut yritykset ovat jo ehtineet kysellä vastavalmistuneen *Kolab*-projektin perään toivoen siihen toteutettavaksi ominaisuuksia, joita Saksan valtio ei tullut tilanneeksi. *Kolab*-projektin alunperin toteuttaneet yritykset ovat nyt keränneet nämä kyselijät yhteen ja kiinnostuneet ovat päättäneet rahoittaa yhteisesti projektin seuraavaa versiota *Kolab2*:ta. Tämä on mahdollisesti ensimmäinen kerta, kun Open Source -projekti toteutetaan useamman pienemmän asiakkaan yhteisellä "kimppatilauksella".

Hieman samalla lailla toimii *JBoss*-nimistä Java-sovelluspalvelinta kehittävä *JBoss*-niminen yritys. Java-sovelluspalvelin sinänsä on iso ja monimutkainen ohjelmisto, eikä voida ajatella, että monikaan yritys olisi ollut valmis maksamaan sen toteuttamisesta kokonaisuudessaan. Näin ei ole tehtykään, vaan *JBoss* on ollut Open Source -periaatteella saatavana internetistä jo vuosia, kehittyen pikku hiljaa koko ajan. Yleensä *JBoss*ia käyttävä asiakas on siihen sellaisenaan tyytyväinen. Mutta aina välillä joku kaipaa siihen jotain pientä lisäominaisuutta. Tällöin he voivat tilata tämän uuden ominaisuuden toteutuksen esimerkiksi *JBoss*-yritykseltä (tai vaihtoehtoisesti teettää sen myös jollain toisella ohjelmointiyrityksellä, sillä koodihan on avoin) ja näin *JBoss*-sovelluspalvelin kehittyy hitaasti mutta varmasti, pienin askelin. Jos siis kauhukirjallisuutta ei pystynytään myymään pieninä palasina, niin näyttäisi siltä, että Java-sovelluspalvelimia voi.

Vaikka "maksetaan työstä" -malli siis ei tunnu toimivan ihan kuluttajatasolle vietyinä, näyttäisi kuitenkin siltä, että asiakkaana ei tarvitse olla Saksan valtion kokoinen talousmahti, jotta sitä voisi kuitenkin soveltaa onnistuneesti.

Tuomio: "Maksetaan työstä" -malli on kaunis ajatus, mutta toistaiseksi ei ole näyttöä, että se toimisi myös yksityishenkilöille ja muille pienille kuluttajille tarkoitetuissa tuotteissa. Keskisuurista yrityksistä ylöspäin malli kuitenkin näyttäisi toimivan erittäin hyvin.

24<http://www.kuro5hin.org/story/2003/4/27/195833/305>

25Itse asiassa ne epäonnistuivat niin täydellisesti, että kukaan ei tunnu enää edes muistavan minkä nimisiä ne lyhyen elinkaarensa aikana mahtoivat olla, mistä johtuen tämä analyysi joudutaan kertomaan vain nimettömänä anekdoottina.

26<http://sourceforge.net/>

32. KAHDET LISENSSIT (MYSQL, TROLLTECH QT)

Mahdollisesti taloudellisesti menestyksekkäin strategia Open Source -ohjelmistoissa on ollut niin kutsuttu *kaksoislisensointi*. Kaksoislisensoinnin mallia ovat soveltaneet varsinkin ns. "matalan tason" työkaluluonteisia ohjelmistoja valmistavat yritykset.

MySQL Ab on ruotsalainen yritys, joka tunnetaan MySQL-nimisestä tietokantaohjelmastaan. MySQL-tietokanta on erittäin suosittu varsinkin Linux-alustalla toimivien www-sivustojen tietovarastona. MySQL-tietokantaa voi ladata ilmaiseksi verkosta saman GPL-lisenssin käyttöehdoin kuin Linuxiakin. Vaihtoehtoisesti maksullisen version voi lisensoida 500 dollarin hintaan. Mutta miksi kukaan maksaisi 500 dollaria jostain, jonka voisi saada ilmaiseksikin?

Maksullista versiota käyttävät ne, jotka eivät halua itse sitoutua Open Sourcen ehtoihin. GPL-lisenssi nimittäin sisältää vaatimuksen, että ohjelmoija joka käyttää omassa ohjelmassaan jotain GPL-lisensoitua koodia tai ohjelmaa, on velvollinen julkaisemaan myös oman työnsä tulokset GPL-lisenssin ehtoilla – lähdekoodineen kaikkineen.²⁷ Tämän takia ne ohjelmoijat, jotka tekevät suljettuja ohjelmia, eivät voi käyttää MySQL:n ilmaista Open Source -versiota, vaan joutuvat pulittamaan 500 dollaria niin sanotusta "kaupallisesta lisenssistä". Yleensä nämä asiakkaat ovat myös verraten tyytyväisiä ostokseensa, sillä 500 dollaria on edullinen hinta laadukkaasta tietokantatuotteesta.

MySQL:lle tämä järjestely on mitä ilmeisimmin ollut loistava. Ilmaisen GPL-version olemassaolo takaa sen, että MySQL on yksi maailman suosituimmista tietokannoista. Tunnettuus on luonnollisesti tärkeää mille tahansa tuotteelle, mutta tietokoneohjelmalle on erityisen tärkeää, että ihmiset oppivat käyttämään ohjelmaa. Jos ohjelmistotalon työntekijä on aiemmin käyttänyt MySQL:ää ilmaiseksi kotipaikkakuntansa kennelyhdistyksen kotisivujen tietovarastona, hän todennäköisesti haluaa käyttää sitä myös yrityksensä suljetun ohjelmistoprojektin osana sen sijaan, että joutuisi opettelemaan jonkun muun tietokantatuotteen käyttöä.

GPL-version olemassaolo on myös vauhdittanut tietokannan kehittymistä. Open Source -periaatteiden mukaisesti myös monet MySQL-yhtiön ulkopuoliset ohjelmoijat ovat parannelleet tietokantaa ja näin yhtiön myymä tuote paranee "ihan itsestään". Tällä lailla hakkeriyhteisö ikään kuin maksaa ilmaisella työnteolla ja käyttäjäpalautteen antamisella sen, minkä suljettujen ohjelmien valmistajat maksavat rahassa.

Samanlaista mallia on soveltanut myös *Trolltech*, joka on suosittu Qt-käyttöliittymäkirjaston kehittäjä. Käyttöliittymäkirjaston tehtäviin kuuluu tarjota ohjelmoijalle tietokoneohjelman perustavaa laatua olevia rakennuspalikoita. Sellaiset asiat kuin painonappi, valikko, tekstikenttä ja vierityspalkki ovat kaikki käyttöliittymäkirjaston osia. Näistä palikoista ohjelmoija sitten kokoaa varsinaisen tietokoneohjelman.

Myös Qt:n vahvuutena on, että sillä voi tehdä ohjelmia sekä Windowsille että Linuxille. Se on siis hyvä valinta – ja yksi harvoja vaihtoehtoja – sellaiselle ohjelmoijalle, joka haluaa että hänen ohjelmansa toimii mahdollisimman monella eri käyttöjärjestelmällä.

Qt-kirjastoa käytetään Linuxissa suositun KDE-työpöytäympäristön perustana. KDE:n mukana käytetään Qt-kirjaston versiota, joka on julkaistu GPL-lisenssin ehdoilla. Mutta samoin kuin MySQL, myös Trolltech myy Qt:sta erillisiä lisenssejä, joita voi käyttää suljettujen ohjelmien rakennuspalikoina. Järjestelyn edut ovat samat kuin MySQL:lläkin. GPL-versio tuo tunnettua ja KDE-yhteisön ohjelmoijat ovat tärkeä voimavara myös Qt:n kehityksessä. He antavat palautetta ja usein myös kehittävät Qt:tä edelleen. Nämä parannukset taas hyödyntävät myös Qt:n kaupallista versiota.

Kaksoislisensointi herättää yhden mielenkiintoisen eettisluontoisen kysymyksen. Miten nämä yhtiöt pärjäisivät maailmassa, joka olisi 100% Open Source? Kysymys ei todellakaan ole ajankohtainen, mutta se on relevantti niille, jotka uskovat, että Open Source -malli on fundamentaalisesti parempi kuin suljettujen ohjelmistojen "pihtaamismalli". Jos suljettujen ohjelmistojen malli on niin huono, että kenenkään ei tulisi sitä käyttää, niin miten käy silloin näille yhtiöille, jotka saavat tulonsa suljettujen ohjelmien tekijöiltä? Ja voimmeko avoimuuden kannattajina ja pihtaamisen vastustajina silloin suositella tällaista liiketoimintamallia kenellekään?

Kysymykseen ei tunnu toistaiseksi olevan hyvää vastausta. Totuus on, että kaksoislisensointi on ainakin tällä hetkellä erittäin toimiva malli, joka tuntuu tyydyttävän kaikkia osapuolia. MySQL ja Trolltech saavat siitä leipänsä ja vieläpä voita leivän päälle. Suljettujen ohjelmien tekijät saavat laadukkaita työkaluja edullisesti. Ja Open Source -ohjelmoijat saavat samat työkalut heitä tyydyttävillä ehdoilla. Vieläpä niin, että Open Source -ohjelmien kehitys osittain kustannetaan suljettujen ohjelmien tekijöiltä, eli "vihollisleiriltä" saaduin varoin.

Kaikki siis ovat tyytyväisiä, ja kysymys jää avoimeksi. Vastaus ei tunnu kiinnostavan ketään.

Tuomio: MySQL:n ja Trolltechin käyttämä malli on mahdollisesti menestynein Open Source -maailman liiketoimintamalli. Sitä ovat heidän lisäksi käyttäneet myös monet muut saman tyyppiset yritykset. Malli ei kuitenkaan sovellu kaikkiin tapauksiin. Erityisesti se ei toimi loppukuluttajille tarkoitetuissa tietokoneohjelmissa, vaan kaikki mallia soveltaneet yritykset myyvät tuotteita toisille ohjelmoijille: ohjelmointikirjastoja tai tietokantoja. Malli läpäisee myös hakkerieettisen tarkastelun, vaikkakin avoimeksi jää kysymys siitä, miten nämä yhtiöt pärjäisivät puhtaassa Open Source -taloudessa.

27Tässä kohden on huomautettava, että kaikki Open Source ohjelmat eivät sisällä yhtä tiukkoja vaatimuksia. Monia Open Source -ohjelmia tai niiden osia voi myös hyödyntää suljetuissa ohjelmistoissa. Mutta GPL:n alaisten ohjelmistojen, kuten Linuxin kernelin ja MySQL:n kanssa sellainen ei ole sallittua.

33. VÄHÄN MOLEMPIA (XIMIAN EVOLUTION, CODEWEAVERS, TRANSGAMING)

Mahdollisesti taloudellisesti menestyksekkäin strategia Open Source -ohjelmistoissa on ollut niin kutsuttu *kaksoislisensointi*.

Kaksoislisensoinnin mallia ovat soveltaneet varsinkin ns. "matalan tason" työkaluluonteisia ohjelmistoja valmistavat yritykset.

MySQL Ab on ruotsalainen yritys, joka tunnetaan MySQL-nimisestä tietokantaohjelmastaan. MySQL-tietokanta on erittäin suosittu varsinkin Linux-alustalla toimivien www-sivustojen tietovarastona. MySQL-tietokantaa voi ladata ilmaiseksi verkosta saman GPL-lisenssin käyttöehdoin kuin Linuxiakin. Vaihtoehtoisesti maksullisen version voi lisensoida 500 dollarin hintaan. Mutta miksi kukaan maksaisi 500 dollaria jostain, jonka voisi saada ilmaiseksikin?

Maksullista versiota käyttävät ne, jotka eivät halua itse sitoutua Open Sourcen ehtoihin. GPL-lisenssi nimittäin sisältää vaatimuksen, että ohjelmioija joka käyttää omissa ohjelmassaan jotain GPL-lisenssoitua koodia tai ohjelmaa, on velvollinen julkaisemaan myös oman työnsä tulokset GPL-lisenssin ehtoilla – lähdekoodineen kaikkineen.²⁷ Tämän takia ne ohjelmoijat, jotka tekevät suljettuja ohjelmia, eivät voi käyttää MySQL:n ilmaista Open Source -versiota, vaan joutuvat pulittamaan 500 dollaria niin sanotusta "kaupallisesta lisenssistä". Yleensä nämä asiakkaat ovat myös verraten tyytyväisiä ostokseensa, sillä 500 dollaria on edullinen hinta laadukkaasta tietokantatuotteesta.

MySQL:lle tämä järjestely on mitä ilmeisimmin ollut loistava. Ilmaisen GPL-version olemassaolo takaa sen, että MySQL on yksi maailman suosituimmista tietokannoista. Tunnettuus on luonnollisesti tärkeää mille tahansa tuotteelle, mutta tietokoneohjelmalle on erityisen tärkeää, että ihmiset oppivat käyttämään ohjelmaa. Jos ohjelmistotalon työntekijä on aiemmin käyttänyt MySQL:ää ilmaiseksi kotipaikkakuntansa kennelyhdistyksen kotisivujen tietovarastona, hän todennäköisesti haluaa käyttää sitä myös yrityksensä suljetun ohjelmistoprojektin osana sen sijaan, että joutuisi opettelemaan jonkun muun tietokantatuotteen käyttöä.

GPL-version olemassaolo on myös vauhdittanut tietokannan kehittymistä. Open Source -periaatteiden mukaisesti myös monet MySQL-yhtiön ulkopuoliset ohjelmoijat ovat parannelleet tietokantaa ja näin yhtiön myymä tuote paranee "ihan itsestään". Tällä lailla hakkeriyhteisö ikään kuin maksaa ilmaisella työnteolla ja käyttäjäpalautteen antamisella sen, minkä suljettujen ohjelmien valmistajat maksavat rahassa.

Samanlaista mallia on soveltanut myös *Trolltech*, joka on suosittu Qt-käyttöliittymäkirjaston kehittäjä. Käyttöliittymäkirjaston tehtäviin kuuluu tarjota ohjelmoijalle tietokoneohjelman perustavaa laatua olevia rakennuspalikoita. Sellaiset asiat kuin painonappi, valikko, tekstikenttä ja vierityspalkki ovat kaikki käyttöliittymäkirjaston osia. Näistä palikoista ohjelmioija sitten kokoaa varsinaisen tietokoneohjelman.

Myös Qt:n vahvuutena on, että sillä voi tehdä ohjelmia sekä Windowsille että Linuxille. Se on siis hyvä valinta – ja yksi harvoja vaihtoehtoja – sellaiselle ohjelmoijalle, joka haluaa että hänen ohjelmansa toimii mahdollisimman monella eri käyttöjärjestelmällä.

Qt-kirjastoa käytetään Linuxissa suositun KDE-työpöytäympäristön perustana. KDE:n mukana käytetään Qt-kirjaston versiota, joka on julkaistu GPL-lisenssin ehtoilla. Mutta samoin kuin MySQL, myös Trolltech myy Qt:sta erillisiä lisenssejä, joita voi käyttää suljettujen ohjelmien rakennuspalikoina. Järjestelyn edut ovat samat kuin MySQL:lläkin. GPL-versio tuo tunnettuutta ja KDE-yhteisön ohjelmoijat ovat tärkeä voimavara myös Qt:n kehityksessä. He antavat palautetta ja usein myös kehittävät Qt:tä edelleen. Nämä parannukset taas hyödyntävät myös Qt:n kaupallista versiota.

Kaksoilisenointi herättää yhden mielenkiintoisen eettisluontoisen kysymyksen. Miten nämä yhtiöt pärjäisivät maailmassa, joka olisi 100% Open Source? Kysymys ei todellakaan ole ajankohtainen, mutta se on relevantti niille, jotka uskovat, että Open Source -malli on fundamentaalisesti parempi kuin suljettujen ohjelmistojen "pihtaamismalli". Jos suljettujen ohjelmistojen malli on niin huono, että kenenkään ei tulisi sitä käyttää, niin miten käy silloin näille yhtiöille, jotka saavat tulonsa suljettujen ohjelmien tekijöiltä? Ja voimmeko avoimuuden kannattajina ja pihtaamisen vastustajina silloin suositella tällaista liiketoimintamallia kenellekään?

Kysymykseen ei tunnu toistaiseksi olevan hyvää vastausta. Totuus on, että kaksoilisenointi on ainakin tällä hetkellä erittäin toimiva malli, joka tuntuu tyydyttävän kaikkia osapuolia. MySQL ja Trolltech saavat siitä leipänsä ja vieläpä voita leivän päälle. Suljettujen ohjelmien tekijät saavat laadukkaita työkaluja edullisesti. Ja Open Source -ohjelmoijat saavat samat työkalut heitä tyydyttävillä ehtoilla. Vieläpä niin, että Open Source -ohjelmien kehitys osittain kustannetaan suljettujen ohjelmien tekijöiltä, eli "vihollisleiriltä" saadiin varoin.

Kaikki siis ovat tyytyväisiä, ja kysymys jää avoimeksi. Vastaus ei tunnu kiinnostavan ketään.

Tuomio: MySQL:n ja Trolltechin käyttämä malli on mahdollisesti menestynein Open Source -maailman liiketoimintamalli. Sitä ovat heidän lisäksi käyttäneet myös monet muut saman tyyppiset yritykset. Malli ei kuitenkaan sovellu kaikkiin tapauksiin. Erityisesti se ei toimi loppukuluttajille tarkoitetuissa tietokoneohjelmissa, vaan kaikki mallia soveltaneet yritykset myyvät tuotteita toisille ohjelmoijille: ohjelmointikirjastoja tai tietokantoja. Malli läpäisee myös hakkerieettisen tarkastelun, vaikkakin avoimeksi jää kysymys siitä, miten nämä yhtiöt pärjäisivät puhtaassa Open Source -taloudessa.

27T ässä kohden on huomautettava, että kaikki Open Source ohjelmat eivät sisällä yhtä tiukkoja vaatimuksia. Monia Open Source -ohjelmia tai niiden osia voi myös hyödyntää suljetuissa ohjelmistoissa. Mutta GPL:n alaisten ohjelmistojen, kuten Linuxin kernelin ja MySQL:n kanssa sellainen ei ole sallittua.

34. EPÄONNISTUMISEN TARINA (COREL)

Ikään kuin edellisten esimerkkien vastakohtana – tarinana Windows-maailman yhtiöstä joka kokeili kepillä Linux-jäätä – tarkastelemme seuraavaksi *Corelia*. Windowsissa toimivista grafiikka- ja toimisto-ohjelmistaan tunnettu Corel teki aikoinaan lyhyen visiitin Linux-maailmaan. Ja nimenomaan vain visiitin. Corel esiintyy valitettavasti tässä kappaleessa varoittavana esimerkkinä, sillä se ei onnistunut kuulumaan Linux-porukkaan. Siitä ei koskaan ihan täysin tullut "yhtä meistä".

Corel lähti satsaamaan omaan Linux-jakeluversioon vuonna 1999, samaan aikaan kun se oli julkaisemassa Linux-versioita suosituista WordPerfect Office ja CorelDraw ohjelmistaan. Koska siihen aikaan markkinoilla olleet Linux-versiot eivät olleet vielä yhtä helppokäyttöisiä kuin nykyään, arveli Corel, että keskiverto WordPerfectin käyttäjä tarvitsisi myös helpommin ymmärrettävän Linuxin alleen. Siksi yhtiö päätyi kehittämään myös omaa Linux-versiotaan.

Koska Linuxille ei tuohon aikaan ollut hyvää tekstinkäsittelyohjelmaa, ainakaan sellaista joka olisi tyydyttävästi pystynyt käsittelemään Microsoftin Word-tekstinkäsittelyohjelman tiedostoja, odotti Linux-yhteisö eniten tulevaa WordPerfectiä, mutta myös helppokäyttöisemmälle Linuxille nähtiin tilausta.

Corelilla sen sijaan oli muitakin motiiveja Linux-seikkailuilleen. Se oli jäänyt toimisto-ohjelmien kisassa sivustakatsojaksi, kun Microsoft jylläsi monopoliasemassaan. Pari vuotta aiemmin se oli ilmoittanut siirtävänsä kaikki ohjelmansa Java-alustalle, hakien siten riippumattomuutta Microsoftista. Javan osoittauduttua täysin sopimattomaksi tähän tarkoitukseen, oli seuraavaksi pelimerkit asetettu Linuxille. Corel oli valmis satsaamaan mihin tahansa tekniikkaan, jolla voitaisiin hyökätä Microsoftin valta-asemaa vastaan.³⁰ Panoksena oli oikeastaan tulevaisuuden Microsoftiksi pääseminen. Yhtiö halusi tarjota vaihtoehtoista käyttöjärjestelmää ja sille vaihtoehtoista toimisto-ohjelmistoa ja vieläpä piirustus- ja kuvankäsittelyohjelmia.

Corel teki työnsä osittain ihan hyvin. Maailman suosituinta piirrosohjelmaa tekevä yhtiö jätti lähtemättömän jäljen muun muassa KDE-työpöytäympäristöön, jonka ohjelmoijat oppivat näkemään eron taitelijan ja insinöörin piirtämässä kuvakkeissa. Jos ei muuta, niin Linux sai Corelilta ainakin pirteän ulkoasun. KDE:n lisäksi Corel panosti myös wine-projektiin, jonka avulla se toi WordPerfectin ja CorelDrawn Linuxille.

Corel Linux OS ei kuitenkaan ollut markkinoille tullessaan mikään järisyttävä mullistus. Esimerkiksi suomalaista käyttäjää kummastutti puuttellinen skandinaavisten merkkien tuki, mikä oli muissa Linuxeissa (luonnollisestikin, onhan Linux syntynyt Suomessa) ollut kunnossa jo vuosia. Vaikka panostus graafiseen ilmeikkyyteen ja helppokäyttöisyyteen olikin kiitettävää, niin perusasioiden puuttuminen antoi toisaalta valjun kuvan uutuudesta, ja oikeastaan kumosi helppokäyttöisyys-uudistusten tuoman edun.

Corel Linux perustui suosittuun Debian Linuxiin. Siinä missä Debian tuohon aikaan sisälsi kuuden CD-levyn täydeltä erilaisia Linux-ohjelmia, ei Corel kuitenkaan omaan Linuxiinsa ollut ottanut mukaan yhtä CD-levyä enempää. Yksinkertainen on kaunista ja tyvestä on hyvä kiivetä puuhun, mutta monet Corelia kokeilevat Debian-käyttäjät luonnollisesti pettyivät ennemmin tai myöhemmin, kun jotain heidän kaipaamaansa ohjelmaa ei Corelilta löytynytäkään. Ongelmaan oli kuitenkin ratkaisu. Corel Linux oli nimittäin edelleen yhteensopiva Debianille tehtyjen ohjelmien asennuspakettien kanssa. Corel Linuxin käyttäjä pystyi siis laajentamaan Linuxiaan Debianin erittäin kattavista varastoista.

Sinänsä hyvä ominaisuus ei kuitenkaan pidemmän päälle pelastanut Corelia. Loppujen lopuksi monet Corel Linuxin käyttäjät päätyivät käyttämään Debiania, sieltähän kaikki hyvät ohjelmat tulivat joka tapauksessa. Eikä vanhassa ja vakaassa Debianissa ollut muitakaan Corelia vaivaavia pikku ongelmia, esimerkiksi muut kuin englanninkieliset aakkostot toimivat niin kuin pitääkin. Debianin lisäksi myös muut vanhat Linux-versiot pitivät pintansa Corelia vastaan. Perusasiat oli kunnossa ja helppokäyttöisyyteen oli panostettu lisää niissäkin. Loppujen lopuksi Corelin uutuus ei pärjännyt kisassa.

Corelin Linux-seikkailu osoitti, kuinka vaikeaa on tulla uutena pelurina haastamaan alalla kauemmin toimineita yrityksiä. Windows-ohjelmistotalon parhaat osaajat eivät vuoden Linux-kokemuksen jälkeen vetäneet vertoja alalla jo yli viisi vuotta – ja yleisemmin Unix-maailmassa jopa pari vuosikymmentä – toimineille guruille. Corelin Linuxissa ei ollut suurempaa vikaa, muttei todallakaan mitään ihmeellistäkään.

Vaikka ison ja perinteikkään ohjelmistotalon Linux-panostus sai myös paljon julkisuutta, se ei myöskään käynnistänyt mitään Windows-käyttäjien kansainvaellusta Linuxiin. Kun Corel Linux tuli, Windows-käyttäjät jatkoivat Windows-käyttäjinä ja Linux-käyttäjät jatkoivat kukin oman Linux-brändinsä käyttäjinä. Suurudensa ansioista Corel kieltämättä sai lyhyen visiittinsä aikana ihan mukavasti kokeilijoita, mutta Linuxin historiassa se jäi loppujen lopuksi vain marginaalimerkinnäksi.

Myöskään WordPerfect ja CorelDraw eivät Linux-maailmassa menestyneet. Niitäkin odotettiin innokkaasti, sillä Linuxille ei aiemmin oltu julkaistu näin merkittäviä Windows-maailmasta tuttuja ohjelmistoja. Loppujen lopuksi ne jäivät kuitenkin kauppojen hyllyille. Corelin vanhat asiakkaat olivat Windows-käyttäjiä, joten he eivät Linux-versioilla tehneet mitään. Corelille lienee ollut yllätys, että myöskään Linux-käyttäjät eivät ohjelmista kiinnostuneet. Word-tiedostomuotoa tukeva WordPerfect sai jonkinlaisen vastaanoton, mutta muuten Linux-projekti floppasi. Tyypillinen asenne Linux-käyttäjien joukossa oli, että WordPerfect oli vain huono vaihtoehto Microsoftin Wordille. Molemmat olivat yhtä suljettuja ohjelmistoja, ei siis mitään mielenkiintoista Open Sourcea kannattavan Linux-käyttäjän mielestä. Ja sitäpaitsi Microsoftin Word oli – karua mutta totta – monta kertaa WordPerfectiä parempi. Jos siis suljettuja ohjelmistoja ylipäätään päätyisi käyttämään, niin miksei sitten ainakin käyttäisi parasta mahdollista, joka siis ei ollut WordPerfect.

Kaksi vuotta myöhemmin Corelin Linux-seikkailut olivat tulleet tiensä päähän. Rahat oli loppu, suurin osa niistä oli kulutettu jo aiemmin epäonnistuneeseen Java-seikkailuun. Yhtiö yritti epätoivoisesti paikata kassavajettaan yhdistymällä Borlandin kanssa, mutta viime hetkellä Borlandin omistajat tajusivat mistä oli kysymys ja yhdistyminen peruuntui. Ilman käteistä yhtiö oli jo konkurssin partaalla, kun viime hetkellä monopoli oikeudenkäyntinsä kanssa kamppaileva Microsoft antoi tekoehditystä kilpailijalleen paksun setelinipun muodossa. Microsoft oli aiemmin samalla lailla estänyt Applen konkurssin. Se piti surkeassa kunnossa olevia kilpailijoitaan hengissä todistaakseen oikeudessa, että se ei ollut monopoliasemassa.

Kuin sattumalta kaikki Corelin Linux-projektit loppuivat vain viikkoja Microsoftin osakkaaksitulon jälkeen. Myöhemmin Linux-toiminnot myytiin Xandros-nimiselle yhtiölle, joka julkaisee edelleen helppokäyttöistä Linux-jakeluversiota. Viime vuoden aikana Xandros on jopa noussut ihan varteenotettavaksi peluriksi, vaikka ei markkinaosuuksissa vielä isojen Linux-jakelujen kanssa kilpailekaan.

Corelin virheet voidaan – jälkiviisaushan on aina helppoa – tiivistää kahteen pääkohtaan. Ensinnäkin sen motiivit olivat väärät. Se kehitti viimeisillä varoillaan lyömäasetta Microsoftia kohtaan, sen sijaan että olisi työskennellyt omien asiakkaidensa tarpeiden tyydyttämiseksi.

Toisekseen se syyllistyi virhearviointiin, joka muistuttaa jonkin verran Nokian WAP-seikkailuja. Corel selvästikin ajatteli, että jos he paketoivat Linuxin kauniiseen Corelin logolla varustettuun siniseen laatikkoon, niin pelkästään tällaisen Linuxin olemassaolo saa asiakkaat ryntäämään uutuuden kimppuun. Näinhän ei tietenkään ole ja varmaan Nokia ja Corelkin ovat tämän asian jo oppineet.

Ja loppujen lopuksi Corelia hyljittiin siksi, että se ei ollut "yksi meistä". Corelin insinöörit olivat käyneet hakemassa Linuxin osaset internetistä, työstivät niitä ihan kunnioitettavasti, pistivät pakettiin ja kaupan hyllylle. Avoimeen kehitykseen tottunut Linux-yhteisö ja yrityksen seinien sisällä työskentelemään tottunut Corel eivät kuitenkaan täysin kohdanneet. Jos vuorovaikutusta olisi ollut enemmän ja aikaisemmassa vaiheessa, olisi perustason puutteilta – kuten skandinaavisten merkistöjen ongelmat – varmasti välttytty kokonaan. Kun Corel myöhemmin sai Linuxistaan saksan-, ranskan- ja hollanninkieliset versiot tehtyä, se markkinoi Linuxiaan ensimmäisenä monikielisenä Linux-käyttöjärjestelmänä. Tämä oli jo törkeää arroganssia muita Linux-yhtiöitä ja Linux-yhteisöä kohtaan, ottaen huomioon että Corel ei suinkaan ollut näitä käännöksiä tehnyt itse, vaan kopioinut ne muilta jo olemassaolevilta Linux-tekijöiltä, lähinnä Debianilta.

Myös monet Corelia pienemmät yhtiöt ovat yrittäneet samaa. "Hei, netissä on näitä Open Source -ohjelmia ilmaiseksi, kopioidaan ne, pistetään CD:lle ja pakettiin ja sitten myydään ne", he ajattelevat. Näistä pienemmistä yhtiöistä ei välttämättä koskaan edes kuulla mitään, Corel oli sentään niin iso, että sen seikkailu aiheutti edes jonkinlaista huomiota. Mutta siitä huolimatta, tällainen asenne on jo asiakkaankin aliarvioimista. Ennen kaikkea sellaisella käytöksellä ei pääse mukaan "piireihin". Ja yksin jääneenä yhtiö oli heikompi kuin muut.

Corelin kannalta Linux-seikkailu oli surullinen tarina, mutta Open Source -periaatteen ja Linux-käyttäjien kannalta tämäkin surullinen tarina oli voitto. Jos kyseessä olisi ollut perinteinen suljetun ohjelmiston projekti, niin se olisi vain kadonnut vähin äänin. Projekti olisi lopetettu ja ohjelma kadonnut sen mukana. Mutta Open Source -maailmassa tehty työ ei katoa ja asiakas voi luottaa siihen, että elämä jatkuu vaikka oma myyjä menisi konkurssiin tai luopuisi tuotteen tukemisesta.

Corelin wine-projektiin tekemä työ on edelleen osa nykyistä wineä. Winen avulla voi edelleen käyttää CorelDrawta ja WordPerfectiä Linuxissa.³¹ Myöskään KDE ei ole Corelin visiitin jälkeen ollut entisensä. Tylsä ja insinöörimäinen tyyli on poissa ja KDE on kehittynyt todella taiteelliseksi työpöytäympäristöksi, joita alan konkarit Apple ja Microsoft jopa kopioivat! Vaikka nykyinen KDE on jo ihan eri taitelijoiden käsialaa, oli Corelin lyhyt osallistuminen tähän projektiin käänteentekevä, eikä tehty työ mennyt hukkaan vaikka firma tekikin kuperkeikan.

Tuomio: Corel pelasi Linux-pelinsä Open Sourcen säännöillä, mitä hakkerietiikkaan tulee. Yhtiö teki aidosti paljon ja hyvää työtä Linuxinsa, mutta tuote floppasi – Corel on Linux-maailman WAP. Vain siksi että teet jotain, ei tarkoita että asiakkaat ryntäävät ostamaan sitä. Ja ennen kaikkea: kunnioita sitä yhteisöä, johon tulet uutena jäsenenä.

Linuxin maskotti, pingviini on sattumalta laumaeläin ja symboloi sen takia hyvin myös hakkeri-ideologiaa. Pingviinit lämmittelevät pysyttelemällä lähellä toisiaan. Corel yritti yksin, ja metaforaa jatkamalla voitaisiin sanoa, että Corel Linux paleltui kuoliaaksi.

30Corelin seikkailuja onkin hyvä verrata toisen luvun "Suvaitsevaisuus"-kappaleessa esitettyyn Linusin asenteeseen. Häntähän "ei juurikaan kiinnosta [Microsoft], sillä hän ei itse käytä Windowsia. Linuxia hän on tehnyt omaksi huvikseen". Ehkä Corelinkin olisi pitänyt keskittyä enemmän omiin asioihinsa sen sijaan että lähes meni konkurssiin hyökätessään Microsoftia vastaan.

31Tosin se vaatii nykyään paljon enemmän käsityötä kuin silloin kun samat ohjelmat sai valmiiksi paketoituna suoraan Corelilta.

35. ÄLÄ OLE LIIAN AHNE (JAVA, GHOSTSCRIPT)

Yksi Corelin Linux-seikkailun opeista oli, että Linux-yhteisö suhtautui hylkivästi CorelDrawiin ja WordPerfectiin, koska ne olivat suljettuja ohjelmia. Linux-käyttäjät olivat ymmärtäneet ja käytännön kokemustensa kautta todenneet Open Sourcen hyödyt, ja siksi suljetut ohjelmistot eivät enää jaksaneet heitä kiinnostaa, oli niistä sitten Linux-versio tai ei. Miksi palata takaisin vanhaan ja huonoksi todettuun käytäntöön, kun siitä oli juuri päästy irti?

Niille yhtiöille, jotka edelleen tekevät ja myyvät perinteisiä suljettuja ohjelmistoja – ja nämä yhtiöt ovat edelleen suuri enemmistö – Open Source on tuonut uudenlaisen haasteen. On nimittäin vain ajan kysymys, milloin suosittu suljetun ohjelmiston rinnalle ilmestyy avoin kilpailija. Koska suljetut ohjelmistot eivät yleensä pysty kilpailemaan Open Source -vastineidensa kanssa ainakaan hinnalla, tietää tasaveroisen Open Source -vaihtoehdon markkinoilletulo yleensä tukalaa aikoa suljetulle ohjelmalle. Esimerkiksi juuri tätä kirjoitettaessa OpenOfficen yleistymisen tietää lopun alkua Microsoft Officen jo liian kauan jatkuneelle valtakaudelle. On vain ajan kysymys milloin tietokantajätit Oracle, IBM ja Microsoft joutuvat yritysten palvelinhuoneissa tosissaan kilpailemaan kehittyviä MySQL:ää ja PostgreSQL:ää vastaan – jo nyt nämä Open Source -tietokannat pyörittävät suurinta osaa internetin web-sivuista. Tietenkin joissain tapauksissa mitään kilpajuoksua ei ole ikinä ollutkaan. Esimerkiksi avoin Apache www-palvelin on alusta asti hallinnut www-palvelimien markkinoita yli 50%:n osuudellaan suljettujen www-palvelimien jäädessä heti alkuun kuriositeeteiksi.

Mielenkiintoinen kysymys siis onkin, kauanko suljettujen ohjelmistojen valmistaja voi olettaa myyvänsä tuotettaan, ennen saman alan Open Source -kilpailijan syntymistä. Tähän kysymykseen voidaan löytää vastaukseksi ainakin tiettyjä peukalosäännön tapaisia havaintoja.

On tietenkin ilmeistä, että mitä yksinkertaisempi ohjelma on, sitä helpompi sille on tehdä kilpailija – suljettu tai avoin. Eric Raymond tarjoaa The Cathedral and the Bazaar -kirjassaan myös toista itsestäänselvältä kuulostavaa sääntöä. Mitä suositumpi ohjelma, sen todennäköisemmin siitä on Open Source -versio. Tämäkin periaate tuntuu uskottavalta. Käyttäjien, eli asiakkaiden, määrä perinteisestikin lisää kilpailua ja laskee hintoja. On selvää, että näin on myös Open Source -ohjelmistojen syntyprosesseja tarkasteltaessa. Mitä useampi käyttäjä, sen suurempi hyöty Open Source -ohjelmasta on ja sitä helpommin sen tekemiseen (tai teettämiseen) löytyy motivaatiota.

Mutta yksi mielenkiintoinen sääntö voitaisiin muotoilla seuraavasti: Liika ahneus kääntyy äkkiä itseään vastaan. Esimerkiksi OpenOfficen olemassaolo on osittain tietenkin seurausta Eric Raymondin säännöstä – toimisto-ohjelmillahan on suuri käyttäjäkunta. Mutta osittain OpenOfficen kehitystä on varmasti vauhdittanut myös se, että Microsoft on monopoliasemansa turvin röyhkeästi nostonut Office-pakettinsa hinnan niin korkeaksi, että yksityisille ihmisille ja pienille yrityksillekin on kerta kaikkiaan kohtuutonta maksaa perustekstinkäsittelystä niin paljon. Vaihtoehtoiselle toimisto-ohjelmalle oli tämän takia suorastaan huutava tarve.

Ahneusteoriaa tukee myös *Sunin Java-ohjelmointikielen* tarina. Java syntyi Sunin laboratorioissa 90-luvun alkupuolella. Sen vahvuutena pidettiin uutta alustariippumatonta virtuaalikone-mallia, jonka ansiosta samaa Java-ohjelmaa pystyi ajamaan Windowsissa, Unixissa tai missä tahansa Javaa tukevassa käyttöjärjestelmässä. Tämä ominaisuus tuli kuin tilauksesta, kun internet ja www alkoivat levitä vuoden -95 tienoilla. Samoin kuin html-sivuja pystyi lukemaan millä tahansa tietokoneella, pystyisi Java-appletteja (suomeksi sovelmia, "pieniä sovelluksia") käyttämään millä tahansa tietokoneella.

Vaikka applettien käyttö alkuinnostuksen jälkeen lopahtikin, vakiinnutti Java hiljalleen asemansa ja on nykyään jo maailman suosituin ohjelmointikieli.³² Alustariippumattomuudella oli tähän varmasti osansa, mutta paljon on johtunut varmasti myös siitä, että selkeiden olio-ohjelmointirajapintojensa, yksinkertaisuutensa ja eräänlaisen akateemisen kauneuden johdosta Javaa opetetaan nykyään kaikissa yliopistoissa.

Java-kielellä on luonnollisesti toteutettu myös monia Open Source -ohjelmistoja. Eryisesti Apache-säätiö on näyttänyt mainetta monilla Java-maailman ohjelmilla, joita hyödyntävät myös kaupalliset Java-yhtiöt. Tästä huolimatta itse Javasta, eli kirjoitetun lähdekoodin tietokoneohjelmaksi muuntavasta *Java-kääntäjästä* ja ohjelman suorittamiseen tarvittavasta *Java-virtuaalikoneesta*, ei ole olemassa varteenotettavaa Open Source -toteutusta.

Javan nuoruusvuosina sille kehiteltiin kilpailijaa Kaffe-nimisen Open Source -projektin puitteissa. Kaffe oli aina hieman jäljessä Sunin virallisesta Javasta, mutta kuitenkin monesti ihan käyttökelpoinen. Nykyään Kaffe on kuitenkin jäänyt pahasti jälkeen muusta Java-maailmasta. Se on monissa yksittäistapauksissa edelleen käyttökelpoinen, mutta jo viisi vuotta sitten julkistettua Java 2 standardia se ei vieläkään tue.

Kaffe on pelkkä virtuaalikone, ei kääntäjä. IBM:itä taas löytyy suhteellisen käyttökelpoinen Open Source Java-kääntäjä nimeltä Jikes. Jikes on Kaffeen verrattuna pysynyt hyvin mukana Javan kehityksessä. Mutta Jikes taas on vain kääntäjä, se taas ei osaa suorittaa ohjelmia. IBM:illä on kehityksen alla myös Jikes-virtuaalikone, mutta se ei ihan vielä täytä virtuaalikoneelle asetettuja vaatimuksia, vaikka joitain yksittäisiä Java 2 -sovelluksia sillä jo voikin ajaa.

Myös Free Software Foundation on viime vuosina pannut lusikkansa Java-soppaan, ja ruvennut puuhaamaan Java-tukea kunniakkaaseen GCC-kääntäjänsä, joka jo ennestään tukee lähes kaikkia muita ohjelmointikieliä. Tämä GCJ-projekti on suurin piirtein Jikes-virtuaalikoneen tasoa: melkein, mutta ei ihan toimiva.

Javankin kanssa on vain ajan kysymys, ennen kuin Open Source -vaihtoehdot korvaavat Sunin "virallisen" Javan. Javaa odottaa sama kohtalo kuin C- ja C++ -kieltä, jossa Free Software Foundationin GCC on jo pitkään ollut standardi, johon kaupallisten pelureiden suljettuja vaihtoehtoja verrataan.

Sun on kuitenkin pystynyt pyristelemään vastaan kohta 10 vuotta. Ottaen huomioon kuinka suosittu kieli Java on myös Open Source -leirissä, tämä on merkittävää. Olisi voinut olettaa, että hakkerit tekevät ideologiaansa sopivan vaihtoehdon huomattavasti rivakemmin.

Osittain pitkä aika johtuu varmaankin tehtävän vaikeudesta. Ohjelmointikielen kääntäjän tekeminen on ohjelmoinnin vaikeimpia tehtäviä, verrattavissa esimerkiksi Linuxin kernelin tai GCC:n (joka siis on juuri kääntäjä) tekemiseen. Olen joskus lukenut erään TKK:n professorin sanoneen, että kaikista maailman ohjelmoijista vain alle yksi prosentti pystyy edes yrittämään tämän tason tehtävää. Lisäksi Javassa on ongelmana se, että kääntäjän lisäksi tarvitaan myös virtuaalikone, jonka tekeminen on vähintään yhtä haastavaa.

Mutta varmaan Open Source -vaihtoehdon syntymistä on myös hidastanut se, että Sun on koko ajan jakanut omaa Javaansa ilmaiseksi. Vaikka Java ei ole koskaan ollut Open Source, se on aina ollut ilmaiseksi vapaasti kaikkien halukkaiden käytettävissä.

Tämä tosiasia tuo esiin mielenkiintoisen puolen hakkeriyhteisössä, jota tässä tarkastelemme. Vaikka hakkereiden joukosta löytyy Richard Stallmanin kaltaisia idealisteja, jotka ehdottomasti kieltäytyvät käyttämästä mitään suljettua ohjelmistoa, näyttäisi siltä että useimmat hakkerit eivät ole näin jyrkkiä. Loppujen lopuksi monille sittenkin kelpaa myös se, että ohjelma on tarpeeksi halpa tai mielellään ilmainen.

Sunin strategiaa Javan suhteen on kritisoitu osuvasti siten, että siinä yhdistetään suljettujen ja avointen ohjelmistojen huonot puolet: Toisaalta Java kehittyi hitaasti ja kärsii laatuongelmista (tämä siis johtuen suljetusta kehitysmallista), toisaalta Javan levittäminen ilmaiseksi ei juuri hyödytä Sunia taloudellisesti. Mutta voi hyvinkin olla että Sunin Java-strategia on kuitenkin – vahingossa – ollut nimenomaan onnistunut. Jos Sun olisi lähtenyt Javan kanssa Microsoftmaisen rahastuksen tielle, olisi kilpailijoita – myös suljettuja – syntynyt hyvinkin nopeasti. Nyt Sun on kuitenkin löytänyt avoimuuden ja ahneuden välimaastosta sopivan tasapainone, minkä ansiosta se on saanut nauttia valta-asemastaan näinkin pitkään. Tästäkin huolimatta on vain ajan kysymys ennen kuin Jikes tai GCJ saavat Sunin etumatkan kiinni. Kun luet tätä, se on saattanut jo tapahtua.

Sunin Java-strategia on varmasti ollut monien sattumien seurausta. Sen sijaan *Artifex Software* käyttää vastaavanlaista taktiikkaa suosittuun *Ghostscript*-ohjelmansa jakelussa hyvinkin tietoisesti. *Ghostscript* on PostScript- ja PDF-tiedostojen katseluun, luomiseen, käsittelyyn ja tulostukseen käytetty ohjelma, joka on suosittu varsinkin Unix-maailmassa ja Open Source -yhteisössä ja se on pitkään ollut käytännössä ainoa varteenotettava kilpailija Adoben PS- ja PDF-tuotteille.³³

Alunperin Wisconsinin yliopiston suoijissa syntyneestä *Ghostscript*istä on olemassa kaksi eri versiota. Lähes avoin AFPL *Ghostscript*, sekä täysin vapaa GPL *Ghostscript*. AFPL *Ghostscript*ä voi ladata ilmaiseksi verkosta, mutta sen kaupallinen jälleenmyynti on kiellettyä. Kaupallisiin tarkoituksiin *Artifex* lisensoi *Ghostscript*ä erikseen ja mainitseekin muun muassa tulostin- ja kopiokonevalmistaja Xeroxin referenssiasiakkaanaan. Vaikka AFPL *Ghostscript* onkin ilmainen, jakaa *Artifex* vielä erikseen *Ghostscript*in GPL-versiota, joka siis täyttää tiukimmatkin Open Source -ehdot. Käytännössä kulloinkin GPL *Ghostscript* vastaa aina edellistä, noin vuoden vanhaa AFPL *Ghostscript*ä. Näin siis kaupallisen lisenssin ostanut asiakas saa aina vähän enemmän kuin vanhempaa GPL-versiota käyttämällä saisi. Toisaalta jakamalla *Ghostscript*ä ilmaiseksi, ja tämän lisäksi jakamalla hieman vanhentunutta 100% Open Source -versiota *Artifex* pitää huolen siitä, että tiukimmallaakaan Free Software ideologioilla ei ole mitään motivaatiota lähteä haastamaan *Ghostscript*ä.

Tuomio: Tässä kappaleessa olemme tutkineet suljettuja ohjelmistoja myyvien yritysten selviämistä Open Source -ohjelmistojen luomassa paineessa. Yhteenvedona voidaan todeta, että pihtaamistakin voi olla monen tasoista. Enemmistö Open Source -yhteisön kannattajista ovat valmiita hyväksymään "reilun kompromissin" ja ovat tyytyväisiä suljettuihinkin ohjelmistoihin silloin kun ne ovat helposti ja edullisesti saatavilla. Ainakaan keneltäkään ei tunnu riittävän motivaatiota sellaisen ohjelmiston haastamiseen. Sen sijaan liika pihtaaminen kääntyy helposti itseään vastaan. Älä siis ole liian ahne!

32Silloin kun ohjelmistoprojektia päästään tekemään puhtaalta pöydältä ja ohjelmointikieli voidaan valita täysin vapaasti.

33Adobe on sekä PostScript- että PDF-standardien luoja. Molempia tekniikoita voidaan käyttää teksti- tms. dokumenttien sähköiseen jakeluun, mutta PS on varsinaisesti tulostimien ohjauskieli, kun taas PDF on tullut tunnetuksi standardina tapana jakaa paperimuotoisia dokumentteja sähköisesti, esimerkiksi internetissä.

36. OPEN SOURCE JA KOUKKU (RED HAT NETWORK JA SUSE YAST)

Täysin avoimen mallin puitteissa toimiminen on osoittautunut käytännössä haasteelliseksi monelle Linux-yritykselle. Kuten edellisessä kappaleessa opimme, Linux-yrityksen asiakkaat ovat vapaita vaatimaan parasta ja edullisinta palvelua, jopa palvelun tarjoajan vaihtamisen uhalla. Tämä asettaa Linux-yrityksen niin sanotusti selkää seinää vasten ja monet yritykset ovatkin koittaneet helpottaa tukalaa tilannettaan liittämällä palveluihinsa jonkinlaisen koukun, joka avoimuudesta huolimatta saa asiakkaat pysymään uskollisina. Tarkastelemme tässä kappaleessa esimerkkeinä kahta tällaista tapausta.

Red Hat on tunnetuin Linux-käyttöjärjestelmää ja palveluja myyvä yritys ja onkin tavallaan sopivaa aloittaa eri Linux-yritysten läpikäynti juuri Red Hatista. Red Hat on yhtiönä ollut vahvasti sitoutunut Free Software -ideologiaan ja alusta asti julkaissut myös kaikki omat tuotteensa samalla avoimella GPL-lisenssillä kuin Linuxkin julkaistaan. Tästä huolimatta löytyy yhtiön historiasta myös joitakin koukkuja.

Vuoteen 2003 asti tärkein ja tunnetuin Red Hatin tuote oli juuri Linux-käyttöjärjestelmä. Suurin osa käyttäjistä kopioi sitä täysin ilmaiseksi internetistä, mutta moni myös osti Red Hatin CD-levyjä kaupasta. Red Hat on kuitenkin jo pitkään panostanut liiketoiminnan kasvattamiseen palvelupuolella. Koska yhtiön palkkalistoilta löytyy Linux-maailman lahjakkaimpia ohjelmoijia – eli hakkereita – niin yhtiön myymillä konsultointipalveluilla onkin tiettyä uskottavuutta. Esimerkiksi tunnettu verkkokauppa Amazon luotti Red Hatin konsulttiapuun siirtäessään internet-palvelimensa Linux-alustalle.

Palvelupanostuksista huolimatta Red Hatin tärkein tulonlähde toiminnan alkuvuosina oli juuri Linuxin myyminen CD-levyinä. Koska CD-levyjen sisältö oli myös mahdollista ladata ilmaiseksi suoraan internetistä, niin monet käyttäjät päätyivät hankkimaan Red Hat Linuxin sitä kautta.

CD-levyjen myyntiin liittyy tietty CD-levytuotannon logistiikasta seuraava jännite, joka oli Red Hatin kannalta taloudellisesti epäedullinen. Nimittäin kun CD-levyn sisältö – eli Linux ja siihen liittyvät ohjelmat – on saatu kursittua kokoon, ne voidaankin julkaista internetissä digitaalisessa muodossaan heti seuraavana päivänä. Samaan aikaan fyysisen CD-levyn matka kaupan hyllylle on vasta alkanut. CD-levyjen teettäminen, kansien painaminen ja kuljetukset kauppoihin ovat kaikki aikaa vieviä prosesseja ja usein kauppaan tulevat levyt olivat kuukauden tai parikin myöhässä internetissä leviäviin serkkuihinsa verrattuna. Näin pitkä myöhästyminen taas johti siihen, että nekin Red Hatin käyttäjät, jotka mielellään olisivatkin ostaneet levyn, päätyivät kärsimättömyyttään kuitenkin kopioimaan sen ilmaiseksi suoraan internetistä.

Red Hat yritti yhdessä vaiheessa ratkaista ongelmaa panttaamalla uusien versioidensa internet-julkistusta. Uudet versiot päästettiin internet-levitykseen vasta kun fyysiset CD-levyt olivat olleet kaupan hyllyillä joitakin viikkoja. Vaikka ratkaisu olikin ymmärrettävä, se herätti myös jonkin verran kritiikkiä. Kyseessähän oli kaikesta huolimatta eräänlainen pihtaaminen. Taktiikka oli mahdollisesti epäedullinen myös Red Hatille itselleen. Koska Linux kehittyi siihen aikaan niin huimaa vauhtia, oli parin kuukauden mittainen odotus suorastaan pitkä aika. Uuden version panttaaminen johti siihen, että heidän tuotteensa oli jo julkaisuhetkenä vanhentunut. Taktiikka myös antoi tasoitusta muille Linux-jakelijoille, jotka julkaisivat omat versionsa internetiin ilman viivytystä. Loppujen lopuksi Red Hat ei tainnut itsekään uskoa taktiikkaansa ja luopui siitä jonkin ajan kuluttua.

Internet-aikana käyttöjärjestelmätoimittajan tärkeimmäksi tehtäväksi on tullut järjestelmästä löytyvien vikojen pikainen ja mahdollisimman automaattinen korjaaminen. Tietokoneohjelmista löytyviä vikoja voidaan nimittäin käyttää tietoturtoihin. Esimerkiksi lähes kaikki Windows-maailmassa viime vuosina levinneet virukset ovat käyttäneet joko Windowsista itsestään tai Outlook-sähköpostiohjelmasta löytyneitä vikoja. Linux on osoittautunut tässä suhteessa Windowsia luotettavammaksi, eikä samanlaista virusepidemiaa ole Linuxissa vielä nähty, mutta teoriassa sama ongelma uhkaa kaikkia käyttöjärjestelmiä. Siispä tietoturvan ylläpitämisen yksi peruslähtökohtia on virheiden mahdollisimman pikainen korjaaminen. Käytännössä tämä vastuu lankeaa käyttöjärjestelmän toimittajalle, oli sen nimenä sitten Red Hat, Microsoft tai joku muu.

Red Hat -järjestelmän virheettömyyden ylläpito perustuu up2date-ohjelman käyttöön ja siihen liittyvään Red Hat Network -palvelun tilaamiseen. Red Hat Networkiin liittynyt asiakas voi sieltä hakea kaikki tarjolla olevat päivitysohjelmat parilla pikaisella napin painalluksella.

Red Hatin nykyinen liiketoiminta perustuu oikeastaan täysin Red Hat Networkiin liittyvien vuosimaksujen myymiseen. Red Hatin myymää käyttöjärjestelmää ei enää ollenkaan erotella omaksi tuotteekseen, vaan käyttöjärjestelmän hankkiminen ja vuoden päivityspalvelu kuuluvat aina samaan pakettiin. Hinnoittelukin perustuu vuosittaiseen tilausmaksuun, eikä enää CD-levyjen ostamistapahtumaan. Enterprise-tuotteet, joihin Red Hat nykyään yksinomaan keskittyy, ovat huomattavan kalliita. Konekohtainen vuosimaksu vaihtelee halvemmissa 179 – 349 dollarin versioista kalliimpiin 1499 – 18 000 dollaria maksaviin suurenkonepalvelimiin. Suurten asiakkaiden kanssa tehtävien konsultointisopimusten lisäksi Red Hatin tärkeimmäksi tulonlähteeksi onkin muodostunut juuri up2date-ohjelma ja siihen liittyvä Red Hat Networkin vuosimaksu.

Up2date-ohjelma on vahvasti sidottu juuri Red Hat Networkiin ja siihen liittyvään vuosimaksuun. Monissa muissa Linuxeissa vastaavat helppokäyttöiset päivitysohjelmat sallivat myös ohjelmien asentamisen ja päivittämisen kolmansilta osapuolilta. Tämä on käyttäjän kannalta hyvä ominaisuus, sillä se mahdollistaa sellaisten eksoottisempien ohjelmien helpon asennuksen ja ylläpidon, joita ei itse Linux-jakelun mukana tule. Ja tähän juuri on Open Sourcen idea, että käyttäjälle jää vapaus koota tietokoneensa niistä ohjelmista, mitkä hänelle parhaiten sopivat. Red Hatin kannalta se kuitenkin tarkoittaisi sitä, että asiakas voisi näiltä kolmansilta osapuolilta mahdollisesti päivittää koko käyttöjärjestelmänsä, eikä siksi tarvitsisi enää kallista Red Hat Network -sopimustaan ollenkaan. Toki Red Hatin Linuxin päivittäminen on mahdollista ilman up2date-ohjelmaa, mutta kaikki vaihtoehdot tavat vaativat kuitenkin enemmän tai vähemmän käsityötä. Ainoa virallisesti hyväksytty tapa on Red Hat Networkin käyttäminen.

Pakottamalla up2date-ohjelmansa käyttäjät kalliisiin vuosimaksuihin, Red Hat on yllättävästi valinnut tulonlähteekseen aika aggressiivisen ja ennen kaikkea epäavoimen koukun. Vaikka itse up2date-ohjelman lähdekoodi on kyllä avoin ja Open Source -periaatteen mukainen, sen käyttäminen on Red Hatin historia ja periaatteet huomioiden yllättävän suljettua. Kyse ei ole pelkästään rahasta, jonka tienäminen luonnollisestikin on voittoa tavoittelevalla yritykselle täysin sallittua, vaan ennen kaikkea valinnanmahdollisuuksien rajoittamisesta. Rahallisen tuoton tavoittelussa Red Hat on sitonut up2date-ohjelman omiin palvelimiinsa ja siten sulkenut ulkopuolelle koko muun internetin ja kaiken potentiaalin mitä siihen liittyy. Red Hatin käyttäjä joutuu siis tekemään itselleen epäedullisen valinnan. Jos käytät Red Hatia, on sinun hankittava kaikki ohjelmasi Red Hatin kautta. Jos tarvitset ohjelmia, joita Red Hat ei tarjoa, joudut elämään ilman. Ohjelmien asentaminen muualta kuin Red Hat Networkista on tietenkin mahdollista, mutta koska up2date-ohjelma ei niitä tue, niiden asennus ja ylläpito on kuitenkin käytännössä turhan hankalaa²⁰.

Vertailun vuoksi annamme kaikille tässä luvussa esitellyille yrityksille lyhyen arvosanan. Tai kutsuisimmeko sitä mieluummin tuomioksi. Mikä siis on tuomio Red Hatin kohdalla? Red Hat on Linux-yritysten joukossa yksi vahvimmin Free Software -ideologiaan sitoutuneista. Kaikki Red Hatin omat ohjelmat julkaistaan myös GPL-lisenssillä. Red Hat Networkin suljettu luonne on jonkin verran ristiriidassa hakkerietikan kanssa, mutta juuri ja juuri hyväksyttävissä.

Palvelumalliin ja yritystuotteisiin siirtyminen on ollut taloudellisesti kannattava strategia. Red Hat julkaisi syksyllä 2003 historiansa ensimmäisen voitollisen tuloksen, ja on sen jälkeen pysytellyt selvästi plussan puolella. Vuoden 2004 syksyllä voitollinen tulos oli jo 12 miljoonaa dollaria, mikä on paljon suhteessa 46 miljoonan dollarin liikevaihtoon. Voimakkaan kasvun jälkeen Red Hat onkin Red Hat Networkissa löytänyt hittituotteen ja lunastanut paikkansa aikuisten yritysten maailmassa, jossa ei eletä pelkästään hienolla tekniikalla, vaan toiminnan on oltava myös taloudellisesti kannattavaa.

Tuomio: Taloudellisesti menestyvä yritys, jonka toiminta kestää hakkerieettisen tarkastelun, mutta parantamisen varaakin olisi.

SUSE LINUX on vahvat saksalaiset juuret omaava Linux-yhtiö, jolla on yhtä vanha ja kunniakas historia kuin Red Hatillakin, ja sitä pidetään yleisesti Linux-jakelijoiden kilvassa kakkosena. Vielä 2000-luvun taitteessa SUSE oli liikevaihdossa ja työntekijöiden määrässä mitattuna Red Hatin kanssa tasaveroinen, mutta joutui sitten taloudellisiin vaikeuksiin ja melkein puolitti työntekijöidensä määrän. Yhtiö pysyi hengissä suureksi osaksi IBM:n tukemana.

Yhtiötä pidetäänkin IBM:n tärkeimpänä Linux-partnerina ja IBM:n tietokantatuotteet ja suurkonearkkitehtuurit ovat yleensä ensimmäisinä tuettuina juuri SUSE:n tuotteissa. SUSE Linuxia tukevat myös kaikki muut merkittävät ohjelmistotalot, muun muassa tietokantavalmistaja Oracle tai suurkonevalmistajat Cray ja SGI. Tätä taustaa vasten on helppo uskoa, että maailman parhaat Linux-ohjelmoijat löytyvät Red Hatin lisäksi juuri SUSE:ltä. IBM-yhteistyö toi kesällä 2003 SUSE:lle myös historiallisen sopimuksen Münchenin kaupungin 14 000 työaseman siirtämisestä Linuxille.

Melkoinen uutispommi Linux-maailmassa tuli syksyllä 2003, kun vanha tietotekniikkajätti Novell ilmoitti ostaneensa SUSE:n 210 miljoonalla dollarilla. Novell tunnetaan NetWare-käyttöjärjestelmästä, joka 90-luvun alkuvuosina oli suosituin yritysten lähiverkkojen tiedosto- ja tulostuspalvelinten käyttöjärjestelmä. Vaikka yhtiöllä edelleenkin on laaja ja uskollinen asiakaskunta, se on 90-luvun lopulta lähtien hävinnyt markkinaosuuksia Microsoftille ja monille koko yhtiö oli jo tietotekniikan historiaan kuuluva muinaisjäännös. Vuonna 2003 Novell kuitenkin päätti panostaa Linuxiin ja hankki taloonsa alan huippuosaamista ostamalla ensin Ximianin (josta kerrotaan myöhemmin) ja pari kuukautta myöhemmin maailman toiseksi suurimman Linux-jakelijan SUSE:n. IBM pysyi edelleen kuvioissa mukana, sillä SUSE-kauppaan liittyen yhtiö teki 50 miljoonan dollarin sijoituksen Novelliin. Yhdessä yössä SUSE LINUXista oli tullut vanhan ja kunnioitetun ja ennen kaikkea taloudellisesti vakaan tietotekniikkayhtiön tuote ja Novellista oli taas tullut kuuma nimi verkkopalvelimien toimittajana.

SUSE:n selviämistarina on havaittavissa periaate, joka pätee yleisemminkin. Lahjakas ammattilainen pärjää aina. Tässä nimenomaisessa tapauksessa kävi niin, että SUSE:sta oli tullut IBM:lle niin tärkeä asiantuntijakumppani, että yhtiön päästäminen konkurssiin ei kerta kaikkiaan ollut vaihtoehto. IBM tarvitsi SUSE:a oman Linux-strategiansa toteuttamisessa ja IBM:n tukemana yhtiö sitten selvisi kuiville muista vähemmän onnistuneista seikkailuistaan.

Novell-kaupassa oli kysymys samasta asiasta. Koska Linux on sekä avoin että ilmainen käyttöjärjestelmä, niin ei ole mitään syytä maksaa satoja miljoonia pelkästään oman Linux-version ostamiseksi. Tästä ei ollutkaan kyse, vaan Novell sai kaupassa nimenomaan huippuluokan Linux-osaamista, ja vieläpä halvalla jos minun mielipidettäni kysytään.

Samoin kuin Red Hatilla, SUSE:n kaksi tärkeintä tulonlähdetä ovat olleet konsultointi ja CD-levyjen myynti. Päinvastoin kuin Red Hat, SUSE on toistaiseksi myös jatkanut tavallisen kuluttajan kukkarolle sopivien pakettien myyntiä.

SUSE valitsi aikoinaan CD-levymyynnissä strategian, joka on saanut osakseen voimakasta kritiikkiä. Avointen ja vapaasti levitettävien Linux-ohjelmien lisäksi SUSE Linuxissa on SUSE:n itse kehittämä asennus- ja hallintaohjelma YAST, jonka lisenssiehdot eivät ole Open Source -määritelmän mukaisia. Asennusohjelmaa – ja sitä kautta koko CD:tä – voi kyllä kopioida vapaasti, mutta ei kuitenkaan myydä. Tällä tavoin SUSE on tehokkaasti pystynyt estämään halvat klooniversiot omasta Linuxistaan, jotka muilla Linux-yrityksillä jonkin verran nakertavat tätä tulonlähdetä. SUSE:n CD-levyjä ei myöskään voi kopioida internetistä, mikä on poikkeuksellista verrattuna muihin Linux-jakeluihin. Tätä kautta SUSE:n käyttäjät ovatkin käytännössä joutuneet muiden Linuxien käyttäjää enemmän turvautumaan kaupasta ostettuihin CD-levyihin saadakseen haluamansa.

Vaikka SUSE:lla on täysi laillinen oikeus määritellä omien ohjelmiansa lisenssiehdot niin tiukiksi kuin vain haluaa, on oikeutettua kysyä, miten yhtiö kehtaa tällaisia rajoituksia tehdä, kun CD:n muu sisältö on Open Sourcea ja siksi ilmaiseksi ja vapaasti SUSE:n käytettävissä. Missä on vastavuoroisuusperiaate? Ja onko SUSE:n strategia tulkittavissa niin, että maailman toiseksi suurin Linux-jakelija ei usko, että Open Source -ohjelmistoilla voisi oikeasti tehdä kannattavaa liiketoimintaa?

Tästä oikeutetusta kritiikistä huolimatta SUSE on loppujen lopuksi päässyt aika helpolla ja on koko ajan ollut kunnioitettu Open Source - yhteisön jäsen. Esimerkiksi hieman vastaavanlaista strategiaa yrittänyt Caldera sai paljon tylymmän vastaanoton ja onkin sen jälkeen käytännössä täysin sanoutunut irti Linux-liiketoiminnoistaan poltettuaan ensin kaikki sillat takanaan. SUSE:n strategian suosiollisemman vastaanoton syynä lienee se, että tosiasiallisesti SUSE on aktiivisesti mukana niin Linux-kernelin, KDE-ikkunointiympäristön kuin monen muunkin tärkeän Open Source -projektin kehityksessä. Nämä painavat enemmän hyvässä vaakakupissa kuin Open Source -periaatteesta poikkeaminen yhdessä tapauksessa.

Novellin ostettua SUSE:n yhtiö päätti julkaista tässä mainitun YAST - ohjelman samalla GPL-lisenssillä jolla Linuxiakin jaetaan, joten sitä myötä tämäkin koukku voidaan nyt siirtää historian lehdille kuuluvaksi ja pieni kauneusvirhe muuten huippulaadukkaassa Linux-jakelussa voidaan unohtaa. Tässä kappaleessa olemme kuitenkin tutkineet vanhaa SUSE:a ja sen koukustaktiikkaa. Tuomio siis perustuu siihen. Novellin ratkaisut ovat uutta tulevaisuuden SUSE:a ja jäävät tulevien kirjailijoiden arvioitavaksi.

Tuomio: Talousvaikeuksista selvinnyt yhtiö, jonka liiketoiminnan tukijalka on vahvaan osaamiseen perustuva konsultointi, erityisesti eksoottisissa suurneuympäristöissä. SUSE on yhtiönä esimerkki siitä, että asiansa osaava ammattilainen pärjää aina. Yhtiön Linux -jakeluersio on suosittu, mutta kokonaisuuden toteutus ei valitettavasti tässä tarkastelussa läpäise asettamiamme eettisiä kriteereitä. Siinä missä Red Hatin koukku ylitti riman juuri ja juuri, SUSE:n yhtä täpäpästä alittaa sen.

20Selvyydeksi: Red Hatin vuodesta 2003 julkaisemassa ilmaisessa, lähinnä harraste- ja testikäyttöön suunnatussa Fedora Core Linuxissa up2date ohjelmaa on muutettu niin, että käyttäjä voi vapaasti lisätä myös muita www-osoitteita joista ohjelmia voi asentaa ja päivittää. Varsinaiset yrityksille suunnatut Red Hat -tuotteet ovat kuitenkin tätä kirjoitettaessa edelleen sidotut Red Hat Network -palveluun.

37. VAPAUTETUT I: NETSCAPE/MOZILLA

Linuxin nousun yhteydessä on usein herännyt keskustelua myös suljettujen ohjelmistojen koodin julkaisemisesta avoimesti Open Source -tyyliin. Joskus suljetun ohjelmiston tekijä saattaa kuulla sen suuntaisia toiveita asiakkaaltaan, toisaalta taas ohjelman tekijä itse saattaa olla kiinnostunut avoimuuden tuomista eduista. Pari vuotta sitten USA:ssa käydyssä Microsoftin monopolioikeudenkäynnissä esitettiin, että esimerkiksi Microsoftin Officein tekijät saavat epäriilua hyötyä siitä, että heillä on pääsy allaolevan Windows-käyttöjärjestelmän lähdekoodiin, kun taas kilpailijoilla ei tätä etua ole, ja siksi vaadittiin jopa Windowsin lähdekoodin julkistamista. Oikeus ei kuitenkaan määrännyt tätä tehtäväksi.³⁴ Vanhoja suljettuja ohjelmistoja on kuitenkin viime vuosina "vapautunut" yhä kiihtyvään tahtiin ja seuraavissa kappaleissa käymme läpi joitakin sellaisia tarinoita.

Ensimmäinen varteenotettava ohjelma, joka siirrettiin suljetusta kehitysmallista Open Source -projektiksi oli kaikkien webin käyttäjien tuntema *Netscape*. Tämä historiallinen tapahtuma tammikuussa 1998 vaikutti osaltaan muun muassa koko "Open Source" -termin keksimiseen.³⁵ Kerrotaan, että Netscapen johtajat olivat lukeneet muun muassa Eric Raymondin kirjoittamaan *The Cathedral and the Bazaarin*, ja vahvistuneet uskossaan, että avoin kehitysmalli voisi pelastaa Netscapen täydelliseltä tuholta.

Vaikka vuotta 1998 voidaankin monin tavoin pitää vuotena jolloin Linux ja avoin kehitysmalli todella astuivat parrasvaloihin ja yrity maailman tietoisuuteen – ja tämä oli paljon juuri Netscapen käynnistämän kehityksen tulosta – niin Netscapen päätös ei kuitenkaan ollut niin suuri lahja Open Source -yhteisölle kuin aluksi olisi voinut kuvitella. Microsoftin Internet Explorer oli jo ottanut markkinajohtajan aseman selainsodassa ja oli myös laadullisesti edellä Netscapen selainta, jonka laatu taas tuntui heikkenevän versio versiolta. Julkistuksesta ei kulunut kauaa ennen kuin Netscape, tuo Internet-buumin aloittanut ja ikonisoitunut nuori yritys, oli tullut pisteeseen, jossa sitä pantiin lihoiksi. Ennen kuin vuosi 1998 oli kulunut päätökseensä Sun ja AOL olivat jakaneet keskenään Netscapen jäämistön. AOL peri selaimen ja siihen liittyvän *Mozillaksi* ristityn Open Source -projektin. Netscape-yhtiö jäi elämään "uuden talouden" suurimpana legendana, mutta todellisuudessa se oli vain 4 vuoden ikäinen kirkas tähdenlento, jonka tilivuodet yhtä lukuunottamatta jäivät pahasti tappiollisiksi. Ja ehkä tämä juuri kuvasikin hyvin tuota vuosituhannen vaihteen "uutta taloutta".

Suuren alkuinnostuksen jälkeen Mozilla-projekti osoittautuikin ainakin osittain pettymykseksi. Pian paljastui, että kilpailussa Microsoftia vastaan Netscapen ohjelmoijat olivat aikataulupaineiden alla sortuneet tuottamaan todella sekavaa ja laadutonta koodia. Parin ensimmäisen vuoden ajan Mozilla toimi lähinnä siivousprojektina. Lopulta Mozilla-koodarit tulivat siihen tulokseen, että olisi parempi yksinkertaisesti kirjoittaa selaimen tärkein osa, www-sivuja näyttävä komponentti, kokonaan uudestaan. Vasta tämän jälkeen alettiin projektiin saamaan edes jonkinlaista ryhtiä. Lopulta Mozilla 1.0 julkaistiin neljä ja puoli vuotta projektin aloittamisen jälkeen.

Eräs Mozilla-projektin johtohenkilöitä, Jamie Zawinski totesi osuvasti, että Open Source ei ollut taikapölyä, ja oli siinä ihan oikeassa. Mikään mahti maailmassa ei pysty pelastamaan ohjelmointiprojektia, joka on sotkeutunut spagettikoodiin, rönsyilyyn ja kaikkeen mihin laadusta tinkiminen johtaa. Vaikka Mozillasta on nyt julkaistu jo useita oikein hyviä versioita ja varsinkin virheiden määrä on pienentynyt Netscapen ajoista, oli hitaus ja koodin rönsyily pitkään puheenaihe johon projektin yhteydessä palattiin. On mahdollista, että kyse on myös projektin kulttuuriin liittyvästä ongelmasta. Historian perintönä saatu Netscapen löysä asennoituminen laatuun tuntui vielä painavan.

Mozillasta erotettiin erilliseksi projektikseen Firefox-selain, joka perustuu samaan koodiin kuin Mozilla, mutta jota lähdettiin kehittämään eri henkilöiden vetämänä. Ensimmäinen virallinen versio, Firefox 1.0 julkaistiin marraskuussa 2004 ja siinä oli vihdoin hitaus ja muut ongelmat saatu lopullisesti kuriin. Vanha Netscape oli vihdoin noussut kuolleista. Nopeampana, värikkäämpänä ja monia uudempiä tekniikoita tukevana se oli tullut haastamaan jo monta vuotta paikallaan junnanneen dinosauruksen Internet Explorerin valta-asemaa.

Vuodesta 1998 vuoteen 2004 on pitkä matka IT-alan vauhdissa. Täytyy kuitenkin muistaa, että Mozilla-projektiin liittyi muita tärkeitä tavoitteita, jotka se kyllä täytti jo paljon aikaisemmin. Vaikka Microsoft lopulta saavutti kirkkaasti yli 90% valta-aseman selainmarkkinoilla, jota se tätä kirjoitettaessa edelleen pitää, takasi Mozilla-projekti pelkällä olemassaolollaan sen, että webistä ei tullut Microsoftin omaisuutta. Useamman selaimen olemassaolo varmisti käytännössä, että kotisivuntekijät – ainakin asiansa osaavat sellaiset – pitäytyivät yhteisissä standardeissa eikä Microsoft saanut täydellistä monopoliasemaa ja saneluvaltaa.

Ja Firefoxin muodossa projekti lopulta saavutti myös varsinaisen tavoitteensa, siitä syntyi maailman paras selain, kilpailijoitaan hienompi ja nopeampi. Vaikka Mozilla ja Firefox ovat edelleen marginaalitekijöitä mikäli tilastoihin on uskomisen – Firefox tosin on vakaassa nousussa – olemme ensimmäistä kertaa pitkään aikaan sellaisessa tilanteessa, että Internet Explorer on kolmesta suurimmasta kilpailijastaan³⁶ teknisesti jäljessä! Vaikka alkuperäisestä Netscape-selaimesta ei prosessissa paljoa ole jäänyt jäljelle, täytyy olla kiitollinen niille, jotka Mozilla-projektin käynnistivät. Mozillan ympärille rakentunut yhteisö on osoittautunut vahvemmaksi kuin ala-arvoinen spagettikoodi, jonka se joutui perimään. Mozilla-projekti onkin antanut Open Source -yhteisölle paljon enemmän kuin Mozilla-selaimen. Esimerkiksi lähes kaikkien Open Source -ohjelmointiprojektien nykyään käyttämä Bugzilla-työkalu on lähtöisin Mozillasta, ja se on myös monien ohjelmisto-yritysten sisäisessä käytössä suljettujen ohjelmistojen tuotannossa.

Lopuksi on myös todettava, että Mozillan myötä sulkeutui eräs merkittävä historiallinen ympyrä. Netscape-selain perustui alun perin akateemisessa ympäristössä NCSA:ssa kehitettyyn Mosaic-selaimen.³⁷ Kun Netscapen koodi vapautettiin Mozilla-projektin alaisuuteen, se siis palasi ikään kuin juurilleen. Ja oikeastaan Mozillan www-selain on vain pieni osa suurempaa tarinaa. World Wide Web -teknologia nimittäin kehitettiin alunperin CERNin tutkimuskeskuksessa, josta se annettiin maailmalle avoimesti ja vapaasti käytettäväksi. Ja ennen webin keksimistä koko muu internet ja sen teknologiat ja standardit olivat samalla lailla kehittyneet Unix-hakkerien avoimuuden periaatteita noudattavassa yhteisössä. Kun avoimesta Apatchestakin tuli webin yleisin palvelin, oli suljettu Netscape itse asiassa pienoinen kummajainen muuten avoimessa maailmassa ja Mozilla taas eräänlainen kotiin palannut tuhlaajapoika.

Tuomio: Netscapen koodin avaaminen toimi lähtölaukauksena Open Source -liikkeen popularisoinnille ja teki siitä vakavasti otettavan vaihtoehdon myös muiden yritysten suunnitelmissa. Open Source oli Netscapelle eräänlainen puolustautumismekanismi Microsoftin murskaavaa ylivaltaa vastaan ja tältä osin strategia osoittautui onnistuneeksi. Sen sijaan teknisesti Mozilla-projekti kohtasi suuria haasteita. Monta vuotta holtittomasti paisunut spagettikoodi ja siitä seuraavat ongelmat eivät avoimuuden seurauksena maagisesti kadonneet ja ongelmien selvittämiseen kului enemmän aikaa kuin kukaan olisi osannut odottaa.

34Pian George W. Bushin tultua valtaan pitkään jatkunut oikeudenkäynti saatiin näppärästi sovittua siten, että Microsoft todettiin syylliseksi monopoliaseman väärinkäyttöön, mutta se ei saanut siitä mitään rangaistusta.

35Aiemmin siis oli käytetty vain termiä "Free Software".

36Mozilla-perheen (Mozillan ja siitä johdettujen Netscapen, Firefoxin, Galeonin jne...) lisäksi kaksi muuta IE:n kilpailijaa ovat suljettu Opera ja avoimen lähdekoodin Konqueror, johon perustuu myös Applen uusi Safari-selain.

37Ja niin perustuu myös suljettu Internet Explorer.

38. VAPAUTETUT II: INTERBASE/FIREBIRD

Borland tuli 90-luvulla tunnetuksi hyvistä ohjelmointityökaluistaan. Itsekin tein ensimmäisen Java-sovellukseni ilmais-CD:ltä saamalla *Borlandin* JBuilder-ohjelmalla. Syistä joita kukaan ei vielä kukaan tunnu ymmärtävän, tämä maineikas yhtiö joutui 90-luvun loppupuolella kuitenkin jonkinlaiseen identiteettikriisiin. Jonkinlaisessa brändinluontitaroituksessa se yhtäkkiä muutti nimekseen *Inprise*. Kun kohta tämän jälkeen kävi selväksi, että kukaan ei ollut ikinä kuullutkaan *Inprise*-nimisestä yrityksestä ja kaikki edelleen kutsuivat *Borlandia* *Borlandiksi*, yhtiön johto päättikin muuttaa nimeksi *Inprise/Borland.com* ja kohta sen jälkeen *Borland/Inprise*³⁸ ja lopulta takaisin tuttuun ja turvalliseen *Borlandiin*. Kaikista hulluista IT-alan tarinoista joita 2000-luvun taitteesta on kerrottu, tarina *Borland-Inprise-Borlandista* on mielestäni skitsofreenisin.

Yksi *Borlandin* tuotteista, jonka se oli ostanut itselleen vuonna 1991, oli SQL-tietokanta nimeltä *InterBase*. (*InterBase* itse oli syntynyt jo vuonna 1984.) *InterBase* ei ollut suurten serkkujensa *Oracle*n ja *IBM DB2:n* veroinen, mutta pienemmässä tietokantakäytössä kuitenkin suhteellisen yleinen.

Juuri ennen vanhan vuosituhannen päättymistä, joulukuussa 1999 tämä *Borlandin*³⁹ ammattitaitoinen ja määrätietoinen johto ilmoitti *InterBasen* kehittäjille, että sen kehitys oli päätetty lakkauttaa. Ilmeisesti johto ei ollut tyytyväinen *InterBasen* tuottamaan voittoon, jota kuitenkin kuulemma oli. Tuote ei siis ollut tappiollinen, mutta oli syystä tai toisesta päätetty kuitenkin lakkauttaa.

Borlandin asiakkaita päätös ei miellyttänyt. Sellaiset suuret asiakkaat kuten *Nokia* ja *Motorola* käyttivät *InterBasea* laajasti ja tärkeissä projekteissa, ja olivat jopa valmiit ostamaan koko *InterBase*-osaston *Borlandilta*, jos muu ei auttaisi. Kun näin isot asiakkaat tarpeeksi monta kertaa kovistelivat *Borlandia*, *Borland* lopulta taipui kompromissiin. Se päätti julkaista *InterBasen* Open Source -lisenssillä.

Ja sen pituinenko se? Mitä vielä, tarinahan on vasta alkamassa.

Kun *Borlandin* johtajat lopulta tajusivat kuinka vahvasta tuotteesta olikin kysymys, he tulivat yhtäkkiä katumapäälle. Kyllähän näin hyvää tuotetta kannattaisi myydä jatkossakin, kun isot asiakkaat tulevat oikein rahasäkkien kanssa vaatimaan sen myymistä. Samalla kun ensimmäinen Open Source -versio, *InterBase 6.0* julkaistiin, *Borland* aloikin vetäytymään hankkeesta. Open Source -versiolle ei yhtiön puolesta myönnytykään minkäänlaista takuuta tai tukea ja kaiken kaikkiaan yhtiö palasi vanhaan ja hyväksi havaittuun suljettuun malliin, veloittaen asiakkaitaan yksi lisenssi per kone -periaatteella.

Tästä sekavasta ja osin huvittavastakin tarinasta syntyi *Firebird*, *InterBasen* aidosti avoin Open Source -versio. *Firebird*-projektin ympärille kerääntyi joukko ohjelmoijia, jotka olivat saaneet tarpeekseen *Borlandin* poukkoilusta ja päättivät jatkaa avoimen version työstämistä omin päin. *Borlandin* omat ohjelmoijat eivät osallistuneet *Firebirdiin*, mutta mukana oli joitakin *Borlandin* ulkopuolisia guruja, jotka olivat jo 80-luvulta asti työskennelleet *InterBasen* ensimmäisten versioiden parissa sekä muita *InterBasen* parissa toimineita ohjelmoijia ja konsultteja.

Firebirdin ensimmäiset vuodet muistuttavat monilta osin Mozillan tarinaa. Kun suljetun ohjelmiston koodi julkistetaan, paljastuu yleensä kerralla kaikki se roska, mikä vuosien saatossa on kaikessa hiljaisuudessa lakaistu maton alle. Ensimmäisiä yllätyksiä oli varmasti se, että InterBasen koodin kääntäminen ajettavaksi ohjelmakoodiksi on kuuleman mukaan tuottanut monta tuhatta erilaista varoitusta mahdollisista virheistä koodissa. Nämä "kääntäjän varoitukset" eivät välttämättä ole vielä vakavia virheitä eivätkä yleensä estä ohjelmaa toimimasta, mutta se että niitä on ollut tuhansia kertoo jotain asenteesta, jolla ohjelmoijat ovat työskennelleet.

Kun ohjelmaa kehitetään alusta asti avoimesti, ei tilanne yleensä pääse kehittymään niin pahaksi kuin InterBasen ja Mozillan tapauksessa. Suljettujen ohjelmistojen kehityksessä on kiusaus mennä yli siitä, missä aita on matalin. Virheistä ei välitetä, vaan jos ohjelma kuitenkin kääntyy ja käynnistyy ja näyttää päällisin puolin toimivalta, se pistetään oitis myyntiin. Open Source -ohjelmien laadun taas jokainen pystyy itse toteamaan ja tekijöiden henkilökohtainen mainekin on vaakalaudalla. Julkisuudessa työskentely asettaa riman korkeammalle.

Myös Firebirdin hakkerit siis joutuivat ensimmäisen vuoden ajan toimimaan lähinnä siivousprojektina. Ja suursiivouksen aikana tulisi kaapista löytymään vielä todellisia luurankoja.

InterBasessa, kuten tietokantatuotteissa yleensä, on mahdollista määritellä eri käyttäjätunnuksia ja näille käyttäjille erilaisia oikeuksia. Tietokannan ylläpitäjällä voi olla oikeus tehdä mitä vaan, sihteerillä voi olla oikeus lisätä ja poistaa tietueita ja muilla käyttäjillä ainoastaan oikeus lukea tietoja, mutta ei muuttaa niitä. Käyttäjätunnukset ja salasanat sekä tiedot siitä, kenellä mitäkin oikeuksia on, pitää luonnollisestikin tallettaa johonkin, jotta tietokantaohjelma voi niitä käyttää.

Tietokantatuotteen tapauksessa ohjelmoija on sikäli onnekaassa asemassa, että tietokantaahan juuri voidaan käyttää käyttäjätietojen tallettamiseen! Ja niin oli InterBasen tapauksessa tehtykin. Vaan tällainen järjestely johtaakin mielenkiintoiseen muna-kana ongelmaan. Jotta tietokannan tietoihin pääsisi käsiksi, pitää ensin tarkistaa käyttäjätunnus ja salasana. Mutta käyttäjätunnus ja salasana onkin talletettu tietokantaan, johon ei pääse käsiksi ennen kuin on ensin tarkistettu käyttäjätunnus ja salasana!

Miten sinä ratkaisisit tällaisen ongelman? Ratkaisuja on varmasti monia, mutta ratkaisu johon Borlandin ohjelmoijat päätyivät vuonna 1992, on varmasti kaikkein suoraviivaisin. InterBaseen lisättiin (ns. "kovakoodattuna") ylimääräinen käyttäjätunnus ja salasana, jolla oli kaikki oikeudet kaikkiin tietokantoihin! Tätä käyttäjätunnusta käytettiin InterBasen koodissa sisäisesti ohittamaan muna-kana ongelma, eli tarkistamaan varsinainen käyttäjätunnus ja salasana. Mutta nyt kun koodi oli avoin⁴⁰ ja kaikki tarkistukset ohittava salasana kaikkien tiedossa, sitä pystyi käyttämään myös kenen tahansa tietokantaan murtautumiseen.

Myös toinen lähes yhtä iso mörkö löytyi. Testausta helpottamaan Borlandin laaduntarkkailuosasto oli vaatinut InterBaseen lisättäväksi komentoa, jolla tietokannan sisällön pystyi sotkemaan tai kokonaan tuhoamaan.⁴¹ Jostain käsittämättömästä syystä vaadittiin myös, että tämä komento jätettiin myös myytävään InterBasen versioon. Tietenkään komennosta ei kerrottu InterBasen käyttöohjeessa, koska komento oli vain laaduntarkkailuosaston sisäiseen käyttöön, mutta "tuhoa kaikki tiedot" -tyyppisen komennon jättäminen asiakkaille myytävään versioon kuulostaa kuitenkin melkoiselta venäläiseltä rulettilta. Vielä ihmeellisempi on väite, että näin oli tehty nimenomaan laaduntarkkailuosaston vaatimuksesta.

Miten Firebird-projektille sitten on käynyt?

Ihan hyvin, kiitos kysymästä. Pahimpien sotkujen siivoamiseen kului reilu vuosi, jonka jälkeen meni vielä puoli vuotta ennen kuin Firebird 1.0 saatiin julkaistua maaliskuussa 2002. Vaikka Firebird-hakkerit selvisivät omasta suursiivouksestaan ripeämmin kuin Mozilla-sotkun kanssa painineet toverit, osoittaa tämäkin kuinka valtavasta urakasta vanhan ohjelman kehitysvastuun perimisessä on kysymys.

Vaikka Firebird onkin menestynyt kohtalaisesti, se ei ole saanut Open Source -yhteisössä samanlaista vastaanottoa kuin Mozilla, sillä avoimet MySQL ja PostgreSQL täyttävät jo hakkeriyhteisön tietokantatarpeet paremmin kuin hyvin. Sen sijaan Firebird on löytänyt yleisönsä vanhasta InterBase-käyttäjäkunnasta, joiden ei enää tarvitse olla yhtä huolissaan Borlandin oikuista.

Tuomio: InterBasen tarinasta käy ilmi, minkälainen riski suljettujen ohjelmistojen käyttö on. Mitä sinä teet sinä päivänä kun valmistaja päättää lopettaa sinulle elintärkeän tuotteen jatkokehittämisen? Yrityksen IT -infrastruktuurin rakentaminen Open Source -ohjelman varaan on turvallinen ratkaisu, koska silloin valta päättää omasta tulevaisuudesta on ohjelman käyttäjällä itsellään. Firebirdin ensimmäinen vuosi paljasti Mozillan tapaan, mikä ero on suljettujen ja avointen ohjelmien laadulla. Open Source -ohjelmista on sanottu, että niiden tietoturva on parempi, koska avoimuus auttaa paljastamaan koodissa olevat virheet. Firebirdin tapauksessa näin kävikin, mutta kesti kuitenkin yli kuusi kuukautta ennen kuin kriittinen tietoturva-aukko löydettiin, vaikka se oli koodissa täysin ilmeinen ja helposti havaittavissa.

38Viimeistään tässä vaiheessa saa jo hymyillä. Tässäkin kirjassa olemme aikaisemmin puhuneet määrätietoisuutta ja suunnitelmallisuutta vastaan, mutta rajansa kaikella!

39Tai tarkkaan ottaen "Inprisen", mikä oli yhtiön nimi juuri joulukuussa 1999.

40On huomattava, että koodin avoimuus ei ollut ongelma, vaan alkuperäisen Borlandin insinöörin laiskuus ja vastuuttomuus. Salasana on ollut helposti löydettävissä myös ennen InterBasen Open Source -julkistusta tutkimalla InterBasen konekielisiä tiedostoja tavallisella tekstieditorilla. On täysin mahdollista että joku krakkeri on näin tehnytkin, onhan "takaovi" ollut olemassa lähes 10 vuotta.

41Todennäköisesti komentoa on käytetty luomaan keinotekoisia virhetilanteita, jotta voitaisiin testeissä tutkia miten ohjelma niistä selviää.

39. VAPAUTETUT III: QUAKE

id Software jää historiaan yhtiönä, joka julkaisi ensimmäisen kolmiulotteisen ammuskelupelin PC-koneille. Et tosin varmaankaan ole koskaan kuullut *Hovortank 3D*:stä – minä en ainakaan ollut – mutta sen seuraajat *Doom* ja *Quake* ovat legendaarisia. Varsinkin *Quake*, jota pystyi pelaamaan verkkopelinä sai uskollisen kannattajajoukon. Ilmeisesti mikään ei ole hauskeempaa kuin ryömiä viemärissä ja ampua möröiksi pukeutuneita kavereitaan liekinheittimellä – siis virtuaalisesti.

Niinpä pelien kehittäjä *John Carmack* päätti jouluna 1999 antaa *Quake*-faneille lahjan. Hän julkaisi *Quake 1:n* lähdekoodin Open Source -lisenssillä internetissä. Kyseessä ei ollut mikään suuri uhraus. Pelin myynti oli jo laskenut olemattomiin ja *id Software* myi jo uusia entistä hienompia pelejä, muun muassa *Quake 2*:ta ja *Quake 3*:a, joiden koodia ei julkaistu.

Hyvin pian iloisen yllätyksen jälkeen *Quake*-maailmassa vallitsi kuitenkin sekasorto. Lähdekoodin julkaiseminen oli nimittäin johtanut siihen, että verkkopelissä ruvettiin fuskaamaan! Osaavat ohjelmoijat tekivät omaan *Quake*ensa esimerkiksi sellaisia muutoksia, että heidän pelihahmonsa automaattisesti väisti kaikkia luoteja, jotka tulivat heitä kohti.⁴² Tai sitten he ohjelmoivat hahmonsa salamannopeasti tähtäämään ja ampumaan kaikki viholliset heti kun ne olivat näkyvissä. Vielä yksi hauska huijaus perustui siihen, että ohjelmoitiin pelin seinät läpinäkyviksi, jolloin pelaaja pystyi näkemään nurkan takana väijyvän vihollisen (joka kuitenkin itse käyttää pelistä sellaista versiota, jonka seinät eivät ole läpinäkyviä).

Vaikka tällaisten röntgenkatseiden ja automaattiväistämisten ohjelmoiminen on sinänsä itsessään mielenkiintoinen tehtävä, jonka paremmuudesta voisi myös kisailla, niin monia *Quake*-pelaajia huijausversiot kuitenkin häiritsivät. Ei ole kovin kivaa pelata kaveria vastaan, johon on mahdotonta osua ja joka näkee sinut jo nurkan takaa.

Eric Raymond, Open Source -aatteen innokas markkinamies, kiirehti julkaisemaan esseen, jossa hän ruoti tätä mielenkiintoista ongelmaa. Hän esitti, että tämä oli jälleen yksi esimerkki siitä, miten suljetuissa ohjelmissa päädyttiin helposti ratkaisuihin, jotka eivät oikeasti olekaan turvallisia ja eivät kestä avointa tarkastelua, ja että jos *Quake* olisi kehitetty alusta asti avoimen kehitysmallin mukaisesti, olisi tällaiset ongelmat otettu huomioon jo silloin, eikä niitä olisi siksi päässyt syntymään.

Mutta tällä kertaa Eric oli väärässä! *Quaken* ongelmat eivät johtuneet siitä, että sen tekijät olisivat suljetussa maailmassaan sortuneet tietoturvaan heikentäviin oikopolkuihin. Itse asiassa pelissä ei voinut huijata esimerkiksi lisäämällä itselleen loputtomasti aseita ja panoksia, koska tällaiset asiat oli peliä suunniteltaessa nimenomaan estetty. Kaikki huijaukset (röntgenkatsetta lukuunottamatta) olivat luonteeltaan sellaisia, että niissä pelaaja käytti omaa tietokonettaan eräänlaisena apuvälineenä, esimerkiksi tähtäämiseen. Nämä huijaukset olisivat periaatteessa olleet toteutettavissa jo aikaisemmin ilman lähdekoodiakin, mutta koodin julkistus teki niiden tekemisen tietenkin paljon helpommaksi ja ehkä myös hauskeemmaksi.

Itse asiassa Quake-huijauksiin liittyvä ongelma oli toisaalla tiedostettu jo paljon aiemmin. Nimittäin kirjeshakin pelaajat olivat jo kauan sitten huomanneet, että ei ole kiva otella kaveriaan vastaan, jos on epäily siitä että tämä käyttää pelissä apunaan tietokonetta! Tämän takia kirjeshakki onkin nykyään kovin harvinaista.

Samalla kun opittiin ymmärtämään huijausten luonne ja syyt paremmin, ymmärrettiin myös, että tämä tapaus oli erityinen juuri peleille. Todellisessa elämässähän me käytämme tietokonetta nimenomaan helpottamaan vaikeita tehtäviä. Sotilaatkin käyttävät tietokonetta apunaan tähtäämiseen, aivan kuten Quake-huijarit, eikä se ole kenenkään mielestä epärealistista. Tai jos haluamme pitäytyä rauhanomaisemmissa esimerkeissä, kukaan ei tietenkään syyttäisi minua huijauksesta, jos käyttäisin tietokoneohjelmaa apunani löytääkseni eBay-verkkohuutokaupasta edullisimman vaihtoehdon jollekin tavaralle, jonka haluan ostaa. Juuri sitä vartenhan tietokoneet on keksitty!

Mutta pelimaailma on erilainen. Pelien ideaan liittyy, että kaikki pelaavat samoilla säännöillä, ilman ylimääräisiä teknisiä apuvälineitä. Jos haluaisin matkustaa mahdollisimman nopeasti 42 kilometrin matkan, niin tekisin sen varmastikin autolla tai vaikkapa helikopterilla. Mutta jos kyseessä olisikin maraton-kilpailu, olisi tietenkin huijausta ajaa matka autolla.

Sitä paitsi se olisi kovin tylsää. Ja tähän tulokseen tulivat loppujen lopuksi myös Quake-pelaajat. Pidemmän päälle huijaaminen on tylsää, sillä se vie pelistä kaiken jännityksen. Erään pelaajan neuvo kuulostaa osuvilta: "Minä pelaan Quakea niinkuin yleensäkin kaikkia pelejä. Vain ystäväni kanssa – sellaisten jotka eivät huijaa."

Tuomio: Tarina Quake-huijauksista on mielenkiintoinen, sillä lopputuloksena oli, että Open Source -malli aiheuttaa sellaisia ongelmia, joita suljetulla ohjelmalla ei käytännössä ole. Tapaus kuitenkin rajoittuu juuri verkkopelaamiseen, tai sanotaanko enemmän "etäpelaamiseen" johon kirjeshakkikin kuuluu. "Todellisen maailman" asioiden parissa toimittaessa vaikuttaisi edelleen siltä, että Open Source -malli johtaa suljettuja ohjelmistoja parempaan tietoturvaan. Lopuksi on vielä todettava, että on kuitenkin olemassa paljon verkkopelejä,⁴³ joiden koodi on avointa, ihan niinkuin edelleen varmaan löytyy niitä, jotka pelaavat kirjeshakkaa.

Kaikissa edellä mainituissa tapauksissa on käynyt ilmi, että aiemmin suljettuna kehitetyn ohjelman lähdekoodin julkistaminen ei olekaan niin ruusuista kuin voisi kuvitella. Monen suljetun ohjelman koodi on yksinkertaisesti niin laadutonta, ettei se kestä päivänvaloa. Kaapista paljastuu usein pahojakin luurankoja. Quaken tapauksessa taas Open Source -mallista seurasi täysin yllättäviä ongelmia, joita kukaan ei ollut osannut odottaa.

Nämä kertomukset asettavat mielestäni vaatimukset esimerkiksi Windowsin lähdekoodin julkistamisesta uuteen valoon. Ties mitä luurankoja sieltäkin löytyisi? Microsoft puolustautui aikoinaan oikeudenkäynnissä koodin julkaisuvaatimuksia vastaan sanomalla, että koodin julkistaminen olisi jopa uhka USA:n kansalliselle turvallisuudelle. Open Source -leirissä naurettiin kun tämä kuultiin Microsoftin omasta suusta. "Sitähän me olemme sanoneet koko ajan, että buginen Windows on uhka meidän kaikkien turvallisuudelle!" Mozillan, Firebirdin ja Quaken tarinoista viisastuneina Microsoftin väite on kuitenkin syytä ottaa vakavasti. Tiivistäen voisi sanoa, että tietoturvan kannalta olisi parempi, että suljettuja ohjelmia ei käytetä, mutta kun niitä nyt kuitenkin käytetään, lienee parempi että ne myös pidetään suljettuina.⁴⁴

Mozilla, InterBase ja Quake edustavat kukin hieman erilaista filosofiaa suhteessa kysymykseen, miksi koodi kannatti julkistaa.

Netscapen johtajat päätyivät julkistamaan Mozillan eräänlaisena puolustuskeinona. Open Source -projektina Mozilla pystyi saavuttamaan sen, mihin Netscape ei kyennyt sitä suljettuna ohjelmistona viemään. Edes Microsoft ei pystynyt murskaamaan avointa koodia. Netscapen tappio oli kuitenkin jo lähellä, koodi avattiin viime hetkessä.

InterBasen tapauksessa taas suojaa hakivat pikemminkin tuotteen asiakkaat. Suljetun ohjelmiston jatkokehityksen lopettaminen on sen käyttäjille aina uhka. Uhka, jolta avoimien ohjelmien käyttäjät välttyvät.

Quaken tapauksessa ei ollut kyse kenenkään suojautumisesta, vaan kyseessä oli ehkä pikemminkin kulttuuriteko. Monia muita pelejä on Quaken jälkeen myös "päästetty vapauteen". Niiden lähdekoodista on iloa faneille, vaikka pelin taloudellinen kukoistusaika on jo kauan sitten mennyt. Esimerkiksi monia pelejä voidaan siirtää uusille alustoille, jos vain lähdekoodi on saatavilla. Vanhat Windows-pelit tulevatkin Linuxiin, tai vanha Commodore 64 -peli Windowsiin. Open Source on siis myös kulttuuriteko!

42En voi välttyä ajattelemasta, että Matrixin käsikirjoittajat olisivat pelanneet Quaken tällaista versiota!

43Yksi suosituimmista Open Source -ammuskelupeleistä on BZFlag: <http://BZFlag.org/>
Sitä voi pelata sekä Windowsilla, Macintoshilla että Linuxilla ja muilla Unix-varianteilla.

4412. helmikuuta 2004 suuri osa Windowsin lähdekoodista vuoti internetiin erään Microsoftin yhteistyökumppanin tietoturvaongelman yhteydessä (<http://slashdot.org/articles/04/02/13/165231.shtml>). Nyt meillä on siis Windowsin kannalta pahin mahdollinen tilanne: Sitä on kehitetty suljettuna ohjelmuna, mistä johtuen siinä on enemmän tietoturva-aukkoja kuin esimerkiksi Linuxissa. Koodi on vuotanut julkisuuteen, minkä takia krakkerit voivat nyt helposti etsiä siitä näitä aukkoja. Mutta toisaalta ne ohjelmoijat, jotka voisivat auttaa Microsoftia vikojen löytämisessä ja korjaamisessa eivät kuitenkaan voi sitä tehdä, koska koodin hallussapito on laitonta. Microsoftia siis uhkaavat nyt kaikki ne vaarat mitä suljetun koodin yhtäkkiseen julkistamiseen voi liittyä, mutta samalla se ei edelleenkään pääse hyötymään Open Source -prosessin positiivisista puolista.

40. VAPAUTETUT IV: STAROFFICE/OPENOFFICE.ORG

Suurin Linux-käyttäjien ja sellaiseksi haluavien ongelma oli pitkin 90-lukua kunnollisen toimisto-ohjelmiston puute. Itse asiassa sellaista ei tainnut olla millekään Unixille edes olemassa. Yliopistojen tutkijat kirjoittivat tekstinsä "koodikielisinä" LaTeXilla tai HTML:llä käyttäen esimerkiksi Richard Stallmanin ohjelmoimaa Emacs-tekstieditoria. Suurin osa varmaankin käytti ihan vaan Windows-tietokonetta ja siinä joko Microsoftin Wordiä tai Corelin WordPerfectiä, ihan niinkuin yliopistomaailman ulkopuolellakin tehtiin. Toimisto-ohjelman – ja erityisesti tekstinkäsittelyohjelman – puuttuminen olikin helposti ratkaiseva este Windowsista Linuxiin siirtymisessä.

Toimisto-ohjelmien puutetta syntyi tietenkin paikkaamaan joitakin Open Source -projekteja. AbiWord ja Kword ovat suosittuja tekstinkäsittelyohjelmia, kun taas Gnumeric ja Kspread ovat Excel-taulukkolaskennan korvaajia ja KPresenter PowerPointin korvaaja. Microsoft Officen korvaaminen ei kuitenkaan ole mikään helppo tehtävä, ja jos totta puhutaan, niin yksikään näistä projekteista ei vielä tänäkään päivänä ole sillä tasolla, että ne pystyisivät tosissaan kilpailemaan MS Officen kanssa.

Ongelma ei ole siinä, etteikö niillä voisi tekstiä kirjoittaa, vaan siinä, että ne eivät kyllin hyvin osaa lukea Microsoftin Wordillä tehtyjä tiedostoja. Aikakautena jolloin paljon tiedostoja lähetetään sähköpostin liitetiedostoina, on välttämätöntä pystyä virheettömästi avaamaan monopoliaseman saavuttaneen Microsoftin tiedostomuotoja. Koska nämä tiedostomuodot eivät ole julkisia standardeja, ei kilpailevan ohjelman tekeminen ole aivan helppoa, vaan vaatii paljon arvaamista, yritystä ja erehdyistä.

Hieman paremmin samaa puutetta paikkasivat suljetut ohjelmistot. Applix oli suosittu toimisto-ohjelmien paketti, jota myytiin aikoinaan monen Linuxin kylkiäisenä. Corelin WordPerfect teki lyhyen Linux-vierailun, ennen kuin Corel lopetti Linux-seikkailunsa äkisti ja jätti WordPerfectin Linux-käyttäjät tyhjän päälle.

Hyvin suosittu oli myös saksalaisen *StarDivisionin* valmistama *StarOffice*. Sen suosiota lisäsi myös se, että ohjelma oli yksityiselle käyttäjälle ilmainen. Lisäksi sitä pystyi Linuxin ohella käyttämään myös Windowsissa. Vaikka StarOffice ei koskaan ole ollut yhtä suosittu kuin WordPerfect tai Word, sillä oli kuitenkin kohtuullisen pitkä ja kunniakas historia ja sen ensimmäiset versiot olivat jo DOS-ajoilta asti.

Elokuussa 1999 Sun ilmoitti ostaneensa StarDivisionin ja sen mukana siis myös StarOffice-ohjelmiston ja ilmoitti samalla, että se aikoi julkaista StarOfficen koodin Open Source -lisenssillä. Koska StarOffice jo muutenkin oli suosittu toimisto-ohjelma Linux-käyttäjien keskuudessa, otettiin uutinen ilolla vastaan.

Sun hoiti StarOfficen Open Source -julkistuksen paremmin kuin Netscape ja Borland omat vastaavansa. Kesti lähes vuosi, ennen kuin mitään koodia oikeasti julkaistiin. Tänä aikana koodia siistittiin, ja esimerkiksi saksankielistä dokumentaatiota käännettiin englanninkieliseksi, jotta avoin koodi olisi mahdollisimman monen hakkerin ymmärrettävissä. Aikaa meni myös sellaisen koodin poistamiseen, joiden tekijänoikeudet olivat kolmansilla osapuolilla, ja joita ei sen takia voitu julkistaa Open Source -lisenssillä. Lopulta koodi julkaistiin heinäkuussa 2000 ja lokakuussa julkistettiin *OpenOffice.org* web-sivut, josta muodostui koodia kehittävä hakkeriyhteisön kotipaikka ja toimisto-ohjelman Open Source -version nimi.⁴⁵

Myöskään StarOffice ei ollut mikään unelmalahja hakkeriyhteisön vastaanotettavaksi, vaan se oli laadultaan verrattavissa InterBaseen ja Netscapeen. Käyttäjälle näkyvin omituisuus oli se, että StarOfficessa kaikki ohjelmat tekstinkäsittelystä ja taulukkolaskennasta aina omaan sähköpostiohjelmaan ja selaimeen asti olivat yksi ja sama ohjelma. Ja jotta ohjelma näyttäisi samalta sekä Linux- että Windows-käyttäjille, oli sen ympärille rakennettu ihan oma käyttöliittymä, joka korvasi oikean käyttöjärjestelmän aina "Käynnistä"-nappulaa ja tiedostonhallintaa myöden. Vaikka tarkoituksena on ollut helpottaa eri käyttöjärjestelmien eroista johtuvia vaikeuksia, on useimpien kohdalla varmaankin käynyt juuri päinvastoin. StarOffice vaikutti tasapuolisen oudolta sekä Windows- että Linux-käyttäjälle.

Myös OpenOffice oli siis alkuun enimmäkseen siivousprojekti. Selaimesta ja sähköpostiohjelmasta luovuttiin kokonaan, sillä näille oli jo olemassa paremmat Open Source -vaihtoehdot. Yhdeksi möhkäleeksi sullottua toimistopakettia eroteltiin erillisiksi tekstinkäsittely-, taulukkolaskenta- jne. -ohjelmiksi. Sunin huolellinen valmistelu ja sitoutuminen projektiin, sekä muun hakkeriyhteisön valtava kiinnostus tähän tärkeään "puuttuvaan lenkkiin" autoivat alkuvaikeuksien yli. Jo vuosi OpenOffice.org-projektin perustamisen jälkeen, saatiin julkaistua versio "Build 638c", joka oli jo suhteellisen vakaa ja käyttökelpoinen testiversio uudesta avoimesta toimisto-ohjelmasta. OpenOffice 1.0 julkaistiin vappuna 2002. Jo silloin Build 638c:n kaltaisia esiversioita oli www-sivuilta käyty lataamassa yli 6 miljoonaa kappaletta.

Linux on jo pitkään ollut hyvä valinta palvelimen käyttöjärjestelmäksi, mutta pitkin 90-lukua siitä ei ollut vakavasti otettavaksi työpöytä tietokoneeksi. OpenOffice on muuttanut kaiken. Jo ennen lopullista 1.0 versiota, alkoivat monet kaupungit ja valtiot selvittämään mahdollisuuksia siirtyä myös toimistojen työpöydillä Linuxia ja OpenOfficea käyttäviin tietokoneisiin. Useat saksalaiset kaupungit Münchenin johdolla ovatkin jo tällaisen päätöksen tehneet. Yrityksmaailmassa esimerkiksi Novell on todennut, että suurimpana Linux-yhtiönä sillä on suorastaan velvollisuus näyttää esimerkkiä muille ja aikoo luopua Microsoftin ohjelmistojen käytöstä pikimmiten.

Siirtymistä Microsoft-monopolista avoimuuden maailmaan helpottaa sekin, että sekä Mozilla että OpenOffice toimivat myös Windowsilla. Näin voidaan siirtyminen toteuttaa helposti siten, että ensin siirrytään käyttämään avointa www-selainta, sähköpostia,⁴⁶ tekstinkäsittelystä ja taulukkolaskentaa, mutta pitäydytään Windowsissa. Kun tämän jälkeen aletaan vaihtamaan kaiken tämän alla olevaa Windowsia Linuxiin, ei käyttäjä oikeastaan huomaa muutosta ollenkaan, vaan jatkaa samojen ohjelmien käyttöä edelleen. Esimerkiksi Novellin siirtyminen tullaan toteuttamaan juuri tällä tavoin. Myös sinä voit helposti ottaa ensimmäisen askeleen kohti avoimempaa tietojenkäsittelyä vaikka heti. Mozillan ja OpenOfficen asentaminen omalle koneellehan ei edes maksa mitään!

OpenOffice oli ratkaiseva edistysaskel Linuxin työpöytäkäytön mahdollistamisessa, mutta lopuksi meidän on vielä esitettävä kysymys, mitä Sun tästä kaikesta hyötyi? Oliko StarDivisionin ostaminen pelkkää hyväntekeväisyyttä vai mahdollisesti kiusantekoa Sunin arkkihollista Microsoftia vastaan?

Hyväntekeväisyydestä ei ollut kysymys, mutta Sunin suhteella Microsoftiin on voinut olla vaikutusta asiaan. Sunin suhde Linuxiin on ollut vaikea. Linux on Sunin oman Solaris-käyttöjärjestelmän pahin kilpailija, ja syö jatkuvasti yhä suuremman osan Solaris-markkinoista. Tästä syystä Sun on kokenut Linuxin uhkana, ja jotkut pääjohtaja Scott McNealyn älyttömimmistä Linux-kommenteista vetävät jopa vertoja Bill Gatesin kuuluisalle "Linux on kommunistinen ja anti-amerikkalainen järjestelmä" -lausaukselle.

Työpöytäohjelmistoissa taas tilanne on päinvastainen, sillä siinä Linux ja Solaris ovat yhdessä altavastaajina Microsoftin haastajia. Solariksen työpöytäohjelmat onkin suurimmaksi osaksi kopioitu Linuxin GNOME:sta, mutta ilman toimisto-ohjelmapakettia se oli kuitenkin vielä puutteellinen. Sun lahjoitti StarOfficen Open Source -liikelle, koska oli ilmeistä, että avoin kehitysmalli oli ainoa mahdollisuus haastaa Microsoft.

Tiedä sitten onko Sunin asenne "pahin vihollisesi on toisaalta paras liittolaisesi" skitsofreenisuutta vai postmodernismia, mutta ainakin OpenOffice on ollut menestys ja se on ollut sitä myös Sunin näkökulmasta. Sun on myynyt omaa StarOffice-versiotaan jo monta miljoonaa, tosin suurin osa näistä varmasti hyvin halvalla, esimerkiksi yliopistoille. Vuoden 2003 lopulla Sun kuitenkin julkisti jymyutisen. Se oli tehnyt Kiinan valtion kanssa sopimuksen, jonka tavoitteena on jopa 200 miljoonan työpöytäkäyttöön tarkoitetun Linuxin myyminen 50 dollarin kappalehintaan. OpenOfficen, kuten myös Mozillan ja monen muun Linuxin työpöytäohjelmiston olemassolo, olivat välttämätön edellytys tällaisen kaupan tekemiselle ja sisältyivät luonnollisesti myytyyn pakettiin. Kiina-kaupan jälkeen yhtiö on tehnyt lisää Linux-kauppoja eri puolille maailmaa, esimerkiksi Isonsa Britannianssa on ollut kysyntää Sunin toimittamille Linux-työpöydille.

Tuomio: OpenOffice on mahdollisesti onnistunein ja varmasti tärkein projekti, jossa vanha suljettu ohjelmisto on julkaistu Open Source -lisenssillä. OpenOffice 1.0:n julkistus poisti viimeisetkin esteet yritysten ja julkishallinnon laajamittaisille Linux-suunnitelmille. Näyttäisi siltä, että OpenOffice tulee osoittautumaan myös Sunin kannalta taloudellisesti onnistuneeksi investoinniksi.

45Sun myy edelleen myös StarOffice nimellä toimisto-ohjelmaa, joka on lähes identtinen OpenOfficen kanssa, mutta sisältää joitakin ei avoimia laajennuksia, kuten fontteja ja Access-tyylisen tietokantaohjelman.

OpenOfficen virallinen nimi on juuri "OpenOffice.org" (lyhennettynä OOo), sillä pelkkä "OpenOffice" oli jo rekisteröity tuotemerkki. Lähes poikkeuksetta käytetään epävirallisissa yhteyksissä nimeä OpenOffice.

46Mozillasta löytyy myös sähköposti. Vaikka Linuxille on monia muitakin sähköpostiohjelmaa, eikä Mozilla ole näistä suosituin, on se kuitenkin ainoa joka löytyy sekä Windowsille että Linuxille.

41. VAPAUTETUT V: ECLIPSE

5.11.2001 IBM julkisti lehdistötiedotteen: "40 miljoonan dollarin arvoinen ohjelmisto lahjoitettu Open Source -yhteisölle". Lehdistötiedotteesta kävi ilmi, että kyseessä oli jonkinlainen Java-kielen ohjelmointiin käytettävä ohjelma, mutta tosiasiaa suurin osa hakkeriyhteisöstä, tai muutenkaan IT-alan väestä, ei ollut koskaan kuullutkaan *Eclipsestä*.

Java-markkinoilla oli jo monta kohtuullisen hyvää ohjelmointiympäristöä. IBM oli tajunnut, että sen oma java-ohjelmointiympäristö, jonka nimi vielä silloin ei ollut Eclipse, ei sen takia ollut mikään järkevä investointi. Sekin oli kyllä ihan kohtuullisen hyvä, mutta oli selvää, että se ei ollut kilpailijoitaan mitenkään ratkaisevasti parempi. Se ei siis tulisi lyömään laudalta kilpakkumpaneitaan, vaan pikemminkin oli suuri riski, että kävisi toisin päin.

Toisaalta IBM ei myöskään halunnut luopua ohjelmointityökalusta, sillä se oli välttämätön osa IBM:n tarjonnassa, jonka varsinaisina päätuotteina olivat kalliit Unix- ja Linux-palvelimet, DB2-tietokanta ja Websphere Java-sovelluspalvelin. Asiakkaille olisi vaikea myydä sovelluspalvelintä, jos ei heille samalla voinut tarjota työkalua jolla sovelluskomponentit palvelimelle ohjelmoidaan. Ohjelmointityökalun hankkiminen joltakin kolmannelta osapuolelta – esimerkiksi Sunilta tai Borlandilta – ei myöskään ollut houkutteleva vaihtoehto. Rahaa valuisi turhaan yhtiöstä ulospäin, ja ennen kaikkea, silloin oli vaarassa joutua liian riippuvaiseksi Borland-Inprise-Inprise/Borland-Borland/Inprise-Borland tyylisten yhtiöiden hassuista päähänpistoista.

Mielenkiintoinen tilanne siis. Toisaalta Eclipseä oli pakko kehittää, mutta toisaalta taas siihen investoiminen oli taloudellisesti kannattamatonta. Tähän oli ratkaisuna Open Source.

IBM oli ymmärtänyt Linuxin ja Open Sourcen idean jo ennen Eclipse-julkistusta. Kun monet muut IT-alan yritykset kokivat ilmaiseksi jaettavien ohjelmistojen vakavaksi uhaksi omalle olemassaololleen, oli IBM:ssä ymmärretty, että työnteke ei loppu, vaikka ihmiset siirtyisivätkin Linuxiin. Itse asiassa IBM:llä oli jo kokemusta Linuxista ja nämä kokemukset olivat sellaisia, että asiakkaat ostivat IBM:n palvelimia ihan yhtä paljon kuin ennenkin ja maksoivat niistä ihan yhtä hyvin kuin ennenkin, vaikka niissä olikin Linux. Ja itse asiassa ihmiset ostivat IBM:n palvelimia enemmän kuin aiemmin, nyt kun niissä kerran oli Linux. Ainoana erona vanhoihin Unix-palvelimiin oli se, että Linuxin kehitys ei ollut yksistään IBM:n varassa. Toisin sanoen työvoimakustannukset laskivat samalla kuin tulot kasvoivat!

Kun tietoisuus Eclipsestä – ja siitä mikä tai mitä Eclipse oikein oli – lisääntyi, se keräsi ympärilleen lisää kehittäjiä ja paljon käyttäjiä. Tällä hetkellä sitä voidaan jo pitää kiistatta markkinoiden parhaana ja suosituimpana Java-kehitysympäristönä. Siitä on siis hyvin nopeasti kehittynyt menestystarina, paljon nopeammin kuin esimerkiksi OpenOfficesta tai Mozillasta. Javan lisäksi sille on kehitetty moduuleja myös C- ja C++-ohjelmointia varten, ja myös moniin muihin tarkoituksiin, kuten esimerkiksi Python-kielen ohjelmointiin. Open Source -kehitysmalli on siis jälleen näyttänyt vahvuutensa ja rikkautensa ja lyönyt laudalta muut kilpailijat.

Siinä missä OpenOffice paikkasi vakavan puutteen Linux-maailmassa, oli Eclipsen tilanne vähän toinen. Java-ohjelmointityökaluista ei ollut puutetta, niitä oli liikaa! Ne olivat kaikki kohtuullisen hyviä, mutta yksikään ei ollut lähelläkään sitä loistavuutta, mitä Eclipse on tänään. Monen erilaisten työkalujen olemassaolo oli haitaksi kaikille Java-markkinoilla toimiville yrityksille. Ohjelmat olivat keskenään epästandardeja, niistä aiheutui turhia koulutuskuluja ja niin edelleen. Kun kaikki toimijat kokoontuivat yhteisen ja avoimen ratkaisun taakse, päästiin vähemmällä vaivalla nopeasti paljon parempiin tuloksiin.

Useimmat Open Source -projektit ennen Eclipseä olivat olleet yksityisten ihmisten – sellaisten kuin Linus Torvalds – yhteistyöprojekteja tai enimmilläänkin yksityisten ihmisten perustamien voittoa tavoittelemattomien kattojärjestöjen – sellaisten kuin Apache Software Foundation – hallinnoimia. Mozilla ja OpenOffice olivat kylläkin vanhojen emoyritystensä hallinnoimia, mutta ne eivät olleet yritysten välisiä yhteistyöprojekteja, vaan niiden osallistujina oli enimmäkseen yksityisiä ohjelmoijia. Eclipsen taakse IBM oli kuitenkin alun alkaen kerännyt vakuuttavan määrän yhteistyöyrityksiä. Jo ensimmäisessä lehdistötiedotteessa mainittiin 150 yhteistyöyrityksen joukosta erityisesti Red Hat, Rational ja TogetherSoft. Pian mukaan liittyi vielä lisää yrityksiä, muun muassa Oracle ja Borland. On rohkaisevaa huomata, että myös yritykset kykenevät organisaatiossa osallistumaan menestyksekkäästi Open Source -projekteihin.

Tuomio: Puolivalmis Eclipse putosi kuin salama kirkaalta taivaalta Open Source -yhteisön syliin ja siitä kehittyi nopeasti erittäin toimiva ja monipuolinen Java-ohjelmointityökalu, joka löi laudalta lukuisat kilpailijansa. IBM on vanhoista IT-yrityksistä oppinut Open Source -maailman säännöt kaikkein parhaiten ja vanhaksi dinosaurukseksi haukuttu "iso sininen" porskuttaa vahvasti eteenpäin kaikilla rintamilla, käsi kädessä Linuxin pingviinin kanssa.

42. VAPAUTETUT VI: BLENDER

D-animaatioita tekevä *Blender* syntyi 1995 hollantilaisen animaatiostudio NeoGeon sisäisenä ohjelmistoprojektina. Se korvasi NeoGeon aiemmin käyttämän animointiohjelmiston, joka myös oli aikoinaan kehitetty yhtiön sisäiseen käyttöön.

Vaikka keskiverto tietokoneenkäyttäjä yleensä mieltää Microsoftin ja muut valmisohjelmistoja tuottavat yritykset ohjelmistotuotannon ruumiillistumaksi, on tosiasia kuitenkin se, että enemmistö maailman ohjelmoijista tekevät juuri Blenderin kaltaisia yritysten sisäisiä ohjelmistoja. Ohjelmoijia on töissä kaikkialla, pankeista ja vakuutuslaitoksista aina elokuvastudioihin asti, koska se mitä näissä yhtiöissä tarvitaan, ei ole ostettavissa kaupan hyllyltä.

Koska Blender oli niin loistava ohjelma, sen tekijä *Ton Roosendaal* päätti ulkoistaa ohjelman kehityksen erilliseen yhtiöön, jonka nimeksi tuli Not a Number. NaNin oli tarkoitus lisensoida Blenderiä myös muiden animaatiostudioiden käyttöön ja saada tuloja tämän ympärille tarjottavista konsultointipalveluista. Uutta Blenderiä esiteltiin animaatioalan Siggraph-konferenssissa ja se herätti heti suurta kiinnostusta. Pian tämän jälkeen yhtiö onnistuikin keräämään riskisijoittajilta pääomaa yhteensä 4,5 miljoonaa euroa ja kohta Blenderin parissa uurasti jo 50 täyspäiväistä ohjelmoijaa. Ensimmäinen versio – jonka versionumero tosin oli jo 2.0 – julkaistiin kesällä 2000 ja vuoden loppuun mennessä Blenderillä oli jo 250 000 rekisteröitynyttä käyttäjää.

Valitettavasti joku kuitenkin meni pieleen ja pääoma ehdittiin kuluttamaan loppuun ennen kuin oli edes kunnolla päästy vauhtiin. Huhtikuussa Blenderin taakse saatiin kuitenkin uusia sijoittajia ja kehitystä jatkoi uudestisyntynyt, mutta huomattavasti supistettu NaN-yhtiö. Tämä uusi NaN-yhtiö julkaisi puoli vuotta myöhemmin Blender Publisherin, joka oli ajan henkeen sopivasti ja muodikkaasti suunniteltu kolmiulotteisen interaktiivisen median julkaisemiseen webissä. Tuotteen myyntiluvut olivat kuitenkin pettymys ja niin nämäkin sijoittajat päättivät lopettaa yhtiön tukemisen. Yhtiö lopetti kaikki toimintonsa ja sen myötä loppui myös Blenderin jatkokehitys.

Vaikka Blender ei ollut valloittanutkaan maailmaa, oli sen ympärille kuitenkin syntynyt jo uskollinen käyttäjien joukko. Näiden käyttäjien palautteesta rohkaistuneena, Ton Rosendaal päätti yrittää Blenderin pelastamista vielä kolmannen kerran. Ja tällä kertaa hän onnistui.

Ton perusti Blender Foundation -nimisen säätiön ja teki sopimuksen Blenderin oikeudet omistavien sijoittajien kanssa. Sopimuksen mukaan sijoittajat myisivät Blenderin säätiölle 100 000 eurolla GPL-lisenssin ehdoilla, joka voisi siten jatkaa Blenderin kehittämistä Open Source -kehitysmallin periaattein.

Monta miljoonaa menettäneille riskisijoittajille tarjous oli tietenkin 100 000 euroa parempi kuin ei mitään. Heillä ei ollut enää mitään hävittävää. Tonillekaan sopimus ei sinällään sisältänyt mitään riskiä, mutta miten hän saisi koottua vaaditut 100 000 euroa?

Tonin ja kaikkien muiden Blenderiä seuranneiden yllätykseksi ja järkytykseksi Ton ja muut NaN-yhtiön entiset työntekijät saivat summan kerättyä vain seitsemässä viikossa! Sunnuntaina 13. lokakuuta 2002 Blender luovutettiin GPL-lisenssillä Open Source -yhteisön huolehtiviin käsiin.

Open Source -ohjelmistona Blender heräsi uudelleen henkiin. Uusia versioita on jo ehditty julkaisemaan useampia. Käyttäjämäärä on kasvussa ja ohjelman ympärille on syntynyt oma yhteisönsä. Tuhannet käyttäjät osallistuvat keskusteluihin Blenderin [www-sivuilla](http://www.blender.org) ja [irc:ssä](http://irc.blender.org). Blenderin [www-sivuilta](http://www.blender.org) on linkkejä useille sivuille joilla voi käydä ihailmassa Blenderillä tehtyjä kuvia ja lyhytelokuvia. Ohjelmaa voidaan käyttää myös pelien tekemiseen ja joitakin Open Source -pelejä on jo ohjelman pelimoottorin varaan ehditty rakentamaan.

Myös rahoituspuolella Blenderin tekijät ovat siirtyneet perinteisestä riskisijoitusmallista Open Source rahoitusmalliin. Blender Foundationin vuoden 2003 varainkeruukampanjana myytiin uutta Blender-opaskirjaa 3000 kappaleen verran. Vuoden 2004 alussa taas säätiö sai EU:lta tutkimukseen ja tuotekehitykseen apurahoja 140 000 euron verran. Näiden pienten ja suurten tulojen, sekä lukuisten projektiin osallistuvien vapaaehtoisten hakkereiden ansiosta Blenderin tulevaisuus näyttää nyt ruusuisemmalta kuin koskaan.

Tuomio: Tässä luvussa olemme monien esimerkkien kautta pohtineet, onko Open Source -kehitysmallin varaan mahdollista rakentaa toimivaa liiketoimintaa. Monista esimerkeistä huolimatta et ehkä vielä ole täysin vakuuttunut, tai ajattelet korkeintaan, että Open Source voi olla suljetun kehitysmallin rinnalla "ihan ok, ihan toimiva" malli. Tarina Blenderistä kuitenkin pakottaa meidät kysymään asiaa toisin päin. Mitä vikaa on suljetussa ja riskisijoitusvetoisessa "perinteisessä" ohjelmistokehityksessä, joka kahdesti epäonnistui Blenderin kanssa ja melkein tappoi loistavan tuotteen? Tuotteen jolla kuitenkin oli edessään loistava tulevaisuus – kunhan vain löydettiin ensin oikea liiketoimintamalli!

43. TIE PÄÄTTYY, MATKA ALKAA (LINUX WEEKLY NEWS)

"The end of the road" (tie päättyy) otsikoi *Linux Weekly News*⁴⁷ yllättäen etusivullaan 25. heinäkuuta 2002. Vuonna 1998 aloittanut Linux Weekly News oli ehtinyt vakiinnuttaa asemansa Linux-yhteisön luotettavimpana, asiantuntevimpana ja analyttisimpänä verkkojulkaisuna. Vaikka se oli nyttemmin siirtynyt Linux Today'n ja Slashdotin tapaan julkaisemaan myös päivittäisiä linkkejä eri Linux-uutisiin, sen tukijalka oli kuitenkin viikottaisessa omia artikkeleja sisältävässä numerossa.

Heinäkuun viimeinen numero sisälsi tavalliseen tapaan Open Source -maailmaan liittyviä artikkeleita: RealNetworksin ilmoituksesta siirtyä Open Source -kehitysmalliin; USA:ssa ehdotetusta lainsäädännöstä, joka antaisi elokuvastudioille ja levy-yhtiöille oikeuden ilman minkäänlaista polliin tai oikeuden antamaa päätöstä häiritä ja mahdollisesti myös murtautua vapaaviltoisten internet-käyttäjien tietokoneisiin; tietoturvaongelmista PHP:ssa ja SSH:ssa; Linux kernelin cli() ja sti() funktioista sekä Ogg Vorbis 1.0:n ja Debian GNU/Linux 3.0:n julkaisusta. Ja sitten, aivan yllättäen, oli etusivun lopussa juttu otsikolla "The end of the road".

Linux Weekly News, tuttavallisesti LWN, oli tullut siihen tulokseen, ettei bannerimainonta ollut kestävä tapa rahoittaa ilmaista verkkolehdeä.⁴⁸ LWN oli myös jonkun aikaa vastaanottanut lukijoiltaan lahjoituksia PayPal-järjestelmän kautta, ja vaikka näistä saadut tulot olivatkin jonkin verran mainostuloja suuremmat, ei niistäkään ollut viisihenkeisen toimituksen elättämiseksi. Harkittuaan vielä mahdollisuutta LWN:n muuttamisesta maksulliseksi verkkolehdeksi, olivat LWN:n tekijät lopulta päättäneet laittaa lapun luukulle. Ensimmäinen elokuuta oli ilmestynvä viimeinen LWN. Mitään ei ollut enää tehtävissä ja päätös oli lopullinen.

Lukijoiden reaktio⁴⁹ ilmoitukseen yllätti kaikki, niin toimittajat kuin lukijat itsensäkin. Luonnollisestikin arvostetuimman Linuxiin keskittyvän journalistisen julkaisun loppuminen oli kaikille suuri pettymys, jopa shokki. Mutta se millaisia konkreettisia ilmenemismuotoja tämä pettymys sai, oli suorastaan historiallista.

Kirjoittanut 25. heinäkuuta 2002 20:26 UTC (to) other-iain:
Olen lukenut LWN:ää noin 4 vuotta. Arvioisin, että sopiva tilaushinta olisi ollut 30 dollaria vuodessa. Se tekee yhteensä 120 dollaria. Olen jo lahjoittanut 20 dollaria aiemmin, joten 100 on vielä maksamatta. Laitoin juuri nuo 100 dollaria tulemaan PayPalin kautta. Ne ovat maksu saamistani palveluista. olen pahoillani, että lopetatte, mutta ymmärrän syyn. [...] Onnea LWN:n jälkeisiin töihin. Ilmoitelkaa mihin kirjoitatte seuraavaksi, jotta voimme olla kuulolla.

Kirjoittanut 26. heinäkuuta 2002 0:46 UTC (pe) BogusUser:
Minäkin rekisteröidyin antaakseni lahjoituksen ja kiittäkseni loistavasta työstänne menneinä vuosina. Jos nämä ylimääräiset lahjoitukset eivät ole tarpeeksi, niin voittehan ainakin juoda oluet minun laskuuni.

Kirjoittanut 26. heinäkuuta 2002 14:19 UTC (pe) jgm
Toivottavasti jokin ihme tapahtuu. Olen lahjoittanut 100 dollaria minäkin. Toivottavasti siitä on apua, että pystyisitte jatkamaan. Jos ei, niin kyseessä on myöhästynyt :(tilauseräni loistavista julkaisuista.
LWN on ollut paras paikka seurata GNU/Linux, Free Software ja Open Source Software -maailman tapahtumia. Tulen todella kaipaamaan teitä! Olen erityisesti nauttinut Kernel-sivun artikkeleista. Torstaipäiväni eivät tule enää olemaan

samoja ilman viikottaista LWN:ää!
Oli miten oli, parhaat onnentoivotukset teille kaikille ja kiitokset loistavasta työstänne LWN:n parissa. Toivottavasti löydätte vielä tavan jatkaa.

Seuraavan viikon numeroon mennessä lahjoituksia oli tullut yhteensä 25 000 dollaria! Vaikka tämäkään ei ollut viiden hengen yritykselle kuin puolentoista kuukauden liikevaihto, se sai kuitenkin toimituksen miettimään lopettamis päätöstään uudestaan. Viikon tauon jälkeen LWN päätti sittenkin siirtyä maksulliseksi verkkojulkaisuksi. Päivittaiset uutislinkit ja niiden keskustelufoorumit säilyisivät kaikille avoimina, mutta viikottainen omia artikkeleja sisältävä julkaisu olisi avoin vain tilaajille ja tulisi maksamaan 5 dollaria kuussa. LWN kuitenkin kammosi suljettua mallia niin paljon, että toimittajat päättivät lisäksi, että kaikki viikkoa vanhemmat artikkelit olisivat luettavissa myös muille kuin tilaajille.

Vaikka LWN joutuikin hieman vähentämään vakituisesti kirjoittavien toimittajien lukumäärää ja siirtymään osittain freelancereiden käyttöön, se pystyi tämän jälkeen jatkamaan käytännössä täysin entisellään. Vuoden 2003 aikana LWN osoitti kyntensä, kun SCO väitti, että Linux sisälsi SCO:lta laittomasti kopioitua koodia. Kun SCO:n valheet aluksi erottaa jyvät akanoista heti alkuunsa. Kun SCO julkaisi kaksi esimerkkiä "Linuxiin laittomasti kopioituista" koodinpätkistä, pystyi Bruce Perens yhdessä LWN:n lukijoiden kanssa alle päivässä jäljittämään molempien koodinpätkien todellisen alkuperän ja todistamaan, että toinen oli peräisin jo 70-luvulta ja toinen olikin ilmeisesti – ironista kyllä – SCO:n laittomasti kopioimaa Open Source -yhteisön kirjoittamaa koodia.

Tutkiva journalismi oli osoittautunut hintansa arvoiseksi.

Tuomio: Open Source -yhteisöä on joskus kritisoitu IT-alan vapaamatkustajiksi. Paljon on puhetta vapaudesta, mutta mistään ei olla valmiita maksamaan. Viimeistään LWN:n tapaus osoitti nämä puheet perättömiksi. Linux-käyttäjät eivät ehkä ole valmiit luovuttamaan rahojaan monopolille, jonka ohjelmien hinnasta 90% on ilmaa. Missään tapauksessa he eivät ole valmiita luopumaan perustavista vapauksistaan, kuten lähdekoodin vapaasta jakamisesta. Mutta he ovat valmiita maksamaan jostain, jolla oikeasti on merkitystä heille. 25 000 dollaria LWN:n tiillä todistaa sen.

47 <http://lwn.net>

48 Ei ehkä kuulosta järin yllättävältä johtopäätökseltä, mutta muista että sellaisetkin asiat kuin bannerimainonta tuntuivat kultaisella 90-luvulla fiksuilta ideoilta.

49...joka on edelleen luettavissa artikkeliin liittyvistä kommentteista osoitteessa <http://lwn.net/Articles/5409/>

44. KIITOS, MUTTA ME HALUAMME MAKSAA (MANDRAKE CLUB)

Koska olen itsekin suorittanut siviilipalveluksen, niin tarina Mandrake Linuxin synnystä on aina kiehtonut minua. MandrakeSoftin perustaja *Gaël Duval* on kertonut, että idea omasta Linux-jakeluversiosta syntyi vuonna 1997, kun hän suoritti siviilipalvelustaan ja hänellä siksi oli paljon vapaa-aikaa. Kuulostaa jotenkin tutulta, mutta eipä vaan itselleni tullut mieleen perustaa tekemisen puuttessa yritystä, joka osaltaan tulisi muuttamaan maailmaa.

Neljästä suuresta Linux-brändistä *Mandrake* on selvästi nuorin. Gaël julkisti ensimmäisen Mandrake Linuxin kesällä 1998. Hän oli ihastunut KDE-työpöytäympäristöön, josta oli juuri julkaistu versio 1.0. Silloinen markkinajohtaja Red Hat ei kuitenkaan ollut sisällyttänyt KDE:tä omaan Linuxiinsa, koska sillä oli poliittisia erimielisyyksiä KDE:ssä käytetyn Trolltechin Qt-kirjaston lisensoinnista.⁵⁰ Gaël otti Red Hatin silloisen 5.1 version, lisäsi siihen KDE:n ja teki joitain muitakin lisäyksiä. Hän pisti kokoamansa paketin levitykseen julkiselle ftp-palvelimelle, antoi sille nimen *Linux-Mandrake 5.1* ja lähti sen jälkeen parin viikon kesälomalle.

Palatessaan lomiltaan hänen postilaatikkonsa pursusi innostunutta palautetta! Oli selvää, että "Red Hat with KDE" oli osunut jonkinlaiseen tarpeeseen. Yhdessä Frédéric Bastokin ja Jacques Le Maroisin kanssa Gaël perusti MandrakeSoft-nimisen yhtiön jatkamaan työtä Mandrake Linuxin parissa ja jo vuonna 1999 Mandrake oli yksi suosituimpia Linux-versioita, voitti useita palkintoja ja kunniamainintoja ja ylsi lopulta vuonna 2000 USA:n myyntiluvuissa tasoihin markkinajohtaja Red Hatin kanssa.

Vuoden 1999 kolmas vuosineljännes tuotti jo sadantuhannen euron liikevoiton. Aika hyvin pieneltä yksivuotissyntymäpäiväään viettävältä yhtiöltä! Mutta tämä tulisikin jäämään viimeiseksi voitolliseksi sarakkeeksi Mandraken kirjanpidossa pitkään aikaan.

Koska elettiin "uuden talouden" huumaa kuumimmillaan, Mandrake sai nauttia riskisijoittajien yhtiöön karräämistä rahakasoista. Gaël puhui lehtihaastatteluissa Nasdaqiin listautumisesta. Uusien sijoittajien mukana yhtiöön tuli myös uusi johto, jonka tehtävänä oli pistää sijoitetut rahat pyörimään. Yhtiöön palkattiin runsaasti uusia ohjelmoijia siivittämään yhtiön suosittua Linuxin kehitystä. Lisäksi rahaa sijoitettiin uusille toimialoille, muun muassa Linux-koulutukseen ja varsinkin eLearningiin. Verkko-opiskelu olikin vuoden 2000 suurimpia kiinnostuksen kohteita "uuden talouden" liikemiesten joukossa.

"Kokenut ja ammattitaitoinen maailmanluokan johto" – näin Gaël on sarkastisella äänensävyllä päätynyt jälkeensä silloista johtoa nimittelemään – teki työtä käskettyä. Mandraken liikevaihto kasvoi räjähdysmäisesti, mutta valitettavasti lähinnä menopuolella. Vain vuosi ensimmäisen voitollisen neljänneksen jälkeen yhtiö teki kolmen kuukauden kirjanpitojaksolla jo 3 miljoonaa euroa tappiota ja seuraavana vuonna lukema oli saatu nostettua jo 5 miljoonaan! Luku on hyvä suhteuttaa siihen, että saman vuosineljänneksen tulot olivat kolmesataatuhatta euroa.

Kun uusi johto vielä ehdotti "kulujen karsimiseksi" oman Linux-version kehityksen lopettamista ja keskittymistä pelkästään eLearningiin, oli alkuperäisten omistajien mitta täysi. Ulkoa yhtiöön tullut johto sai lähteä, ja Gaël otti taas ohjat käsiinsä.

Red Hat oli alusta asti ollut hyvin uskollinen Free Software ideologialle ja julkaissut kaikki omat ohjelmistonsa GPL-lisenssin ehdoilla, mutta Mandrake pani vielä siitakin paremmaksi. Red Hatilla oli tapana julkaista näytteitä tulevista Linuxeistaan "Rawhide"-nimellä kulkevana pakettina. Mandrake sen sijaan avasi koko kehitysprosessinsa siitä kiinnostuneille ulkopuolisille hakkereille. Kuka tahansa saa osallistua Mandrake Linuxin tulevaisuuden suunnitteluun sitä varten tarkoitetuilla sähköpostilistoilla. Hyvien ideoiden lisäksi vastaanotetaan myös vapaaehtoisten tekemää koodia, vaikka yhtiön omat työntekijät tietenkin tekevät suurimman osan työstä.

Kuulostaako ajatus utopistiselta? T ekisitkö sinä ilmaiseksi jotain ja luovuttaisit sen yhtiölle, jotta voit sitten ostaa tekemäsi tuotteen heiltä takaisin? Ei kuulosta kovin uskottavalta ajatukselta, mutta niin vaan Cooker-projektissa oli jo ensimmäisenä vuonna mukana yli 200 innokasta hakkeria. Tiedä sitten onko "Cooker" niin onnistunut nimi tällaiselle projektille – yleensähan sanotaan että mitä useampi kokki, sitä huonompi soppa – mutta sen ansiosta Mandrake Linux on aina ollut hyvin lähellä käyttäjiään ja aina pyrkinyt vastaamaan heidän tarpeisiinsa. Ja juuri tästä Open Sourcessa on kysymys.

Mandraken strategiaan kuului alusta asti myös jokaisen uuden Mandrake-version kohdalla julkaista netistä ladattavat tiedostot heti kun ne olivat valmiit.⁵¹ T ässä se erottui myös edukseen Red Hatista, joka siihen aikaan pimitti netistä ladattavan version julkaisemista siihen asti, että CD-levyt olivat ehtineet kauppojen hyllyille. Gaël selitti lukuisissa haastatteluissa, että hänen mielestään oli yhtiölle eduksi, että uusi versio oli mahdollisimman nopeasti ihmisten käytettävissä, sen sijaan että sitä keinoitekoisesti pihdattaisiin monta kuukautta. Sitä paitsi strategia oli tietenkin myös avoimuuden periaatteen mukainen.

Strategia varmasti osaltaan vaikutti siihen, että Mandraken suosio kasvoi niin nopeasti kuin teki. Mutta oliko strategiasta sitten haittaa jälkijunassa tulevien fyysisten CD-levyjen myynnille? Varmasti jonkin verran olikin.

Uskollisimmat käyttäjät kuitenkin ymmärsivät vastavuoroisuuden periaatteen ja marssivat kauppaan ostamaan CD:t pelkästään tukeakseen suosikki-Linuxinsa tulevaisuutta. Itse CD:illä he eivät tehneet mitään, olivathan he käyttäneet netistä ladattavaa versiota jo useamman kuukauden. He nyt vaan sattuivat ymmärtämään, että ilman myyntituloja heidän suosikki-Linuxinsa ei kauaa pystyisi jatkamaan. Sitä paitsi, kyllähän näin hyvästä tuotteesta maksoi ihan mielellään.

T ästä vastavuoroisuusperiaatteesta kehittyi vähitellen ajatus uudenlaisesta liiketoimintamallista: *Mandrake Clubista*. CD:t ovat hitaita valmistaa ja monet, jotka niitä ostavat, eivät oikeastaan edes tarvitse niitä vaan ostavat niitä ainoastaan tukeakseen Mandrakea. Miksi siis antaa heille CD:tä ollenkaan, kun senkin valmistaminen kuitenkin maksaa? Mandrake Club on Mandraken käyttäjien kerho, tai oikeastaan eräänlainen "tilausmaksu", vähän samaan tapaan kuin lehteäkin tilataan. Kerhoon voi liittyä maksamalla eritasoisia vuosimaksuja, 60 dollarista aina 1200 dollariin vuodessa. Useimmat käyttäjät maksavat joko 60 tai 120 dollarin vuosimaksun.

Kerhoon kuulumisesta seuraa myös joitain käytännön etuja. Kerhon salasanaa käyttämällä voi helposti ladata itselleen erityisesti Mandrakeen sovitettut versiot Javasta, RealPlayeristä, Flashistä, Acrobatista ja muista vastaavista ohjelmista, jotka ovat ilmaisia mutta eivät kuitenkaan vapaasti levitettävissä. Ilman kerhoa, Mandraken käyttäjä joutuu surffailemaan ympäri Internetiä keräämässä samoja ohjelmia, ja asentamaan jokaisen yksitellen. Erittäin suosituksi on osoittautunut kerhon jäsenilleen tarjoama "RPM Voting" – mahdollisuus äänestää siitä, mitä Open Source -ohjelmia seuraavaan Mandrake Linuxin versioon tulisi sisällyttää. Näin myös ne Mandraken käyttäjät, joiden ohjelmointikyvyt eivät ole Cookeriin tarvittavaa tasoa, voivat vaikuttaa oman Linuxinsa tulevaisuuteen. Mutta näiden ja joidenkin muiden verrattaen pienien hyödykkeiden lisäksi, ei jäsenmaksun maksamisesta ole mitään varsinaista hyötyä.

Tiivistäen Gaël Duvalin argumentti Mandrake Clubin puolesta on siis seuraava: Free Software -ideologian periaatteita noudattaen, me haluamme antaa teille Mandrake Linuxin täysin vapaasti, ilman mitään ylimääräisiä koukkuja. Me haluamme myös antaa tuotteillemme tietoturvapäivitykset täysin vapaasti ja ilmaiseksi, ilman mitään ylimääräisiä koukkuja. Mutta pidemmän päälle emme pysty tekemään tätä, ellei joku maksa työntekijöidemme palkkaa. Pyydämme siksi, että tukisitte Mandrake Linuxia, jonka olemme teille ilmaiseksi antaneet, liittymällä Mandrake Clubiin.

Kuulostaako taas hieman utopistiselta? Etkö usko että joku haluaisi maksaa siitä, että saa kuulla kerhoon, johon kuulumisesta ei ole mitään hyötyä? Aivan kuten Linux Weekly Newsinkin tapauksessa, Mandrake Club on osoittanut, että Linux-käyttäjät ovat valmiita maksamaan ilmaiseksi saamastaan palvelusta, silloin kun palvelu tai tuote on hintansa arvoinen. Vuonna 2003 Mandrake Clubissa oli jo noin 14 000 aktiivista jäsentä ja lisäksi muutama yritysjäsen (joiden jäsenmaksut ovat kalliimpia).

Tämä on pieni määrä jos sitä verrataan siihen, että Mandrake Linuxilla on kaiken kaikkiaan useampi miljoona käyttäjää. Mutta jos nämäkin 14 000 käyttäjää ovat maksaneet keskimäärin 120 dollarin jäsenmaksun, tulee siitä jo 1,68 miljoonaa dollaria vuodessa, mikä ei MandrakeSoftin kokoiselle yritykselle ole mikään pieni summa! Mikä kaikkein parasta, nämä tulot menevät sataprosenttisesti yhtiön ohjelmoijien palkkojen maksuun, sen sijaan että niillä valmistettaisiin CD-levyjä, joita niiden ostaja ei yleensä edes tarvitse.

Vaikka yrityksen lähestulkoon täydelliseen tuhoon ajaneesta johdosta oltiin päästy eroon, ei kaikista vanhoista sitoumuksista päästy irti yhtä helposti. Vanhan johdon tekemiin eLearning-sopimuksiin liittyi maksuvelvoitteita vuosiksi eteenpäin ja vaikka MandrakeSoft saneerasikin toimintaansa rajusti, jouduttiin vanhoista synneistä kärsimään vielä monta tappiollista vuosineljännestä. Mutta Gaëlin pitkäaikainen uskollisuus avoimuuden periaatteelle oli poikanut Mandrakelle joukon uskollisia käyttäjiä, joiden avulla yritys tulisi selviämään näiden vaikeiden aikojen yli.

Kaikkein pahimpina aikoina Mandrake joutui käytännössä kerjäämään lahjoituksia asiakkailtaan. Mandrake ei ikinä listautunut Nasdaqiin, mutta loppuvuonna 2001 osakkeita myytiin vähän tuntemattomamman ranskalaisen Euronext-pörssin kautta yhtiön uskollisimmille tukijoille, omille asiakkaille. Ja pidemmän päälle he ehkä osoittautuisivat paremmiksi omistajiksi kuin "maailmanluokan" riskisijoittajat. Kun osakeannista kerätty pääoma oli lopussa ja velkoja ja maksusitoumuksia riitti aina vaan, joutuivat yhtiön perustajat taas vetoamaan käyttäjiin, jotka taas riensivät ostamaan klubijäsenyyksiä ja muita yhtiön tuotteita. Yhtiö vetosi asiakkaidensa tukeen vielä joulukuussa 2002 ja vielä kolmannen kerran asiakkaat riensivät apuun.

Ensin Mandrake Club ja sen päälle vielä nämä varainkeruukampanjat ja aina vaan Mandrake tuntui löytävän uskollisuutta asiakkunnastaan. Kovien arvojen IT-alalla moni yritys tuskin uskaltaisi toivoa asiakkailtaan näin paljoa, mutta Mandraken tapauksessa pätee sanonta: sitä näyttää, mitä kylvää. Yhtiö oli alusta asti toiminut avoimesti ja reilusti asiakkaitaan kohtaan, nyt asiakkaat auttoivat sitä selviytymään.

Avusta huolimatta MandrakeSoft joutui hakeutumaan tammikuussa 2003 velkasaneeraukseen, suojautuakseen kuukauden sisällä lankeavista viimeisistä eLearning-veloista. Reilu vuosi myöhemmin keväällä 2004 velkasaneeraus saatiin päätökseen ja mikä parasta, yhtiö alkoi myös lähes viiden vuoden tauon jälkeen tekemään taas voittoa. Red Hatin hylättyä kotikäyttäjät keskittyäkseen kalliisiin yrityspalvelimiin, moni entinen Red Hat -käyttäjä on päätenyt Mandraken asiakkaaksi ja yhtiö näyttää pitkästä aikaa taas vahvalta Linux-pelurilta. Saneerausohjelman jälkeiset tilinpäätökset ovat kaikki olleet plussan puolella ja vuoden 2004 yhtiö teki pari isoa Linux-kauppaa Ranskan kahden ministeriön kanssa, laajentaen näin kotikäyttäjien suosikkilinuxista myös rahakkaampiin yrityssopimuksiin.

Tuomio: MandrakeSoft on ollut kaupallisista Linux-jakeluista avoimin ja on vahvasti sitoutunut Free Software -ideologiaan. Yhtiö on myös aikansa lapsi, Internet-kuplan vuodet olivat tähdenlentoa Mandrakelle. Mandrake Club -strategiallaan yhtiö on määrätietoisesti rakentanut samanlaista liiketoimintamallia minkä Linux Weekly News tuli myöhemmin löytäneeksi puolivahingossa. Jos olet avoin, reilu, ystävällinen ja ennen kaikkea asiakkaillesi hyödyksi, he ovat valmiita maksamaan ja tukemaan sinua vaikeidenkin aikojen yli.

50Qt:tä ei silloin vielä levitetty GPL-lisenssin ehdoilla, mistä syystä koko KDE-projekti sai useaksi vuodeksi Richard Stallmanin Free Software Foundationin vihan ylleen.

51Yhtiö poikkesi tästä periaatteestaan syksyllä 2003 version 9.2 julkistuksen yhteydessä. Mandrake 9.2 oli ensin saatavilla vain Mandrake Clubin jäsenille ja täysin vapaaseen levitykseen se tuli vasta samaan aikaan myytävien CD-levyjen kanssa.

45. TALKOOT JA PYYKKINARUPARADOKSI (DEBIAN)

Tässä luvussa olemme tarkastelleet, miten Open Source -maailmassa toimivat yritykset selviävät taloudellisesti. Muiden joukossa olemme tarkastelleet neljästä tunnetuimmasta Linux-brändistä kolmea: Red Hatia, SUSE:a ja Mandrakea. Luku ei olisi täydellinen ilman neljättä: *Debiania*.

Debian ei ole yritys. Voiko sitä siis tarkastella liiketoimintamallina? Ehkä ei. Mutta emme voi ohittaa sitä tosiasiaa, että Debian on kuitenkin olemassa. Se ei ainoastaan ole olemassa, se on yksi vanhimpia ja suosituimpia Linux-versioita ja hakkaa monissa tilastoissa kilpailevat Linux-jakelut mennessään.

Oletko koskaan asentanut Windows-konetta, tai katsellut jonkun toisen asentavan sellaista? Kun olet asentanut Windowsin, sinulla on toimiva tietokone,⁵² mutta ei sen enempää. Seuraavaksi sinun pitää asentaa tekstinkäsittelyohjelma (valitse vaikka OpenOffice) ja sähköpostiohjelma (asenna vaikka Mozilla). Mutta tämä on vasta alkua. Mitäs jos tarvitset kuvankäsittelyohjelmaa? Entä toimiiko printterisi? Ja aiotko pelata pelejä? Nämä kaikki pitää asentaa erikseen ja hankkiakin eri paikoista, kun käytät suljettuja ohjelmistoja.

Sen sijaan jos käytät Linuxia ja vapaita ohjelmistoja, kaikki voidaan paketoita kätevästi samaan pakettiin. Kun olet asentanut Linuxin, olet samalla asentanut satoja muita hyödyllisiä ohjelmia, muun muassa OpenOfficen ja Mozillan. Koska ohjelmia saa vapaasti levittää, on Linux-myyjäsi – oli se sitten Red Hat, SUSE, Mandrake tai Debian tai joku muu – voinut koota sinulle kaiken tarvittavan yhteen pakettiin. Ja Linux-tietokoneesi on täysin käyttökunnossa alle tunnissa.

Ja mitattuna mukana tulevien ohjelmien määrällä Debian on ylivoimaisesti suurin: 8710 erilaista ohjelmistoa!⁵³ Ja tämän kaiken voi siis asentaa itselleen ilmaiseksi verkosta tai vaihtoehtoisesti DataClubista saatavilla noin 8 euroa maksavilta CD-levyiltä. 8710 erilaista ohjelmistoa, koottuna nästisti yhteen helposti asennettavaan ja toimivaan pakettiin.

Muut Linux-versiot ovat yleensä saatavilla vain Intelin valmistamille, niin sanotuille x86-arkkitehtuurin tietokoneille ja lisäksi ehkä jollekin toiselle arkkitehtuurille kuten PowerPC (Apple-tietokoneet) tai IA-64 (Intelin uusi 64-bittinen arkkitehtuuri) tai AMD:n Opteronille. Ainakin Red Hat ja SUSE tukevat näiden lisäksi muun muassa IBM:n suuren arkitehtuureja. Debian on kuitenkin tässä suhteessa omaa luokkaansa. Se toimii yhdellätoista eri arkkitehtuurilla!⁵⁴

Debian-projektissa on tällä hetkellä lähes 1000 aktiivista ohjelmoijaa, eli hakkeria. Suurimmalla Linux-yhtiöllä Red Hatilla, on noin 600 työntekijää ja heistäkin vain noin 300 ovat insinöörejä.⁵⁵

Debian on ei-kaupallinen organisaatio, johon kuuluu Debiania tekevien ohjelmoiden lisäksi muun muassa Debianin www- ja tiedosto-palvelinten ylläpitäjiä ja tietenkin joku, joka on lahjoittanut myös itse palvelimet ja niiden internet-yhteydet, tai ainakin niihin tarvittavat varat. Sitten on lainopillisia neuvonantajia – hekin luonnollisesti vapaaehtoisia – ja käyttöoppaiden kirjoittajia ja käännöstyötä tekeviä.⁵⁶ Ja tietenkin näin isoon projektiin kuuluu aina hieman byrokratiaa – jopa Open Source -maailmassa – joten Debianilla on myös vaaleilla valittuja johtajia.

Debianin perusti vuonna 1993 *Ian Murdock*. Linus ja kumppanit olivat tuossa vaiheessa saaneet kernelinsä niin hyvään kuntoon, että Linuxia ja sen päällä ajettavia muita vapaita ohjelmistoja pystyi jo käyttämään. Nämä erilliset vapaat ohjelmointiprojektit olivat kuitenkin hajan hajan ympäri Internetiä, eikä kukaan ollut vielä koontanut niistä yhtenäistä käyttöjärjestelmätuotetta. Vuonna 1993 syntyivät kuitenkin ensimmäiset Linux-distribootit, eli Linux-jakeluversiot ja Debian oli yksi niistä.

Koska Linuxin kerneli, bash-komentotulkki, gcc-kääntäjä, emacs-editori ja kaikki muutkin täydelliseen käyttöjärjestelmään tarvittavat enimmäkseen GNU-projektin valmistamat työkalut oli kehitetty avoimella kehitysmallilla, oli hänelle luonnollista, että myös hänen Linux-jakeluversionsa tulisi olemaan avoin. Kaupalliset Linux-yhtiöt tulivat kuvaan ihan vähän myöhemmin. Ja hyvä niin – monimuotoisuus ja vapaus ovat kantavia periaatteitamme. Muistathan?

Ja niin alkoi pikkuhiljaa kehittymään Debian-projekti⁵⁷, jossa alati kasvava joukko hakkereita ylläpiti vapaaehtois pohjalta yhä kasvavaa vapaiden ohjelmistojen kokoelmaa, vapaata käyttöjärjestelmää. Yksi projektin aikaisia johtohahmoja oli muuten Bruce Perens, joka on nykyään yksi tunnetuimmista Open Sourcen puolestapuhujista. Brucen jäljiltä Debianista on aina löytynyt muun muassa kattava kokoelma radioamatööreille soveltuvia ohjelmistoja.

Ensimmäinen "valmis" versio Debianista sai (poikkeuksellisesti) versionumeron 1.1 ja tämä historiallinen tapahtuma oli kesäkuussa 1996. Tuolloin siis monet kaupalliset yritykset, muun muassa Red Hat ja SUSE ja myös pahamaineinen Caldera olivat jo päässeet hyvään vauhtiin omien Linux-versioidensa julkaisemisessa.

Debian on olemassaolollaan tasapainottanut Linux-maailmaa, tarjoamalla vaihtoehdon kaupallisille Linux-brändeille. Siinä missä useimmat Linux-yritykset julkaisevat uusia versioita Linuxistaan jopa kahdesti vuodessa, on Debianilla saattanut mennä kaksikin vuotta uuden version julkaisemisessa. Debian on ehkä muita Linuxeja enemmän noudattanut periaatetta "tuote julkaistaan sitten kun se on valmis". Jopa sitten kun Debian vihdoin saa jotain julkaistua, ovat monet ohjelmat jo julkaisuhetkellä suhteellisen vanhoja, sillä Debianissa harrastetaan erittäin konservatiivista testaus- ja laadunvarmistuspolitiikkaa. Osittain hitaus johtuu myös siitä, että Debianin periaatteisiin kuuluu julkistaa kaikki 11 erilaista arkkitehtuuria samanaikaisesti. Useimmat muut Linux-jakeluversiot julkaisevat kunkin arkkitehtuurin version sitä mukaa kun saavat niitä valmiiksi.

Vaikka monet meistä tietokonefriikeistä tykkäämme aina kokeilla koneillamme kaikkea uutta, niin Debianin rauhallinen tahti itse asiassa sopii monille erittäin hyvin. Yrityksmaailmassa – jossa ihmiset haluavat tehdä välillä töitäkin – on ihan positiivista, ettei tarvitse yhtenä olla päivittämässä kaikkia yrityksen koneita uusimpaan versioon. Toisaalta Debianin omia tietokonefriikkejä hidas tahti ei haittaa, sillä he käyttävät usein niin sanottua "testing" versiota Debianista. Testing on koko ajan kehittyvä kokoelma ohjelmia, josta loppujen lopuksi kehitty seuraava Debianin versio. Vaikka se ei ole virallinen versio, on sekin luonnollisesti julkisesti saatavilla, kysehän on Open Source -projektista!⁵⁸

Ja nyt kun sekä Red Hatin että SUSE:n suunnitelmia on välillä ollut vähän vaikeata arvailla, kun Mandraken tulevaisuus on taloudellisten ongelmien takia ollut vaakalaudalla ja monet pienemmät Linux-brändit ovat ihan oikeasti kadonneet, on Debianin ei-kaupallisuus tuntunut suorastaan kilpailuedulta. Debian ei ole ollut talousvaikeuksissa, sen suunnitelmat eivät ole muuttuneet puolen vuoden välein, eikä kukaan ole yllättäen tarjoutunut ostamaan Debiania – eihän sellainen olisi mahdollistakaan.

Vakautta arvostavien käyttäjien lisäksi on luonnollista, että Debian kiinnostaa varsinkin ohjelmiojia ja tietokonefriikkejä. Tämä näkyy esimerkiksi Debiania asennettaessa. Siinä missä muut Linux-yritykset ovat kehittäneet omia asennusohjelmiaan mahdollisimman automatisoiduiksi ja kivan värikkäiksi, on Debianin asennusohjelmalla ollut tapana kysyä sellaisia mielenkiintoisia asioita kuin: "Mitä kernelin ajuritiedostoa verkkokorttisi käyttää?" tai "Mikä on monitorisi virkistystaajuus?" Todelliset nörtit eivät tiedä mitään hauskeempaa kuin vastailla tällaisiin kysymyksiin, mutta toisaalta on kiva että me tavalliset kuolevaiset voimme käyttää jotain mahdollisimman automaattista ja kivan värikästä Linuxia.

Vapaaehtoisten ylläpitämänä jättiprojektina Debian toteuttaa myös Open Sourcen periaatetta yhdessä tekemisestä. Ohjelmiojille ja muille tosinörteille – aina Linus Torvaldsista lähtien – on aina ollut hauskeempaa tehdä itse oma käyttöjärjestelmänsä, sen sijaan että ostaisi sen kaupasta värikkääseen pahvipakettiin pakattuna.

Tässä mielessä Debian muistuttaa vähän suomalaista talkooperinnettä. Suomalaiset talkoohan järjestetään silloin, kun suomalainen haluaa rakentaa itselleen vaikka kesämökin jonkin kauniin järven rannalle. Silloin hän ei suinkaan osta valmista jalasmökkiä, joka rekka-autolla tuodaan perille. Eihän siinä olisi mitään hauskaa. Mökki rakennetaan tietenkin talkoilla. Talkoot ovat sitä, että kutsutaan kaikki hyvät kaverit ja pari vähän tuntemattomampaakin tyyppiä ja sitten mökki rakennetaan yhdessä. Talkoiden jälkeen mennään saunaan ja tulevan kesämökin tuleva emäntä tarjoaa työläisille keittoa. Illan päätteeksi saatetaan juoda jopa saunaoluetkin.

Debian-projekti on kuin suuret virtuaalitalkoot. Usein Debianin vapaaehtoistyöntekijät kokoontuvat internetin IRC-keskustelukanaville ja ohjelmoivat sitten yhdessä Debianiin tulevia uusia ohjelmia. Siinä koodaamisen lomassa voi sitten chattailla vähän asiaan liittyen ja vähän asian vierestäkin. Ollaan yhdessä ja pidetään hauskaa. Joku voi juoda oluttakin ja virtuaalisesti skoolata onnistuneen työn kunniaksi. Ja hauskanpidon sivutuotteena syntyy maailman paras käyttöjärjestelmä.⁵⁹

Mutta palataanpa lopuksi vielä Debianiin liiketoimintamallina. Voidaanko talkoita kutsua liiketoimintamalliksi? Tämän luvun eräänlaisena alitajuisena oletuksena on ollut, että Linuxin varaan on voitava rakentaa kannattavaa liiketoimintaa, jotta Open Source olisi ajatuksena ensinkään uskottava. Debian (ja Eclipse, Blender, Mozilla ja monet muut tämän luvun esimerkit) kuitenkin todistavat muuta. Debian on olemassa ja sitä käytetään laajasti, mutta kyseessä ei ole liiketoimintaa harrastava yritys.

Me jotka olemme koko pienen ikämme eläneet tässä ylikaupallistuneessa kapitalistisessa pienessä maailmassamme, olemme unohtaneet, että kaikkea ei voi mitata rahassa. (Enkä nyt puhu ystävydestä ja rakkaudesta ja sellaisesta, vaan ihan materialistista asioista.) Me ajattelemme, että koska Microsoftin liikevaihto on niin suuri, siitä seuraa loogisesti, että Windows on hyvä käyttöjärjestelmä. Mutta mikä on Debianin liikevaihto? Ei sellaista olekaan.

Olemme unohtaneet, että tietokoneohjelman tarkoitus ei ole tuottaa liikevaihtoa. Tietokoneohjelman tarkoitushan on tuottaa muun muassa kirjoja, kirjoja, pöytäkirjoja, kuvia, elokuvia, www-sivuja, musiikkia, puhelinluetteloita, tiliotteita ja palkkakuitteja ja muita tarvitsemiamme asioita. Ja kaikkia noita voi tuottaa vaikka Debianilla, siis Debian on hyödyllinen ja mielekäs otus, joka ei kuitenkaan taida olla liiketoimintamalli.

Tätä mieliämme kahlitsevaa ajattelua on joskus yritetty havainnollistaa niin kutsutulla *pyykkinaruparadoksilla*. Jos nimittäin menen kauppaan ja ostan pyykinkuivauskoneen, niin jossain jonkun tilaston mukaan nähdään, että Suomessa on sähkönkulutus noussut. Ja näin olen osaltani vaikeuttanut Kiiton ilmastositomuksen Suomelle määrittämien rajojen saavuttamista.⁶⁰

Jos kuitenkin päätän heittää kuivauskoneeni tunkiolle ja kuivatankin pyykkini ulkona pyykkinarulla, niin missään tilastossa ei sanota, että Suomessa on aurinkoenergian käyttö lisääntynyt. Tämä ei kuitenkaan tarkoita, etteikö pyykkinaru olisi hyvä tapa kuivattaa pyykkiä. Ja tätä juuri on pyykkinaruparadoksi.

Vanha Debian-aktiivi Bruce Perens on ainakin ymmärtänyt pyykkinaruparadoksin opetuksen. Linux-käyttäjän kannaltahan ei ole oleellista, kuinka monta käyttöjärjestelmää hän on ostanut. Oleellista on, että työt saadaan tehtyä. Jotkut Brucen tavanneet yritysjohtajat ovat ilmeisesti tajunneet saman, sillä Bruce on eräiden toistaiseksi nimettömien toimeksiantajien toimesta äskettäin ruvennut puuhaamaan uutta Linux-jakeluvärsiota nimeltä UserLinux – käyttäjien tekemä Linux. UserLinuxin tavoitteena ei ole myydä mahdollisimman monta Linuxia, eikä UserLinuxin käyttäjien tavoitteena ole ostaa mahdollisimman monta Linuxia, he vain haluavat saada työnsä tehtyä. UserLinuxin tavoitteena on olla Debianin kaltainen kaikille avoin Linux-jakeluvärsio, mutta siitä on tulossa Debiania karsitumpi ja automatisoidumpi, yrityskäyttöön suunnattu Linux.

Tätä kirjoitettaessa UserLinux oli vielä "pilke Brucen silmäkulmassa" ja nähtäväksi jää miten se loppujen lopuksi konkretisoituu. Red Hatin tuoreimmat muutokset tuotevalikoimassaan ja hinnoittelussaan ovat varmasti osaltaan toimineet kipinäinä UserLinux-keskustelulle ja saaneet projektin taustalla olevat organisaatiot ymmärtämään Debianin kaltaisen mallin edut. *"Tiedän eräänkin yrityksen⁶¹ joka on investoinut miljoonia dollareita Linuxin kehittämiseksi IA-64 prosessoreilla toimivaksi. Erityisesti on mainittava, että Red Hat ei osallistunut tähän työhön ollenkaan. Nyt tämä yritys joutuu ostamaan oman työnsä takaisin Red Hatilta, kalliilla hinnalla ja vieläpä maksamaan kappaleittain,⁶² käytettäväksi heidän koneissaan."* Jonkin verran uskottavuutta projektille tuo se, että Bruce mukaan taustaorganisaatiot ovat luvanneet rahoittaa UserLinuxia miljoonalla dollarilla vuodessa ja kyseiset organisaatiot käyttävät yhteensä noin 50 000 tietokonetta. Nyt ei siis enää puhuta hakkereiden yhteistalkoista ja ajanvietosta, vaan isoista yrityksistä jotka haluavat saada työnsä tehtyä.

Tuomio: Debian on kaksinkertaisesti Open Source. Sen sisältämät ohjelmat kehitetään Open Source -kehitysmallilla, mutta sen lisäksi Debian itse syntyy Open Sourcena. Tuhannen vapaaehtoisen voimin kehitettävä Debian on monilla mittareilla mitattuna suurin Linux-jakeluversio, ja 2000-luvun alkua vaivanneen IT-laman aikana se on jopa ollut taloudellisesti vakain vaihtoehto Linuxin käyttäjille. Debianista ei tämän kirjan tarkasteluperspektiivistä katsottuna yksinkertaisesti löydy pahaa sanottavaa.

52Niin no, joidenkin mielestä sinulla on pahasti viallinen tietokone, jos asennat siihen Linuxin sijaan Windowsin. Mutta kuvitellaan nyt kuitenkin että sinulla on toimiva tietokone.

53Tämä tilasto on joulukuulta 2003. Luku on ollut viime vuosina jyrkässä nousussa, joten tätä lukiessa se on jo varmasti paljon korkeampi.

54Kirjan mennessä painoon oli tämäkin luku jo kasvanut.

55Tietenkin voisi sanoa, että nykyään myös IBM ja HP ovat Linux-yhtiöitä ja heillä on työntekijöitä sitten vähän enemmänkin.

56Ja tarkoitan siis tässä ihan oikeita ihmisiä, jotka kääntävät englanninkieliset oppaat vaikkapa suomeksi. Ohjelmoijat taas käyttävät niitä kääntäjä-ohjelmia, jotka kääntävät heidän kirjoittamansa koodin tietokoneen ymmärtämäksi binäärikoodiksi.

57Jos kerron että lanin vaimon nimi on Debra, niin arvaatko jo mistä projekti sai nimensä?

58Debianin "testing" ja "unstable" versiot toimivat siis samoin kuin Mandraken Cooker-projekti ja Debian on itse asiassa tältä osin ollutkin Mandraken esikuva.

59Debianin tekijöihin voi tutustua tältä sivulta löytyvillä irc-kanavilla: <http://www.linuxs.mine.nu/debian-faq/>

60Enlanningkielisen käännöksen amerikkalaisille lukijoille huomautettakoon, että teillä ei ole tässä suhteessa mitään syytä huoleen. Teidän Texasista kotoisin oleva presidenttinnehan ei hyväksynyt Kioton ilmastopimusta, joten te voitte käyttää sähköä vapaasti ilmastosta välittämättä.

61Vaikka Burce ei sitä paljastakaan, niin kyseessä on HP.

62 Muista, että käyttöjärjestelmän lisensoiminen kappaleittain on Open Source -maailmassa epätavallista.

46. LASITALO, TÄYSIN LÄPINÄKYVÄ YRITYS (FIKTIIVINEN)

Mandraken ja Linux Weekly Newsin tapauksessa, kun asiakkaat ovat ruvenneet tukemaan taloudellisesti yrityksiä "ei mistään", eli ei suorana kompensaaiona mistään tietystä lukumäärästä tuotteita, on noussut olennaiseksi raportoida näille asiakkaille myös yhtiön taloudellisesta tilanteesta. Viesti asiakkailta on ollut: "Me tuemme teitä mielellämme, mutta teidän pitää kertoa paljonko rahaa tarvitsette." Jotkut ovat jopa menneet askeleen pidemmälle ja ehdottaneet, että yritykset voisivat vaikka julkistaa koko kirjanpituksensa, jolloin kaikki voisivat milloin tahansa nähdä tarkat tiedot siitä, minkälainen taloudellinen tilanne yrityksellä kulloinkin on.

Toistaiseksi yksikään tuntemani yritys ei tällaista ole tehnyt, eikä se pörssiyritykselle käytännössä kovin helppoa ehkä olisikaan, pörssiyrityksille on tarkat säännöt siitä, mitä milloinkin saa julkisuuteen tiedottaa. Mutta ajatus on kuitenkin mielenkiintoinen. Se ei ehkä ole kaikilta osin täysin realistinen, eikä se myöskään suoraan liity Open Sourceen, mutta käyttäkäämme kuitenkin tämän luvun viimeiset sivut tällaiseen ajatusleikkiin. Jos yritys yrittäisi olla mahdollisimman avoin, kuinka avoin se silloin voisi olla ja mitä kaikkea siitä seuraisi?

Vaikka kysymyksenasettelumme on vähän äärimmäisyyksiä hakeva, se ei ole täysin vailla todellisuuspohjaa. Esimerkiksi Nokiaa on tässä kirjassa aiemmin käytetty esimerkkinä yrityksestä, joka on valitettavan ihastunut salassapitosopimuksiin. Laajaan alihankkijaverkostoonsa päin Nokia ei kuitenkaan ole pelkästään salaileva, päinvastoin. Alihankkijoihinsa päin Nokia pyrkii saamaan molemminpuolisen tiedonkulun toimimaan niin sujuvasti ja avoimesti kuin vain mahdollista.⁶³ Jos esimerkiksi alihankkija pystyisi oma-aloitteisesti seuraamaan kännyköiden myyntilukuja suoraan Nokian tietokannoista, hän pystyisi varautumaan kasvaneeseen kysyntään jo etukäteen, ennen kuin seuraava varsinainen tilaus Nokialta päin tulee. Näin vältytään useilta pullonkauloilta ja pitkä ja monisyinen tuotantoketju hyrrää kuin rasvattu.

Avoimuusajattelumme on siis toisaalta hyvinkin perusteltavissa. Kuinka avoin avoin yritys sitten voisi olla ja mitä siitä seuraisi? Nyt kun IT-firmat rakentavat pääkonttorinsakin pleksilasista, on paikallaan tehdä pieni ajatusleikki: Millainen olisi täysin läpinäkyvä yritys?

Täysin läpinäkyvä yrityksemme ei välttämättä toimisi IT-alalla, mutta sen tuote olisi tietenkin soveltuvien osien Open Source -henkinen.⁶⁴ Tämä on tietenkin itsestään selvää. Mutta mitä muuta?

No, ensinnäkään sillä ei tietenkään olisi ainoatakaan yrityssalaisuutta. Salassapitosopimukset olisivat kiellettyjä. Mandraken Cookerin tavoin tuotekehittäminen olisi avointa, esimerkiksi yhtiön sähköpostitilat olisivat avoimia myös yrityksen ulkopuolisille.

Budjetit ja kirjanpito olisivat julkisia, siinä määrin kuin se vain on mahdollista jaärkevää. Esimerkiksi yksittäisten työntekijöiden palkkoja ja sairaslomia ei tietenkään voida julkistaa, mutta muuten kaikki olisi julkista.

Tuotekehityksen lisäksi myös yhtiön hallituksen kokoukset ja muut johtajatasen neuvottelut olisivat julkista omaisuutta. Ei siis vain tehtyjen päätösten osalta, vaan myös hylätyt ideat ja keskustelunavaukset kirjattaisiin – tai voisihan ne nykyaikana vaikka videoita.

Noudattamalla tällaisia sataprosenttisen avoimuuden vaatimuksia, syntyisi täysin läpinäkyvä yhtiö. Mikä vastakohta niille Enroneille, joiden pillin mukaan talousmaailma nykyään tanssii. Pleksilasitetut toimistotalot ovat näennäisesti läpinäkyviä, mutta oikeastihan niiden lasipinta ei ole läpinäkyvää vaan seinä on heijastava. Meidän kuvitteellinen yrityksemme taas olisi todellinen, oikeasti läpinäkyvä *lasitalo*.

Miten tällaiselle läpinäkyvyyttä harrastavalle yritykselle sitten kävisi? Jos Open Source -maailmasta tekemiimme havaintoihimme on luottamista, niin avoimet prosessit yhtiön tuotekehityksessä ja johdossa avaisivat mahdollisuuden yhtiön ulkopuolelta tulevalle yllättävälle avulle. Näin tuote kehittyisi nopeammin ja se olisi ajan mittaan kilpailijoitaan laadukkaampi. Yritys voisi saada täysin uudenlaisia yhteistyökumppanuuksia, tai vaikkapa asiakkailtaan ehdotuksia myyntiorganisaationsa uudistamisesta, samaan tapaan kuin Open Source -ohjelmointiprojekteissa ulkopuoliset lähettävät parannusehdotuksia projektin koodiin.

Toisaalta Open Source -tuote on tietenkin myös kilpailijoiden käytössä ja tämä on yrityksellemme haaste. Avoimuuden periaatetta noudattava yritys ei voi suojautua tekijänoikeuksien, patenttien, yrityssalaisuuksien ja muiden perinteisten pihtaamismallien taakse, vaan sen on kohdattava myös kilpailu avoimesti. Tämä pakottaisi yrityksen olemaan jatkuvasti varpaillaan ja jatkuvasti uudistamaan ja kehittämään itseään. Nämä ovat tietenkin sanoja, joita jokainen yritys toistaa jonkinlaisena maagisena mantrana, mutta avoimessa tilanteessa ne ovat todellisia vaatimuksia – elinehtoja.

Sataprosenttisesta läpinäkyvyydestä seuraisi myös joitain yllättäviä ja mahdollisesti hankaliakin tilanteita, joissa yritys joutuisi miettimään liiketoiminnan perusasioita täysin uudesta näkökulmasta. Vähän samalla lailla kuin ensimmäiset Linux-yhtiöt joutuivat miettimään toimintaansa IT-alalla, jossa Open Source oli uusi käsite.

Esimerkiksi tällaisesta uutta ajattelua vaativasta tilanteesta sopisi vaikka tarjouskilpailuun osallistuminen. Tarjouskilpailuthan toimivat niin, että tilaaja vastaanottaa tiettyyn päivämäärään mennessä tarjouksia useilta myyjäehdokkailta. Tilajalle luovutetut tarjoukset pidetään ensin salassa ja vasta tarjousajan päätyttyä valitaan edullisin tarjous ja työ tilataan sen tekijältä. Salassapidon takia kukaan tarjouksen jättäjä ei voi varmasti tietää kilpailijoidensa tarjouksien sisältöä. Tarjouskilpailun pointtina on muodostaa tarjoajien kesken kilpailutilanne, jotta tilaaja saisi haluamansa tuotteen tai palvelun mahdollisimman edullisesti.

Meidän läpinäkyvyyden periaatetta noudattava yritys olisi kuitenkin tarjouskilpailussa tietenkin muita huonommassa asemassa. Samalla kun muiden tarjoukset ovat salaisia, voisivat muut käydä meidän yrityksen www-sivuilla tarkastamassa johtoryhmän pöytäkirjoista, mihin hintaan aiomme tarjouksen tehdä ja ehkä jopa löytää yksityiskohtaisen laskelman siitä, miten kyseiseen hintaan on päästy. Edullisemman tarjouksen tekeminen olisi silloin tietenkin naurettavan helppoa.

Voi hyvinkin olla, että avoimuutta edustava yrityksemme tämän johdosta häviäisi monia tarjouskilpailuja. Mutta asia ei kuitenkaan ole ihan näin yksinkertainen. Tarjouskilpailujen ongelmanahan on juuri kysynnän ja tarjonnan laista seuraava eräänlainen uhkapeli. Kukaan myyjähän ei todellisuudessa tee tarjoutua sen perusteella, mitä kustannuksia tarjottuun työhön liittyy. Sen sijaan myyjät pyrkivät arvailemaan toistensa tarjouksia ja tekevät sitten omansa sen perusteella, mitä uskoavat muiden tarjoavan. Tavoitteena on tehdä mahdollisimman korkea tarjous, joka on kuitenkin niin edullinen että tilaaja valitsee juuri sen. Siksi monet tarjoukset eivät välttämättä ole niin edullisia kuin ne voisivat olla.

Avoin ja läpinäkyvä yrityksemme voikin suojautua tätä ongelmaa vastaan tekemällä aina pelkästään rehellisiä tarjouksia! Rehellisyydellä tässä tarkoitetaan sitä, että yrityksemme päättää tarjouksensa yksinomaan työhön liittyvien kustannusten perusteella ja tekee rehellisesti niin edullisen tarjouksen kuin pystyy, ilman että tarjous olisi kuitenkaan tappiollinen. Nyt, jos löytyy joku, joka tekee vielä edullisemman tarjouksen, ei ole mitään syytä surra menetettyjä kauppia, sillä eihän tämän halvemmalla myyminen olisi ollut meille kannattavaa kuitenkaan. Toisaalta voi olla, että tarjouksemme on niin edullinen, että kukaan ei pysty sen parempaan. Tällöin olemme täysin ansaitusti voittaneet tarjouskilpailun, salaamatta yhtään mitään!

Jos tilannetta ajatellaan tilaajan, eli asiakkaan kannalta, niin tällainen avoimin kortein pelaava yritys olisi tietenkin erittäin mieluinen yhteistyökumppani. Tilaaajan kannalta on tietenkin ikävää, että hänelle tehtävät tarjoukset perustuvat lähinnä arvailuihin siitä, kuinka paksun potin kukin tarjoaja uskoo pystyvänsä tällä kertaa pahaa avistamattomalta asiakkaalta huijaamaan. Tarjoushinnan avoimesti laskeneen yrityksen tarjoukseen taas voi helpommin luottaa. Jos esimerkiksi saisin tällaiselta yritykseltä tarjouksen, jonka loppusumma olisi 1 miljoona euroa ja toiselta, suljetuin kortein pelaavalta yritykseltä tarjouksen jonka loppusumma olisi 999 999 euroa, niin lähinnä nauraisin jälkimmäiselle, mutta valitsisin ensimmäisen.

Tuomio: Tämän kappaleen yritys oli täysin fiktiivinen ja kyseessä on vain ajatusleikinomainen esimerkki siitä, mitä kaikkea avoimuudesta voikaan seurata. Mikäpä minä olen omia ideoitani hyväksi tai huonoiksi tuomitsemaan. Tämän kirjan tavoitteena on ollut herättää ajatuksia avoimuudesta ja tämä kappale on esimerkki sellaisesta ajattelusta ja toimikoon rohkaisuna ja haasteena yhä avoimemmalle ajattelulle lukijan omassa elämässä.

63No, jos totta puhutaan, niin salailun ja pihtaamisen kulttuuri on meissä kaikissa niin vahva, että joskus tällaisen avoimen tiedotuskulttuurin luominen ei olekaan niin helppoa kuin voisi kuvitella, mutta ainakin teoriassa Nokia on alihankkijoihin kohtaan mahdollisimman avoin.

64Seuraavassa luvussa käsitellään Open Source -ideologian soveltamista muualla kuin IT-alalla.

47. YHTEENVETO

Tässä luvussa on käsitelty useita erilaisia tapauksertomuksia Open Source -maailman yrityksistä ja organisaatioista. Perinteisten IT -yritysten haastaminen avoimuuden pelisäännöillä on pakottanut alan pelureita ajattelemaan liiketoimintaansa uusilla tavoilla ja tästä on syntynyt useita menestyksekkäitä liiketoimintainnovaatioita. Esimerkiksi Linux Weekly Newsin tapauksessa kyse ei edes ole niinkään Open Source -ideologiasta, vaan pelkästään internetiin luonnostaan kuuluvasta avoimuudesta ja pärjäämisestä sen kanssa. Debian ja pyykkinaruparadoksi taas auttaa meitä ymmärtämään, että joskus oikeaa vastausta voi olla vaikea löytää, koska ehkä emme alun perin ole asettaneet kysymystäkään oikeista lähtökohdista.

Olen pyrkinyt tuomitsemaan jokaista esimerkkiä sen mukaan, miten hyvin yritys on onnistunut taloudellisesti, ja toisaalta sen mukaan, miten hyvin esimerkki on sopusoinnussa Open Source -aatteen periaatteiden kanssa. Ajattelin, että näin lopuksi olisi kiva tiivistää tämän pitkän luvun anti kuvana, joka kertoo enemmän kuin tuhat sanaa.⁶⁵ Luvun esimerkit on sijoitettu kuvaan niin, että taloudellisesti menestyksekkäät esimerkit ovat ylhäällä ja huonosti menestyneet alhaalla. Open Sourcen periaatteita noudattavat taas ovat oikealla ja pihtaamisen periaatetta noudattavat vasemmalla.

| | | | |
|--|---|---|---|
| <p>Pihtaava, mutta älä ole liian ahne (Sun & Java)</p> | <p>Kerää Open Source ohjelmistoja CD-levylle, mutta rajoita levyn levitysoikeuksia (SuSE)</p> <p>Pihtaava, mutta älä ole liian ahne (Ghostscript)</p> <p>Myy suljettuja ohjelmistoja Open Source -ympäristöön (Ximian, Code weavers, Transgaming)</p> | <p>Kaksoisilensointi (MySQL, Trolltech)</p> <p>Ole alallasi maailman paras ja asiantuntevin (SuSE)</p> <p>Myy Open Source -ohjelmistoja CD-levylle koottuna (Red Hat, monet muut)</p> <p>Julkaise vanhan pelin, jota ei enää myydä, lähdekoodit (Quake)</p> | <p>Talkoot ja pyykkinaruparadoksi (Debian)</p> <p>Maksetaan työstä (Saksa & Kroupware)</p> <p>Mesenaattimalli (Larry Wall, Linus, KDE)</p> <p>Maksetaan työstä (keskisuuret yritykset & Kolab, JBoss)</p> <p>Tarvitset ohjelmaa, mutta sen kehittäminen suljettuna on taloudellisesti kannattamatonta (Mozilla, Eclipse, Blender...)</p> <p>Anna tuote ilmaiseksi ja pyydä asiakkaita maksamaan siitä, että voit jatkossakin tehdä niin (Mandrake, LWN)</p> |
| | <p>Yritä kovasti itse, sen sijaan että tekisit yhteistyötä (Corel)</p> | | <p>Maksetaan työstä (Stephen King, juom arahamalli, työnvälityspörssi)</p> |

65Tässä luvussa tosin on jo nyt reilusti yli 20 tuhatta sanaa – tietokoneeni osasi laskea sen ihan itse, eikä olekin ihmeellistä – mutta toivottavasti kuvasta on kuitenkin jotain iloa.

LUKU 4

48. LUKU 4

49. AVOIN ELÄMÄ

50. KIRJALLISUUS

51. HARRY POTTER JA OPEN SOURCE
TAIAT

52. WIKIPEDIA, ÄÄRIMMÄISEN AVOIN
TIETOSANAKIRJA

53. DMOZ.ORG, AVOIN WWW-SIVUJEN
HAKEMISTO JA MOZDEX, AVOIN
HAKUKONE

54. KOULUKIRJOJA, KEITTOKIRJOJA,
MUUTA FAKTAA JA VÄHÄN FIKTIOTAKIN

55. PROJECT GUTENBERG

56. MUSIIKKI

57. CREATIVE COMMONS JA AVOIMEN
INFRASTRUKTUURIN LUONTI

58. OPEN SOURCE -ELOKUVAILTA

59. OPEN CORES – VAPAATA RAUTAA

60. OPEN COLLECTION – MUOTIA,
BRÄNDEJÄ JA KANGASTA

48. LUKU 4

jossa Harry Potterista taivotaan saksalainen, muurahaiset tunaroivat,
Google saa kilpailijan ja henkinen ja fyysinen maailma kohtaavat.

49. AVOIN ELÄMÄ

Rob McEwen ei ollut kullankaivajan poika, mutta hänen isänsä tapasi kuitenkin kertoa tarinoita 1800-luvun kultakuumeesta. Ehkäpä näillä tarinoilla oli vaikutusta, kun Rob vuonna 1989 päätti jättää työnsä sijoitusneuvojana ja siirtyä Ontarion Red Laken alueella toimivan kullankaivoksen enemmistöosakkaaksi.

Robin kullankaivosta ei kuitenkaan valitettavasti vaivannut ainakaan kultakuume. Louhintakustannukset olivat korkeat ja kullankaivoksen maailmanmarkkinahintakin oli alhaalla. Päälle päätteeksi kaivoksen työläisetkin menivät lakkoon. Rob kuitenkin uskoi kaivokseensa. Red Laken alueelta oli louhittu yhteensä 18 miljoonaa unssia kultaa ja yksi aivan vieressä toiminut kaivos oli tuottanut 10 miljoonaa unssia. Robin kaivoksesta oli toistaiseksi louhittu vasta 3 miljoonaa unssia, mutta Rob oli varma, että sama suoni jatkui hänenkin kaivokseen. Se piti vain löytää!

Vuonna 1999 Rob osallistui MIT yliopiston järjestämään IT -seminaariin. Kuinkas ollakaan seminaarissa puhuttiin myös Linuxista ja uudesta vallankumouksellisesta tavasta tehdä tietokoneohjelmia. Rob innostui. Tätä Open Sourcea hänen kullankaivoksensakin tarvitsi!

Niinpä Rob julkisti maaliskuussa 2000 Goldcorp Challenge -kilpailun. Kaikki hänen kaivoksensa toiminnasta vuosien saatossa kerääntynyt data julkaistiin kilpailun nettisivulla ja osallistujien tehtävänä oli kertoa, mistä seuraavaksi kannattaisi porata. Kilpailu oli kaikille avoin ja palkintorahoja oli jaossa yhteensä 575 000 dollaria.

Robin idea oli kaivosalalla ennenkuulumaton. Jokainen kullankaivaja tietää, että omista löydöistä ei muille puhuta – kaivosten salaisuuksien kanssa mennään vaikka hautaan. Myös Robin omat geologit vastustivat aluksi ideaa. Kaivoksen tietojen paljastaminen voisi esimerkiksi houkutella paikalle ostajia, jotka haluaisivat kaapata kaivoksen Robilta ja muilta omistajilta. Omien korttien paljastaminen ei tällä alalla monestakaan syystä ollut hyvä idea. Tai näin siis ainakin oli tapana ajatella.

Kilpailu kuitenkin osoittautui menestykseksi. Tutkijat ympäri maailman luonnollisesti kiinnostuivat ennenkuulumattomasta mahdollisuudesta päästä penkomaan ihan oikean kaivoksen dataa. Kiinnostuneita tutkijoita 50 eri maasta kävi lataamassa kilpailun datatiedostoja yli 1400 kertaa. Voittajaksi kilpailun raati valitsi kahden australialaisyhtiön – Fractal Graphicsin ja Taylor Wall & Associatesin – rakentaman virtuaalisen kolmiulotteisen mallin kaivoksesta. Mallin avulla australialaiset, jotka eivät olleet koskaan käyneet edes Kanadassa, pystyivät ennustamaan, mistä kultaa seuraavaksi löytyisi. Voittosumma oli 105 000 dollaria.

Ja miten Robin kaivokselle sitten kävi? Iskö hän Open Source -kultasuoneen? Toistaiseksi Robin yhtiö on porannut neljää viidestä voittaneesta ehdotuksesta, ja jokaisesta on löytynyt kultaa. Vuonna 1996 Red Laken kaivos tuotti 53 000 unssia vuodessa ja yhden unssin tuotantokustannukset olivat 360 dollaria. Vuonna 2001 – ensimmäisenä kilpailun jälkeisenä vuonna – kultaa löytyi 500 000 unssin edestä ja koska kultaesiintymä oli nyt aiempaa rikkaampi, yhden unssin kustannuksiksi tuli enää vaivaiset 59 dollaria. Kullankaivoksen maailmanmarkkinahinta oli 307 dollaria, joten voittoa tuli runsaasti. Vanhoilla menetelmillä jatkaminen taas ei olisi tähän hintaan ollut edes mahdollista. Ilman Robin Open Source -idea ja Goldcorp Challenge -kilpailua koko kaivos olisi hyvinkin voitu joutua sulkemaan.⁶⁶

Tässä viimeisessä luvussa tarkoituksenamme on laajentaa avoimuuden periaatetta ulos tietokoneiden maailmasta meidän tavallisten ihmisten elämään. Kuten Rob osoitti, avoimuutta voidaan soveltaa jopa kullankaivuuseen. Mihin muuhun sitä voisi soveltaa? Monet tämän luvun tarinoista ovat Robin kultakilpailun tavoin todellisia esimerkkejä elävästä elämästä. Toiset taas ovat eräänlaisia ehdotuksia, vielä toteutumattomia avoimuuden vallankumouksen osasia. Tärkeintä on kuitenkin muistaa, että nämä kaikki ovat vain esimerkkejä. Robin ja muiden tarinoiden tarkoituksena on osoittaa, että avointa ajattelutapaa voi hyödyntää kaikkialla.

Robin tarina ei liity Linuxiin eikä tietokoneisiin. Se kertoo meille, että Open Sourcen ei tarvitse olla vain tietokoneohjelmien erikoisuus. Haastamalla vanhat salailun ja pihtaamisen perinteet avoimella ajattelulla, voi itse kukin olla omalla alallaan yhtä vallankumouksellinen kuin Richard Stallman ja Linus Torvalds ohjelmoijina tai Rob McEwen kullankaivajana. Avoimella ajattelulla voit iskeä kultasuoneen – kirjaimellisesti! Meillä on jo avoin koodi, mutta miksi tyytyisimme siihen, kun niin paljon muutakin on tarjolla? Kysymys on vain tavoista ajatella, tavoista tehdä erilaisia, jokapäiväisiä asioita ja elää jokapäiväistä elämäämme. Ja se elämä voi olla Avoin Elämä.

66Robin tarinan kertoi Fast Company -lehden verkkoversio kesäkuussa 2002: <http://www.fastcompany.com/online/59/mcewen.html>

50. KIRJALLISUUS

Tietokoneohjelmien "omistaminen" perustuu tekijänoikeuslakiin ja Richard Stallmanin kehittämä GPL-lisenssi perustuu tämän lain uudenlaiseen soveltamiseen. On siis johdonmukaista ajatella, että Open Source -ajattelua voisi soveltaa muihinkin tekijänoikeuslain alaisiin teoksiin. Ja näin tietenkin onkin ja näin on jo tehtykin. Ja aivan aluksi oli kirjallisuus...

Kirjapainon keksiminen nimittäin oli se kimmoke, joka johti vähitellen meidänkin nykyään tuntemiin tekijänoikeuslakeihin. On siis jotenkin luonnollista aloittaa siitä. Kirjojen julkaisemisen ekonomia myös muistuttaa paljon ohjelmistoteollisuutta. Varsinainen työ on kirjan kirjoittamisessa, jonka jälkeen kirjaa voidaan painaa haluttu määrä kappaleita – ei ehkä ihan ilmaiseksi, mutta halvalla kuitenkin. Nykyään internet-aikana on tietenkin mahdollista, että kirjaakin levitetään digitaalisessa muodossa, jolloin kopiointikustannukset ovat oikeasti täysin olemattomat.

Miksei sitten kirjoja ole tapana julkaista Open Source -periaatteita noudattaen? Ehkäpä niinkin yksinkertaisesta syystä, että kukaan ei ole tullut ajatelleeksi, että niinkin voisi tehdä.

Toisaalta, voi myös olla, että Open Sourcesta ei ole kirjan kirjoittamisessa samanlaista etua, kuin siitä on tietokoneohjelmoinnissa. Tietokoneohjelman tekeminenhän on yleensä monimutkainen prosessi, johon vaaditaan joskus jopa usean sadan hengen monen vuoden työpanos. Tässä monimutkaisessa prosessissa avoimuudesta saatava etu on suljettuihin ohjelmiin selvä (ja tästä siis ovat todisteena Linux, Apache ja niin edelleen).

Mutta kirjan kirjoittaminen ei ole yhtä monimutkaista. Itse asiassa tyyppilisen kirjan pystyy ihan hyvin kirjoittamaan yksinään, eikä siihen yleensä mene edes yhtä vuotta, paitsi jos kirjaa kirjoittaa viikonloppuisin sivutoinään, niinkuin minä nyt tätä kirjaa. Toisekseen, jotta kirjaa voisi kirjoittaa yhteistyössä muiden kanssa, pitäisi kirjoitustyötä pystyä jotenkin järkevästi jakamaan osiin. Monen kirjan kanssa tämä voisi olla hyvinkin vaikeaa. Esimerkiksi kirjailijoiden pitäisi keskustella todella tarkasti yhteisestä juonesta, henkilöhahmoista ja niin edelleen, jotta kirjaprojekti säilyisi yhtenä kokonaisuutena. Loppujen lopuksi olisi varmasti helpompaa, että joku kirjoittaa kirjan ihan itse.

Tästäkin huolimatta, on kyllä olemassa Open Source -kirjoja. Nämä kirjat eivät kuitenkaan ole juonellisia dekkareita, vaan ne ovat esimerkiksi Linuxin käyttöoppaita.

Jo hyvin aikaisessa vaiheessa, kun ensimmäiset GNU-ohjelmat alkoivat saada ilmaa siipiensä alle, kävi selväksi, että vapaalla ohjelmalla pitää luonnollisesti olla myös vapaa käyttöopas. Ja kuinka ollakaan, käyttöoppaan kirjoittaminen sujuu oikein hyvin Open Source -prosessia hyödyntäen. Käyttöoppaan voi helposti jakaa useisiin erillisiin osiin, joiden parissa eri kirjoittajat voivat työskennellä. Lisäksi käyttöopasta on yleensä tarve päivittää sitä mukaa kun ohjelmakin kehittyy. Tässä suhteessa Open Source käyttöopas onkin paljon parempi kuin opas jota ei kuka tahansa saisi muuttaa, sillä yleensähan suurin osa vanhasta oppaasta voidaan hyödyntää myös uuden ohjelman käyttöoppaassa. Vain muuttuneet osat korjataan ja ohjelman uusista ominaisuuksista kirjoitetaan uusia lukuja, mutta koko opasta ei tarvitse kuitenkaan kirjoittaa uudestaan. On huomattava, että perinteisten kirjojen tapauksessa ei tällainen työskentelytapa olisi tekijänoikeuslain mukaan sallittua. Perinteisen ajattelutavan mukaan ei vanhaa käyttöopasta saisi hyödyntää, ellei sen alkuperäinen kirjoittaja anna siihen lupaa. Tämän takia Open Source -yhteisön jäsenet julkaisevatkin aina sekä ohjelman koodin että käyttöoppaan mukana tekstin, jossa tällainen lupa annetaan heti alkuunsa ja kaikille halukkaille.

Ja koska monella kustantajalla alkaa tässä vaiheessa varmasti housun puntit tuitista, on syytä kerrata kaikki se, mitä olemme aiemmissa luvuissa jo käyneet läpi tietokoneohjelmien osalta. Nythän nimittäin moni varmasti ajattelee, että kirjan julkaiseminen internetissä Open Sourcen ehdoilla on huono bisnes, joka varmasti koituu kustantamoiden tuhoksi. Tämä on suurin piirtein sama ajatuskuvio, jonka perinteiset ohjelmistotalot käyvä läpi kun ensimmäistä kertaa kuulevat Linuxista. Oikeasti kuitenkin asia on täysin päinvastoin. Netistä löytyvä Open Source -kirjahan on kustantajalle kuin ilmaista rahaa. Valmiiksi kirjoitettu kirja, joka vain odottaa painamista ja myyntituloja. Ja niin vain on, että monet fiksut kustantajat ovat tajunneet tämän asian juuri näin. Eriten Open Source -kirjoja on ilmeisesti julkaissut O'Reilly, eli sama kustantaja joka aikoinaan otti Perl-ohjelmointikielen kehittäjän Larry Wallin palkkalistoilleen, ja otti paikan Open Source -liikkeen hovihankkijana kirjojen osalta. Mutta muutkin kustantamot ovat löytäneet Open Source -markkinat. Prentice Hall on lähtenyt hakemalla hakemaan Open Source -kirjailijoita sarjaan, jonka kummisetänä toimii Open Source -yhteisön puhemies Bruce Perens ja jota myös markkinoidaan hänen nimellään.

Open Source -kirjoja siis löytyy. Mutta olemme nyt pahasti jumahtaneet taas Linuxin alueelle. Löytyykö siis Open Source -kirjallisuutta muutenkin kuin Linuxin käyttöoppaista?

51. HARRY POTTER JA OPEN SOURCE TAIAT

Viime syksynä huomioni kiinnittyi uutiseen saksalaisista Harry Potter -faneista. Lasten ja nuorten rakastaman noitapojan seikkailuistahan ilmestyi kesäkuussa 2003 sarjan viides kirja "Harry Potter ja Feeniksin kilta".

Niin, siis silloinhan ilmestyi kirjan englanninkielinen versio. Saksankieliset käännökset ilmestyivät sitten joskus myöhemmin. Mutta todellinen fani ryntää tietenkin heti kauppaan ja tavaa läpi uuden seikkailun alkuperäiskielellään, sillä välin kuin odottelee saksankielisen käännöksen julkistamista, jonka hän sitten aikoinaan myöskin hankkii.

Ei aivan. Tuo riittää ehkä aivan tavalliselle keskiverto fanille, mutta todellinen todellinen fani menee vielä pidemmälle. Hän tekee käännöksen itse. Tai ei tietenkään aivan itse, mutta yhdessä muiden kanssa.

Jokin aika englanninkielisen Potterin julkaisemisen jälkeen organisoitui www.harry-auf-deutsch.de nettisivuston alaisuuteen yli tuhat saksankielistä Harry Potter -fania. Näiden vapaaehtoisten kääntäjien keski-ikä oli niinkin alhainen kuin 16 vuotta. Jotkut kääntäjistä olivat koulussaan lukeneet vasta pari kolme vuotta englantia.

Käännöstyö jaettiin viiden sivun mittaisiin paloihin. Toiset vapaaehtoiset osallistuivat käännöksen oikolukemiseen ja kommentoivat eri käännösvaihtoehtoja. Harry Potterin maagiseen maailmaan liittyvän erikoissanaston kääntämiseksi perustettiin sivustolle erityinen Harry Potter Englanti-Saksa-Englanti sanakirja ja sanakirjan käännöksistä käytiin myös vilkasta keskustelua. Miten esimerkiksi kääntää saksaksi sana "Squib", joka tarkoittaa henkilöä, jonka vanhemmat ovat noitia, mutta joka ei itse omaa maagisia kykyjä?

Kuten arvata saattaa, joutui tällainen projekti äkkiä napit vastakkain virallisen saksankielisen Harry Potterin kustantajan lakimiesten kanssa. Nythän onkin niin, että tällaisten epävirallisten käännösten levittäminen internetissä ei ilman alkuperäisen kirjailijan lupaa ole sallittua ollenkaan. Harry-auf-Deutsch -sivusto sai onneksi riidan sovittua kustantajan kanssa siten, että sivustosta tehtiin suljettu kerho ja käännöstyön lopullinen tulos ei ole julkisesti saatavana vaan sitä jaetaan vain työhön osallistuneiden kesken.

Yhtä hyvin ei käynyt joidenkin muiden maiden Potter-faneille. Muun muassa tsekkiläiset potteristit olivat myös tympääntyneet odottelemaan yhdeksän kuukautta kestäväää virallista käännösprosessia ja saksalaisten fanien tavoin he pulauttivat omin päin epävirallisen käännöksen huomattavasti nopeammin. Tämän käännöksen verkkosivu kuitenkin suljettiin tsekkiläisen Potter-kustantamon toimesta, mikä myöhemmin herätti kustantamon nuorissa asiakkaissa ansaittua suuttumusta.

Vaikka nämä Potter-käännökset vähän venyttävätkin tekijänoikeuslain rajoja, niin ne ovat kuitenkin osoitus Open Source -ajattelun voimasta. Siinä missä useimmat ammattikäntäjät käyttävät yhden romaanin kääntämiseen vajaan vuoden, esimerkiksi Tsekissä ja Suomessa yli yhdeksän kuukautta, saa satojen tai jopa tuhansien vapaaehtoisten kääntäjien joukko työn tehtyä parissa kuukaudessa, tai parissa viikossa, jos laadusta tingitään. Eikä ole mitään syytä epäillä, että lopputulos välttämättä olisi mitenkään puutteellisempi viralliseen versioon verrattuna. Päinvastoin. Mikäli Linux-ilmion taika on siirrettävissä romaanien kääntämiseen, on hyvinkin mahdollista, että Open Source -käännöstä on paljon hauskeampi lukea, kuin kustantajan suljettua ja virallista versiota.

Harry Potter on ehkä tullut luoneeksi uuden internet-ajalle sopivan tavan kääntää kirjoja, mutta hän on myös valloittanut ykkössijan eräissä perinteikkäämmässä kirjallisuuden alakulttuurissa. Niin sanottu "fan fiction" -kirjallisuus on pitkään tarjonnut suosituille romaani- ja elokuvahahmoille elämän virallisten tarinoidensa ulkopuolella. Fan fiction -tarinoissa tunnettu kirja- tai vaikkapa TV-sarjan hahmo seikkailee jonkun muun kuin oikean kirjoittajansa kirjoittamassa tarinassa. Näitä tarinoita kirjoittavat harrastusluontoisesti hahmon vannoutuneet ihailijat, eli fanit. Usein tarinat voivat olla lyhyitä novelliluontoisia pätkkiä, mutta on täysin normaalia, että innokaat fanit jaksavat kirjoittaa näitä epävirallisia tarinoita kokonaisen romaanin verran. Ja taas kerran täytyy painottaa sanaa "harrastusluontoisesti", sillä tällaisten tarinoiden kaupallinen myyminen ilman alkuperäisen tekijän lupaa olisi tekijänoikeuslain vastaista. Itse asiassa ilmainenkaan tarinoiden levittäminen internetissä ei taida olla ihan laillista, mutta se ei tunnu juurikaan menoa haittaavan.

Internet on tietenkin antanut vauhtia myös fan fiction -kulttuurille. Eräs suosittu www-sivusto alan harrastajille on www.fanfiction.net, jossa uusia tarinoita julkaistaan useita päivässä. Alexandre Dumas'n kolme muskettisoturia taistelevat täällä 42 lisätarinan verran, Sherlock Holmes ratkaisee 347 ennen julkaisematonta mysteerä ja Homer Simpsonistakin on 91 fanitarinaa. Kukaan ei kuitenkaan pysty kilpailemaan Harry Potterin kanssa, joka reippaasti yli sadallatuhaanella fanitarinalla on jättänyt kauas taakseen jopa pitkään ykkössijaa pitäneen Star Warsin (8800 tarinaa). Täältä voit käydä esimerkiksi lukemassa, mitä Harry puuhailee kesälomillaan, eli virallisten J.K. Rowlingin kirjoittamien tarinoiden välillä, tai mitä tapahtuu kun Harry kasvaa täysi-ikäiseksi ja menee yliopistoon!⁶⁷

67...ja alkaa muun muassa seurustelemaan. Eräs suosittu fan fiction alalaji on K-18 tarinat.

52. WIKIPEDIA, ÄÄRIMMÄISEN AVOIN TIETOSANAKIRJA

Vaikka monet kirjat soveltuvat parhaiten yhden henkilön kirjoitettaviksi, on toisenlaisia julkaisuja, joilla on tyypillisesti useampia tekijöitä – joskus hyvinkin monta. Yksi tärkeä tällainen julkaisujen alatyyppejä on tietosanakirjat.

Tietosanakirjathan, päinvastoin kuin Harry Potterit ja muut romaanit, koostuvat erillisistä artikkeleista. Kunkin hakusanan artikkelin kirjoittajaksi pyritään sitten saamaan kyseistä alaa hyvin tunteva asiantuntija. Tietosanakirjaa ei myöskään tarvitse kirjoittaa missään tiettyssä järjestyksessä, vaan kaikki tekijät voivat omilla tahoillaan samanaikaisesti paneutua omaan osaansa. Näyttäisi siltä, että tietosanakirjan voisi hyvinkin saada julkaistua juuri Open Source -metodeja käyttäen. Ehkä et olekaan kovin yllättynyt, jos kerron, että tällainen tietosanakirja onkin olemassa.

37-vuotias *Jimmy Wales* oli Linus Torvaldsin ja Richard Stallmanin ihailija ja hänellä oli myös visio oman Open Source -projektin käynnistämiseksi. Koska Jimmy ymmärsi jonkin verran Open Source -projektien dynamiikasta, hänkin oli tajunnut, että tietosanakirjan tekeminen soveltuisi hyvin Open Source -ideologiaan. Niinpä hän vuonna 1999 perusti Nupedia-projektin.

Nupediasta ei kuitenkaan tullut sitä, mitä Jimmy oli suunnitellut. Kahden vuoden aikana projekti onnistui kuluttamaan merkittävän summan määrärahoja ja sai tuona aikana julkaistua kokonaiset kaksitoista tietosanakirja-artikkelia. Siis 12 artikkelia kahdesa vuodessa! Mikä meni vikaan?

Nupedia oli periaatteessa kaikille avoin projekti – kuka tahansa sai osallistua artikkelien kirjoittamiseen. Käytännössä tekijäksi ei kuitenkaan päässyt noin vain. Tulevan kirjoittajan piti ensinnäkin lähettää hakemus, joka sitten hyväksyttiin joskus ja jossain. Kaikki artikkelit myös tarkastutettiin ja muokattiin useampaan kertaan muilla tekijöillä – näinhän tietosanakirjoja yleensäkin tehdään. Ja tämän monimutkaisen tarkastusprosessin siis läpäisi loppujen lopuksi 12 artikkelia! Näin ei siis selvästikään kannata organisoida Open Source -projektia.⁶⁸

Epäonnistuneen Nupedian tilalle syntyikin tammikuussa 2001 *Wikipedia*.⁶⁹ Se hyödynsi 90-luvun lopulla levinnyttä *WikiWikiWeb*-tekniikkaa. Wiki on www-sivu, jota voi paitsi lukea, myös kirjoittaa ja muokata. Alun perin wikejä käytettiin esimerkiksi ryhmätyöskentelyssä tai yritysten sisäisillä www-sivuilla intraneteissa, mutta viimeistään Wikipedia on todistanut tekniikan mahdollisuudet myös julkisen, kaikille avoimen internetin puolella.

Wikissä jokainen sivu sisältää esimerkiksi alareunassaan linkin, josta sivua pääsee välittömästi muokkaamaan. Tähän ei tarvita mitään erityistä ohjelmaakaan, vaan muokattava sivu aukeaa suoraan www-selaimeen, jolla sivua luetaankin.⁷⁰ Joissain tapauksissa sivun muokkaamiseen ei tarvitse edes käyttäjätunnusta, mutta silloinkin kun kirjoittaminen edellyttää oman tunnuksen rekisteröimistä, pääsee uusi käyttäjä muokkaamaan sivua heti rekisteröinnin jälkeen. Vaivaton osallistuminen muokausprosessiin on olennaista, kaikenlainen tarkistus, esisensuuri ja uusien käyttäjien hyväksymisprosessit ovat pannassa.

Useimmilla jotka ensimmäistä kertaa kuulevat Wiki-tekniikasta, laukeaa yleensä jonkinlainen välitön epäilyrefleksi. Tämä ei voi toimia oikeasti! Myönnän itsekin ajatelleeni näin, kerran kauan sitten.

Jos julkisessa internetissä oleva sivu on vapaasti kenen tahansa muokattavissa, niin eikö tämä ole suorastaan hankaluuksien kerjäämistä? Voisi kuvitella, että kaiken maailman vandaalit ja muutten vaan murrosikää potevat hyökkäävät sankoin joukoin tällaisen tilaisuuden kimppuun.

Vaikka wiki-filosofiaan kuuluukin mahdollisimman esteetön ja tehokas pääsy sisällön tuottamiseen ja julkaisemiseen, ei se tietenkään tarkoita, etteikö minkäänlaisia tarkistuksia saisi tehdä koskaan ikinä. Niinpä uusien artikkelien kirjoittamisen lisäksi Wikipedian laaja vapaaehtoisten armeija osallistuu myös tehtyjen muutosten tarkistamiseen. Tietosanakirja-tyyppisessä sivustossahan tämä on välttämätöntä jo pelkästään faktojen tarkistamisen ja oikolukemisenkin takia, mikä on ihan yhtä tärkeä työtehtävä kuin itse kirjoittaminen.

Kaikissa wiki-ohjelmissa on näppäriä työkaluja tarkistusten helpottamiseksi. Kaikki uusimmat muutokset löytyvät kootusti omalta sivultaan, josta niitä on helppo käydä läpi. Diff-työkalu näyttää suoraan muuttuneet osat, joten muuttunutta sivua ei välttämättä tarvitse lukea läpi kokonaan. Ja jos kyseessä on ilkeä, tai muutten vaan ei satu jostain tekstistä tykkäämään, niin muutosten peruminen on yhtä helppoa ja kaikille avointa kuin niiden tekeminenkin oli.

Jonkin verran vandalismia siis tietenkin esiintyy, mutta tällaiset yritykset jäävät kuitenkin hyvin lyhytikäisiksi. Asiallisilla kirjoittajilla on häiriköitä vastaan niin suuri miesylivoima, ettei häiriöitä juurikaan ehdi Wikipediassa näkemään. Tosin jotkut kohteet ovat ilkeäntekijöille liian houkuttelevia. Wikipedian etusivu ei nykyään enää olekaan vapaasti editoitavissa, koska jossain vaiheessa muodostui kansanhuviksi laittaa siihen jättimäisiä peniksen kuvia.

Muurahaisia on yleensä pidetty klassisena esimerkkinä ahkerista työntekijöistä. Varmasti jokainen pieni lapsi on joskus marssitettu muurahaispesän luo ihmettelemään, miten niin pienet ötökät voivatkaan saada rakennettua noin ison keon. (Ja tarinan opetushan on, että yhteispelillä se sujuu ja ahkeruus ja kyllä te tiedätte, sillä teidätkin on pienenä viety katsomaan sitä kekoa.)

Luin kerran jostain⁷¹ erästä tutkimuksesta, jonka mukaan muurahaispesässä noin 20% muurahaisista tekee täysin typeriä juttuja, kuten purkavat seinää, jonka toiset ovat juuri rakentaneet tai vievät vasta kerätyn ruokavaraston jonnekin, mistä kukaan ei löydä sitä tai jotain muuta, joka käytännössä sabotoi kaikkea sitä, mitä muut ahkerasti yrittävät rakentaa. Teorian mukaan nämä muurahaiset eivät ilmeisesti kuitenkaan ole ilkeitä, vaan yksinkertaisesti vain tyhmiä.

Tulit varmaan ajatelleeksi samaa kuin minäkin, että tiedät joitakin organisaatioita ihan ihmismaailmassa, joissa vähintään 20% tuntemistasi tyypeistä ovat juuri noin ääliömäisiä! Yleensä meillä ihmisillä onkin organisaatioissamme keinot, joilla näitä muurahaispesän ongelmia estetään. Jos joku tekee jotain typerää, niin tehdään säännöt, joissa se kielletään. Sitten pitää myös olla sääntö, että kaikkien pitää lukea säännöt. Sitten tarvitaan paljon päälliköitä ja muita valvoja, jotka valvovat, että sääntöjä luetaan ja noudatetaan ja että kukaan ei vahingossakaan tee mitään typerää. Loppujen lopuksi meillä on organisaatioissamme iso joukko ihmisiä miettimässä sääntöjä, kirjoittamassa sääntöjä ja valvomassa sääntöjä. Ja ne jotka jäävät jäljelle, ovat ensisijaisesti huolissaan siitä, etteivät vain tulisi rikkoneeksi sääntöjä ja tekevät sen takia kaiken hyvin varovasti.

Open Source -projekteja on kritisoitu siitä, että niiden olematon hierarkia ja organisoinnin puute johtaa tehottomuuteen. Kun kukaan ei valvo kokonaisuutta, käy niin, että moni voi vahingossa tehdä samaa työtä kuin joku toinen ja joku kolmas voi tulla tehneeksi jotain täysin turhaa, jota ei kukaan ikinä tule tarvitsemaan. Tällöin meidän vaistomainen refleksimme kehoittaa meitä lisäämään sääntöjä ja lisäämään päälliköitä ja lisäämään suunnittelua.

Linux ja Wikipedia kuitenkin todistavat päinvastaista. Säännöt ja suunnittelu, se se vasta turhaa työtä onkin. Ei muuta kuin suoraan asiaan ja kaikki puurtamaan olennaisen parissa. Suunnittelulla rakennetaan Nupedioita (ja Linuksen mukaan Hurd-kerneleitä), muurahaiskekomaisella reippaalla työnteolla taas tehdään Linuxeja ja Wikipedioita.

Varsinkin ottaen huomioon, että eivät ne suunnittelijatkaan aina niin fiksuja ole. Tällöin voi käydä niin, että noudattamalla huonoa suunnitelmaa koko organisaatio saadaan tekemään jotain todella typerää. Pieni päällekkäisyys ja suunnittelemattomuus on siis vain hyväksi, onpahan ainakin isompi todennäköisyys, että joissain päällekkäisistä projekteista on jotain järkeä.

Wiki-filosofian mukaan suuri enemmistö ihmisistä osaa ja haluaa tehdä oikein ja suorastaan haluaa näyttää itsestään parhaat puolet. Tämän halun kahlitseminen sääntöihin ja turhiin tarkistusprosesseihin on paljon ääliömäisempää kuin se pieni haitta, mitä se pieni vähemmistö muurahaisista joskus saa aikaan. Muista aina, että siitä 20% prosentista huolimatta miljoonat isät ja äidit käyvät joka kesä lastensa kanssa nimenomaan ihaillessaan muurahaiskekoa, eivät kritisoi massa sitä.

Wikipediallakin on sääntöjä,⁷² joita kirjoittajien tulisi noudattaa. Tärkein sääntö kuitenkin kuvaa osuvasti wiki-filosofiaa: "Jos joku sääntö saa sinut masentumaan tai hermostumaan, niin että et enää haluakaan osallistua wikiin, unohda silloin säännöt kokonaan ja tee mitä alunperin aioitkin tehdä."

Nupedian kaksi vuotta olivat käytännössä yhtä tyhjän kanssa. Tietenkin vahingosta viisastuu, mutta muuten ei juuri mitään jäänyt käteen. Tätä kirjoitettaessa Wikipedia on juuri täyttänyt kolme vuotta, miten sille on käynyt?

Wikipedian alkutaipaleella oli Jimmy Walesin lisäksi mukana *Larry Sanger*, joka työskenteli palkattuna Wikipedian parissa ensimmäiset 13 kuukautta. Tämä on mielenkiintoinen yksityiskohta, jos verrataan Wikipediaa Linuxiin. Linuskin ehti työskennellä Linuxinsa parissa enemmän tai vähemmän yksin noin vuoden, ennen kuin muut ohjelmoijat ympäri maailmaa alkoivat vähitellen tuottamaan yhä suuremman osan Linuxin koodista. Molemmissa tapauksissa tarvittiin joku työntämään aluksi vauhtia, pieni alkupanos, jonka jälkeen matka lähtee sujumaan itsestään.

Jos eksyi käyttämään Wikipediaa sen ensimmäisen elinvuoden aikana, saattoi usein käydä niin, että haettua sanaa ei löytynyt, koska kukaan ei ollut vielä kirjoittanut siitä artikkelia. Tällöin kävijä ei luonnollistikaan saanut eteensä mitään virheilmoitusta tai tyhjää sivua, vaan wiki-filosofian mukaisesti sivulla odotti silloin tyhjä tekstikenttä, johon puuttuva artikkeli ystävällisesti pyydettiin kirjoittamaan. Vähitellen niitä sitten kertyi, vaihtelevan tasoisia. Jotkut olivat ylä-asteikäisten, englantia vieraana kielenä opiskelevien koululaisten, toiset taas oppineen oman alan professorin kirjoittamia. Vapaaehtoisuuteen perustuva työtapa jätti tietenkin omat jälkensä artikkeleihin. Wikipediassa on jo varhain ollut erittäin kattava selonteko hakusanoista "beer" (olut) tai "Linux". "Microsoft Windows" -käyttöjärjestelmää käsittelevässä artikkelissa taas mainitaan ilmeisen tärkeänä faktana myös emulaattorit, joiden avulla Windows-ohjelmat voi siirtää Linuxille! Voisi sanoa että Wikipedia on "itsensä näköinen" tietosanakirja.

Nykyään Wikipedialla on vapaaehtoisia toimittajia, niin kutsuttuja *Wikipediaaneja*, jo useita tuhansia. Tammikuussa 2002, siis vuoden ikäisenä, uusia kirjoituksia ja korjauksia tehtiin jo tuhat päivässä, seuraavan vuoden tammikuussa luku oli tuplaantunut kahteentuhanteen päivässä. Artikkeleiden lukumäärä lähenee tätä kirjoitettaessa jo 200 000 rajaa ja sen lisäksi Wikipediasta on aloitettu käännösten tekeminen yhteensä 55:lle muulle kielelle. Esimerkiksi suomenkielisessä Wikipediassa⁷³ artikkeleita oli jo sievoiset 3200 kappaletta.

Mielenkiintoinen tilastotieto on myös, että kesällä 2003 Wikipedia ohitti kävijöiden määrässä mitattuna kunniakkaan *Encyclopedia Britannican* verkkoversion. Vertailu ei ole aivan reilu, sillä Britannican artikkelit ovat luettavissa vain maksaville asiakkaille, mutta on siis kuitenkin täysin korrektia väittää, että kolmen vuoden ikäinen Wikipedia on internetin suosituin tietosanakirja. Sitäpaitsi Jimmy aikoo myös haastaa Britannican sen omalla kotikentällä. Wikipediasta tullaan lähitulevaisuudessa julkaisemaan *Wikipedia 1.0*, valikoidut 75 000 artikkelia painettuna paperille ja saatavana myös CD-romppuna.⁷⁴

Wikipedian yhteydessä huomautetaan usein, että kaikkien sanakirjojen⁷⁵ kuningas, *Oxford English Dictionary* oli myös tavallaan Open Source -projekti. Sen ensimmäinen painos julkaistiin 1928 ja siinä oli kahteentoista niteeseen sidottuna luetteloitu yli 400 000 englannin kielen sanaa. Ensimmäistä painosta oli työstetty yhteensä 54 vuotta ja ensimmäiset päätoimittajat olivat julkaisuvuonna jo siirtyneet manalan majoille.

OED:n nimittäin kirjoittivat suurimmaksi osaksi lukuisat hankkeesta innostuneet vapaaehtoiset kielitieteen harrastajat, jotka keräsivät sanoja niin sanomalehdistä ja kaunokirjallisuudesta kuin toreilta ja kulmakuppiloista. Sanat ja niiden selitykset lähetettiin sitten kirjeitse päätoimittajalle, jonka tehtävä oli koordinoida projektia. Eräs tunnettu vapaaehtoistyöntekijä oli J.R.R. Tolkien, jonka kerrotaan työstäneen lähinnä W:llä alkavia sanoja. (Faneille tiedoksi, että ainakin "warm", "wasp", "water", "wick", "wallop", "waggle" ja "winter" ovat alunperin hänen käsialaansa, joten nyt voitte käydä lukemassa vielä yhden Tolkienin kirjan, jota ette aiemmin ole lukeneet.) Murhasta tuomittu William Chester Minor taas kirjoitti sanakirjaa vankisellistään. Ja erään hollannissa asuneen rouva Calandin kerrotaan sanoneen, ettei hän voinut sietää miehensä ainaista uppoutumista "siihen kirottuun sanakirjaan".⁷⁶ Vaikka OED lopullisessa julkaistussa muodossaan ei ollutkaan vapaasti kopioitavissa, niin näyttäisi siltä, että hakkerihenkinen työtapa tunnettiin jo silloin.

68Miten sitten Open Source -projekti kannattaa organisoida, jotta se olisi mahdollisimman menestyksenkäs? Tästä aiheesta kirjoitti Eric S. Raymond kuuluisassa teoksessaan "The Cathedral and the Bazaar":
<http://www.catb.org/~esr/writings/cathedral-bazaar/>

69<http://wikipedia.org/>

70Tyyppillisesti Wiki-sivun muokkaamisessa ei myöskään käytetä html-koodia, mikä olisi keskiverto surffaaajalle liian vaikeaa, vaan sivun sisältö kirjoitetaan hyvin yksinkertaisella, tavallista tekstiä muistuttavalla koodilla, jonka wiki-ohjelma sitten muuttaa html-sivuksi.

71Okei, tässä pitäisi nyt olla lähdeviite, mutta ei kai hyvää juttua kannata sellaisen pikkuseikan takia jättää kertomattakaan!

72<http://en.wikipedia.org/wiki/Wikipedia#Policies>

73<http://fi.wikipedia.org/>

74Jimmy Walesista ja Wikipediasta kirjoitti Wired-aikakauslehti marraskuussa 2003:
<http://www.wired.com/wired/archive/11.11/opensource.html> . Artikkelisi sisältää monia muitakin esimerkkejä Open Source -metodien soveltamisesta mitä erilaisimmilla ihmiselämän ei-tekniisillä alueilla. Näistä joitain on mainittu myös tässä kirjassa, toisia taas ei.

75Huomaa että Oxford English Dictionary siis on *sanakirja* (engl. dictionary) kun taas Wikipedia ja Britannica ovat *tietosanakirjoja* (engl. encyklopedia). Tietosanakirjan tavoitteena on kertoa sinulle "kaikki mitä ikinä halusit tietää", kun taas sanakirjan tavoitteena on luetteloida kielen kaikki sanat, niiden ääntämishjeet, syntyhistoriat ja eri merkitykset. Wikipediasta on olemassa myös rinnakkaisprojekti nimeltä Wiktionary (<http://wiktionary.org>) jonka tavoitteena on tuottaa monikielinen sanakirja. Wiktionary oli tätä kirjoitettaessa jo vuoden vanha, ja englanninkieliseen laitokseen oli kertynyt yli 30 000 sanaa, eli vajaa 10% OED:n ensimmäisen laitoksen sanamäärästä.

76Kirjasta "The Meaning of Everything, The Story of the Oxford English Dictionary". Simon Winchester.
Lähde: <http://www.ralphmag.org/CI/oed.html> ja <http://www.askoxford.com/worldofwords/oed/wordsearchers/?view=uk>

53. DMOZ.ORG, AVOIN WWW-SIVUJEN HAKEMISTO JA MOZDEX, AVOIN HAKUKONE

Wikipedian lisäksi on tässä mainittava joitakin muita vastaavia projekteja. Wikipediaa ja Nupediaa vanhempi *dmoz.org Open Directory Project*⁷⁷ on internetin linkki-hakemisto. Ensimmäinen ja edelleenkin suosituin vastaava (suljettu) hakemisto on internet-buumin ikoniksikin noussut Yahoo!⁷⁸

Vuonna 1998 näytti siltä, että automatisoidut hakukoneet olivat tulleet tiensä päähän. Suosituin hakukone Altavista ei pystynyt käsittelemään räjähdysmäisesti kasvavaa www-sivujen määrää, vaan hakutulokset olivat aina vaan käyttökeltomampia. Jo pelkkä www-sivujen määrä aiheutti sen, että oli vaikea enää löytää hakemaansa ja tilannetta vielä pahensi se, että erilaiset markkinointifirmat olivat oppineet väärinkäyttämään www-sivujen avainsanaluetteloita päästäkseen hakutulosten kärkeen. Tilanne oli vähän niinkuin tänä päivänä sähköpostin kanssa: teitpä mitä tahansa niin asiallisimmillakin hakusanoilla hakutulosten kärjessä oli enimmäkseen pornoa ja... no, lähinnä pornoa, sillä Viagraahan ei vielä silloin tunnettu.⁷⁹

Niinpä tulevaisuus näyttikin olevan yhä enemmän Yahoo!n kaltaisten ihmistyövoimalla koottujen hakemistojen varassa. Tähän markkinarakoon isivät *Chris Tolles* ja *Rich Skrenta*. Heille oli selvää, että räjähdysmäisesti kasvavan webin kokoaminen yhtään mihinkään hakemistoon suorastaan vaati Open Source -tyyppistä lähestymistapaa. Niinpä he kesäkuussa 1998 perustivat tarkoitusta varten projektin, jolle antoivat nimen – kuinka pieni maailma muuten onkaan – GnuHoo, eli avoin Yahoo!. Koska kyseessä ei kuitenkaan ollut virallinen Gnu-projekti, he Richard Stallmanin pyynnöstä vaihtoivat nimeksi NewHoo. Myöhemmin Yahooon kanssa kilpaileva Netscape luopui omista hakemistohankkeistaan ja osti NewHoon oman portaalsina alustaksi, jolloin siitä tuli Open Directory ja lopuksi se sai kodin osoitteesta dmoz.org.

Dmoz, eli NewHoo, oli alusta asti menestys. Jo ensimmäisen kuukauden aikana se oli kerännyt 31 000 linkkiä, jotka 400 vapaaehtoista toimittajaa olivat järjestäneet 3 900 kategoriaan. Vain viikkoa myöhemmin toimittajia oli jo 1200 ja linkkejä 40 000!⁸⁰

Samana vuonna 1998 automatisoitujen hakukoneiden valtiut siirtyi Altavistalta uudelle tulokkaalle Googlelle, jonka kehittyneen PageRank-algoritmin ansiosta hakutulosten järjestykseen saatiin taas jotain tolkkua. Samalla kisa automaattisten hakukoneiden ja toimitettujen hakemistojen välillä taisi taas kääntyä automatisoitujen hakukoneiden eduksi. Merkillepantavaa kuitenkin on, että Google ja monet muut hakukoneet käyttävät dmozia yhtenä tietolähteenä oman tietokantansa luonnissa. Käyttäessäsi Googlea käytät siis Open Sourcea monella tavoin. Ensinnäkin Googlen palvelimet perustuvat Linuxiin ja Open Source -koodiin⁸¹ ja toiseksi Googlen hakukone hyödyntää Open Source -tietolähdettä.

Vaikka toimitetut hakemistot hävisivät kisan Googlea vastaan, mutta Open Source yhteisö ei silti luovuttanut. Huhtikuussa 2004 putkahti hakukonekisaan uusi tulokas nimeltä *MozDex*.⁸²

MozDex on täydellisen avoin hakukone. Se ei ainoastaan perustu Open Source koodiin, vaan tarkoituksena on olla avoin myös mitä hakutulosten esittämiseen tulee. Mistä tiedämme, että hakutulos, jota Google tarjoaa ensimmäisenä, on oikeasti paras? Voimmeko olla varmoja, ettei joku Googlen työntekijä ole käynyt sorkkimassa tietokantaa, tai ettei ykköspaikkaa ole myyty parhaiten maksavalle? Vaikka luotammekin Googleen, niin emme voi tietää varmasti. MozDexin tavoitteena on tarjota meille hakutuloksia, joista jokainen voi tarkistaa, miksi parhaiten sijoittuneet linkit todella ovat parhaita. Jokaisen hakutuloksen vieressä on "explain"-linkki, josta pääsee tarkistamaan, millä perusteilla tietty hakutulos on saanut pisteitä tietokannan muihin sivuihin verrattuna.

MozDexin hakukoneen ohjelmoijan *Doug Cuttingin* sanoin: "Internet-hakukoneiden määrä on laskussa. Tämän päivän harvainvalta voi kohta muuttua monopoliksi, jolloin yksi yhtiö hallitsisi kaikkea tiedonhakua ja hyödyntäisi tätä asemaansa taloudellisesti omaksi edukseen. Tämä ei ole internetin käyttäjien kannalta toivottavaa kehitystä".

Tätä kirjoitettaessa MozDex on vasta kokeiluasteella, eikä sen tietokannasta vielä löydy läheskään kaikkia internetin www-sivuja. Mutta 9. huhtikuuta 2004 jää kuitenkin historiaan päivänä, jolloin Open Source tuli mukaan hakukonekisaan. Google voitti ensimmäisen erän, mutta iskeekö Open Source vielä takaisin? Se jää nähtäväksi.

77<http://dmoz.org/>

78<http://www.yahoo.com/>

79Viagra tuli markkinoille vasta 1998 (<http://en.wikipedia.org/wiki/Viagra>) ja ainakin minun sähköpostiini tulvivan roskapostin joukossa viagra-mainokset ovat suurin yksittäinen joukko.

80Wide Open News, 12.6.1999: "License to search".
<http://web.archive.org/web/20011108043741/www.wideopen.com/story/224-2.html>

81Tosin itse Googlen koodi on kuitenkin huippusalaista.

82<http://www.mozdex.com/>

54. KOULUKIRJOJA, KEITTOKIRJOJA, MUUTA FAKTAA JA VÄHÄN FIKTIOTAKIN

Yksi kirjatyyppejä, joka on läheistä sukua Linux-käyttöoppaille ja tietosanakirjoille on koulu- ja oppikirjat. Myös oppikirjojen sisältö on usein suhteellisen helposti jaettavissa osiin, jolloin Open Source -menetelmää voidaan käyttää. Oppikirjat ovat myös houkutteleva aihepiiri siksi, että aika usein opettajat joka tapauksessa tuottavat omaa materiaalia virallisen oppikirjan ohuen – tämä on yleistä varsinkin yliopistoissa. Joillain yliopistokursseilla ei ehkä sopivaa kirjaa ole löytynyt ollenkaan, vaan opiskelijat ovat täysin luennoitsijan luentokalvojen varassa. Ja edelleen voi joillain kursseilla olla niin, että kurssin aihepiiri ei ole löydettävissä mistään yhdestä kirjasta, vaan luentorunko nojaa kahdesta tai useammasta kirjasta koottuihin palasiin.

Mikä voisi olla paremmin Open Sourcelle soveltuva tilanne. Jos oppikirjat olisivat vapaasti opettajien käytettävissä – ja nyt siis sanalla "vapaasti" tarkoitetaan nimenomaan Open Source -tyyppistä vapautta, eli vapaata kopiointioikeutta ja vapaata muuntelu-oikeutta – niin niistä voisi helposti koota opettajan omaan opintosuunnitelmaan täydellisesti istuvan räätälöidyn oppikirjakokonaisuuden. Kustakin tarjolla olevasta kirjasta voisi poimia parhaat palat, turhat luvut voisi jättää kokonaan pois ja jos joku luku ei oikein vastaa sille asetettuja korkeita vaatimuksia, voi opettaja täydentää sitä itse tai jopa kirjoittaa sen kokonaan uusiksi. Näinhän opettajat tekevät paljolti jo nytkin, mutta sen sijaan että lopputuloksena olisi sekava vanhentuneiden kirjojen, kopioiden ja käsinkirjoitettujen muistiinpanojen sekamelska, olisikin vapaan kopioinnin sallivassa järjestelmässä lopputuloksena täydellinen oppikirja, joka eräänlaisena evolutiivisena prosessina kehittyisi yhä täydellisemmäksi ja täydellisemmäksi ja päivittyisi sitä mukaa kun kehitys asettaa oppiaineelle uusia vaatimuksia.

Myös opiskelijoille Open Source -oppikirjat olisivat luonnollisesti houkutteleva vaihtoehto. Monesti opintojen loppuvaiheessa ei ole epätavallista, että johonkin kapeaan tekniikan alaan keskittyvä kurssikirja maksaa lähemmäs kahtasataa euroa. Avoin kurssikirja tulisi kustantajille luonnollisesti halvemmaksi painattaa, aivan samalla lailla kuin Linux on halvempi kuin Windows. Sitäpaitsi opiskelijalla olisi avoimen kurssikirjan tapauksessa myös mahdollisuus tulostaa kirja itse, tai tulostaa vain tarvitsemansa sivut, jolloin tarvittavan materiaalin hankintakustannukset laskisivat edelleen, ja kaupan päälle puitakin säästyisi! Ja ääritapauksessa köyhä opiskelija voisi lukea vaikka koko kirjan tietokoneen ruudulta, täysin ilmaiseksi.

Kun perustelut Open Source -oppikirjojen puolesta ovat näin vakuuttavat, olisi kumma, jos niitä ei jo löytyisi Internetistä. Ja löytyyhän niitä. Viime vuosina Internet on suorastaan tulvinut erilaisia projekteja avoimien oppikirjojen kirjoittamiseksi. Jotkut projektit ovat vasta ideointiasteella, kun taas toisilla on jo esittää valmista tavaraa. On kansallisia, tietyn maan opintosuunnitelmaa noudattavia projekteja ja kansainvälisiä (eli englanninkielisiä) projekteja. Jostain syystä jopa Kalifornian osavaltioilla on oma projekti yläasteikäisten koululaisten oppikirjojen tuottamiseksi.

Tästä tulevaisuuden oppikirjojen suoranaisesta runsaudenpulasta voi olla vaikea muodostaa minkäänlaista kokonaiskuva, mutta näyttäisi siltä, että wiki-filosofia on jälleen osoittanut voimansa. Nimittäin yksi pisimmälle päässeistä projekteista näyttäisi olevan Wikipedian koodiin perustuva *Wikibooks*,⁸³ joka on ainakin näin aluksi keskittynyt pääasiassa juuri kurssikirjojen tuottamiseen, tosin erilaisia oppaitakin löytyy. Wikibooks on tätä kirjoitettaessa vasta puolen vuoden ikäinen, mutta jo nyt enemmän tai vähemmän valmiita englanninkielisiä kirjoja on kymmeniä, yksittäisiä lukuja jo yli tuhat. Suomenkielisessä Wikibooksissa oli tätä kirjoitettaessa yksi kirja, joka oli ehtinyt saada statuksen "valmis". Tämä oli sosiaalitieteen alaisuuteen kuuluva "Työoikeus".⁸⁴ Wikissä mikään ei tietenkään ole koskaan lopullisesti valmista, vaan tämä status on ehkä pikemminkin verrattavissa tietokoneohjelmien versionumeroon "1.0".

Mainittakoon tässä vielä, että Wikipedia näyttäisi poikineen kesällä 2003 jälkeläisiä oikein kunnolla. Jo mainittujen Wiktionaryn ja Wikibooksin lisäksi on käynnissä projektit Wikisource, kokoelma vapaita tekstejä, joita ei kuitenkaan ole alunperin kirjoitettu wiki-menetelmällä (Raamattu, kreikkalaiset filosofit, USA:n itsenäisyysjulistus, jne...); Wikitravel, avoin matkaopas; Wikiquote, kokoelma kuuluisia sitaatteja,⁸⁵ ja Wikiversity, Wikibooksia hyödyntävä yliopistotasoinen koulutusmateriaalin kokoelma.

Mutta Wikibooks ei suinkaan ole ainoa avoimia oppikirjoja sisältävä verkkosivusto, vaan tasokkaita kirjoja on internet pullollaan. Yksi kattavimmista hakemistoista tässä suhteessa on *Benjamin Crowellin* ylläpitämä *The Assayer*.⁸⁶ Assayer ei siis ole mikään wiki, vaan ainoastaan linkkihakemisto muualla internetissä sijaitseviin avoimiin kirjoihin. Itse hakemiston lisäksi sivustolla on myös mahdollisuus arvioida ja pisteyttää lukemaansa kirjaa.

Vaikka Assayerissä periaatteessa luetteloidaan mitä tahansa kirjallisuutta, näyttäisi siltä, että avoimia kirjoja löytyy eniten juuri käsittelemistämme aiheista, toisaalta on Linux-käyttöoppaita ja toisaalta oppikirjoja. Oppikirjojen joukossa matematiikka, fysiikka ja kemia ovat jostain syystä ylivoimaisesti parhaiten edustettuina. Näiden alojen läheisellä suhteella tietotekniikkaan ja sitä kautta Linuxiin saattaa hyvinkin olla vaikutusta. Toiseksi suurin alalaji onkin sitten fiktio ja kaunokirjallisuus, mikä sinänsä on ilahduttavaa. Elämähän ei ole koulua varten...

Benjamin itse on myös kunnostautunut avoimen kurssikirjallisuuden tuottajana kirjoittamalla kaksi yliopistotason fysiikan oppikirjasarjaa: "*Light and Matter*" ja "*Simple Nature*". Näistä vanhempaa *Light and Matter*ä käytetään jo kurssikirjana ainakin seitsemässätoista yliopistossa. Kaikesta päätellen Open Source -kirjallisuudella todellakin on jotain annettavaa tuleville opiskelijasukupolville.

Ja koska elämä ei tosiaan ole koulua varten (vaan koulu elämää) niin siirtykäämme lopuksi meidän kaikkien yhteiseen lempiaiheeseemme, eli ruokaan. Avoimia Linux-käyttöoppaita nimittäin täydentää loistavasti *Matthew Balmerin* "*The Open Source Cookbook: Fuel for Geeks*".⁸⁷

Kirjasta arvatenkin on hyötyä nuorille vasta kotoaan muuttaneille nörteille, jotka jonain päivänä heräävät todellisuuteen, että äiti ei enää häiritsekään pitkiä koodaussessioita kutsumalla aina välillä valmiiseen ruokapöytään. Koska keittokirja todellakin on suunnattu nörteille, selitetään ennen varsinaisia reseptejä huolellisesti ja kuvien kera, mitä ovat ja mihin tarvitaan sellaisia perustarvikkeita kuin kattila, mikroaaltouuni ja leikkuulauta. Tämän jälkeen käydään myös läpi peruselintarvikkeet, jotka olisi syytä löytää jokaisen nörtinkin keittiöstä, siis muutakin kuin kokista ja pakastepizzaa. Lopuksi on vielä lyhyt sanakirja, josta voi selvittää mistä on kysymys jos reseptissä puhutaan Juliennesta, marinoinnista tai vatkaamisesta.

Perusteellisen pohjatyön jälkeen itse reseptit ovatkin jo sellaisia, että ei hätää vaikka äitikin tulisi kylään. Kirjaa Slashdotin⁸⁸ keskustelupalstalla yhteisesti työstänyt kansainvälinen hakkeriyhteisö todella tarjoaa kirjassa parhaita palojaan. Mitä sanotte vaikka sellaisestakin gastronomisesta seikkailusta kuin friteerattu kenguru? Resepti luonnollisesti alkaa tiedolla siitä, mistä verkkokaupoista ei-australialaiset voivat tilata kengurunlihaa. Kyllä me nörtit osaamme!

Myös Wikibooksista löytyy muuten keittokirja,⁸⁹ joten nörttien terveys pitäisi tästedes olla turvattu. Kunhan vaan muistaisivat välillä irroitautua tietokoneidensa äärestä.

83<http://wikibooks.org/>

84<http://wikibooks.org/wiki/Tyoikeus>

85Esimerkiksi: "Minä elän.", kirjailija Aleksis Kivi; tai "Noin kaukaa he eivät osuisi edes norsuun...", kenraali John Sedgwick, pohjoisten joukkojen komentaja USA:n sisällissodassa. Molemmat sitaatit kokoelmasta "Kuuluisia viimeisiä sanoja", http://quote.wikipedia.org/wiki/Famous_last_words

86<http://www.theassayer.org/>

87<http://www.ibiblio.org/oscookbook/>

88<http://slashdot.org/>

89<http://wikibooks.org/wiki/Cookbook>

55. PROJECT GUTENBERG

Yksi erittäin ansiokas vapaasti levitettävään kirjallisuuteen liittyvä projekti, joka ilman muuta ansaitsee tulla mainituksi tässä yhteydessä, on *Project Gutenberg*.⁹⁰ Gutenberg-projektin palvelimelle on kerätty sähköiseen muotoon lähes kaikki vapaasti levitettävissä olevat klassikot, mitä maailman kirjallisuudesta löytyy.

Päin vastoin kuin tietokonealalla, joka on vielä kovin nuorta, on kirjallisuudessa iso joukko kuuluisia teoksia, joita saa vapaasti kopioida, julkaista ja levittää internetissäkin. Tämä johtuu eri maiden tekijänoikeuslakeihin sisältyvästä määräaikaaisuudesta. Tekijänoikeuden suoja on nimittäin voimassa vain tietyn ajan, jonka jälkeen teoksesta tulee *yleistä omaisuutta* (englanniksi *Public Domain*). Määräajan pituus vaihtelee eri maissa ja riippuu myös teoksen tyyppistä, mutta nykyään se voi olla esimerkiksi 50-70 vuotta tekijän kuolemasta.⁹¹ Koska sata vuotta sitten kuolleita ohjelmoijia ei juurikaan ole,⁹² ei tämä seikka ole juurikaan vaikuttanut esimerkiksi Linuxin kehitykseen. Mutta kun puhe on kirjallisuudesta, tilanne onkin toisenlainen. Yli-inhimillisen pitkiksi venyneistä suoja-ajoista huolimatta valtaosa kirjallisuuden klassikoista on jo vapautunut tästä suojasta. Niinpä Gutenbergin arkistoista löytyykin e-kirjoina kattava kokoelma antiikin filosofeista ja eri raamatunkäännöksistä lähtien aina Shakespearin näytelmiin ja Moby Dickiin ja edelleen Liisaan ihmemaassa ja Peter Paniin asti.

Project Gutenberg sai alkunsa *Michael Hartin* käsissä niinkin varhaisessa vaiheessa kuin 1971. Sähköisenä projektina kyseessä on siis todella pitkäikäinen projekti, samoihin aikoihin ottivat ensimmäisiä haparoivia askeliaan esimerkiksi ensimmäinen UNIX-käyttöjärjestelmä ja C-ohjelmointikieli. On siis suorastaan vähän ihmeellistä, että niinkin varhaisessa vaiheessa oli edes olemassa tietokonetta, jota Michael pystyi projektiin käyttämään. Illinoisin yliopistolla kuitenkin tällainen tietokone oli ja Michael oli yksi niistä, joille yliopisto päätti lahjoittaa tietokoneen käyttöaikaa.

Muut tietokonetta siihen aikaan käyttäneet tekivät sillä yleensä jotain hyvin alkeellista ohjelmointia, kuten esimerkiksi ohjelmoivat ohjelmointikieltä. Tämä on vähän niinkuin rakentaisi vasaraa, jolla sitten aikanaan rakennetaan rakennusteline, jonka varaan voidaan lopulta rakentaa talon seinät ja kun talo on valmis, naiset pääsevät sisustamaan sitä. Kaiken kaikkiaan oikein miehistä hommaa, verrattavissa siihen kun luolamies keksii tulen.

Michael ei kuitenkaan uskonut, että hänestä ikinä olisi sellaisiin projekteihin. Niinpä hän yritti keksiä tietokoneen käyttöajalle jotain muuta käyttöä. Olihan kyseessä kuitenkin resurssi, joka ostettuna olisi siihen aikaan ollut 100 miljoonan dollarin arvoinen! Ei sitä sopinut käyttämättäkään jättää.

Mietittyään asiaa tunnin ja 47 minuuttia – näin kerrotaan – Michael ilmoitti, että tulevaisuudessa tietokoneiden suurin arvo ei olisi niiden tarjoamassa laskentatehossa, vaan informaation tallennuksessa, sekä sen rajattomassa välittämisessä ja etsimisessä. Ei huonommin arvattu! Näin sai alkunsa Project Gutenberg, jonka ensimmäisenä tallenteena Michael naputteli koneelle USA:n itsenäisyysjulistuksen, joka julkisena asiakirjana juurikin oli yhteisomaisuutta – Public Domain.

Vaikka kyseinen itsenäisyysjulistus ei ehkä varsinkaan meidän muilla mantereilla asuvien mielestä kuulukaan länsimaisen kirjallisuuden Top 10 kärkeen, oli sen valinnalle puhtaan isänmaallisuuden lisäksi ihan käytännöllisiäkin perusteita. Se oli sopivan lyhyt, jotta sen sai mahtumaan sen ajan levykkeelle. Moby Dickien ja raamattujen aika tuli vasta myöhemmin, teknologian kehittyessä.

Vuonna 2003 Gutenberg-arkisto oli jo kasvanut yli 10 000 e-kirjaan, jotka julkaistiin 4 gigatavun kokoisella merkkipaalua juhlistavalla DVD-levyllä. Projektin tavoitteena on nyt julkaista vuoteen 2015 mennessä miljoona vapaata e-kirjaa. Pienestä alusta huolimatta Gutenberg-projekti on osoittanut, millaisia voimavaroja vapaudessa ja vapaaehtoistyövoimassa piilee.⁹³ Projektin historia myös muistuttaa meitä siitä, että nyt eletävä Open Source -ajattelun nousukausi ei itse asiassa olekaan mitään uutta keksintöä. Kuten Richard Stallmankin aina nuorempiaan opastaa, 1970-luvulla IT-ala oli Open Source. Muutos suljetumpaan tuli vasta myöhemmin. Sama pätee e-kirjoihin. Michael Hart oli liikkeellä jo 1970-luvulla, Stephen King vasta 2000-luvulla.

90<http://www.gutenberg.net/>

91IT-ässä yhteydessä on hyvä valistaa lukijaa myös asiaan liittyvästä sivujuonteesta. Alun perin teosten suoja-ajat olivat huomattavasti lyhyempiä, USA:ssa esimerkiksi vaivaiset 14 vuotta. Salaperäisesti aika on kuitenkin pidentynyt nykyisiin sukupolvet ylittäviin mittoihinsa. Mielenkiintoinen huomio on se, että suoja-aikaa on uusilla laeilla pidennetty aina silloin, kun Mikki Hiiren (julkaistu ensin vuonna 1928) tekijänoikeudet ovat olleet umpeutumaisillaan. Ilman viimeisintä 20 vuoden pidennystä 1998, Mikki Hiirestä olisi tullut meidän kaikkien yhteistä omaisuutta 2003.
http://writ.news.findlaw.com/commentary/20020305_sprigman.html
<http://www.wired.com/news/politics/0,1283,17327,00.html>
<http://reason.com/links/links011703.shtml>

92Yli sata vuotta sitten kuolleita ohjelmoijia löytyy tietotekniikan esihistoriasta ilmeisesti tasan kolme. Ohjelmoitavan kutomakoneen 1801 keksinyt ranskalainen Joseph-Marie Jacquard (http://en.wikipedia.org/wiki/Jacquard_loom) ja "differenssikoneen" sekä "analyttisen koneen" myös 1800-luvulla suunnitellut britti Charles Babbage (<http://en.wikipedia.org/wiki/Babbage>) sekä hänen tuttavansa Ada Lovelace (http://en.wikipedia.org/wiki/Ada_Byron), joka kirjoitti analyttiselle koneelle ohjelmia, vaikka sitä ei koskaan valmistettukaan. Myös jotkut muut 1800-luvun matemaatikot ovat pohdiskelleet ohjelmoitavan laskukoneen valmistamista, mutta nämä kolme ovat tyypillisesti ne, jotka mainitaan tietokoneiden esihistoriasta puhuttaessa.

93Lähteet: <http://www.gutenberg.net/about.shtml> ja <http://www.upi.com/view.cfm?StoryID=20040106-041656-1684r>

56. MUSIIKKI

Vaikka Internet on vaikuttanut vallankumouksellisesti myös musiikkimaailmaan, on suurimmat lehtiotsikot toistaiseksi kirjoitettu Napsterista ja sen jälkeläisistä, sekä levy-yhtiöiden näitä vastaan käymästä viivytystaistelusta. Vaikka nämä niin sanotut peer-to-peer tekniikat sinänsä ovatkin vallankumouksellisia, ei käy kieltäminen että niiden käyttäjät ovat tässä kohden menneet lain väärälle puolelle. Tekijänoikeussuojan alaista musiikkiahan ei saisi levittää internetissä ilman tekijän lupaa.

Vaikka käyttäjien välisen tiedostojenvaihdon ensimmäisenä mahdollistanut Napster joutuikin oikeuden päätöksellä lopulta päättämään päivänsä, ei mp3-tiedostojen vaihtorumba juurikaan hidastunut. Tilalle tuli sen sijaan kourallinen uusia tekniikoita, jotka oli teknisesti toteutettu niin, ettei niitä voitu sulkea yhdeltä keskuspalvelimelta samalla lailla kuin Napsteria. Oikeuksiaan ja ennen kaikkea rahavirtojaan suojeleva äänilevyteollisuus ei ainakaan vielä ole luovuttanut toivottomalta tuntuvaa taisteluaan, vaan yrittää pistää kapuloita rattaisiin milloin mitenkään.

Tätä kirjoitettaessa levy-yhtiöt ovat juuri onnistuneet pahentamaan jo ennestäänkin huonoa mainettaan muun muassa haastamalla oikeuteen tiedostonvaihto-ohjelmia käyttäneen varattoman newyorkilaisen yksinhuoltajaäidin 12-vuotiaan tyttären. Tiedostonvaihto-ohjelmia myyvät yritykset taas veivät pisteet kotiin maksamalla tyttöparan 2000 dollarin vahingonkorvaussopimuksen. Internetissä oltiin myös ehditty järjestämään kansalaiskeräys samaisen 2000 dollarin keräämiseksi. Jotenkin tuntuu siltä, että levy-yhtiöt ovat taistelussaan kovin yksin.

Viime vuonna monet artistitkin ovat ottaneet kantaa levy-yhtiöitä vastaan.⁹⁴ Tunnetuin näistä lienee elektronisen musiikin kärkinimiin kuuluva *Moby*,⁹⁵ joka on avoimesti puolustanut tiedostoja kopioivia fanejaan. Hän sanoo olevansa vilpittömän iloinen siitä, että ihmiset haluavat kuunnella hänen musiikkiaan ja jos he pystyvät kuuntelemaan sitä tiedostonvaihto-ohjelmien avulla niin se sopii hänelle oikein hyvin. Moby'n laskelmien mukaan moni hänen faneistaan on itse asiassa alunperin tutustunut hänen musiikkiinsa juuri internetistä lataamiensa tiedostojen kautta, seikka josta hän on pikemminkin iloinen kuin suutuksissaan. Moby'n lisäksi huhutaan monen muunkin muusikon salaa syöttävän omia kappaleitaan tiedostonvaihto-ohjelmiin, siis aivan päinvastoin kuin levy-yhtiöiden puheista voisi kuvitella. Offspring-niminen yhtye kertoi eräässä TV-haastattelussa jopa laittaneen uuden levynsä oheistuotteeksi joitakin mp3-tiedostoja, mutta levy-yhtiö oli kuitenkin poistanut ne ennen levyä julkaisemista.

Takavuosien Wham!-yhtyeen solisti George Michael onkin ensimmäisenä kovan luokan pop-tähtenä ehtinyt ilmoittamaan⁹⁶, että keväällä 2004 julkaistu levy jää hänen viimeiseksi. Sopimus levy-yhtiön kanssa päättyy ja vastaisuudessa hän aikoo julkaista kaikki uudet kappaleensa (vanhoihin levy-yhtiön omistamiin kappaleisiin sahan hänellä ei ole mitään oikeuksia) kotisivullaan, josta fanit saavat niitä vapaasti kopioida. Kappaleiden kopiointi tulee olemaan ilmaista, mutta kotisivun kautta on mahdollisuus lahjoittaa Georgen valitsemiin hyväntekeväisyyskohteisiin.

Ahneita isäntiään vastaan kapinoivat Offspringin rokkivaarit antavat kyllä suurisanaisia haastatteluja, mutta omien mp3-tiedostojen ujuttaminen salaa tiedostonvaihto-ohjelmiin kopioitavaksi ei kuitenkaan ole vielä Open Sourcea. Heitä ehkä harmittaa, että allekirjoitus jossain vanhassa levytyssopimuksessa kieltää heitä laittamasta omia kappaleitaan omille kotisivuilleen. George Michael osoittaa muille mallia oikeaan suuntaan, mutta se on vasta ensimmäinen askel. Tietävästi hänkään ei ole antamassa omaa musiikkiaan kenen tahansa muun hyödynnettäväksi tai myytäväksi samanlaisilla ehdoilla kuin esimerkiksi Linus antoi oman koodinsa.

Joitakin ihan oikeita vallankumouksellisia sentään löytyy. *Magnatune*-niminen pieni levy-yhtiö⁹⁷ julistaa etusivullaan: "We're a record label. But we're not evil." (Olemme levy-yhtiö, mutta emme ole pahoja.) Omistaja *John Buckman* perusti yhtiön nähtyään, miten hänen vaimolleen kävi perinteisen levytyssopimuksen ja perinteisen levy-yhtiön kanssa. Levyjä myytiin noin tuhat ja vaimo sai tekijänä korvausta huimat 45 dollaria. Sopimuksen seurauksena hän menetti kaikki oikeudet omaan musiikkiinsa seuraavaksi kymmeneksi vuodeksi, eikä siis voi esimerkiksi laittaa omaa musiikkiaan omalle kotisivulleen siitäkään huolimatta, että levy on myyty loppuun eikä sitä voi ostaa tai tilata enää mistään. John halusi perustaa levy-yhtiön, joka olisi artistin kannalta tällaista perinteistä järjestelyä parempi vaihtoehto.

Magnatunen kotisivulta voi yhtiön levyttämää musiikkia ladata täysin laillisesti joko yksittäisinä kappaleina tai jopa kokonaisen levyn kerrallaan. Lisäksi sivulta voi kuunnella jatkuvaa henkilökohtaista nettiradiolähetystä, jonka musiikkitarjonnan kukin kuulija voi säätää mieleisekseen. (Näinkin teknisesti yksinkertaisen ja hauskan idean toteuttaminen on perinteisten tekijänoikeuskäytäntöjen alaisuudessa käytännössä mahdotonta.) Luonnollisesti yhtiöltä voi myös tilata ihan oikeita fyysisiä CD-levyjä, siitähän viime kädessä on kysymys. Valikoima on nopeasti kasvanut suhteellisen kattavaksi. Magnatunen leiriin kuuluu jo yli sata artistia ja levyjä on ehditty julkaisemaan jo kaksisataa. Lahduttavaa on, että myös eri musiikkilajit ovat hyvin edustettuina. länikuisen teknon lisäksi varsinkin klassista musiikkia löytyy paljon ja jazziaakin jo ihan mukavasti.

Magnatunen musiikki myös ylittää riman, mitä avoimuuden vaatimuksiin tulee. Vaikka musiikin kaupallisesta käytöstä pitääkin sopia ja maksaa erikseen, on sivulta löytyvä musiikki käytettävissä kaikkiin epäkaupallisiin tarkoituksiin Open Source -periaattein. John käyttää periaatteesta termiä *Open Music*. Musiikkia voisi siis käyttää esimerkiksi oman elokuvansa taustamusiikkina, tai jostain kappaleesta voi tehdä oman version. Monet artistit julkaisevatkin tätä silmällä pitäen Magnatunen sivuilla myös kappaleidensa "lähdekoodia", eli esimerkiksi midi-tiedostoja tai erillisiä ääniraitoja. Ainoa ehto kappaleiden uusiokäytölle on, että käytön pitää olla ei-kaupallista ja alkuperäiset tekijät on myös mainittava. Ja ennen kaikkea myös uutta teosta pitää jakaa muille näillä samoilla ehdoilla.

Viimeinen "ShareAlike"-ehto muistuttaa Richard Stallmanin copyleft-periaatetta, jota suurin osa Linux-ohjelmista käyttää. Toisaalta taas Open Source -tietokoneohjelmissa on nähty hyväksi luopua epäkaupallisuuden vaatimuksesta. Koska ohjelmien levityksestä kuitenkin muodostuu ihan oikeita kuluja – esimerkiksi tyhjän CD-levyn hinta tai postituskulut – ei epäkaupallisuus tuntunut edistävän GNU-liikkeen tavoitteita, joka oli GNU-ohjelmistojen levittäminen mahdollisimman laajalle. Historia onkin osoittanut, että kaupallisuus ei ole ollut Linuxille haitaksi – päinvastoin. Muut Stallmanin GPL-lisenssiin sisältyvät avoimuuden vaatimukset ovat riittäneet takaamaan avoimuuden ja reiluuden säilymisen ja esimerkiksi estäneet monopoliin syntyminen.

Vaikka Magnatunen musiikin käyttöehtoihin sisältyykin vielä epäkaupallisuuden vaatimus, on yhtiön toiminta kuitenkin askel oikeaan suuntaan, varsinkin verrattuna perinteisten levy-yhtiöiden suoranaiseen riistokapitalismiin. Ehkäpä epäkaupallisuuden vaatimuskin on vain välttämätön ylimenovaiheen paha, josta voidaan tulevaisuudessa luopua.

Tämän seuraavan askeleen onkin jo ottanut *Opsound*-niminen levy-yhtiö.⁹⁸ *Opsoundin* sivuilta löytyvä musiikki on kaikin puolin täysin vapaasti käytettävissä, ainoastaan ShareAlike-ehto varmistaa, että materiaali pysyy vapaana ja avoimena jatkossakin. Nyt voisin siis jo käyttää tätä musiikkia elokuvassa, jonka voisin myydä rahasta jollekin TV-kanavalle, kunhan antaisin samalla TV-kanavalle oikeuden käyttää koko elokuvaani samoilla ehdoilla ja kanava puolestaan sitoutuisi antamaan saman oikeuden edelleen kaikille katsojilleen. Tämä malli alkaa kaikin puolin muistuttaa GPL-lisenssiä, jolla Linuxia myydään. Hyvältä siis näyttää!

Opsound ei tilastojen valossa ole vielä päässyt yhtä pitkälle kuin Magnatune. Artisteja on suurin piirtein saman verran ja erinäköistä musiikkiakin löytyy noin neljänsadan tiedoston verran, mutta ensimmäistä levyä ollaan tätä kirjoitettaessa vasta julkaisemassa. *Opsoundin* valikoima tuntuu kallistuvan enemmän juuri konemusiikin puolelle, mutta mielenkiintoinen erikoisuus ovat pikemminkin ääniefekteiksi luokiteltavat nauhoitukset erilaisista ympäristöistä. Oma suosikkini on kappale nimeltä "Pildammsparken", nauhoitus ruotsalaiselta lintujärveltä.

Haluaisin lopettaa tämän musiikkikatsauksen ilahduttavaan uutiseen, jonka luin jokin aika sitten synnyinseutuni erästä paikallislehdessä. Uutinen kertoi paikallisesta säveltäjä-sanoittajasta liris Lumpeesta, joka oli julkaissut reilut sata omaa viisuaan sisältävän laulukirjan. Ilahduttavaa uutisessa oli, että laulukirjansa esipuheessa hän toteaa ykskantaan, että "kaikkia lauluja saa vapaasti kopioida ja laulaa, siksi ne on tähän painettu."

Tämän takiahan musiikkia on kautta aikain tehty. Jossain välissä me olemme unohtaneet tämän, ja musiikista on tullut levy-yhtiöiden omaisuutta, joilta sekä tekijöiden että kuulijoiden on pyydettävä lupaa omaan harrastukseensa. liriksen esimerkki oli siitä liikuttava, että tämä vanha rouva ei varmaan ole ikinä kuullutkaan Linuxista, eikä hänellä välttämättä ole mitään käsitystä siitäkään, mitä mp3-tiedostojen levittäminen internetissä tarkoittaa. Hänelle tämä oli vain täysin normaali tapa julkaista omat laulunsa. Siksihän ne on tehty, että niitä laulettaisiin.

liriksen avoimuus on ainakin pienessä mittakaavassa tuonut myös taloudellista menestystä, sillä minä tietenkin oitis tilasin yhden kappaleen hänen laulukirjaansa, vaikken kovin kova laulumies olekaan. Kuvittelisin, että tämän kaltaisille laulukirjoille löytyisi kysyntää muutenkin.

94Esimerkiksi rock-tähti Courtney Love syyttää levy-yhtiöitä itseään todellisesta piratismista artikkelissaan "Courtney Love does the math", <http://dir.salon.com/tech/feature/2000/06/14/love/index.html> .

95<http://www.moby.com/>

96<http://news.bbc.co.uk/2/hi/entertainment/3499534.stm>

97<http://magnatune.com/>

98<http://www.opsound.org/>

57. CREATIVE COMMONS JA AVOIMEN INFRASTRUKTUURIN LUONTI

Vaikka eri kulttuurien aloille onkin syntynyt jo jonkin verran Open Source -tyyppisiä teoksia, on varsinaisen Linuxin kokoluokkaa olevan vallankumouksen edessä vielä joitakin haasteita. Niitä voisi kutsua vaikka logistisiksi ongelmiksi. Open Source -ohjelmistojen puolella nämä ongelmat on jo ratkaistu, mutta kirjallisuus, musiikki ja muut taidelajit tulevat vaatimaan omat ratkaisunsa.

Yksi este Open Source -teosten vallankumouksen edessä on se, että nykyisissä tekijänoikeuslaeissa kaikki on aina lähtökohtaisesti kiellettyä. Tekijänoikeuden logiikka menee niin, että kaikilla teoksilla on välttämättä tekijä (teoksia ei synny tyhjästä) ja tekijällä on aina kaikki oikeudet omaan teokseensa, ellei hän nimenomaisesti niitä jollekin luovuta tai erityisesti salli esimerkiksi teoksen kopiointia. Aiemmin esimerkiksi USA:n tekijänoikeuslaissa teos oli lähtökohtaisesti vapaasti käytettävissä, ja vain erityisellä ©-merkillä varustetut teokset kuuluivat tekijänoikeuden alaisuuteen. Tekijän oli siis varta vasten ilmoitettava teoksensa suojatuksi, jos niin halusi, ja muuten oletettiin, että teosta sai vapaasti käyttää ja levittää.

Vaikka vapauden asialla ollaankin, niin periaatteessa nykyinen käytäntö on kuitenkin parempi. Se suojaa tekijöitä ahneilta kustantajilta ja levy-yhtiöiltä, joiden lakimiehet varmasti keksisivät teknisiä puutteita tekijöiden ©-merkkien käyttötavoissa, mikäli laki sen käyttöä edelleen vaatisi, ja siten päätyisivät omimaan teoksia omaan käyttöönsä maksamatta oikealle tekijälle edes nykyisiä vähäiä korvauksia.

Mutta asialla on toinenkin puoli. Nyt kaikkeen pitää saada lupa. Se, että löytää internetistä hyvän runon, artikkelin tai musiikkikappaleen, ei vielä oikeuta hyödyntämään sitä esimerkiksi omassa teoksessaan. Ensinnäkin pitäisi saada kyseiseltä tekijältä lupa. Tämä voi olla hyvinkin vaikeaa, sillä harvoinpa esimerkiksi mp3-tiedostojen yhteydestä löytyy artistin sähköpostiosoitetta tai puhelinnumeroa. Niinpä esimerkiksi joku laulu saattaa jäädä esittämättä konsertissa, koska sen tekijältä ei ole saatu siihen lupaa, koska tekijää ei kohtuullisin ponnistuksin voitu tavoittaa. Kuitenkin maailma on täynnä liris Lumpeen kaltaisia muusikkoja, jotka mielellään haluaisivat, että heidän laulujaan lauletaan, kopioidaan ja esitetään konserteissa.

Tämän takia olisi tärkeää, että säveltäjät, kirjailijat, artistit ja taiteilijat ympäri maailman oppisivat huomioimaan tämän esteen, samalla tavalla kuin liris oli huomionut sen oman laulukirjansa esipuheessa. Meidän tavallisten pulliaisten, jotka haluamme jakaa laulumme ja sanojamme toistemme kanssa, tulee siis oppia toimimaan tekijänoikeuslainsäädännön ehdoilla.

Tietokoneohjelmien maailmassa ohjelmoijat ovat tällaisen käytännön oppineet. Richard Stallmannin aikanaan kehittämä GPL-lisenssi on muodostunut standardiksi, ja se tai muu vastaava lisenssi mainitaan aina ohjelmien koodin alussa. Vain pari riviä, josta koodin lukija näkee heti, että hän saa käyttää tämän tiedoston koodia myös omissa ohjelmissaan. GPL-lisenssistä on muodostunut käsite, tavaramerkki, jonka Open Source -ohjelmoijat tunnustavat. Sen mainitseminen riittää kertomaan, että tämän koodi on vapaata, voimme kaikki yhdessä käyttää sitä. Jos tätä lyhyttä mainintaa ei olisi, niin mitään ei olisi tehtävissä. Silloin kopiointi ja koodin käyttäminen olisi automaattisesti kiellettyä.

Jotta Open Source -vallankumous voisi tosissaan levitä myös kirjallisuuden, musiikin ja muiden taiteiden puolelle, tulisi kaikkien listien kaltaisten viisujen tekijöiden ja muiden artistien oppia lisäämään teostensa alkuun lyhyt GPL-lisenssiä vastaava ilmoitus. Tällöin teoksesta näkisi heti, miten sitä saa käyttää, eikä lupaa tarvitsisi erikseen kysyä. Varsinkin ne, jotka julkaisevat kirjoituksiaan, valokuviaan tai piirroksiaan kotisivuillaan internetissä, voisivat olennaisesti vaikuttaa vapaasti käytettävissä olevan materiaalin määrään, jos tällainen tapa leviäisi. Pienistä puroistahan kaikki suuretkin putoukset saavat alkunsa. Samanlaisista pienistä puroista muodostuu GPL:n alainen kokonainen Linux-käyttäjärjestelmä.

GPL-lisenssiä on käytetty myös kirjojen tai musiikin tekijänoikeuksissa, mutta koska siinä puhutaan nimenomaan tietokoneista ja lähdekoodista, se ei ole kovinkaan luontevaa. Tilalle onkin kehitetty vastaavanlaisia lisenssejä, jotka sopivat paremmin tähän tarkoitukseen. Eniten julkisuutta, ja myös käyttäjiä, on saavuttanut *Creative Commons* -brändin⁹⁹ alainen lisenssikokoelma. Se on kerännyt taakseen joukon tämän alan raskaita nimiä, muun muassa Project Gutenbergin kaltaista vapaata kirjallisuutta julkaisevan *Eric Eldredin* sekä lakitieteen professori *Lawrence Lessigin*, jotka yhdessä vastustivat viimeisintä tekijänoikeuden suoja-ajan pidennystä USA:n korkeimmassa oikeudessa (häviten vain niukasti). *Creative Commons* on myös panostanut lisenssivaihtoehtojensa tunnetuksi tekemiseen, jotta taiteilijat oppisivat lisäämään kopioinnin sallivan (CC)-logon teostensa oheen, niinkuin ohjelmoijat lisäävät viittauksen GPL:ään omiin koodeihinsa.

Creative Commonsin sivuilta pöytälaatikkokirjailijat ja viisumaakarit voivat helposti valita sopivan lisenssin omalle teokselleen. Prosessi ei vaadi lakitieteen opintoja, vaan sopiva lisenssi löytyy parilla hiiren napsautuksella. Esimerkiksi tekijä voi sallia tai kieltää teoksensa kopioinnin ja levittämisen kaupalliseen tarkoitukseen, tai vaatia, että hänen nimensä mainitaan myös kaikkien muokattujen teosten tekijäluettelossa. *Creative Commonsin* iskulauseena onkin "Some Rights Reserved" (jotkut oikeudet pidätetään) vastakohtana tekijänoikeusmerkinnöissä yleensä nähtyyn "All Rights Reserved" (kaikki oikeudet pidätetään). Jos tekijä haluaa julkaista teoksensa täysin vapaasti käytettäväksi, eli tehdä siitä ns. Public Domain, voi teoksen yhteyteen lisätä logon, jossa lukee uljaasti "No Rights Reserved".

Näitä eri lisenssien logoja voi nykyään jo jonkin verran nähdäkin joillain www-sivuilla. *Creative Commonsista* on pikku hiljaa tulossa kirjallisuuden ja taiteiden GNU-liike, ja (CC)-logosta sen GPL-lisenssi.

Juridiikan lisäksi on vielä toinen logistinen ongelma vapaiden teosten tiellä, nimittäin vapaiden teosten löytäminen. Open Source -ohjelmistojen ympärille on vuosien saatossa kasvanut www-sivustoja, joilta löytyy kymmeniä tuhansia Open Source -ohjelmistoja. Esimerkiksi SourceForge isännöi lähes sataa tuhatta Open Source -projektia. Kirjallisuuden, musiikin ja muiden taiteiden alalla tällaiset kattavat hakemistot ovat vielä syntyneet. Creative Commons on kehittänyt oman *Get Content* hakukoneensa, joka etsii (CC)-lisensoituja teoksia ja aiemmin mainitut *Project Gutenberg* ja *The Assayer* ovat myös hyviä hakemistoja. Kehitystä kuitenkin tapahtuu koko ajan ja varmaan tulevaisuudessa vapaiden teosten löytäminen on vielä nykyistäkin helpompaa.

Yksi ongelma tällaisen hakemiston luonnissa on tiedostojen valtava koko. Ohjelmointikoodihan on vain ihmisten kirjoittamaa tekstiä ja vaikka kymmenien tuhansien ohjelmistojen kerääminen yhdelle www-palvelimelle onkin valtava urakka, se on kuitenkin mahdollista. Project Gutenberg on kerännyt kirjallisuutta omalle palvelimelleen vastaavalla tavalla, eiväthän kirjatkaan ole muuta kuin isoja tekstitiedostoja. Valokuvat ja muut graafiset taideteoksetkin vielä joten kuten ovat tallennettavissa tietokoneella. Sen sijaan musiikki ja varsinkin elokuvat vievätkin niin paljon kovalevytilaa, että niiden laajamittainen tallentaminen johonkin hakemistoon ja samoin niiden levittäminen internetissä ovat ihan aidosti teknisiä haasteita. Nykypäivän niin sanotuilla laajakaistayhteyksillä yksi mp3-muotoon pakattu 3 minuutin poppikappale tulee vielä perille minuuteissa, mutta Xvid-formaattiin pakkattua kokoillan elokuvaa joutuukin odottelemaan monta tuntia. Tuhansien elokuvien tarjoaminen samalta palvelimelta ladattavaksi on nykypäivän tekniikalla käytännössä mahdotonta.

Lisäksi pitäisi ottaa huomioon, että Open Source -filosofiaan kuuluu, ettei vastaanottajalle toimiteta pelkkää lopputuotetta – toimivaa tietokoneohjelmaa tai katseltavaa elokuvaa – vaan hänelle pitää tarjota alkuperäisen tekijän kanssa tasavertaiset mahdollisuudet muokata työtä edelleen. Tämä tarkoittaa sitä, että elokuvia ei voitaisi levittää vahvasti pakattuna, vaan ne pitäisi levittää vähintäänkin laadultaan DV-tasoisina. Lisähaastetta tuo se, että levytykseen pitäisi laittaa kaikki käytetty raaka-materiaali, jota tyypillisessä elokuvassa on kymmeniä, joskus satojakin tunteja!

Creative Commons on siis onnistunut tekemään lisensoinnin koukerot helpoiksi, mutta vielä puuttuvat tehokkaat työkalut vapaiden teosten etsimiseen ja levittämiseen internetissä. Perinteinen www-palvelinmalli ei ehkä koskaan tule taipumaan monen tunnin videomateriaalin levittämiseen internetissä. Sen sijaan viime vuosina yleistynyt peer-to-peer tekniikka – jota nykyään valitettavasti käytetään internetissä lähinnä musiikin ja elokuvien laittomaan kopioimiseen – saattaisi siihen kelvata. Kuka tekisi meille *Creative Commons Napsterin*?

Tässä kohden on myös hyvä muistaa, että 1980-luvulla, kun internet oli vielä lapsenkengissään, ei yksinkertaisten tietokoneohjelmienkaan jakaminen internetissä ollut helppoa. Monilla ei ollut verkkoyhteyttä ollenkaan, ja niilläkin joilla oli, se oli niin hidas, että normaalia sähköpostia pidemmät lähetykset vaativat jo odottelua. Tuohon aikaan myös GNU-ohjelmien pääasiallinen levityskanava olikin internetin sijaan nauhojen¹⁰⁰ lähettäminen postitse. Itse asiassa tämä ohjelmaa sisältävien nauhojen postimyynti oli siihen aikaan Stallmanin tärkeä tulonlähde. Internetin teknisistä rajoitteista huolimatta ei siis ole mitään syytä, miksei elokuviakin voisi jo ruveta tuottamaan ja levittämään Open Source -menetelmällä.

99<http://creativecommons.org/>

100Levykkeet olivat 80-luvulla vasta yleistymässä, niissä oli monia epäyhteensopivia standardeja ja lisäksi yhdelle levykkeelle ei edes mahtunut juuri mitään. CD-levyjä ei vielä tunnettu.

58. OPEN SOURCE - ELOKUVAILTA

Olemme jo käsitelleet kirjallisuutta ja musiikkia. Mitenkäs sitten elokuva? Tässä kohden olin ajatellut kirjoittaa, että Open Source -elokuva on vielä tekemättä. Mutta kuinkas ollakaan, sellainenkin jo löytyy. Historian ensimmäinen ja ilmeisesti tällä hetkellä ainoa Creative Commons -lisenssillä vapaasti saatavissa oleva kokoillan elokuva on *Brian Flemingin* ohjaama "*Nothing So Strange*".¹⁰¹

Nothing So Strange on niin sanottu indie-elokuva (sanasta "independent", itsenäinäinen) eli se on tuotettu pienellä budjetilla ilman suurten Hollywood-studioiden tukea. Elokuva selvittelee dokumenttityylisesti maailman rikkaimman miehen, Bill Gatesin salamurhan selvittelyä. Lehdistöiedotteen mukaan elokuvan juoni muistuttaa Oliver Stonen JFK elokuvaa, jossa vastaavasti selvitetään presidentti Kennedyn salamurhaa, todisteiden katoamista, viranomaisten vaikeumista ja muita koukeroita.

Mikä yhteensattuma, että ensimmäinen Open Source -elokuva käsittelee juuri Bill Gatesin, Microsoft-monopolin perustajan ja Open Source -ohjelmistojen vannoutuneen vastustajan murhaa. Ja kyseessä nimenomaan on yhteensattuma, sillä alunperin elokuva ei ollut Open Source, vaan se tuotettiin ihan perinteisin elokuva-alan suljetuin menetelmin. Lieneekö Flemming Open Sourcesta silloin vielä kuullutkaan?

Kävi kuitenkin niin, että elokuvateatterit eivät elokuvasta kiinnostuneet eivätkä suostuneet ottamaan sitä ohjelmistoihinsa. Ilmeisesti elossa olevan todellisen henkilön murhaaminen ei heidän mielestään ollut näyttämisen arvoinen elokuvajuoni tai sitten elokuva oli pelkästään huono. En tiedä, en minäkään ole sitä katsonut.

Pitääkseen elokuvan hengissä Flemming päätti julkaista siihen käytetyt raakamateriaalit Open Sourcena. Elokuvan juoni huomioonottaen voisi kuvitella, että jotkut Linux-fanit olisivat kiinnostuneet ostamaan joko varsinaisen elokuvan DVD:n tai parhaimmalla tapauksella innostuneet kehittämään raakamateriaalista täysin omia versioita, joissa juoni kehittyi aivan eri lailla kuin alkuperäisessä versiossa.

Vaikka Brian onnistuikin saamaan elokuvalleen tällä tavoin jonkin verran julkisuutta, näyttäisi kuitenkin siltä, että mitään todellista Open Source -liikettä ei elokuvan taakse ole syntynyt. Kukaan ei ole tehnyt elokuvasta vaihtoehtoisia versioita, eikä raakamateriaalia sisältävää DVD:tä ilmeisesti ole tilattu ainuttakaan kappaletta.

Vaikka yksi Open Source -elokuva siis tosiaan löytyykin, voin siis kuitenkin kirjoittaa, että ensimmäinen Open Source -elokuvatuotanto on edelleen tekemättä. Flemmingin arvokasta lahjoitusta Creative Commons -lisensoitujen teosten joukkoon mitenkään vähättelemättä, on tietenkin selvää, että Open Source -ideologian suurin vahvuus ei ole lopullisen tuotteen ilmaisuudessa, vaan koko tuotantoprosessin erilaisuudessa. Flemming onkin ilmaissut kiinnostuksensa kuvata joskus joku toinen elokuva alusta asti Open Sourcena. Voimmekin jäädä jännittämään, kuka ensimmäisenä onnistuu viemään loppuun tällaisen historiallisen tuotannon, ensimmäisen todellisen Open Source -elokuvan?

Sillä välin voikin olla mielenkiintoista pohtia, minkälainen ensimmäisestä Open Source -elokuvesta tulisi? Millä lailla se eroaisi siitä Hollywoodin bulkkituotannosta, jota televisiomme suoltavat ilta toisensa perään?

Jos aloitetaan ihan perusteista, niin elokuvan käsikirjoituksen pitäisi tietenkin olla itsessään Open Source. Se voisi perustua johonkin Project Gutenbergista löytyvään klassikkoon, mutta todennäköisesti tarina kirjoitettaisiin varta vasten elokuvaa varten. Jo käsikirjoituksen tekemiseen osallistuisi monia ihmisiä, jotka tekisivät yhteistyötä internetin yli. Ehkäpä sen teossa käytettäisiin wiki-tekniikkaa?

Itse kuvausten järjestämisessä Open Source elokuvalla on vaikeuksia, mutta myös mielenkiintoisia mahdollisuuksia. Open Source -menetelmälle tyypillistä on, että kuka tahansa voi milloin tahansa ja missä tahansa osallistua teoksen minkä tahansa osan luomiseen tai muokkaamiseen. Elokuvasa näyttelijät ovat kuitenkin tässä suhteessa ongelma. Normaalistihan elokuvissa ainakin tärkeimpien hahmojen näyttelijät ovat alusta loppuun samat. Olisi aika hämmentävää, jos keskellä elokuvaa Richard Geren hahmoa näyttelisivin yhtäkkiä Danny DeVito ja loppukohtauksessa näyttelijäksi vielä vaihtuisi yllättäen Vesku Loiri!

Ohjelmoijien kieltä käyttäkseni, näyttelijän henkilö ei ole modulaarinen. Sitä ei voida jakaa useamman osallistujan tehtäväksi. Mutta tulevaisuudessa ehkä tämäkin on toisin. Jo nyt esimerkiksi Matrix-elokuvatrilogian viimeisissä osissa oli aika pitkiäkin taistelukohtauksia tuotettu täysin tietokoneella animoimalla. Kun tietokoneet vielä kehittyvät, on lähitulevaisuudessa mahdollista skannata näyttelijästä tarkka kolmiulotteinen kokovartalomalli, jonka jälkeenokuva voidaan tehdä täysin tietokoneella, sillä välin kun näyttelijät itse löhöilevät porealtaassa. Kun tämä tekniikka saadaan laajempaan käyttöön, voidaan elokuvan roolihahmojen liikkuminen ja puhe kuvata missä vaan, ja animoinnin voi tehdä kuka tahansa, ihan niinkuin Open Source -menetelmässä tehdään. Jo nyt tämä olisi mahdollista, jos tyydyttäisiin tekemään Toy Storyn, Antzin tai Hulkin kaltainen animaatioelokuva. Sellainenhan voitaisiin tehdä vaikka käyttäen Blender-ohjelmaa, josta kerroin kolmannessa luvussa.

Näyttelijöiden tai erikoistehosteiden animointi on joka tapauksessa mielenkiintoinen Open Source -lähestymistavan kannalta. Elokuvien animoidut erikoistehosteet nimittäin vaativat aivan uskomattoman paljon laskentatehoa. DreamWorks ja muut Hollywoodin studiot ylläpitävät tätä tarkoitusta varten niinsanottuja animointifarmeja, joissa tuhansia huippunopeita tietokoneita on kytketty yhteen verkkoon ja näiden tietokoneiden ainoa tehtävä on animoitujen kohtausten renderointi, eli valmiiden kuvien laskeminen animaation tekijän antamista liikeradoista ja muista käskyistä. Kaikesta tästäkin laskentatehosta huolimatta renderointi on kuitenkin edelleen hidasta puuhaa. Yhden kohtauksenkin valmistumista joutuu helpostikin odottelemaan yön yli, tai yksinkertaisissakin tapauksissa ainakin kahvitaun verran.

Animointi soveltuu erittäin hyvin Open Source -metodille. Olisi täysin mahdollista kerätä yhteen tuhansia vapaaehtoisia internetin käyttäjiä, jotka suostuisivat lainaamaan tietokonettaan seuraavan elokuvahtin valmistukseen. Tällaista *hajautettua laskentaa* tehdään paljon jo nyt. Suosittuja laskennan kohteita ovat muun muassa sähköisten viestien etsiminen avaruuden taustakohinasta,¹⁰² sopivien proteiinien löytäminen syöpälääkkeen keksimiseksi¹⁰³ tai ihan vaan matemaattisten alkulukujen estiminen.¹⁰⁴ Hajautetun laskennan idea perustuu siihen, että suurimman osan ajasta tietokoneemme ovat päällä, mutta niitä ei käytetä mihinkään. Tietokoneen käyttäjä on milloin kahvilla, milloin lounaalla tai puhelimessa. Silloinkin kun luemme www-sivuja tai kirjoitamme sähköpostia, suurin osa tietokoneemme potentiaalisesta laskentatehosta jää kuitenkin käyttämättä. Näiden projektien www-sivuilta tietokoneen käyttäjä voi asentaa itselleen ohjelman, joka hyödyntää tätä käyttämättä jäänyttä laskentatehoa ja lähettää sitten laskennan tuloksen takaisin palvelimelle, joka kerää kaikki tulokset yhteiseen tietokantaan.

Mielenkiintoinen tulos hajautetun laskennan alueelta on vuodelta 1999, jolloin Distributed.net¹⁰⁵ osallistui RSA-yhtiön järjestämään kilpailuun DES-algoritmiin perustuvan salausavaimen murtamiseksi. Distributed.net osallistui kilpailuun lähes sadantuhannen koneen muodostamalla verkostolla, jotka mursivat salauksen alle vuorokaudessa. Mielenkiintoinen tilastotieto kilpailusta oli, että Distributed.net:in vapaaehtoisten muodostaman verkoston laskentateho ylitti yli kaksinkertaisesti Electronic Frontier Foundationin¹⁰⁶ tarkoitusta varten rakentaman 250 000 dollaria maksaneen supertietokoneen. DES-kilpailu paljasti, millaista voimaa yhteisöllisyydessä on ihan raa'asta laskentavoimastakin puhuttaessa.¹⁰⁷

Jos jonkun elokuvan erikoistehosteet – oli elokuva sitten Open Source tai ihan perinteistä Hollywood-tuotantoa – joskus haluttaisiin tuottaa vastaavalla lailla hajautetusti, on täysin varmaa, että vapaaehtoisia löytyisi miljoonittain. Kukapa ei haluaisi ylpeänä kantaa T-paitaa, jossa lukisi esimerkiksi: "Minun tietokonettani käytettiin Matrixin loppukohtauksen erikoistehosteiden valmistuksessa."

Mutta näyttelijäongelmaan on kuitenkin toinenkin ratkaisu. Mitä jos elokuvassa ei olisi päärooleja ollenkaan? Useimmissa elokuvissahan on nykyään hyvin rajattu määrä päärooleja, eikä sivuroolejakaan yleensä kovin runsaasti. Tähän on ihan selvä syy, näyttelijät ovat kallista työvoimaa. Mutta Open Source elokuvassa tällaista ongelmaa ei ole. Vapaaehtoiset ympäri maailman haluaisivat varmaan osallistua elokuvan tekoon. Miksemme siis tekisi elokuvaa, jossa olisi kymmeniä tai jopa satoja pieniä päärooleja? Statisteja voisi olla jopa tuhansia, niin paljon kuin heitä vain saataisiin paikalle. Tällaisia tuhansien näyttelijöiden spehtaakkelejahan ei Hollywoodissa ole tuotettu enää kymmeneen vuosiin, koska se on liian kallista. Nykyään kaikki massakohtaukset tehdään tietokoneella kloonamalla parinkymmenen seisoskelijan joukko vaikka tuhannen hengen ihmismassaksi. Mutta Open Source -elokuvaamme tällaiset ongelmat eivät vaivaa. Mitä enemmän tekijöitä, sen hausempaa! Ihan niinkuin Linuxissakin.

Haluamme siis käsikirjoituksen, jollaista yksikään Hollywoodin studio ei suostuisi kuvaamaan. Käsikirjoituksessa olisi kymmeniä henkilöhahmoja ja tuhansia statistteja. Tapahtumapaikat olisivat ympäri maailmaa mitä eksoottisimmissa paikoissa ja niitä olisi niin monta, että yhdenkään elokuvayhtiön matkabudjetti ei pystyisi sellaista toteuttamaan.

Kun elokuva olisi satojen näyttelijöiden ja kuvaajien avulla saatu kuvattua, tulisi editoinnin ja jälkikäsittelyn vuoro. Kuvattua raakamateriaalia jaettaisiin internetissä ja tarvittaessa tekijät lähettäisivät toisilleen materiaalia nauhoina tai DVD-levyinä postitse. Eri puolilla maailmaa editoitaisiin kuvamateriaalista hyvinkin erilaisia versioita, joista lopullinen elokuva vähitellen muodostuisi. Elokuvan musiikiksi valittaisiin tietenkin Creative Commons -lisenssin alaisia kappaleita, ja ehkä ääniefektejä käytettäisiin Opsound-arkistosta.

Lopputulos olisi varsinainen mestariteos. Se olisi monumentti satojen kirjoittajien, säveltäjien, muusikoiden, kuvaajien, näyttelijöiden, ohjaajien, ompelijoiden, rakentajien, animoijien ja monien muiden tekijöiden yhteistyöstä.

Mutta mitä elokuva sitten käsittelisi? Millainen juoni sopisi tällaiseen elokuvaan, jossa rooleja ja tapahtumapaikkoja on satoja? Ironista kyllä, yksi hyvä ehdotus elokuvan juoneksi voisi olla Linuxin ja vapaiden ohjelmistojen kehitys. Vapaita ohjelmistojahan kehittävät mitä erilaisimmat ihmiset eri puolilla maailmaa, kaikkina vuorokaudenaikoina. Mikä olisikaan hienompaa kuin tehdä näitä ihmisiä ja heidän työtään käsittelevä elokuva Open Source -menetelmällä? Brian Flemming, jos luet tätä, ota minuun yhteyttä!

101<http://www.nothingsostrange.com/>

102<http://www.seti.org/science/setiathome.html>

103<http://www.chem.ox.ac.uk/curecancer.html>

104<http://www.mersenne.org/> (Osoitteesta <http://www.aspenleaf.com/distributed/> löydät myös kymmeniä muita vastaavanlaisia projekteja.)

105<http://www.distributed.net/des/>

106<http://www.eff.org/>

107Lisäksi kilpailun tulos osoitti – ja tämä oli järjestäjien varsinainen päämäärä – kuinka haavoittuva oli siihen aikaan yleisesti käytetty, ja USA:n hallinnon suosittama DES-salaus. RSA:n järjestämät kilpailut lopulta pakottivat USA:n sallimaan DES:iä vahvempien salausmenetelmien viennin.

Toinen mielenkiintoinen tarina liittyen vahvojen salausten maastavientikieltoon on Philip Zimmermanin kehittämä PGP-ohjelma. Sen sisältämiä vahvoja salausmenetelmiä ei myöskään olisi saman lain mukaan saanut viedä USA:n ulkopuolelle. Philip ystävineen kuitenkin julkaisi ohjelman lähdekoodit tulostettuna paperille – 6000 sivua joista tuli yhteensä 12 nidettä. Nämä kirjat vietiin lentokoneella Eurooppaan, jossa ne skannattiin takaisin tietokoneelle ja lähdekoodista käännettiin taas toimiva ohjelma. Ajatuksena oli, että kirjojen maastavientiä ei voitu kieltää, koska se olisi rikkonut USA:n perustuslain sananvapausartikkelia. (<http://www.pgpi.org/pgpi/project/scanning/>)

59. OPEN CORES – VAPAATA RAUTAA

Open Source -ideologian jatkaminen kirjallisuuteen ja muille luoville aloille on aika ilmeistä. Mutta toinen mielenkiintoinen kysymys on, jos tietokoneohjelmia voidaan tuottaa ja levittää avoimella mallilla, niin mitenäs sitten itse tietokoneet? Jos pystyn nyt käyttämään kauttaaltaan Linuxia ja muita Open Source -ohjelmia tietokoneellani, niin milloin itse tietokoneesta tulee yhtä avoin? Tavallaanhan PC-tietokoneet ovat avoin arkkitehtuuri siinä mielessä, että tietyt standardit täyttävistä komponenteista voi vapaasti valiten koota oman tietokoneensa. Mutta nämä komponentit ja niiden tuotantoprosessit eivät itsessään ole mitenkään avoimia.

Äkkiseltään voisi ajatella, että Open Source -malli ei toimi tietokoneiden komponenteissa, eli niin sanotusti "raudassa". Ohjelmien lähdekoodilla, kirjan tekstillä, elokuvilla ja äänitiedostoilla on kaikille yhteistä se, että digitaaliseen muotoon tallennettua informaatiota voidaan kopioida rajattomasti ja käytännössä ilmaiseksi. Tietokoneiden komponentit taas ovat ihan oikeaa fyysisen maailman materiaa, joille pätee perinteiset kysynnän ja tarjonnan lait. Empiiriset havaintomme maailmasta sanovat, että yksi tietokone on yksi tietokone, eikä se siitä mihinkään leviä.

Mutta empiirisistä kokemuksistamme huolimatta, on olemassa projekteja Open Source -tyyppisten tietokonekomponenttien valmistamiseksi. Ja ainakin tunnetuin näistä, *Opencores.org*¹⁰⁸ näyttäisi menestyvän hyvin ja saaneen jo joitakin tuotteita sarjavalmistukseen ja markkinoille! Mistä siis on kysymys?

Minähän en elektroniikasta ymmärrä sen enempää kuin sinäkään, mutta ennen kuin jatkamme olisi meidän kuitenkin tässä vaiheessa hyvä paneutua integroitujen piirien valmistukseen.

Tietokoneen komponenttejahan ovat esimerkiksi mikroprosessori, erilaiset ram-muistit, modeemi tai verkkokortti internet-yhteyttä varten, yksi kortti johon näyttö kytketään ja toinen joka tuottaa ääntä kaiuttimiin. Kaikki nämä erilliset komponentit ovat kiinni emolevyssä, joka itsessään on jo elektroniikan mestarinäyte, sisältäen useampia yksittäisiä mikrosiruja sekin. Kuluttajanäkemyksen mukaanhan nämä komponentit tulevat ulos jostakin tehtaasta, pakattuna erilaisiin muovijä ja pahvipaketteihin ja sitten ne kootaan yhteen, lisätään näppäimistö, hiiri, kaiuttimet ja näyttö ja tietokone on valmis.

Mutta nyt paljastuukin, että näiden erilaisten mikrosirujen valmistus muistuttaakin itse asiassa kovasti juuri ohjelmointia. Niiden valmistusta ei tosin yleensä kutsuta ohjelmoinniksi, vaan suunnitteluksi, eli designiksi. Mikrosirujen suunnitteluun on jopa ihan omat ohjelmointikielensäkin. Yleisimmät kielet ovat nimeltään VHDL¹⁰⁹ ja Verilog. Nämä VHDL- tai Verilog-kielellä kirjoitetut integroitujen piirien kuvaukset ovatkin sitten tosi rankkaa kamaa. Niitä lukeakseen pitäisi ymmärtää vähän sekä elektroniikkaa että ohjelmointia, ja tällainen tavallinen pulliainen kuin minä saa niistä yleensä vaan päänsäryn. Mutta ne jotka tätä kieltä osaavat, valmistavat siis meidän joka päivä käyttämiämme mikrosiruja. Enkä nyt puhu pelkästään tietokoneista. Mikrosirujahan on niin jääkaapeissamme kuin autoissakin, sellutehtaissa ja hisseissä.

Näistä VHDL-kieltä sisältävistä tiedostoista saadaan suhteellisen automaattisilla prosesseilla tehtyä ensin ns. porttitason kuvaus ja näistä loogisista porteista edelleen piirin fyysinen malli, layout. Tästä valmistetaan lopulta maski, eli eräänlainen muotti, jonka avulla sitten piikiekolle valetaan tai säteilytetään pieniä transistoreita ja johtimia.

Näin siis mikrosirutetaan valmistuslinja suhteellisen automaattisesti muokkaa puolijohdemateriaalista ohjeiden mukaisen valmiin sirun, joka pakataan ja lähetetään tietokonehtaalle, autotehtaalle tai jääkaappitehtaalle. Tietenkin tällaisen valmistuslinjan rakentaminen ja ylläpito on ihan oma juttunsa sekkin. Muun muassa nämä tehtaot ovat leikkaussaliakin puhtaampia, täysin pölyttömiä. Samaten valmistusprosessiin liittyy paljon muutakin magiaa, mihin emme tässä nyt kovin syvästi paneudu. Aikaavievä työvaihe piirin suunnittelussa on muun muassa ns. ajastus. Koska olemme nyt tekemisessä fysiikan maailman kanssa, niin täytyy ottaa huomioon, että sähkösignaali liikkuu piirissä ainoastaan valon nopeudella, ei yhtään nopeammin. Siksi loogisesti oikein suunnitellusta VHDL-tiedostosta ei vielä synny oikein toimivaa mikrosirua, ellei ensin varmisteta, että sähkösignaali oikeasti ehtii transistorista A transistoriin B tarpeeksi nopeasti ja toisaalta ei liian nopeasti. Mutta tämä yleistajuisuuteen pyrkivä tarinamme ei kuitenkaan sukella tämän syvemmälle, vaan meille riittää uskoa, että tästä on kyse: VHDL-tiedoston ohjeista valmistetaan muotti, jonka avulla puolijohdemateriaalista saadaan valmis mikrosiru.

Eli on osoittautunut, että suurin työ yksittäisen mikrosirun valmistuksessa onkin itse asiassa sen suunnittelu. Tarvitaan ohjelmoijia, jotka kirjoittavat koodia, eli sitä VHDL-kieltä. Sitten tarvitaan testaajia, jotka testaavat, että siru toimii suunnitellusti. Testaaminenkin tehdään luonnollisesti ennen kuin yhtäkään sirua on oikeasti valmistettu, muodostamalla VHDL-tiedostosta malli tietokoneohjelman sisään, jonka avulla sirun toimintaa simuloidaan. Testaaminen onkin mikrosirujen valmistuksessa erittäin tärkeää ja sen osuus suunnittelutyöstä on huomattavasti suurempi kuin esimerkiksi tietokoneohjelmien valmistuksessa. Tämähän johtuu tietenkin siitä, että mikrosiruja ei ole kovin helppoa korjailta jälkeensä, joten niiden pitää olla käytännössä virheettömiä, ennen kuin valmistuslinja viritetään tuottamaan niitä tuhatmäärin.

Tämä yksinkertaistettu selostukseni ei sentään päde ihan kaikkiin mikrosiruihin. Esimerkiksi uusiin tietokoneisiin laitettavien nopeiden Pentium, Opteron tai Athlon prosessorien valmistaminen on oikeasti melkoista salatiedettä, jossa jatkuvasti uhmataan fysiikan lakeja. Näidenkin prosessorien suurin työsaika on suunnittelussa, eikä niiden valmistus siltä osin periaatteessa eroa muista mikrosiruista. Kuitenkin näiden huippuprosessorien viimeiset tehot saadaan irti tuotantolinjaan varta vasten tehtävillä virityksillä ja muulla käsityöllä. Menemättä taaskaan yksityiskohtiin, niin haasteena on saada mahdollisimman paljon transistoreja ja johtimia sijoiteltua puolijohteelle mahdollisimman tiheään. Ja sitten kun tuotantolinja saadaan tekemään tarpeeksi pieniä transistoreja, niin seuraava ongelma onkin, että miten ne sähkösignaalit kuljettavat elektronit saadaan pysymään oikeassa johtimessa. Jos johtimet ovat liian pieniä ja liian tiheässä, sähkösignaalit pomppivat yhdestä johtimesta toiseen, kaikki menee sekaisin ja prosessori on käyttökelvoton.

Näin siis uusimpien ja hienoimpien mikroprosessorien kohdalla. Sen sijaan monien muiden prosessorien valmistus on – näin minulle on kerrottu – oikeasti lähes niin helppoa kuin yllä kuvasin, silloin kun ei tarvitse lyödä mitään nopeusennätyksiä. Viedään valmis VHDL-tiedosto mikrosiruja valmistavalle alihankkijalle ja hetken kuluttua toisesta päästä saadaan valmiita prosessoreita.

Opencores saikin runsaasti julkisuutta,¹¹⁰ kun sen Open Source -koodista joulukuussa 2003 tosiaan valmistettiin ihan oikea integroitu piiri. Nyttemmin samaa sirua ovat valmistaneet useammatkin valmistajat, mutta ensimmäisen historiallisen Open Source -sirun valmisti Flextronics Semiconductor, eli sama yhtiö joka valmistaa alihankintana siruja niin kännyköitä valmistavalle Nokialle, kuin suurimman osan internetin reitittimistä valmistaneelle Ciscolle. Tämä viimeistään todisti, että Open Source voi toimia myös tietokoneen komponenteissa, eli ns. rautatasolla.

Ei niin, etteikö Opencoresin siruja olisi toteutettu jo paljon aikaisemmin niin kutsuttuina FPGA-siruinä.¹¹¹ FPGA-siruja käytetään elektroniikassa laajalti niiden joustavuuden takia, mutta ne ovat myös Opencoresin ja muiden integroitujen piirejä opetusmielessä tai harrastuksenaan suunnittelevien suosiossa. Kyseessä on eräänlainen yleiskäyttöinen mikrosiru. Se tulee tehtaalta tyhjänä ja käyttäjä voi itse "ladata" siihen haluamansa toiminnot, jotka hän siis on ensin kirjoittanut vaikkapa VHDL-kielellä. Lyhenteessä esiintyvä P-kirjain tuleekin juuri sanasta "programmable", ohjelmoitava. Prototyypin teossa FPGA-sirut ovat arvatenkin korvaamattomia, sillä ne ovat yleensä uudelleen ohjelmoitavia. Eli sirun testauksen jälkeen omaa VHDL-koodiaan voi muokata ja saman sirun voi sitten käyttää uudelleen uuden version kokeilemiseen.

FPGA-piiriin "lataaminen", eli ammattikielellä ohjelmoiminen, voidaan tehdä tietokoneeseen kytkettävällä erityisellä laitteella, johon tyhjä FPGA-piiri asetetaan. Jotkut FPGA-piirit kuitenkin unohtavat sisältönsä aina kun tietokone sammutetaan ja nämä piirit ladataan lennossa uudelleen joka kerta tietokoneen käynnistyessä. Periaatteessa FPGA-piirejä voisi verrata polttaviin CD-asemiin. Alunperinhän me kaikki elimme siinä luulossa, että CD-levyjä voi valmistaa vain CD-levytehtaissa. Mutta sitten joku keksi polttavat CD-asetat, jonka jälkeen CD-levyjä pystyikin valmistamaan millä tahansa kotitietokoneella. Tietenkään kotipolttoiset levyt eivät ole yhtä laadukkaita ja kestäviä kuin tehtaassa prässätyt, eikä kotona muutenkaan kannata suuria määriä valmistaa. Mutta polttavien CD-asettien ansiosta kuka tahansa musiikin harrastaja voi halutessaan julkaista oman CD-levyn. Vastaavasti sähköinsinööri voi FPGA-sirulla tehdä ihka oman prosessorinsa.

Ja vaikka Pentiumin ja Opteronin tehoihin ei näillä yleiskäyttöisillä siruilla päästäkään, niin kuulemma moderneilla FPGA-sirulla on jo mahdollista valmistaa teholtaan muinaista 486dx:ää vastaava mikroprosessori. Suomeksi sanottuna tämä tarkoittaa sitä, että reipas sähköinsinööri voisi kotikonstein rakentaa itselleen 90-luvun alun tasoa olevan tietokoneen, joka koostuisi täysin Open Source -komponenteista! FPGA-sirun lataamiseen tarvittavat laitteet eivät edes ole kalliita. Kiinnostuneille tiedoksi, että erilaisia Development-kittejä löytyy muun muassa Altera ja Xilinx nimisiltä yhtiöiltä, ja niiden hinnat ovat noin sata tai sitten tuhansia dollareja. Sitä paitsi pitää muistaa, että oikein reipas sähköinsinöörinhän voi rakentaa myös lataamisessa käytetyn piirilevyn itse, jolloin se ei maksa juuri mitään.

108<http://www.opencores.org>

109Kukapa näistä kirjainyhdistelmistä ottaa selvää, mutta VHDL on lyhenne sanoista "*VHSIC Hardware Description Language*". VHSIC taas on lyhenne sanoista "*Very High Speed Integrated Circuits*".

110"Open Source päätty vihdoin piille", otsikoi esimerkiksi nörttien suosima verkkosivusto Slashdot 8. joulukuuta 2003:
<http://slashdot.org/articles/03/12/08/2326236.shtml>
Itse sirusta lisää osoitteessa:
<http://www.opencores.org/projects.cgi/web/or1k/silicon>

111Tässä kohden on hyvä selventää käytettyä terminologiaa. Opencores.org projektin osallistajat tuntuvat käyttävän termiä FPGA laajasti, kuvaamaan kaikenlaisia ohjelmitavia mikrosiruja. Tämän takia sitä käytetään tässä merkityksessään myös tässä kappaleessa. Todettakoon kuitenkin tasapuolisuuden vuoksi, että monien tämän alan insinöörien mielestä FPGA:t ovat vain yksi ohjelmitavien sirujen alatyyppejä. Kiinnostuneille lisätietoja löytyy taas kerran Wikipediasta:
http://en.wikipedia.org/wiki/Programmable_logic_device

60. OPEN COLLECTION –

MUOTIA, BRÄNDEJÄ JA KANGASTA

Opencores-projektin saavutukset avaavat Open Source -metodille aivan uusia mahdollisuuksia. Projekti on jo nyt osoittanut, että Open Sourcea voi soveltaa myös muualla kuin virtuaalimaailmassa. Kun tietokoneohjelmat, kirjallisuus, musiikki ja elokuvat on jo katettu, niin Opencores tuo Open Source-ajattelun fyysisten tuotteiden valmistukseen. Voiko Open Sourcea siis soveltaa ihan mihin tahansa?

Jos ajatellaan niitä tavaroita ja tuotteita joita kaupat myyvät ja joiden keskellä me elämme, niin loppujen lopuksi monissa tuotteissa on hyvin vähän kysymys materian, eli raaka-aineen omistamisesta ja työstämisestä. Sen sijaan moneen tuotteeseen liittyy yllättävän paljon ominaisuuksia, jotka soveltuvat oikein hyvin Open Source -menetelmälle, koska ne ovat *aineettomia*, eli *immateriaalisia*. Tietokoneohjelmat ja kirjat ovat käytännössä täysin aineettomia, siksi ajatus niiden valmistamisesta ja levittämisestä Open Sourcena on niin ilmeinen. Mutta myös muiden tuotteiden valmistamiseen liittyy paljon aineetonta työtä ja siksi lähes mikä tahansa tuote voi ainakin osittain hyötyä Open Sourcesta. Tämän kirjan alussa käytiin öljyä esimerkkinä raaka-aineesta, rajallisesta luonnonvarasta, jonka tuotanto ja myynti välttämättä noudattaisi kysynnän ja tarjonnan lakeja, eli pihtaamisen logiikkaa. Rob McEwen kuitenkin osoitti meille, että jopa kullankaivuussa voi hyödyntää Open Source -ajattelua. Jos sitä voi hyödyntää kullankaivuussa ja mikrosirujen tuotannossa, niin varmaankin sitten ihan missä vaan muuallakin – jopa öljynporauksessa.

Otetaan vaikka esimerkiksi vaatteet. Voisiko vaatteita valmistaa Open Source -menetelmällä? Miksi ei, onhan ohjelmointi alunperinkin saanut alkunsa juuri kutomakoneesta.¹¹² On siis aika taas kerran yhdistää ohjelmointi ja vaatteiden tuotanto – tällä kerralla tosin olisi muotimaailman vuoro ottaa oppia hakkereilta.

Vaatteet ovat hyvä esimerkki hyvin tavallisesta tuotteesta, jota me kaikki käytämme päivittäin. Kuitenkaan vaatteet eivät todellakaan ole mikään arkipäiväinen tai vähäpätöinen asia. Kysy vaikka ensimmäiseltä vastaantulevalta teiniltä, jos et minua usko.

Vaatteissa ei ole kysymys vain vaatteesta. Ensinnäkin on kysymys *brändistä!* Brändin takia joku ostaa enemmän Niken lenkkarit, eikä halvempia brändittömiä. Brändin takia Nike maksaa Tiger Woodsille satoja miljoonia dollareita mainosmiehenään toimimisesta, kun taas lenkkarit valmistavalle aasialaiselle ompelijälle maksetaan pari dollaria päivässä.

Tietenkin vaatteissa on myös – aivan kuten mikrosiruissakin – kysymys *designista*. Kun puhutaan vaatteen tekemisestä, ei nykyään enää välttämättä tarkoiteta itse ompelua, vaan nimenomaan uusien vaatteiden leikkausten, kankaiden ja värien suunnittelua – *designia*. Ja jos puhutaan lenkkareista – vaikkapa Niken – niin niihin liittyy vielä meidän insinöörienkin tekemää *designia*. Nykyäänhän lenkkarit nimittäin ovat varsinaisia insinööri suunnittelun mestarinäytteitä. On ilmasäiliöllisiä kengänpohjia ja jalkahien tuuletusreikiä, niin että ihan tavallisesta kävelystäkin on nykyään tullut välineurheilua. Kun *design* on valmis, niin sen varsinaisen vaatteen valmistus onkin sitten ihan sivuseikka, jossa lähinnä kiinnostaa se, miten sen voisi teettää mahdollisimman halvalla. Ja tämän takia sen vaatteen sitten ompeleekin joku aasialainen parin dollarin päiväpalkalla.

Mutta tarina ei vielä lopu tähän. Kun aasialaiset ovat kiltisti ommelleet ison kasan vaatteita, niin ne pitää saada jaeltua eurooppalaisiin ja amerikkalaisiin kauppoihin ja tällöin on tietenkin kyse *logistiikasta*. Ja logistiikassakin on lähinnä kyse informaatiotekniikasta, tietenkin kuorma-autojen, laivojen ja lentokoneiden lisäksi.

Brändit, *design*, logistiikka, projektinhallinta ja niin edelleen. Nämä kaikki ovat täysin aineettomia asioita, eli Open Sourcen vahvaa aluetta. Mutta miten perustetaan Open Source -vaatebrändi? Tämän ymmärtämiseksi meidän on otettava mallia Richard Stallmanin GPL-lisenssistä ja sen periaatteista. Tarvitsemme lisäksi jakamista helpottavia työkaluja, sellaisia kuin SourceForge tai BitKeeper, joita Linux-ohjelmoijat käyttävät omassa työssään. Ja sitten tarvitaan tietenkin suunnittelijoita, ompelijoita ja kauppiaita – kaikenlaisia tekijöitä, ihan niinkuin Linuxissakin. Ja koska kyse on brändistä, tarvitsemme myös nimen. Kutsukaamme vaatebrändiämme nimellä *Open Collection*.

Open Collection on vaatebrändi, johon kaikki saavat osallistua. Lapset ja mummot, vaatealan tai taiteen opiskelijat aina maailman tunnetuimpiin muotialan suunnittelijoihin asti. Kaikki halukkaat saavat tehdä ompelukaavoja, kutomisohjeita, tekstuureja, kuvia, logoja ja kaikkea mitä vaatteiden suunnittelussa nyt yleensäkin tehdään. Nämä kaavat tallennetaan internetissä olevaan *www*-sivustoon, jossa ne ovat muiden käytettävissä. Tämä kaikki tietenkin GPL-tyyppisin ehdoin, eli löytämänsä kaavaa tai kuviota saa vapaasti käyttää, kunhan vastavuoroisesti lahjoittaa muiden käyttöön siihen tekemänsä muutokset.

Aivan niinkuin toimiva tietokonekin koostuu useammasta pienestä ohjelmasta, myös valmis vaate koostuu useammasta erillisestä *designista*. Joku ehkä on kehittänyt tietyn kangastyypin, jollaista vaatteeseen käytetään. Toinen taas on taiteillut kuvion, joka kankaaseen painetaan, mutta kolmas suunnittelee eri väryhdistelmät. Neljäs taas suunnittelee leikkaukset ja saumat kun taas viides taskujen muodon, ja niin edelleen. Vaatteiden teko on yhteistyötä, niinkuin niin moni muukin työ tässä maailmassa. Ja yhteistyö on Open Sourcen vahvaa aluetta.

Kuten Linux-jakelijoitakin, varmasti myös Open Collection -vaatteiden valmistajia ilmestyisi useampia. Ja miksi ei ilmestyisi? Jos muodikkaita designeja saisi ilmaiseksi verkosta ja kysyntää löytyisi, niin totta kai valmistajat kiinnostuisivat niistä. Ja samoin kuin Linux eroaa Microsoftista, myös Open Collection eroaisi perinteisistä vaatebrändeistä siinä, että sitä ei omistaisi kukaan. Se olisi meidän kaikkien yhteinen brändi. Aasialainen valmistaja ei joutuisi maksamaan kalliita lisenssimaksuja brändin omistajalle vaan voisi sen sijaan maksaa enemmän palkkaa työntekijöilleen, eli niille jotka todella tekevät ne vaatteet. Suurten vaatehtaiden lisäksi olisi myös pienimuotoisempaa valmistusta. Teinityöt voisivat halutessaan ommella uudet muotivaatteensa itse ja mummot voisivat kutoa Open Collection -villapuseroita joululahjaksi lastenlapsilleen. Jotkut tekisivät vaattensa itse säästääkseen rahaa, toiset siksi että itse tekeminen on hauskeempaa ja suurin osa ostaisi vaatteensa edelleenkin kaupasta, koska heillä on tärkeämpääkin tekemistä kuin ompeleminen. Siis kaikki ihan niinkuin Linuxinkin kanssa. Jotkut ostaisivat halvimmän valmistajan teettämiä vaatteita, kun taas toiset suosisivat kotimaista tai vaikkapa käsin ommeltua. Ihan niinkuin Linuxinkin kanssa. Koulussa oppilaat voisivat ompelutunnilla tehdä itselleen omat Open Collection -paidat ja painattaa niihin haluamansa kuvio kuvaamataidon tunnilla. Tai tulostaa se. Kyllä vaan, värimustesuihkutulostimiin voi nykyään ostaa erityistä paperia, josta tulostetun kuvan voi silytsraudalla painaa vaikka T-paitaan. Huomaatko, miten me insinöörit todellakin olemme tehneet kaikkemme, jotta vaatealan ihmiset voisivat toteuttaa Open Source -vallankumouksen seuraavan vaiheen?

Open Collection olisi myös siitä erilainen brändi, että siinä näkyisi Open Source -proessin jättämä jälki. Kaksi samanlaista paitaa eivät varmaankaan olisi täysin identtisiä, jos niillä on eri valmistaja. Ne voisivat kyllä olla saman designin pohjalta tehtyjä, mutta toisessa olisi käytetty meksikolaista puuvillakangasta, kun taas toisessa intialaista pellavaa. Tietenkin kotikutoiset paidat olisivat jokainen yksilöllisiä myös. Mutta tähän on juuri sitä mitä me haluammekin. Open Sourcessa on juuri kysymys tekemisen ilosta, monimuotoisuuden ja jokaisen tekijän yksilöllisyyden arvostamisesta!

Toinen asia mikä erottaa Windowsin ja Linuxin toisistaan on se, että Windows on yhtiön valmistama, kun taas Linux on ihmisten valmistama. Tässä on huomattava ero ja ne jotka ovat siirtyneet Linuxin käyttäjiksi ovat yleensä hyvin innoissaan huomattessaan sen. Kun tietokoneen käyttäjällä on ongelmia kasvottoman yhtiön valmistaman suljetun ohjelmiston kanssa, hänen pitää soittaa yhtiön kasvottomaan tuotetukeen. Ja yleensä tuotetuessa vastaa automaattinen puhelinvastaaja, joka käskää painamaan 1, jos sinulla on tällainen ongelma, 2 jos sinulla on toisenlainen ongelma ja jos haluat puhua ihmisen kanssa, niin sitten paina 3. Ja jos erehdyt sitten painamaan kolmosta, niin saat kuulla nauhoitetun viestin kertovan, että "valitettavasti kaikki linjamme ovat nyt varattuja, ole hyvä ja odota vuoroasi."

Sen sijaan, jos löydät virheen Open Source -ohjelmasta, vaikkapa Linuxista, voit kertoa siitä sähköpostitse suoraan ohjelman tekijöille. Sähköpostiosoite löytyy ainakin ohjelman lähdekoodista, tosin yleensä se löytyy sitäkin helpommin esimerkiksi ohjelman ohjesivuilta. Ja yleensä tekijät ovat hyvin iloisia saamastaan sähköpostista ja vastaavat kiittäen, koska olet kertonut heille miten heidän ohjelmaansa voisi parantaa. Jos suljetun ohjelman valmistanut yhtiö ylipäättään vastaisi tällaiseen viestiin, se olisi todennäköisesti yhtiön lakimies, joka kiistäisi koko virheen olemassaolon.

Open Collection -vaatteiden pesuohjelapuissa ei siis saisi lukea "Made in Hong Kong", vaan Open Source -hengen mukaisesti niissä lukisi jotain tyyliin: "Nämä lenkkarit on ommellut Lu Shanghaista, designin on tehnyt Fabrizio Brasiliasta." Jos mukana olisi vielä ompelijan sähköpostiosoitekin, voisi lenkkareistaan iloinen ostaja jopa lähettää lyhyen kiitosviestin Lulle. Samalla hän voisi vaikka kysyä tämän työolosuhteista ja varmistaa, että hän saa varmasti asianmukaista palkkaa ja ettei valmistuksessa ole käytetty lapsityövoimaa. Näin Open Source voisi jopa pureutua niihin ongelmiin, joista globalisaatiota vastustava liike on viime vuosien aikana pitänyt niin kovasti meteliä.

Vaatteet ovat vain yksi esimerkki tuotteesta, jota voisi valmistaa Open Source -menetelmällä. Tämä kirja on kuitenkin nyt kirjoitettu ja vielä kirjoittamatta jääneet avoimuusinnovaatiot jäävät lukijan oman kekseliäisyyden ja mielikuvituksen varaan. Olemme nähneet miten Open Source voi mullistaa maailmaa niin kullankaivuussa, tietosanakirjan julkaisussa kuin vaateteollisuudenkin alalla, ja toivon, että nämä esimerkit voisivat rohkaista tulevaisuudessa yhä uusiin innovaatioihin. Itse uskon, että tämän kirjan tarinat ovat vasta alkua ja että tulevaisuudessa on vielä paljon tilaa uusille Stallmaneille ja Torvaldseille.

Miten olisi esimerkiksi ravintola, joka julkaisisi kaikki salaiset reseptinsä internetissä asiakkaiden käyttöön? Myös muut ravintolat voisivat käyttää reseptejä ja vastaavasti tämä ravintola saisi käyttää niiden reseptejä. Ja asiakkaat voisivat myös lisätä kokoelmaan omia suosikkireseptejään, ja tilata vaikka omaa reseptiään viiden tähden ravintolan parhaimman kokin valmistamana. Ja sitten niillä ravintoloilla voisi olla yhteinen menu ja yhteinen logo ja brändi. Eli Open Source -ravintolaketju. Sen nimeksihän voisi laittaa McOpen...

Niinpä niin. Ideoitahan riittää, varsinkin kun innostuu. Näiden tarinoiden kirjoittaminen on saanut ainakin minut innostuneeksi. Toivottavasti sinutkin, sillä innostuneisuushan on yksi hakkeriyhteisön hyveistä. Toiveeni on, että nämä hakkeriyhteisön tarinat voisivat kannustaa meitä kaikkia entistä avoimempaan ajattelutapaan ja käyttäytymiseen – avoimeen elämään.

112Kuten jo aiemmassa alaviitteessä on todettu, on maailman ensimmäisenä ohjelmitavana koneena, ja sitä kautta kaikkien tietokoneiden esi-isänä – pidetty ranskalaisen Joseph Marie Jacquardin vuonna 1801 keksimää ohjelmitavaa kutomakonetta.
(http://en.wikipedia.org/wiki/Jacquard_loom)

EPILOGI

61. EPILOGI

61. EPILOGI

Juuri kun tätä kirjaa valmisteltiin painokuntoon, ilmestyi Linuxin (lähdekoodin) tuoreimman version mukana Linuxen kirjoittama essee, jossa hän selostaa kernel-projektissa vuosien varrella kehittyntä johtamiskulttuuria. Linus lisäsi kirjoituksensa 10. lokakuuta 2004 Linuxin lähdekoodien tiedostoon *Documentation/ManagementStyle* kommentilla "Esi-isäimme savitauluille riipustamia viisaita neuvoja".

Käyttöjärjestelmän lähdekoodien kylkeen haudattu kirjoitus ei ole juurikaan julkisuutta saanut, mutta siinä kiteytyy niin paljon elämäkokemusta ja viisautta, että sen tulisi ilman muuta kuulua jokaisen yritysjohtajan lukemistoon. Siinä hiljaisen nörtin sisältä kuoriutuu loistava psykologi ja ihmisjohtaja, joka kertoo mitä hän on oppinut toimiessaan menestyksekkäästi jo yli 13 vuoden ajan yhden maailman suurimman ohjelmistoprojektin projektipäällikkönä.

Ja mikä olisikaan parempi tapa lopettaa tällainen kirja kuin antaa mestarille itselleen viimeinen sana.

LINUX-KERNELIN

JOHTAMISTYYLI

62. LINUX-KERNELIN JOHTAMISTYYLI,
KIRJOITTANUT LINUS TORVALDS

62. LINUX-KERNELIN

JOHTAMISTYYLI, KIRJOITTANUT LINUS TORVALDS

Tämä lyhyt kirjoitus kuvaa suositeltavaa (tai kuviteltua, riippuen keneltä kysyt) linux-kernelin johtamistyyliä. Tarkoituksena on seurata OhjelmointiTyyli-kirjoituksen jalanjäljissä ja tavoitteena on välttyä tulevaisuudessa vastaamasta (*) samoihin tai samantapaisiin kysymyksiin uudelleen ja uudelleen.

Johtamistyyli on hyvin henkilökohtainen asia ja paljon vaikeammin määriteltävissä kuin yksinkertaiset koodin tyylisäännöt, joten tällä kirjoituksella ei välttämättä ole mitään tekemistä todellisuuden kanssa, tai sitten on. Koko juttu alkoiikin aika kieli poskessa, mutta se ei tarkoita ettei se voisi olla tottakin. Sinä päätät.

Ainiin. Kun puhumme nyt "kernelin johtajista", kyse on pelkästään teknisistä tiiminvetäjistä, eikä kenestäkään joka johtaa jotain niinkuin yrityksissä on johtajia. Jos tehtäviisi kuuluu hankintojen hyväksyminen tai tiedät jotain tiimisi budjetista, niin melko varmasti et ole tällainen kernelin johtaja. Siinä tapauksessa nämä ohjeet eivät liity sinuun, tai voivathan ne liittyäkin.

Heti kärkeen suosittelen, että käyt ostamassa kirjan "Menestyvän ihmisen 7 elämäntapaa". ÄLÄ lue sitä vaan polta se. Se on hieno symbolinen ele.

(*) Tämä kirjoitus ei niinkään vastaa itse kysymykseen, vaan tarkoituksena on tehdä selväksi se valitettava tosiseikka, että meillä ei ole harmainta aavistusta, mitä kysymykseen tulisi vastata.

Oli miten oli, aloitetaan:

Luku 1: Päätöksenteko

Yleisesti uskotaan, että johtaja tekee päätöksiä ja että päätöksenteko on tärkeää. Ja mitä isompi ja vaikeampi päätös, sitä isompi johtaja tarvitaan sen tekemiseen. Tämä on sekä syvällistä että ilmeistä, mutta ei kylläkään ole totta.

Koko jutun juoni on _välttää_ päätösten tekemistä. Erityisesti, jos joku pyytää sinua valitsemaan "vaihtoehdon (a) tai (b), sinun pitää nyt sanoa miten jatketaan", olet liemessä. Kyllähän johtamiesi ihmisten tulisi tuntea yksityiskohdat sinua paremmin, joten jos he pyytävät sinulta neuvoa tekniseen päätökseen, olet kusessa. Ei sinulla ole mitään pätevyyttä tehdä päätöstä joka kuuluu heille.

(Seuraus: jos johtamasi ihmiset eivät tunne yksityiskohtia sinua paremmin, olet myös kusessa, tosin aivan eri syystä. Nimittäin silloin olet selvästikin väärässä työssä, ja _heidän_ tulisi johtaa sinua.)

Eli pelin henki on _välttää_ päätösten, ainakin isojen ja vaikeiden päätösten tekemistä. Pienten päätösten tekeminen, joilla ei ole mitään merkitystä, on ok ja saa sinut näyttämään siltä, että hallitset johtajanhommasi. Joten kernelin johtajan tulee pyrkiä tekemään isoista ja vaikeista päätöksistä pieniä ja merkityksettömiä, joista kukaan ei välitä.

On hyvä huomata, että olennainen tunnusmerkki ison ja pienen päätöksen välillä on se, voiko tehtyä päätöstä myöhemmin korjata. Mikä tahansa päätös voidaan muuttaa pieneksi päätökseksi varmistamalla, että jos olet väärässä (ja tulet olemaan väärässä), pystyt myöhemmin vielä perumaan päätöksen ja korjaamaan virheesi ja sen seuraukset. Mikä parasta, pääset näyttämään johtajan kykyjäsi kaksinverroin tekemällä _kaksi_ päätöstä – väärän _ja_ oikean.

Ja ihmiset pitävät sellaista johtajuutena (niin hölmöä kuin se onkin).

Niinpä isojen päätösten välttäminen kiteytyykin siihen, ettei pidä tehdä päätöksiä, joita ei voi myöhemmin perua. Älä anna ajaa itseäsi nurkkaan. Nurkkaan ahdistettu rotta voi olla vaarallinen – nurkkaan ajettu johtaja on vain sääliittävä.

Ja nythän on niin, että koska kukaan ei ole niin tyhmä, että antaisi kernel-johtajalle mitään sen suurempaa taloudellista valtaa, useimmat päätökset on aika helppo perua. Koska sinulla ei ole mahdollisuutta tuhata mielettömiä summia rahaa, joita sinulla ei olisi mahdollisuutta maksaa takaisin, niin ainoa päätös mitä sinun tarvitsee perua on tekninen päätös, ja niiden peruminen on hyvin helppoa: ilmoitat vain kaikille, että olit epäpätevä älykkämpi, sanot että olet pahoillasi, ja deleteit kaiken minkä parissa alaisesi ovat työskennelleet viime vuoden. Osoittautuu, että tekemäsi päätös ei ollutkaan kovin iso päätös, koska sen peruminen oli näin helppoa.

Joillain on kuitenkin ongelmia tämän lähestymistavan kanssa, kahdesta syystä:

– On yllättävän vaikeaa myöntää, että olet idiootti. Haluamme yleensä ylläpitää itsestämme ulospäin kaunistavaa kuvaa, ja siksi on usein kovin vaikeaa tunnustaa julkisesti olleensa idiootti.

– Alamaisille insinööreillesi voi myös olla vaikeaa kuulla, että koko heidän viimeisen vuoden työnsä on ollut hukkaan heitettyä. Joten vaikka itse _työ_ oli helposti peruttavissa pelkästään deleteimalla tiedostot, saatatkin peruuttamattomasti menettää kyseisen insinöörin luottamuksen. Ja muistathan: "peruuttamattomasti" oli juuri se sana mitä yritimme välttää, ja näin päätöksestäsi tulikin kuitenkin iso päätös.

Onneksi molemmat ongelmat ovat vältettävissä jos vain myönnät heti kättelyssä, ettei sinulla ole harmainta aavistusta ja ilmoitat etukäteen, että päätöksesi ei ole vielä lopullinen ja saattaa olla kaikin puolin huono idea. Pidätä aina oikeus muuttaa mieltäsi ja tee tämä _hyvin selväksi_ kaikille. On paljon helpompi myöntää olevansa tyhmä, kun et ole vielä oikeastaan tehnyt mitään tyhmää.

Niinpä, kun päätös tosiaan osoittautuu tyhmäksi, ihmiset vain pyörittelevät silmiään ja sanovat "Oho. Se teki sen taas."

Tämä ennaltaehkäisevä epäpätevyuden myöntäminen saattaa myös saada ne, jotka oikeasti tekevätkin jotain myös miettimään kannattaako kyseistä asiaa tehdä vaiko eikö. Sillä jos _he_ eivät ole varmoja idean loistavuudesta, sinun ei missään tapauksessa pidä rohkaista heitä lupaamalla, että heidän työnsä tullaan hyväksymään kerneliin. On hyvä heidänkin miettiä kahdesti ennenkuin lähtevät tekemään mitään suurempaa projektia.

Muista: heidän pitääkin tietää yksityiskohdista enemmän kuin sinun, ja yleensä he jo kuvittelevat tietävänsä kaiken. Joten parasta mitä johtaja voi tehdä, ei suinkaan ole itseluottamuksen ruokkiminen, vaan pikemminkin sopiva annos kriittistä ajattelua.

Ainiin. Toinen hyvä tapa välttää päätöksiä on näyttää sääliittävältä ja kysyä "eikö me voida kokeilla molempia?" Usko minua, se toimii tosi hyvin. Jos ei ole heti selvää kumpi vaihtoehto on parempi, he kyllä aikanaan huomaavat sen itsekin. Voi käydä niinkin, että molemmat tiimit turhautuvat ongelmiinsa ja luovuttavat.

Edellinen saattaa kuulostaa epäonnistumiselta, mutta yleensä se on merkki siitä, että molemmissa projekteissa oli jotain vikaa, ja syy miksi he eivät osanneet päättää oli se, että molemmat olivat väärässä. Ja sinä olet sankari ja välitit taas hienosti yhden päätöksen, joka olisi voinut mennä mönkään.

Luku 2: Ihmiset

Useimmat ihmiset ovat idiootteja, ja johtajana joudut elämään sen kanssa, ja mikä on ehkäpä vielä pahempaa, _he_ joutuvat elämään _sinun_ kanssa.

Osoittautuukin, että teknisten virheiden peruminen on helppoa, mutta luonnevian korjaaminen on paljon vaikeampaa. Sen kanssa on vain opittava elämään – myös sinun.

Joten kun valmistaudut kernelin johtajan tehtävään, on parempi ettet turhaan polttele siltoja etkä pommita viattomia alkuasukkaita, etkä myöskään suututa liian montaa ohjelmoijaa. Osoittautuu nimittäin, että ihmisten suuttuttaminen on suhteellisen helppoa, ja heidän lepyttämisen kovin vaikeaa. Siksi "suuttuttaminen" selvästikin kuuluu kategoriaan "peruuttamaton" ja on siten ensimmäisen luvun kiellon alaisuuteen kuuluva asia.

Nämä pari sääntöä on helppo muistaa:

- (1) älä kutsu ketään molopääksi (ainakaan julkisesti)
- (2) opettele pyytämään anteeksi silloin kun unohdat säännön (1)

Säännön #1 ongelma on, että sen rikkominen on tosi helppoa, koska on miljoona eri tapaa sanoa "senkin molopää" (*), joskus et edes huomaa sanovasi niin, ja lähes aina olet täysin vakuuttunut olevasi oikeassa.

Ja mitä vakuuttuneempi olet omasta kannastasi (ja jos totta puhutaan, niin voisit kutsua melkein _ketä tahansa_ molopääksi ja olisitkin täysin oikeassa), sitä vaikeampi on myöhemmin pyytää anteeksi.

Tähän ongelmaan on oikeastaan vain kaksi ratkaisua:

– opi pyytämään anteeksi todella hyvin

– opi osoittamaan "kiintymystäsi" tasapuolisesti kaikille, niin että kukaan ei koe olevansa silmätikkunasi. Ja jos olet tarpeeksi luova, he saattavat jopa ottaa sen huumorilla.

Huomaa, että ylenpalttinen kohteliaisuuksien viljely ei ole toimiva ratkaisu. Kukaan ei luota henkilöön, joka selvästikään ei ole rehellinen.

(*) Paul Simon teki laulun "Fifty Ways to Lose Your Lover" (Viisikymmentä tapaa menettää rakkaasi), koska valitettavasti "A Million Ways to Tell a Developer He Is a D*ckhead" (Miljoona tapaa kertoa ohjelmoijalle että hän on molopää) ei soinnu yhtä hyvin. Mutta uskon että hän harkitsi sitäkin.

Luku 3: Ihmiset II – Hyvät

On siis osoittautunut, että useimmat ihmiset ovat idiootteja, ja valitettavasti siitä seuraa, että sinäkin olet yksi heistä, ja vaikka voit lohduuttaa sillä, että varmasti olet keskimääräistä fiksumpi (sillä kukaan ei koskaan kuvittele olevansa keskitason alapuolella), meidän on kuitenkin myönnettävä, ettemme ole terävimmistä päästä, ja että löytyy ihmisiä jotka ovat vähemmän idiootteja kuin itse olet.

Joillekin fikset ihmiset ovat ongelma. Toiset käyttävät heitä edukseen.

Kernelin johtajana sinun tulee kuulua jälkimmäiseen ryhmään. Nuoleskele heitä kaikin tavoin, sillä he tekevät työstäsi helpompaa. Erityisesti, he pystyvät tekemään päätöksiä puolestasi, ja siitähän alunperinkin oli kysymys.

Joten jos törmäät johonkukaan itseäsi fiksumpaan, peesaa heti niin paljon kuin pystyt. Johtajantehtävistäsi tulee yksinkertaisia, sanot vain "Kuulostaa hyvältä – saat vapaat kädet" tai "Kuulostaa hyvältä, mutta entä X?". Varsinkin jälkimmäinen vaihtoehto on hyvä, sillä joko voit käyttää sitä oppiaksesi jotain uutta aiheesta "X" tai sitten pääset olemaan _erityisen kova_ johtaja tuomalla esiin asian, jota sinua fiksumpi tyyppi ei ollut tullut ajatelleeksi. Molemmissa tapauksissa voitat.

On syytä olla varuillaan ja muistaa, että lahjakkuus jollain alueella ei välttämättä seuraa muille aloille. Joten saatat ohjata ihmisiä tiettyyn suuntaan, mutta hei, he saattavat olla hyviä siinä mitä he tekevät ja surkeita kaikessa muussa. Onneksi kuitenkin ihmisillä on sellainen luonnollinen ominaisuus, ikäänkuin painovoima, että he yleensä palaavat takaisin niiden asioiden pariin, missä he ovat hyviä, joten kun ohjaillet heitä uusille urille, et ole onneksi tekemässä mitään peruuttamatonta, kunhan et vain työnä liian kovaa.

Luku 4: Syyllinen

Asiat menevät pieleen, ja silloin tarvitaan syyllistä. Se olet sinä.

Syyllisyyden myöntäminen ei oikeastaan ole kovin vaikeaa, erityisesti kun ihmiset yleensä ymmärtävät, ettei se ollut _yksin_ sinun vikasi. Mistä pääsemmekin tärkeään puheenaiheeseen: Erityisen hyvä tapa myöntää syyllisyytensä, on tehdä se jonkun toisen puolesta. Se saa sinut tuntemaan itsesi hyväksi ihmiseksi, hän taas on hyvillään kun häntä ei syytetä, ja tyyppi joka menetti koko 36GB pornohakemistonsa sinun töppäilyjesi takia joutuu kyräillen myöntämään, että ainakin olet rehti kaveri kun tunnustit reilusti syyllisyytesi.

Tee sitten _kahden kesken_ selväksi ohjelmoijalle, joka oikeasti töppäsi (jos pystyt hänet löytämään), että hän töppäsi. Ei niinkään siksi, että hän oppisi virheistään, vaan jotta hän tietää olevansa palveluksen velkaa. Ja mikä vielä tärkeämpää, hän on todennäköisesti juuri oikea henkilö korjaamaan vian. Koska – niin juuri – sinä se et ainakaan ole.

Luku 5: Mitä tulee välttää

Ihmiset suuttuvat jos heitä kutsuu "molopääksi", mutta vielä enemmän he suuttuvat, jos sen tekee täysin vakavalla naamalla ja kiihkomielisen julistavaan sävyyn. Ensimmäistä voi vielä pyytää anteeksi, mutta jälkimmäisessä tapauksessa ei sellaista mahdollisuutta yleensä saa, sillä kukaan ei enää kuuntele sinua.

Koska jokainen kuvittelee olevansa muita parempi, meitä _todella_ ärsyttää se, kun joku toinen kuvittelee olevansa jotain. Vaikka kuinka olisit moraalisen ja intellektuellimpi kuin kaikki me muut, ei ole järkevää erityisemmin korostaa sitä, paitsi tietenkin jos todella _haluat_ ärsyttää muita (*).

Vastaavasti ei kannata olla liian kohtelias tai vähäeleinen. Kohteliaisuudesta tulee helposti liioiteltua, niin että ongelmat eivät pääse esiin, ja oletan kuullut sanonnan, "Internetissä ei vähäeleisiä ihmisiä näy eikä kuulu". On paras käyttää kovaa ja tylppää esinettä, kun taot asiiasi ihmisten kalloon, muuten et voi luottaa, että kukaan tajuaa mitä yrität sanoa.

Annos huumoria yleensä auttaa sekä suorasukaisuuteen että moralismiin. Liioittelu aina naurettavuuteen asti voi olla hyvä keino tehdä kantansa selväksi, ilman että siitä tulee vastapuolen kannalta noloa, sillä kaikki luulevat, että sinä vain pelleilet. Se on siis hyvä tapa murtaa sen suojamuurin läpi, millä jokainen suojaa itseään kritiikiltä.

(*) Vinkki: sellaiset internetin keskustelupalstat, joilla ei ole mitään tekemistä työsi kanssa, ovat hyvä paikka purkaa turhautumistasi muihin ihmisiin. Kirjoita oikean törkeän luokkaavia viestejä kunnan röhönaurulla höystettynä, silloin kun tarvitset sisäistä puhdistumista. Kunhan et likaa omaa pesää.

Luku 6: Miksi juuri minä?

Johtajana siis enimmäkseen joudut ottamaan syyt niskoillesi toisten töppäilyistä ja tekemään kaikille harvinaisen selväksi kuinka lahjaton olet, ja on tietenkin järkevää kysyä, että mitä järkeä siinä sitten on?

Ensinnäkin. Vaikket välttämättä saa – tai saatat saadakin, mistä minä tiedän – joukkioittain teinityttöjä (tai poikia, emme me tuomitse ketään ja/tai emme ole sovinnisteja) kirkkumaan pukuhuoneesi ovelle, _tulet_ kuitenkin kokemaan valtavan henkilökohtaisen saavutuksen tunteen siitä, että olet "johtaja". Älä välitä siitä, että todellisuudessa johtajuutesi on sitä, että teet kaikkiesi, että ylipäättään pysyisit muiden perässä ja painat hiki hatussa vain jottet jäisi muista jälkeen. Kaikki kuitenkin pitävät sinua johtajanaan.

Se on loistotyö, jos sinusta on siihen.

(Alkuperäinen englanninkielinen *ManagementStyle*- tiedosto löytyy Linuxin lähdekoodin hakemistosta *Documentation*. Linus lisäsi sen sinne 10.10.2004 kommentilla "Esi-isämme savitauluille riipustamia viisaita neuvoja". Kätevämmin kirjoitus löytyy myös [www:stä](http://wn.net/Articles/105375/) osoitteesta <http://wn.net/Articles/105375/>)