



The GIMP 2.2.x:

Zvezda je rođena

Distribucije:

Od nule do GNU/Linuxa
Malo, manje, najmanje

Softver:

gDesklets

Hardver:

Multimedijalne tastature
Drajveri, drajveri...

Radionica:

Gimp i crvene oči
Uvod u Piton (Python)
Srebrna Lisica

Pravna klinika:

GNU i zakon

Legum servi summus ut
liberi esse possimus - deo 1.

Distro



LFS

Kako napraviti GNU/Linux
iz ništavila.

Software



GIMP

Istorija jednog od najboljih
programa za obradu
rasterske grafike.

Hardware



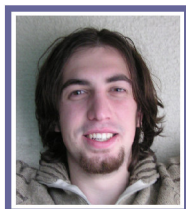
Modemi

Uputstvo kako da
pokrenete svoj softverki
modem pod GNU/Linux i
*BSD sistemima



GNUzilla Februar 2005!

OHRABRUJUĆE DELUJE POGLED NA BROJ PREUZIMANJA GNUZILLE JANUAR 2005. LEPA SATISFAKCIJA ZA DALJI RAD ALI I OBAVEZA DA SE OPRAVDA VAŠE POVERENJE I ISPUNE VAŠA OČEKIVANJA VEZANA ZA BUDUĆNOST.



Pre nego što se osvrnemo na prethodni mesec i kažemo koju reč o novom broju, želimo da izrazimo zahvalnost zajednici na poverenju koje nam je ukazala. Napori koje smo uložili da GNUzillu pretvorimo u realnost su i više nego urodili plodom, što je potvrdilo činjenicu da je našoj sceni ovakav medij više nego dobrodošao. Naročito je vredno pomena i to da je naš rad naišao na odjek u zemljama bivše Jugoslavije, pre svega Makedoniji i Hrvatskoj. Dobre kritike, koje se, pre svega, odnose na inicijativu i dizajn, su se mogle čuti i u inostranstvu.

Novosti u drugom broju GNUzille se pre svega ogledaju u tehničkom unapređenju časopisa. Drugi broj sa sobom donosi poboljšanje vizuelnog identiteta kao i smanjenje veličine PDF fajla u obliku kog se časopis distribuira. Dobili smo puno kritika i molbi, kako na forumu tako i preko e-pošte, koje se tiču veličine PDF datoteke januarskog broja (oko 5MB), tako da smo ovoga puta napravili dve verzije – jednu sa odličnim kvalitetom slika (4.3MB) i jednu sa slikama malo lošijeg kvaliteta (2.6MB). Čitaoci sa modemskim vezama će ovo naročito znati da cene.

Pored tehničkih izmena, uveli smo dve nove rubrike u časopis – Radionica i Pravna klinika. Ovim putem želimo da poželimo dobrodošlicu Slobi (bio je uključen od početka projekta, ali nije pisao za prvi broj) i Milošu.

Nadamo se da će se u budućnosti još kreativnih ljudi priključiti časopisu i svojim doprinosom pomoći da GNUzilla bude još bolja. Uživajte!

~ Ivan Jelić

Redakcija:

Ivan Jelić
Ivan Čukić
Marko Milenović
Petar Živanić
Danilo Šegan
Aleksandar Urošević
Srđan Anđelković
Slobodan Kovačević

Saradnici:

Miloš Popović

Priredivač:

Mreža za Slobodan Softver
www.fsn.org.yu

Kontakt adresa:

gnuzilla@fsn.org.yu



Sadržaj:

Uvodna reč	2
Sadržaj	2

Distribucije

Od nule do GNU/Linuksa	3
Malo, manje, najmanje	5

Softver

gDesklets	6
Zvezda je rođena	9

Hardver

Multimedijalne tastature	13
Drajveri, drajveri...	15

Stalne rubrike

Radionica

Gimp i crvene oči	21
Uvod u Piton (Python)	23
Srebrna Lisica	25

Pravna klinika

Legum servi summus ut liberi esse possimus – deo 1.	27
---	----

GNUzilla

Sav materijal u ovom časopisu je objavljen pod Free Documentation License. Autorska prava pripadaju autoru.

Od nule do GNU/Linux

Predstavljanje distribucije Linux From Scratch

PRVI PROBLEM ZA NEKOGA KO ŽELI DA SE ORIJENTIŠE NA GNU/LINUX PREDSTAVLJA IZBOR DISTRIBUCIJE. OBIČNO NOVAJLIJE NA RAZNIM MESTIMA POSTAVLJAJU PITANJE NA KOJE NE POSTOJI SAVRŠEN ODGOVOR: „KOJA DISTRIBUCIJA JE NAJBOLJA?“. IPAK, NAJREALNIJI ODGOVOR BI BIO „TVOJA DISTRIBUCIJA“. ZAGRIZITE JABUKU!



Ako ste nekada poželili da distribuciju Linuxa koju koristite prilagodite svojim potrebama, verovatno ste naišli na prepreku tome — ili zbog straha da nešto ne pokvarite, ili zbog neznanja šta, kako i kada da uradite. Još ako tek planirate da probate (i na dalje koristite) Linux, verovatno vam nikad i nije palo na pamet nešto tome slično. Krojenje sopstvene distribucije zvuči kao težak, dugotrajan i rizičan posao; a kao najbolnije, nameće se pitanje da li će posle svega to uopšte raditi. Pre par godina je Žerard Bikmens (*Gerard Beekmans*) došao na sličnu ideju, i nakon kompleksnog pokušaja došao do rešenja — distribucije po svojoj meri. Zbog suštine pravljenja, nazvao je ovaj posao jednostavno – *Linux From Scratch*.

LFS je knjiga koja postupno opisuje korake koje treba preduzeti da bi se izgradila sopstvena distribucija unapređivanjem, podešavanjem, prevođenjem i instaliranjem paketa iz izvornog koda.

Ali da počnemo od suštine. Uporedićemo LFS sa kompletnim projektom za izgradnju kuće. Za kuću imate projekat sa predračunom potrebnog materijala i nacrtima gde, šta i kako treba ugraditi. Kod LFS-a imamo knjigu u kojoj su popisani svi potrebni paketi i njihove popravke (pečevi), sa adresama odakle se isti mogu preuzeti. Tu se takođe nalazi kompletan i detaljan opis šta sa kojim paketom i kojim redosledom treba raditi.

Knjiga je podeljena na tri celine, i to: Uvod, Pripremanje za izgradnju i Izgradnja LFS sistema. U okviru ovih celina postoji ukupno sedam poglavlja.

1. Uvod

U uvodnom delu je ukratko opisano šta će se u kom poglavlju raditi, izmene postojećih i dodavanje novih paketa potrebnih za izgradnju LFS sistema, izvori pomoći i potrebnih paketa, uputstvo kako postavljati pitanja u slučaju potrebe, koji su najčešći problemi prilikom podešavanja paketa i sistema.

2. Pripremanje nove particije

U ovom poglavlju je opisan način i preporučena struktura particija diska za novi LFS sistem, način pravljenja sistema datoteka i montiranja novonapravljenih particija.

3. Materijal: Paketi i nadogradnje

Ovde su navedeni svi paketi koje je potrebno pripremiti, kao i koje su sve nadogradnje neophodne a koje preporučljive i za koje pakete.

Objašnjeno je zašto i gde treba smestiti sve pripremljene pakete za kasnije korišćenje u izgradnji.

4. Završna priprema

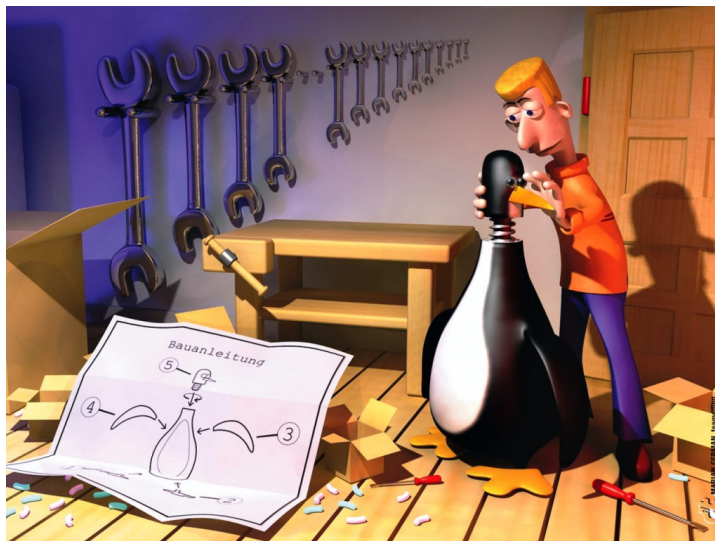
U ovom delu je navedeno koje systemske promenljive treba postaviti kao i koje nove direktorijume treba napraviti. Objasnjeno je i zašto pripremanje LFS-a ne treba raditi kao root već kao korisnik *lfs* i član grupe *lfs*, i kako ih napraviti. Takođe, naveden je način merenja vremena potrebnog za izgradnju pojedinačnih paketa, odnosno kompletnog sistema, pošto korisnici pre početka izgradnje nečega žele da znaju koliko će vremena utrošiti za taj posao. Prisutan je i opis kompleta za testiranje prevedenih i pripremljenih paketa.

5. Izgradnja privremenog sistema

Ovde je opisan način pripremanja privremenog sistema na korisničkom sistemu (distribuciji koju već koristite) za izgradnju konačnog LFS-a. Za ovo je pre svega potreban kernel 2.6.2 ili noviji. U ovom poglavlju se konačno prelazi na konkretan rad — izgradnju osnove prevođenjem osnovnih paketa u dva prolaza (prvo se pojedini alati prevode jednom, korišćenjem alata sa glavnog sistema, a drugi put se prevode korišćenjem tih novih verzija, zbog kompatibilnosti sa novim sistemom). Svi ovi paketi se smeštaju u privremeni */tools* direktorijum na LFS particiji.

6. Izgradnja LFS sistema

Kao što se i može pretpostaviti, u ovom poglavlju je detaljno opisan način montiranja potrebnih virtuelnih sistema datoteka, prelaska u pripremljeni privremeni LFS sistem *chroot* naredbom, postavljanja odgovarajućih prava pristupa i pravljenjem potrebne strukture direktorijuma i simboličkih veza. Opisan je način pravljenja potrebnih identifikacionih datoteka (*passwd*, *group*) kao i blok uređaja (*/dev*). Nakon ove pripreme prelazi se na dugo očekivano prevođenje paketa za novi LFS sistem. Ovaj proces je najdugotrajniji i možda najrizičniji, jer od njegovog uspešnog izvođenja zavisi stabilnost i funkcionalnost budućeg sistema. Obavezno je testiranje paketa *glibc* (koji pored kernela predstavlja osnovu GNU/Linux sistema), *binutils*, *gcc*, a preporučljivo je za *coreutils*. Tu je i opis i struktura datoteka */etc/syslog-ng* i */etc/initab*.



7. Podešavanje pokretačkih skripti

LFS je baziran na *SysV* inicijalnim skriptama posebno prilagođenim za ovaj sistem. Ove skripte dolaze u *LFS-Bootscripts* paketu i instaliraju se pojedinačno za svaki paket za koji je potrebno, a što je navedeno u samom postupku instalacije paketa. U ovom je poglavlju opisan način funkcionisanja ovog načina iniciranja Linux sistema, podešavanje vremenske zone, konzole, ulazne opisne datoteke (*/etc/inputrc*), skripti za opisivanje sistemskih promenljivih *BASH*-a, podešavanja mrežnih datoteka (*/etc/hosts*, */etc/sysconfig/network-devices* i */etc/resolv.conf*).

8. Pravljenje LFS sistema butabilnim

Vreme je da se definiše struktura tvrdog diska LFS sistema (*/etc/fstab*), pripremanje kernela i programa za učitavanje sistema (*boot loader*). LFS koristi *GRUB*.

9. Kraj

Na samom kraju pravljenja sopstvene distribucije Linuxa, ostalo je još definisati verziju (*/etc/lfs-release*), eventualno se prijaviti kao još jedan korisnik koji je uspešno završio pravljenje LFS-a, i na samom kraju je ostalo još da se restartuje računar.

ŠTA SADA?

Ako je izgradnja LFS-a protekla bez problema, ili ste probleme odmah otklonili, pravilno ste podesili loader, obrećete se u vašem no-

vom ličnom Linuxu – konzoli. I šta sada?

Verovatno se pitate šta ste dobili za sav trud uložen u izgradnju LFS-a? Vreme je da vam skrenemo pažnju da ovo nije kraj. Nakon uspešno završenog postavljanja LFS-a predstoji nešto složeniji proces izgradnje korisničkog okruženja — X i drugih servera, korisničkih uslužnih programa, radnog grafičkog okruženja, alata koji vam trebaju za svakodnevni rad, klijenata za mrežnu komunikaciju i drugo. Jednom rečju, dobili smo čistu, brzu i stabilnu platformu na kojoj možemo da do-

gradimo pakete po volji. Po onoj analogiji sa početka priče, kuća je završena, postavljen je krov, omalterisana je, postavljena elektro i vodovodna instalacija. Sada ostaje onaj možda dosadniji deo: krečenje, kupovina nameštaja, uređivanje enterijera, sređivanje dvorišta. Kad završimo sa tim, sledi uživanje u novom toplom domu.

Ovaj se proces izvršava po knjizi *BLFS (Beyond Linux From Scratch)*, koja sada neće biti opisana već vam ostavljamo da je sami prelistate i vidite šta vam se nudi, jer je specifična – ponuđeno je više modela od kojih treba izabrati onaj koji najbolje odgovara korisniku. Recimo, opisana je izgradnja okruženja Gnom i KDE, a na vama je da odlučite hoćete li praviti i kasnije koristiti oba ili samo jedno od njih.

Korisna adresa:

<http://www.linuxfromscratch.org>

~ Aleksandar Urošević

MenuetOS

Malo, manje, najmanje

MenuetOS – Operativni sistem za Asembli programere

Koliko puta ste pogledali staru gomilu disketa na vašem stolu i pitali se zašto ih uopšte čuvate? Mali medijum za čuvanje podataka kojeg je pregazilo vreme. No, možda im se još uvek može naći primena...ili bar jednoj od njih.



Prosečnom korisniku se na pomen operativnog sistema pred očima javlja glomazan softver koji zahteva čitav kompakt disk. a onda su jednog dana ljudi okupljeni oko projekta Menuet OS, koristeći Asembli programski jezik napisali nešto što pleni izgledom a staje na jednu disketu. Menuet OS predstavlja pravo iznenađenje za sve one koji nađu dovoljno vremena da preuzmu arhivu od 1,4MB, stave je na disketu i potom pokrenu svoj sistem uz pomoć iste.

Ovaj operativni sistem u sebi nosi podršku 32-bitnu X86 Asembli programiranje. No, sam OS nema korene u UNIX ili POSIX standardima, već je apsolutno nezavistan operativni sistem izdat pod OJL (GPL).

Menuet sa sobom nosi sve one divne stvari koje se kod jednog operativnog sistema očekuju: multitasking, multithreading, grafičko okruženje koje podržava rezolucije do 1280x1024 za 16 miliona boja, TCP/IP, PPP, drajvere za mrežne kartice (automatski je detektovao podešavanja za moju mrežu), mrežne aplikacije za rad sa FTP, HTTP, SMTP i POP3 serverima, IRC, HTTP, NNTP, TFTP klijente, mogućnost menjanja izgleda grafičkog okruženja...

Menuet OS možete preuzeti sa matične prezentacije www.menuetos.org. Autori su nam omogućili dva načina stavljanja Menueta na

disketu. Prvi način je namenjen Windows™ korisnicima i podrazumeva pokretanje EXE datoteke koja vrši kopiranje podataka na disketu. Drugi način je namenjen korisnicima UNIX-sličnih sistema, i podrazumeva preuzimanje IMG datoteke sa servera i kopiranje iste na disketu putem komande: `dd if=mso0xx.img of=/dev/fd0`.

Kada ste uspešno pripremili disketu, potrebno je da ponovo pokrenete svoj sistem, ali uz pomoć diskete. Menuet OS će se učitati i posle nekoliko postavljenih pitanja vezanih za vašu grafičku karticu i miša (toliko jednostavnih da ih ovde neću ni pominjati) sistem će se učitati i pred vama će "eksplozirati" hvale vredno grafičko okruženje.

Naravno, nemojte očekivati da MenuetOS može zameniti vaš kompletni operativni sistem. Pored lepe grafike, sistem u sebi nosi niz uslužnih programa za korišćenje Interneta, slušanje muzike ili zabavu. U sistem je uključena kompletna podrška za asembli 32-bitno programiranje tako da predstavlja pravu stvar za sve one zainteresovane za ovaj jezik. No, to svakako nije dovoljno za svakodnevni rad. I sami autori ovaj projekat smatraju hobi-jem.

Bez obzira na vaše afinitete prema ovom ili onom operativnom sistemu, toplo vam preporučujem da se poigrate MenuetOS-om. A što se disketa tiče: odstranite višak, ali ipak zadržite bar jednu.

~ Marko Milenović

Samo ulepšanje ili nešto više?

gDesklets

gDesklets omogućava da pokrenete male programe (aplete), koji imaju dve funkcije: da ukrase desktop i da pri tome imaju neku namenu.

GDESKLETS JE ONO ŠTO BI SVAKO VOLEO DA VIDI NA SVOM DESKTOPU. ANALOGNI SAT, ZAUZETOST DISKA PO PARTICIJAMA, TEMPERATURU PROCESORA I GRAFIČKE KARTE, KALENDAR, ČAK I NIVO MASTILA U KETRIDZIMA HP ŠTAMPAČA. MOGUĆNOSTI SU PRAKTIČNO NEOGRANIČENE, I SVE ŠTO ZAMISLITE VEĆ POSTOJI ILI ĆE POSTOJATI U BUDUĆNOSTI.

U osnovi gDesklets se sastoji iz tri dela: srca gDesklets sistema koje predstavlja daemon pokrenut u pozadini, Senzor koji obrađuje podatke i reaguje na akcije korisnika (u stvari to je modul u Pitonu) i Display - ono što vidite na desktopu.

Instalacija

Instalacija nije teška i na osnovu mog iskustva lako prolazi na većini distribucija. Najpre treba instalirati Python 2.3. Svaka Linux distribucija dolazi sa nekom verzijom Pitona tako da je po svoj prilici on već instaliran na vašem računaru. Odmah zatim treba instalirati sledeće pakete:

```
pygtk
pyorbit
gnome-python
swig (nije potrebno za
gDesklets 0.33+)
```

Ukoliko budete imali problema prilikom kom-

pajliranja dotičnih paketa, preporučujem da pronadete već pripremljene pakete (rpm, deb, tgz) za vašu distribuciju. Oni sigurno postoje i to je najlakši način rešavanja ovih zavisnosti. Treba još znati da neki deskleti zahtevaju i dodatne pakete. Tako na primer za čuveni starterbar treba još instalirati i paket:

```
pyxdg
```

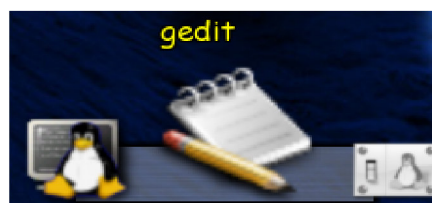
Nakon toga sa zvaničnog sajta [gDesklets](#) projekta treba preuzeti najnoviju verziju i instalirati je tradicionalnim komandama

```
$ ./configure && make
# make install (opciono checkinstall)
```

Takođe, u isto vreme je potrebno preuzeti i neke desklete sa iste lokacije. Nakon svih ovih instalacija potrebno je pokrenuti

```
$ gdesklets shell
```

i pred vama se otvara pozor u kom, pored osnovnih, postoje i složeni podešavanja. Za instalaciju novih deskleta potrebno je slediti File > Install package. Ukoliko sve



bude u redu trebalo bi da dobijete poruku u kojoj se kaže da je desklet uspešno instaliran. Sada još samo da ih pokrenemo - obeležite desklet koji želite pokrenuti i File > Run seleded display i to bi bilo to. Postoji mogućnost pokretanja gDesklets prilikom pokretanja X-a. Za ovo postoji nekoliko načina, a najjednostavniji bi bio: Application > Desktop preferences > Advanced > Sesion i na kartici Startup programs treba kliknuti na Add i otvoriti

/usr/local/bin/gdesklets start. Ili možete dodati sledeću liniju (kao poslednju) u vašu ~/.xinitrc datoteku

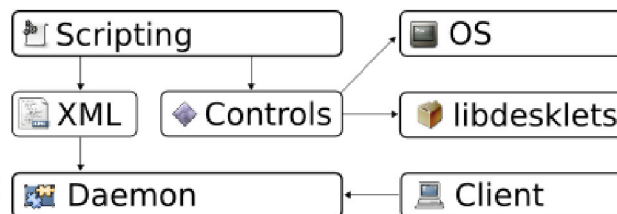
```
gdesklets start
```

Kako sve funkcioniše?

Na početku smo spomenuli od čega se sastoji ovaj sistem, a sada ćemo videti kako sve to funkcioniše.

Rekli smo takođe da je daemon (samostartujući program) srce gDesklets sistema. On se pokreće u pozadini i ima ulogu da omogući komunikaciju sa klijentima. Klijenti šalju instrukcije da-

monu i on ih izvršava (na primer, otvaranje novog apleta). Korisnički interfejs (UI) apleta, to jest ono što korisnik vidi i sa čim vrši interakciju, je definisan pomoću ADL, moćnog XML baziranog jezika. Tako definisani korisnički interfejes se može dalje poboljšavati sa takozvanim inline skripama. To su ustvari Piton skripte koje su napisane direktno u XML kod ili mogu biti pozivane iz XML koda i izvršavane u takozvanom secure sandbox-u. Na ovaj način se može pristupiti naizgled nepristupačnim podacima, na primer informacijama o hardveru itd. To je ustvari kontrolisan



način pristupanja podacima izvan sandbox-a, a za to se koristi biblioteka libdeskelts.

Delovi deskleta

Svaki desklet se sastoji iz dva dela: senzor (sensor) i displej (display). Korisnički interfejs je definisan u display datoteci, dok se senzor koristi za obradu podataka. Drugim rečima displej definiše izgled, dok senzor definiše mogućnosti jednog deskleta. Za pravljenje novih deskleta nije potrebno nikakvo znanje Piton programskog jezika, sve što treba je da poznajete ADL (XML) i koristite već postojeće senzore. Suprotno tome, pisanje novi senzora zahteva poznavanje programiranja u Pitonu, mada ni to nije nije teško. Posle instalacije gDesklets sistema pojavljuje se direktorijum ~/.gdesklets i u njemu se nalaze datoteke i direktorijumi potrebni za rad. U direktorijumu Controls nalaze se kontrole koje je korisnik instalirao, u direktorijumu Display nalaze se deskleti, a u direktorijumu Sensors senzori koje je korisnik instalirao.



ADL

ADL (Applet Description Language) je kako mu i samo ime kaže, alat (jezik) za opis deskleta. Baziran je na XML-u ali tako dizajniran da bude fleksibilan i ne previše komplikovan. On podržava inline skripte koje mogu menjati svojstva deskleta u toku izvršavanja. Glavne odlike ovog jezika su da je čitljiv korisniku, nezavisan od rezolucije monitora i bogat mogućnostima. Kako svako učenje počinje od nekog "Hello World" programa, tako ću i ja ovde početi od "zdravo svete" deskleta. Za ovo će vam treba neki desklet sa svojim senzorom. Otvorite njegovu .display datoteku u nekom uređivaču teksta (najbolje gedit), napišite sledeće:

```
<?xml version="1.0" encoding="UTF-8"?>
<display window-flags="sticky" bg-color="white">
  <label value="Zdravo svete!" font="Serif 20" color="red"/>
</display>
```

Pokrenite ga sa gDesklets sistemom, i dobićete vaš prvi desklet na desktopu. Svakom ko zna da čita enleski jezik ovaj kod je jasan sam po sebi. Ovo je statički desklet koji ne koristi senzor i na desktopu ispisuje "Zdravo svete". U redu, šta će nam desklet koji ne koristi senzor? Hajde da se time pozabavimo. Senzor je izvor podataka za desklet i nema ograničenja koliko senzora jedan desklet može da koristi, čak jedan senzor može pozvati više puta. Svaki senzor u displej (.display) datoteci mora imati svoj jedinstveni ID. Evo kako ćemo pozvati jedan senzor:

```
<sensor id="s" module="Time"/>
```

U ovom slučaju smo koristili atribut modul da bi pozvali senzor Time. Ukoliko senzor ne postoji ili ga gDesklets ne pronalazi dobićete poruku sa greškom. Sada, pošto smo dodelili senzoru njegov ID (u našem slučaju "s") možemo mu pristupati, na primer watch atributom, na sledeći način:

```
<label watch="value=s:out">
```

Ovde je potrebno da budemo pažljivi i da ne dodeljujemo osobine koje nemaju smisla. Dodavanjem osobine boja proizvoljnom tekstu dovešće do greške. Takodje je moguće više puta ga povezati na sledeći način:

```
watch="value=s:text,
color=s:color, font=s:font"
```

s tim da se razdvajanja vrše zapetom. Ja ne znam Piton, jel mogu da koristim senzore? Naravno, moj prvi savet je da naučite Piton



jer nije teško. Korisnici koji znaju neki drugi jezik ili znaju da pišu bash skripte mogu da koriste specijalni

senzor koji se naziva (External Sensor). On dolazi sa gDesklets sistemom i omogućava da napišete desklet u bilo kom jeziku, ali ste ograničeni samo na neinteraktivne desklete (aplete). Evo jednog lakog primera:

```
<?xml version="1.0" encoding="UTF-8"?>
<display window-flags="sticky, below" bg-color="white">
  <sensor id="s" module="External, tail /var/log/messages"/>
  <label
watch="value=s:value"/>
</display>
```

Ovaj modul izvršava komandu tail /var/log/messages. Postoji mogućnost dodavanja i trećeg argumenta, a to je vrednost u milisekundama nakon kojih će se komanda ponovo izvršiti. Ako treći argument ne postoji, po početnim vrednostima on je 1000 ms, a ako želimo komandu da izvršimo samo jednom, ovde uzimamo vrednost 0. Ovde se mogu

izvršavati i kompleksije skripte tako što treba komande razdvajati sa tačka-zarezom(;). Postoji još jedan način da iz inline skripti pristupite bilo kojim podacima loristeći kontrole (Controls). Šta su kontrole? To su ustvari Piton moduli koji se lako koriste i mogu da se dele, a sistem gDesklets dolazi sa nekoliko ovih kontrola (vreme, datum ...). Da bi ste pristupili nekoj kontroli morate uraditi sledeće:

```
<control id="vreme" interface="ITime:9y703dqtfnv4w373case-rz68r"/>
```

sada da bi dobili vreme možemo da koristimo skript na sledeći način:

```
<script>
  h, m, s = vreme.time
  print "Vreme je:"
  print "%2d:%2d:%2d" % (h, m, s)
</script>
```

E sada da dodamo našem satu mogućnost konfiguracije. To ćemo uraditi na sledeći način:

```
<prefs>
  <page label="Prva strana">
    <string label="Vremenskazona" bind="vreme.-
timezone"/>
  </page>
  <page label="Druga strana">
    <string label="Nema nista"/>
  </page>
</prefs>
```

Ovim smo dobili mogućnost podešavanja vremenske zone u dijalog prozoru koji se bude otvorio. Još treba da dodamo funkciju koja pokreće sat na sledeći naći način:

```
<script><![CDATA[
  def kreni(new_value):
    h, m, s = new_value
    Dsp.okvir.value = "%2d:%2d:%2d" % (h, m, s)
    vreme.bind("time", kreni)
]]></script>
```

O ADL se može napisati cela knjiga, tako da je ovo samo jedan jako mali deo cele priče, koji će biti dovoljan da vas upozna i uvede u pisanje novih deskleta ili možda razvijanje gDesklets sistema. Ovim se vraćamo na pitanje sa početka, da li je sistem gDesklets ulepšavanje desktopa ili



nešto više? Ja na ovo pitanje još uvek neću odgovoriti. To ću prepustiti korisnicima da prosude sami. Samo ću kao konstataciju reći da je sistem još uvek u ranoj fazi razvoja (trenutno aktuelna je verzija 0.33). Šta dalje? Ako ste korisnik ovog sistema ili će te to u budućnosti postati posetite www.gnomesupport.org/forums i tamo postavite sliku vašeg desktopa ili potražite pomoć za eventualne probleme na koje niđete. Nadam se uskoro otvaranju domaćeg sajta posvećenog ovoj temi ili makar zasebnog foruma na nekom od postojećih.

Svakako treba posetiti:

www.gdesklets.gnomedesktop.org

www.gnomedesktop.org

www.gnomesupport.org/forums

Vidimo se tamo.

~Srđan Anđelković

Deset godina GIMP-a (deo prvi)

Zvezda je rođena

Istorija razvoja projekta „The GIMP“

DAVNE 1995. GODINE IZ GLAVA DVA STUDENTA NASTAO JE DANAS JEDINI OZBILJAN GNU PROGRAM ZA OBRADU BITMAPIRANE GRAFIKE. DESET GODINA SE ON RAZVIJAO OD ZAMISLI DO VELIKOG PROJEKTA KOJI NI GRAFIČKE PROFESIONALCE NEĆE OSTAVITI RAVNODUŠNIM. ČAK JE, PORED OSNOVNE NAMENE, OVAJ PIONIRSKI GNU PODUH VAT, SASVIM SLUČAJNO IZRODIO POPULARNU GRAFIČKU PODLOGU I JOŠ POPULARNIJE RADNO OKRUŽENJE. ZOVU SE JEDNOSTAVNO: GIMP, GTK+ I GNOM.

U početku beše želja...

Dva studenta na Berkliju, jula 1995. godine umesto da stečeno znanje u programiranju ulože na pisanje prevodilaca u lisp-u, rešila su da napišu program za obradu bitmapiranih slika. Iako nijedan od njih dvojice nije imao kontakta sa likovnom umetnošću, imali su tehničkog predznanja za realizaciju takvog projekta. Tako su Spenser Kimbal (*Spencer Kimball*) i Piter Matis (*Peter Mattis*) utemeljili korene programa za manipulaciju slikama „General Image Manipulation Program“ (GIMP).

Pre nego što su objavili prvu javnu verziju 0.54 (januar 1996.) koja je predvljala funkcionalan proizvod, na GIMP-u su radili oko deset meseci. U prvom izdanju, GIMP je imao dosta ugrađenih alata, sadržao je sistem dodatka (*PlugIn*), pa su drugi razvojni timovi mogli da pišu samostalne programe koji se bezbedno mogu dodati u glavni program bez narušavanja njegove funkcionalnosti. Postojali su osnovni alati za crtanje i operacije nad kanalima i opoziv akcije (*Undo*), što drugi programi ove namene nisu imali. GIMP je postao omiljen program među korisnicima, zaštićen GNUovom opštom javnom licencom i imao je zanimljivo i jednostavno ime koje se lako pamti. Uprkos mogućnostima, program se često rušio što zbog grešaka u glavnom programu, što zbog grešaka u dodacima.

U početku je bio baziran na Motif grafičkoj biblioteci, što je predstavljalo problem velikom broju korisnika, obzirom da je Motif veoma skupa komercijalna platforma (radi se o hiljadama dolara). Iz istog razloga, veliki broj programera, koji su želeli da daju svoj doprinos projektu pišući sopstvene dodatke, nisu to mogli da rade.

Kao i svaki drugi projekat na početku razvoja i GIMP je imao veoma mali broj korisnika koji su podržavali njegov razvoj. Spenser i Piter

nisu očekivali veći uspeh svog programa, bili su zadovoljni činjenicom da je njihov pionirski projekat koliko toliko zaživeo.

Predstavljanje javnosti

Jedan od prvih koraka u cilju promocije programa nakon izdanja 0.54 bio je organizovanje dopisnog društva za razvojni tim. Autorima je bila potrebna povratna informacija o upotrebljivosti njihovog čeda, a dopisno društvo je bio jedan od načina da korisnici urade tako nešto.

Kada su počela da pristižu pitanja vezana za osnovnu upotrebu programa, 5. jula 1996. godine *gimp-list* je podeljen na *gimp-user* i *gimp-developer*. Ubrzo je projekat dobio podršku i u

vidu glavne Internet prezentacije na serverima Berklija. Zak Bin (*Zach Beane – Xach*) kao i veliki broj drugih korisnika, počeli su sa pravljenjem uputstava tipa „korak po korak“ kako izvesti neku operaciju u programu.

Jedan od naprednijih korisnika, Lari Juing (*Larry Ewing*), u prvom izdanju GIMP-a nacrtao je čuvenu maskotu GNU/Linuks operativnog sistema – pingvina poznatijeg kao Tux¹.

Nova grafička osnova - serija

0.60

Piteru je bilo dosta Motif-a, pa je odlučio da napiše sopstvenu grafičku osnovu od početka. Nazvao je



gtk (za „Gimp Tool Kit“) i gdm (za „Gimp Drawing Kit“). Nikada nije ni zamišljao da bi jednog dana gtk mogao da postane jedna od najkorisnijih grafičkih osnova; hteo je samo da napravi osnovu za GIMP. Ubrzo nakon pojave gtk-a, došlo je do promene punog naziva GIMP-a iz „General Image Manipulation Program“ u „GNU Image Manipulation Program“.

Razvoj 0.6x izdanja bio je eksperimentalan. Sistem dodataka nije bio kompatibilan sa starim API-jem iz 0.54, tako da dodaci pisani za to izdanje nisu mogli da se koriste u novom. Autori dodataka su sporo prevodili svoje radove na novi API. Problem je nastao i oko dokumentacije jer niko nije želeo da piše dokumentaciju za seriju 0.6x koja je bila privremena i u razvoju.

Realno, ovo izdanje je imalo manju upotrebnu vrednost od prethodnog 0.54 izdanja; program se učestalije blokirao. Broj preuzimanja nove verzije znatno je opao i pored predstavljanja novih mogućnosti programa na berklisnoj prezentaciji (rad sa slojevima, automatizacija poslova skriptama, dobro osmišljene transformacije).

Serijski 0.6x dala je razvojnom timu mogućnost da se ugradi veliki broj unapređenja, poveća stabilnost u radu kao i da se doda veći broj alata za crtanje. Ovaj period evolucije GIMP-a najbolje je opisao Andreas Dilger citatom: „*It's good to see that the next version of GIMP is here. Let the enhancing begin!*“

Serijski 0.99 - dug put do 1.0

Izdanje 0.99 objavljeno je 26. februara 1997. godine. Osnovni cilj je bio prevođenje dodataka na novi API, koji je izostao u prethodnom 0.60 izdanju. Objavljeno je i novo izdanje gtk/gdk nazvano GTK+². U ovoj verziji su ugrađena revolucionarna unapređenja prethodnog gtk modela.

Ubrzo je objavljeno osveženo izdanje 0.99.9, nakon čega je nastala duža pauza. Trent Jarvi (*Trent Jarvi*) je napravio veliki broj sitnih izmena na ovom izdanju, a Spenser i Piter su 9. juna 1997. godine objavili nova izdanja GTK+ i GIMP 0.99.10. To je bio njihov poslednji rad na projektu.

Spenser i Piter su diplomirali, zaposlili se i više nisu imali vremena da rade na projektu. Nisu odredili naslednika koji bi nastavio njihovo rukovođenje, a čak su zaboravili i da najave napuštanje projekta.

Pred-11 izdanja

Napuštanje projekta od strane inicijatora usporilo je njegov razvoj. Pojavilo se nekoliko neoficijelnih predizdanja. Pošto je prikupljanje zakrpa postao veliki problem, privremeni nosilac projekta Met Hokinns (*Matt Hawkins*) objavio je izdanje pre-11.

Novi nosilac projekta postao je Federiko Menja Kintero (*Federico Mena Quintero – Quartic*), koji je na prvo mesto stavio stabilnost i upotrebljivost programa. Za izdanje 0.99.14, Quartic je najavio zaustavljanje dodavanja unapređenja i otklanjanje grešaka, sve do izdanja 1.0. Neka od zanimljivih unapređenja, koja su većim delom bila gotova, nisu ugrađena zbog insistiranja na stabilnosti programa.

Quartic je održavao GIMP u razvoju, sve dok nije prešao na druge

projekte. Ipak, nije bilo većih pauza između njegovog odlaska i dolaska novog nosioca projekta.

Nove komunikacije

Dopisna društva su korisna kada funkcionišu a veb strane su odlične za davanje saveta i predstavljanje mogućnosti projekata. Međutim, ništa se ne može porediti sa razgovorom uživo. Februara 1997. godine projekat GIMP je dobio IRC kanal #gimp na serveru irc.freenode.org. Kanal je bio otvoren za diskusije i otklanjanje otkrivenih problema. Veoma brzo je postao jedan od glavnih izvora tehničke podrške za korisnike programa. Na posletku je postao i izvor informacija o željama i potrebama korisnika za novim mogućnostima programa, kao i informisanje o načinu na koji pojedinci mogu da pomognu projektu. Preko kanala je širena i ideja o održavanju projekta u životu od strane članova razvojnog tima.

Podela posla

Otišli su začetnici projekta, Spenser i Piter, a potom i Kuartik. GIMP se podelio u projekat razvoja grafičke osnove (GTK+) i projekat razvoja programa. Došao je trenutak da oba projekta dobiju nosioca.

U novoj podeli, imenovani su članovi timova: Mains Sing (*Mainsh Singh - yosh*) je zadužen za objavljivanje novih izdanja. Adrijan Likins (*Adrian Likins*) za održavanje podataka. Lari Juing (*Larry Ewing - lewing*), Metju Vilson (*Matthew Wilson - msw*) i drugi su radili na pronalaženju i otklanjanju grešaka.

I sada je bilo dosta ljudi koji su istovremeno radili na više projekata, ali su svi oni radili kao tim bez vođe. Svaki pojedinac je radio na svom doprinosu projektu i uvek se znalo kome se treba obratiti za neki savet ili kada nešto zapne.

Pomoć na sve strane

Zak je 13. aprila 1997. godine započeo projekat GIMP News³, koji je stekao popularnost svakodnevnim izveštajima o razvoju GIMP-a. Ako je neko napravio novi dodatak, Zak ga je pronašao i o tome obavestio javnost. Ako je izdata nova verzija programa, slao je na dopisna društva vest i o tome. Njegova uputstva i formular za prijavu grešaka, koji je kasnije dodat, bili su direktno povezani na stranu sa vestima.

Tada se pojavio i veliki broj novih dodataka pa je pokrenut i Registar dodataka⁴, koji omogućava autorima da osveže svoje dodatke i ostalima da najave svoje planove u vezi sa razvojem novih dodataka. Ako ste autor, prijavite svoj dodatak na ovoj adresi, tako da korisnici uvek imaju uvid u njegov razvoj. Svaki korisnik odavde može da preuzme i prevede dodatak koga nema u zvaničnoj distribuciji programa.

Krajem maja 1997. godine Set Dž. Burdžiz (*Seth J. Burgess*) započeo je projekat GIMP Bugs⁵. Tu je prihvaćen veliki broj prijavi grešaka koje su ranije slate na dopisno društvo *gimp-developer* i organizovane za lakši uvid razvojnog tima i otklanjanje grešaka. U početku je sve rađeno ručno, uz pomoć nekoliko Perl skripti, pošto je dnevno prijavljivano samo nekoliko grešaka, da bi se kasnije ceo proces automatizovao.

Često zanemarivana oblast projekta bila je dokumentacija, pa su dva korisnika, Karin i Olof Kilander (*Karin i Olof S. Kylander*) 7. oktobra 1997. godine započeli projekat Gimp Users Manual⁶, koji je predstavljao sveobuhvatan popis mogućnosti u GIMP-u, opisanih na preko 200 strana dokumentacije. Danas ovaj projekat sadrži preko 600 strana i nalazi se na serverima ITC u Virdžiniji⁷.

Početkom aprila 2000. godine zaživeo je projekat Gimp user group⁸ koji okuplja korisnike GIMP-a. Na ovoj lokaciji se nalazi najsvežiji katalog uputstava za dobijanje zanimljivih efekata u novijim izdanjima programa.

Komercijalizacija

Nasuprot mnogim projektima slobodnog softvera, kod GIMP-a postoji javna komercijalizacija. Pojedini članovi razvojnog tima GIMP-a videli su šansu da zarade prilagođavajući kôd ili pišući dodatke u komercijalne svrhe. Takođe su prodavali CD-ROM-ove sa GIMP izdanjem 1.0, upotpunjene odgovarajućom bitmapiranom grafikom i dopunskim programima. Oformljen je i WilberWorks⁹.

Ipak, ovo ne može naškoditi razvoju GIMP-a pošto je sav rad na projektu i dalje zaštićen GNU-ovom opštom javnom licencom i novac nije glavni motiv učesća u razvoju.

Sjedinjavanje mrežnih resursa

Kao odgovor na potrebu da se svi izvori vezani za GIMP projekat centralizuju, Šon Amundson (*Shawn Amundson - Snorfle*) je 7. oktobra 1997. godine registrovao domen **gimp.org**. Ovde se nalazi velika količina podataka koja će vas informisati o samom projektu, od izvornog kôda do saveta i trikova. Osnova za ovaj tekst preuzeta je iz Istorije GIMP-a sa ove adrese.

Publikacije

Majkl Hamel (*Michael J. Hammel*) je u periodu od oktobra 1997. do januara 1998. godine u časopisu *Linux Journal*¹⁰ napisao četvorbrojni serijal o GIMP-u. Iako je program opisan samo površno, uradio je to na pravi način. Dobro napisani i kvalitetni tekstovi predstavili su GIMP Linuks zajednici, kao i svim drugim čitaocima Linuks literature. GIMP je takođe postigao publicitet zahvaljujući *Graphic Muse*¹¹ tekstovima u *Linux Gazette*¹².

Ko je Wilber?



Maskota GIMP-a je Vilber (*Wilber*). Vilbera je 25. septembra 1997. godine osmislio Tomas Kosminen (*Tuomas Kuosmanen - tigt*). Postao je prepoznatljiv znak koji asocira na GIMP. Globalno gledano, Vilber je za GIMP ono što je TUX za GNU/Linux.

A šta je bilo posle?

Sećate se GTK+? Neki članovi razvojnog tima GIMP-a su došli na suludu ideju da je to odlična osnova za pisanje raznih alata i da se između ostalog na njemu može izgraditi čitavo radno okruženje. U početku je i ova ideja, kao i mnoge druge, smatrana nemogućom misijom. Danas smo svedoci kvaliteta i upotrebljivosti *Gnoma*¹³, projekta koji je zasnovan na *Gtk+* grafičkoj podlozi.

Konačno se dogodilo „čudo“ o kome Spenser i Piter nisu ni sanjali. 5. juna 1998. godine u 17:17 po srednje evropskom vremenu objavljeno je GIMP izdanje 1.0. Bio je dovoljno stabilan program koji se mogao izboriti i sa profesionalnim zahtevima obrade bitmapirane grafike.

U sledeće dve godine radilo se na razvojnem izdanju 1.1. GIMP se već

razvio u izuzetno stabilan i veoma korišćen program, kada je 24. decembra 2000. godine objavljeno stabilno izdanje 1.2.0. Ovo je bio početak jednog novog poglavlja u istoriji GIMP projekta. Naime, naredne dve godine nije objavljeno ni jedno stabilno izdanje. Tek je 14. juna 2003. godine objavljeno izdanje 1.2.5 koje je sadržalo veliki broj ispravki i poboljšanja. Posle toga započet je rad i među članovima razvojnog tima objavljeno razvojno izdanje 1.2.6, ali ono nije ugledalo svetlost dana za nas obične smrtnike. Međutim, razvoj GIMP-a sa izdanjem 1.2.0 nije stao. Tokom narednog perioda rađeno je na razvojnoj seriji 1.3.x. Prvo stabilno izdanje ove serije, 1.3.0 objavljeno je 13. novembra 2001. godine. Tokom naredne dve godine izdata su još 23 izdanja iz ove serije, a poslednje 1.3.23 je objavljeno 24. novembra 2003. godine. Kompletan kod je napisan od početka, prešlo se na Gtk+ 2.x grafičku podlogu, ispravljen je veliki broj grešaka, unapređene su opcije podešavanja alata i dodat alat za vektore, dodate su podrška za datoteke veće od 2GB, keširanje od preko 4GB i manipulacija EXIF informacijama u zaglavlju JPEG datoteka, interni pretraživač pomoći je prebačen na GtkHtml2.

U toku prošle 2004. godine, tačnije od 7. januara, razvijano je stabilno izdanje 2.0. Posle četiri predizdanja i jednog kandidata (RC), 2.0.0 izdanje je predstavljeno javnosti 23. marta 2004. godine, implementiran je Gnom HIG standard (*Human Interface Guidelines*), unapređen alat za pisanje teksta, otklonjeno je dosta sitnih grešaka. U sledećih sedam meseci razvijana je stabilna serija 2.2. Izdanje 2.2.0 objavljeno je 19. decembra 2004. godine. Ispravljeno je dosta grešaka od prošlog stabilnog izdanja, dodata mogućnost uvoza Photoshop¹⁴ (.act) paleta, probni prikaz filtera i transformisanja, akcije koje su u toku sada se mogu prekinuti pritiskom na Escape taster, dodati novi filteri, uvoz SVG kontura kao putanje, podešavanja prečica sa tastature, „cut'n'paste“ između GIMP-a i drugih programa. Za ovo izdanje je bio organizovan konkurs za uvodni ekran¹⁵ (*splash*).

Suma sumarum

GIMP je sastavljen od preko 230.000 linija C kôda, od čega je najveći broj linija u toku evolucije izdanja 1.2 u izdanje 2.0 napisan od početka. Sa stanovišta korisničke upotrebe programa, GIMP 2.x je u osnovi ostao sličan izdanju 1.x; mogućnosti su slične onima koje su postojale na izdanjima iz 1.x serije, tako da se korisnici ranijih verzija mogu snaći i u novom okruženju. Kôd je prečišćen i omogućeno je lakše dodavanje novih mogućnosti u program, mada se taj deo manje tiče korisnika.

Program je zahvaljujući projektu lokalizacije na srpski jezik¹⁶ preveden na srpski tokom 2003. godine.

Trenutno aktuelna verzija u trenutku pisanja ovog teksta je 2.2.2.

~ Aleksandar Urošević

Internet adrese pomenute u ovom tekstu:

- 1 <http://www.isc.tamu.edu/~lewing/gimp/notes.html>
 - 2 <http://www.gtk.org>
 - 3 <http://xach.dorknet.com/gimp/news>
 - 4 <http://registry.gimp.org>
 - 5 <http://www.gimp.org/bugs>
 - 6 <http://www.dtek.chalmers.se/~d95olofs/manual/gimpmana.html>
 - 7 <http://www.itc.virginia.edu/unixsys/gimp/gimpdoc-html>
 - 8 <http://gug.sunsite.dk>
 - 9 <http://www.wilberworks.com>
 - 10 <http://www.linuxjournal.com>
 - 11 <http://www.graphics-muse.org>
 - 12 <http://www.ssc.com/lg>
 - 13 <http://www.gnome.org>
 - 14 <http://www.adobe.com/products/photoshop>
 - 15 <http://gimp.org/contest>
 - 16 <http://www.prevod.org>
-

Multimedijalne tastature

Kako podesiti specijalne tastere da rade pod GNU/Linux sistemom

BEZ OBZIRA NA NASLOV OVOG TEKSTA, OVDE NEĆEMO UČITI TASTATURU KAKO DA ZAPEVA, A NI KAKO DA PRIKAŽUJE FILMOVE ILI NEKE DRUGE MULTIMEDIJALNE SADRŽAJE. OVDE ĆEMO DA NAUČIMO NAŠ GNU/LINUX DA DETEKUJE I PREPOZNAJE SPECIJALNE TASTERE KOJA NAŠA TASTATURA IMA.

U poslednje vreme, tastature više nemaju standardan skup tastera nego svaki model ima neke specijalne, kao što su, na primer, taster Internet, tasteri Volume Up, Volume Down i slični. Uz drajvere za tastaturu se isporučuju i programi koji će tim posebnim tasterima dati značenje koje vama odgovara. Naravno, proizvođači se uopšte ni ne trude da naprave takve programe i za Linux.

Da li to znači da bi trebalo da se pomirimo sa situacijom da ćemo moći da koristimo samo ovih stotinak standardnih tastera? Naravno da ne. Posle ovog teksta videćete da imate mnogo veću kontrolu nad vašom tastaturom nego što je imaju oni smrtnici koji koriste Microsoft Windows i originalne drajvere i programe proizvođača.

Problemi sa kernelom

Ako koristite kernel iz serije 2.4, ovaj, i sledeći odeljak, u većini slučajeva, možete slobodno preskočiti. Jedan od problema koji nastaje pri prelasku sa 2.4 na 2.6 seriju je taj da specijalni tasteri, opet u većini slučajeva, neće biti uopšte mapirani. Samim tim, kernel neće prosleđivati informacije o njima.

Da bi ste proverili koji su tasteri mapirani, a koji ne, idite u konzolni režim i kucajte sledeće (obratite pažnju da znaci \$ i # predstavljaju odzivnik u konzoli i da se oni ne kucaju):

```
$ su
```

Sada ukucajte šifru root korisnika i pritisnite Enter. Unesite sledeću komandu:

```
# watch -n 1 "grep key /var/log/messages | tail"
```

Na ekranu ćete dobiti spisak sa nekoliko poslednjih poruka kernela u kojima se spominje reč "key" i taj spisak će se osvežavati svake sekunde. Počnite sa pritiskanjem specijalnih tastera. Za svaki taster koji nije mapiran, dobićete sledeće poruke (ili slične):

```
... kernel: atkbd.c: Unknown key released (translated set 2, code 0x85 on isa0060/serio0).
```

```
... kernel: atkbd.c: Use 'setkeycodes e005 <keycode>' to make it known.
```

Obratite pažnju na drugi red gde vam kernel daje uputstvo na koji

način da mapirate taj taster. U ovom slučaju "magični broj" tastera koji je bio pritisnut je e005. Za svaki od tastera koji nisu bili prepoznati zapišite negde taj broj. Prekinite program pomoću Ctrl+C.

Sledeći korak je da saznamo koji su key-kodovi slobodni da bi smo mogli da ih zauzmemo našim tasterima. Unesite sledeće:

```
# getkeycodes
```

Dobićete tabelu odnosa scan-kodova i key-kodova. Tu vidite koji se brojevi od 89-127 ne pojavljuju i njih, po svom slobodnom nahođenju, pridružujte scan-kodovima koje ste napisali na papiru. Sada pomoću komande setkeycodes mapirajte te

zapisane kodove. Obratite pažnju da se promene koje pravi setkeycodes brišu posle restartovanja sistema, tako da ne postoji mogućnost da nešto upropastite. Loša strana toga je da ćete, kada sve dobro podesite, morati u inicijalizacioni skript za podizanje računara da ubacite sve te komande.

Nedovoljno neiskorišćenih key-kodova

Ako se ispostavi da nemate dovoljno neiskorišćenih key-kodova da bi ste mapirali sve nepoznate tastere, onda nastaju muke jer setkeycodes program ne dozvoljava da unesete brojeve veće od 127.

Šta raditi u tom slučaju? Ili odustat, ili skinuti izvorni kôd setkeycodes programa i liniju u kojoj se proverava da li je prosleđeni keykod manji od 127 jednostavno izbrisati ili, što je još bolje, ograničenje promeniti u 255. Program kompjli-



rati i koristiti umesto standardnog. Sada ćete biti u stanju da za key-kodove koristite sve brojeve od 89-255.

xev, xkb

Ako ste, kojim slučajem, pomislili da je ovo kraj vašim mukama, grdno ste se prevarili. Ostaje vam još da naučite X sistem da prepoznaje te nove tastere.

Pokrenite iz emulatora terminala program xev. Ovaj program otvara mali prozor i na standardni izlaz ispisuje sve informacije koje tom prozoru šalje X server. Slično kao na početku, pritiskajte specijalne tastere i zapisujte key-kodove koji se ispisuju u konzoli. Primetićete da se ovi key-kodovi ne poklapaju sa onima koje ste unosili prilikom kucanja setkeycodes naredbe. Obratite pažnju na to da zapisujete kodove samo onih tastera koji vraćaju NoSymbol poruku.

Sada otvorite /etc/X11/xkb/symbols/inet fajl vašim omiljenim editorom. Odaberite neku od tastatura koja je najbližnja vašoj i dodajte zapisane kodove koje vam je vratio xev. Ako je kod veći od 128, onda od njega oduzmite 128, prebacite u heksadekadni sistem i dodajte slovo I na početak.

Na primer, ako je xev vratio 139, onda u inet fajl dodajte red:

```
key <I0B> { [ XF86NekoImeTastera ] };
(139 - 128 = 11, a u heksadekadnom zapisu 11 je 0B)
```

Sada aktivirajte tu tastaturu, ponovo pokrenite xev program i pritiskajte tastere. Umesto NoSymbol reči, u našem slučaju iz prethodnog reda će se ispisati XF86NekoImeTastera. Ako vam se još negde pojavljuje NoSymbol, to znači da ste u nekom koraku napravili grešku. U tom slučaju, jedino što vam preostaje je da se igrate i eksperimentišete sa brojevima dok vam tasteri ne prorade.

KHotKeys, Keyboard Shortcuts

Sada kada vaš sistem zna da prepoznaje te tastere, bilo bi lepo da oni nešto korisno i rade. U tu svrhu iskoristite KhotKeys u KDE-u ili Keyboard Shortcuts u Gnomu.

Prednost ovih programa, u odnosu na standardne programe koji dolaze uz tastaturu je taj da određeno dugme može biti korišćeno i u kombinaciji sa tasterima Shift, Alt i Control – na primer, možete podešeti da, kad pritisnete taster Internet, da se pokreće Mozilla Firefox, a kad pritisnete Shift+Internet da se pokrene Konqueror i slično.

Zaključak

Ovo je, moram priznati, jedan od primera kod kog je nemoguće reći da je GNU/Linux user-friendly operativan sistem. Naveden postupak uopšte nije onoliko jednostavan koliko bi trebalo da bude. Većina korisnika želi da samo uključi tastaturu i da ona radi.

Sa druge strane, postavlja se pitanje da li vredi truditi se oko tastature samo da bi proradili još koji tasteri? Smatram da jeste, pogotovo za one kojima su šake češće na tastaturi nego na mišu – produktivnost i brzina rada se na taj način drastično povećavaju jer svaki od tih tastera zamenjuje po nekoliko kliktaja mišem.

~ Ivan Čukić

Intel, Lucent, Conexant

Drajveri, drajveri...

Instalacija drajvera za modeme tri najpoznatija proizvođača

U PRETHODNOM BROJU SMO GOVORILI O SOFTVERSKIM MODEMIMA, KAKO IH PREPOZNATI, I KOJI MODELI SU NAJBOLJE PODRŽANI NA GNU/LINUX SISTEMIMA, ALI SMO SE SAMO LETIMIČNO DOTAKLI INSTALACIJE DRAJVERA. TA TEMA JE DOVOLJNO OPŠIRNA I BITNA DA ZASLUŽUJE POSEBAN PROSTOR KOJI ĆEMO JOJ POSVETITI U OVOM BROJU.

Ne postoji tačan statistički podatak o tome koje su marke modema i u kom procentu zastupljene na našem tržištu, ali je sigurno da preko 95% računara poseduje modem koji koristi čip jednog od tri proizvođača. Nije teško pogoditi da su u pitanju Intel, Lucent ili Conexant.

Kao što je već pomenuto, u prethodnom broju smo objasnili kako prepoznati koji čip koristi Vaš modem i kako naći drajver za njega. Priču nastavljamo, pod uslovom da ste uspešno prepoznali čip koji modem koristi i da posedujete odgovarajući drajver.

Instalacija drajvera u binarnom obliku

Distribucija softvera, pa i drajvera, se generalno može vršiti u dva oblika: binarnom i source (izvorni kod). Binarni paketi sadrže već kompajlirane module za određenu verziju kernela određene distribucije, instaliraju se veoma lako, skoro uvek jednom komandom. Mogu se naći u nekoliko različitih oblika odnosno sa nekoliko različitih ekstenzija, što zavisi od distribucije koju koristite. Ako koristite RedHat, Mandrake ili SuSE trebaće vam .rpm paketi, ukoliko koristite Debian, Ubuntu ili neku Debian baziranu distribuciju koristićete .deb pakete, dok Slackware koristi .tgz, Gentoo .tbz... Drajveri za modeme u imenu paketa sadrže informaciju za koju verziju kernela i distribucije su namenjeni tako da je veoma teško pogrešiti pri odabiru paketa pri preuzimanju sa interneta. Najbitniji momenat je sama instalacija paketa koja se obavlja na različite načine, u zavisnosti od tipa paketa.

RPM paketi su izvorno napravljeni za potrebe Red Hat distribucije, ali je puno drugih distribucija prihvatilo ovu vrstu paketa za izgradnju svog sistema, tako da postoji relativno veliki broj distribucija na kojima je zastupljen. Instalacija je veoma jednostavna i svodi se na sledeću komandu:

```
#rpm -i ime_paketa.rpm
```

Pošto je postupak uspešno obavljen, Vaš drajver će biti instaliran, a modem spreman za upotrebu.

Deb pakete koristi Debian distribucija i sve Debian bazirane distribucije. Instalacija je podjednako laka kao i instalacija rpm paketa, što znači da je jedna komanda dovoljna:

```
#dpkg -i ime_paketa.deb
```

nakon koje će se na ekranu istpisati informacija o tome kako se i koli-

ko uspašno obavila instalacija.

TGZ je vrsta paketa karakteristična za Slackware distribuciju i nije teško naslutiti da se instalacija i ove vrste paketa svodi na sledeću komandu:

```
#installpkg ime_paketa.tgz
```

Instalacija drajvera iz binarnih paketa je zaista laka i trivijalna. Prednost ovakvog načina instalacije je i to što nije potreban izvorni kod kernela da bi se drajver instalirao, što je neophodno kada su u pitanju drajveri iz izvornog koda. Nedostatak ovakvog metoda je moguća nestabilnost u radu ili još gore, nemogućnost korišćenja modema čiji su drajveri instalirani iz binarnih paketa. Ukoliko ne uspete da osposobite modem uz pomoć drajvera iz binarnih paketa ili binarni paket koji vam je potreban ne postoji, ostaje mogućnost drajvera iz izvornog koda.

Instalacija drajvera iz izvornog koda

Za razliku od binarnih paketa, da bi drajver iz izvornog koda poradio potrebno je uložiti malo više truda. Na prvom mestu, instalacija nije zajednička za sve vrste modema kao što je slučaj sa binarnim paketima, već se razlikuje od vrste do vrste čipova, pa čak se postupak razlikuje i kada je u pitanju isti čip ali različita verzija kernela (recimo 2.4.x i 2.6.x). Izvorni kod se distribuira u obliku arhiva koje mogu imati različite ekstenzije i to: tar, tgz (iako se Slackware binarni paketi obeležavaju isto, mogu se naći i arhive sa ovakvom ekstenzijom), tar.gz, tar.bz2. Ove arhive je potrebno prvo raspakovati



što se postiže jednom od sledećih komandi (u zavisnosti od vrste paketa):

```
#tar -zxvf ime_paketa.tar
#tar -zxvf ime_paketa.tgz
#tar -zxvf ime_paketa.tar.gz
#tar -jxvf ime_paketa.tar.bz2
```

Kada se arhiva raspakuje dobićete novi direktorijum koji sadrži izvorni kod drajvera i komandom

```
#cd novi_direktorijum
```

treba preći u novi direktorijum i obaviti ostatak posla.

Intel modemi su najbolje podržani modemi kada je u pitanju GNU/Linux platforma. Svi noviji čipovi ovog proizvođača imaju drajvere kako za 2.4 tako i za 2.6 verzije Linux kernela. Pored toga, izuzetno su laki za instalaciju i retko kada je potrebno dodatno intervenisati posle instalacije drajvera da bi modem radio. Pošto ste raspakovali arhivu sa drajverom i prešli u novi direktorijum potrebno je izvršiti tri komande da bi drajver bio iskompajliran (preveden) i instaliran, a to su:

```
#make clean
#make IME_ČIPA
#make install
```

Lakoća instalacije Intel drajvera leži u tome što se postupaci instalacije svih modela razlikuju samo u jednom segmentu, a to je komanda za prevođenje koda koja zavisi od modela čipa. Prema tome, pri instalaciji svog drajvera ćete IME_ČIPA zameniti odgovarajućom oznakom čipa koji poseduje vaš modem, i to izgleda ovako:

```
#make 536 (za modeme za 536 ili 536EP čipom)
#make 537 (za modeme za 537 ili 537EP čipom)
#make ham (za modeme za ham čipom)
```

Ovo će u 99% slučajeva biti dovoljno da modem proradi jer će sve potrebne operacije (pravljenje device nod-a - uređaja koji predstavlja vaš modem i podešavanje startovanja modula pri podizanju sistema). Ukoliko se dogodi neka nepredviđena okolnost i modem ne proradi, uvek ostaje mogućnost da se stvar uradi "ručno". Naravno, potrebno je da se drajver uspešno iskompajlira (znači make 537 npr.), a ako skripta ne instalira drajver kako treba treba probati sledeće:

```
#insmod -f Intel537.o (Intel537.ko za 2.6 verzije kernela)
```

Ovom komandom se instalira modul dobijen kompajliranjem koda,

```
#rm /dev/537
```

uklanja se device nod 537 ako postoji,

```
#mknod /dev/537 c 240 1
```

mknod komanda će napraviti novi device nod za vaš modem (240 je podrazumevana vrednost, a ako ne proradi podledati

```
/proc/devices gde će pisati koji broj odgovara vašem modemu) ,
```

```
#ln -s /dev/537 /dev/modem
```

ovom komandom se pravi simbolički link ka modemu kako bi ga programi koji će ga koristiti lakše pronašli (obično je podrazumevana lokacija koju programi za povezivanje

na internet koriste /dev/modem).

Ovo bi trebalo da bude dovoljno da modem proradi i gotovo uvek je ovaj postupak dovoljan ukoliko klasičan

postupak instalacije zakaže. Naravno, ako vaš modem nije 537, poslužite se analogijom sa primerom. 537 zamenite modelom vašeg modema kao što je već objašnjeno.

Link za preuzimanje drajvera možete naći kako na www.linmodems.org tako i na Intelovim stranicama posvećenim preuzimanju drajvera.

Kada su u pitanju Lucent modemi situacija je malo komplikovanija. Na prvi pogled stvari ne izgledaju tako jer postoji jedan model Lucent modema (Lucent LT) koji uspešno radi na GNU/Linux sistemima (i ne samo Linuxima), ali je instalacija drajvera malo komplikovanija nego kada su u pitanju Intel modemi. Drajver za 2.4 kernele je dobro urađen, postupak instalacije obavlja sve što je potrebno dok se to ne može reći za drajver namenjen 2.6 kernelu.

Ako još uvek koristite 2.4 kernel, nećete imati problema prilikom instalacije drajvera. Potrebno je svega tri koraka da bi drajver bio instaliran i to su:

```
#!/build_module
#!/tinst2
#!/autoload
```

```
root@attitude: /usr/local/src
root@attitude: /usr/local/src # tar -jxvf ltmodem-2.6-alk-6.tar.bz2
ltmodem-2.6-alk-6/
ltmodem-2.6-alk-6/Makefile
ltmodem-2.6-alk-6/linuxif.h
ltmodem-2.6-alk-6/lt_modem.c
ltmodem-2.6-alk-6/ltmndobj.o
ltmodem-2.6-alk-6/serial.c
ltmodem-2.6-alk-6/README
ltmodem-2.6-alk-6/ltmndobj.o-8.31
ltmodem-2.6-alk-6/ltmndobj.o-pre-8.31
root@attitude: /usr/local/src #
```

Ukoliko se ove tri komande uspešno izvrše, modem je spreman za upotrebu. Nakon instalacije je preporučljivo pogledati

```
/etc/modules.conf fajl u kom bi trebalo da postoje sledeće tri linije:
alias char-major-62 lt_serial
alias /dev/tts/LT0 lt_serial
alias /dev/modem lt_serial
```

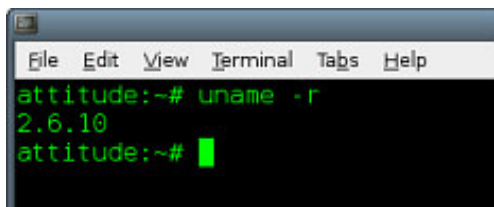
Ove linije obezbeđuju da se modul učita pri podizanju sistema, ako ne postoje unesite ih. Ova verzija drajvera radi veoma pouzdano i ne bi trebalo da prouzrokuje probleme. Ovaj postupak za instalaciju nije jedini, već je na početku instalacije moguće kreirati odgovarajući .rpm ili .deb paket komandom

```
#!/build_deb ili
#!/build_rpm
```

i instalirati prema uputstvu za instalaciju drajvera iz binarnih paketa.

Ukoliko koristite 2.6 kernel, trebaće malo više (ali ne i previše) truda da biste instalirali drajver. Kada se pojavio Linux kernel u 2.6 verziji, pokušaji osposobljavanja Linucent LT modema su se svodili na podršku 2.6 kernela da učita module namenjene 2.4 kernelima, što nije davalo korisne rezultate. Na sreću, ubrzo su se pojavili drajveri koji su omogućavali korišćenje ovog modema na 2.6 kernelima. Jedina stvar koja nije bila tako dobra je instalacija. Naravno, iskusnijim korisnicima ovo nije problem, ali onima sa malo manje znanja i iskustva može biti.

Na adresi <http://linmodems.technion.ac.il/packages/ltmodem/kernel-2.6/> mogu se pronaći arhive sa drajverima među kojima se najbolje pokazao izvorni kod drajvera zapakovan u arhivu sa imenom ltmodem-2.6-alk-6.tar.bz2. Prvi korak u instalaciji



ovog drajvera predstavlja kompajliranje modula. U nekim ranijim verzijama drajvera je potrebno da znate koju verziju kernela imate na sistemu, što je

moguće saznati komandom

```
#!/uname -r
```

Nakon toga, potrebno je editovati fajl pod imenom Makefile u kom ćete pri početku naići na liniju:

```
KERNEL_DIR := /usr/src/linux-2.6/
```

Umesto dela linux-2.6.6 u ovoj liniji treba uneti informaciju o vašem kernelu, odnosno ono što ste dobili kada ste otkucali komandu `uname -r` i sačuvati promene. Nakon što je Makefile editovan, treba izvršiti komandu

```
#!/make
```

da bi ste iskompajlirali module potrebne za rad vašeg modema. Kada je verzija drajvera koja je pomenuta u pitanju (ltmodem-2.6-alk-6.tar.bz2), nije potrebno editovati Makefile jer je problem rešen na bolji način, tako da sada skripta sama pronalazi putanju ka izvornom kodu kernela. Pošto se kompajliranje završi, treba prekopirati module `ltmodem.ko` i `ltserial.ko` u `/lib/modules/ime_vašeg_kernela/other` direktorijum. Naravno, ime_vašeg_kernela je isto ono što je upisano

prilikom editovanja fajla Makefile, a dobija se komandom `uname -r`. Dakle, komande koje treba izvršiti su sledeće:

```
#!/mkdir /
lib/modules/ime_vašeg_kernela/other
```

ovom komandom se pravi direktorijum `other` u

```
/lib/modules/ime_vašeg_kernela/ direktorijumu,
```

```
#!/cp ltmodem.ko /
lib/modules/ime_vašeg_kernela/other
```

```
#!/cp ltserial.ko /
lib/modules/ime_vašeg_kernela/other
```

a ovim komandama se kopiraju moduli u direktorijum iz kog će se učiti.

Nakon što su moduli smešteni na svoje mesto, treba napraviti device nod za vaš modem komandom

```
#!/mknod --mode=0640 /
dev/ttyLT0 c 62 64
```

Potrebno je podesiti da dozvole za `/dev/ttyLT0` budu identične dozvolama fajla `/dev/ttyS0`, što se može uraditi uz pomoć Konquerora ili Nautilusa, u tabu `permissions`, `properties` prozora koji se dobija odabirom istoimene opcije u meniju koji se dobija klikom desnog tastera miša na fajl. Poželjno je napraviti link ka modemu komandom

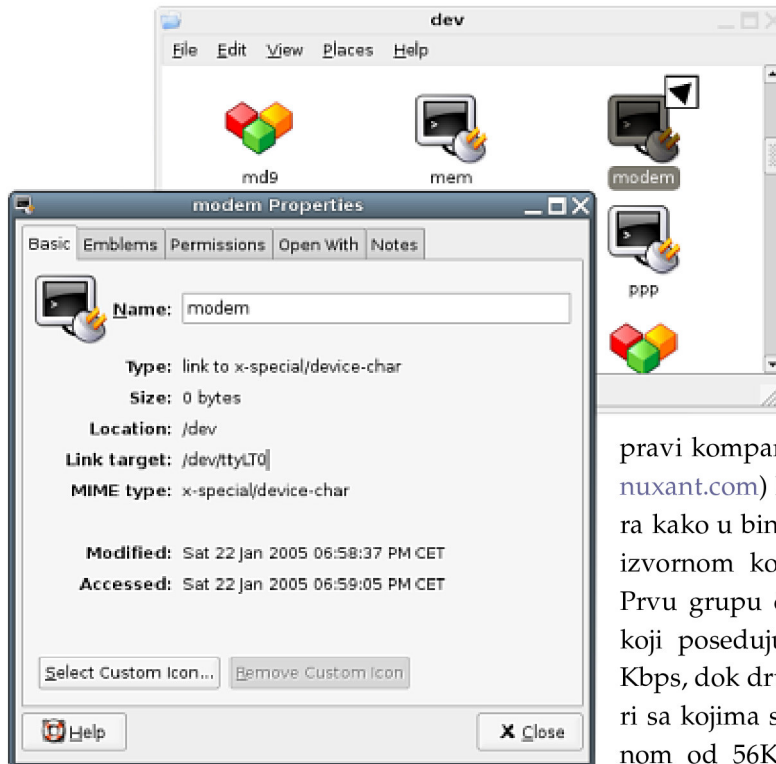
```
#!/ln -s /dev/ttyLT0 \
/dev/modem
```

Ostalo je da se moduli učitaju i obezbedi njihovo učitavanje prilikom podizanja sistema. Moduli se učitavaju komandom

#depmod -a
 ali je ponekad potrebno specijalno pozvati učitavanje modula za LT modem komandom

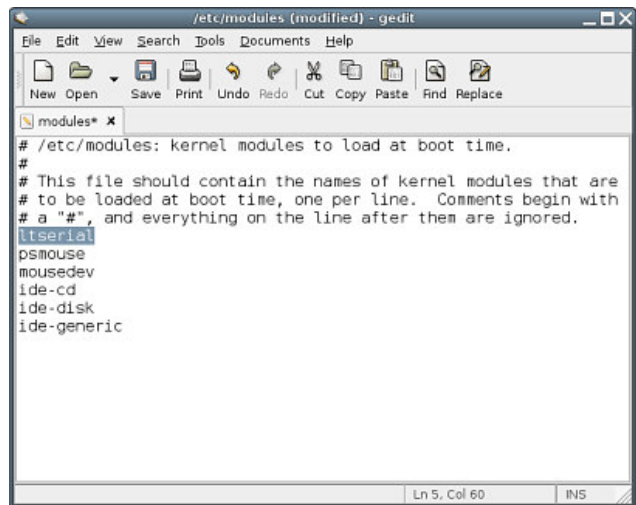
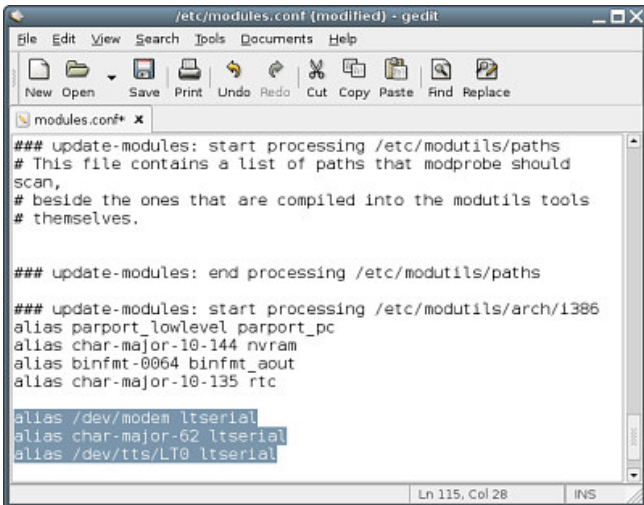
#modprobe ltserial

Pošto ste učitali module možete pokušati da se povežete na internet vašim omiljnim programom za ovu namenu, a da bi se moduli učitali automatski prilikom sledećeg podizanja sistema potrebno editovati fajlove modprobe.conf ili modules.conf u koje treba uneti sledeće redove:



se uzme u obzir činjenica da je Conexant kupio dobro poznatog Rowcwell-a čiji modemi se mogu pronaći na najstarijim konfiguracijama sa kojima se možemo sresti. Na žalost, drajvere za ove modeme

pravi kompanija Linuxant (www.linuxant.com) koja drajvere distribuira kako u binarnom obliku, tako i u izvornom kodu, ali u dve grupe. Prvu grupu čine besplatni drajveri koji poseduju ograničenje na 14.4 Kbps, dok drugu grupu čine drajveri sa kojima se može surfovati brzinom od 56Kbps, ali koštaju 14.95 dolara. Reakcija na zajednice na naplaćivanje drajvera je bila veoma



```
alias /dev/modem ltserial
alias char-major-62 ltserial
alias /dev/ttyLT0 ltserial
```

Ovi fajlovi su različito raspoređeni u zavisnosti od distribucije. Putanje na kojima treba potražiti ove fajlove su /etc ili /lib/modules. Kao mogućnost postoji editovanje fajla /etc/modules u koji treba uneti ltserial. Konsultujte forume i diskusione grupe namenjene distribuciji koju koristite da biste znali koji fajl treba da editujete i gde se on nalazi.

Ka što je rečeno na početku, instalacija ovog drajvera za 2.6 kernel nije tako jednostavna, ali bi sada trebalo da bude daleko jasnije šta i kako treba uraditi ne bi li vaš LT radio na 2.6 kernelu.

Conexant modemi ne zaostaju po kvalitetu za Intel ili Lucent modemima. Ovi modemi su veoma dugo prisutni na tržištu, naročito ako

burna, dok se na internetu mogu pronaći zakrpe koje limitirane drajvere osposobljavaju da rade punom snagom. Na Linuxant sajtu se može videti da se plaćanjem licence dobija aktivacioni kod koji je namenjen jednom modemu, tako da je jasno je da korišćenje zakrpa koje ograničnim verzijama omogućavaju da rade kao neograničene predstavlja nelegalan čin tako da se nećemo baviti tim postupkom. Postoji beta verzija drajvera namenjena 2.4 kernelima, koja se pokazala veoma po-

uzdanom, ali za 2.6 kernele se mora koristiti finalna verzija, odnosno treba platiti 15 dolara za surf na 56K. GNU filozofija je zasnovana na slobodi, što znači da je bitno da softver bude slobodan (ali ne i uvek besplatan), a čitajući licencu prilikom preuzimanja drajvera jasno se daje do znanja korisniku da su ovi drajveri sve, samo nisu slobodni, a još se i plaćaju. Već viđeno, ali ne na GNU/Linuxu.

Instalacija je veoma laka i svodi se dve komande od kojih je prva zajednička za sve modele i glasi:

```
#make install
```

a druga zavisi od tipa modema.

```
#hcfpciconfig
```

će obaviti ostatak posla oko instalacije i podešavanja modema ukoliko je on HCF model, dok je za HSF model potrebno otkucati komandu

```
#hsfpciconfig
```

koja će završiti posao kod ovog modela. Ako posedujete USB eksterni model modema, umesto prethodnih komandi potrebna je jedna od ove dve u zavisnosti od modela:

```
#hcfusbconfig
```

```
#hsfusbconfig
```

Pošto je modem instaliran i konfigurisan može mu se pristupiti na lokaciji /dev/ttySHCF0 (/dev/ttySHCFUSB0 za USB modele) ili /dev/ttySHSF0 (/dev/ttySHSFUSB0 za USB modele), u zavisnosti od modela modema, naravno. Može biti veoma korisno napraviti link komandom

```
#ln -s /dev/ttySHCF0 /dev/modem
```

```
#ln -s /dev/ttySHCFUSB0 /dev/modem
```

```
#ln -s /dev/ttySHSF0 /dev/modem
```

```
#ln -s /dev/ttySHSFUSB0 /dev/modem
```

radi lakšeg pronalaženja modema od strane programa za povezivanje na internet, opet u zavisnosti od modela modema.

Ovo je dovoljno da vaš Connexant proradi na 14.4Kbps, a može li brže? Na vama je da odlučite.

FreeBSD...

Ceo tekst, osim ovog dela, se tiče GNU/Linux sistema jer su softverski modemi najbolje podržani na njima, ako se izuzme operativni sistem zbog kog oni i jesu takvi kakvi su. FreeBSD je po zastupljenosti na drugom mestu među Unixolikim sistemima, ali su softverski modemi daleko manje zastupljeni na njemu nego na GNU/Linuxu. Tačnije, samo jedan model softverskih modema je podržan kada je u pitanju FreeBSD, a to je Lucent LT. Drajver za ovaj modem je veoma dobro razvijen, a od 5.3 verzije FreeBSD-a je uvršten u grupu paketa koji se nalaze na instalacionim disku, tako da se može instalirati pri instalaciji sistema. No, ta verzija drajvera može prouzrokovati određene probleme u kernelu (kernel trap 12) tako da je treba koristiti samo za preuzimanje novije verzije. Kada se drajver instalira modemu možete pristupiti na lokaciji /dev/cual0, ali je bitno napomenuti da kppp neće raditi, već je konekciju potrebno obaviti pomoću ppp komandne linije. Preuzimanje, odnosno instalacija nove verzije drajvera se može obaviti na dva načina, u duhu instalacije softvera na FreeBSD

sistemu. Prvi način predstavlja preuzimanje binarnog paketa sa ekstenzijom .tbz i instalaciju komandom

```
#pkg_add ltmodem-1.4.tbz
```

dok drugi način predstavlja preuzimanje izvornog koda drajvera sa FreeBSD servera. Ovakav drajver se instalira na sledeći način:

```
#cp ltmodem-1.4.tar.gz
```

```
/usr/ports/distfiles
```

```
#cd /usr/ports/comms/ltmdm
```

```
#make install clean
```

Naravno, da biste instalirali novi, teba prvo deinstalirati stari drajver što se može uraditi pozivanjem sistemskog instalacionog programa komandom

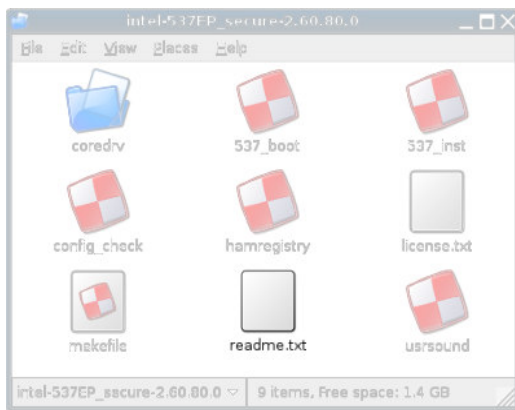
```
#sysinstall
```

U trenutku pisanja teksta LT drajver postoji u verziji 1.4.

Surfovanje internetom na FreeBSD-ju uz pomoć Lucent LT modema je krajnje komforno, a veza je stabilna što pokazuje da je drajver veoma dobro razvijen i pouzdan.

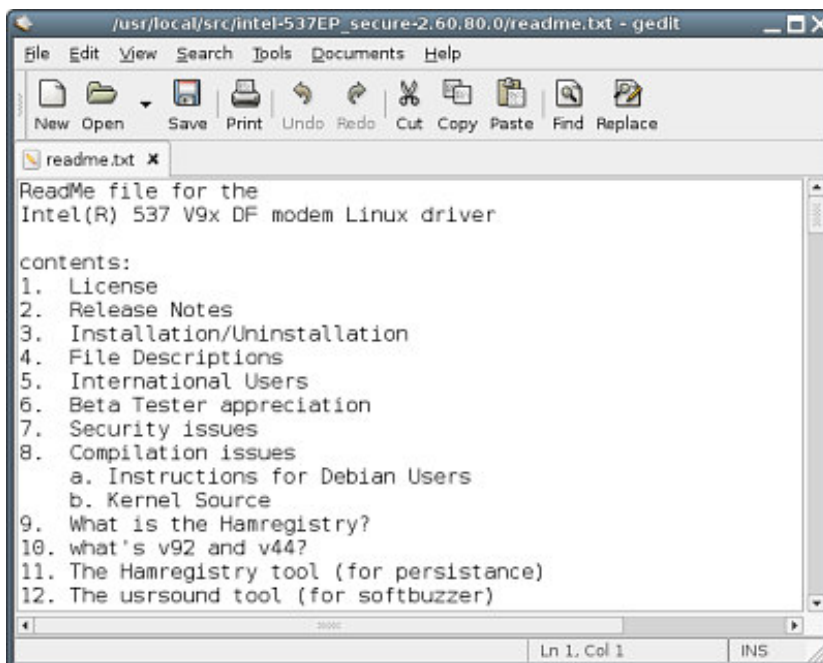
Na kraju

Svi drajveri predstavljeni u ovom takstu se instaliraju iz konzole sa root privilegijama. Znak # u primerima komandi ne treba kucati jer on predstavlja indikator root dozvola (kada ste običan korisnik pisaćće \$). Kao što je pomenuto na početku, ukoliko instalirate drajver iz izvornog koda, biće vam neophodan izvorni kod kernela koji se na većini distribucija instalira sa instalacionih diskova. Potrebno je u menadžeru paketa pretražiti pakete koji u nazivu sadrže reč kernel-source i instalirati odgovarajući, a koji je odgovarajući znaćete komandom uname -r, jer će paket koji bude pronađen sadržati oznaku dobijenu ovom komandom u svom imenu. Grafičkih frontenda za instaliranje drajvera još uvek nema, osim kada su u pitanju drajveri u binarnom odliku koji



se mogu instalirati grafičkim alatima karakterističnim za određene distribucije. Veoma je bitno, zato to ponovo naglašavam, pročitati README ili INSTALL fajlove u direktorijumu drajvera jer se u njima obavezno nalaze sve informacije koje mogu biti

potrebne.



Kao što se može videti, softverski modemi su prilično dobro podržani kada je u pitanju GNU/Linux platforma. Pojedine distribucije poseduju prekompajlirane drajvere koji se mogu instalirati pri instalaciji sistema, dok druge pokušavaju same da razviju drajver za svoje korisnike. Primer toga predstavlja Ubuntu koji poseduje drajver za Lucent LT modem prekompajliran za podrazumevani kernel na ovoj distribuciji, tako da korisnici ove distribucije ne moraju da se zamaraju relativno komplikovanom instalacijom drajvera iz izvornog koda. Softverski modemi su slaba tačka GNU/Linux sistema iz prostog razloga što postoje na većini konfiguracija, kako na onima koje već koriste GNU/Linux, tako i na potencijalnim Linux mašinama. Stoga, svaki napor je dragocen jer što više hardvera treba da radi na Linux-u i to kako treba.

U svakom slučaju, uvek su tu hardverski modemi čija se instalacije svodi na priključivanje kabla u računar. Dakle, cena koja ne prelazi deset eura i instalacija drajvera ili nekoliko desetina eura, kabl koji treba ubaciti u odgovarajući port i bolja veza, presudite sami.

~Ivan Jelić

Gimp tip

Gimp i crvene oči

Od ovog broja u časopis uvodimo novu rubriku. Nadamo se da će trikovi i korisni saveti koji se nađu ovde biti od pomoći

Uklanjanje "crvenih očiju" sa fotografija (red-eye remove)

GIMP (GNU Image Manipulation Program) je do skora smatran ameterskim poduhvatom grupe ljudi iz Open Sourcea i najčešće je bio potiskivan od strane nekih nadaleko čuvenijih programa kao što su Adobe Photoshop, Macromedia Fireworks, Photo Paint... Razloga za ovako nešto zaista nema jer je GIMP prerasao u jedan kompletan alat za manipulaciju fotografijama i verno služi kako amaterima tako i majstorima web dizajna. Da je to zaista tako u sledećih nekoliko brojeva časopisa pokušaću da vas uvedem u GIMP-ov svet i podelim sa vama par prostih trikova koji će vam zasigurno trebati.

Najnovije verzije GIMP-a uvek možete preuzeti sa <http://www.gimp.org>. Trenutno aktuelna verzija gimpa je 2.2.1. Na adresi <http://gug.sunsite.dk> (Gimp User Group) aktivan je i forum. Uputstva, tutoriali i knjige u štampanim ili html verzijama mogu se naći na obe adrese.

Zamislite situaciju...

Slikali ste neke ljude vašim digitalnim foto aparatom, doneli slike kući prebacili ih u kompjuter i otvorili. Razočaranje je nastupilo kad ste shvatili da oči svetle na gotovo svim slikama. Naravno da je sad tu Photoshop ili ACDSee lako biste ispravili slike, ali sta sad?

GIMP skripta za uklanjanje crvenih očiju

Da biste uopšte mogli da instalirate bilo koj plugin za GIMP morate imati i gimpovo plugin razvojno okruženje (GIMP plugin development environment) koj obično dolazi uz svaku distribuciju Linuxa pod nazivom libgimp-2.x-dev. Ukoliko je instaliran, moći ćete da pokrenete

```
# gimpool-2.x
```

Najpre je potrebno da skine-

te izvorni kod plugina redevye sa adrese: <http://registry.gimp.org/plugin?id=4212>. Gotovo. Sada treba da kompajliramo plugin. Udajte u direktorijum u kome vam se nalazi skinit izvorni kod (redevye.c) i pokrenite sledeću komandu

```
# gimpool-2.0 --build redevye.c
```

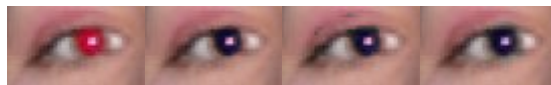
Dobićete binarni kod plugina koj se zove redevye. Kopirajte ga u \$HOME/.gimp-2.x/plugins, odnosno za gimp 2.2 verziju:

```
# cp redevye $HOME/.gimp-2.2/plugin-ins/
```

Pri sledećem pokretanju gimpa plugin će biti aktiviran

Rad sa skriptom

Dakle, vaša slika izgleda kao prva prikazana slika. Najlakši način za uklanjanje crvenih očiju




je jednostavni odabir Auto Red Eye Remover. U meniju koj se nalazi iznad fotografije odaberite Filters, Misc, Auto Red Eye Remover i plugin će sam obaviti zatatak. Rezultat se vidi na drugoj slici.


Ukoliko želite da fino podesite stepen uklanjanja crvene boje bolje vam je da koristite Red Eye Remover opciju iz istog menija, treća slika je uradjena ovom opcijom tako što je skala nameštena na 8. Negativne vrednosti skale neće bitno uticati na sliku dok će jako velike vrednosti uništiti svaki tračak crvene boje na celoj fotografiji.

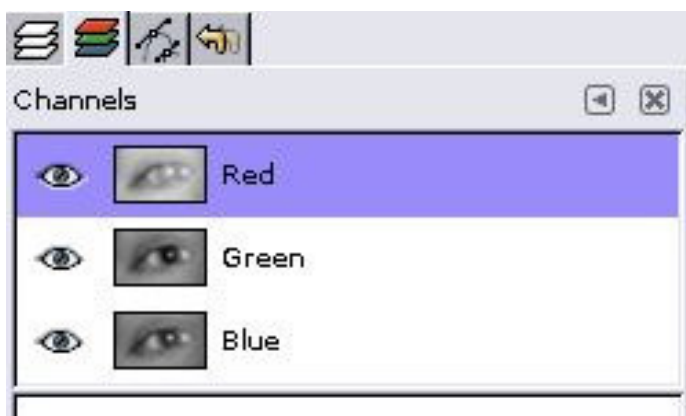
Treći i najefikasniji način da uklonite crvene oči je prethodno selektovanje dela koj želimo da "odcrvenimo". Da biste selektovali samo crveni deo očiju najbolje je da koristite Ellipse Select alat-



ku . Jednostavno prevucite krug preko dela koj želite da “odcrvenite”, odaberite meni Filters, Misc, Red Eye Remover i pojaviće vam se skala koju možete da stavite na maksimalnu pozitivnu vrednost. Rezultat je daleko bolji nego prethodni (treće slika).

Uklanjanje crvenih očiji Dodge/Burn alatom

Za one koji žele da uklone crvene oči bez prethodnog instaliranja plugina može da posluži alatka koja je već integrisana u GIMP. Otvorite sliku i malo je uvećajte da bi vam bilo lakše da njom baratate. Odaberite Dodge/Burn alat  i podesite ga na sledeći način. Uzmite “veličinu četkice” koja odgovara veličini crvenog dela oka, u polju gde vam stoji Type Dogle/Burn alatke odaberite Burn, i u polju Mode odaberite Midtones. Kao što je prikazano na slici, potrebno je isključiti zeleni (Green) i plavi (Blue) kanal boja,



tako da sa slike skidamo samo crvene tonove. Sada jednostavno “crtajte” po onim mestima slike gde ima suviše crvene boje i na kraju smanjite sliku rezultat je više nego dobar. Na ovaj način je lako odstraniti i druge crvene fleke od sunca koje se javljaju na vašim fotografijama.

~Miloš Popović



Programski jezik za početnike ili profesionalce?

Uvod u Piton (Python)

Guido van Rossum je tvorca ovog programskog jezika, a ime potiče iz serije BBC-ija "Monty Python's Flying Circus".

ŠTA JE PITON? OBJEKTNO ORIJENTISANI, IINTERPRETIRANI PROGRAMSKI JEZIK VISOGOK NIVOVA. PITON SA SOBOM DONOSI MOGUĆNOST POVEZIVANJA SA MNOGIM BIBLIOTEKAMA, A POŠTO JE NAPISAN U C-U POSTOJI I MOGUĆNOST PROŠIRIVANJA U SAMOM C-U ILI C++. PORED SVEGA TOGA POSEDUJE VEOMA ČISTU SINTAKSU KOJU JE VEOMA LAKO NAUČITI, TE KAO TAKAV JE PRAVI IZBOR ZA POČETAK PROGRAMIRANJA..

Sa Pitonom ste u mogućnosti da pišete proste skripte ili razvijate složene programe ili programske pakete. Sa njime će te lako razvijati Web servise, kontrolisati procese, obrađivati tekst ili XML dokumente. Piton poseduje sve internet protokole (HTTP, FTP, SMTP, POP ...) i omogućava pristup raznim bazama podataka (DB/2, MySQL, DCOraqlke, DCOra-kle2 i mnogim drugim). Za izradu Grafičkog korisničkog interfejsa (GUI) najčešće se koriste wxPython, PyQt i PyGTK, pored njih postoji i podrška za TK, Delphi i mnoge druge.

Osnovne odlike

- Jednostavnost

Piton poseduje jednostavnu sintaksu. Svako ko zna Engleski jezik će sigurno shvatiti veći deo programa samo čitanjem.

- Lakoća učenja

Možda stvar direktno povezana sa predhodnom, ali takođe veoma bitna za omasovljavanje.

- Slobodan i otvorenog koda

Da, Piton je 100 % slobodan, što je presvega bitno za njegov razvoj.

- Piton je jezik visokog nivoa (high-level)

U jednom napisanom Piton programu nemorate da se bavite stvarima poput memorije koju on koristi.

- Portabilnost

Svojstvo koje proističe direktno iz trećeg, tako da postoji u verzijama za sve poznate operativne sisteme (Windows, MAC, i naravno GNU/Linux i Unix). Jedan te isti byte code se može pokrenuti na svim platformama.

- Interpretirani

U ovom jeziku ne postoji poseban korak vezan za prevođenje, kompilaciju (compilation), nego se program izvršava direktno iz izvornog koda (source code). Piton često sam iz izvornog koda pravi takozvani byte code. Ovim se dobija na brzini ali i jednostavnosti jer nemorate da brinete oko stvari kao što su linkovanje nekih biblioteka itd.

- Objektno orijentisani

U Pitonu je moguće i takozvano proceduralno, funkcionalno i objektno orijentisano programiranje. U Pitonu sve je objekat.

- Proširivost

Ukoliko je to potrebno moguće je napisati program u C-u i onda ga kombinovati sa Pitonom.

- Velika biblioteka

Standardna biblioteka koja dolazi sa Pitonom je poprilično velika. Ovo se zove "batteries included" u filozofiji Pitona. Neke od najznačajnijih su Python Imaging Library, VTK, MayaVi, Numeric Python, ScientificPython.

Kratko poređenje sa drugim jezicima

- Perl

Perl je još jedan slobodan i veoma moćan programski jezik. Piton i Perl imaju neke sličnosti (Unix skripte itd.), ali imaju i neke razlike. Ako ste nekada videli ili možda pokušali napisati Perl i Piton skriptu razlika bi vam se sama pokazala. Takođe se može pronaći

mišljenje da je Perl prava stvar za iskusne programe koji su probali skoro sve i uvek im je nešto smetalo ili nedostajalo, dok je Piton idealan za početnike koji žele veoma lako da savladaju objekte i objektno orijentisano programiranje. Programi napisani u Pitonu u odnosu na one napisane

u Perlu su definitivno kraći, lakši za pisanje i shvatanje. Značajna prednost koji Perl možda ima u odnosu na Piton je biblioteka zvana "CPAN - the Comprehensive Perl Archive Network" i kako joj samo ime kaže to je ogromna kolekcija Perl modula. Jedan od glavnih razloga za ovo po mom mišljenju to što je Perl mnogo više u upotrebi od Pitona.

- Java, C++

Generalno Piton programi bi trebali biti sporiji od onih razvijenih

u Javi (pominje se negde 3-5 puta) ali je potrebno značajno manje vremena za razvoj. Ta razlika se može pripisati razlici novoa, naprimer Piton programeri nemoraju da se bave deklaracijom tipova i argumenata promenljivih. Uzmimo za primer računanje izraza a+b, Piton mora prvo proveriti objekat a i b i utvrditi njihov tip, koji nije poznat u vreme izvršavanja. Za C++ važi otprilike sve gore rečeno samo što je razlika u brzini sada 5-10 puta, ali takođe bitna stvar koja se pominje je da ono što Piton programeri urade za dva meseca C++ programerima je potrebno oko godinu dana. Svakako ovi programski jezici zajedno čine najbolju kombinaciju.

Piton u upotrebi

- Stabilnost

Možda je razumno postaviti pitanje stabilnosti, Koliko je Piton stabilan? Sa sigurnošću se može reći da je veoma stabilan, nova stabilna verzija izlazi na svakih 6-18 meseci i ovakav trend će se nastaviti i u budućnosti.

- Koliko se Piton koristi?

Na to pitanje je teško dati odgovor, jer je programski jezik potpuno slobodan i može se preuzeti sa različitih lokacija. Takođe svaka GNU/Linux distribucija dolazi sa nekom verzijom Pitona, a i mnogo stvari je napisano u ovom jeziku. Čuveni RedHat instaler (Anaconda) je u potpunosti napisan u Pitonu, a mnoge distribucije koriste ovaj, doduše malo izmenjeni ili ne instaler. Pomenuo bih još samo i Bleder projekat.

- Nadogradnja

Da, moguća je nadogradnja stare verzije Pitona novom. Počev od verzije 2.3 uključen je paket PySDDb, kao zamena za stari bsddb modul. Zbog te zamene procedura nadogradnje 2.2 verzije 2.3 je nešto komplikovanija ali nije teška. Zajedno sa Pitonom 2.3 dolaze dve skripte db2pickle.py i pickle2db.py, one se nalaze u Tools/scripts direktorijumu. Sa njima možete konvertovati vaše stare fajlove u novi format, na sledeći način

```
python2.3 <putanjado>/db2pickle.py database.db database.pck
sada treba preimenovati na sledeći način
mv database.db olddatabase.db
```

i na kraju

```
python2.3 <putanjado>/pickle2db.py database.bd database.pck
```

Kao što se vidi tačna komanda se razlikuje, odnosno može se razlikovati u zavisnosti od instalacije Pitona.

- Moduli

Moduli se koriste onda kada ne želite da pišete neke funkcije više puta, odnosno kada se neke funkcije ponavljaju na više različitih mesta. Modul je u suštini datoteka sa ekstenzijom .py (datoteka.py), koji sadrži definisane funkcije. Modul možete koristiti u nekom drugom modulu ili programu, sve što je potrebno je samo da ga pozovete. Za ovo u Pitonu postoji više načina:

```
import ime_modula
```

primetite da nema ekstenzije. Tada neku funkciju u tom modulu pozivate na sledeći način:

```
ime_modula.ime_funkcije
```

drugi način za pozivanje modula je

```
from imemodula import *
```

Sve što je potrebno je da se modul nalazi na određenom mestu, da ga Piton može pronaći. To može biti trenutni direktorijum, PYTHONPATH (/usr/lib/python2.3 ...) ili instalacioni direktorijum.

Za detaljnije informacije svakako treba pogledati "Python Library Reference"

Da se vratimo sa pitanje sa početka da li je Piton za početnike ili profesionalce? Apsolutno fleksibilan programski jezik koji podjednako mogu koristiti i jedni i drugi. Početnici koji žele da nauče programiranje ali i profesionalci koji žele brzo da razvijaju aplikacije.

Korisna adresa

www.python.org

~ Srđan Anđelković

Radionica

Srebrna lisica

Kako prilagoditi izgled Firefox pretraživača

TOKOM ISTORIJE, MNOGI LJUDI SU POKUŠAVALI DA LOVE NORMALNE LISICE, DA IM FARBAJU KRZNO U BOJU SREBRNE LISICE I DA IH TAKVE PRODAJU PO DESET PUTA VEĆOJ CENI. MI ĆEMO SE U OVOJ RADIONICI BAVITI SLIČNIM POSLOM, SAMO ŠTO ĆE NAM NAGRADA BITI LIČNA SATISFAKCIJA, A NE NOVAC.

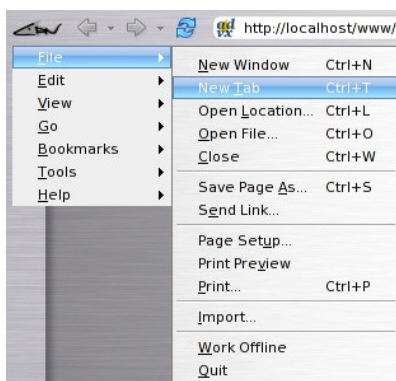
Ekstenzije

Jedna od najljepših strana Firefox pretraživača je mogućnost da ga promenite u toj meri da ga ni rođena majka (Mozilla fondacija) ne bi prepoznala. Na početku ovog teksta ćemo se pozabaviti nekim od interesantnijih ekstenzija koje se tiču izmene samog izgleda programa.

Compact menu

Pomoću ovog proširenja ćete moći da sklonite kompletan meni sistem programa i da ga "stavite" pod jedno jedino dugme. Imate mogućnost biranja da li želite da to dugme bude dugme sa stilizovanim sličicom ajkule ili da na njemu piše samo "Menu".

Ako niste zadovoljni sličicom ajkule, otvorite fajl compact.jar (Java Archive je zapravo ZIP arhiva sa promenjenom ekstenzijom) i zamenite fajl compact.png koji se nalazi u skin/classic/compact/ direktorijumu arhive.



Stop/Reload

Zbog čega bi ste imali dva dugmeta – jedno za Stop, a jedno za Reload – kad možete imati samo jedno kao što je to, na primer, u Safari pretraživaču za MacOS X. Dok se neka strana učitava, prikazivaće se dugme Stop, a u suprotnom Reload. Potrebno je napomenuti da ova ekstenzija ne izgleda lepo kad se koristi neka tema za Firefox koja je ne podržava. Sa podrazumevanom temom sve radi kao podmazano.

Tabbrowser Extensions

Ovo je jedno od najbolje urađenih proširenja pravljenih za Lisicu koje vam, pored ostalog, dodaje dugme za zatvaranje na svaki tab u programu. Ako vam nisu potrebne sve mogućnosti ove ekstenzije, a želite malo "x" dugme u desnom ćošku svakog taba, skinite Tab X proširenje.

CSS

Sada kad smo napravili nekoliko hirurških zahvata i sabili ceo meni i paletu sa alatkama (toolbar) u jedan red, možemo da počnemo sa farbanjem krzna. Za farbanje će nam biti potrebno malo znanja CSS-a

(Cascade Style Sheet). Ovaj princip možete koristiti i za menjanje izgleda Thunderbird klijenta za e-poštu samo što, u tom slučaju, morate sa interneta preuzeti DOM Inspector proširenje za Thunderbird.

Svaki element u oba programa podleže zakonima koje je propisao W3C (www.w3c.org) u CSS standardu. To znači da skoro svakom elementu grafičkog okruženja možete menjati font, pozadinu, okvir, margine... Ovde ćemo se pozabaviti nekim sitnijim trikovima tek da vidite koji je princip, pa da možete kasnije sami da eksperimentišete.

Pozadina

Prvo otvorite fajl userChrome.css koji se nalazi u chrome poddirektorijumu vašeg profila (za Firefox ~/ .mozilla/firefox/ime, i ~/ .thunderbird/ime za Thunderbird) ili ga kreirajte ako on ne postoji.

Dodajte sledeće redove u njega:

```
toolbar,
menubar,
toolbar#toolbar-menubar,
statusbar#status-bar,
hbox#browser,
#sidebar-splitter,
#sidebar-box {
    background-image: \
        url("bkg.png") !important;
    border: 0px solid black \
        !important;
}
```

bkg.png zamenite putanjom do fajla koji želite da bude prikazan u pozadini.

Obratite pažnju na to da se promene neće videti u programu dok ga ne restartujete.

Margine

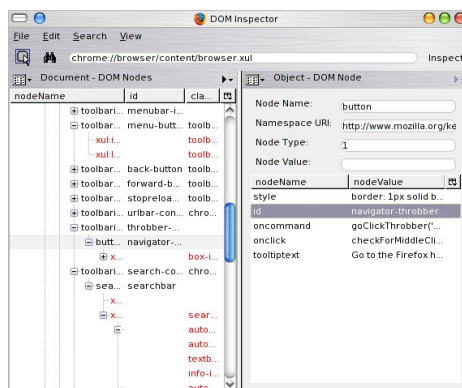
Ako želite da sklonite prazan prostor između dugmića u paleti sa alatima, u userChrome.css fajl unesite sledeće CSS naredbe:

```
toolbarbutton {
  padding: 0px !important;
  margin: 0px !important;
}
```

Naprednija podešavanja

Ako znate CSS, onda se ne morate zaustaviti na ovim, i na trikovima na koje naletite na internetu. Možete uraditi šta god poželite. Princip rada ću pokazati na sledećem primeru – dodaćemo okvir onoj maloj sličici koja se animira kada se učitava stranica a koja se zove trober (throbber).

Prvi korak svake izmene se sastoji u tome da nađete element kome želite da promenite osobine. To možete uraditi na sledeći način:



Pokrenite DOM Inspektora (Menu > Tools > DOM Inspector, ili kombinacijom tastera Ctrl+Shift+I). U inspektoru idite na File > Inspect a Window i tu odaberite prozor na kome se trober nalazi.

Sada idite na Search > Select Element by Click i posle toga pređite u glavni prozor i kliknite na trober. U DOM Inspektoru će biti obeležen traženi element – button čvor sa id atributom “navigator-throbber”. Kliknite desnim dugmetom miša na tabelu sa kolonama nodeName i nodeValue, pa kad se otvori meni, kliknite na Insert. Bićete upitani da unesete ime atributa koji želite da dodate. Tu unesite “style” (bez znakova navoda). Sada ćete biti u prilici da unesete CSS osobine za taj element. U našem ćemo tu ukucati “border: 1px solid black;”. Stavite OK i pogledajte vaš glavni prozor.

Ništa se nije desilo!? Nije. To znači da smo naleteli na element kome ne možemo da promenimo osobine. Sada dolazi na red tehnika pokušaja i promašaja – odabraćemo u DOM drvetu element koji je “otac” našeg elementa – to će biti toolbaritem čvor sa id atributom “throbber-box”. Probaćemo prethodni postupak na njemu – stavljamo style=“border: 1px solid black;” idemo na OK. Eto, dobili smo punu crnu ivicu debljine 1px (piksel).

Sada je došlo na red da te izmene zapamtimo. Otvorite userChrome.css i dodajte sledeći red:

```
toolbaritem#throbber-box { border: 1px solid black; }
```

Za one koji ne znaju CSS, reč koja se nalazi ispred “tarabe” označava ime čvora u DOM drvetu, a reč koja je iza označava vrednost id atributa čvora na koji primenjujemo stil koji se nalazi između vitičastih zagrada.

~ Ivan Čukić



Čuvanje Slobode pravnim sredstvima

Legum servi sumus ut liberi esse possimus* – deo 1.

Moramo li biti potčinjeni zakonima kako bi očuvali Slobodu?

GNU je iz ideje u stvarnost pretočio slobodu. Ipak, jednom ostvarena sloboda se mora štiti svim sredstvima jer tek tako sva njena suština dolazi do izražaja a trud velikog broja dobrih ljudi ne gubi na smislu.

GNUUUU!

Svoj logični početak ova priča ima u upoznavanju sa GNU projektom. Ulaziti u kompletnu analizu istorijske pozadine bi nas odvelo u stranputicu te ću se ovde samo na kratko zadržati na tome šta je to GNU i kako je nastao. Ozbiljnije razmatranje ostavljam nekom drugom za neku drugu priliku.

Cilj GNU (GNU's Not Unix) projekta je razvijanje slobodnog a Unix kompatibilnog softvera tj. operativnog sistema a svoju istoriju započinje davne 1983.

Mnogi ljudi pogrešno shvataju reč sloboda u kontekstu Slobodnog softvera. Ako kažemo "sloboda" to ne znači i besplatno. Za svoj Slobodni softver vi možete ali i ne morate platiti neku cenu. U svakom slučaju su vam na raspolaganju tri(četiri) različite slobode:

- 1) Sloboda da softver kopirate i delite sa prijateljima ili saradnicima.
- 2) Sloboda da program menjate po sopstvenoj volji i da imate uvid u izvorni kod softvera.
- 3) Sloboda da jednom izmenjen softver distribuirate i na taj način pomognete razvoju zajednice.

Do početka devedesetih je ostvaren osnovni cilj - razviti UNIX sličan operativni sistem. Do tog trena su sve komponente, sem kernela, bile razvijene. Onda je stvoren Linux, slobodni kernel, i označio početak prave GNU revolucije.

Ipak, GNU nije ograničen samo na operativne sisteme. Kao što je već navedeno, cilj jeste slobodan softver u svakom obliku. Slobodan softvera je praktično ograničena samo jednim - zakonima o patentima koji zabranjuju slobodu istog time što sputavaju 3(4) osnovne slobode koje promoviše GNU.

Strah od Slobode!

Ne bi bilo pošteno kada u ovoj priči ne bi bila pomenuta Inici-

jativa za Otvoreni kod (Open Source Initiative - OSI), jer ćemo se u priči o licencama jako često vraćati na njih. Neka mi svi poštovaoci GNU ideje oprostite ako smatraju da ih ne treba pominjati - sa pravne strane su oni jako važni, a ovo je ipak pravna strana slobode.

Suštinska razlika između ova dva pokreta leži u pogledu na vrednosti i u pogledu na svet. Za ljude u OSI-ju je pitanje softvera otvorenog koda praktično pitanje, a ne etičko. Jednom je neko to lepo postavio: "Otvoreni kod je metodologija razvoja, Slobodni softver je društveni pokret". Dakle, suštinska razlika leži u tome kako ko gleda na svet. Dok GNU pokret otvoreno govori o etici i slobodi, Otvoreni kod je tiši na tu temu jer smatra da će na taj način biti prihvatljiviji u poslovnom svetu. A poslovni svet ne voli previše priče o slobodi, moralu i etici. Neko će reći da se ne živi od ideja već od para. Tačno. Određena količina tihog rada može da koristi zajednici da bude bliža poslovnim saradnicima, ali niko ne sme da odstranjuje slobodu da se o slobodi govori i da se ona promoviše.

Ova dva pokreta se razlikuju u osnovnim principima što ih ne čini zaraćenim taborima. Zamislite ih kao dva politička tabora unutar jedne te iste zajednice. Baš zbog toga se neću libiti da u ovoj priči pominjem i koristim informacije koje nosi pokret

Otvorenog koda.

U nas se mnogo toga ne shvata ozbiljno. Na našu žalost, ni ideja slobodnog softvera nije baš najbolje shvaćena. Puno ljudi govori loše stvari o Slobodnom softveru a opet, svakog dana upotrebnim aplikacijama koje podležu nekoj od licencija proizašlih iz Opšte javne licence (GPL). Ironija svega toga je što o tom softveru pričaju sve najbolje. Čini se da kod nas ni ideja slobode nije najbolje shvaćena, a opet svi o njoj govore i svi traže neku slobodu. Dajte im je na dlanu i oni je neće videti.

Iza same ideje o Slobodnom softveru stoje sjajna pravna rešenja za očuvanje te iste slobode. Osnovno i, za nas, najvažnije jeste ono koje u sebi nosi Opšta javna licenca (GPL). Detaljnu proču o njoj ostavljam za sledeće brojeve.

Definition libertatis

Definisanje slobodnog softvera je pitanje slobode korisnika da upotrebljavaju, kopiraju, distribuiraju, proučavaju, menjaju i poboljšavaju softver. Tačnije, odnosi se na četiri slobode:

0. Sloboda da se program koristi u bilo koje svrhe.
1. Sloboda da se program prouči i prilagodi sopstvenim potrebama. Pristup izvornom kodu je preduslov toga.
2. Sloboda da se distribuiraju kopije kako bi se pomoglo zajednici.
3. Sloboda da se softver unapredi, da se poboljšanja izdaju kako bi se doprinelo zajednici. Pristup izvornom kodu je preduslov toga.

Program je deo Slobodnog softvera ako korisnici imaju sve navedene slobode. Prema tome, postoji sloboda distribuiranja kopija sa ili bez prepravki, besplatno ili uz naknadu, bilo kome i bilo gde.

Sloboda pravljenja prepravki na softveru podrazumeva da ste apsolutno slobodni da program prilagođavate sopstvenim potrebama, a da te prepravke ne objavljujete. Ako ih, pak, objavite, o tome ne morate nikog posebno obavestiti. Ova sloboda podrazumeva da za prepravke ne morate tražiti dozvolu.

Sloboda upotrebe programa podrazumeva da bilo koja osoba i bilo koja organizacija imaju pravo da softver koriste na bilo kom kompjuteru, za bilo koji posao bez obaveze da stupaju u kontakt sa tvorcem softvera.

Sloboda redistribucije podrazumeva pravo da se binarne kao i izvršne kopije programa dele. Ovo svakako uključuje i izvorni kod programa.

Sve ovo ne bi imalo smisla da nema pristupa izvornom kodu softvera. Dakle, taj pristup je preduslov Slobodnog softvera.

Ipak, određena pravila vezana za distribuciju mogu postojati dokle god nisu u suprotnosti sa centralnim slobodama. Primer za to je tzv. "Copyleft" koji podrazumeva da pri distribuciji softvera ne smete ni na koji način ograničiti centralne slobode korisnicima.

Slobodan softver ne znači i nekomercijalan softver. On mora biti dostupan i za komercijalnu upotrebu, komercijalni razvoj i komercijalnu distribuciju.

Pravila o tome kako zapakovati prepravljene softver mogu postojati dokle god ne krše četiri centralna pravila. Pri izradi licence autor takođe može zahtevati da dobije kopiju softvera koji ste menjali.

Kada govorite o slobodnom softveru, najbolje je da izbegnete priče o "davanju" ili "poklanjanju" svog softvera besplatno. Kao što vidite, nema tu nikakvog odricanja od svog dela ili besplatnog rada. Svoje delo štite baš tom slobodom koju postavlja GNU definicija. Vaše ime i delo, bez obzira na izmenu dela i njegovu upotrebu nisu ne smeju biti zanemarena, što OJL jasno i precizira. Ozbiljna pravna regulativa stvorena oko definicije Slobodnog softvera vam omogućava da svoje delo učinite slobodnim a opet zaštićenim.

Kao što je već napomenuto, ova sloboda se ne odnosi samo na softver već pokriva svaki domen ljudskog stvaralaštva. To će vam postajati jasnije kako budemo prolazili kroz GNU svet.

~ Marko Milenović