

hyväksymispäivä arvosana

arvostelija

UNIX:in historia

Kehitysyhteisön rooli UNIX:in alkuvuosina

Mikko Visa

Helsinki 15.1.2004
58303315 Tietojenkäsittelytieteen historia seminaari
Tietojenkäsittelytieteen laitos
HELSINGIN YLIOPISTO

Sisältö

1. Johdanto.....	2
2. Yleishistoria.....	2
3. Sosiaalinen ulottuvuus.....	4
4. UNIX:in ja akateemisen maailman symbioosi.....	5
5. UNIX tänä päivänä.....	7
6. Yhteenveto.....	8

1. Johdanto

Tämä seminaarityö käsittelee UNIX:ia [RiT74] ja sen historiaa painottuen erityisesti UNIX:in ja akateemisen maailman yhteistyöhön. Toisessa kappaleessa selostetaan yleinen historia ja tärkeimmät vaiheet UNIX:in synnyssä. Kolmannessa kappaleessa vertaillaan UNIX:in sosiaalista kehitystapaa nykypäivän opensource-liikkeeseen. Neljännessä kappaleessa paneudutaan syvemmin UNIX:in ja akateemisen maailman symbioosiin.

UNIX-käyttöjärjestelmä täyttää 35 vuotta vuonna 2004. Etupäässä Ken Thompsonin ja Dennis Ritchien Bell Laboratories:ssa [Bell] kehittämä järjestelmä on sittemmin saanut satoja ellei tuhansia seuraajia ja klooneja. Maailmanlaajuisen suosion saavuttanut suomalaisen Linus Torvaldsin kehittämä Linux-käyttöjärjestelmä on käytännössä klooni UNIX-käyttöjärjestelmästä. Silloinen UNIX ei ollut niinkään suuri tekninen innovaatio vaan lähinnä sen aikaisiin järjestelmiin verrattuna lähinnä yksinkertaistus. Kuitenkin UNIX näytti, että kohtalaisen pieni käyttöjärjestelmä pystyi toimimaan monilla eri laitealustoilla. Se myös näytti, että käyttöjärjestelmä voi olla useaan ympäristöön helposti siirrettävä ja kaiken lisäksi edullinen [Sal94].

2. Yleishistoria

Vuonna 1965 Bell Labs, General Electrics (GE) ja Massachusetts Institute of Technology (MIT) päättivät toteuttaa käyttöjärjestelmän yhteisprojektina koska kaikilla oli sellaiselle tarvetta. Bell Labs tarvitsi käyttöjärjestelmän ohjelmointi- ja ajoympäristön insinööreille ja tutkijoille sekä yleiseen tietojenkäsittelyyn [Pei85]. Projektin nimeksi tuli Multics. Tarkoituksena oli saada aikaan interaktiivinen usean samanaikaisen käyttäjän osituskäyttöjärjestelmä (vrt. eräajo). Multics-projekti kuitenkin venyi ja yhteistä päämäärää ei loppujen lopuksi löytynyt, joten Bell Labs jättäytyi siitä vähitellen pois vuonna 1968 [Pei85]. Multics oltiin kuitenkin saatu jollain tapaa toimivaksi ja erityisesti Ken Thompson oli uuteen järjestelmänalkuun mieltynyt. Kun Bell Labs jätti yhteistyön Thompson ja muut joutuivat palaamaan takaisin vanhaan GECOS käyttöjärjestelmään, joka ei miellyttänyt Thompsonia. Ken Thompson, apunaan Dennis Ritchie ja muutama muu henkilö, alkoi siksi kehittämään omaa käyttöjärjestelmäänsä käyttäen useita ideoita hyväksi Multics-projektista. Mielenkiintoinen seikka on se, että tähän mennessä varsinainen käyttöjärjestelmien kehitys ei ollut suuresti kiinnostanut Thompsonia eikä Ritchietä. Kun he nyt joutuivat palaamaan takaisin vanhaan ympäristöön, kiinnostuivat he oman järjestelmän kehittämisestä. Laitealustanaan heillä oli ainoastaan nurkasta löytynyt käyttämätön DEC:n PDP-7-tietokone, joka oli varsin pieni järjestelmä

vielä siihenkin aikaan. Lukuisista hakemuksista huolimatta Thompson ja Ritchie eivät olleet onnistuneet saamaan rahoitusta parempaan PDP-11-koneeseen. Yhtiön silloinen johto ei pitänyt järkevänä investoida suuria summia epävirallisen järjestelmän kehitykseen [Rit79]. Kuitenkin vuonna 1969 UNIX kehitettiin PDP-7-laitealustalla ja assembler-kielellä. Samoihin aikoihin Bell Labs'in lakiosastolla oli tarve tekstinkäsittelyjärjestelmälle ja Thompson ja Ritchie saivat heidät kiinnostuneiksi UNIX-järjestelmässä toimivasta vaihtoehdosta. Tämän yhteisprojektin johdosta UNIX-ryhmä sai viimein vuonna 1970 uuden tietokoneen (PDP-11) ja lakiosasto sai tekstinkäsittelyjärjestelmänsä [Moh85]. Seuraavan radikaalin muutoksen UNIX koki, kun se vuonna 1973 kirjoitettiin uudelleen Dennis Ritchien ja Brian Kernighanin samoihin aikoihin kehittämälle C-kielelle [Rit93]. Tämä päätös ei johtanut ulkoisiin muutoksiin, mutta sisäisesti UNIX:sta tuli huomattavasti yleisempi ja rationaalisempi [Rit79]. UNIX:in menestys perustuikin Ritchien mukaan juuri sen ohjelmakoodin luettavuuteen, muunneltavuuteen ja sen ohjelmien sekä kernelin siirrettävyyteen.

Jo ensimmäisellä laitealustalla UNIX:ssa oli monia nykypäivään säilyneitä perusominaisuuksia [Rit79]:

- Hierarkkinen tiedostojärjestelmä, missä oheislaitteetkin esitetään tiedostoina
- Prosessien kontrollointi (prosessien luominen ja käyttäminen)
- I/O-uudelleenohjaus (<,>)
- Komentotulkki (shell)

Sen sijaan esimerkiksi prosessien välinen kommunikointi, taustaprosessien ajaminen ja putket puuttuivat. Putket tulivat UNIX:iin vuonna 1972. Ne mahdollistivat eri komentojen liittämisen toisiinsa ilman, että komentojen tarvitsi tietää mitään toistensa tavoista lukea tai tulostaa tietoa. Mitään kiinnitettyjä tiedostomuotoja ei UNIX:ssa ylipäätään ollut. Periaatteena oli muutenkin, että itse käyttöjärjestelmässä ei ollut mitään turhaa tai ylimääräistä vaan mahdollisimman paljon tehtiin käyttäjätasolla (vrt. linux-kernelin moduuleihin). Tosin tämä periaate on saattanut saada alkuunsa siitä, että PDP-7 oli hyvin pieni laite (se vastasi tehoiltaan Commodore 64-konetta) jo siihen aikaan. Tämä pakotti kehittäjät karsimaan kaiken turhan pois. Siihen asti käyttöjärjestelmiin oli sisällytetty oikeastaan kaikkea mahdollista mitä kehittäjät tunsivat, että joku saattaisi tarvita. Bell Labsin silloinen Multics-projektipäällikkö Victor Vyssotsky toteaa haastattelussa hämmästyneensä kun, UNIX:ssa ei ollut esim. binääri-desimaali muuntamista sisäänrakennettuna [Pei85].

UNIX:ia kehittäessä seurattiin kuitenkin tietoisesti tiettyjä peruseriaatteita kaikkien komponenttien

osalta. Ensimmäinen näistä periaatteista oli se, että jokainen ohjelma tai komponentti tekee vain yhden asian ja se tekee sen hyvin. Eli tehdään mielummin uusi ohjelma uutta työtä varten kuin monimutkaistetaan olemassaolevaa. Jokaisen ohjelman tuli toimia hyvin yhteen muiden kanssa. Tuli olettaa, että ohjelman tuloste on toisen ohjelman syöte. Tuota toista ohjelmaa ei välttämättä edes tunnettu. Ohjelman tulosteita ei siis sotkettu turhalla tiedolla ja binäärimuotoista tulostetta pyrittiin välttämään samoin kuin interaktiivista syötettä [Sal94] ja [Hau94]. Ohjelma tuli lisäksi suunnitella ja rakentaa mahdollisimman nopeasti, jopa kokonainen käyttöjärjestelmä. Eli se tuli saada nopeasti kokeiltavaksi ja toimivaksi. Ei pitänyt myöskään epäröidä hylätä huonosti tehtyjä komponentteja ja uudelleenkirjoittaa niitä.

Näistä periaatteista tuli perusta ja universaali liittymä UNIX:in eri ohjelmien välillä. Näitä periaatteita pyritään edelleenkin noudattamaan useimmissa UNIX-klooneissa.

UNIX kehitettiin nimenomaan miellyttäväksi ympäristöksi ohjelmoida ja ajaa ohjelmia. Tällaista järjestelmää Ken Thompson jäi nimenomaan kaipaamaan Multics-projektin epäonnistumisen jälkeen [Rit77].

Suuri osa UNIX:in kehityksestä Bell Labsissa tapahtui ennen vuotta 1978. Tässä vaiheessa moni alkuperäinen kehittäjä oli jo vaihtanut työnkuvaan. Samaan aikaan UNIX oli kuitenkin noussut tärkeäksi järjestelmäksi Bell:n sisällä, joten sille perustettiin tukiryhmä. Tämä tukiryhmä ei ollut niin kiinnostunut UNIX:in kehittämisestä ja uusista innovaatioista vaan, se lähinnä yritti saada aikaan mahdollisimman stabiilin version siitä. Vuodesta 1978 Bell Labs ei enää antanut UNIX:in lähdekoodia ilmaiseksi vaan alkoi veloittamaan siitä. Siten UNIX versio 7, joka oli tehty mahdollisimman siirrettäväksi järjestelmästä toiseen, muodostaa pohjan kaikille muille olemassa oleville UNIX klooneille. Vuonna 1979 Bell Labs ilmoitti kaupallistavansa UNIX:in kokonaan. Tässä vaiheessa yliopistot ottivat ohjat käsiinsä, ensimmäisten joukossa Kalifornian Yliopisto [Iva85]. Se kehitti oman versionsa UNIX:sta, jota se kutsui nimellä BSD UNIX (Berkeley Software Distribution). DARPA:n sponsoroima Internetin kehitys tapahtui BSD UNIX ympäristössä. Sitten moni kaupallinen UNIX-klooni perustui BSD UNIX:iin. BSD:n suureen suosioon vaikutti se, että sillä oli hyvin selkeä ja salliva lisenssi. BSD antoi myös tunnustusta niille, jotka siihen kehittivät ohjelmia. Lisäksi sen hinta muodostui lähinnä jakelumedian hinnasta [Sal94].

3. Sosiaalinen ulottuvuus

UNIX:in kehitystavasta voidaan löytää monia yhtäläisyyksiä nykyisin hyvin suosittuun opensource-toimintatapaan. Etenkin tapa, jolla UNIX alunperin kehitettiin ja jaeltiin muistuttaa hyvin paljon sitä tapaa, millä vaikkapa Linuxia tai lukuisia muita opensource-ohjelmia kehitetään ja jaellaan tänä päivänä. Kehitysyhteisö on laaja, joskus jopa maailmanlaajuinen, ja se tekee tiivistä yhteistyötä. Lähestulkoon kuka tahansa voi lähettää tekijöille korjausehdotuksia tai jopa suoraan korjauksia (patch), jotka tekijät joko hyväksyvät tai eivät hyväksy. Juuri näin kehitettiin myös UNIX. Koska Bell Labs ei voinut mainostaa eikä tukea UNIX:ia saati julkistaa korjauksia siihen se pakotti käyttäjät yhteistyöhön ja näin syntyivät käyttäjäyhteisöt. Käyttäjät jakoivat keskenään tietoa, ohjelmia ja korjauksia sekä ohjelmiin että rautaan [Sal94]. Etenkin sen jälkeen, kun yliopistot hankkivat itselleen UNIX-lisenssin, alkoi erittäin suotuisa yhteistyö kehittäjien ja akateemisen maailman välillä. UNIX:in jakelutapa ja alhainen hinta eivät suinkaan olleet Bell Labsin päätös, vaan johtui siitä, että Bell Labs ei silloisen sopimuksena USA:n viranomaisten kanssa saanut myydä muuta kuin teleliikenteen alan ratkaisuja. Muussa tapauksessa UNIX olisi saattanut jäädä täysin kaupalliseksi tuotteeksi ja tietojenkäsittelytieteen historia olisi täysin erilainen kuin mitä se nyt on. Tämä seikka siis johti siihen, että UNIX:ia pystyttiin jakamaan vain omakustannehintaan.

4. UNIX:in ja akateemisen maailman symbioosi

UNIX:in kehittämisessä käytettiin hyväksi akateemisen tutkimuksen tuloksia. Tämä on seurausta siitä, että Thompson ja Ritchie suunnitellessaan UNIX:ia käyttivät hyväkseen oppimiaan asioita Multics-projektissa. UNIX ei siis ollut sattumanvarainen muutaman hakkerin projekti vaan sitä lähdettiin tekemään, koska sellaiselle järjestelmälle oli tarvetta. Tosin tuo tarve oli lähinnä Bell Labsin tutkijoilla eikä mitään virallista määräystä tai suunnitelmaa toteuttaa käyttöjärjestelmä siis ollut [Vys85]. Tapa, jolla UNIX kehitettiin, poikkesi radikaalisti perinteisestä tavasta rakentaa järjestelmiä. Tuosta tavasta voidaankin vetää analogia tämän päivän opensource-liikkeeseen. Koska Bell Labs ei saanut myydä UNIX:ia kaupallisesti eikä liioin myydä tukea siihen, se päätti myydä sitä haluaville tahoille omakustannehintaan (tämä oli kuitenkin mahdollista). Tämä johti siihen, että erityisesti monen yliopiston tietojenkäsittelytieteen laitokset kiinnostuivat järjestelmästä ja hankkivat sen itselleen. Koska UNIX:in mukana tuli täydellinen lähdekoodi, yliopistot saattoivat tehdä omia parannusehdotuksia järjestelmään. Kun palaute otettiin huomioon, UNIX:in suosio tietysti vain lisääntyi ja se sai osakseen jatkuvasti enemmän hyväksyntää kehittäjäyhteisöltä. Jo vuoden päästä UNIX:in ensimmäisen version julkaisun jälkeen vuonna 1970 syntyi UNIX:in ja akateemisen maailman välille symbioosi. Ensimmäisten joukossa oli Berkeleyssä sijaitseva Kalifornian yliopisto, joka myöhemmin kehitti oman version UNIX:sta nimeltä Berkeley Software

Distribution (BSD). Ken Thompson vieraili Kalifornian yliopistossa professorina vuosina 1975-76 ja vei näin lisää UNIX-osaamista akateemiseen maailmaan. Myös Bell Labs kutsui usein tutkijoita akateemisesta maailmasta viettämään muutaman vuoden Bell Labs:lla [Iva85]. Akateeminen maailma siis hyötyi UNIX:ista ja UNIX puolestaan hyötyi akateemisesta maailmasta. Tämä yhteistyö määritteli harmonian, joka oli tyypillinen varhaiselle UNIX-yhteisölle [McK85].

Toisenlaisia näkökulmia on myös esiintynyt. Esimerkiksi John Stoneback kommentoi artikkelissaan [Sto85], että yliopistojen ja UNIX:in välinen kiinteä suhde saattaisi vaikuttaa negatiivisesti luovuuteen ja nerokkuuteen, joka johti itse järjestelmän (UNIX) kehittämiseen.

Akateeminen maailma hyötyi UNIX:sta

Akateemisessa maailmassa oli jo pitkään ollut tarve paremmille tietojenkäsittelyn järjestelmille. Tuohon aikaan moni tietojenkäsittelytieteen laitos tuskaili uusien ja kalliiden laitehankintojen kanssa. Tietokonekeskukset eivät helposti antaneet erityislaitteita vaativille laitoksille laitteita käyttöönsä eivätkä liioin ottaneet vastaan akateemikoiden parannusehdotuksia laitteistoihin. Lisäksi suuret keskustietokoneet olivat aivan liian kalliita laitoksille. UNIX:in tulo, erityisesti sen ositusominaisuudet, oli näille laitoksille iso hyppäys eteenpäin teknologisessa kehityksessä.

Lisäksi UNIX:in hinta oli minimaalinen. Lähdekoodin sisältävä akateeminen UNIX-lisenssi maksoi vain muutamia satoja dollareita (omakustannehinta). Lisäksi UNIX toimi pienissä laitteissa, jotka mahtuivat pienien laitosten budjetteihin, esim. PDP-11 [Com85].

UNIX:in etu oli myös se, että sitä oli helppo muokata ja ymmärtää (koska UNIX oli uudelleenkirjoitettu C:llä, joka oli korkean tason kieli) [Com85]. Siihen aikaan yksi ihminen pystyi ymmärtämään koko koodin (n. 10 000 riviä), joten muutoksien tekeminen ei ollut vaikeaa.

UNIX:in kautta opiskelijat pääsivät ensimmäistä kertaa tutustumaan oikean käyttöjärjestelmän täydelliseen koodikantaan joka ei ollut ollut mahdollista aiemmin. Esimerkiksi New South Wales:in yliopisto Australiassa oli eräs ensimmäisiä UNIX:ia käyttäviä yliopistoja [Iva85]. Vaikka kampuksen tietokonekeskus ei suostunut tukemaan UNIX:ia, paikallinen tietojenkäsittelytieteen laitos otti sen käyttöönsä. Tämän seurauksena laitoksen käyttöjärjestelmien opetus muuttui radikaalisti, kun opiskelijoilla ensi kertaa oli pääsy täydellisen käyttöjärjestelmän lähdekoodiin. Lisäksi se johti tiiviiseen yhteistyöhön ja ajatustenvaihtoon muiden yliopistojen kanssa Usenet:in

kautta, koska virallista tukea UNIX:iin ei ollut. Tämä poiki lopulta Australian UNIX systems User Group:in.

UNIX:in avulla pystyttiin hyödyntämään iso osa suurempien laitosten tekemistä tutkimuksista koska kaikki käyttivät sitä pohjana tutkimuksissaan [Sto85]. Erääksi hyödyksi on myös mainittu se seikka, että UNIX oli todistus siitä, että ohjelmistoja todella pystyttiin tekemään kuten siihen aikaan opetettiin [Sto85]. UNIX antoi uskottavuutta uudelle tieteenalalle, tietojenkäsittelytieteelle, joka niihin aikoihin pyrki todistelemaan olemassaoloaan varteenotettavana tieteenhaarana.

Lisäksi UNIX:in kasvuvauhti nopeutui huomasti kun Bell Labs julkaisi version 32/V, joka oli muunnelma versiosta 7. Tämä versio toimi DEC:in VAX-koneella. VAX-järjestelmistä tuli suosittuja yliostojen keskuudessa samalla kun niiden suosio muuallakin kasvoi. Kasvu puolestaan rohkaisi yliopistoja hankkimaan enemmän VAX-laitteita tutkimusprojekteihin, jotka tarvitsivat suurempia osoiteavaruuksia. VAX:it olivat tähän tarkoitukseen edullinen ja tehokas hankinta. Yliopistot pystyivät näin helposti ja edullisesti kasvattamaan tietojenkäsittelykapasiteettiaan [Com85].

UNIX hyötyi akateemisesta maailmasta

Kun Bell Labs julkaisi UNIX:in version 7, siinä oli jo mukana yliopistomaailmasta tulleita lukuisia uusia ominaisuuksia ja parannuksia. Samaan aikaan myös kysyntä taitavista ohjelmoijista kasvoi. Koska yliopistoilla oli UNIX saatavilla, useat opiskelijat tunsivat järjestelmän hyvin siirtyessään työelämän puolelle. Tämä puolestaan rohkaisi työnantajia valitsemaan UNIX:in käyttöjärjestelmäkseen, koska niin moni osasi sujuvasti käyttää ja muokata järjestelmää. Jossain vaiheessa tutkijat alkoivat käyttää UNIX:ia perustana lähes kaikessa työssä [Com85]. Tämä johtui luonnollisesti siitä, että UNIX oli saatavilla joka puolella. Lisäksi tutkijat pystyivät muokkaamaan järjestelmää miten halusivat. Eri tutkijoiden välinen yhteistyö hyötyi tilanteesta, koska useilla oli nyt sama järjestelmä ja pystyttiin esimerkiksi verifioimaan toisen tutkijan teorioita ja kokeita. Pystyttiin myös entistä helpommin jatkamaan jonkun muun työtä, kun ympäristöt olivat samat. Koska sekä tutkimuksessa että tavanomaisessa toimistokäytössä oli nyt sama järjestelmä, pystyttiin tutkimustulokset siirtämään entistä nopeammin laboratorion tuotantopuolelle.

5. UNIX tänä päivänä

AT&T (ent. Bell Labs) myi oikeudet UNIX:iin vuonna 1993 Novell:lle, joka puolestaan myi oikeudet SCO:lle vuonna 1995. Tällä hetkellä on epäselvää kuka omistaa oikeudet UNIX:iin. Asiasta on vireillä useita oikeusjuttuja. Sekä SCO että Novell väittävät omaavansa oikeudet. SCO väittää lisäksi, että Linux sisältää kaupallista UNIX-koodia ja on vaatinut korvauksia useilta tahoilta. IBM ja RedHat ovat nostaneet vastakanteet. Yleinen uskomus on, että SCO on nostanut kanteet vain ja ainoastaan pönkittääkseen osakekurssiaan. Tätä väitettä tukee se fakta, että suurin osa SCO:n johtoryhmästä on osakkeiden arvon nousun jälkeen myynyt runsaasti osakkeita [Groklaw].

Voidaan kuitenkin sanoa, että UNIX:iin perustuvat käyttöjärjestelmät ovat viime vuosina nousseet jälleen pinnalle. Näistä etupäässä Linux on saavuttanut suuren suosion etenkin palvelinpuolella. Kaikista maailman www-palvelimista noin 67% on Linux-pohjaisia Apache-palvelimia (tieto perustuu Netcraft'in tilastoihin 15.1.2004 [Netcraft]). Myös Linuxin asema työasemissa näyttää vahvistuvan kaiken aikaa, etenkin valtion laitoksissa ja muissa yhteiskunnallisissa instituutioissa. Tämän lisäksi Linux on lyönyt läpi akateemisessa maailmassa ja ohjelmistokehityksessä.

6. Yhteenveto

Ruohonjuuritasolta alkunsa saanut UNIX:in ja sen seuraajien tarina tietojenkäsittelytieteen historiassa on ollut pitkä, kenties pidempi kuin minkään muun ohjelman. Näyttäisi myös siltä, ettei taru ole vielä lopussa pitkään aikaan vaikka matkassa on ollut lukuisia mutkia.

Lähteet

- [Moh85] August Mohr. The Genesis Story. *Unix Review*. Volume 3, January 1985.
- [Com85] Douglas Comer. Pervasive UNIX: Cause For Celebration. *Unix Review*. Volume 3, January 1985.
- [McK85] Marshall Kirk McKusick. Unix Unleashed – The university role of research in maintaining system vitality. *Unix Review*. Volume 3, October 1985.
- [Pei85] Ned Peirce. Putting Unix in Perspective – An Interview with Victor Vyssotsky. *Unix Review*. Volume 3, January 1985.
- [Iva85] Peter Ivanov. An Interview with John Lions. *Unix Review*. Volume 3, October 1985.
- [Sto85] John Stoneback. The Collegiate Community – Does UNIX leave room for growth? *Unix Review*. Volume 3, October 1985.

- [RiT74] Dennis M. Ritchie, Ken Thompson. The UNIX Time-Sharing System. Communications of the ACM. Volume 17(7), July 1974.
- [Rit77] Dennis M. Ritchie. The UNIX Time-Sharing System – A Retrospective. Tenth Hawaii International Conference on the System Sciences, Honolulu. January 1977.
- [Rit79] Dennis M. Ritchie. The Evolution of the Unix Time-Sharing System. Language Design and Programming Methodology Conference, Sydney Australia. September 1979.
- [Sal94] Peter H. Salus. Unix at 25. *Byte Magazine*.
- [Rit93] Dennis M. Ritchie. The Development of the C Language. ACM SIGPLAN Notices, Volume 28, No. 3 March 1993.
- [Hau94] Rhonda Hauben. Unix and Computer Science. *The Amateur Computerist*. Volume 6(1), 1994.
- [Bell04] Bell Laboratories: <http://www.bell-labs.com/> [15.1.2004]
- [Netcraft04] Netcraft: Web Server Survey Archives
http://news.netcraft.com/archives/web_server_survey.html [15.1.2004]
- [Groklaw04]<http://www.groklaw.net> [15.1]