

Kratka priča o sustavima otvorenog koda

radna verzija

ili kako zbog čega i zašto sustavi otvorenog koda

Autori:

Hrvoje Horvat

Hrvoje Varga

Radne inačice dokumenta:

Autor	Poglavlja	Datum izmjene
Hrvoje Horvat i Hrvoje Varga	1-7	2017-2018
Hrvoje Horvat	2.2	09.07.2018
Hrvoje Horvat	1-7	09.02.2019

Licenca: GPL v.3.

Sadržaj

1	Predgovor.....	4
2	Što je otvoreni kod.....	5
2.1	Što je prethodilo “open source” pokretu i kako se razvijao.....	5
2.2	Ponovni povratak u povijest.....	6
2.2.1	Priča o Unixu.....	6
2.2.2	Priča o BSDu i FreeBSD.....	9
2.2.2.1	Tranzicija BSD UNIXa u Open Source.....	9
2.2.3	Priča o tvrtki SUN Microsystems.....	11
2.2.3.1	Tranzicija u Open Source.....	11
2.2.4	Priča o web pregledniku Netscape.....	12
2.3	Kako je nastao pojam i ideja „Open Source”.....	12
2.3.1	Kako je došlo do razvoja Linuxa.....	15
3	Licence otvorenog koda.....	17
3.1	Što su opće licence otvorenog koda i čemu služe ?.....	17
3.2	Zbog čega je važan odabir licence.....	17
4	Koje su prednosti sustava otvorenog koda.....	18
4.1	Tko sve koristi Linux i sustave otvorenog koda.....	27
4.2	Tko još (pаметan) razvija programe i sustave otvorenog koda.....	30
4.3	Filmovi i literatura koju preporučujemo.....	31
5	Primjeri razvoja sustava otvorenog koda.....	32
6	Kako razvijati sustave otvorenog koda i živjeti od toga.....	33
6.1	Dodatne informacije i primjeri.....	33
7	Izvori informacija.....	33

1 Predgovor

U novije vrijeme, sve više smo svjedoci događaja koji možda izgledaju sporadični. Naime govorimo o prelasku tvrtki, državnih ili gradskih institucija na sustave otvorenog koda. Najnoviji takav slučaj je grad Barcelona. Oni su najavili kako će tijekom 2018 godine uložiti 70% gradskog budžeta za softver, koji su inače izdvajali za plaćanje licenci raznog vlasničkog softvera (Microsoft Windows, Office i drugi), u proces prelaska na sustave otvorenog koda (Engl. *Open Source*). Kako bi uspjeli ovaj projekt dovesti do kraja, odlučili su zaposliti 65 programera u lokalnim tvrtkama, čiji budući posao će biti prilagodba postojećeg softvera otvorenog koda kao i izrada novog, za potrebe grada.

Što su oni odabrali:

- za operacijski sustav: **Ubuntu Linux**
- za službeni uredski programe: **Libre Office**
- za internet preglednik: **Firefox**
- za poslužitelj elektroničke pošte (*e-mail*): **Open-Xchange**

Izvorni dokument: <https://itsfoss.com/barcelona-open-source/>

Ovaj projekt prelaska sa vlasničkog softvera na softver otvorenog koda je jedan od svijetlih primjera, kojima se Barcelona priključila inicijativi: **Public Money, Public Code** - *Javni novac, Javni kod*. Peticiju za prelazak na programe otvorenog koda, za sve javne (državne) institucije, na razini Europske unije, možete popuniti i vi, na adresi:

<https://creativecommons.org/2017/09/18/sign-petition-public-money-produce-public-code/>.

Važan prvi korak, prije ovoga odradila je grupa entuzijasta u Bugarskoj, koja je upornim promoviranjem sustava otvorenog koda uspjela s nemogućim: izglasan je zakon, s kojim je za državu i sve državne institucije, dozvoljen razvoj, nadogradnja i implementacija, SAMO i ISKLJUČIVO sustava otvorenog koda.

Izvorni dokument: <https://thepolicy.us/bulgaria-got-a-law-requiring-open-source-98bf626cf70a>

Kako vam ova vijest ne bi izgledala kao sporadičan pokušaj, dalje u tekstu taksativno ćemo navesti i mnoge druge primjere upotrebe programa i sustava otvorenog koda.

Nadalje, pokušati ćemo odgovoriti na pitanja o tome kako uopće razvijati sustave otvorenog koda i živjeti od toga, čak i u našem okruženju. Koji su izazovi i problemi u razvoju takvih programa i sustava i kako to sve funkcionira. Jeli to uopće moguće ?.

Namjera izrade ove kratke priče o sustavima otvorenog koda je slijedeća:

1. Senzibilizacija šire javnosti o prednostima upotrebe i razvoja sustava otvorenog koda, posebno za sve državne, županijske i gradske institucije.
2. Lobiranje za izradu zakona koji bi, na razini države, za sve državne i druge javne institucije, minimalno:
 1. kako bi svi **novi** programi i sustavi u razvoju, nadogradnji i implementaciji morali biti otvorenog koda, slobodni za korištenje, modifikacije, te distribuciju i bez ograničenja za upotrebu
 2. kako svi **novi** prateći programi ili sustavi, poput baza podataka, aplikacijskih, mrežnih i drugih poslužitelja, moraju biti otvorenog koda, slobodni za korištenje, modifikacije, te distribuciju i bez ograničenja za upotrebu.

Primjer zakona: <https://translate.google.com/translate?act=url&depth=1&hl=en&ie=UTF8&prev=t&nurl=translate.google.com&sl=bg&tl=en&u=http://lex.bg/laws/ldoc/213555445>

3. Upoznavanje pojedinaca, tvrtki, sveučilišta i fakulteta o prednostima upotrebe i razvoja sustava otvorenog koda
4. Promoviranje razvoja i upotrebe sustava otvorenog koda na sveučilištima, fakultetima i školama
5. Pokretanje novih tvrtki koje bi razvijale, održavale ili pružale podršku za sustave otvorenog koda
6. Osvještavanje pojedinaca i tvrtki o komercijalnoj strani sustava otvorenog koda

2 Što je otvoreni kod

Otvoreni softver odnosno kôd (Engl. *Open source*) je softver čiji su izvorni programski kôd i/ili nacrti, odnosno dizajn, dostupni javnosti na uvid, korištenje, izmjene i daljnje distribuiranje. Dakle izvorni kod je dostupan korisnicima koji ga mogu mijenjati, prepravljati i poboljšavati njegov sadržaj.

To znači kako uz **open source** programe dolazi i izvorni kod samih programa, pisanih u nekom programskom jeziku. Ovo nije slučaj sa vlasničkim softverom.

Danas postoji veliki broj licenci koje definiraju prava i obveze kako autora, tako i korisnika softvera otvorenog koda. Obično autor zadržava pravo da se kod daljnje distribucije njegovog koda, mora vidjeti tko je prvi autor, odnosno autor prve verzije programa, dok je obično obveza korisnika da sačuva informaciju o autoru programa.

Međutim postoji cijeli niz licenci, prema kojima se programski kod može razvijati i distribuirati te koristiti.

Neke od češće korištenih licenci:

- **GPL** - *GNU General Public License* (veliki broj GNU programa, Linux Kernel, OpenOffice, WordPress, ...)
- **LGPL** - *GNU Lesser General Public License* (Primjerice 7-Zip)
- **OSL** - *Open Software License* (Magento)
- **Apache License** (Apache Web server, Apache Ant ,Jackrabbit , ...)
- **CDDL** - *Common Development and Distribution License* (pr. Open Solaris , ZFS, GlassFish)
- **SISSL** - *Sun Industry Standard Source License* (pr. OpenOffice)
- ...

Kasnije ćemo se vratiti na licence i njihovu važnost.

2.1 Što je prethodilo “open source” pokretu i kako se razvijao

1970 godine razvijena je prva verzija **UNIX** operativnog sustava a 1.1.1970 se smatra datumom rođenja *UNIXa*. Razvili su ga Ken Thompson i Dennis Ritchie za tvrtku *AT&T Bell Laboratories*. Zbog njegove dostupnosti i portabilnosti prihvaćen je vrlo brzo u akademskoj zajednici, kao i u raznim tvrtkama. Dizajn *UNIXa* je imao veliki utjecaj na razvoj operativnih sustava kasnije.

1977 godine na sveučilištu *Berkeley (University of California, Berkeley)* razvija se projekt „**free operating system**” i nastaje **BSD UNIX (Berkeley Software Distribution)**, koji je bio baziran na verziji 6 AT&T Unixa. Pošto je *BSD* sadržavao i dijelove izvornog koda od navedenog *UNIXa*, u vlasništvu *AT&T*a isti ih tuže početkom 1990 se spor riješava nagodbom. Iz *BSDa* , nastaje nekoliko distribucija otvorenog koda: **FreeBSD**, **NetBSD** i **OpenBSD** , koje su i danas u razvoju i upotrebi.

1985 na sveučilištu *Carnegie Mellon* počinje se razvijati jezgra (*kernel*) operacijskog sustava, pod projektom **Mach**. On postaje baza za kasniji razvoj mnogih drugih kernela i operativnih sustava , poput **OSFa**, **GNU Hurda**, te **Darwina** a koji je pak baza **MacOSa** i **iOSa** , te mnogih drugih.

1988 godine **DEC**, **IBM** i **HP** formiraju *Open Software Foundation* za razvoj nove verzije *UNIXa* (*OSF*), prva verzija izlazi 1992, a fondacija se zatvara 1994 ali iz nje izlaze druge verzije *UNIXa*: *Digital UNIX*, *True64*, kao i razne napredne tehnologije.

Za više detalja o nastanku i razvoju Unixa, pogledajte knjigu:

<https://www.opensource-osijek.org/knjige/kratka-povijest-UNIXa-od-UNICSa-do-FreeBSDa-i-linuxa.pdf>

2.2 Ponovni povratak u povijest

U ovoj cjelini (2.2.x) govoriti ćemo malo detaljnije o povijesti razvoja UNIXa koja je uopće i dovela do razvoja drugih operativnih sustava i programa otvorenog koda. Stoga ćemo morati pratiti nekoliko naizgled odvojenih priča. Naime sve ove priče govore o uzročno posljedičnim vezama u razvoju programa i sustava, koje ćemo spomenuti a iz kojih se mogu izvući pouke i danas.

Ako vas ove povijesne činjenice i uzročno posljedične veze ne zanimaju, slobodno preskočite na poglavlje 2.3.

2.2.1 Priča o Unixu

Priča o nastanku UNIXa prati nevjerovatne ljude, vizionare i vrhunske inženjere. Ljude koji su bili spremni odreći se puno toga, na putu k ostvarenju svojih vizija. Nevjerovatne pojedince i grupe ljudi koji su cijelo to vrijeme bili puni entuzijazma te nisu bili spremni na kompromise. Upoznati ćemo se (ili se podsjetiti) ovih stvarnih velikana IT svijeta koji su promijenili naš svijet a da toga možda nismo niti svjesni.

Bez njih ne bi bilo niti interneta (barem onakvog kakav danas poznajemo), Unixa, Linuxa, Mac OS-a, ili iPhone-a i Androida a ne bi bilo niti BlackBerry-ja. Ne bi bilo niti programskog jezika C, kao niti jezika koji su nastali na osnovi programskog jezika **C, C++ , Go, Rust, Java, Java Script, Limbo, LPC, C#, Objective-C, Perl, Python, Swift, Verilog** i mnogih drugih.

Samo programski jezik Java danas koristi preko 3 milijarde računala i raznih uređaja : od “set-top box” uređaja, preko pisača, web kamera, navigacijskih sustava, raznih terminala, medicinskih uređaja, kućanskih aparata, televizora i sl.

Dodatno Java *Virtual Machine* koriste i drugi programski jezici, poput jezika: **Scala, Groovy, JRuby** - implementacije **Ruby** jezika te **Jython** - implementacije Python jezika. I programi za Android se pišu u programskom jeziku Java, uz korištenje Android biblioteka. Potom se kod prevodi (Engl. compile) u bytecode za **Java Virtual Machine**, koji se zatim prebacuje u bytecode za Android Runtime (ART) iz kojega se i pokreću svi Android programi. Java Virtual Machine (JVM) odnosno Java virtualni stroj je program koji interpretira i prevodi Java program u Java bytekod, a koji se potom može izvršiti (pokrenuti).

Osim toga ne bi bilo niti operativnih sustava koji su danas u upotrebi (ili bi bili znatno lošiji), te raznih, vrlo važnih tehnologija koje se nalaze ugrađene u sve današnje operativne sustave. Dodatno i većina operativnih sustava je razvijena u C jeziku ili nekoj njegovoj “izvedenici”, a ne zaboravimo i na upravljačke programe (Engl. Driveri) za iste sustave, koji se također moraju razvijati u nekom programskom jeziku poput C, C++ ili slično.

Možda niste znali ali **Appleovi MacOS** i **IOS** su također bazirani na određenim varijantama UNIXa.

Ne zaboravimo niti sve mrežne uređaje koji su okosnica Interneta, od usmjerivača (Engl. Router), preklopnika (Engl. Switch), vatrozida (Engl. Firewall), IPS i IDS uređaja i sličnih. Svi oni upogonjeni su u vrlo velikoj mjeri nekom varijantom UNIXa ili Linuxa.

Tako primjerice **Cisco Systems** za svoj **IOS** (operativni sustav) koristi (**Free**)**BSD** Unix, dok za **IOS XR** koristi **QNX Unix** odnosno za **NX-OS** koristi **MontaVista Linux** itd. S druge strane **Juniper Networks** koristi **FreeBSD** Unix za sve svoje mrežne uređaje. **Dell Networking** uređaji za primjerice N seriju preklopnika koriste Linux (**DNOS 6.x OS**), dok za S i Z seriju (koju su naslijedili kupnjom tvrtke **Force 10**, koriste **FTOS** koji je baziran na **NetBSD** Unixu ...

Nadalje i velika većina digitalnih telefonskih centrala (ako ne i sve) koristi neku varijantu UNIXa. Sve mobilne mreže ili barem vrlo veliki dio uređaja, servisa i opreme koji stoji iza njih također koristi neki UNIX. Pošto pristup internetu u konačnici završava kod nekog telekoma ili ISP-a, ponovno dolazimo do zaključka da bi “Internet” bez UNIXa postao upitan.

Ne bi bilo niti razvijene robotike: ne zaboravimo da su svi operativni sustavi u robotici ili direktno nasljednici nekog UNIXa ili su ideje njihovog razvoja potekle od njih. Najpoznatiji primjeri Unix-a koji pripadaju Tzv. kategoriji “*Real Time*” operativnih sustava (koji zadovoljavaju potrebe robotike koja je najčešća u industriji) su :

- **QNX** (Danas “Black Berry”) - pr. BlackBerry 10 je baziran na QNX Realtime UNIXu
- **VxWorks**
- i deseci drugih *Realtime* operativnih sustava

1969 godina

Krajem 1960 istraživačka tvrtka Bell Labs je bila uključena u projekt sa sveučilištem **MIT** i tvrtkom **General Electric** na razvoj “time-sharing system” operativnog sustava, koji je nazvan **Multiplexed Information and Computing Service (Multics)**. On je omogućavao većem broju korisnika paralelan (istovremeni) rad, odnosno pristup “*Mainframe*” sustavu.

Mainframe sustavi su snažna i kompleksna računala koja su se koristila unutar vrlo velikih organizacija ili tvrtki za kritične i zahtjevne zadatke (aplikacije). Pojam “Mainframe” je nastao od naziva “main frames” - nazovimo ga glavnim okvirom (ormarom ili ormarima) unutar kojih su se nalazile komponente poput glavne memorije (RAM), CPU (central processing unit) i svih ostalih komponenti. Ova riječ se veže za vrlo snažne sustave ali i prva komercijalna računala za poslovnu primjenu. Mainframe sustavi su se intenzivnije počeli razvijati tijekom 1960, te se razvijaju i koriste i danas.

Mainframe računala se koriste i danas za vrlo specifične namjene, recimo u bankarskim ili nekim drugim osjetljivim sustavima. Takav je primjer IBM zEnterprise System koji se razvija i koristi i danas

Nezadovoljan napretkom u razvoju ovog sustava **Bell Labs** se povukao. Ken Thompson koji je radio kao programer u njihovom razvojnom odjelu a i tada je radio na razvoju *Multicsa*, odlučio je napisati (razviti) svoj operativni sustav. S obzirom da je i dalje imao pristup razvojnom okruženju *Multicsa* prvo je odsimulirao novi datotečni sustav te "paging" sustav za memoriju. Ubrzo je zaključio da mu je potreban hardver koji je znatno jeftiniji i pristupačniji. U Bell labsu je naišao na malo korišteni hardver DEC PDP-7 koji je poslužio svrsi.

Na navedenom DEC PDP-7 hardveru, je dvojac Ken Thompson i Dennis Ritchie, uključujući Rudda Canadaya razvio novi operativni sustav.

Ovaj operativni sustav je uključivao:

- hijerarhijski datotečni sustav
- koncept procesa
- sustav posebnih (*device*) datoteka, vidljivih u današnjim UNIX i Linux sustavima unutar direktorija `/dev/` a koji predstavljaju same uređaje (hardver).

Osim toga razvili su **CLI** (Engl. *command-line interpreter*) te nekoliko uslužnih programa. Unutar dodatnih mjesec dana Thompson je implementirao cijeli sustav u programskom jeziku Assembler. Novi operativni sustav, koji će kasnije postati UNIX, sadržavao je assembler (programski jezik), osnovni tekst Editor i Shell (CLI).



Nevjerojatan dvojac: Ken Thompson i Dennis Ritchie, odlučio je zapravo, promijeniti svijet, ne prihvaćajući kompromise i postojeća rješenja. Zapitajmo se što bi bilo s razvojem Unixa, za početak, da je navedeni dvojac, bez pogovora, slušao svoje "autoritete" (koji su praktično odustali od razvoja), bez razmišljanja i vlastitog intelekta. Naime oni su se kao dobri znanstvenici i inženjeri, vodili i premisom "Uvijek preispituj autoritete". Dakle razmišljaj svojom glavom, ne dopusti drugima, mada i bili autoriteti bilo koje vrste (znanstveni, menadžerski i sl.), da zamagle ili obezvreduju vlastite ili tuđe spoznaje, vizije i ideje.

... i zemlja je nekada bila ravna ploha, za sve autoritete tog vremena, zar ne ?

Početak 1970 Peter G. Neumann je ovaj projekt nazvao Unics (UNiplexed Information and Computing Service) za razliku od Multicsa (Multiplexed Information and Computer Services). Tako je novi projekt dobio ime a uskoro su uslijedila sredstva za razvoj i tranzicija imena u UNIX.

Vrlo brzo, razvojnom Unix timu su se pridružili i mnogi drugi

Početak 1970 Peter G. Neumann je ovaj projekt nazvao **Unics** (*UNiplexed Information and Computing Service*) za razliku od Multicsa (Multiplexed Information and Computer Services). Tako je novi projekt dobio ime a uskoro su uslijedila sredstva za razvoj i tranzicija imena u UNIX.

Paralelno s razvojem UNIXa, Dennis Ritchie i Ken Thompson krenuli su i s razvojem programskog jezika **C**. Pošto je originalna inačica UNIXa razvijena na PDP-11, u Assembleru, ovaj dvojac je odlučio preći na programski jezik **B**, koji je ranije razvio Ken Thompson uz pomoć kolege Dennisa Ritchiea. Pošto programski jezik **B** nije mogao iskoristiti sve prednosti hardvera PDP-11 te je imao i neke druge nedostatke, odlučili su razviti i novi programski jezik, koji je nazvan **C**. Intenzivniji razvoj **C** jezika je krenuo 1972. godine, na UNIXu koji se pokretao na PDP-11 hardveru.

C jezik u početku nije razvijan s idejom portabilnosti na drugi hardver ali je ona vrlo brzo postala potrebna. Ubrzo nakon toga, **C** jezik, je portan (prenesen/razvijen) i na druge platforme: Honeywell 6000 a kasnije i na IBM System/370. Iste godine veći dio UNIXa je također prepisan u C jeziku te je već tijekom 1973 i gotovo cijeli UNIX kernel prepisan u C jeziku.

Negdje do 1977, Ritchie i Stephen C. Johnson napravili su dodatne modifikacije unutar **C** jezika koje su omogućile jednostavniju portabilnost i samog UNIXa na druge platforme.



Isti dvojac (Ken Thompson i Dennis Ritchie), vođen inženjerskim instinktom i načinom razmišljanja, ne pristaje na kompromise, već radi ono što bi i svaki izvrstan inženjer **morao** raditi: razvijati kvalitetne, robusne, pouzdane i sigurne proizvode.

Bilo bi dobro sve napraviti iz početka, ali prvo je potrebno razviti novi programski jezik (**C**) pa u njemu, od nule, napisati cijeli **Unix**. Ovakav problem je svakom izvrsnom inženjeru, samo izazov i poticaj. Negdje nakon distribucije Sedme verzije (V7) UNIXa 1978, razvoj je prebačen na grupu "UNIX Support Group (USG)" koja ga nastavlja razvijati. AT&T je najavio **Unix System III** krajem 1981 godine kao prvo izdanje koje nije razvijano unutar *Bell Labsa* već unutar **USG-a**. Ova USGova prva inačica (nakon sedme inačice UNIXa (V7)) nazvana je: **UNIX System III** (3) koja je razvijena tijekom 1982- 1983 godine. Ona je zapravo bila kombinacija raznih AT&T verzija UNIXa.

Unix System V (5)

USG je ubrzo (1983 godine) objavio **Unix System V** (čita se pet) koji je bio nasljednik **Unix System III. System V Release 4 (SVR4)**: Tijekom 1988 te 1989. godine *USL* i *Sun Microsystems* su razvili novu inačicu (4) *Unix System V*, uz upotrebu najboljih tehnologija iz drugih Unixa.

System V Release 4 (SVR4) je uveo mnoga poboljšanja, funkcionalnosti i standarde koji se koriste i danas.



Razvoj UNIXa se nastavio, te je tvrtka AT&T kasnije licencirala svoju inačicu 5 edukacijskim institucijama a inačicu 6 i komercijalnim korisnicima. Škole su plaćale licencu 200 U\$ a svi ostali 20.000 U\$.

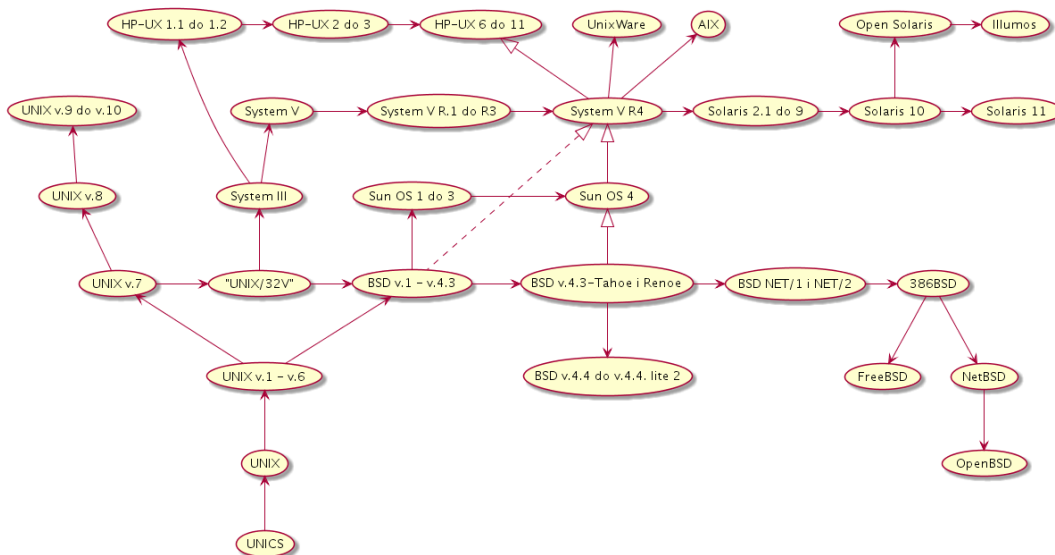
Unix System V R4

Razvojem **Unix System V (5)** sustava, u jednom trenutku, zapravo pojavom podinačice R4, dakle pojavom: **Unix System V R4**, uvedeni su:

- napredni mrežni model koji je uključivao **TCP/IP** mrežni protokol
- virtualni datotečni sustav (**VFS**)
- mrežni sustav za dijeljenje datoteka (**NFS**)
- tada vrlo napredni UNIX datotečni sustav (**UFS**)
- grafičko sučelje (**OpenWindows GUI**)
- napredni shell (**korn shell**)
- **ANSI C** programski jezik
- **init** skripte (kasnije nazvan **SysV init**)

Kao i i druge vrlo važne promjene koje će, kako će i vrijeme pokazati, značajno utjecati na razvoj svih drugih operativnih sustava.

Tvrtka **Bell Laboratories** je još razvila inačice **Unix System: V8** (1985.g.), **V9** (1986.g.) i **V10** (1989.g.) međutim upravo **Unix System V R4** je bio pokretač velikih promjena. Pogledajmo i kako su se razvijale razne inačice UNIXa, o kojima ćemo govoriti u slijedećim poglavljima:



2.2.2 Priča o BSDu i FreeBSDu

Ponovno ćemo se vratiti u povijest jer ova priča počinje negdje u vrijeme razvoja prvih AT&T odnosno Bell Labsovih inačica Unixa.

Profesor Bob Fabry koji je radio na kalifornijskom sveučilištu Berkeley, pokrenuo je nabavku (kupnju) UNIX Source licence od AT&T-a. Dakle licence koja obuhvaća izvorni kod njihovog UNIXa. Plan je bio modifikacija postojeće verzije AT&T UNIXa, prema njihovim željama odnosno planovima. Tako je nastao **BSD Unix** (Engl. *Berkeley Software Distribution*). S obzirom na to kako su i druga sveučilišta postala zainteresirana za rad na ovoj inačici Unixa, krenulo se s kompiliranjem prve inačice "**Berkeley Software Distribution**" (**1BSD**), koja je objavljena početkom 1978.



Zanimljivo je i to kako je 1975 godine, Ken Thompson "uzeo slobodno" od Bell Labsa i došao na sveučilište Berkeley kao gostujući profesor. On je pomogao pri instalaciji AT&T UNIXa verzije 6 te je krenuo s implementacijom programskog jezika Pascal na istu platformu.

Razvijale su se inačice za inačicama (**1BSD**, **2BSD**, **3BSD**). Na osnovi inačice **3BSD**, koja je bila široko prihvaćena a koja je i primijećena od strane **DARPA** agencije, razvijena je i inačica **4BSD**, krajem 1980. Negdje sredinom 1980 profesor Fabry je potpisao ugovor s **DARPA** agencijom (Engl. *Defense Advanced Research Projects Agency*) za potrebe **ARPAnet-a**, preteče današnjeg interneta.

Sada je osnovni cilj razvoja cijele generacije 4 BSD Unixa postao podrška za DARPA-in "**Internet networking protocol**" odnosno TCP/IP set protokola. Vrlo brzo implementacija ovog mrežnog protokola omogućila je komunikaciju uređaja (poslužitelja) koji su bili u lokalnim mrežama, povezanim ili **Token Ring** ili **Ethernet** mrežama te komunikaciju preko mreža koje povezuju velike udaljenosti (WAN mreže), kao što je bio DARPA-ina **ARPANET** mreža.

Razvijale si se nove pod inačice: 4.0, 4.1 i 4.2, a na inačicu 4.2 je značajno utjecalo DARPA povjerenstvo. Dodatno su implementirana mnoga poboljšanja, među kojima je uvođenje "**disk quota**" i "**job control**" mehanizama. Mike Karels i Marshall Kirk McKusick preuzimaju vodstvo u upravljanju daljim razvojem.

4.3BSD (1986.g.) : Nakon nekoliko mjeseci intenzivnog testiranja TCP/IP dijela, od strane DARPA agencije, utvrđeno je kako je TCP/IP iz 4.2BSD verzije superioran u odnosu na druge implementacije (jer su se razmatrale i implementacije drugih tvrtki) te kako će on ostati unutar nove inačice 4.3.BSD.

Uslijedile su inačice **4.3 BSD Tahoe** (1988.g.) te **4.3 BSD Reno** (1990.g.).



U međuvremenu (između **Tahoe** i **Reno**) inačica pojavile su se pod inačice koje su jednostavno nazvane: **4.3BSD Net1** (početak 1989.g.) i **4.3BSD Net2** (sredina 1991.g.). Ove dvije inačice su imale zadatak izbacivanja kompletnog (svog) koda koji je bio vlasnički (Engl. *Proprietary*) odnosno pokriven "UNIX source" licencom od tvrtke AT&T. **To je označilo ogromnu prekretnicu u daljem razvoju sustava otvorenog koda (open source).**

2.2.2.1 Tranzicija BSD UNIXa u Open Source

Sve do inačice **4.3 Tahoe**, svi korisnici BSD Unixa su prvo morali platiti AT&T-u *source* licencu. To je bilo stoga što BSD sustav nikada nije objavljen od strane sveučilišta Berkeley, samo u binarnom formatu, već je cijeli **BSD Unix** distribuiran sadržavajući cjelokupni izvorni kod kompletnog UNIX operativnog sustava sa svim njegovim dijelovima.

S vremenom su troškovi AT&T-ove "source" licence sve više rasli (klasična priča o pohlepi i menadžmenu) te su svi oni željni razvoja svojih varijantu UNIXa baziranih na BSDu, a koji su poglavito koristili odlično razvijeni TCP/IP i pripadajuće mrežne servise odustali od daljeg rada. Uz odustajanje sve više njih je tražilo od *Berkeleya* odvajanje svog mrežnog koda, kako za TCP/IP tako i za sve pripadajuće servise, te izdavanja istih pod nekom drugom, normalnijom/manje pohlepnom, licencom.



Osmišljena je **BSD licenca**, koja je bila liberalna. Svatko je mogao koristiti ili distribuirati izvorni ili promijenjeni programski kod ili binarno, bez plaćanja ikakve licence. Jedini zahtjev je bio da se izvorni dokument s pravima upotrebe, autorima i sl. moraju distribuirati nepromijenjeno. Praktično se navodi da program sadrži izvorni kod Kalifornijskog sveučilišta Berkeley i pojedinaca koji su sudjelovali u razvoju.

BSD licenca je tzv. *non-copyleft* licenca, za razliku od recimo GNU GPL licence. To znači da ako program razvijate prema BSD licenci, programski kod MOŽETE objaviti u cijelosti ali imate i mogućnost da određeni dijelovi vašeg programa budu dostupni samo i isključivo u binarnom obliku. Binarni oblik podrazumijeva nedostupnost samog programskog koda. To implicira mogućnost da možete imati program koji se sastoji od dvije datoteke, prvoj možete objaviti programski kod u cijelosti, a drugu možete kompilirati te ju distribuirati u binarnom obliku - iz kojega se programski kod ne može iščitati.

Na početku je i Berkeley naplaćivao 1.000 \$ za traku (vrpču za pohranu podataka) na kojoj je bio pohranjen izvorni kod ali je svatko mogao kopirati kod od nekog drugoga tko je već kupio izvornu traku (medij s podacima). Čak se vrlo brzo pojavilo nekoliko internet stranica koje su preko FTP protokola omogućavale pristup samom kodu.

U međuvremenu je osnovana **CSRG** (Engl. *Computer Systems Research Group*) radna grupa čiji zadatak je bio praktično pisanje cijelog novog UNIXa od početka, ali pod BSD licencom.

Networking release 2

Zbog uspjeha prvog **Networking Releasea**, CSRG, Keith Bostic (vodeći programer) je krenuo u prepisivanje sve većeg djela programskog koda i ostatka BSD Unixa, pod BSD licencom. Dakle plan je bio razviti ga od početka i objaviti mu izvorni programski kod.

Prvo se krenulo s raznim uslužnim programima (*UNIX utilities*), koji su morali biti napisani od nule, ne koristeći niti liniju koda od strane AT&T-a odnosno koristeći samo dostupne opise funkcionalnosti postojećih UNIX programa.

Jedino što bi autori dobili bi bilo spominjanje njihovog imena u Berkeley licenci, za programski kod (program) koji bi napisali

Ubrzo se lista "odrađenih" programa povećala na 100 i nastavila je rasti te su unutar 18 mjeseci dovršeni svi važni programi i biblioteke.

Ostao je sam kernel koji je bio i najveći zadatak. Kernel je polako revidiran, datoteka po datoteka, te se polako izbacivao licencirani (AT&T Unix 32/V) programski kod te razvijao novi (slobodni) kod.

Nakon mjeseci rada, ostalo je još samo nekoliko datoteka koje je trebalo promijeniti. Međutim ipak je odlučeno kako će se konačna inačica objaviti što prije, te je objavljena pod imenom **Networking Release 2**, sredinom 1991. godine (s time da je ostalo još šest datoteka koje nisu "odrađene").

Ponovno je nekoliko stotina pojedinaca i organizacija platilo po 1.000 US\$ kao potporu za razvoj, koji se još više ubrzao.

Paralelne priče: BSD/386 te 386/BSD, NetBSD i FreeBSD

Berkeley je razvijao i "proprietary" odnosno "licenciranu" verziju od strane nove tvrtke BSDi (*Berkeley Software Design*) koju je osnovalo sveučilište Berkeley. Ova inačica je imala naziv **BSD/386**.

William Jolitz, koji je do tada radio u BSDi, je sam odlučio odraditi i ostatak posla tada (onih šest datoteka) te je nakon 6 mjeseci rada objavio programski kod, te sve kopilirao i izradio kompletno novu inačicu operativnog sustava na 386 PC arhitekturi, koju je nazvao **386/BSD**.

Nažalost zbog manjka vremena, William Jolitz se više nije mogao posvetiti daljem radu, zbog sve većih potreba i želja za razvojem, novim funkcionalnostima i sl. BSDi je kasnije s BSD/386 krenuo s prodajom licenci i podrške. Prednost ove verzije (BSD/386) razvijane od strane BSDi je bila u vrlo opširnim i detaljnim uputama (*man* stranicama), od naredbi, do cijelog sustava. Odličnih objašnjenja sintakse, primjera i sl. Ovaj operativni sustav je kasnije nazvan **BSD/OS**.

NetBSD i FreeBSD

Već par mjeseci kasnije, nakon izdavanja Jolitzove 386/BSD verzije, formirala se aktivna grupa korisnika tog sustava te je oformila grupu imena NetBSD koja je nastavila s razvojem. Početkom 1993 doradili su verziju koju su nazvali NetBSD.

Njihov plan razvoja je bio usmjeren na multiplatformnost, dakle mogućnost pokretanje NetBSDa na što većem broju hardverskih platformi. Do 1998 njihova distribucija je bila isključivo dostupna preko mreže (interneta), ali bez izrađenog instalacijskog medija (CD i sl.) te je bila namjenjena samo "hard-core" tehničkim korisnicima.

S druge strane, nekako s vremenom formiranja NetBSD grupe formirana je i FreeBSD grupa, čiji cilj je bio samo podrška za PC platformu (arhitekturu računala), s jednostavnijom instalacijom te su izdali instalacijski CD-ROM disk 1993 godine. S obzirom na jednostavnu instalaciju te veliku promociju na sajmu Comdex FreeBSD je najšire prihvaćen te se njegov razvoj drastično ubrzao. FreeBSD je dobio dodatnu popularnost uvođenjem emulacije Linuxa, koja je omogućavala pokretanje Linux programa (binarnih datoteka) direktno na FreeBSD sustavu uz sve prednosti FreeBSDa kao pouzdanog, robusnog i performantnog operativnog sustava.

1995 godine pojavio se i OpenBSD, koji je izrastao iz NetBSD grupe. Njihov fokus je na povećanju sigurnosti cijelog sustava. Vrlo brzo su i oni krenuli s izdavanjem (i prodajom) CD-ROM instalacijskih medija.

U priči s NetBSD i FreeBSD Unixom, razvijene su stotine Unix programa kao i novi operativni sustav Net/Free/BSD, pod BSD licencom, s javno dostupnim odnosno otvorenim programskim kodom, koji se kasnije nastavio koristiti i razvijati

2.2.3 Priča o tvrtki SUN Microsystems

Za vrijeme studija na sveučilištu **Stanford** u Kaliforniji, Andy Bechtolsheim je za projekt “**Stanford University Network communications**”, napravio računalo koje je nazvao **Sun-1**. Osnovna namjena mu je trebala biti upotreba kao radne stanice za CAD programe. Ova radna stanica je koristila Motorolin procesor 68000 koji je imao ugrađen napredni **MMU** (Engl. *Memory Management Unit*) a koji je omogućavao Unixu, upotrebu sustava virtualne memorije.

Prvu verziju radne stanice Bechtolsheim je napravio od rezervnih dijelova koje je dobio od sveučilišnog odjela računalnih znanosti te drugih dobavljača koji su radili u Silicijskoj dolini.



Od 1940 do 1950 godine, tadašnji dekan Frederick Terman, je poticao sveučilište i studente na pokretanje tehnoloških tvrtki. Njegov plan je bio iskoristiti zemljište u vlasništvu Stanforda i pretvoriti ga u Tzv. Stanford Industrial Park, kasnije preimenovan u **Stanford Research Park**. Dodatno, osmislio je kooperativni program, od kojega su koristi imale nove tvrtke, studenti i sveučilište. **Stanford Industrial Park** je tako postao prvi poslovni inkubator za buduće tehnološke tvrtke, kojima bi sveučilište pomagalo u radu i razvoju. Sve tvrtke su potpisivale ugovor na pet godina, što se smatralo dovoljnim, za samostalan dalji rad i razvoj. Jedna od prvih tvrtki koja se tamo doselila je i poznati Hewlett-Packard, koju su osnovala dvojica Stanfordskih studenata (William Hewlett i David Packard). Malo po malo, broj tvrtki je rastao, te se sve pretvorilo u ono što je danas poznato kao Silicijska dolina.

Početak 1982 godine, radu Bechtolsheima su se priključila još dvojica Stanfordskih studenata: Vinod Khosla i Scott McNealy koji su u okviru Stanfordskog inkubatora pokrenuli tvrtku Sun Microsystems. Vrlo brzo im se priključio i Bill Joy s Berkeleyja (poznat po razvoju BSD Unixa) tako da se i on smatra jednim od osnivača.



Ime tvrtke **Sun** je nastalo od riječi **Stanford University Network**

Negdje 1987 godine tvrtka je razvila vlastiti procesor (CPU), imena **SPARC**, RISC arhitekture. Uslijedile su nove generacije računala (počevši od Sun-4), na ovom procesoru. Prvi SPARC procesori su bili 32. bitni (SPARC V7 arhitektura), do razvoja 64-bitnih SPARC procesora 1995 godine (SPARC V9 arhitektura).

Početak 1990-tih krenuli su u razvoj SPARC procesora s više jezgri, Tzv. SMP (Symmetric Multi Processing) procesori. Njihov naziv je bio UltraSPARC. Samim time se proširila i namjena, sada se pojavljivao sve veći broj Sunovih poslužitelja. S vremenom se sve više povećavao broj jezgri i funkcionalnosti te su uslijedile nove generacije ovih procesora, poput Sparc T1, T2, T5 ,...



Sun Microsystems je i prije razvoja **UltraSPARC T1** arhitekture procesora, patentirao tehnologiju u kojoj se na jednoj fizičkoj jezgri može izvršavati više threadova (niti) ali ju je ozbiljnije počeo upotrebljavati upravo na ovoj generaciji procesora. Te je već u slijedećoj generaciji UltraSPARC T2, koja je na tržište izašla 2007.godine, došao do brojke osam (8). Dakle na jednoj fizičkoj jezgri se može pokretati do osam (8) niti, što sustav vidi kao osam logičkih jezgri a što je drastično ubrzalo rad sustava i aplikacija.

Isti koncept je kasnije upotrijebio i **Intel**, koji ju je nazvao **Hyper-Threading**, uz sličan dizajn poput UltraSPARC T2, barem što se tiče FPU, Integer ALU jedinica, cjevovoda i cache memorije.

2.2.3.1 Tranzicija u Open Source

I konačno, uslijedio je prelazak u open source svijet. Naime do tada je **Sun** licencirao i prodavao licence za procesore SPARC, drugim proizvođačima, koji su ih proizvodili.

2006. godine odlučeno je da će se dizajn UltraSPARC T1 procesora objaviti prema novoj licenci : GNU GPL. Dakle sve od dizajna i simulatora, do cijele arhitekture Ultra SPARC je objavljeno u GNU GPL licenci. Stoga je pokrenut projekt imena OpenSPARC.

2007. godine Sun je objavio i cijeli dizajn procesora UltraSPARC T2 u GNU GPL licenci, te je iz njega nastao OpenSPARC T2 s 8 jezgri, 16 cjevovoda sa 64 niti (threads) odnosno 64 logičke CPU jezgre.

Sun je istovremeno s radom na SPARC procesorima, radio i radne stanice a kasnije poslužitelje sa Intel i AMD x86 (32.bitni) i x64 (64.bitni) procesorima.

SunOS, Solaris i napredne tehnologije

Operacijski sustav **SunOS**, razvijen je na osnovi BSD Unixa za prva računala tvrtke Sun Microsystems. Razvoj se nastavio sve do inačice 4 (SunOS v.4) ali se tada ime operacijskog sustava promijenilo u **Solaris**.

Solaris je ubrzo sa BSD Unixa kao baze, prešao na **Unix System V Release 4**, već s inačicom Solaris 2 (1992. godine). Kupnjom raznih tvrtki i njihovih patenata, i dolaskom do njihovih tehnologija, dodatno se razvijao i Solaris, kao i razni sustavi i programi koji su uslijedili.

Od SunOSa do Solarisa, razvijale su se i razne tehnologije i protokoli, koji su i danas standardi, a neki od njih su superiorni i danas. S vremenom je tvrtka Sun Microsystems, kupovala, ne samo rješenja, već cijele tvrtke i njihove patente koji su se kasnije iskoristili za razvoj novih programa i sustava, od kojih su za open source zajednicu najpoznatiji:

- **MySQL AB** - tvrtka koja je razvila [MySQL](#) bazu podataka, koju je Sun Microsystems kupila tvrtka Oracle, izašla još naprednija SQL baza podataka imena **MariaDB**
- **StarOffice**, koji je Sun Microsystems kupio a kasnije objavio njegov kompletan izvor kod, pod nazivom: **Open Office**, koji je potom preuzet od open source zajednice i objavljen pod nazivom: **LibreOffice**, a koji se i danas razvija, i u kojemu je ova knjiga i napisana
- Programski jezik **JAVA**, koji je objavljen pod licencom otvorenog koda **GPL**, tijekom 2005 godine
- Sun Microsystems je u trenutku objavljivanja inačice **Solaris 10** operacijskog sustava, odlučio otvoriti kompletan programski kod, pod otvorenom CDDL (Common Development and Distribution License) licencom, tijekom 2005. godine. Već 2006 godine cijeli programski kod Solarisa 10 prebačen u projekt koji je Sun Microsystems pokrenuo, a koji je nazvan **OpenSolaris**.

Iz **Open Solaris** projekta su u open source svijet došle i mnoge napredne tehnologije, koje su kasnije ugrađene u mnoge druge operative sustave.

2.2.3.2 MySQL i Open Office

aaa

... U IZRADI ...

2.2.4 Priča o web pregledniku Netscape

aa

... U IZRADI ...

2.3 Kako je nastao pojam i ideja „Open Source”

Od kraja 1970.g. Unix je sve više bio prihvaćan, od državnih institucija i sveučilišta do većih tvrtki. Pošto je još uvijek bio vezan za skupi hardver a i cijena licenci je polagano rasla, nije bio prihvaćen u malim tvrtkama ili od strane pojedinaca. U tom trenutku, cijeli programski kod - od samog Unixa do pratećih programa, bio je javno dostupan odnosno objavljivan kao *Open Source*.

To što je programski kod bio objavljen i dostupan, ne znači kako se nisu morale plaćati licence.

Kako su troškovi razvoja rasli, a vjerojatno je rasla i pohlepa za zaradom, polagano je razvoj krenuo u smjeru zatvaranja programskog koda za javnost. Dakle više nitko osim vlasnika takvih programa, više nije imao pristup programskom kodu, koji bi mogao optimirati, proširiti ili bilo kako mijenjati. Dodatno pojavili su se novi zakoni koji su autorska prava drastično proširili i na ovo područje.

Tih godina [Richard Stallman](#) je bio student na sveučilištu MIT, te član hakerske zajednice unutar koje se programski kod slobodno dijelio. Nedugo zatim, trend zatvaranja koda se proširio te pogodio i hakersku zajednicu, kojoj je *Stallman* bio član.

Frustriran ovim promjenama, [Richard Stallman](#) pokreće 1983. godine **Free Software Movement** inicijativu za razvoj aplikacija otvorenog koda. Već 1984. godine pokreće i [GNU projekt](#), koji kreće s razvojem prvih programa otvorenog koda (Open Source).

1985 godine, Stallman objavljuje dokument [GNU Manifesto](#), u kojemu opisuje problematiku i uzroke pokretanja GNU projekta, kao i daljnje ciljeve u radu. Ovim dokumentom želio je senzibilizirati zajednicu na negativne trendove u razvoju (zatvaranje programskog koda), te dati svoj doprinos i s moralne i filozofske strane.

Jedan od važnijih krajnjih ciljeva GNU projekta je razvoj slobodnog operativnog sustava imena **GNU**. Put do toga je trebao proteći u razvoju svih programa koji bi bili potrebni za operativni sustav. Pod pojmom “slobodnog” se misli na: “slobodnog za korištenje, mijenjanje i distribuciju”.

Iste godine [Richard Stallman](#) osniva i zakladu [Free Software Foundation \(FSF\)](#) kojoj je inicijalni cilj podupiranje razvoja “*Free Software Movement*” inicijative, koja promovira slobodu učenja, distribucije, stvaranja i modificiranja računalnih programa.

S ovim korakom jača se i pravna potpora svim navedenim projektima i u konačnici razvoju programa i sustava otvorenog koda.

Prema načelima **FSF** nastala je prva od licenci otvorenog koda Tzv. **GNU GPL** odnosno *GNU General Public License* ili prevedeno *GNU opća javna licenca*, ali tek kasnije, negdje tijekom 1989. godine. Naime sve do tog trenutka licence većine programa oduzimale su vam mogućnost (slobodu) dijeljenja s drugima, mijenjanja njenog izvornog koda (ako je uopće bio dostupan) i distribucije istog. Druge licence su vam to sve omogućavale ali uz znatna financijska sredstva, koja su rasla iz godine u godinu.

Kasnije se s ovakvim sličnim (**FSF**) načelima razvijao i BSD Unix, koji je kasnije migrirao u *open source* domenu 1989.godine. Dakle ove dvije priče iako razdvojene, krenule su s istom željom i prema istom cilju.

GNU GPL licenca je Tzv. **Copyleft** licenca, koja govori kako programski kod koji je napisan, može biti kopiran ili mijenjan. Ali sav programski kod nastao od izvornog koda također mora biti objavljen pod istom licencom dakle javno dostupan. To znači da ako uzmete programski kod objavljen pod GNU GPL licencom i mijenjate ga ili proširujete, sve što ste radili MORATE javno objaviti i to ponovno pod istom licencom. Dodatno to znači da ne smijete razvijati svoje programe ili sustave bazirane na GNU GPL programskom kodu ako ih ne mislite dijeliti s drugima, odnosno ako ih želite sakrivati ili zatvarati od drugih.

Ponovni povratak u povijest

Vratimo se na 1984.godinu i GNU projekt

Naime jedna od želja (i planova) je bila, sve programe koji postoje u komercijalnim *Unix* operativnim sustavima napisati “od nule” odnosno ne koristeći niti jednu liniju koda od izvornih programa, jer bi to bilo kršenje autorskih prava. Dakle željelo se napisati nove programe s istom ili proširenom funkcionalnosti, postojećih i objaviti ih pod nekom od *Open Source* licenci.

Veći dio ovog posla, a ovdje govorimo o stotinama i stotinama programa, je dovršen negdje u toku 1992. godine. Cijeli niz programa koje su razvijali, postaje poznat pod nazivom *GNU paketi* ili *GNU programi*.

Slika : Richard Stallman (2014.g)

Autor: By Thesupermat - Own work, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=37264906>



[Richard Stallman](#) zaslužan je za razvoj:

- [Free Software Foundation](#) zaklade
- [GNU projekta](#)
- [Emacs](#) - napredni tekst or
- [GCC \(GNU Compiler Collection\)](#) - znan kao : [gcc compiler](#)
- [GNU Debugger](#)
- [GNU make](#) - *Build automation* alata.

Neovisno o razvoju **FSF** i **GNU** 1991. godine Linus Torvalds razvija jezgru (Kernel) operativnog sustava i slijedi logičan korak odnosno povezivanje stotina i stotina programa pisanih unutar GNU pokreta i **Linusovog** kernela. Od tog trenutka povezano je sve što je bilo potrebno za stvaranje novog operativnog sustava danas poznatog pod nazivom Linux (GNU Linux).

U to vrijeme, pojavile su se i neke kontroverze oko naziva novog operativnog sustava nazvanog **Linux**. Naime Stallman i FSF zaklada nisu priznavali naziv Linux kao naziv za cijeli operativni sustav. Oni su inzistirali na nazivu **GNU/Linux** ili **GNU+Linux** stoga što su tražili neko priznanje za dugogodišnji rad tisuće programera, koji su razvijali one stotine i stotine GNU programa, koji su sada praktično zalijepljeni uz Linux kernel, te postali dio operativnog sustava, koji je zaboravio na njih (barem u imenu).

Sve današnje distribucije Linuxa koriste i GNU programe i programe razvijene pod BSD licencom, tako da smo dobili znatno veći broj korisnih programa. Osim većeg broja programa, dogodilo se i to da su se pojavili programi s istom funkcionalnosti, jedan nastao i razvijen iz GNU projekta a drugi iz BSD priče. Ovo se pokazalo vrlo dobrim jer se sada pojavila mogućnost odabira: boljeg, bržeg i sigurnijeg programa.

Pojavili su se i deseci drugih licenci, te stotine programa koji su razvijani pod njima. Na kraju, broj dostupnih programa se drastično povećao, na tisuće i tisuće, pa tako prosječna instalacija bilo koje od distribucija Linuxa, dolazi s nekoliko tisuća instaliranih programa.

Ideja i misao vodilja *Free Software Movementa* se temelji na nekoliko osnovnih načela slobode :

- sloboda da se softver može pokretati,
- sloboda da se softver može proučavati i mijenjati,
- sloboda da se softver može kopirati i distribuirati (dijeliti) , u izvornom ili promijenjenom obliku

2.3.1 Kako je došlo do razvoja Linuxa

[Linus Benedict Torvalds](#) je tijekom studija, na Helsinškom sveučilištu, 1990. godine prvi puta susreo UNIX i to na DEC MicroVAX poslužitelju, koji je pokretao [ULTRIX](#) (DEC-ova varijanta Unixa). Ovaj za njega novi operativni sustav bio je toliko poticajan za njega, da je za diplomski rad odabrao temu, naziva (Linux) **A Portable Operating System** odnosno prijenosni operativni sustav. Torvalds je kasnije radio i na MINIXu, koji je jedna od varijanti UNIXa, namijenjena uglavnom akademskim zajednicama. Kako je razvoj **kernela** radio na MINIXu, u rad je uključio i neke od programa iz njega. Konačni cilj mu je bio pokrenuti svoj operativni sustav na novom **386 PC** računalu koje je kupio.

MINIX je napravio [Andrew S. Tanenbaum](#), profesor računalnih znanosti. Objavio ga je 1987. godine kao minimalni UNIXoidni operativni sustav, odličan za učenje o principima rada operativnih sustava. Cijeli izvorni kod MINIXa je objavljen ali pod licencom koja nije bila otvorena kao recimo GNU GPL. Tek negdje u toku 2000. godine, licenca je promijenjena u BSD licencu, koja je znatno otvorenija.

Profesor **Andrew S. Tanenbaum** je objavio i knjigu o MINIXu (*Operating Systems: Design and Implementation*), koja je također Linusu bila inspiracija za razvoj svog operativnog sustava.

Zanimljivo da je za razvoj kernela koristio **GNU C compiler**, razvijen u GNU projektu, iako je bilo i drugih dostupnih C kompilera. *GNU C Compiler* se i danas najčešće koristi za razvoj Linux kernela i mnogih drugih sistemskih komponenti za Linux.

25.8.1991 godine, Linus je na [Usenet](#) news grupi : “**comp.os.minix**” objavio:

```
Hello everybody out there using minix - I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).
```

Kasnije, kako se kernel razvijao, dodavao je i druge komponente i programe, kako bi mogao izgraditi rudimentarni operativni sustav. Većina tih programa, preuzeta je iz MINIXa. Kako se novi operativni sustav **Linux** razvijao, svi MINIX programi su izbačeni i na njihovo mjesto su došli **GNU** programi. I sam Linux kernel je kasnije objavljen pod **GNU GPL** licencom.

Slika : Linus Torvalds (2002.g)

Autor: By Unknown photographer who sold rights to the picture to linuxmag.com - Linuxmag.com; The image is from an article in a December 2002 issue of Linux Magazine[1], CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=17991>



Linus Torvalds je zaslužan za :

- [Linux kernel](#)
- [Linux](#)
- [GIT](#) sustav za verzioniranje

Linus Torvalds je, gledajući nazad, u vrijeme razvoja Linuxa, izjavio:

1. Da je GNU kernel bio dostupan 1991. godine, on vjerojatno ne bi radio svoj kernel (Linux)
2. Da razvoj 386BSD (iz kojeg je nastao FreeBSD, NetBSD i dr.), nije kasnio, zbog pravnih problema, odnosno da se pojavio 1991.g., on vjerojatno također ne bi radio na Linuxu.

Od tada do danas pojavile su se razne varijante Linux-a, od kojih su se ustalile tri grane na osnovi kojih su nastali na deseci drugih Linux distribucija:

- Debian
- Slackware
- RedHat

Iz navedenih distribucija, nastale su na stotine i stotine drugih distribucija, od kojih je veliki broj i danas u upotrebi. Čak je došlo do toga da je na osnovi navedenih distribucija nastalo nekoliko većih distribucija, na kojima su se razvile nove distribucije.

Tako je primjerice na bazi *Debian*a, razvijen *Ubuntu*, a samo na bazi njega postoji **69** raznih distribucija Linuxa.

Linux se danas intenzivno koristi na raznim poslužiteljima i stolnim računalima. Osim toga, koristi se i na velikom broju specijaliziranih uređaja:

- set-top box i drugi multimedijalni uređaji
- uređaji za navigaciju
- mrežni uređaji : switchevi (preklopnici), routeri (usmjerivači) i sl.
- kućanski aparati

Nadalje, koristi se u mnogim državnim institucijama, akademskoj zajednici (škole, sveučilišta, fakulteti) i industriji. 99.6% najjačih poslužitelja na svijetu koristi Linux (Pogledajte [TOP 500](#) - popis 500 najjačih superračunala na svijetu). Više primjera o detaljima upotrebe Linuxa, vidjeti ćete u poglavljima koja slijede.

3 Licence otvorenog koda

U ovom poglavlju govoriti ćemo o licencama otvorenog koda, što su one i čemu služe.

Za početak, morate biti svjesni nekoliko činjenica o licencama:

- licence otvorenog koda (*open source*) nisu javne odnosno *Public Domain*. što znači kako s njima dobivate određena prava a određena prava dajete/ograničavate drugima
- potrebno je dobro proučiti licence otvorenog koda i odabrati licencni model koji najbolje pokriva vaše potrebe i na najbolji način štiti vaša/projektna prava
- programe i sustave otvorenog koda je moguće i prodavati primjerice prema **GPL licenci**, ali ste dužni kupcu dostaviti i samu (GPL) licencu.
- kod nekih licenci, moguće je razvijati programe otvorenog koda, korištenjem/povezivanjem s programima koji su komercijalni odnosno koji nisu GPL ili programi otvorenog koda općenito, što vrijedi za neke licence, poput **LGPL** licence
- neke licence dopuštaju razvoj programa, odnosno pisanje programskog koda u više licenci, pa je tako moguće:
 - imati razne dijelove programa objavljene pod različitim licencama otvorenog koda
 - imati razne dijelove programa objavljene dijelom pod licencama otvorenog koda, a dio koda može biti zatvoren
 - ... sve ovisno o kombinaciji licenci za koje se odlučite. Za osnovni popis funkcionalnosti, koje vam omogućuje ili onemogućuje određena licenca, pogledajte: https://en.wikipedia.org/wiki/Comparison_of_free_and_open-source_software_licenses

Detaljan popis i opis svih licenci otvorenog koda, možete vidjeti na: <http://opensource.org/licenses/alphabetical>

3.1 Što su opće licence otvorenog koda i čemu služe ?

Aaaaa

... U IZRADI ...

3.2 Zbog čega je važan odabir licence

Aaaaa

... U IZRADI ...

4 Koje su prednosti sustava otvorenog koda

Korištenjem Linuxa i drugih sustava otvorenog koda (engl. *Open Source*), u odnosu na sustave zatvorenog koda, koje često nazivamo komercijalnim programima/sustavima odnosno vlasničkim softverom, dobivamo mnoge prednosti.

Prva o njih je sigurno **Kvaliteta**. Naime sve komponente Linuxa, počevši od Linux kernela (jezgre) do svih programa, kao i drugih programa i sustava otvorenog koda, razvijani su od strane stotina ili tisuća programera a testirani od strane stotina ili tisuća programera i korisnika istih sustava. Svi uključeni u razvoj i testiranje, a ovdje govorimo, ovisno o programu, i o stotinama tisuća ljudi, kako je praksa pokazala, vrlo brzo odnosno višestruko brže nego kod komercijalnog softvera, uočavali su greške i ispravljali ih ili radili na poboljšanjima, optimizacijama (primjerice ubrzanjima) ili dodavanju novih funkcionalnosti. Sve navedeno drastično je povećalo kvalitetu konačnog rješenja.

Ovdje možemo citirati tvorca Linux kernela:

Linus Torvalds: “**Given enough eyeballs, all bugs are shallow**”.

Dakle ako dovoljno veliki broj ljudi gleda u određeni problem ili grešku (*bug*), svi problemi će vrlo brzo biti otkriveni (i riješeni).

Ako se već bavite razvojem programa, postavite si sljedeća pitanja :

- Koliki postotak ukupnog vremena u razvoju svog programa ili sustava trošite na testiranje u širem smislu riječi ?
- Koliko vremena planirate za dokumentaciju vašeg programa i sustava ?

I došli smo do **sigurnosti**. Sigurnost je usko povezana s paradigmatama vezanim za kvalitetu jer kao što veliki broj programera i korisnika koji razvijaju softver povećava kvalitetu, istovremeno otkriva i sigurnosne propuste i ispravlja ih. U praksi puno brže od komercijalnih rješenja. Naime niti najveće korporacije nemaju na tisuće zaposlenih programera i testera na svakom pojedinom softveru, a pošto je njihov izvorni kod skriven od javnosti nitko drugi im ne može puno pomoći - osim ukazati na problem kada se već pojavio i napravio štetu.

Ako se već bavite razvojem programa, postavite si sljedeća pitanja :

- koliki postotak vremena u razvoju svog programa ili sustava trošite na sigurnost :
 - od faze dizajna
 - preko implementacije do svih faza testiranja ?

Sljedeća stvar je **fleksibilnost**. Komercijalnim proizvođačima programa i sustava u konačnici nije u interesu imati najbolji, najbrži, najsigurniji i najkvalitetniji program ili sustav sa svim funkcionalnostima, koje neki korisnici ili grupe korisnika trebaju jer za to treba imati puno resursa, od programera, testera, hardvera , Jedan od problema u razvoju je i fleksibilnost ali i pitanje koliko je vremena potrebno uložiti u razvoj, a da bi program i/ili sustav bio stvarno fleksibilan. Iza svega navedenog zapravo stoje pitanja koja su specifična kod razvoja sustava zatvorenog koda odnosno proizvođača takvih sustava:

- kako će vam nametnuti potrebu da plaćate nove verzije programa i sustava ako vam je postojeći dobar ili čak odličan
- kako će vas natjerati da kupujete nadogradnje zbog novih funkcionalnosti, ali i sigurnosnih nadogradnji ili popravaka nekih (njihovih) grešaka
- kako će vas natjerati da kupite novi hardver (računalo) ako je postojeći ili novi program ili sustav brz i optimalno radi.

S druge strane sustavi otvorenog koda ne boluju od ovakvih (i drugih) stvari, te je na vama odluka kada ćete primjerice mijenjati vaš hardver (računalo) jer budimo iskreni - nova verzija programa nije napravljena na brzinu samo da se proda pa se kasnije ispravljaju (i često naplaćuju) greške. U ovoj priči vi ste besplatni tester nečijeg programa/sustava, koji još i plaćate. Također, kod pisanja kôda, ako vaš kôd nije fleksibilan u širem smislu riječi, često će vas netko već upozoriti na to i/ili ispraviti vaš “loše” napisan programski kôd. Ova priča se ne zaustavlja na programima. Možda niste znali ali postoje i drugi sustavi, poput centralnih procesora (Engl. *CPU*), koji su objavljeni pod nekom od *otvorenih* licenci, dakle dostupnih svima. Na ovaj način dizajn procesora je otvoren, i svatko može potencijalno otkriti neku manu ili predložiti bilo koje poboljšanje. Primjeri otvorenog dizajna procesora su:

- OpenSPARC <https://en.wikipedia.org/wiki/OpenSPARC>
- OpenPOWER https://en.wikipedia.org/wiki/OpenPOWER_Foundation
- ...

Postoji i cijeli niz projekta otvorenog hardvera - pogledajte stranicu: https://en.wikipedia.org/wiki/Open-source_hardware.

No vratimo se na programe - ako se već bavite razvojem programa, postavite si sljedeća pitanja :

- Koliko je program ili sustav koji razvijate fleksibilan i koliko pazite da kôd bude moguće ponovno iskoristiti (engl. Code reuse).

Interoperabilnost : Programi i sustavi otvorenog koda se u praksi, obavezno drže standarda, te ne uvode neke svoje formate datoteka ili standarde, izmišljene protokole i slično. Stoga imate veću vjerojatnost kako će program ili sustav otvorenog koda, koji vi koristite, raditi u kombinaciji s programom i sustavom koji koriste vaši klijenti ili netko drugi. Također je velika vjerojatnost ako primjerice koristite neki mrežni protokol koji je pisan prema načelima otvorenog koda, iz jednog izvora ili su dio jednog rješenja ili proizvoda, da će isti raditi s drugim rješenjima ili proizvodima.

Naime ovo je najviše slučaj kod raznih kompleksnijih sustava, ali i raznih protokola (primjerice mrežnih protokola) i uređaja ali i kod programa i njihovih međusobno većinom namjerno ili ne, nekompatibilnih formata datoteka.

Podrška : Suprotno mišljenju, podrška za programe i sustave otvorenog koda postoji u istoj mjeri kao i za komercijalne programe i sustave. Vezano za Linux, pogledajte cijene komercijalnih Linux distribucija, uz koje možete dobiti ili doplatiti tehničku podršku. Navesti ćemo samo nekoliko njih:

- Red Hat Enterprise Linux: <https://www.redhat.com/en/store>
- SuSe Linux Enterprise Server: <https://www.redhat.com/en/store>
- ...

A druge strane postoje i distribucije Linuxa koje se mogu slobodno kopirati i instalirati ali svejedno možete platiti podršku za njih, pogledajmo i nekoliko takvih slučajeva:

- Ubuntu Linux: <https://www.ubuntu.com/support/plans-and-pricing>
- ...

I naravno postoji i mogućnost plaćanja podrške, pojedincima ili drugim vanjskim konzultantima ili trećim tvrtkama, koje s time bave, a nalaze se u vašem susjedstvu (gradu ili državi).

U praksi, podrška je višestruko bolja, kvalitetnija i uslužnija za programe i sustave otvorenog koda. Zbog nekoliko činjenica.

Podršku za komercijalne programe i sustave pruža vam tvrtka koja je razvila taj sustav a njeni partneri zaduženi su samo za rješavanje određenih problema. Pri tome, kôd samog programa je i dalje u vlasništvu i dostupnosti **samo** tvrtke vlasnika, pa ovdje često dolazi do razvlačenja odgovornosti te gubljenja vremena za vas kao korisnika njihovog programa/sustava. Tvrtka vlasnik programa/sustava ← > partnerska tvrtka

- Za sve probleme koji traže angažman tvrtke vlasnika programa/sustava, pogotovo ako je to inozemna tvrtka, bez obzira da li ima predstavništvo u vašoj zemlji, ćete se prema raznim statistikama (i mom osobnom iskustvu) načekati a u dosta slučajeva, dobiti ćete rješenje problema koji ste prijavili uz cijeli niz novih problema .
- Podršku za programe i sustave otvorenog koda, uključujući mogućnost promjena samog kôda, što uključuje : popravke grešaka, dodavanje novih funkcionalnosti, raznih optimizacija prema vašim željama i slično, može vam odraditi lokalna tvrtka (iz vašeg grada/županije/države). Što je još bitnije tih lokalnih tvrtki može biti nekoliko, što dovodi do zdrave konkurencije Tada će se svi boriti da pruže bolju i kvalitetniju uslugu za vas kao korisnika nekog programa i sustava otvorenog koda, koji vjerojatno već koriste milijuni ljudi širom svijeta i za koji također postoji podrška od strane stotina ako ne i tisuća tvrtki širom svijeta.
- Za sve programe i sustave otvorenog koda, podršku možete dobiti i od zajednice (engl. *community*) koja je i razvila određeni program ili sustav ali vrlo često je iz te zajednice i izrasla i neka fondacija ili tvrtka kojoj/kojima također možete platiti još višu (i bržu) razinu podrške.

Pitanje za one koji su i dalje skeptični prema podršci za programe otvorenog koda i koji bi platili (i plaćaju) “koliko god treba” :

Da li ste platili komercijalnu podršku za neki program ili sustav otvorenog koda i ostali nezadovoljniji u odnosu na *Close Source* komercijalne sustave za koje (već) plaćate podršku ?

Ako i sami razvijate neki *close source* program ili sustav, Pitajte vaše krajnje korisnike (preko nekog posrednika), kakva je podrška za vaš program ili sustav, odnosno, koliko su zadovoljni ?

Došli smo i do **troškova**. Naime troškovi programa i sustava otvorenog koda su nekada i višestruko manji od troškova komercijalnih programa i sustava :

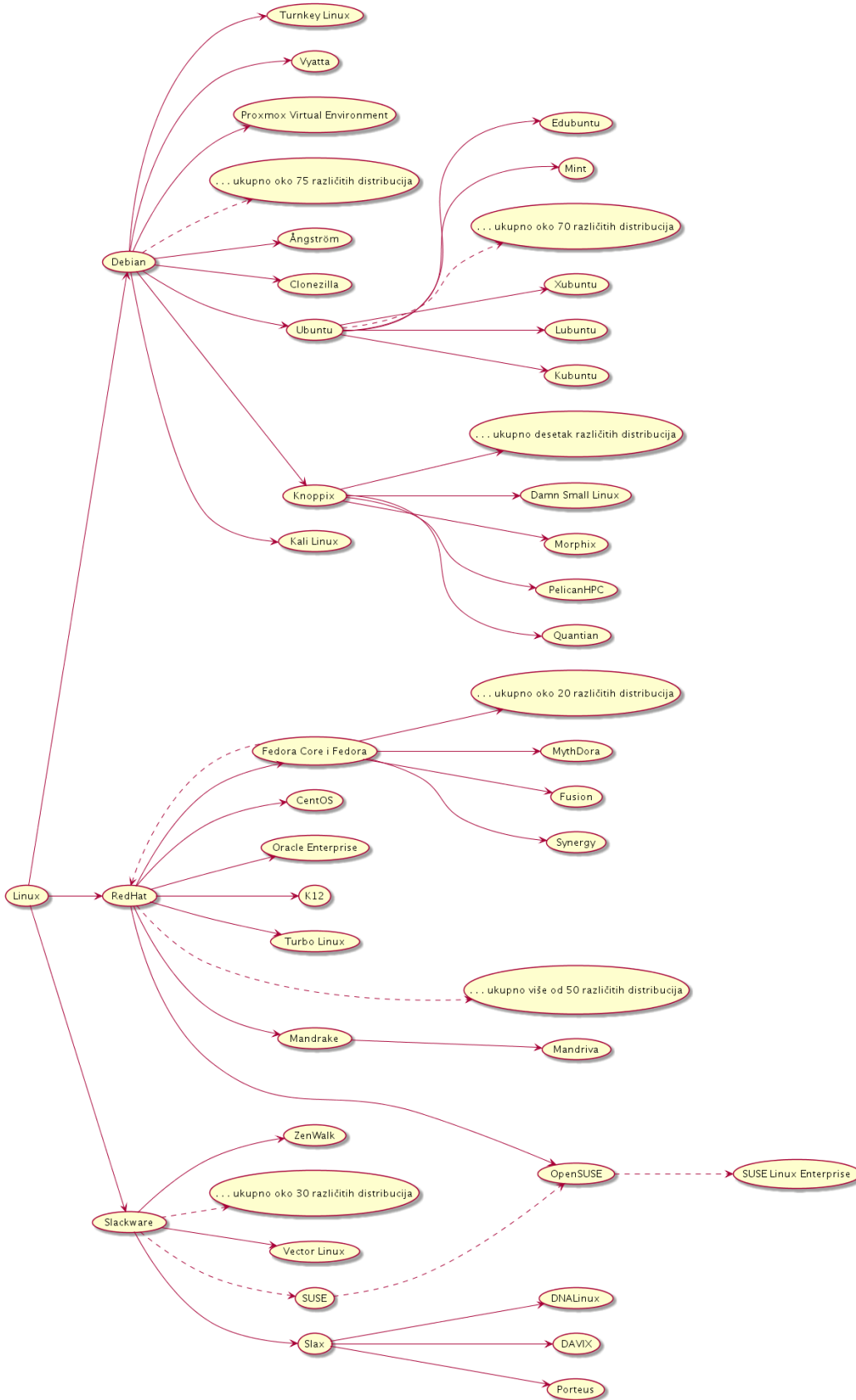
- Većina Linux operativnih sustava je besplatna ali kao što smo vidjeli, za većinu njih možete plaćati direktnu podršku i to često zakladi [*Foundation*] ili tvrtki koja (uglavnom) stoji iza njih ili naravno lokalnoj tvrtki. U slučaju kada za kompletnu podršku plaćate lokalnu tvrtku, novac ostaje u vašem gradu/županiji/državi, od poreza, prireza, plaća radnicima te tvrtke i slično.
- Za Linuxe za koje se plaća podrška (primjerice RedHat, SuSe , ...) cijena je puno manja od drugih komercijalnih operativnih sustava, a uz tu cijenu dobivate i određenu osnovnu podršku, koja je prema mnogima [i meni],puno bolja,brža i kvalitetnija od onih “drugih”. Osim toga možete nadoplatiti i za dodatnu podršku ili jednostavno kupiti samo Linux uz osnovnu podršku, a za sve ostalo, podršku platiti lokalnoj tvrtki.
- Ostali programi su besplatni za osobnu i/ili komercijalnu upotrebu, za sve njih također možete plaćati podršku, po sličnom modelu kao i za Linux.
- Troškovi potencijalnog školovanja ljudi u vašoj tvrtki zbog prelaska na programe i sustave otvorenog koda i cjelokupne migracije na sustave otvorenog koda: kada se sve zbroji, moguće je nekoliko scenarija uz određene varijacije:
 - Troškove školovanja i migracije plaćate ili odradujete sami, početni troškovi su manji a dugoročni troškovi su višestruko manji
 - Troškove školovanja i migracije plaćate (lokalnoj tvrtki), početni troškovi su isti ili približni a dugoročni troškovi su višestruko manji
 - Troškove školovanja i migracije plaćate (lokalnoj i/ili inozemnoj tvrtki), početni troškovi su isti ili zanemarivo veći a dugoročni troškovi su manji.

Lista nekih od programa i sustava otvorenog koda, prema kategorijama, navesti ćemo samo kratko:

Operacijski sustavi, bazirani na Linuxu (tzv. Linux distribucije):

- **CentOS:** <http://www.centos.org/>
- **Debian:** <http://www.debian.org/>
- **Fedora:** <http://fedoraproject.org/>
- **openSUSE:** <http://www.opensuse.org/>
- **Slackware:** <http://www.slackware.com/>
- **Ubuntu:** <http://www.ubuntu.com/>
- ... i još nekoliko stotina drugih distribucija Linuxa https://en.wikipedia.org/wiki/List_of_Linux_distributions

Pogledajmo koje su to sve grane Linux distribucija, aktivne i danas:



Nadalje tu su i derivati **FreeBSD** Unixa :

- FreeBSD, <http://www.freebsd.org/>
- OpenBSD, <http://www.openbsd.org/>
- NetBSD, <http://www.netbsd.org/>

Pogledajmo i cijeli niz programa razvijanih kao programi otvorenog koda.

Neki od mrežnih servisa :

- **Apache HTTP Server** (Web poslužitelj), <http://httpd.apache.org/>
- **Lighttpd** (Web poslužitelj), <http://www.lighttpd.net/>
- **Nginx** (Web poslužitelj), <http://nginx.org/en/>
- **Apache Tomcat** (Java Servlet Container), <http://tomcat.apache.org/>
- ...

FTP poslužitelji:

- **FileZilla** Server, <https://filezilla-project.org/>
- **ProFTPD**, <http://www.proftpd.org/>
- **PureFTPD**, * <http://www.pureftpd.org/project/pure-ftpd>
- **vsftpd**, <https://security.appspot.com/vsftpd.html>
- **Libre SSL**, <https://www.libressl.org/>
- ...
- **OpenSSH** (SSH poslužitelj), <http://www.openssh.com/>
- **OpenVPN** (VPN poslužitelj), <http://openvpn.net/>
- **OpenNTPD** (NTP poslužitelj), <http://www.openntpd.org/>
- **OpenSMTPD** (SMTP poslužitelj), <https://www.opensmtpd.org/>
- **PowerDNS** (DNS poslužitelj), <https://www.powerdns.com/>
- **Unbound** (DNS), <https://www.unbound.net/>
- **BIND** (DNS poslužitelj), <http://www.isc.org/downloads/bind/>
- **Samba** (Windows CIFS/SMB share/domain), <http://www.samba.org/>
- **Squid Proxy Server** (Proxy poslužitelj), <http://www.squid-cache.org/>

Mail poslužitelji i groupware poslužitelji:

- **Postfix**, <http://www.postfix.org/>
- **Sendmail**, <http://www.sendmail.com/>
- **Kolab Groupware**, <https://kolab.org/>
- **Zimbra**, <https://zimbra.org/>
- **iRedMail**, <https://www.iredmail.org/>
- **Java Apache Mail Enterprise Server**, <https://james.apache.org/>
- ...

CMS i e-commerce Sustavi:

- **WordPress**, <http://wordpress.org/>
- **Joomla**, <http://www.joomla.org/>
- **Drupal**, <https://drupal.org/>
- **Magento**, <http://www.magentocommerce.com/>
- **OpenCart**, <http://www.opencart.com/>
- **MediaWiki**, <http://www.mediawiki.org/wiki/MediaWiki>
- **phpBB**, <https://www.phpbb.com/>
- ...

Sigurnost

- **ClamAV** (AntiVirus), <http://www.clamav.net/>
- **Snort** (IDS/IPS), <http://www.snort.org/>
- **Suricata**(IDS/IPS), <http://suricata-ids.org/>
- **SquidGuard**, <http://www.squidguard.org/>
- **KeePass Password Safe**, <http://keepass.info/>
- **TrueCrypt**, <http://www.truecrypt.org/>
- ...

Baze Podataka / Imenični servisi

- **MySQL** (SQL baza), <http://www.mysql.com/>
- **MariaDB** (SQL baza), <https://mariadb.org/>
 - **phpMyAdmin** (PHP Admin za MySQL bazu), <http://www.phpmyadmin.net/>
- **PostgreSQL** (SQL baza), <http://www.postgresql.org/>
- **OpenDJ** (LDAP), <https://forgerock.org/opendj/>
- **ApacheDS** (LDAP), <http://directory.apache.org/apacheds/>
- **OpenLDAP** (LDAP), <http://www.openldap.org/>
 - **phpLDAPadmin** (PHP Admin za OpenLDAP), <http://phpldapadmin.sourceforge.net/>
- ...

Integrirani Razvojni Alati

- **Eclipse**, <http://www.eclipse.org/>
- **Arduino**, <http://www.arduino.cc/>
- **NetBeans IDE**, <https://netbeans.org/>
- **Qt Creator**, <http://qt.digia.com/>
- **Lazarus**, <http://www.lazarus.freepascal.org/>
- **Aptana Studio**, <http://www.aptana.com/>
- **MonoDevelop**, <http://monodevelop.com/>
- ...

Programski Jezici

- **AWK**
 - GAWK (GNU AWK) <http://www.gnu.org/software/gawk/manual/gawk.pdf>
- C#, Visual Basic, ASP.NET 5 (by Microsoft), <http://blogs.msdn.com/b/dotnet/archive/2014/11/12/net-core-is-open-source.aspx>
- Erlang, <http://www.erlang.org/>
- Java, <http://www.java.com/>
- Perl, <http://www.perl.org/>
- Ruby, <https://www.ruby-lang.org/>
- Python, <http://www.python.org/>
- PHP, <http://php.net/>
- Scala, <http://www.scala-lang.org/>
- Swift (zamjena za Objective-C, by Apple, za OS X, iOS i Linux) , <https://developer.apple.com/swift/>
- Tcl/Tk, <http://www.tcl.tk/>
- ...

Uredski Alati i uređivači teksta

- LibreOffice, <http://www.libreoffice.org/>
- Stolno izdavaštvo (DTP) : Scribus, <http://www.scribus.net/>
- Dijagrami i sheme : Dia, <https://projects.gnome.org/dia/>
- Obrada teksta i DTP :
 - LaTeX : <https://www.latex-project.org/>
 - TeXstudio (GUI za LaTeX) : <http://www.texstudio.org/>
- Evince, <https://projects.gnome.org/evince/>
- HomeBank, <http://homebank.free.fr/>
- Zim, <http://zim-wiki.org/>
- FreeMind, <http://freemind.sourceforge.net/>
- XMind, <http://www.xmind.net/>
- NotePad++, <http://notepad-plus-plus.org/>
- Emacs, <http://www.gnu.org/software/emacs/>
- VIM, <http://www.vim.org/>
- Geany, <http://www.geany.org/>

Mrežni Alati

- Wireshark, <http://www.wireshark.org/>
- WinSCP, <http://winscp.net/>
- FileZilla, <https://filezilla-project.org/>
- PuTTY, <http://www.chiark.greenend.org.uk/~sgtatham/putty/>
- Zenmap/nmap, <http://nmap.org/zenmap/>
- Xming, <http://www.straightrunning.com/XmingNotes/>

Upravljanje Projektima

- Trac, <http://trac.edgewall.org/>
- Redmine, <http://www.redmine.org/>
- Jenkins, <http://jenkins-ci.org/>
- Bugzilla, <http://www.bugzilla.org/>
- Gerrit, <http://code.google.com/p/gerrit/>

Verzioranje koda i drugi programi

- TortoiseHg, <http://tortoisehg.bitbucket.org/>
- Mercurial, <http://mercurial.selenic.com/>
- Git, <http://git-scm.com/>
- ...
- WinMerge, <http://winmerge.org/>
- ...

Virtualizacija

- Proxmox, <http://www.proxmox.com/>
- Linux kernel virtualizacija (KVM), http://www.linux-kvm.org/page/Main_Page

- OpenStack, <https://www.openstack.org/>
- VirtualBox, <https://www.virtualbox.org/>

Sistemske Alati

- 7-Zip, <http://www.7-zip.org/>
- SystemRescueCd, <http://www.sysresccd.org/>
- Clonezilla, <http://clonezilla.org/>
- HardInfo, <http://hardinfo.berlios.de/>
- K3b, <http://www.k3b.org/>
- Brasero, <https://projects.gnome.org/brasero/>
- GParted, <http://gparted.sourceforge.net/>
- Wine, <http://www.winehq.org/>

... i tisuće i tisuće drugih programa.

Za malo veću listu programa, pogledajte stranicu: <https://www.opensource-osijek.org/wordpress/najpoznatiji-projekti/>.

Važno je razumjeti kako iza većeg dijela navedenih programa odnosno projekata stoji neka tvrtka, koja podupire njegov razvoj a i kojoj možete platiti za komercijalnu podršku. Naravno iza svih projekata stoji i povećana zajednica programera i korisnika, kojoj se također možete obratiti. Dodatno, postoje i treće tvrtke koje također pružaju komercijalnu podršku.

Postoje i naravno, *open source* projekti, za kojima se nije pojavio veliki interes, te se slabo ili gotovo nikako ne razvijaju, kao i oni čiji razvoj je prilično spor. Sve ovisi i o tome iz kojeg je tehnološkog područja projekt, te da li je ta uska zajednica spremna doprinijeti istom projektu ili ga samo koristiti. U svakom slučaju postoje razna granična područja ali srećom više je onih koji su se s godinama dokazali u radu.

S druge strane postoje i određeni sustavi koji su razvijani kao sustavi otvorenog koda, neki imaju paralelnu ili dijelom zasebnu granu razvoja s *open source* izvorom, a iza njih u svim slučajevima itekako stoje komercijalne tvrtke.

Neki od primjeru su:

- NAS/SAN sustavi:
 - **OpenFiler**, <https://www.openfiler.com/>
 - **FreeNAS** odnosno njegova komercijalna verzija, **TrueNAS**, <https://www.ixsystems.com/truenas/>
 - ...
- Router/Firewall/VPN/... :
 - **OpenWRT** odnosno komercijalni **DD-WRT** projekt <https://www.dd-wrt.com/site/index>
 - **pfSense**, <https://www.pfsense.org/get-support/>
 - ...
- IPS/IDS
 - **Snort**, <https://www.snort.org/> iza kojega sada stoji *Cisco*
 - **Suricata**, <https://suricata-ids.org/>
 - ...

... MALO UREDITI ---

Sumarizirajmo

U konačnici imamo mnoge prednosti razvoja programa i/ili sustava otvorenog koda:

- ulaže se znatno manje vremena i resursa u razvoj, testiranje i promidžbu jer na istom projektu može surađivati više tvrtki i pojedinaca, od koji, svako ulaže samo djelić radnog vremena.
- svatko može uložiti svoje ideje i svoj način razmišljanja u projekt, što rezultira kvalitetnijim konačnim rješenjem
- rasprave/planiranja su otvorenog tipa, i samim time moguće je brže/bolje uočavanje potencijalnih slabosti/problema u razvoju. Radi se po principu odabira najbolje ideje/rješenja, za svaku fazu razvoja
- poboljšanje kvalitete samog koda jer je moguće razvijati kod prema inženjerskim pravilima razvoja, koja se kod klasičnog razvoja uglavnom zanemaruju ("nema se vremena i slično"), programski kod postaje puno ljepši jer je svima vidljiv i dostupan ispravkama, komentiranju i optimizacijama
- porast kvalitete konačnog rješenja jer potencijalno postoji cijela vojska testera, raznih struka i pogleda na proizvod odnosno rješenje
- tvrtke, institucije i zaposlenici na direktan način podupirući neki *Open Source* projekt i radeći na njemu, razvijaju i svoje sposobnost a ovo je praktično besplatno školovanje i reklama za sve navedene
 - potraga za kvalitetnim kadrom se može skratiti, pogledom na aktivne programere i njihov konkretan rad na konkretnim *Open Source* projektima
 - Za Sveučilišta i Fakultete:
 - Sveučilišta i fakulteti koji su podupirali mnoge *Open Source* projekte su profitirala na mnogo područja:
 - od boljeg angažmana studenata
 - preko boljeg školovanja
 - do bržeg zapošljavanja studenata koji su radili konkretne projekte
 - te povećanja zanimanja za određeni Fakultet i Sveučilište, kao i rasta ugleda istih
 - Sveučilišta i Fakulteti će u dogledno vrijeme postajati sve više financijski neovisni te će morati optimizirati sve svoje troškove, u čemu *Open Source* programi i razvoj istih uvelike može pomoći
 - Iz mnogih *Open Source* projekata su izrasli i novi poslovi za sveučilišta i/ili fakultete te nastajale nove tvrtke
- ...

Mane razvoja klasičnih komercijalnih programa i sustava sa zatvorenim kodom:

- cilj da se u što kraćem vremenu napravi i proda određeni softver
- greške se znatno kasnije popravljaju i uočavaju, kada su već napravile neku štetu
- vrlo često se događa da je cijeli dizajn promašen od početka
- za profesionalno dokumentiranje najčešće se nema vremena, te se na kraju gubi puno vremena na proučavanje koda i "što je pisac htio reći"
- dodatne funkcionalnosti se samo ubacuju da se zadovolje potrebe kupca (prodaje), što kasnije dovodi do problema
- uglavnom ne postoji metodologija i plan razvoja pa se sve radi stihijski što vodi do problema koji nastaju tek kasnije u radu
- zbog nedostatka pravog razvoja softver postaje sve tromiji, nestabilniji, malo malo se mora optimizirati i krpiti
-

4.1 Tko sve koristi Linux i sustave otvorenog koda

Nakon, po nekima samo teorije, o prednostima upotrebe Linuxa i drugih sustava otvorenog koda, pogledajmo primjere upotrebe u praksi. Naime sve veći broj država i državnih agencija od Sjedinjenih Američkih Država, preko Europe do Japana, sve više koristi Linux kao i razne druge programe i sustave bazirane na principima otvorenog koda. Ne zaboravimo i sve veći broj tvrtki: od najvećih korporacija do malih tvrtki sa svega nekoliko zaposlenih.

Krenimo od upotrebe Linuxa na mjestima koja će vas možda i začuditi

Državne agencije i ministarstva :

- Ministarstvo obrane Sjedinjenih Američkih Država (U.S. Department of Defense):
 - Koriste RedHat Linux, prema nekim izvorima ovdje je za RedHat Linux najveći broj instaliran računala unutar jedne tvrtke ili institucije
- Mornarica Sjedinjenih Američkih Država : Flota nuklearnih podmornica (U.S. Navy Submarine Fleet):
 - Poznato je da koriste Linux baziran na Red Hat Linuxu
- Federalni sudovi (U.S. Federal Courts) u Sjedinjenim Američkim Državama
- Savezna uprava za civilno zrakoplovstvo (Federal Aviation Administration) Sjedinjenih Američkih Država
- Francuski parlament:
 - Koriste Ubuntu Linux sa OpenOffice paketom, instalirano je ~1.100 računala
- Španjolska (većina državnih institucija i agencija)
 - Napravili su svoju distribuciju Linuxa, baziranu na Debian Linuxu, naziva : LinEx
- Pošta Sjedinjenih Američkih Država (U.S. Postal Service):
 - Svi poslužitelji su migrirani na Linux : preko 900 Clustera sa Linux poslužiteljima, širom države
- Češka pošta :
 - Koriste SuSe Linux sa 4.000 Poslužitelja i 12.000 računala
- Njemačka : Grad Minhen, migrirao ~15.000 stolnih računala sa Microsoft Windows na Linux
- Nizozemski sustav za Ministarstvo unutarnjih poslova (Internet Research and Investigation Network (iRN)) :
 - Koriste Ubuntu Linux na ~2.200 računala
- Njemački nacionalni sustav za zapošljavanje (Bundesagentur für Arbeit)
 - Koriste OpenSuSe Linux na ~13.000 računala
- NASA (National Aeronautics and Space Administration), od poslužitelja i stolnih računala do drugih sustava (pr. Space Shuttle misija STS-83 je koristila Linux za kontrolu eksperimenata).
- Sustav napredne kontrole prometa u gradu San Francisco
- Japanski vlakovi tkzv “Bullet Trains” tj. “Shinkansen” uz sustav kontrole i nadzora vlakova i cijelog sustava, koristi Linux

Školstvo :

- Brazilsko školstvo:
 - Koriste Linux na : 520.000 Računala u 50.000 škola širom države
- Njemačka Sveučilišta :
 - Koriste SuSe Linux za poslužitelje i desktop računala u 33 sveučilišta, za 560.000 studenata
- Savezna Država Indiana (SAD), školski sustav :
 - Koriste Linux za 22.000+ studenata i učenika
- Italija, regija Bolzano : školski sustav sa 16.000 studenata i učenika
- Makedonija : školski sustav
 - Linux stolnih računala : 5.000 i 180.000 Ubuntu Thin Client računala
- Rusija : cijeli školski sustav i sve državne institucije :
 - Plan je nabavka ~700.000 stolnih računala i ~300.000 poslužitelja upogonjenih Linuxom i Open Source programima, baziranih na Baikal CPU (inicijalni plan je bio ARM arhitektura bazirana na ARM Cortex-A57 ali je napravljen zaokret i odabrana je MIPS arhitektura bazirana na : MIPS Warrior P5600) (6.mj.2015).
- Švicarska: pojedini kantonalni školski sustavi :
 - Koriste Linux s OpenOffice paketom : Ženevski kanton : ~9.000 računala
- ...

Neke od velikih tvrtki ili korporacija koje koriste Linux i OpenSource za poslovanje:

- Google : na poslužiteljima i stolnim računalima
- IBM : na poslužiteljima i stolnim računalima
- Panasonic : na poslužiteljima i stolnim računalima
- Cisco : na poslužiteljima i stolnim računalima
- Amazon : za skoro svaku komponentu poslovnog sustava se koristi Linux, kako su i sami izjavili : “everything that happens in them is driven by Linux”
- Virgin America : sustav za zabavu unutar zrakoplova “in-flight entertainment system”
- Peugeot :
 - Koriste: 20.000 Novell Desktop Linuxa i 2.500 SuSe Linux Enterprise Server
- Wikipedia
- Burza u New York-u (New York Stock Exchange) :
 - Koriste Red Hat Enterprise Linux
- Londonska burza
- Toyota :
 - Odjeli prodaje koriste Linux za oko 30. različitih sustava: od sustava za naručivanje dijelova, garancije, popravke
- ,

- Komunikacijski sustav i sustav elektroničke zabave unutar vozila (In-Vehicle-infotainment (IVI) systems) koriste Linux
- Facebook
- Twitter
- Siemens
- McDonalds
- Deutsche Bank
- Bank of America
- Bank of New York
- DreamWorks
- NTT Telekom grupa (Nippon Telegraph and Telephone Corporation) - vlasnik preko 30 telekom tvrtki širom svijeta
- Morgan Stanley, Goldman Sachs,
- ...

Osim navedenog i [većina](#) najsnažnijih superračunala u svijetu koristi *Linux*.

4.2 Tko još (pametan) razvija programe i sustave otvorenog koda

Dobro, mnoge državne institucije i tvrtke sve više koriste programe i sustave otvorenog koda ali tko još (pametan) od tvrtki razvija ovakve sustave?.

Navesti ćemo samo nekoliko velikih tvrtki koje su najviše uložile ali i ulažu u programe i sustave otvorenog koda:

IBM

- Uložio je nekoliko milijardi američkih dolara u servise i programe bazirane na Linuxu, kroz Linux Technology Center koji uključuje 300+ stalno zaposlenih Linux programera.
- Donira 40 milijuna američkih dolara u Eclipse (<http://www.eclipse.org/>) (12.5+ milijuna linija kôda).
- Ulaže u “Open Source Initiative” (<http://opensource.org/>).
- Donira u razvoj “Java-base relational database management system” (RDBMS) Apache Derby (<http://db.apache.org/derby>).

SUN Microsystems

- 1990. razvija programski jezik Java koji 1995.g. predstavlja javnosti te 2007.g. licencira pod GPL licencom (6.5 milijuna linija koda).
- 1999. kupuje njemačku tvrtku StarDivision za 73.5 milijuna američkih dolara i između ostaloga, s njom “StarOffice” kojeg kasnije Sun objavljuje kao OpenOffice pod LGPL i SISSL licencama (~10. milijuna linija koda).
- 2005. je razvio i pokrenuo GlassFish (aplikacijski server) kao open source projekt.
- kasnije 2005 godine kupuju tvrtku “Storage Tek” za 4.1 milijarde US\$ te 2008 objavljuje kôd “Sun Storage 7000 Unified Storage systems” koji je baza za razvoj ZFS-a koji je kombinacija datotečnog sustava i “volume nanagera”.
- 2005. objavljuje izvorni kôd kernel i network dijela operativnog sustava Solaris, potom pokreće OpenSolaris s istim kodom pod CDDL licencom, kasnije uključuje i DTRACE, ZFS, Solaris Containers i druge tehnologije (2+ milijuna linija koda).
- 2008. kupuje tvrtku MySQL AB sa MySQL bazom podataka za 1 milijardu američkih dolara. Kasnije izdaje MySQL pod GPL licencom.
- ...

Google

- 2005. kupuje tvrtku Android Inc. te s njom dobiva Android operativni sustav.
- 2007. izdaje Android pod Apache 2.0 licencom (10+ milijuna linija koda).
- 2008. razvija Google Chrome web preglednik te isti izdaje pod nazivom Chromium u open source (2+ milijuna linija koda).
- Razvija Chrome/Chromium OS.
- Razvija Google Web Toolkit (300.000+ linija koda).
- Google Summer of Code (GSoC) je godišnji događaj, prvi puta održan 2005. U njemu Google nagrađuje sa 5,000 američkih dolara (u 2013.) stotine studenata koji uspješno završe zadane open source projekte do kraja događanja.

RedHat

- Praktično živi od Open Source filozofije.
- Ukupni prihodi tvrtke :
 - U 2012 : 1.13 milijardi U\$,
 - u 2014 : 1.53 milijardi U\$,

- u 2015 : 1.79 milijardi US\$,
- u 2016 : 2.05 milijardi US\$
- u 2017 : 2.4 milijarde US\$
- **Red Hat Enterprise Linux (RHEL)** je dostupan samo kroz pretplatu, koja uključuje pristup softveru (ISO image) i zakrpama te raznim nivoima tehničke podrške (ovisno o vrsti pretplate). Proizvod se sastoji većinom od programa koji se distribuiraju kroz open source licence.
- Cijeli kôd za sve programske pakete RedHat Linuxa je javno dostupan: ⇒ CentOS Linux koristi isti taj kôd za izradu svoje distribucije Linuxa.
- razvili su (i napisali) ~12% od cijelog Linux [kernel](#) kôda (prema broju linija koda), na kojem su bazirani svi Linuxi.
- JBoss (aplikacijski Server) (2+ milijuna linija koda), izdan pod LGPL licencom.
- Red Hat sponzorira Fedora projekt (*community* verziju Red Hat Enterprise Linuxa), a u novije vrijeme i **CentOS** projekt.

Microsoft

- Microsoft je na 21. mjestu po sudjelovanju u razvoju Linux kernela (~1% udjela, prema broju linija koda).
- Shvaća prednosti razvoja prema principima Open Source: <http://opensource.mscommunity.hr/> :“Microsoft Hrvatska 5. lipnja 2013. organizira prvu Microsoft Open Source konferenciju u suradnji sa Sveučilištem u Zagrebu, koja je namijenjena upravo vama. Microsoft je kao tvrtka proširio svoju filozofiju i sve više pridonosi open source zajednicama tako da promiče developersku suradnju kako bi se lakše razvijala i održavala interoperabilna IT rješenja.”
- Početkom 2015 objavljuje “.NET Core”, u open source-u. u to je uključeno:
 - C# i Visual Basic
 - Visual F# Tools
 - ASP .NET 5
- Nekoliko mjeseci nakon otvaranja “.NET Core”-a, open source zajednica pronašla je i riješila na stotine bugova te razvila nove funkcionalnosti.

Apple

- Sredinom 2015 objavljuje kako će programski jezik *Swift* odnosno *Swift 2*, koji je nasljednik ili više zamjena za *Objective-C* biti u potpunosti objavljen kao otvoreni kod. Programski jezik *Swift* će se koristiti za razvoj aplikacija za **OS X** i **iOS**, te će raditi na Linuxu. Ovo znači kako će od sada biti moguće razvijati aplikacije za **OS X** ili **iOS** na *Linuxu*.

Što je s malim tvrtkama ?

Osim navedenih velikih igrača, veliki broj malih tvrtki je osnovan na bazi poslovnog modela koji obuhvaća neke (ili sve) segmente komercijalne upotrebe *Open Source* programa i/ili sustava:

- Pružanja tehničke podrške za kupce,
- Razvoja i optimizacije prema potrebama klijenata (kupaca),
- Razvoja zasebnih komercijalnih *Close Source* ili *Open Source* aplikacija za postojeći *Open Source* program ili sustav,
- Integracija sustava kod kupca,
- ...

4.3 Filmovi i literatura koju preporučujemo

Za sve koji žele vidjeti komentare i iskustva ljudi koji su najviše doprinjeli razvoju *Open Source* inicijative, iz prve ruke, preporučujemo da pogledaju dokumentarni film : **Revolution OS** : IMDB : <http://www.imdb.com/title/tt0308808/>

U dokumentarcu nam svoja iskustva govore :

- Richard M. Stallman (Programer, tvorac *Free Software Movement* i *GNU* projekta)
- Linus Torvalds (tvorac Linux kernela)
- Eric Steven Raymond (Programer UNIX programa, kasnije i GNU/LINUX i autor knjige “**The Cathedral and the Bazaar**”, suosnivač “Open Source Initiative (OSI)”)
- Bruce Perens (tvorac definicije *Open Source*, Programer i suosnivač *Open Source Initiative (OSI)*)
- Larry Augustin (Suvlasnik/Osnivač tvrtke *VA Linux Systems*)
- Michael Tiemann : suosnivač je tvrtke *Cygnus Solutions* koja je razvila i *Cygnin*, autor je GNU C++ prevoditelja (Engl. Compiler), GNU C prevoditelja i GNU *Debugger*, bio je predsjednik *Open Source Initiative*, tehnički direktor u tvrtki *RedHat*, član savjetodavnog odbora za *GNOME* projekt, član je odbora za *Embedded Linux Consortium* , , ...
- ...

Dodatno preporučujemo i :

- dokumentarna mini serija : *The Triumph of the Nerds: The Rise of Accidental Empires*, IMDB <http://www.imdb.com/title/tt0115398/>
- dokumentarni film *Secret History of Hacking*, IMDB http://www.imdb.com/title/tt2335921/?ref =nv_sr_1
- dokumentarna mini serija : *Discovery Channel – Rise of the Video Game*, <http://richardhuskey.com/2012/10/03/rise-of-the-video-game/>

Ako se bavite razvojem softvera (ne nužno i *Open Source*), preporučujem da pročitate knjižicu (~35 stranica) koja je promijenila stavove mnogih tvrtki o *Open Sourceu*. Autor knjige je Eric Steven Raymond, koji je od sredine 1980 radio na razvoju raznih programa za Unix:

- Net Hack (igra),
- Razne komponente *Emacs*,
- xlife
- kasnije je radio i na igri *The Battle for Wesnoth*,
- ..

On je praktično cijelo desetljeće radio na razvoju *Unix* programa ustaljenim i stabilnim metodologijama razvoja (poput Katedrale kako i sam kaže) a odatle i naziv knjige. Nakon toga krenuo je u avanturu razvoja programa metodologijom koju je preporučio Linus Torvalds, a koju on naziva modelom tržnice (engl. Bazaar). Dakle razvoja u kojem sudjeluje veći ili veliki broj ljudi, od kojih svatko često objavljuje promjene koda, ... ali o tome u knjizi. On nam opisuje svoja iskustva i probleme uz rješenja, na koje je nailazio u razvoju e-mail klijenta (Fetchmail) na kojem je počeo raditi.

Njegova knjiga je i po mnogima uzrokovala odluku tvrtke *Netscape Communications Corporation* koja je tvorac, u to vrijeme, najboljeg Web preglednika, da kompletan kôd *Netscape Navigator* preglednika, daju u *Open Source* i otvore *Mozilla* fondaciju. Taj potez je kasnije otvorio vrata za razvoj današnjih preglednika baziranih na *Mozilla Firefox* pregledniku koji je razvijan unutar fondacije *Mozilla*.

Knjižica se zove : **The Cathedral and the Bazaar** , možete ju potražiti na internetu : njena izvorna verzija se može pronaći na : <http://www.unterstein.net/su/docs/CathBaz.pdf> ⇒ Postoji i proširena verzija od stotina stranica, koja je naravno komercijalna i zanimljiva ali mislim da je za razumijevanje svega dovoljna i ova izvorna verzija.

5 Primjeri razvoja sustava otvorenog koda

U ovom poglavlju govoriti ćemo o tvrtkama čiji poslovni model počiva na razvoju programa otvorenog koda. Kako je uopće moguće živjeti od razvoja sustava otvorenog koda. I da li je to stvarno moguće, u našem okruženju, vidjeti ćete u ovom poglavlju.

... U IZRADI ...

6 Kako razvijati sustave otvorenog koda i živjeti od toga

... U IZRADI ...

6.1 Dodatne informacije i primjeri

Još jedan od primjera negativnog korporativnoga ponašanja, navodi se kao dokaz, u kojemu *Microsoftov* manager za razvoj jezika C++ u internom dopisu Billu Gatesu 21. veljače 1997. piše:

“

Windows API je toliko sveobuhvatan i toliko učinkovit da bi većina nezavisnih proizvođača softvera bila luda da ga ne koristi. Toliko je duboko ugrađen u izvorni kod mnogih Windows aplikacija da bi troškovi prilagodbe za drugi operacijski sustav bili ogromni. Upravo su ti troškovi prelaska dali našim korisnicima strpljenje da ostanu uz Windowse usprkos svim našim pogreškama, bagovitim pogonskim programima, visokom ukupnom trošku korištenja (TCO), našem povremenom nedostatku vizije i usprkos mnogim drugim teškoćama. [...] Kupci neprestano procjenjuju druge desktop platforme, ali zamjena bi tražila previše napora i truda, tako da se radije nadaju da ćemo poboljšati Windowse nego da ćemo ih natjerati na promjenu. Ukratko, bez ove ekskluzivne franšize zvane Windows API odavno bismo bili mrtvi. [...] Microsoftov aplikacijski softver omogućuje lock-in korištenjem vlasničkih datotečnih formata. Microsoft Outlook koristi vlastiti, nedokumentirani format za spremanje podataka. Sadašnje verzije Microsoft Word uvode

novi format MS-OOXML. Konkurencija radi toga lakše kreira dokumente kompatibilne s Microsoft Officeom, da se ubuduće smanji lock-in. Microsoft je objavio potpun opis datotečnih formata za ranije verzije Worda, Excella i PowerPinta u veljači 2008.

”

Citat je iskorišten u odluci Europske komisije u kojoj osuđuje Microsoftovu poslovnu praksu.

Izvor informacija: <https://sysportal.carnet.hr/node/1644> Autor: Aco Dmitrović.

7 Izvori informacija

(1) **Kratka povijest UNIXa - Od UNICSa do FreeBSDa i Linuxa:**

<https://www.opensource-osijek.org/knjige/kratka-povijest-UNIXa-od-UNICSa-do-FreeBSDa-i-linuxa.pdf>

(2) **Uvod u Linux i Linux napredno:**

http://haw.nsk.hr/arhiva/vol2018/6736/70809/www.opensource-osijek.org/dokuwiki/export/xhtml/wiki:knjige:uvod_u_linux.html