

Август 2013.



ЛИБРЕ!

Часопис о слободном софтверу

број

16

LiMux /Т еволуција



17. август
У LXDE табору су
одлучили да LXDE-Qt
споје са Razor-Qt-ом



22. август
The Document
Foundation је објавио
LibreOffice 4.0.5.



Creative Commons Ауторство-Некомерцијално-Делити под истим условима.



Слободни или власнички софтвер, питање је сада?

Иако је ЛиБРЕ! искључиво часопис о слободном софтверу, немогуће је не упоређивати слободна решења и власничке пандане. У овом броју идемо и корак даље; није нам довољно само упоређивање квалитета софтверских решења, него и однос квалитет/цена. Процењујући овај однос, неки европски градови су закључили да се исплати прећи на слободни софтвер. Њихова калкулација је да могу без губитка у квалитету прећи са власничког на слободни софтвер, и плус да уштеде део пара пореских обвезника. Сви се слажу да овакав подухват неких градова Европе није бесплатан, иако се користе бесплатна слободна софтверска решења, али проста рачуница је довела до процене да су трошкови преласка на слободни софтвер мањи од плаћања лиценци за власнички софтвер. Уштеда је директни бенефит од преласка на слободни софтвер. Посредни бенефит је ангажовање домаћих стручњака и локалних заједница, тако да сва ангажована средства остају у земљи.

Не постоје докази о директној вези између страних инвестиција у нашу привреду и плаћања лиценци за власнички софтвер. Страни инвеститори ће уложити новац у нашу привреду само ако имају неке интересе, типа смањења трошкова одржавања, или због процене да наше тржиште (читај, квалитетни стручњаци) има потенци-

јал да донесе профит инвеститору. Уколико немају никакву конкуренцију на нашем тржишту, неће имати претераног мотива да се боре за ово тржиште, а самим тим инвестиције ће бити само типа смањења трошкова које у случају продаје лиценци нису велике.

Минхенски пројекат *LiMux* је велики пројекат, који је обухватио око 15.000 рачунара. За илустрацију величине овог пројекта дајемо податак да је својевремено Влада Србије и СРЈ, за потребе рачунара у државним органима и државним предузећима у целој земљи, купила 30.000 лиценци. Процена града Минхена је да је преласком на *LiMux* направљена велика уштеда. Нису урачунати само трошкови лиценци него и хардвера који би морао бити замењен или надограђен преласком са старијих на новији власнички софтвер. *LiMux* је знатно умањио трошкове везане за хардвер, јер добро ради и на старијим машинама које још нису за бацање.

Жеља нам је да овим минхенским примером барем покренемо питање о којем треба добро размислити.

Један бенефит од слободног софтвера је уштеда, а други је зарада. Када кажемо комерцијални софтвер, сви помисле одмах на власнички софтвер. Појам „комерцијални софтвер” није искључиво везан за власништво; и



слободни софтвер може бити комерцијални, али много ређе него што је то случај код власничког софтвера. Слободни софтвер се много чешће ослања на донације. Код нас се донирање у новцу погрешно тумачи као прошење. У свету је то легитиман начин прикупљања средстава за покривање трошкова, па и зараду. Људи на западу су навикли да све што користе, а није њихово, и плате. Нарочито осећају обавезу да плате нешто што им је донело неки профит. Због тога радо донирају новац пројектима слободног софтвера, ако га користе и ако им олакшава живот. Када ће и код нас таква пракса заживети, не знамо. Покрећемо и ту тему, па ћемо видети шта ће се десити.

И даље вас позивамо да се придружите нашем тиму, дајете ваше сугестије, критике или можда и похвале; немојте претеривати са похвалама. У ту сврху користите наш форум на адреси:

<https://libre.lugons.org/index.php/forums/>
или нам пишите на већ познату адресу електронске поште libre@lugons.org.

До читања.

ЛИБРЕ! тим

Моћ слободног
софтвера



Број: 16

Периодика излажења: месечник

Главни и одговорни уредник:
Никола Харди

Извршни уредник:
Александар Станисављевић

Лектура:
Јелена Мунћан
Александра Ристовић
Александар Божиновић
Александар Станисављевић

Редакција:

Горан Мекић	Сандрина Димитријевић
Жељко Шарић	Александар Тодоровић
Данило Ђокић	Милован Кривокапић
Дејан Чугаљ	Далибор Богдановић
Дарко Стантић	Александар Брковић
Иван Булатовић	Гаврило Продановић
Златан Васовић	Михајло Богдановић
Стефан Ножинић	Александар Весић
Жељко Попивоца	Владимир Цицовић
Бојан Богдановић	

Сарадници:

Џони Промис	Никола Ненадић
Ненад Мијатовић	Тамара Ђорђевић

Графичка обрада:

Дејан Маглов	Силвија Силађи
---------------------	-----------------------

Дизајн:

Младен Шћекић	Зоран Лојпур
----------------------	---------------------

Контакт:
IRC: #floss-magazin на irc.freenode.net
E-пошта: libre@lugons.org

<http://libre.lugons.org>



ЛиБРЕ! вести стр. 6



Пулс слободе стр. 8

LiMux - IT еволуција стр. 8



Представљамо стр. 12

antiX 13.1 „Luddite“ стр. 12



Tilda терминал емулатор стр. 16

Како да? стр. 18

Мала школа:
Sigil 0.7.2 (4. део) стр. 18

Исправно куцање стр. 23

Ослобађање стр. 27

Како дати допринос програмима отвореног кода иако нисте програмер? стр. 27

Донацијама можемо да направимо разлику стр. 29

Интернет мреже и комуникације стр. 32

Open Wonderland
Направите ваш виртуелан свет стр. 32



Сервер стр. 36

oVirt
Web апликација за управљање платформом за виртуелизацију стр. 36





ISPConfig
hosting control panel

стр. 42



Сам свој мајстор

стр. 45

Git (8. део)
Команде

стр. 45



Mixxx (1. део)
Мултиплатформски DJ програм
отвореног кода

стр. 47



Хардвер

стр. 50

Ардуино контролер (7. део)

стр. 50



ЛИБРЕ! пријатељи





VLC 2.0.8 „Twoflower“

4. август



VLC 2.0.8 „Twoflower“ је мала надоградња верзије 2.0 популарног медија плејера која, између осталог, са собом доноси подршку за већи број

формата.

Користан линк: <http://j.mp/14a5pXV>

Air Conflicts: Pacific Carrier

5. август



Нова игрица, симулатор лета авиона, *Air Conflicts: Pacific Carrier*, долеће на Linux на крилима Steam-а.

Користан линк: <http://j.mp/15JwqkN>

gNewSense 3.0 на бази Debian-а

7. август



gNewSense је са верзијом 3.0 почео да користи Debian као основу, уместо Ubuntu-а којег је до сада користио.

Користан линк: <http://j.mp/15gcjHt>

Windows 3.11 u Linux 3.11

11. август



Пре 20 година објављен је Windows 3.11, а данас је објављен Linux 3.11-rc5.

Користан линк: <http://j.mp/13eG0yг>

ZTE Firefox OS smartphone

12. август



ZTE Firefox OS smartphone глобално доступан по цени од 80 долара.

Користан линк: <http://j.mp/17KrN8g>

Debian слави рођендан

16. август



Срећан ти 20. рођендан, Debian!

Користан линк: <http://j.mp/18zHIZu>

Снајажу се LXDE u Razor-Qt

17. август



У LXDE табору су одлучили да LXDE-Qt споје са Razor-Qt-ом, при чему ће ресурси из оба пројекта

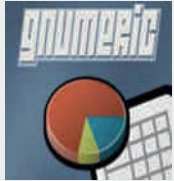


бити максимално искоришћени.

Користан линк: <http://j.mp/13sEFE4>

Gnumeric 1.12.5

20. август



Објављен је *Gnumeric 1.12.5*, канцеларијски програм за рад са таблицама.

Користан линк: <http://j.mp/14XRTd5>

LibreOffice 4.0.5

22. август



Са више од 90 исправљених буба, *The Document Foundation* је објавио *LibreOffice 4.0.5*, скуп канцеларијских програма.

Користан линк: <http://j.mp/16Qf7Se>

calibre 1.0

23. август



7 година након првог и годину дана након последњег, објављено је 1.0 *calibre* издање са многобројним побољшањима.

Користан линк: <http://j.mp/15hg8Am>

SolydXK 201308

23. август



Објављен је *SolydXK 201308*, дистрибуција базирана на *Debian testing*-у са *Xfce* и *KDE* окружењем радне површи.

Користан линк: <http://j.mp/14Lr5H4>

Slackel 4.0

24. август



Објављен је *Slackel 4.0 „Openbox”* издање, лагана *Slackware* базирана дистрибуција.

Користан линк: <http://j.mp/1577D6T>

ЛИБРЕ! пријатељи

LUTHERUS

Et in Arcadia ego!



Think about this



ICT časopis

ictcasopis ict.edu.rs



LiMux - IT еволуција

Аутор: Александар Весић



LiMux

Die IT-Evolution

Већ 10 година је прошло од настанка *LiMux*-а, обимног пројекта Градске управе града Минхена, чији је циљ успешна миграција са *Microsoft*-овог *Windows*-а и *MS Office*-а на слободни софтвер.

Пројекат *LiMux*



LiMux је један од *Linux* деривата који је развијен од стране запослених у одсеку за информатику Градске управе града Минхена. Сам појам је изведен од речи *Linux* и *München*. У међувремену је *LiMux* инсталиран на око 15.000 рачунара на којима раде запослени у Градској управи.

Сам циљ пројекта је да се стекне независност од произвођача софтвера и оперативних система, као и да се избегну прескупе лиценце за подршку, чиме би се постигла уштеда и спречило непотребно трошење новца пореских обвезника. Сам повод за настанак пројекта био је престанак подршке од стране *Microsoft*-а за *Windows NT4* крајем 2003. године, тако да се Градска управа Минхена морала побринути за неопходну замену за тада коришћени *Windows NT*. Након првих гласина о могућности преласка Градске управе на *Linux*, у зиму 2003. у Минхен је стигао тадашњи шеф *Microsoft*-а *Steve Ballmer*, који је на састанку са високим градоначелником Минхена *Christian*-ом *Ude*-ом покушао да утиче на то да се град Минхен ипак одлучи на наставак сарадње; безуспешно, био је то почетак настанка „Пингвина из Минхена”.

10 година после

На овогодишњем *LinuxTag*-у 2013, који је одржан у Берлину, руководилац *LiMux* пројекта *Peter Hofmann*, сумирао је протеклих 10 година од настанка пројекта. По његовим наводима, пројекат се приводи крају у току ове године, и самим тим отпочиње редовна употреба система на свим рачунарима Градске управе. Сам *LiMux* је пустио своје корене и скоро да не постоји опција да у



будућности дође до повратка на комерцијални софтвер. Град Минхен остаје при својој студији изводљивости по којој је Градска управа преласком на *Linux* уштедела 10 милиона евра, и одбацује наводе и тврдње *HP*-а и *Microsoft*-а, критички наводећи да њихова студија није урачунала све цене лиценци за производе од *Microsoft*-а који би били неопходни за неометани рад, нити је урачунало шта је све урађено од стране одсека за информатику (њихово радно време, измене, додатни модули, прилагођавање крајњем кориснику, итд.).



На *Open-IT Summit*-у 2013, који се паралелно одржавао са *LinuxTag*-ом, *Jutta Kreys*s (*LiMux* IT архитекта), причала је о практичним тешкоћама приликом преласка на *LiMux*. На самом почетку су појединачни одсеци и одељења информатике при различитим секторима Градске управе објављивали свако за себе тендере за апликације које нису биле компатибилне са *Linux*-ом. У међувремену је сарадња повећана, и сектори сада раде координирано уз помоћ централног *IT* одсека, и сви

заједно раде на развоју и побољшању *Client*-а. Највише главобоље задају тестови на *Base Client*-у, јер због децентрализоване и комплексне структуре није могуће једноставно понудити надоградње и замолити кориснике да ажурирају софтвер.

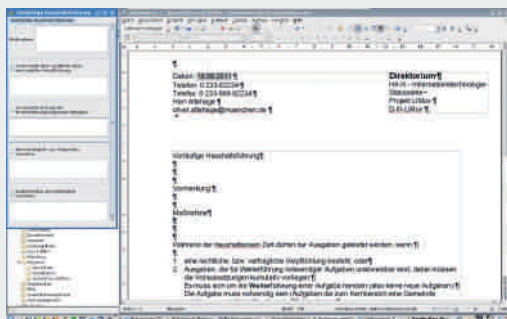
Позитивна страна поделе на међусобно независне секторе и одељења управе је да поједини сектори редовно обавештавају централни одсек о отклоњеним сметњама или, на пример захтевима корисника који су успешно оптимизирани, те је са тим аргументима могуће убедити остале одсеке да изврше ажурирање. Комплексна динамика развоја је зато разлог да за некога ко са стране посматра, примењена софтверска решења делују застарело. Тренутна верзија 4.0 је базирана на *Ubuntu*-у 10.04 *Lucid Lynx*, док је окружење радне површи *KDE* 3.5.

Развојни тим планира да у року од 1 године сви одсеци пређу на употребу *Ubuntu*-а 12.04 који би био обезбеђен са подршком до 2017. године. Окружење радне површи би притом било *KDE SC* 4.8 које би изгледом подсећало на *KDE* 3.5, да би запосленима олакшали прилагођавање. Такође ће бити извршена миграција са *Open Office.org* на *Libre Office*, и то би са новом верзијом *Client*-а била верзија *LibreOffice* 4.0.

Оно што недостаје *LiMux*-у је *Community* стратегија, јер иако град Минхен има свој *GitHub* налог и своју самостално развијену екстензију за обраду формулара у *OpenOffice.org*-у и *LibreOffice*-у, названу *WollMux*, однос према *Upstream* и *Community* пројектима није довољно



јасан. Информатички одсек је успео да измени неке мање функције у *Libre Office*-у за подршку *MS Office Open XML*-у, да би олакшали комуникацију са *EU* администрацијом, али то није извршено путем пројеката већ путем постављања готових закрпи у *Bugtracker*. Поред тога би било једноставно омогућити интерно спакован софтвер путем сопствених складишта програмских пакета колико то лиценца омогућава. Тако на пример *LiMux* користи *Extended Support Release* издање од *Mozilla-e*, које код *Ubuntu-a* није на располагању као пакет.



На самом крају нам господин *Hofmann* поручује да миграција није завршена у потпуности, и да иако у октобру 2013. године пројекат престаје ипак су потребне многе измене. Додатни сервери



се морају преbacити на *LiMux* и потребно је развити неопходне алате за рад службеника. Да би *LiMux* пројекат имао успеха, мора се формулисати јасна *open source* стратегија и однос са заједницом корисника слободног софтвера.



Andreas Heinrich из IBM-а и *Peter Hofmann*

Не би било на одмет да поменемо да постоје слични пројекти у свету, као на

пример у Амстердаму (*Open Amsterdam*), у шпанској Сарагоси (*AZLinux*), у Бечу (*Wienux*) и слично. Нама преостаје да се надамо да ће и наши званичници уочити предности слободног софтвера и сличну иницијативу покренути и на нашим просторима.

Корисни линкови:

- [1] <http://en.wikipedia.org/wiki/LiMux>
- [2] <http://www.muenchen.de/rathaus/Stadtverwaltung/Direktorium/LiMux.html>
- [3] <http://www.wollmux.net/wiki/Hauptseite>
- [4] <http://www.muenchen.de/rathaus/Stadtverwaltung/Direktorium/LiMux/Die-Migration.html>
- [5] <http://www.h-online.com/open/news/item/Linux-brings-over-EUR10-million-savings-for-Munich-1755802.html>

FIRST CONTACT - BALCCON 2K13

06 - 07 SEPTEMBER 2013

[HTTPS://BALCCON.ORG](https://balccon.org)

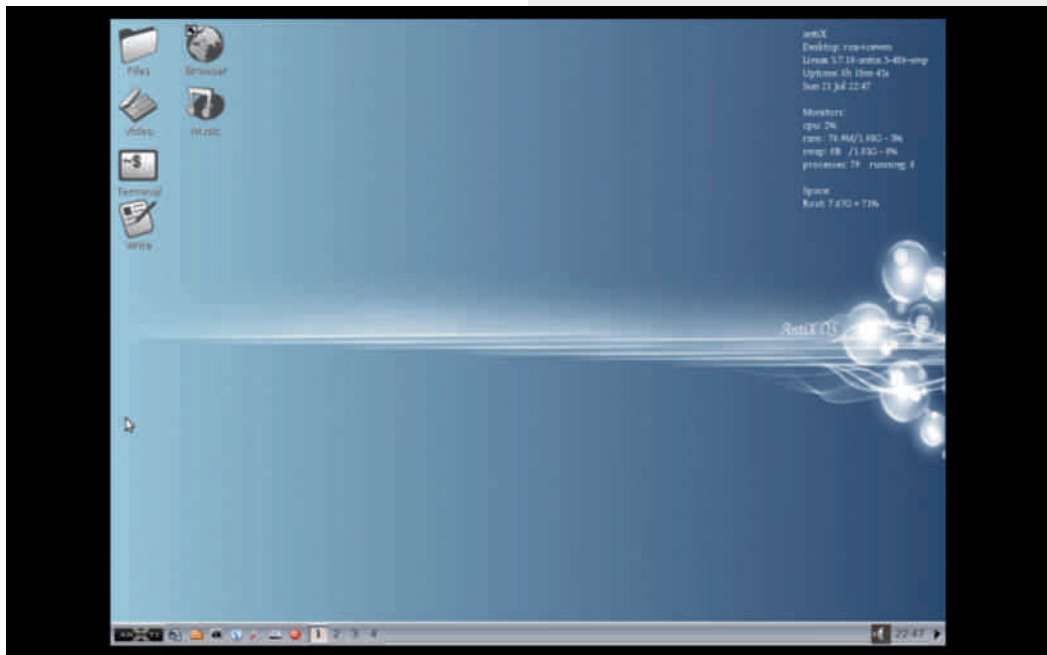
FIRST HACKER CONGRESS IN SERBIA

HACK, PLAY, LEARN, SOCIALIZE



antiX 13.1 „Luddite“

Аутор: Александар Брковић



antiX, још једна дистрибуција намењена првенствено за употребу на старијем хардверу, компатибилна са *Intel-AMD* системима. Употребљива је на рачунарима **PII** и **PIII** класе. За рад на старом хардверу, потребно је минимално 128 [MB] *RAM*-а, за инсталацију на тврди диск довољно је 2,2 [GB] расположивог простора. Може се користити и у *live* режиму, за спасавање података са рачунара.

За разлику од осталих малених и лага-

них дистрибуција, *ISO* датотека *antiX*-а је тешка 675 [MB], што резултира мноштвом преинсталираних програма и потпуном функционалношћу при коришћењу. Будући да се базира на *Debian*-у, није намењен апсолутним почетницима. Потребно је извесно предзнање да би се из *antiX* дистрибуције извукао максимум.

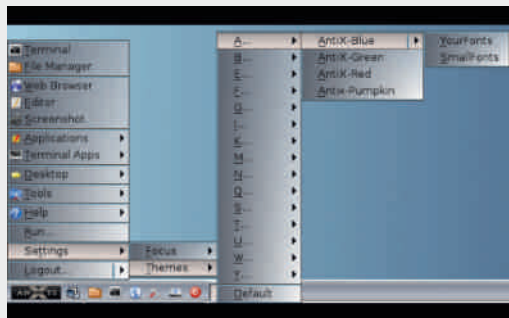
Последње издање *antiX* дистрибуције долази у верзији за 32 и 64 бита, и у примени је показало пристојну интег-



рацију са хардвером новије прои-
зводње. *Touchpad*, резолуција екрана,
звучне картице и тастатура су уредно
откривени и не захтевају додатну
интервенцију корисника.

Live mode

Приликом покретања *live* сесије,
дочекаће вас *iceWM desktop* са плавом
позадином, *conky* апликацијом у
горњем десном углу, и иконицама
доступних програма које се налазе са
леве стране радне површине. Панел се
налази на дну екрана, са менијем за
програме на левој и индикаторима на
десној страни. Једноставан и исполи-
ран интерфејс који се може при-
лагодити или променити прединстали-
раним темама. Промене нису драсти-
чне, али је број доступних тема
импозантан. Поред већ поменутог
iceWM управника прозора, доступни су
Fluxbox и *JWM*. Корисницима неће бити
тешко да изаберу своју подразумевану
комбинацију и да прилагоде изглед
дистрибуције по сопственом укусу. У
односу на остале *Linux* дистрибуције из
ове категорије, *antiX* даје сасвим
пристојан и исполиран *desktop*.



Приликом рада у *live* режиму, *antiX* у

одређеним случајевима тражи лозин-
ку, посебно током конфигурације
интернет конекције. *Root* лозинка је
„root”, а корисничка лозинка је „demo”.

Инсталација

Графички инсталер је прилично пре-
гледан и једноставан за употребу. На
самом почетку процеса инсталације
врши се одабир складишта про-
грамских пакета који ће се користити
након инсталације. Доступне варијанте
су *Wheezy*, *Testing* и *Sid*. *GParted* ће
одрадiti свој део посла око припреме
тврдог диска и израде партиција.

Приликом процеса инсталације, са
леве стране, у инсталеру, доступан је
текст који увелико олакшава сам
поступак, нудећи објашњења поје-
диних корака у процесу инсталације.
На самом крају долази део где се
одређује име система, врста тастатуре,
врши се унос временске зоне,
локализација система, те одређивање
корисничког имена и лозинке.

antiX тражи две врсте лозинке, адми-
нистраторску која је нужна за адми-
нистрацију система и покретање
кључних апликација, и корисничку за
пријаву на систем.

На крају долази инсталација *Grub2*
програма, за покретање система. *Grub2*
је у стању да детектује остале системе
инсталиране на рачунару. Ово је
последњи корак у инсталационом
процесу и систем је спреман за прво
покретање.



Актуелно издање

Након покретања система, *antiX* одмах детектује LAN конекцију и омогућава повезивање на интернет. Међутим, за оне кориснике који имају WiFi конекцију, односно бежично повезивање, настају проблеми при покушају повезивања на интернет. WiFi користи бежично повезивање преко програма *Wicd*, али у конкретном случају није могуће извршити иницијализацију интернет конекције. Решење овог проблема је успостављање везе преко програма *Ceni*, директно из терминала. Након уношења потребних параметара веза се успешно иницијализује. Подржан је и 3G широкопојасни интернет. Успостава везе се врши преко програма *GNOME-PPP* или *GPRS-EDGE-UMTS-Panel*. Довољно је унети корисничке параметре вашег провајдера и након тога се повезати на интернет.

За администрацију на систему постоји пар веома корисних програма:

- *antiX Control Center* – доноси низ функционалности и могућности за подешавање система, почев од најједноставнијих опција, као што је промена позадине на систему, подешавања WiFi мреже, па до могућности промене *autologin* опције, и омогућавања аутоматске пријаве на систем. Погодан је за кориснике који би да избегну мукотпан рад у терминалу, кад је подешавање система у питању. *antiX Control Center* на једном месту интегрише низ једноставних и сложених подешавања; за крајњег корисника веома корисна алатка.

- *Metapackages installer* – од велике помоћи уколико сте нови корисник у *Debian* базираним дистрибуцијама. Помаже вам да инсталација пакета буде крајње једноставна. Пакети су категорисани у логичке целине одакле се врши селекција и инсталација потребних пакета. Међутим, када једном инсталирате одређени пакет, код поновног покретања апликације не постоји могућност да се провери да ли је пакет инсталиран или не. Када би био уклоњен овај недостатак, програм би додатно добио на квалитету и функционалности.
- *Synaptic Package Manager* – подразумевани GUI програм за преузимање и инсталацију апликација из *Debian* складишта програмских пакета; најбољи алат за управљање пакетима у смислу брзине и решавања зависности; корисницима познат из других дистрибуција базираних на *Debian-у*.

Апликације

Упркос томе што је лагана дистрибуција, *antiX* садржи свеобухватан списак апликација. Постоји много прединсталираних програма, што ову дистрибуцију издваја од осталих, смештених у исту класу. Набројаћемо неке програме који се налазе на систему одмах након инсталације:

- *Office: LibreOffice 4.0.3.3 Calc, Draw, Impress, Writer, ePDF viewer, Osmo, Ted rtf text processor.*
- Интернет: *Iceweasel 22.0, Dillo, elinks, Links 2, Ceni, Clawsml, gFTP, GNOME*



PPP, Pidgin IM, py Neighborhood, Transmission.

- Graphics: Gcolor2, gtkam digital camera browser, Mirage, mtPaint, Presentation, screenshot, xsane scan.
- Мултимедија: Asunder CD ripper, Gnome MPlayer, Goggles music manager, GTK YouTube viewer, guvcview webcam, Imagination, simple burn, streamtuner 2, winFF.
- Accessories: Archive manager, clipit, leafpad, live usb, Lucky backup, root terminal, xarchiver, gdebi package installer, Rox term.

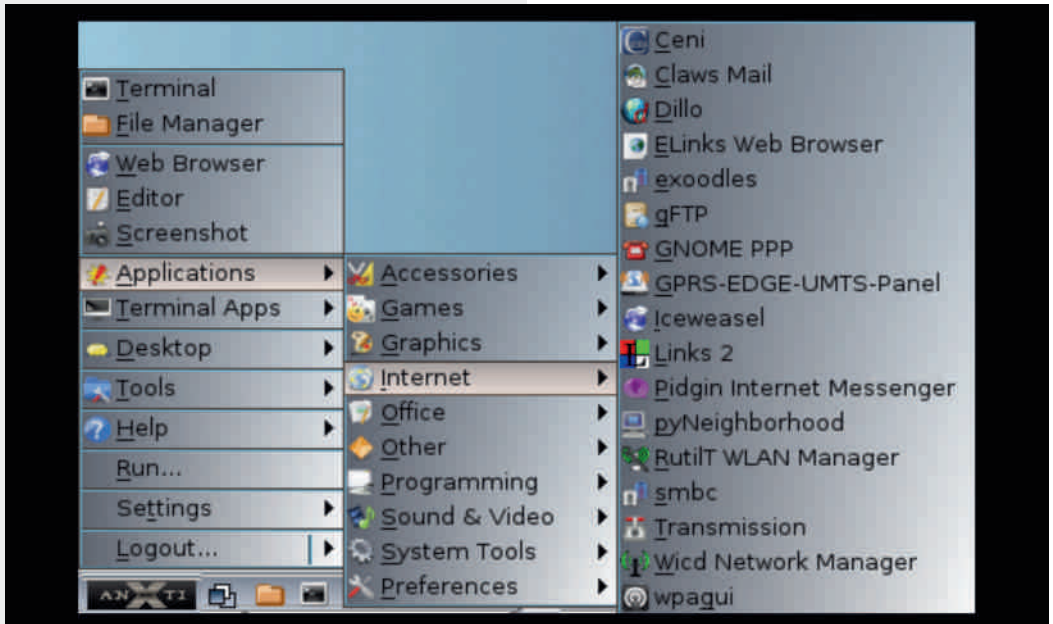
Листа пакета је импозантна, и покрива највећи део потреба корисника. Међутим, неке од апликација нису подразумевано лагане, као што је случај са LibreOffice програмом, те Iceweasel web прегледачем. Корисницима предложимо да изаберу нешто лаганија алтернативна решења уколико се

испостави да су ове апликације претешке за хардвер који користе. Напоменућемо и то да су прединсталирани и мултимедијални кодеци и Adobe flashplugin, те је могућност гледања филмова и видео записа доступна већ од live режима.

Уколико још увек негде чувате ваш стари рачунар који тренутно није у употреби, искористите ову дистрибуцију, инсталирајте је на ваш стари хардвер и уверите се у чињеницу да не постоје границе када је употребљивост Linux дистрибуција у питању.

Корисни линкови:

- [1] http://antix.mepis.org/index.php?title=Main_Page
- [2] <http://mylinuxexplore.blogspot.com/2013/07/antix-131-luddite-review-superb.html>





Tilda терминал емулатор

Аутор: Гаврило Продановић

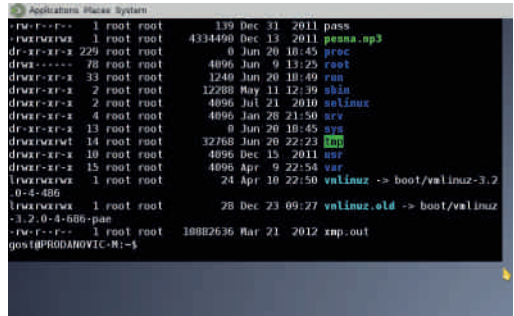
Сви ми који смо некада озбиљно играли пуцачине (или их и даље играмо), знали смо да користимо конзолу која се обично отвара на тилду. *Tilda* терминал емулатор нам баш тај стари доживљај доноси у наше *Xorg* окружење. Када притиснемо *hotkey*, он искочи и спреман је да му унесемо нашу команду и да је одмах изврши. Веома је zgodно, пошто сви знамо колико је терминал потребан у *Linux*-у.

Tilda емулатор је писан са *GTK+* библиотекама, тако да ће се одлично уклопити уз *GTK* окружења радне површи као што су *MATE* или *Xfce*. Када први пут покренемо *Tilda* емулатор, отвориће нам се чаробњак за конфигуравање у којем нас чека велики број опција распоређених у седам језичака. У првом језичку затичемо основне опције. Ту можемо да изаберемо тип и величину фонта који ће бити коришћен, да ли желимо да *Tilda* емулатор буде приказан на свим радним површима и да ли да увијек буде на врху, изнад свих прозора. Ту се налази и опција да изаберемо позицију језичака, јер *Tilda* емулатор подржава рад са више језичака. Такође, ту је *checkbox* којим ћемо потврдити да ли желимо да *Tilda* емулатор при стартовању буде сакривен. У следећем језичку можемо

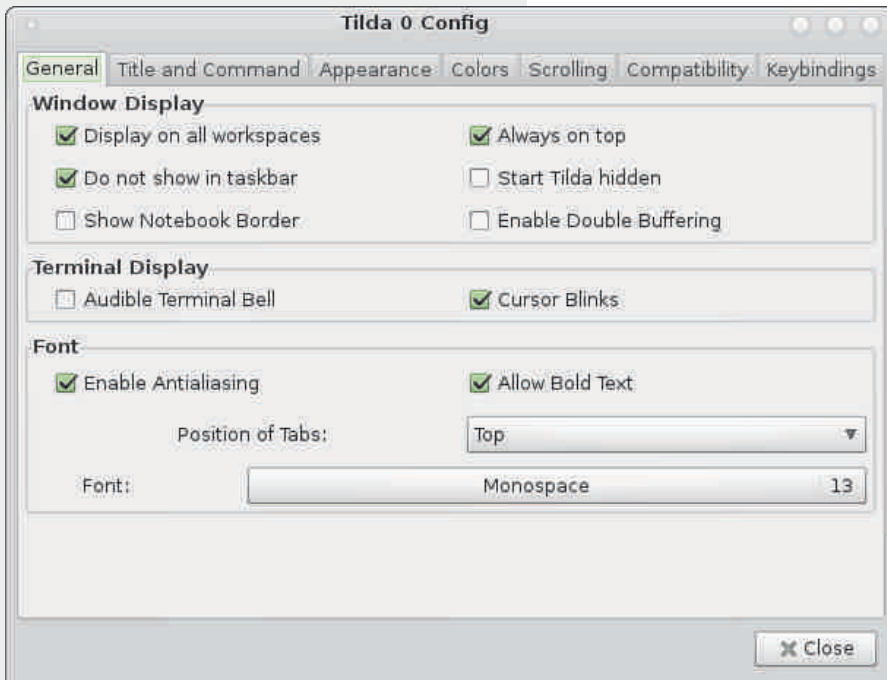
подесити наслов, а такође можемо изабрати да покренемо неку другу команду уместо *shell*-а. *Tilda* емулатор подржава отварање *web* линкова, па можемо изабрати који *browser* желимо да користимо. У следећем језичку који је назван „Appearance“, налазе се подешавања везана за терминал. Овде можемо промјенити висину и дужину терминала коју можемо задати у пикселима или у постоцима у односу на величину екрана. Такође можемо промјенити позицију која је подразумевана, да се *Tilda* емулатор појављује у горњем лијевом углу. Од додатних ствари можемо укључити провидност и ставити неку слику за подлогу уместо подразумеване црне боје. Такође, можемо укључити и анимирано извлачење терминала. У следећем језичку нема много да се изабере, а то је боја текста и боја подлоге. У „Scrolling“ језичку ћемо подесити да ли желимо да нам се *scrollbar* приказује и са које стране да се прикаже. Ту можемо промјенити и величину *scrollback buffer*-а. „Compatibility“ језичак ћемо прескочити, јер се ту налазе неке досадне опције (прим. аут.) и скочићемо до задњег језичка који се назива „Keybindings“. То је и најважнија опција од свих, јер се ту подешава *binding* за отварање терминала, која је подразумевано постављена на **F1**. Такође, ту

је и дугме помоћу којег можемо лако да уграбимо и неку другу комбинацију, ако нам не одговара подразумевљана.

После подешавања можемо напокон да се подмладимо и да почнемо да користимо *Tilda* терминал емулатор. Потребно је једно вријеме навићи се да ту постоји терминал, ако га покренемо при ауто старту графичког окружења. Но, после тога постаје незапамћив, и користимо га као *launcher*, а уједно престајемо користити **ALT+F2** комбинацију. Ако се икада деси да вам један *Tilda* терминал није довољан, можете покренути више различитих терминала са различитим прецицама са тастатуре.



Слика 2 *Tilda* терминал



Слика 1 *Tilda* конфигурација



Мала школа: *Sigil* 0.7.2 (4. део)

Форматирање текста

Аутор: Дејан Маглов

Увод

У прошлом броју смо вам препоручили да припремите своју публикацију у неком погоднијем програму за обраду текста. Најбољи слободни текст процесор за то је *LibreOffice Writer*, мада постоји још много слободних алтернатива (*OpenOffice Writer*, *Calligra Words*, *KOffice Kword*, *Abiword* и слични).

Предности припреме у текст процесору:

1. познато окружење,
2. комфоран рад,
3. лакше форматирање и
4. постојање речника српског језика, што омогућава брзу проверу исправности уписаног текста (енгл. *spellchecking*).

Иако је могуће у *ODT* датотеку осим текста поставити и форматирати илустрације, то у припреми *ePUB*-а немојте чинити јер их *Sigil* неће аутоматски препознати и увести из *ODT* датотеке.

Sigil нема функцију аутоматског увоза из било којег екстерног документа, као што смо рекли у прошлом броју. Без обзира на то, одлично препознаје

скоро сва форматирања урађена у текст процесору, иако текст убацујемо стандардном процедуром копирај/налепи (*copy/paste*). Форматирање није дословно преписано из текст процесора, али је довољно добро за почетак.

Оно што може да буде проблем:

- **Enter** у празном реду у процесору изгледа као проред, *Sigil* га препознаје као празан параграф. То ће довести до вишка прореда.
- Промену фонта у једном документу *Sigil* неће препознати. *Sigil* користи само један фонт док га не научите другачије.
- Ако у тексту имате више нивоа индексираних и неиндексираних листи, могуће да *Sigil* неће препознати те нивое (све ће бити у истом нивоу), па ћете морати исправљати те листе.

Могуће да има још неких форматирања текста које *Sigil* лоше тумачи, али за сада их нисмо приметили. До сада нисмо користили специјалне математичке карактере (формуле), па нисмо сигурни да ли и њих *Sigil* без посебних подешавања може да прикаже (аутор овог текста сумња у то).

Ма колико велика публикација била,



треба је целу убацити у само једну *HTML* страницу и тако у целини уредити. Чак сматрамо да је треба допунити и илустрацијама пре поделе на више одвојених *HTML* страница.

Није пожељно да крајња публикација буде само једна огромна *HTML* страница због лакше навигације и учитавања страница. Што су странице мање, *ePUB* читач ће лакше учитавати странице и читање публикације ће бити брже, без заостајивања. Зато, након комплетног форматирања и илустровања те велике *HTML* странице, предлажемо да је поделите по целинама. Најбоље да то буде подела по чланцима (ако су то целине за себе) или по поглављима. За поделу користите алат *Split At Cursor (CTRL+RETURN)* који дели постојећу *HTML* страницу на две, на месту где се тренутно налазио курсор.

Новонастала *HTML* страница има исти **<head>** таг, као и страница од које је подељена, што значи да ће имати иста подешавања текста; једино се дели **<body>** таг. Због овог препоручујемо дељење велике *HTML* странице на самом крају рада на *ePUB*-у.

Од ове поделе не зависи садржај публикације који служи и као навигација кроз публикацију. Сам садржај може бити и сиромашнији и богатији од броја *HTML* страница, али о томе ћемо причати на самом крају ове мале школе.

Форматирање у *Sigil*-у

И у овом броју ћемо направити малу

дигресију од самог програма *Sigil*-а; заправо, ово и није права дигресија. Форматирање можемо препустити аутоматици *Sigil*-а, али онда нећемо имати потпуну контролу. Препуштање *Sigil*-у да води рачуна о форматирању, доводи до тога да је сво форматирање унутар *HTML* странице, а не у посебној датотеци са стиловима. Ако направимо више *HTML* страница, у том случају ћемо тешко успети да обезбедимо једнообразност форматирања кроз цео документ.

Само централизовано форматирање обезбеђује једнообразност кроз цели документ јер накнадне промене у форматирању се аутоматски примењују на све странице.

Прошли пут смо поменули основе *HTML*-а, а овог пута ће бити мало речи о *CSS*-у (*Cascading Style Sheets*), централизованом каскадном стилизовању *HTML* страница.

CSS

Централизовани стил је зато што су дефиниције стилова свих елемената *HTML* докумената у једној датотеци, и важи за све *HTML* странице једног документа који су са њим у вези.

Каскадни стилови су због тога што неки стилови вреде у целом документу, али само док се не дефинише неки изузетак који важи за неки одређени елемент. На пример, цео текст може бити формиран тако да буде исписан *Arial-regular* фонтом тако што се дефинише да **<body>** таг буде исписан овим фонтом. Ово важи док



се унутар **<body>** тага не појави таг, на пример, **<i>**, који је дефинисан тако да мења фонт тог тага у *Arial-italic*. Надаље, и понеки **<i>** таг може да буде посебно форматиран ако га идентификујемо атрибутом, на пример **<i id="naslov">** или **<i class="podnaslov">**. Овако идентификовани тагови могу имати своје посебне особине (другу боју текста, другу величину слова). Разлика између атрибута **id** и **class** је у томе да **id** мора имати јединствено име за читав **HTML** документ, а име **class** атрибута може да се понавља.

За више детаља о **CSS**-у прегледајте документацију на страницама <http://www.w3schools.com/css/>.

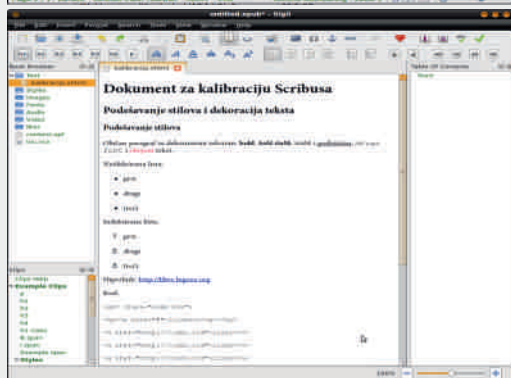
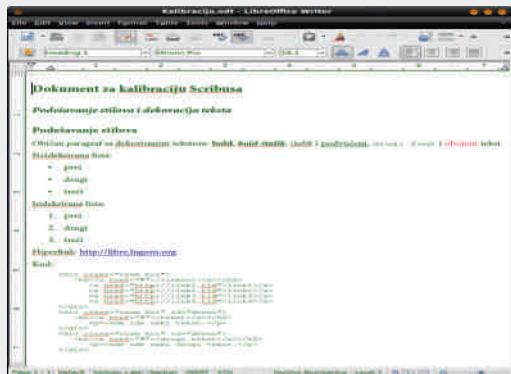
Тема **CSS**-а је преширока да би била представљена унутар ове школе. Због тога се нећемо ни трудити да улазимо у детаље, већ само објаснити:

- како да препознате **HTML** стилове,
- како да их преместите у **CSS** датотеку и
- на крају повежете са **HTML** страницом.

У неком од наредних наставака ћемо покушати мало и да унапредимо форматирање уношењем промена у **CSS** датотеку.

Стилови уграђени у **HTML** датотеку (страницу)

Искористићемо већ припремљену тест страницу *Kalibracija.odt* (припремљена за калибрацију *Scribus*-а у малој школи *Scribus*-а у броју #07) и копирати у *Sigil*. Ово је добар пример који ће илустро-



вати све несавршености које са собом носи овакав начин форматирања текста у *LibreOffice*-у и копирања истог у *Sigil*.

Декорација текста је добро унесена у *Sigil*, **<h>** тагови нису савршени, а најлошије је унет код.

Занемарићемо за сада разлике и погледаћемо документ у *Code View*:

```
<?xml version="1.0"
encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//
W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD
/xhtml11.dtd">
<html
xmlns="http://www.w3.org/1999/xht
```




```

ml">
<head>
<title></title>
<style type="text/css">
.sgc-10 {font-family: FreeMono,
monospace}
span.sgc-9 {font-size: 80%}
.sgc-8 {font-family: Minion Pro}
span.sgc-7 {color: #FF3366}
.sgc-6 {font-family: FreeMono,
monospace}
span.sgc-5 {text-decoration:
underline}
i.sgc-4 {font-weight: bold}
h3.sgc-3 {font-family: Minion
Pro}
h2.sgc-2 {font-family: Minion
Pro}
h1.sgc-1 {font-family: Minion
Pro}
</style>
</head>...

```

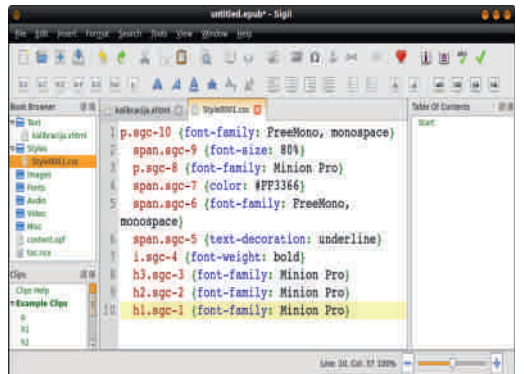
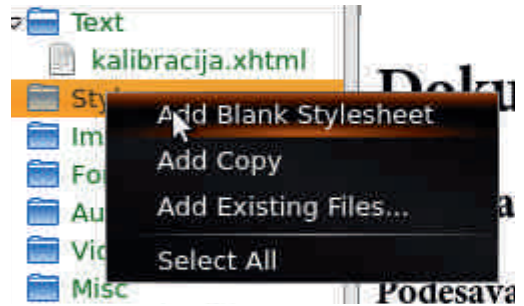
Обратите пажњу на **<head>** таг. У оквиру тог тага наћи ћете и **<style>** таг, а унутар којег се налази форматирање текста у тој *HTML* страници.

Наш циљ је да те стилове преместимо у посебну *CSS* датотеку, а да у **<head>** поставимо само линк до те датотеке.

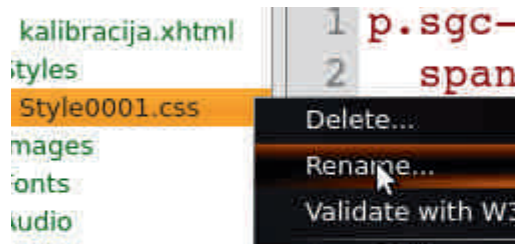
Креирање *CSS* датотеке

У *Book Browser*-у *Sigil*-а видећете да се поред фасцикле *Text* налазе и фасцикле *Styles*, *Image*, *Fonts*... Нас тренутно највише занима фасцикла *Styles*. Десни клик на ту фасциклу отвара плутајући мени из ког је потребно одабрати *Add Blank Stylesheet*. То ће отворити нови језичак у прозору едитора са називом **Style0001.css**.

Сада можемо на ту страницу да прекопирамо све из тага **<style>**, са *HTML* странице. Тако ћемо формирати нашу прву *CSS* датотеку.



У једном документу може бити више *CSS* датотека, по потреби. У већини случајева једна *CSS* датотека је довољна. Због тога можемо да преименујемо ову датотеку из **Style0001.css** у **Style.css** десним кликом на датотеку у *Book Browser*-у и одабиром *Rename...*



Повезивање *HTML* странице са *CSS* датотеком

Сада кад смо формирали *CSS* датотеку,



<style> таг у <head> је непотребан и требало би га заменити линком до CSS датотеке.

```
<link href="../../Styles/Style.css"
rel="stylesheet" type="text/css"
/>
```

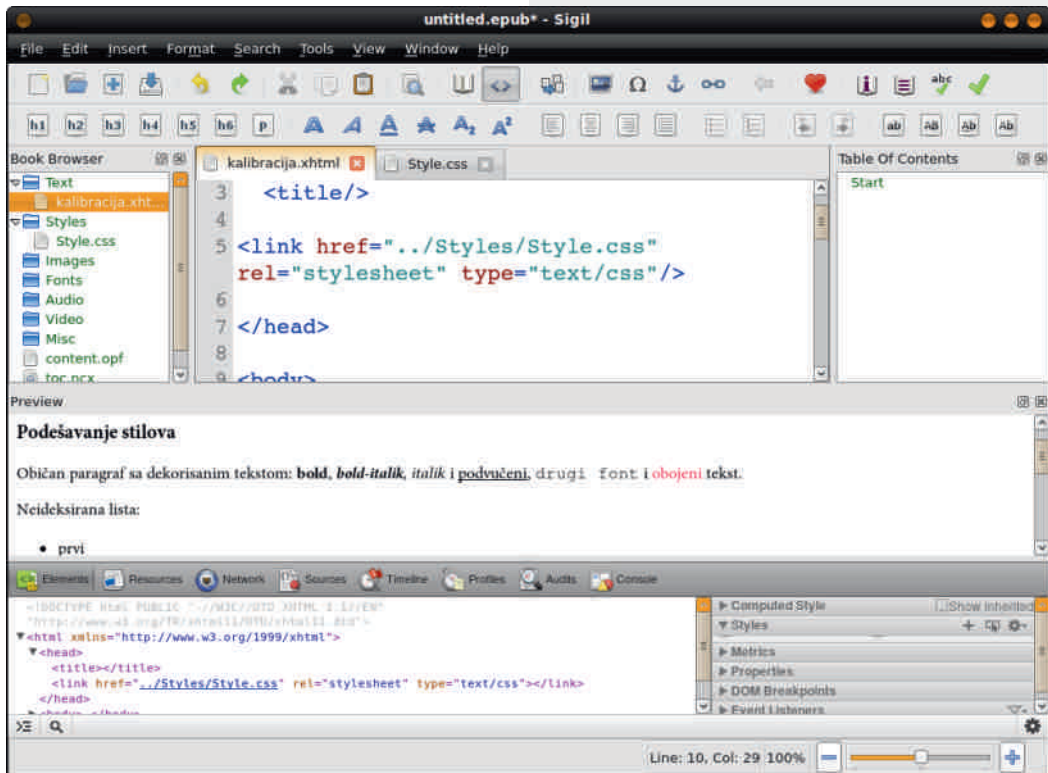
Након тога не би смело да дође до било каквих промена у изгледу странице у *Book View*-у. Можда вам сада није јасно зашто смо пребацивали стилове из *HTML* странице у *CSS* страницу кад имамо само једну *HTML* страницу. Ово добија смисао кад од једне велике *HTML* странице помоћу *Split At Cursor* направимо више мањих *HTML* страница. Свака од њих ће имати само линк ка *CSS* стилу и свака измена на

Style.css страници ће се односити на све *HTML* странице које су са њом у вези.

За крај епизоде

Ово што смо овог пута објаснили је било лакше извести него речима описати. У принципу је све веома једноставно. Сада смо тек ударили темељ за озбиљно уређивање текста у *Sigil*-у. О *HTML*-у и *CSS*-у ће бити још речи у овој малој школи.

Наставиће се...





Исправно куцање

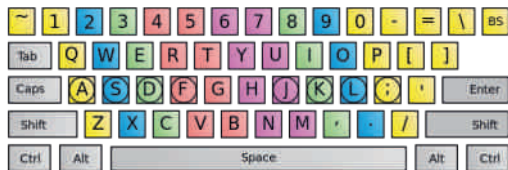
Аутор: Александар Станисављевић

Сећам се да сам као мали правио распоред тастатуре на корици свеске, како бих научио где се који тастер налази. Наставник нам је тада причао да ће доћи време када ће сви морати да знају да раде на рачунару, и био је у праву.

Имајући у виду да сам релативно касно купио рачунар, имао сам мало времена да научим брзо да пишем. Наравно, тада нисам ни размишљао о исправном куцању, већ сам куцао искључиво са два прста. Евентуално бих користио и палчеве за „space“ тастер. Постизао сам завидне резултате, међутим, знао сам да то може много боље. Онда сам видео како неки људи користе свих десет прстију док куцају, а да при томе и не гледају у тастатуру већ у монитор рачунара. Њихово куцање по тастатури ми је тада изгледало као свирање професионалног хармоникаша на хармоници. Тада сам почео да истражујем ту област и да полако вежбао исправно куцање. Међутим, све до недавно, нисам се озбиљније ухватио у коштац са овим проблемом. Имајући у виду да једном недељно ЛиБРЕ! тим има састанке на IRC-у и да често ја водим састанак, постало је веома пожељно да овладам исправним куцањем.

Напорним радом сам постигао завидне резултате и зато желим и вама да помогнем да овладате техником исправног куцања. У наставку чланка ћемо се прво упознати са основама исправног куцања, а затим и са начинима за његово учење.

Исправним куцањем се може назвати оно куцање при коме се користи свих десет прстију, и то без гледања у тастатуру приликом куцања. Једноставно, корисник тачно зна где се који карактер налази и којим се прстом одређени карактер притиска. У ту сврху је и извршена унификација, па тако свака „qwerty“ тастатура има испупчења на тастерима **F** и **J**, која служе да се руке поставе у почетни положај приликом почетка коришћења тастатуре без потребе да се гледа у тастатуру, тако што се на тастере **F** и **J** поставе кажипрст леве, односно десне руке, респективно. Исправно куцање се другачије зове и куцање додиром, јер се приликом овог куцања уопште не гледа у тастатуру већ се она само додирује прстима. Код куцања додиром је веома битно да увек враћамо прсте у почетни положај у етапама између речи или скупа речи; на тај начин смо спремни за куцање нових речи или скупа речи.



Слика 1 Зоне тастера за сваки прст појединачно

Предности оваквог начина куцања су многоструке. Док се код куцања код кога се гледа у тастатуру обично користе два прста, код куцања додиром се користи свих десет прстију. Такође, код куцања додиром нема непрестаног скретања пажње са тастатуре на монитор, и обрнуто, а очи се од монитора могу на кратко одморити и гледањем у неку слику на зиду док се куца.

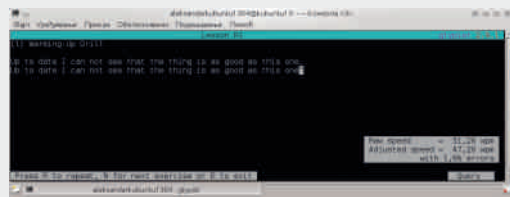
Куцање додиром је вештина којом сви могу овладати. Наравно, за овладавање било којом вештином, па и вештином куцања додиром, потребан је упоран рад. Као показатељ вашег напретка у овладавању овом вештином, најчешће се користи број речи у минути које можете да напишете. Просечна брзина писања је 40 речи у минути, док људи који се баве преваходно писањем (уношење података као део посла), могу да постигну брзину и од 100 речи у минути. Но, подаци о просечној брзини писања не треба да вас деморалишу, пошто ћете у почетку вероватно спорије писати него што то уобичајено радите користећи се старом методом (куцање са два прста); напротив, они треба да представљају циљ до којег требате доћи упорним радом.

За овладавање вештином куцања

додиром, између осталог, постоји много програма уз помоћ којих се, корак по корак, ова вештина учи. Сада ћемо се упознати са неким од њих.

GNU Typist

GNU Typist (такође познат и као *gtypist*) је конзолни програм за учење куцања додиром који је објављен под *GPLv3* лиценцом.

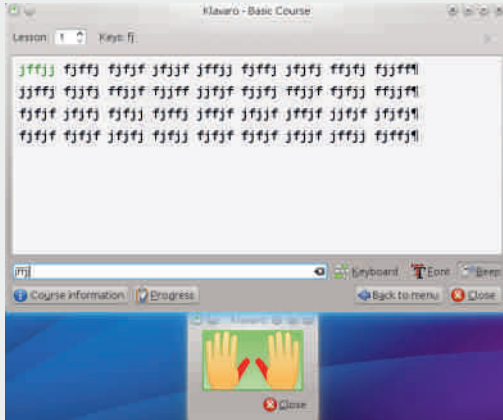


Слика 2 *GNU Typist*

Након сваке завршене лекције, у углу прозора ће бити приказана брзина коју сте остварили као и проценат грешака које сте направили. Имајте на уму да не можете да пређете на наредну лекцију док у текућој не остварите мање од 3 % грешака.

Klavaro

Klavaro је, за разлику од *GNU Typist*-а, програм са графичким корисничким интерфејсом. Поседује неке могућности које почетницима у великој мери могу да олакшају учење куцања додиром. На пример, у основном курсу је могуће да укључите помоћника који ће вам за сваки карактер сликовито показати који прст да притиснете.



Слика 3 Klavaro

Такође, док сте у основном курсу, у сваком тренутку можете отворити цео распоред тастатуре и кликом на одређено слово видите којем прсту то слово одговара.



Слика 4 Интерактивни приказ зона тастера за сваки прст појединачно

Ktouch

Ktouch поседује занимљив графички интерфејс у којем су показатељи вашег напретка у учењу куцања додиром приказани у облику аналогног брзи-

нометра. За сваку лекцију постоји предвиђено време за које је потребно завршити је, брзина писања изражена у броју карактера по минути, као и прецизност куцања изражена у процентима.



Слика 5 Ktouch

Имајући у виду да претходно описани програми дају показатељ квалитета вашег куцања у облику броја речи у минути, а *Ktouch* у облику броја карактера у минути, немојте да вас завања велика вредност овог показатеља *Ktouch*-а.

TuxTyping

Уколико сте дете (или се бар тако осећате), куцање додиром можете да вежбате и играјући се. То вам омогућава програм *TuxTyping* из пројекта *Tux4Kids*. Наравно, доступне су и класичне лекције, али вам ипак препоручујемо оне мало забавније.



Слика 6 *TuxTyping*

Без обзира који програм одаберете за учење куцања додиром, веома је важно да будете истрајни у вежбању. Након одређеног времена и сами ћете приметити позитивне резултате. Такође, веома је битно да се трудите да увек исправно куцате, а не само док радите у програму у којем учите

исправно да куцате. Замислите само како би изгледала музика хармоникаша који се служи само са два прста; да ли би стигао даље од „На крај села жута кућа“. Зато, прионите већ данас на посао.

Корисни линкови:

- [1] http://en.wikipedia.org/wiki/Touch_typing
- [2] <http://www.gnu.org/software/gtypist/>
- [3] <http://klavaro.sourceforge.net/en/>
- [4] <http://ktouch.sourceforge.net/>
- [5] <http://tux4kids.alioth.debian.org/tuxtype/>

21. септембар (субота), од 12 до 16 часова, Дом културе Ловћенац

УЛАЗ СЛОБОДАН



45°40'53"N
19°41'29"E

Ловћенац
2013.



Дан слободе
софтвера

ДАН СЛОБОДЕ СОФТВЕРА У ЛОВЋЕНЦУ

За сваког посетиоца поклон CD са Linux оперативним системом

Како дати допринос програмима отвореног кода иако нисте програмер?

Аутор: Сандрина Димитријевић

Пројекти отвореног кода, који су потпуно бесплатни и којима свако може да допринесе, играју значајну улогу у развоју информационих технологија. Они подстичу слободан приступ информацијама, транспарентност и отвореност, умањују неједнакости између група и појединаца и онемогућавају монопол великих компанија. Многи програмери посвећују део свог слободног времена овим пројектима, зато што подржавају вредности заједнице, али и зато што је то одличан начин да се стекне искуство и научи нешто ново. Међутим, није потребно бити програмер како бисте допринели свом омиљеном пројекту; постоји још много начина како то можете да урадите.



Данас ћемо навести неке од њих.

Донације у новцу

Премда је цео рад на пројектима отвореног кода на волонтерској бази, они ипак имају трошкове. То су трошкови сервера, домена, хостинга и још много тога. Стога, ако сте у могућности, подржите свој омиљени пројекат новчаном донацијом, ма колико она била мала. Особама које одржавају пројекат ће несумњиво бити од користи. Тако ћете помоћи да цео пројекат буде још бољи и да се константно одржава. Већина пројеката има секцију за прикупљање донација на свом званичном сајту.

Писање упутстава

Поред писања и ревизије кода, програмери немају много времена за писање упутстава и документације. Ту на сцену ступају верни корисници, који умногоме могу да помогну популаризацији одређеног пројекта, ако напишу корисничко упутство и поделе га са широком публиком. Упутства можете писати на званичном сајту пројекта, као део званичне документације, али и на корисничким форумима, блоговима и осталим местима на интернету. На тај начин софтвер постаје приступачнији и разумљивији ширим масама.



Помоћ осталим корисницима

За разлику од комерцијалних решења, програми отвореног кода, немају званичну подршку. Корисници често могу да буду збуњени одређеним подешавањима, па не знају како да нешто ураде. У таквим ситуацијама они углавном постављају питања на званичним форумима. У случају да сте добро овладели одређеним програмом, скокните с времена на време до тих форума, и покушајте да саветима помогнете другим корисницима како да реше проблем.

Превођење

Главни интерфејс скоро свих програма доступних на интернету, као и помоћна документација, су углавном на енглеском језику, а ређе на матерњем језику програмера. То може отежати коришћење тог програма на нашем говорном подручју, јер не разумеју сви енглески. Зато је превођење одличан начин да се одређени програм отвореног кода приближи људима са нашег говорног подручја. Велики број пројеката има отворен позив за превођење интерфејса или помоћне документације, али не морате увек да тражите позив. Углавном је довољно да се сами јавите програмерима и понудите своју помоћ; свако ће је радо прихватити.

Сам процес превођења није уопште сложен и на захтева напредно техничко знање; углавном добијете мали текстуални документ са главним насловима, порукама и фразама, које треба да преведете на свој језик.

Приликом превођења водите рачуна о духу језика на који преводите, и избегавајте накарadne изразе. На пример, за српски језик од велике помоћи вам може бити компјутерски речник Микро-књиге

<http://www.mikroknjiga.rs/pub/rmk>.

Помоћ око графике и дизајна

Помоћ дизајнера је често неопходна код пројеката отвореног кода, било да се ради о дизајну логоа или делова корисничког интерфејса. Изглед корисничког интерфејса је нешто што корисници прво примете, па их то може одвратити од коришћења. Ако сте вични дизајну, понудите се да помогнете свом омиљеном пројекту – то вам неће одузети много времена.

Остало

Постоје још многи начини на које можете помоћи омиљеним програмима, све у зависности од тога којим вештинама владате. На пример, можете поставити видео упутство на *YouTube*, помоћи у проналажењу буба у софтверу, пружити консултације у области промоције и маркетинга и још много тога. Не заборавите да, на тај начин, промовишете вредности слободе и једнаког приступа информацијама које, у данашње време монопола и растуће неједнакости међу земљама и појединцима, све више добијају на значају.



Донацијама можемо да направимо разлику

Аутор: Александар Тодоровић

Ми смо као људска раса до сада направили велике кораке – у погрешном смеру. Да, направили смо рачунаре, интернет, отишли на мјесец, излечили понеку болест... Имамо ми тих неколико добрих ствари које смо направили и урадили. А са друге стране? Нуклеарне, хидрогене и друге врсте бомби, тероризам; сваки педаљ земље је под нечијим власништвом, као и сваки комад природе, ропство (и оно старомодно и ово ново у 21. веку када једноставно целе дане радиш за нечији други профит и да би преживео), дечија порнографија и тако даље. Присвојили смо себи једноставно читав свет. Иако живимо са огромним бројем других животињских и биљних врста, нема тог дела земље који није у власништву или неке особе, или неке корпорације, или неке државе. Чак смо и Антарктик успјели подијелити међу собом онолико колико га можемо искористити – за остатак наравно никога није брига (свеједно ће се испитати у скоројој будућности).

Морамо признати да је ово фасцинирајуће. Добили смо читаву једну планету (заправо украли од свих других биљних и животињских врста, али нећемо у детаље) да се множимо, градим,

рушимо, искоришћавамо, а једина ствар која је у 2013. још увијек слободна јесте нешто што смо сами направили – интернет. Зашто је тако? Па одговор је доста једноставан, зато што не припада никоме и зато што је одавно прешао ту границу када може да буде под нечијом контролом тако лако. Свако има право да ради шта хоће и кад хоће на њему, нико га не контролише, нико кориснику ништа не брани да посећује, нико му неће покуцати на врата због неколико кликова... добро, то је била ситуација од пре неколико година, и занимљиво је колико се тога промјенило у задњих пет до десет година.

У каквом стању је интернет данас?

Па добро, још увек је он на неки начин слободно мјесто, али је на добром путу да и он постане нечије власништво. Имамо на једној страни људе који су обични корисници и који раде сасвим обичне радње за данашње време на њему: причају са пријатељима, „скидају“ музику, гледају филмове, читају вести и то је мање више то. Далеко од тога да не постоје људи који раде још много више од овога на интернету, али су такви људи ипак мањина. Са друге стране стоје Владе (пре свега Влада



САД-а, али не заостаје за њом ни Влада Велике Британије, Немачке и Холандије), Холивуд, Силиконска долина и свако други који очекује да ће остварити профит од нечега што је ништа друго него правилно поређан скуп нула и јединица. Таква рачуница једноставно није могућа без потпуне контроле интернета. Нуле и јединице се лако копирају у том истом редоследу и имамо копију тог истог производа који можемо проследити даље, нешто што се не може са, рецимо храном, пићем, хардвером... Дакле, очекивати да се заради нешто продавањем истог збира нула и јединица различитим корисницима који сами могу са тим збиром да раде какве год радње пожелеле (укључујући и копирање), јесте сасвим нереално и једноставно немогуће остварити. Зато смо дошли у време када свакодневно имамо *DMCA* тужбе за ауторска права – на *YouTube*-у нестају милиони видеа сваког дана због тих тужби, *Google* и *Microsoft* уклањају милионе резултата претраге сваког дана, а неке од тих цензура су „грешком“ захватиле бесплатне програме и документарце који представљају конкуренцију комерцијалним производима [2, 3].

Решење?

Уместо да форсирамо овакве начине пословања, потребно је примењивати много једноставније решење за наплаћивање: донације. Са једне стране бисмо тада имали корпорације које би запошљавале програмере да направе што бољи производ да би корисници баш њега користили, а тај производ им се мора доста свидети да би му

донирали нешто (дакле мора бити исправан, без пуно буба (енгл. *bug*), брз, поуздан, графички леп и једноставан за коришћење). Са друге стране бисмо имали обичне кориснике који би могли да приступају свему што пожелеле, а то не сматрамо погубним за произвођаче. Још нисмо упознали особу која не би погледала неки филм у кину уместо на лаптопу. Још нисмо упознали особу која би се бунила да посети концерт аутора за кога је чула преко интернета па јој се свидео. Још нисмо упознали ни особу која не би платила одређену количину новца аутору да добије чврсту копију онога што јој произвођач нуди (као на пример програм, филм или музику на *CD*-у, *DVD*-у или *USB*-у). Дакле, имамо алтернативан начин зараде и у примени је већ у многобројним мањим фирмама широм света.

Шта бисмо све добили тиме?

Као прво, добили бисмо квалитетније производе, квалитетније произвођаче и задовољније кориснике. Дobili бисмо нову креативну радну снагу, јер би сигурно постојао неко од нас ко би покушао да буде део у снимању неког филма, или би покушао да сними песму када би једноставно могао да добије програм преко којег би могао то да уради, а да не мора да плати за такав програм неколико стотина или чак и неколико хиљада долара. Научна достигнућа би била много већа када би сви научници могли да имају све информације које им требају на располагању двадесет четири сата у дану свакога дана. Вероватно не би напредовала



искључиво једна фирма или искључиво једна држава – читава људска заједница би напредовала, овај пут у правом смеру.

Шта бисмо изгубили?

У почетку би вероватно био мањи профит када би се више фирми окренуло овом начину зараде, међутим, ако би тај производ успео да привуче више људи и био квалитетан, не видимо разлог зашто се за тај производ не би донирала велика количина новца. Имамо две могуће будућности: у једној се сав софтвер и мултимедија плаћа, а у другој је све бесплатно. На нама је да одлучимо која ће од тих будућности да се оствари. У почетку би такве фирме вероватно изгубиле део свог профита, али би у мало даљој будућности могле да профитирају вишеструко. Мислимо да се за крај ипак морамо запитати и шта је постао новац у данашње вријеме? Ништа друго него још један збир правилно посланих нула и јединица...

Шта могу ја као појединац да урадим?

Овај корак јесте врло једноставан, али смо у таквом времену да би тај један исправан корак данас донео много лепше сутра када је технологија у питању – донирајте. Много људи једноставно није свесно да за скоро сваки програм постоји нека бесплатна алтернатива. Можда није боља или бржа или графички привлачнија од других комерцијалних производа, али не смете да заборавите да су ти производи лошији само зато што њихови аутори нису у

довољној мери подржани од стране обичних корисника. Дакле, уместо да платите 150 долара за неки комерцијални програм који ће вам служити за одређени посао, донирајте 100 долара неком програму са отвореним кодом који има сличне могућности, па чак и ако је за степеницу нижи од тог комерцијалног по квалитету, изгледу и функционалности. На тај начин ћете добити програм са којим ћете се морати вероватно мало потрудити и направити три клика уместо два, али даћете своју подршку неким добрим програмерима који су се потрудили да вам пруже ту технологију бесплатно, и још ће вам остати 50 долара више у џепу. Када би добар део нас тако поступао редовно, имали бисмо невиђене користи, и као појединци, и као заједница, а комерцијални програми постепено би могли да постану прошлост. Музика би се све више давала бесплатно, а можда бисмо током наших живота доживели и да можемо да гледамо неке филмове без страха да нам полиција због тога покуца на врата (што је сасвим могуће да се деси за, рецимо пет година). Дакле, све што је потребно је направити један корак у супротном смеру од онога на који смо навикли.

Корисни линкови:

- [1] <https://flatrr.com/>
- [2] <https://torrentfreak.com/hollywood-studios-take-down-pirate-bay-documentary-130519>
- [3] <https://torrentfreak.com/microsoft-sensors-openoffice-download-links-130814/>



Направите ваш виртуелан свет



Аутор: Дејан Чугаљ

Трилогија филма „Matrix“, холивудска представа како би могао да изгледа виртуелан свет, представља изванредну причу препуну „скоро“ немогућих детаља, али у експанзији развоја *IT* технологија некако се приближавамо холивудским сновима. Виртуелан свет, појам који је ту, „лебди“ некако око нас, спомиње се скоро у свакој причи која је повезана са *IT* технологијама. Виртуелизација, виртуелне машине, уопштено виртуелан, све су чешћи појмови које можемо чути у данашњем жаргону *IT* речника. У неком контексту виртуелног окружења су сви, почев од *hosting* компанија, преко иоле озбиљнијих фирми, па све до обичних корисника. Обични корисници можда нису ни свесни да користе неки вид виртуелизације, али самом провером *mail*-а на „Gmail-у“, они управо то чине. Када чујемо „облак“, све изгледа и чини се некако апстрактно (нестварно), па из контекста „облак“, издвајамо пројекат отвореног кода написан у *Java* програмском језику под називом *Open Wonderland*.

Визија пројекта *Open Wonderland* је да пружи окружење у коме је могућа интеракција самих учесника, а да при томе то укључује и тродимензионалан свет (3D) у

којем би свако од њих имао своју замишљену личност (*avatar*-а). Интеракција може да буде у било коју сврху, преко пословних састанака па све до едукативних часова, соба за ћаскање итд.

Организације које користе *Open Wonderland* би требало да буду у могућности да представе виртуелно окружење као бољи вид комуникације између пословних партнера, студената, пријатеља. Индивидуално коришћење оваквог скупа алата виртуелизације би, у сваком случају, дружења представило много пријатнијим и омогућило да се сваки од учесника, у неку руку, представи боље и креативније него што се то тренутно ради у свакодневници.

Типови комуникације који су доступни у *Open Wonderland*-у су: аудио-видео комуникација, дељење свих могућих *desktop* апликација, постављање разних билборда, буквално прављење ваших делова света онако како вама то одговара.

Један од најбитнијих делова овог пројекта је проширивост сваког дела истог. Програмери могу да проширују пројекат и да га прилагођавају својим потребама пишући модуле и компоненте које се касније „увозе“ у сам виртуелан свет. Такође је битно овде напоменути да су

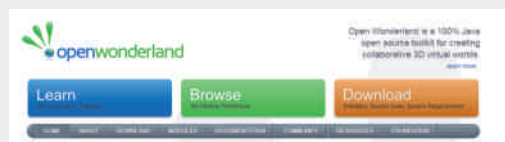


сви делови пројекта отвореног кода, од графике па све до и једне линије кода који је написан у пројекту. *Open Wonderland* је под *GNU General Public License v2.0 + Classpath Exception* лиценцом, док све детаље лиценце можете видети на адреси

<http://openwonderland.org/documentation/lfaq#Licensing>.

Наравно, све ово није намењено само програмерима, чак и обични корисници уз минимално знање могу да покрену сервер као подлогу на којој ће се сва ова прича и одвијати, а у наредним корацима ћемо и показати како све то и да спроведемо у дело. Такође, овде би било пожељно споменути да је ово клијент-сервер апликација, што значи да ћемо на једној машини (коју год ви одаберете, може да буде и ваш сопствени кућни рачунар), покренути сервер на којем ће се налазити виртуелан свет који сте ви сами креирали. Клијентски део целе структуре се покреће тако што се приступи локалној *web* адреси коју ћете добити на крају самог покретања сервера виртуелног света, а реализује се преко *Java applet*-а који се извршава у *web* прегледачу.

Први корак који је потребно да се уради је посета сајта *Open Wonderland*-а, на адреси <http://openwonderland.org/>.



Након тога је потребно преузети *JAR* датотеку која у себи поседује све

неопходне компоненте за покретање сопственог виртуелног света у секцији „Преузимања (*Download*)”, на адреси <http://openwonderland.org/download/binary>.



Потребно је преузети „*Wonderland.jar*” (величина 353 [MB]) датотеку.

Нако што се преузме горенаведена датотека, потребно је споменути који су минимални услови потребни за покретање сервера.

- Подржани оперативни системи:
Windows XP/Vista/7, GNU/Linux (Debian/Ubuntu), OpenSolaris, Mac OS X.
- *Sun Java SE 6 JDK*, верзија 1.6.0, *update 14* или каснији.
- Потребно је имати оригиналан *driver* графичке картице (прим. аут.).

Након преузимања датотеке сачувајте је у неку вама лако доступну фасциклу, и отворите терминал или командни прозор, у зависности од оперативног система, а затим идите до фасцикле у којој је сачувана *JAR* датотека.

Сервер на *GNU/Linux* или *OpenSolaris* системима се покреће командом:

```
java -jar Wonderland.jar
```



на *Mac OS X*-у:

```
java -server -jar Wonderland.jar
```

на *Windows* платформи:

```
(32 bit)
"\Program
Files\Java\jdk1.6.0_16\bin\java"
-jar Wonderland.jar

(64 bit)
"\Program Files
(x86)\Java\jdk1.6.0_16\bin\java"
-jar Wonderland.jar
```

Након покретања сервера потребно је мало времена да се све постави на своје место, па то изискује мало стрпљења; такође видећете много порука исписаних у терминалу, између осталих и упозоравајућих (*warnings*), мада то не би требало да вас брине. Обично је за извршење целог процеса потребно два или три минута.

Након, надамо се успешног покретања, на крају ће вам се исписати порука:

```
-----
Wonderland web server started
successfully.
Log files are in
/Users/username/.wonderland-
server/0.5/log
Web server running on
http://myserver:8080/
-----
```

што значи да је сервер са виртуелним светом успешно покренут, и да се истом може приступити на *web* адреси (у овом случају) `http://myserver:8080/`.

Ако ово све некако не успева, предлог који имамо је да се ипак покреће сервер са још пар додатних параметара, овако:

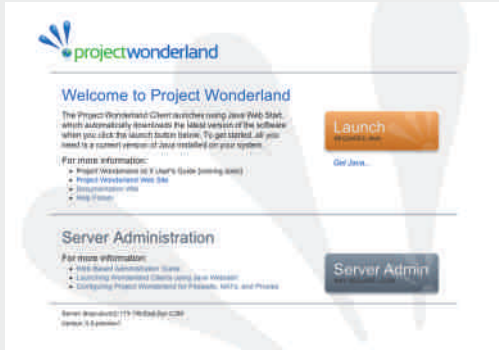
```
java
-Dwonderland.webserver.host=localhost
host -jar Wonderland.jar
```

где је параметар „-Dwonderland.webserver.host=localhost” назив локалне *IP* адресе, где да се покрене сервер. Знамо да је „localhost” предефинисана адреса за *IP* адресу 127.0.0.1, па би крајња, успешна исписана порука изгледала:

```
-----
Wonderland web server started
successfully.
Log files are in
/Users/username/.wonderland-
server/0.5/log
Web server running on
http://localhost:8080/
-----
```

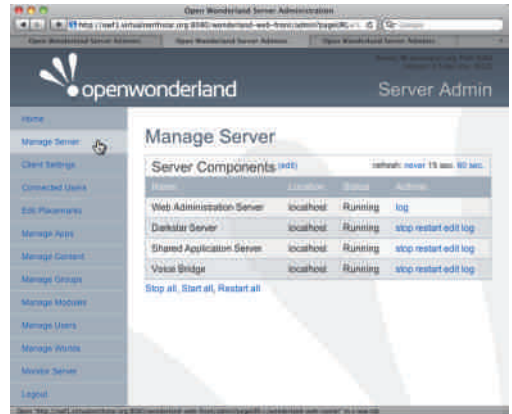
Напоменимо то да не мора да буде локална *IP* адреса, већ може бити било која и серверу ће се моћи приступити на тој *IP* адреси коју ви наведете.

У овом тренутку је довољно да отворите ваш *web* прегледач и да га усмерите на адресу коју сте проследили као параметар; ми приступамо са `http://localhost:8080/`; требало би да видите садржај *web* странице као на слици.

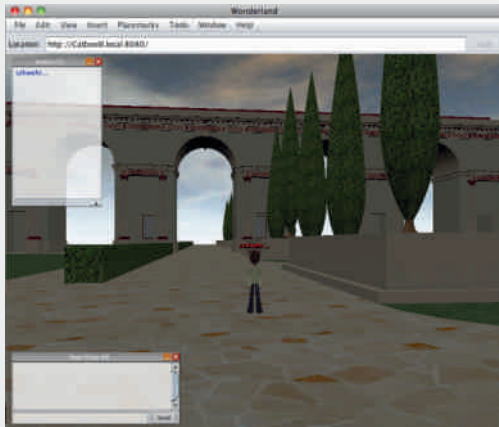


Да би корисник приступио виртуелном свету довољно је да кликне на дугме *Launch* и покренуће му се *Java applet* који ће га за пар тренутака одвести у виртуелан свет.

web прегледачу, и требало би да видите *web* страницу као на слици.



Ваш виртуелан свет је спреман.



Наравно да је ово чланак који **укратко** представља виртуелни свет који ви сами можете да имплементирате, па рекло би се веома лако. Ако сте се заинтересовали и чини вам се да бисте могли да имплементирате неку од својих идеја, дајемо вам корисне линкове одакле можете да почнете. Надамо се да ће вам бити занимљиво.

Корисни линкови:

- [1] <http://openwonderland.org/>
- [2] <http://wiki.openwonderland.org/Wiki.jsp?page=Main>
- [3] <https://groups.google.com/forum/#!/forum/openwonderland>
- [4] <https://sites.google.com/site/openwonderland/tutorials/learning-the-basics-tutorial>



Приступ администраторском делу сервера вашег виртуелног света се обавља кликом на дугме *Server Admin* у вашем

oVirt

Web апликација за управљање платформом за виртуелизацију

Аутор: Иван Булатовић

IТ администраторима је чињеница да никада на располагању немају довољно процесорског времена, RAM меморије и простора на диску, веома позната. Иако су цене сервера и компоненти сада разумније него што је то било деценију раније, економска криза и смањење буџета за ИТ одељења нас форсирају да расположиве ресурсе максимално искористимо. Пођимо од тога да нам је неопходан високо доступан сервис, без SPOF-а (енгл. *Single Point Of Failure*) са комплетно редувантним системом. На листи ће се наћи два сервера са редувантним напајањима, уређаји за масовно складиштење података, уређај за резервно напајање електричном енергијом, редувантна мрежна опрема и софтверско решење за високу доступност сервиса који се инсталира. Почетна улагања су огромна и прогресивно се смањују са бројем сервиса који су неопходни, али је и даље неопходно улагати у нове сервере чији капацитети врло вероватно неће бити потпуно искоришћени. Трошкови због увећане потрошње електричне енергије, одговарајућег хлађења просторије, одржавања самих система, њихове замене у неком

наредном временском периоду и компликованог управљања, подразумевају се током експлоатације.

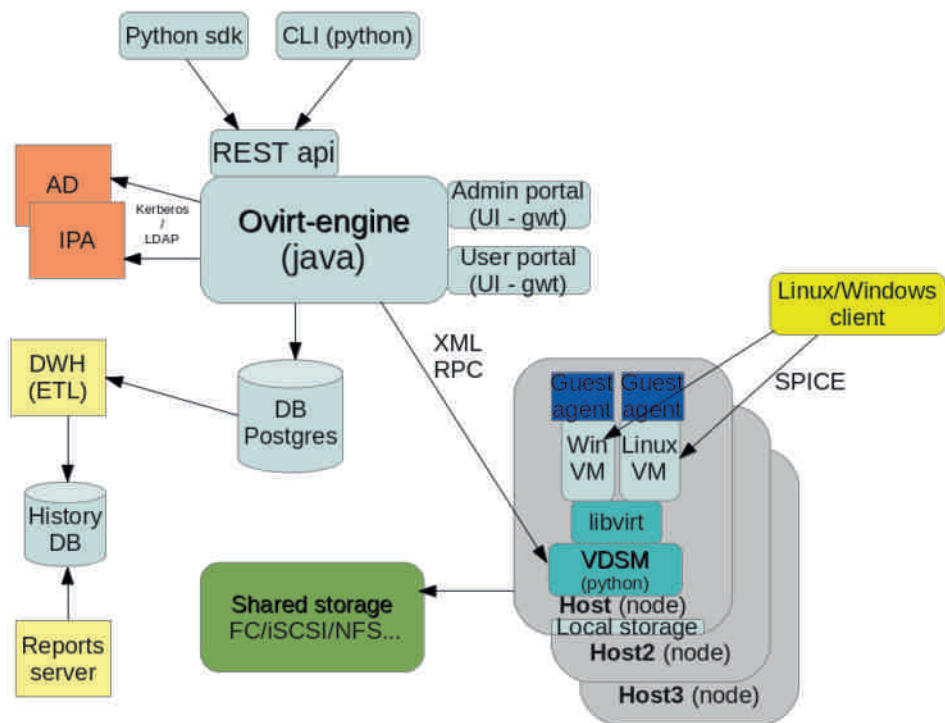
Са овим проблемима се ИТ индустрија ефективно изборила понудом и имплементацијом решења за виртуелизацију хардвера. Уместо додавања нових сервера сваки пут када је неопходно инсталирати неки од софтверских сервиса, могуће је постојеће хардверске капацитете спојити и такав удружени ресурс поделити између сервиса, тако да се он максимално искористи. Најпознатија решења за виртуелизацију нам долазе од компанија *VMWare*, *Citrix*, *Microsoft* и *Red Hat*. Иако *VMWare* држи лидерску позицију, све више компанија жели да се окрене решењима која су базирана на отвореном коду, за шта је чист пример *OpenStack*, иза кога се налазе компаније као што су *Red Hat*, *HP*, *IBM*, *Cisco*, *AT&T* и многе друге.

Овом приликом представљамо вам **oVirt**, апликацију отвореног кода за управљање виртуелизацијом. Као и код већине других апликација сличне намене, неопходан је један сервер за управљање једним или више нодова који сервисирају виртуелне машине. *oVirt* захтева двојезгарни сервер са минимално 4 [GB] RAM-а и 25 [GB]



простора на диску. На слици је приказана архитектура на којој се заснива oVirt.

у *Postgres* бази података, чиме је олакшан приступ и креирање резервних копија. Инсталација је максимално поједностављена и припрем-



Централна компонента oVirt сервера је његов покретач тј. „ovirt-engine“ који је базиран на *Java* програмском језику и који преко *REST* апликативног програмског интерфејса пружа могућност додавања алата за управљање виртуелним машинама. Сервис је мрежног карактера, што се постиже помоћу *Jboss* апликативног сервера, и може му се приступити преко било ког *web* прегледача. Подаци се складиште

љени пакети постоје за *Fedora* и *CentOS Linux* дистрибуције. Довољно је додати одговарајуће складиште програмских пакета, ажурирати систем, инсталирати *ovirt-engine* пакет и покренути инсталацију командом *engine-setup*, при чему је неопходно унети неколико података, попут подразумеваног сервиса за складиштење података, затим лозинку и порт за приступ апликацији и бази података итд.



Следећи корак је припрема нодова који ће сервисирати виртуелне машине. Нодови могу такође бити базирани на *Fedora* или *CentOS* дистрибуцијама, а потребно је само инсталирати **vds** пакет, помоћу којег се обавља *XML-RPC* комуникација на релацији нод – *oVirt* сервер. *VDSM* је *libvirt* оријентисан сервис који за хипервизор користи **KVM** (*Kernel-based Virtual Machine*). Сва даља подешавања обављате помоћу *oVirt* апликативног интерфејса путем *web* прегледача. Прво морате креирати свој *Datacenter*. *Datacenter* се састоји од три компоненте: складишта података, мреже и кластера.

Складиште података се састоји од само једног складишта за *ISO* слике, помоћу којих можете инсталирати неки од подржаних оперативних система, једног складишта за експорт виртуелних машина и једног или више складишта који ће садржати саме слике виртуелних машина. Овим складиштима можете приступити преко оптичке *FC* везе, *iSCSI* или *NFS* протокола, *GlusterFS* домена или локалног складишта (на тврдом диску самог нода).

Мрежу можете подесити из *oVirt* апликације или на самом *oVirt* серверу, при чему ће покретач сам препознати мрежна подешавања. Мрежа може бити премошћена на већ постојећу физичку мрежу, или можете креирати сопствени *VLAN* и подесити *oVirt* тако да обавља функцију *DHCP* сервера и аутоматски додељује *IP* адресе виртуелним машинама у *VLAN*-у. П подразумевани назив мреже је *ovirtmgmt*.

Кластери се могу састојати од једног

или више сервера који имају улогу нодова. Истородне сервере је препоручљиво имплементирати у један кластер, и уколико правилно подесите управљање напајањем сервера помоћу „fence” агената, могуће је унутар истог кластера инсталирати високо доступне виртуелне машине са аутоматским „failover” механизмом. Уколико машина није високо доступна, увек вам преостаје функција живе миграције између нодова унутар истог кластера. Не постоји ограничење по броју кластера унутар једног *Data* центра.

Приликом креирања виртуелне машине на неким од нодова, дефинишете назив виртуелне машине, број процесорских језгара/подножја који су јој доступни, количину *RAM* меморије и колико од тог износа јој се физички гарантује, који тип оперативног система се инсталира (*Windows*, *RedHat* или *Other Linux*), приоритет уређаја приликом *bootstrap* процеса као и инсталациону *ISO* слику. Можете подесити и тип кеширања диска, да ли ће виртуелна машина бити високо доступна или желите да забраните било какав вид миграције те машине (*pinning*). Затим додајете виртуелне мрежне интерфејсе и дефинишете један или више виртуелних дискова. Могуће је додати већ припремљени *LUN* преко *iSCSI* или *FC* протокола, или га креирати на одабраном складишту података. Простор резервисан за дискове може бити преалоциран или танко алоциран (*Thin Provisioned*). Када је простор преалоциран, он је унапред припремљен и заузима онолико простора на складишту колико му је

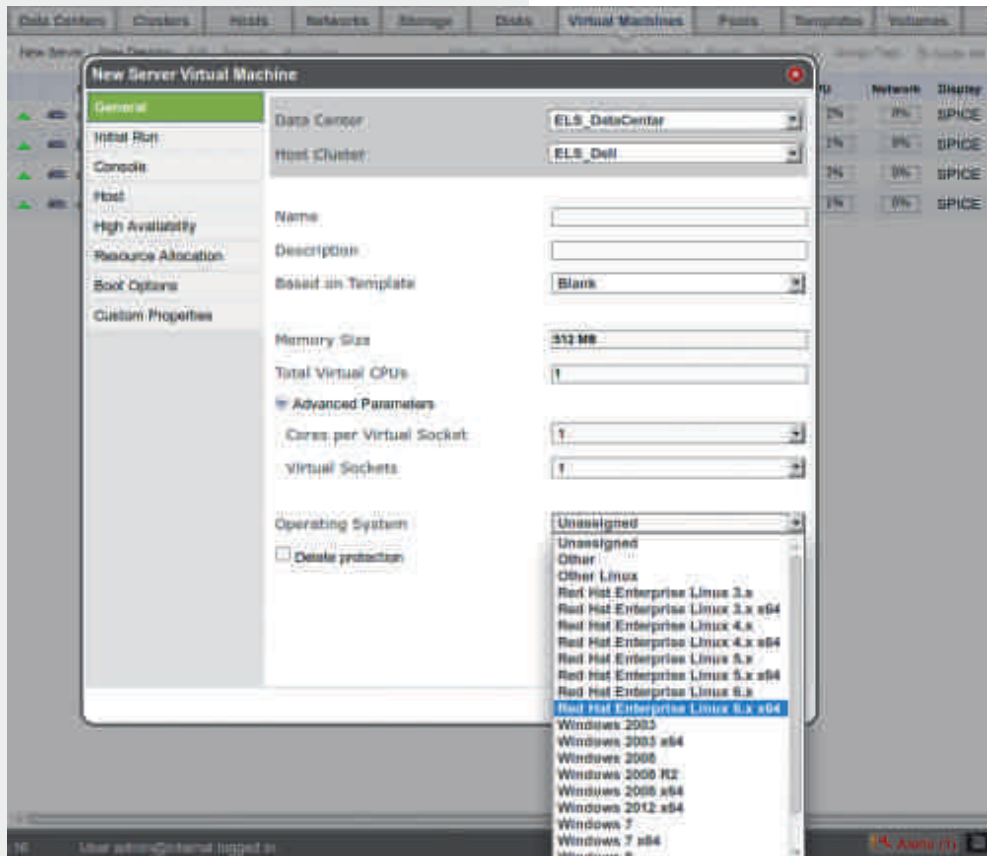


додељено, док код танке алокације простор се динамички алоцира. У првом случају предност је брзина рада, док је у другом могуће на складишту са 300 [GB] простора додати десет дискова од 100 [GB], а касније проширити складиште када се за то стекну услови.

Могућности *oVirt* апликације се не завршавају овде. Могуће је креирати живе снимке виртуелних машина (*Live Snapshots*) које вам омогућују да направите *backup* виртуелне машине без прекида у раду сервиса или искључивања машине. *RAM* меморију је могуће танко алоцирати помоћу **KSM** (*Kernel*

SamePage Merging), чиме се предности оваквог алгоритма у случају дискова преносе и на радну меморију, али захтева доста процесорског времена, што може представљати проблем. За миграцију физичких и виртуелних машина које су креиране помоћу других хипервизора, можете користити *virt-v2v* који је део *libguestfs* пројекта и који подржава *oVirt* формат. Из *web* прегледача можете директно покренути и приступити виртуелној машини путем **SPICE** или **VNC** протокола.

Шлаг на тарту представља чињеница да је *Red Hat Enterprise Virtualization*





(*RHEV*), комерцијално подржана апликација за управљање виртуелним машинама, базирана на *oVirt* апликацији; изгледа и функционише потпуно исто. Ипак, у „enterprise” окружењима, треба се одлучити за *RHEV* због подршке и неколико „нагазних мина” које ћемо споменути у наставку чланка.

Оно што ће вас одушевити код *oVirt*-а, уколико се одлучите да га тестирате, невероватно је једноставан и интуитиван кориснички интерфејс. Будући да *oVirt* архитектура верно пресликава реално/виртуелно окружење, то у многоме помаже и *IT* администраторима приликом управљања истог. Једном када се инсталира и подеси да функционише, ради беспрекорно стабилно на *enterprise* нивоу.

Наравно, ту је и друга страна новчића. Инсталација уме да буде проблематична. Управљање грешкама је катастрофално и захтева од вас много утрошеног времена и копања по *log* датотекама које су све подразумевано на *DEBUG* нивоу. Чак и да одлучите да ниво пребаците на *INFO*, гомила излаза на *INFO* нивоу је загарантована једном поруком која се понавља на сваких неколико минута, а коју можете искључити, само ако имате налог на *Red Hat* корисничком порталу, тј. претплаћени сте на *RHEV* производ па можете приступити бази знања. Не постоји „garbage collection”, тј. свака грешка приликом уклањања виртуелних машина или спајања живих снимака система узрокује да објекат остане приказан у апликацији и може се уклонити само ручно, приступом *postgresql* бази, што никако није

препоручљиво.

Ништа у току администрације *oVirt* апликације (не самих виртуелних машина) вам неће олакшати посао. Документација као да не постоји, више ћете научити са *oVirt* дописних листа и *Red Hat bugzilla* портала, па чак и *YouTube* је од веће помоћи него званична документација. Топло препоручујемо инсталацију *oVirt*-а под виртуелном машином (нпр. помоћу *virt-manager* апликације) и да пре сваког ажурирања урадите клонирање виртуелне машине, злу не требало. Желите да урадите конверзију *Windows* физичких машина? То нећете моћи без *virtio-win* пакета који не постоји за *CentOS* дистрибуцију, а ако покушате да искористите ону из *Fedora* складишта програмских пакета, разочараћете се другачијим путањама до управљачких програма, него што је то случај код *RHEV* решења. Једноставно, *Red Hat* је јасно ставио до знања свима нама да је *oVirt* тврд колачић за употребу у *enterprise* окружењу и да може да послужи искључиво као развојно окружење за *RHEV* или упознавање са *RHEV* апликацијом пре куповине исте, мада је могуће доћи до пробне верзије регистрацијом на *Red Hat* порталу. Ипак, не би било праведно изоставити да сте раније могли тестирати *oVirt* само на *Fedora* дистрибуцији, а да су сада доступни пакети и за *CentOS* који је много стабилнији и који се може узети у обзир за продукциони рад. Да ли је ово везано за померање фокуса на *OpenStack*, нико не може са сигурношћу да потврди, али у сваком случају овакав корак треба поздравити.



Виртуелизација која се базира на отвореном коду има много предности. *oVirt* је једна од многобројних апликација која се ослања на *libvirt* и *KVM*, па је миграција олакшана на неко друго решење, у случају да сте принуђени на такав корак. Већина њих је бесплатна и без икаквих функционалних ограничења, осим у случају да вам је неопходна подршка, па новац можете уложити у унапређење постојеће *IT* инфраструктуре. Све проблеме у раду можете пријавити и на порталима за пријављивање грешака пројекта који се налазе „upstream“, будући да се све ове апликације покрећу на *Linux* дистрибуцијама а не на неком затвореном, власничком решењу, а *API* је одређеним делом заједнички.

oVirt је прави правцати дијамант, нуди потпуну функционалност коју нуде и остала „папрена“ решења за виртуелизацију, када се једном конфигурише, невероватно је стабилан (док год „не чачкате *backend*“) и ако сте спремни да се изборите са нешто лошијом документацијом, може представљати изузетно битан сервис у вашем информационом систему; ако нисте – купите *RHEV*, нећете зажалити. Ако вас је овај чланак заинтересовао за *oVirt*, на *YouTube*-у можете пронаћи неколико одличних видео туторијала који ће вам показати шта све *oVirt* може.

Корисни линкови :

- [1] <http://www.ovirt.org/Home>
- [2] <http://lists.ovirt.org/>
- [3] <http://resources.ovirt.org/releases/>
- [4] <http://www.youtube.com/watch?v=C4gayV6dYK4>

Преглед популарности *GNU/Linux* /*BSD* дистрибуција за месец август

Distrowatch

| | | |
|----|-------------|-------|
| 1 | Mint | 2778< |
| 2 | Debian | 1881> |
| 3 | Ubuntu | 1641> |
| 4 | Puppy | 1150> |
| 5 | Mageia | 1148< |
| 6 | Manjaro | 1108> |
| 7 | openSUSE | 1057> |
| 8 | Fedora | 1022> |
| 9 | PCLinuxOS | 956< |
| 10 | OS4 | 909> |
| 11 | elementary | 884> |
| 12 | Arch | 858> |
| 13 | LXLE | 805> |
| 14 | SparkyLinux | 715> |
| 15 | FreeBSD | 696< |
| 16 | Sabayon | 652> |
| 17 | CentOS | 630< |
| 18 | Kali | 624< |
| 19 | CrunchBang | 617> |
| 20 | Kwheezy | 603> |
| 21 | Lubuntu | 495< |
| 22 | Bodhi | 467> |
| 23 | SolusOS | 465> |
| 24 | Zorin | 456= |
| 25 | Kubuntu | 446< |

Пад <
 Пораст >
 Исти рејтинг =
 (Коришћени подаци са *Distrowatch*-а)

ISPConfig hosting control panel

Аутор: Дејан Чугаљ



ISPConfig је *hosting control panel* за сервере под *GNU/Linux* оперативним системом. Пројекат је отвореног кода, а примарни задатак му је манипулација над једним или више сервера (енгл. *multiserver management*) са једне локације. Под манипулацијом мислимо на управљање (администрирање) *web* сајтовима, управљање налозима електронске поште, *DNS*-ом (*Domain Name System*) и то све преко вашег *web* прегледача (*web-based interface*). Пројекат је покренут у *Projektfarm GmbH* у Немачкој у јесен 2005. године, и лиценциран је под *BSD* лиценцом.

Ако бисмо некако могли другим речима да опишемо сврху *ISPConfig* пројекта, то би било упоређивање са

такође *hosting panel*-ом који је у данашње време најзаступљенији по комерцијалним компанијама које нуде *hosting* услуге, под називом *cPanel* (корисна адреса <https://cpanel.net/>). Ако сте икада били у могућности да правите *web* сајтове или да их постављате на неки комерцијални *hosting* сервер, скоро сигурно сте се и сусрели са *cPanel*-ом и његовим корисним функцијама. Иако је *cPanel* проверен и сигуран панел за управљање сервером, цена лиценце и није баш тако мала.



Заједница отвореног кода нуди нам



ISPConfig, замену за cPanel, и наравно, све је отвореног кода и бесплатно је. У случају да хоћете да покренете своју hosting компанију, а немате средстава, или сте пожелели да своје сајтове држите на свом сопственом серверу, ЛиБРЕ! часопис вам представља ISPConfig пројекат, hosting control panel за сервере под GNU/Linux оперативним системима.

Основне карактеристике

- Манипулација над више сервера из једног control panel-а.
- Подржани web сервери Apache2 (<http://httpd.apache.org/>) и nginx (<http://nginx.org/>).
- Менаџмент сервера за електронску пошту са подршком виртуелних mail корисника.
- DNS сервер менаџмент (Domain Name Management – BIND, Powerdns и MyDNS).
- Виртуелизација (OpenVZ <http://openvz.org/>).
- Администраторска подршка за управљање препродавцима и клијентима.
- Лака конфигурација и подршка за кластер (clusters) и сервере огледала (mirroring).
- MySQL клијент, приступ бази података преко phpMyAdmin-а <http://www.phpmyadmin.net/>.
- FTP, SFTP, SCP
- Webalizer и/или AWStats статистика
- SSL
- SSI
- PHP (Apache2: mod_php, FCGI, CGI и suPHP; nginx: FCGI (PHP-FPM))
- Ватрени зид (firewall)

- Модул за мониторинг сервера

Инсталација

Прво је потребно посетити ISPConfig матични сајт на адреси <http://www.ispconfig.org/> и преузети цео пројекат. Датотека коју преузимамо је *ISPConfig-3.0.5.2.tar.gz*. Након преузимања потребно је распаковати је на неку лако доступну локацију вашег тврдог диска.



Инсталација је поприлично лака и аутоматизована, преко PHP скрипте, и извршава се командом:

```
php -q install.php
```

Ако се инсталира на серверу под Debian/Ubuntu оперативним системом, требало би да су скоро сва почетна подешавања довољна и да сервер буде спреман и функционалан, али искуство нам говори да је подешавање мало захтевније и да уме да буде „пипаво“. Напоменимо да се овим чланком нећемо упуштати у детаље самог подешавања, али савет који имамо је да се све инсталира на некој машини којој је намена управо да буде

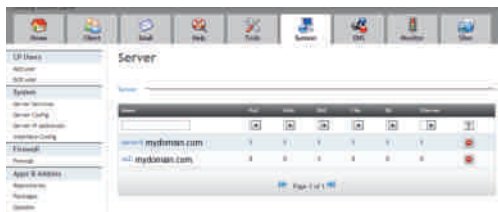
сервер, те из тог разлога је најбоље инсталирати *Ubuntu* сервер дистрибуцију и на тако новоинсталираном серверу подићи *ISPConfig* (прим. аут.).

Кориснички панел

Након успешне инсталације, и одласком на адресу коју сте поставили приликом процеса инсталације у формату:

```
http(s) :<hostname> :8080/  
http(s) :<ip_address> :8080/
```

приказаће вам се поље у које треба да унесете ваше корисничко име и лозинку (основна подешавања праве администраторског корисника са корисничким именом *admin* и лозинком *admin*) и након успешног пријављивања на систем приказује вам се кориснички панел.



ISPConfig нуди три нивоа приступа корисника.

1. Администратори

Имају потпун приступ систему али не и *root* серверу и комплетном конфигурационом делу система.

2. Препродавци (*Resellers*)

Имају приступ скоро свим модулима осим онима за конфигурацију система. Препродавцима је забрањен приступ модулима које је администратор њима забрадио. Могу креирати нове кориснике.

3. Клијенти

Клијенти могу креирати сајтове, налоге електронске поште итд.

Ако сте некад дошли до рачунара који је помало застарео, сувишан или једноставно не знате шта бисте могли са њим, пробајте да му нађете место које ће представљати ваш сервер за хостовање сајтова, како ваших личних, тако и ваших пријатеља, колега. Заједница отвореног кода нам је дала изврсно решење које управља свим задацима потребним за *hosting* услуге, док ће се ЛиБРЕ! часопис потрудити, можда у неком будућем серијалу, да представи детаљну инсталацију и подешавање *ISPConfig control panel*-а.

Корисни линкови:

- [1] <http://www.ispconfig.org/>
- [2] <http://www.howtoforge.com/forums/forumdisplay.php?f=23>
- [3] <http://sourceforge.net/projects/ispconfig/?source=dlp>



Git (8. део)

Команде

Аутор: Златан Васовић

Припремили смо мали подсетник са свим командама и објашњењем за шта служе.

Подешавање и конфигурација

- **help** - помоћ
- **config** - конфигурација *Git*-а

Клонирање и прављење складишта програмских пакета

- **init** - иницијализује складиште програмских пакета
- **clone** - клонира складиште програмских пакета

Снимање

- **status** - приказује стање складишта програмских пакета
- **diff** - приказује измене које су направљене у односу на последњи *commit*
- **add** - додаје датотеку у наредни *commit*
- **rm** - брише и додаје датотеку у



наредни *commit*

- **mv** - мења име датотеке и додаје датотеку у наредни *commit*
- **commit** - додаје све измене у *commit*
- **reset** - ресетује одабран(е) *commit*(е)

Развојне гране

- **branch** - служи за основно управљање развојним гранама
- **checkout** - пребациује на другу развојну грану
- **merge** - спаја развојне гране
- **mergetool** - исправља грешке при спајању грана
- **log** - приказује *log* складишта програмских пакета
- **stash** - чува све измене које нису комитоване (енгл. *commit*)
- **tag** - чува *commit*-е у ознакама (*tags*), као да су верзије

Дељење и ажурирање пројекта

- **fetch** - довлачи све измене из одређеног *remote*-а
- **pull** - довлачи све измене из одређеног *remote*-а и аутоматски покушава да их споји са тренутном развојном граном



- **push** - објављује измене које смо направили
- **remote** - управља *remote*-овима
- **submodule** - додаје складиште програмских пакета као *submodul* тренутног

Поређење и преглед

- **diff** - приказује измене које су направљене у односу на последњи *commit*
- **show** - приказује информације ознака
- **log** - приказује *log* складишта програмских пакета
- **shortlog** - приказује кратак *log* складишта програмских пакета
- **describe** - прави људима јасно име *commit*-а

Закрпе

- **am** - прихвата закрпу послату преко *email*-а генерисану уз **git diff** команду
- **apply** - прихвата закрпу послату преко *email*-а генерисану уз **git format-patch** команду
- **cherry-pick** - бира један *commit* и пребацује га на другу грану
- **rebase** - други начин спајања грана

Исправљање

- **blame** - прегледа датотеке и помаже при исправкама
- **bisect** - бинарни преглед

Email

- **am** - прихвата закрпу послату преко

- email*-а генерисану уз **git diff** команду
- **apply** - прихвата закрпу послату преко *email*-а генерисану уз **git format-patch** команду
- **format-patch** - генерише *email* са закрпом
- **send-email** - шаље *email* са закрпама у *email* фасциклу са нацртима
- **request-pull** - захтева *pull*

Спољни системи

- **svn** - увози *SVN* пројекат
- **fast-import** - увози *Perforce* пројекат

Администрација

- **clean** - једноставно „чисти“
- **gc** - ручно бира објекте са паковање
- **fsck** - проверава базу података и интегритет
- **reflog** - приказује *log* са референцама
- **filter-branch** - мења историју складишта програмских пакета, често се назива „нуклеарном опцијом“
- **instaweb** - покреће *GitWeb*
- **archive** - архивира одређени *commit*, таг или развојну грану

Серверска администрација

- **daemon** - мрежни „демон“ који покреће *Git* протокол
- **update-server-info** - команда која осигурава да ће *HTTP* довлачење (*fetching*) и клонирање радити добро

...

Ово је крај нашег серијала. Надамо се да смо помогли око упознавања са *Git SCM*-ом.



Mixxx (1. део)

Мултиплатформски DJ програм отвореног кôда



Аутор: Александар Тодоровић

Увод у DJ-инг

Сви знамо бар по једног DJ-а и сви знамо да су они ту да нас забаве када одемо у излазак викендом, и да нам пруже незаборавно искуство. Међутим, колико нас зна шта заправо ради DJ када стане за свој пулт, и чиме ради? Када је опрема у питању, DJ има неколико опција: може користити грамофоне, може користити CD плејере, може користити лаптоп са или без MIDI

контролера, а може и комбиновати ове три опције на било који начин. Модернизацијом је дошло до тога да су DJ-еви већином прешли на рад на лаптопу и данас је видети DJ-а који ради искључиво са грамофонима права реткост. Модернизацијом и преласком на лаптоп посао DJ-а је у великој мери олакшан. DJ-еви са грамофонима, иако су реткост, данас се највише цене, зато јер је потребно много више вештине у раду са грамофонским плочама него са лаптопом, а DJ са лаптопом може бити практично свако (доказ је пример Paris



Hilton и Pauly DelVecchi-а из Jersey Shore-а који се представљају као DJ-еви). Данас смо дошли у ситуацију када у сваком селу постоје бар 2-3 DJ-а који већином купе лаптоп, преузму пиратски комерцијални програм и одмах себе називају DJ-ем. Они који воле то умеће и који су спремни да уложе године сакупљајући новац за нормалну опрему и за квалитетне песме, данас су реткост, а док не послушате таквог DJ-а, ретко можете да цените посао DJ-инга уопштено.

Mixxx-ова комерцијална браћа

Што се тиче лаптопа и рада са лаптопом када је DJ-инг у питању, слободан софтвер је далеко иза комерцијалне конкуренције. Избор комерцијалног софтвера је далеко већи: *Traktor*, *Serato DJ/Itch*, *Deckadance*, *Virtual DJ*, *Ableton Live* и други, док постоји само једна алтернатива отвореног кода која је вредна спомена – *Mixxx*. Креирати неки програм који ће DJ-евима омогућити све што једном данашњем DJ-у треба, много је компликовано, и давање таквог кода бесплатно је стварно храбар потез. Софтвер за DJ-еве мора у реалном времену да пушта неколико аудио датотека и да омогући потпуну контролу над тим аудио датотекама, као и додавање ефеката, смањивање и појачавање гласноће песама, беспрекорно детектовање темпа, омогућавање ручног подешавања темпа, синхронизације темпа једне песме са темпом друге песме, инстантно пребацивање са једног дела песме на било који други, подржавање рада са стотину MIDI контролера као и са микрофонима, CD

плејерима и грамофонима, док у исто време мора да пружа потпуну стабилност, јер *crash*-овање програма и потпуни престанак музике за време наступа живео једноставно не долази у обзир. Много труда и рада је потребно да се уложи да би се направио такав програм и да би био употребљив, а *Mixxx* заједница се потрудила да направи један такав, за нас, љубитеље отвореног кода.

Увод у Mixxx

Mixxx је програм отвореног кода који је доступан на *Windows*, *Mac* и *Linux* платформи. Пројекат је почео давне 2001. године, а на његовој изради је учествовало преко стотину програмера који имају страст према музици и DJ-ингу. Тренутно је актуелна 1.11.0 верзија која је изашла у мају ове године, а читав програм је написан користећи *Qt*, *C++*, *JavaScript* и *XML*.

Предности и мане

Mixxx подржава неколико десетина различитих графичких тема и неколико десетина најпознатијих MIDI контролера. Подржава MP3, OGG, FLAC, WAV и AIFF музичке формате, има подршку за грамофоне и микрофоне те подржава *broadcast*, што значи да врло лако можете креирати своју *online* радио станицу и пуштати музику својим пријатељима преко овог програма. Велики минус је то, што *Mixxx* нема подршку за рад са четири *deck*-а, као ни подршку за основне ефекте попут *delay*, *reverb* и *flanger*, што је недопустиво за један програм за DJ-инг, а еквилајзер му је далеко лошији од комерцијалне конку-



ренције. Међутим, постоји и неколико функционалности у којима се овај програм у великој мери истиче над конкуренцијом. Пре свега бисмо истакли готово тренутно учитавање и анализу песме, што у комерцијалним програмима зна да потраје и до минут у зависности од рачунарских ресурса. Ретко који комерцијални DJ софтвер подржава оволико различитих тема, а неке од ових тема су јако привлачне изгледом. Ретко који комерцијални софтвер подржава и оволику количину MIDI контролера које није потребно ручно програмирати. Програм може да ради на много мањој латенцији (закашњење звука од програма до звучника) у односу на конкуренцију и троши много мање ресурса приликом рада. Подршка за Linux се не може наћи нити у једном другом софтверу за DJ-еве, а ни подршка за грамофоне није у великој мери распрострањена међу оваквим програмима.

Преглед интерфејса (*Browser* датотека) је у принципу исти као и код сваког софтвера за DJ-еве – веома непрегледан, док се не искључе непотребне опције. Претрага ради без икаквог кашњења, чак и у нашој колекцији од неких две хиљаде песама. Подржава учитавање колекције из iTunes-а.



Слика 1 Browser датотека

Миксер секција је лепа, прегледна и

функционална. Већ смо нагласили да је мањак квалитетних ефеката практично недопустив, и да је сам еквилајзер веома неквалитетан, али морамо истаћи да темпо ради савршено и да је детекција много боља него код неких програма који су Mixxx-у конкуренција.



Слика 2 Миксер секција

Deck секција је веома једноставна, нема превише опција, сам waveform преглед није савршен и графички није баш најлепши, али задовољава потребе. Једино недостаје могућност зумирања.



Слика 3 Deck секција



Закључак

Све у свему, *Mixxx* је фина алтернатива и добар начин да се уштеди неколико стотина долара, али по функционалности је још увек веома далеко иза комерцијалне браће. Иако има већину функција коју има и комерцијална конкуренција, квалитет тих функција је на nižем степену. Садржи неколико квалитетних функција на којима би му и комерцијална браћа завидела (па и да позајме део кода). Програм није у потпуности стабилан и има неке недостатке, али не смео заборавити оно што нам пружа: мноштво могућности са отвореним кодом, подржавање веома великог постотка данашњих *MIDI* контролера, фин и једноставан интерфејс и једноставан и функционалан поступак рада. Топло га препоручујемо за кућну употребу и журке у кругу пријатеља, али мораће проћи још неколико дорада пре него што постане употребљив у клубовима.

У следећем броју ћемо вам показати како да га користите и како да направите и снимите почетнички микс.

Корисни линкови:

- [1] <http://mixxx.org/>
- [2] <http://mixxx.org/manual/latest/>
- [3] <http://mixxxblog.blogspot.com/>
- [4] <http://mixxx.org/wiki/doku.php>
- [5] <http://mixxx.org/forums/>
- [6] <https://en.wikipedia.org/wiki/Mixxx>

Arduino контролер (6. део)

Аутор: Стефан Ножинић

У овом броју ћемо приказати начин функционисања серијске комуникације између рачунара и *Arduino* контролера. Важно је напоменути да за разлику од осталих платформи, код *Arduino*-а је ово веома елегантно урађено, што кориснике ослобађа многих брига. Наиме, серијска комуникација се обавља преко *USB* кабла који већ повезује *Arduino* и ваш рачунар, чиме сте ослобођени било којих хардверских захтева.

Реч-две о *Arduino* хардверским спецификацијама

Важно је нагласити да *Arduino* и поред *USB* конекције, која је у већини случајева једноставније решење, серијску комуникацију може обављати и преко портова 0 (*RX*) и 1 (*TX*). Ово је добро у случајевима ако желимо да, уместо рачунара, *Arduino* врши серијску комуникацију са неким другим уређајем. Такође је важно напоменути и да поред повезивања ова два *pin*-а, потребно је и масу (*GND*) једног



повезати на масу другог уређаја.

Програмирање – серијска комуникација

Када је *Arduino* уредно повезан на уређај са којим треба да комуницира, све што је потребно јесте испрограмирати *Arduino* да шаље и прима податке. *Arduino* већ има функције за серијску комуникацију које се налазе у класи *Serial* која има статичке методе.

Овде вам представљамо само најбитније функције са објашњењем шта оне заправо раде:

- **Serial.begin(int)** Започиње серијску комуникацију брзине задате параметром (обично 9600), параметар се задаје у *b/s*. Ова функција се најчешће користи унутар *setup()* процедуре.
- **Serial.end()** Обуставља серијску комуникацију и тиме омогућава нормално коришћење *pin*-ова 0 и 1.
- **Serial.available()** Враћа број који представља колико података треба да се прочита (0 ако је *buffer* празан).
- **Serial.read()** Чита један бајт.
- **Serial.write(byte)** Шаље један бајт на

уређај, враћа број послатих бајтова.

Serial Monitor

Arduino радно окружење обезбеђује и *Serial Monitor* који може да чита поруке које *Arduino* пошаље и исто тако да шаље на *Arduino*. Ово знатно олакшава уклањање грешака ако до њих дође. *Serial Monitor* покрећете кликом на **Tools -> Serial Monitor** или једноставно притиском на комбинацију тастера **Ctrl + Shift + M**. Требало би да се појави прозор чији графички интерфејс подсећа на клијент за ћаскање. Ово нимало не чуди јер то и јесте „ћаскање између рачунара и *Arduino* уређаја путем серијске комуникације”.

За крај серијала

Како је ово последњи текст у овом серијалу о *Arduino* платформи, представљамо вам неке занимљиве линкове који се детаљније баве овом тематиком:

- [1] <http://arduino.cc/en/Guide/HomePage>
- [2] <http://arduino.cc/en/Tutorial/HomePage>
- [3] <http://arduino.cc/en/Reference/HomePage>
- [4] http://en.wikipedia.org/wiki/Electronic_component



Software Freedom Day

21. септембар 2013



Ове године Дан слободе софтвера ће бити обележен на 3 локације у Србији:

Београд - Галерија OZONE, Андрићев венац 12

Чачак - Дом културе, Трг устанка 2-6.

Ловћенац - Дом културе

Биће уприличене трибине, презентације, радионице, дељен бесплатни FLOSS рекламни материјал, CD-ови са Linux оперативним системом...

Дођите да се дружимо. Улаз је бесплатан.

