



Abíčko

Časopis serveru abclinuxu.cz

Leden 2008

Vychází také na CD-ROM jako příloha časopisu

PC WORLD

Editoriál

Vítejte u čtení časopisu Abíčko.

Abíčko vychází jako měsíční příloha serveru <http://www.abclinuxu.cz> a obsahuje výběr toho nejzajímavějšího obsahu, který zde byl v minulém měsíci publikován. Touto formou chceme předat čtenářům informace v snadno čitelné podobě vhodné i pro tisk.

Cílem serveru <http://www.abclinuxu.cz> je pomáhat všem uživatelům Linuxu, nezávisle na jejich zkušenostech, platformě či použité distribuci. Motorem, který nás pohání vpřed, je idea vzájemné pomoci a spolupráce. Proto i velkou část obsahu tvoří samotní uživatelé. Zapojit se může kdokoliv, tedy i vy.

Na <http://www.abclinuxu.cz> najdete rozsáhlou databázi návodů na zprovoznění hardwaru pod Linuxem, velice aktivní diskusní fórum, podrobné návody a tutoriály, recenze, archiv ovladačů, informace o linuxovém jádře (včetně populárních Jaderných novin), rozcestník po ostatních linuxových serverech nebo velmi oblíbené blogy. Každý registrovaný uživatel si jej může založit a psát si do něj poznámky nejen o Linuxu.

V neposlední řadě chceme upozornit také na výkladový [slovník pojmů](#) a vznikající [elektronickou učebnici Linuxu](#), na níž se můžete podílet i vy!

Náměty na články zasílejte do konference našich autorů: info@abclinuxu.cz. Sponzoring Abíčka a jiné formy reklamy si objednávejte na adrese: info@stickfish.cz. Ostatní dotazy směřujte na adresu: info@abclinuxu.cz.

Server <http://www.abclinuxu.cz> provozuje firma Stickfish s.r.o., která poskytuje profesionální služby v oblasti Linuxu firmám i jednotlivcům. Zabývá se hlavně bezpečností, instalacemi Linuxu a konfigurací síťových služeb. Více na <http://www.stickfish.cz>.

©2007–2008 Stickfish s. r. o. a autoři článků

Editor a sazba: Vlastimil Ott

Pro nekomerční účely smíte tento dokument jakkoliv šířit v tištěné i digitální podobě. V ostatních případech nás požádejte o svolení na adrese info@abclinuxu.cz.

Typografické konvence

Ve výpisech **zdrojových textů** mohou být použity znaky `\\`. Značí přechod na nový řádek, který ovšem *není* součástí samotného zdrojového textu, byl přidán editorem z důvodu lepšího vzhledu případně nemožnosti text formátovat bez jejich použití.

Obsah

Distribuční novinky – 52/2007	6
Letem světem	6
SCO jde pomalu ke dnu	6
Technologická ředitelka OLPC opouští projekt	6
Krok směrem k OSS ve Vietnamu	6
GoboLinux 014	6
FreeBSD 7.0-RC1	7
Endian Firewall 2.2 Beta 1	8
Annvix 3.0	8
Distribuční rada: snadné rozbalování s unp	8
Distribuční novinky – 1 a 2/2008	9
Letem světem	9
McAfee má obavy z open source	9
Naděje pro open source v britském školství	9
Intel opouští projekt OLPC	10
Potíže ve vedení Gentoo Foundation	10
Novinky o svobodných ovladačích grafických karet	10
AMD/ATI	10
NVIDIA	11
DesktopBSD 1.6	11
KANOTIX 2007 RC7	11
Linux Mint 4.0 Beta „Fluxbox“ a „Debian“	12
Distribuční rada: rychlé převody mezi časovými pásmy v konzoli	12
Distribuční novinky – 3/2008	14
Letem světem	14
Poslední dění okolo Gentoo	14
Obnova Gentoo Foundation	14
Návrat buletinu	14
V čem spočívá krize?	14
Mandriva a Turbolinux se stávají partnery	15
Neoficiální vývojový plán KDE 4.1	15
Qt je nyní i pod GNU GPLv3	15
Linux dostává šanci v ruském školství	15
Zenwalk Linux 5.0	16
openSUSE 11.0 Alpha 1	16
Distribuční rada: dodatečné nastavení některých druhů zrcadel (Gentoo)	17
Distribuční novinky – 4/2008	18
Letem světem	18
Lotus Notes 8.5 bude mít plnou podporu pro Ubuntu	18
Hewlett-Packard uvolnil nástroj pro správu OSS	18
Poslední novinky ze světa svobodných grafických ovladačů	18
Plány na Gentoo 2008.0	19
Německé Ministerstvo práce nasadilo Linux	19
Parsix GNU/Linux 1.0r0	19
GoblinX 2.6 „Micro“	20
Mandriva Linux 2008.1 Beta 1	20

Distribuční rada: přednačítání souborů s readahead-list	21
Gentoo ebuild – 6 (závislosti, sloty, restriktce)	22
Závislosti	22
Závislosti potřebné k sestavení	22
Závislosti potřebné ke spuštění	22
Poinstalační závislosti	22
Implicitní systémové závislosti	22
Syntaxe proměnných *DEPEND	22
Sloty	24
Restriktce	24
Gentoo Linux – Portage Overlay	25
K čemu to je	25
Jak začít	25
Struktura a kategorie	25
Eclass	25
Patche	26
Maskování balíčků	26
Záznamy změn	27
Na co si dát pozor	27
Jaké je KDE 4.0.0	28
Akcelerovaný KWin	28
Plasma	29
KickOff a Krunner	30
ApliKace	31
Dolphin	31
Konqueror	31
Okular	32
Gwenview	32
System Settings	32
Ostatní	32
Technologie	32
Závěrem	34
Knihy: OpenOffice.org WRITER a CALC	35
Ty odborné výrazy...	35
Linux?	35
OpenOffice.org Writer	35
OpenOffice.org Calc	36
Hodnocení	36
Viac či menej otvorene grupvérové systémy	38
Čas pre grupvér	38
Komerčné riešenia	38
Otvorené riešenia	38
Budúcnosť je na webe	39
Spoločná charakteristika	39
Oplatí sa?	40
Inkscape – 6 (vizitka)	41
Základní pravidla	41
Přípravné práce	41

Obsah vizitky	41
Samotná sazba vizitky	42
Ještě logo	44
Tip – tisk vizitek na vlastní tiskárně	45
Ořezové značky	45
Zdrojové dokumenty	46
Bankovní Komerční banky v Linuxu	47
Historie bankovní KB	47
Do banky v Linuxu	47
Mozilla Firefox	48
Opera	48
Konqueror	49
Další testy	49
Certifikační průvodce	49
64bitový Linux	50
Zhodnocení	50
Regulární výrazy	52
Obecné regulární výrazy	52
Rozšířené regulární výrazy	53
GNU sed	54
Perl	55
Regulární výrazy v praxi	55
Zpravodaj o Víně – 339	56
Dostane se Wine do Mac OS X Leopard?	56
Statistika autorů za uplynulé roky	56
Audit Bugzilly	57
Konec projektu Alky	57
Podpora Run/RunOnce	57
Změny v AppDB	59
Změny od správců programů	59
Změny od uživatelů	59
Jaderné noviny – 12. 12. 2007	60
Aktuální verze jádra: 2.6.24-rc5	60
Citáty týdne: Ingo Molnár, Ted T'so, Linus Torvalds	60
Jednodušší syslety	60
Přískrcování zápisu nečistých stránek na disk [writeout]	62
Nové a staré chyby	63
Jaderné noviny – 19. 12. 2007	65
Aktuální verze jádra: 2.6.24-rc5	65
Citáty týdne: Andrew Morton, Ingo Molnár	65
Krátké zmínky: kerneloops, read-mostly a port 80	65
Kerneloops	65
Read-mostly	66
Prodlevy I/O portů	66
revoke() se vrací	67
Jaderné noviny – 2. 1. 2008	69
Aktuální verze jádra: 2.6.24-rc6	69
Citát týdne: David Miller	69

SLUB v nesnázích	69
Racionalizace scatter/gather řetěží	70
Jaderné noviny – 1/2008	73
2.6.24-rc4, „trošku pozdě“	73
2.6.24-rc5, konečná verze do Vánoc nebude	73
Citát: Vyhýbání se OOM killerovi	74
Sbírání dat z jaderných oops	74
Oprášení jádra 0.01	74
Vydání stabilního jádra 2.4.36	74
Citát: Proti tomu se postavím	74
Ospalý Linux	75
Citát: Výhody TuxOnIce	75
2.6.24-rc7, spousta malých změn	75
Citát: Menšina	75
Dekódování oops	76

Distribuční novinky – 52/2007

SCO jde pomalu ke dnu. Technologická ředitelka OLPC opouští projekt. Krok směrem k OSS ve Vietnamu. GoboLinux 014. FreeBSD 7.0-RC1. Endian Firewall 2.2 Beta 1. Annvix 3.0. Distribuční rada: snadné rozbalování s unp.

Luboš Doležel

Letem světem

Tento týden přinesl dvě opravná vydání Debianu: [Debian GNU/Linux 4.0r2](#) ⁽¹⁾ (Etch) a [Debian GNU/Linux 3.1r7](#) ⁽²⁾ (Sarge). [Zenwalk Linux 5.0 Beta](#) ⁽³⁾ přináší několik důležitých inovací. [Parsix GNU/Linux 1.0r0 Test 2](#) ⁽⁴⁾ má lepší skripty sadu skriptů pro živá CD (používá Squashfs s LZMA kompresí), nasazuje TuxOnIce, lwlwifi, GNOME 2.20.2 a jiné novinky. [grml 1.1 RC1](#) ⁽⁵⁾ je živé CD založené na Debianu obsahující kolekci programů a vlastních skriptů pro systémové administrátory. [PCLinuxOS „GNOME 2.21.2“](#) ⁽⁶⁾ je varianta distribuce se zaměřením na GTK+ a GNOME aplikace. [VectorLinux 5.9 Beta 1 „Live CD“](#) ⁽⁷⁾ obsahuje GParted a aktualizovaný instalátor. [Musix GNU+Linux 1.0r3 Test 4](#) ⁽⁸⁾ odstraňuje několik problémů a nabízí čerstvý hudební software.

SCO jde pomalu ke dnu

SCO se opět dostalo ⁽⁹⁾ o krůček blíže své smrti poté, co bylo vyřazeno z Nasdaq. Vyřazení hrozí firmám, kterým se hodnota akcií dostane pod jeden dolar, a současná hodnota akcií SCO se pohybuje okolo deseti centů. Pád firmy nastartovalo soudní rozhodnutí, podle kterého je to Novell (a ne SCO), kdo vlastní autorská práva na Unix. Přitom právě dosti pochybná obvinění o porušování autorských práv Unixu byla základem sporů SCO s Novellem a IBM. Nyní bude soud rozhodovat, kolik firma dluží Novellu za licenční poplatky, přičemž částka se může vyšplhat až k 35 milionům dolarů. Groklaw zanalyzoval finanční situaci SCO a nevypadá to, že by bylo z čeho peníze brát – to, co SCO zbylo, půjde především do kapes právníků.

Technologická ředitelka OLPC opouští projekt

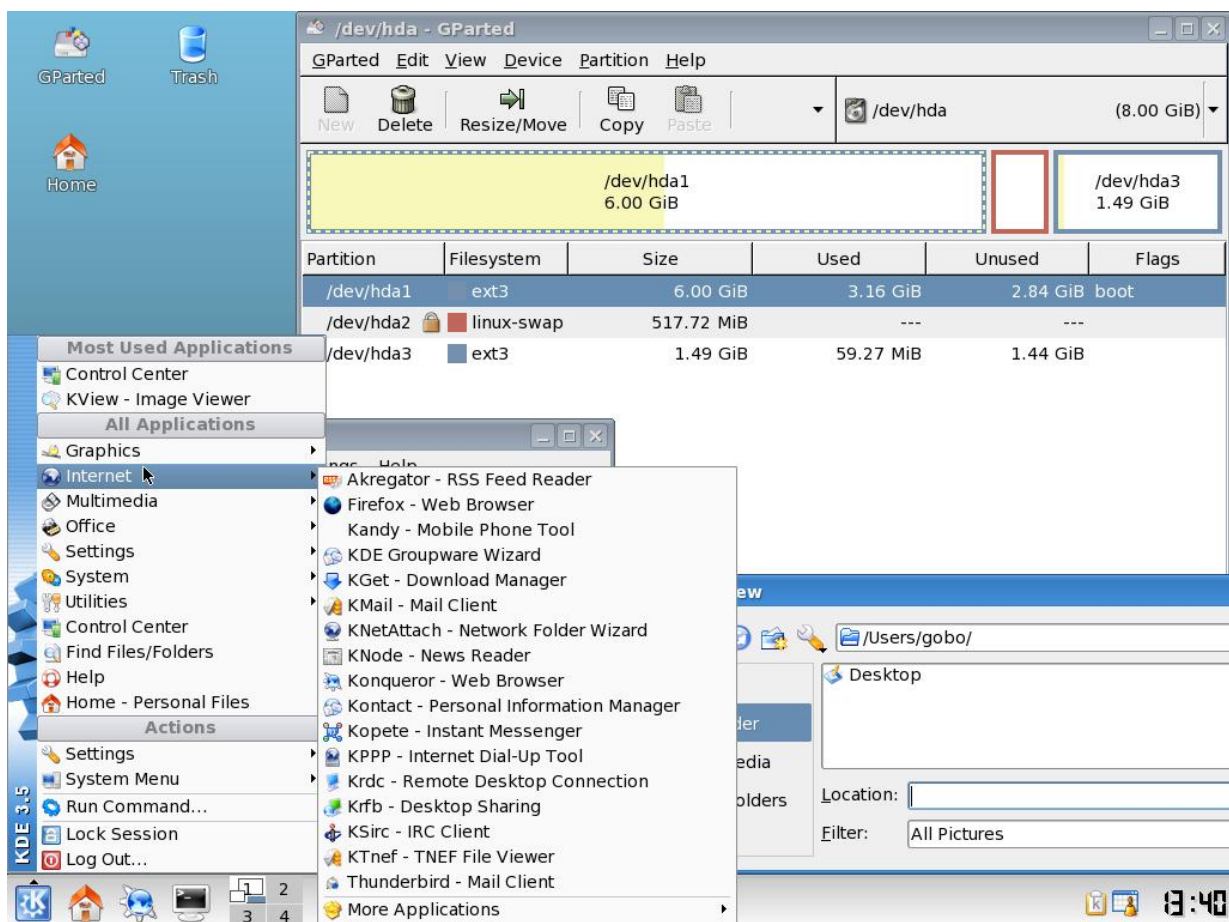
Mary Lou Jepsen, která se projektu [OLPC](#) ⁽¹⁰⁾ účastnila už prakticky od začátku (rok 2005), [opouští svůj post technologické ředitelky](#) ⁽¹¹⁾. V projektu byla zodpovědná za vývoj hardwaru (hlavně displeje) a přinesla světu zajímavé novinky v oblasti úspornosti hardwaru – spotřeba notebooku OLPC se pohybuje mezi 2W až 8W, což znamená běh notebooku až na 21 hodin. Ve svém e-mailu oznámila, že založí firmu, kde své vynálezy komerčně využije – na OLPC však nezapomene a bude projektu nadále poskytovat podporu. V OLPC mezitím nelení a pracují dále – nyní je to například výroba energie pomocí [kravského pohonu](#) ⁽¹²⁾. S OLPC si ale „hrají“ i ostatní lidé: kryptology možná zaujme [distribuce kvantových klíčů](#) ⁽¹³⁾ mezi dvěma laptopy.

Krok směrem k OSS ve Vietnamu

Vietnamské Ministerstvo informatiky a telekomunikací [vydalo doporučení](#) ⁽¹⁴⁾, podle kterého by vládní úřady měly používat open source software, kdykoliv probíhá rozhodování o implementaci různých projektů. Zároveň s tím vyšel seznam open source projektů, ve kterém figurují známá jména jako [OpenOffice.org](#) či produkty Mozilly. Ministerstvo zatím nesdělilo, zda učinilo tento krok jen aby se ušetřily peníze, nebo má věc nějaký hlubší význam (např. podporu rozvoje OSS v zemi). Ve Vietnamu je velmi rozšířené nelicencované používání softwaru, které se rozmohlo i ve firemní sféře.

GoboLinux 014

GoboLinux⁽¹⁵⁾ je modulární distribuce s neobvyklou hierarchií souborového systému – soubory balíčků nejsou rozčleněny podle typu/účelu (/etc, /usr/bin, /usr/share), ale podle programu (/Programs/Autoconf). Původní klasická hierarchie v systému kvůli kompatibilitě existuje, ale je skrytá očím uživatele pomocí jaderného modulu GoboHide⁽¹⁶⁾. Distribuce vychází v podobě instalovatelného živého CD (pro i686). Instalátor distribuce funguje v textovém i grafickém režimu, ale programátoři jej psali jen jednou: distribuce používá vlastní pythonovou knihovnu AbsTK (Abstract Toolkit), která zvládá zobrazování uživatelského rozhraní přes ncurses i Qt. Na aktuálním živém CD verze 014 najdete prostředí KDE 3.5.8 běžící pod X.org 7.2 a dále celou řadu jiných programů včetně OpenOffice.org 2.2.0 – pro kompletní seznam vizte poznámky k vydání⁽¹⁷⁾.



FreeBSD 7.0-RC1

Vyšlo FreeBSD 7.0-RC1⁽¹⁸⁾ a konečná verze by dle původních plánů měla být hotová v polovině ledna, nicméně skluz ve vývoji je nyní přibližně 17 dnů. Zatím se můžeme podívat na novinky, které se do systému dostaly. Jádro má základní podporu časovače HPET, uspání a probouzení by u APM mělo fungovat lépe a jádro také podporuje bezpečnostní audit událostí. Přibyl subsystém firmware pro načítání binárních dat do jádra skrze speciální modul. Byl také integrován RedZone, neboli ochrana proti poškození bufferů (poškození za běhu detekuje při voláních free a realloc). Co se hardwaru týče, dvoujádrové procesory mají ve výchozím stavu s povoleným SMP obě jádra dostupná, byl implementován ovladač kompatibilní s OpenIPMI (Intelligent Platform Management Interface) a systém má předběžnou podporu architektury Sun Microsystems UltraSPARC-T1. Kromě toho můžete FreeBSD spustit i na Xboxu⁽¹⁹⁾ (který je velmi blízký i386). Mezi ostatní novinky patří podpora AGP čipsetů ATI, IPv6 ve firewallu ipfw a ovladači ppp nebo podpora souborového systému XFS (pro čtení). Kromě FreeBSD 7.0-RC1 vyšlo v uplynulém týdnu také FreeBSD 6.3-RC2⁽²⁰⁾.

Endian Firewall 2.2 Beta 1

Endian Firewall ⁽²¹⁾ je čistě open source distribuce založená na Red Hat Enterprise Linuxu. **Je to** ⁽²²⁾ stavový firewall s možností nasazení proxy pro různé protokoly (HTTP, FTP, POP3, SMTP) včetně antivirové kontroly a průchozí e-maily mohou být kromě antiviru filtrovány také antisпамem. Web může být filtrován podle obsahu a v systému je snadné vytvořit VPN. Distribuce vychází ve dvou variantách: Community a Enterprise, která je určena pro firmy a vládní organizace. **Poslední betaverze** ⁽²³⁾ aktualizuje balíčky, vylepšuje GUI, nově podporuje více odchozích linek (uplinks), zvládá přesměrovávání portů z VPN klientů, odchozí i zónový firewall podporují ICMP, jednotlivá pravidla mohou být zadávána podle zón, rozhraní a MAC adres a další novinky.

Annvix 3.0

Annvix ⁽²⁴⁾ je kanadská serverová distribuce, jejímž cílem je poskytovat systém pro e-mailové, DNS, FTP a další servery. Vývojáři si dávají záležet na bezpečnosti a spolehlivosti, a proto integrují zajímavé věci: runit, aby služby vždy běžely, socklog pro bezpečné a spolehlivé logování, AppArmor pro zabezpečení na úrovni jádra atd. Typická instalace (LAMP server) zabírá na disku méně než 400 MB místa. Instalační CD může posloužit také jako záchranné živé CD. **Poslední verze 3.0** ⁽²⁵⁾ přechází z RSBAC na AppArmor, aktualizuje softwarovou výbavu a používá glibc 2.6.1. Tato distribuce dříve vystupovala pod jménem OpenSLS.

Distribuční rada: snadné rozbalování s unp

unp ⁽²⁶⁾ je perlový skript, který vám v konzoli usnadní práci s archivy či různě kódovanými soubory. Místo toho, abyste si museli pamatovat či vždy zjišťovat, jak se konkrétní typ archivu rozbaluje, stačí použít tento skript.

```
unp archiv.zip jinyarchiv.tar.bz2
```

Pokud chcete samotnému rozbalovacímu nástroji předat nějaké speciální argumenty, oddělte je pomocí `--` následovně:

```
unp archiv.tar.bz2 -- -C /var/tmp/vystup
```

unp má jeden zajímavý parametr – pomocí `-u` určíme, že obsah se má rozbalit do adresáře podle názvu archivu (bez přípony). Následující příkaz by tedy rozbalil archiv do podadresáře `materialy`:

```
unp -u materialy.rar
```

■

Distribuční novinky – 1 a 2/2008

McAfee má obavy z open source. Naděje pro open source v britském školství. Intel opouští projekt OLPC. Potíže ve vedení Gentoo Foundation. Novinky o svobodných ovladačích grafických karet: AMD/ATI, NVIDIA. DesktopBSD 1.6. KANOTIX 2007 RC7. Linux Mint 4.0 Beta „Fluxbox“ a „Debian“. Distribuční rada: rychlé převody mezi časovými pásmy v konzoli.

Luboš Doležel

Letem světem

Na svět přišly dvě vývojové verze velkých distribucí: [Mandriva Linux 2008 Spring Alpha 3](#) ⁽²⁷⁾ a [Ubuntu 8.04 Alpha 3](#) ⁽²⁸⁾ s PulseAudio, [Mozilla Firefoxem 3 Beta 2](#) a [X.org 7.3](#). [PCLinuxOS MiniME 2008](#) ⁽²⁹⁾ představuje minimalistickou verzi PCLinuxOS LiveCD. [SME Server 7.3](#) ⁽³⁰⁾ je serverový systém postavený na CentOS 4.6. Distribuce pro staré počítače MEPIS antiX aktualizuje balíčky: [verze 7.01](#) ⁽³¹⁾. [trixbox CE 2.4.0](#) ⁽³²⁾ nabízí zajímavý systém pro použití jako pobočková telefonní ústředna. [Ultima Linux 8.3](#) ⁽³³⁾ staví na Slackware Linuxu a má nástroj pro automatickou aktualizaci systému. Další uživatelsky přívětivou distribucí založenou na Slackwaru je [DARKSTAR Linux 2008.1](#) ⁽³⁴⁾. [Frugalware Linux 0.8 Pre 2](#) ⁽³⁵⁾ aktualizuje balíčky, přechází na [Octave 3.0.0](#) ⁽³⁶⁾ a přináší i jiné novinky. [Parsix GNU/Linux 1.0r0 Test 3](#) ⁽³⁷⁾ aktualizuje a pročištuje systém, vylepšuje nástroje pro běh živého systému a přidává nástroj pro konfiguraci X.org z Ubuntu. [Voltalinux 2.0](#) ⁽³⁸⁾ kombinuje Slackware Linux s balíčkovacím systémem „pkgsrc“ z NetBSD. [GoblinX 2.6 RC1](#) ⁽³⁹⁾ obsahuje pětici správců oken.

McAfee má obavy z open source

Společnost McAfee, která se zabývá vývojem antivirového softwaru, [vyjádřila ve své výroční zprávě obavy](#) ⁽⁴⁰⁾, podle kterých open source v míře, v jaké je v produktech McAfee používán, představuje pro firmu a investory riziko. Tvrdí, že jimi používaný open source software je pod nejednoznačnými licencemi, což může pro firmu mít nepředvídané následky. Obzvláště znepokojující je dle jejich slov to, že licence [GNU GPL](#) ⁽⁴¹⁾ se ještě nedostala před soud (vše bylo doposud vyřešeno mimosoudně), a není tak jistá právní platnost celého dokumentu. Dá se už jen odhadovat, zda McAfee k těmto prohlášením dohnala skutečnost, že vývojáři BusyBoxu začali právními cestami [bojovat za dodržování licence GPL](#) ⁽⁴²⁾, a to pro McAfee jednoduše není pohodlné. Také společnost Tivo před časem varovala investory, že kvůli GPL možná budou muset upustit od používání OSS, neboť právě přístup Tiva byl jedním z důvodů, proč byla nová verze GNU GPL (v3) s přísnějšími podmínkami vytvářena.

Poté, co zástupce Software Freedom Law Center vyřkl nesouhlas s tvrzením o nejednoznačnosti open source licencí, se pokusilo McAfee celou situaci uklidnit. Používání podobných obrátů je prý už po několik let běžná praxe i u jiných společností; jako příklad byl uveden Microsoft a Oracle. Přesto se McAfee prohlašuje za horlivého fanouška open source.

Naděje pro open source v britském školství

[Becta](#) ⁽⁴³⁾ je vládní organizace ve Spojeném království, která má za úkol vést používání informačních technologií v oblasti vzdělávání. V minulosti byla kritizována, že nedostatečně propaguje používání open source (a to i přesto, že uznala jeho výhody) a přispívá tak k zatěžování státního rozpočtu. [Nyní však došlo ke zlomu](#) ⁽⁴⁴⁾, když Becta vydala zprávu, podle které není přechod na Windows Vista a Office 2007 vhodný, neboť nové funkce produktů nevyvažují náklady. Obzvláště pak upozorňuje na rizika s interoperabilitou,

kteřá se mohou u Office 2007 vyskytnout. Bectě se nelíbí způsob, jakým Microsoft licencuje produkty (ten prý poškozujje konkurenční prostředí), a ani způsob implementace [OpenDocument Formatu](#) ⁽⁴⁵⁾ nepovažuje za ideální.

V prohlášení dále uvádí, jaké konkrétní kroky podnikne, aby se postavení open source ve školství zlepšilo:

- Vylepší stránku s přehledem o OSS a jeho výhodách pro školství na svém webu.
- Vypracuje přehled informací o použití open source řešení – poukáže na exemplární nasazení a zanalyzuje situaci.
- Ve spolupráci s open source komunitou sestaví katalog OSS vhodného pro použití v britských školách. Katalog by byl uvolněn pod licencí Creative Commons.
- Vydá rady, podle kterých se open source společnosti budou moci efektivně zapojit do systému dodávání softwarových řešení.

Intel opouští projekt OLPC

Když v červenci 2007 Intel vstupoval do OLPC ⁽⁴⁶⁾, vyvolával tento krok smíšené reakce, protože sám stál za produktem, který XO notebookům [OLPC](#) ⁽⁴⁷⁾ přímo konkuroval: [ClassMate PC](#) ⁽⁴⁸⁾. S jeho příchodem se začalo uvažovat o vytvoření levných notebooků OLPC na bázi Intelu (nyní se používá AMD), ale problém, který by se dal nazvat střetem zájmů, tím vyřešen nebyl. Probíhaly další diskuze a především Nicholas Negroponte (zakladatel OLPC) viděl pouze dvě možné varianty: buď se Intel projektu ClassMate PC vzdá, nebo odejde. [Negroponte říká](#) ⁽⁴⁹⁾, že Intel své členství v radě OLPC zneužíval ke shazování OLPC při snaze prodat svůj vlastní produkt – své členství používal jako důkaz toho, že o produktu ví mnoho a může tedy s jistotou prohlásit, že ClassMate je lepší. Intel dokonce jednou poslal člověka do Peru, aby tam kritizoval OLPC poté, co tam OLPC uzavřelo obchod. Negroponte nevěřil slibům, že se Intel zlepší a začne pro OLPC doopravdy dělat něco prospěšného. To je důvodem, proč se [cesty Intelu a OLPC rozdělily](#) ⁽⁵⁰⁾.

Potíže ve vedení Gentoo Foundation

[Daniel Robbins](#) ⁽⁵¹⁾ ve svém blogu [popisuje potíže](#) ⁽⁵²⁾, které teď provázejí organizaci Gentoo Foundation. Gentoo Foundation přišlo o svou zakládací listinu, což znamená, že prakticky neexistuje. Dle jeho slov nebyla situace vysvětlena a také nebyly zveřejněny kroky pro napravení tohoto stavu. Daniel Robbins je navíc stále veden jako prezident, i když byl už před pěti měsíci ujistěn, že tento údaj bude opraven. Dokonce se zdá, že organizace se rozpadá nejen z právního hlediska, a protože Danielovi na osudu Gentoo stále záleží, přichází s [vlastním řešením](#) ⁽⁵³⁾. Jeho zaměstnavatel [E*TRADE Financial](#) ⁽⁵⁴⁾ souhlasí s tím, aby se vrátil do vedení (ale ne na plný úvazek), kde by dal situaci do pořádku. Dále by chtěl dát Gentoo nějaký směr, neboť podle mnohých ukazatelů distribuce a zájem o ni opadá. To je vidět nejen na [diskuzních fórech](#) ⁽⁵⁵⁾, ale také na [analýze aktivity v CVS repozitářích](#) ⁽⁵⁶⁾.

Novinky o svobodných ovladačích grafických karet

AMD/ATI

AMD uvolnilo další várku dokumentace ⁽⁵⁷⁾ ke svým grafickým kartám, opět bez nutnosti podepisovat [NDA](#) ⁽⁵⁸⁾. První dávka vyšla 12. září 2007 a šlo o více než 900 stran specifikací: referenční příručka registrů pro procesory RV630 a M56. Ta pomohla například ovladačům Avivo a RadeonHD. Důvodem, proč se na další dokumentaci muselo tak dlouho čekat, je nutnost dokumentaci procházet a ověřovat ji nejen z právního hlediska, ale také z hlediska úplnosti informací. Aby se práce uspíšily, AMD přijalo nové lidi. Nová dokumentace se tedy týká M76 a RS690 (i2c a LVTMA části), celkově přibližně 900 stran textu. Během tohoto čtvrtletí by toho mělo přijít ještě více.

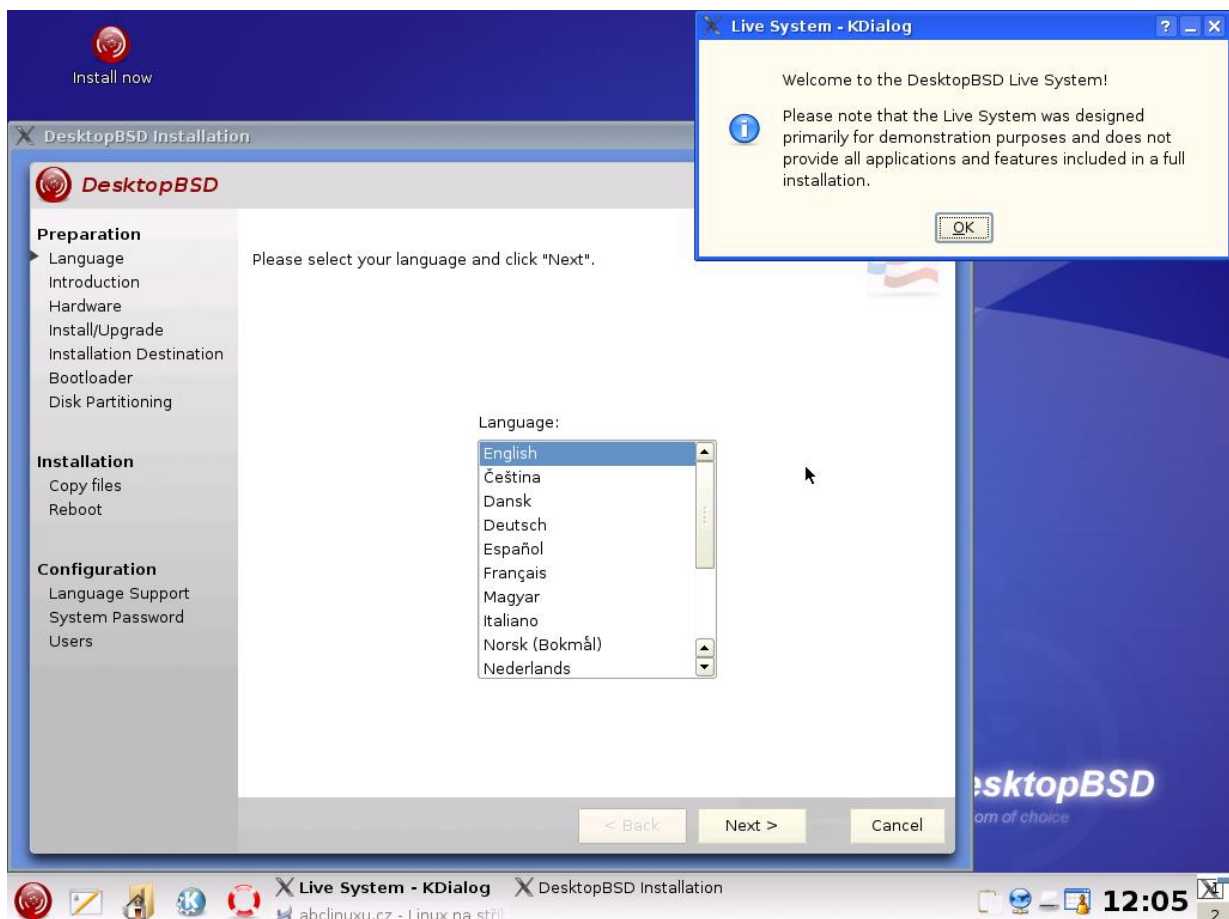
NVIDIA

Tým [Nouveau](#) ⁽⁵⁹⁾ stále bojuje bez dokumentace – poslední novinky najdete v [třicátém třetím vydání Nouveau Companion](#) ⁽⁶⁰⁾. Podařilo se opravit další problémy na platformě PowerPC – jednak šlo o chyby v kódu DRM, pak o špatné zobrazování BIOSu. Na kartách NV04/05 by mělo fungovat ovládání vlastností obrazu Xvideo jako světlost a sytost barev. Dále se začalo pracovat na podpoře RandR 1.2 u starších karet a nyní by mělo chodit i dual link DVI. Ostatní vylepšení a změny se týkaly Xvideo, jeho kvality (konverze YV12 na YUY2), úplnosti (funkčnost texturovaného videa na NV4x) a spolehlivosti (lepší funkčnost na systémech s více obrazovkami).

[Phoronixu se podařilo získat neověřené informace](#) ⁽⁶¹⁾, že NVIDIA chystá odpověď na open source aktivitu AMD, které uvolňuje dokumentaci a spolupracuje s Novellem na ovladači RadeonHD. Zpráva pochází od AIB partnera (výrobce samotných karet, NVIDIA dělá pouze GPU) a není vůbec jisté, zda případně půjde o vylepšení ovladače NV, vydání dokumentace, spolupráci s Nouveau nebo něco úplně jiného. Pokud se ukáže, že je zvěst pravdivá, bude to ukázkou fungování konkurenčního prostředí – ještě v červnu 2006 [bylo postojem firmy](#) ⁽⁶²⁾ projektu Nouveau nepomáhat, ale také ani nebránit ve vývoji.

DesktopBSD 1.6

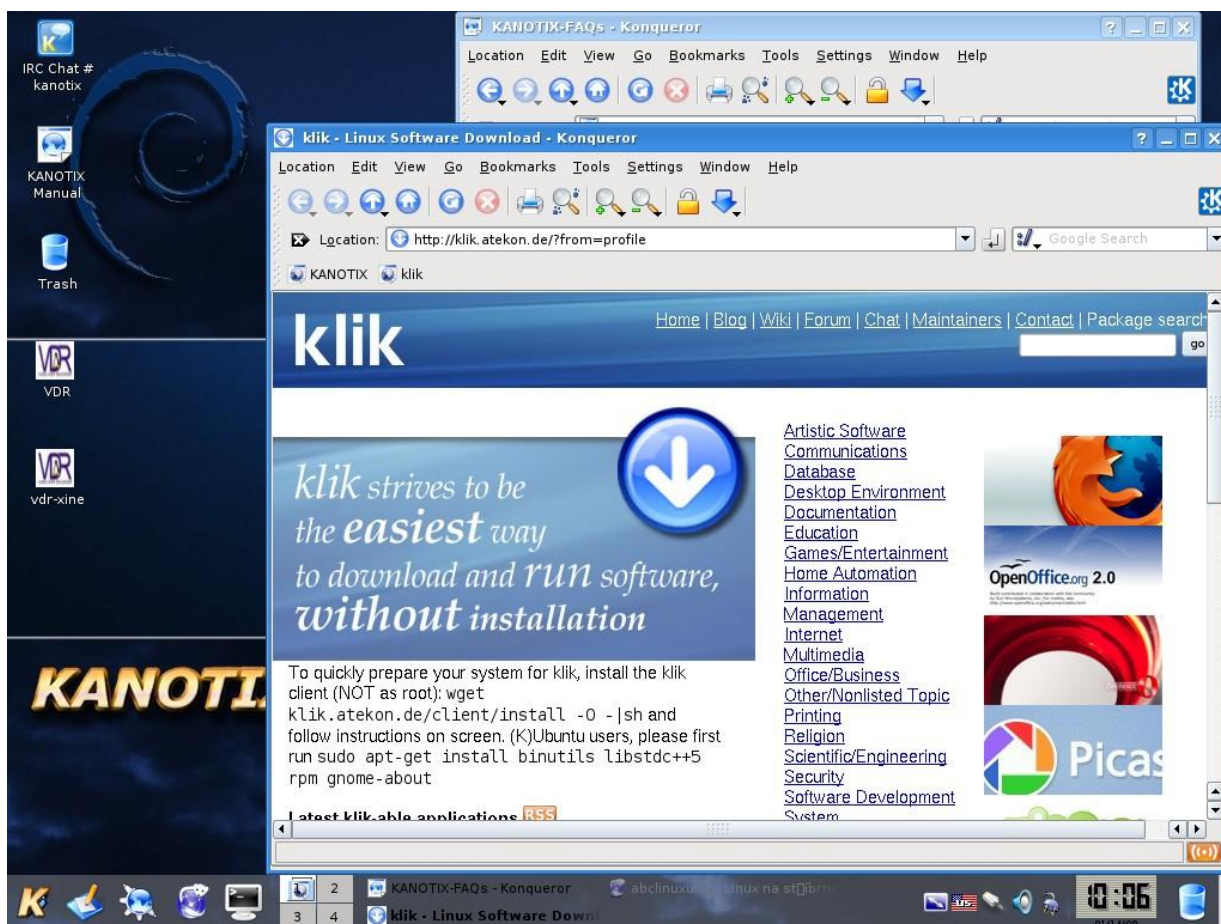
[DesktopBSD 1.6](#) ⁽⁶³⁾ přináší celou řadu různých vylepšení – systém je postaven na základech FreeBSD 6.3 RC2. Vychází pro i386 a se dvěma omezeními pro amd64 (schází proprietární ovladač NVIDIA a není použit GRUB). Instalační CD/DVD nyní mohou pracovat i jako živý systém, přešlo se na grafický server X.org 7.3, na i386 je přítomen ovladač NVIDIA a jsou podporovány víceprocesorové systémy a procesory s více jádry. Vylepšení se dočkal jak instalátor, tak správce balíčků, opraveny byly chyby v různých nástrojích distribuce. Mezi hlavní aplikace patří prostředí [KDE 3.5.8](#), [Amarok 1.4.7](#), [K3b 1.0.4](#), [KTorrent 2.2.2](#), [Pidgin 2.3.1](#) a [Mozilla Firefox 2.0.0.11](#) spolu s [Mozilla Thunderbirdem 2.0.0.9](#). Můžete [stahovat](#) ⁽⁶⁴⁾ DVD pro i386, DVD pro amd64 nebo dvojici CD pro i386 (druhé CD je jazykové).



KANOTIX 2007 RC7

Kanotix⁽⁶⁵⁾ je živá distribuce založená na Debianu. Nabízí aktuální jádro s vlastními patchi a může sloužit při záchraně dat, odstraňování virů z instalací Windows či jen jako bezpečný přenositelný systém. Distribuci je také možné nainstalovat na pevný disk.

Už je sice rok 2008, ale právě nyní vyšla verze 2007 RC7⁽⁶⁶⁾. Vyžaduje alespoň 192 MB RAM (doporučuje se 256), běží na aufs/unionfs a standardně používá ntfs-3g. Prostředím je KDE 3.5.5a s OpenOffice.org 2.0.4, Amarok 1.4.8 a K3b 0.1.4 s nesčetnou řadou dalších programů či skriptů. Systém umožňuje automatickou instalaci ovladačů nvidia a fglrx. K dispozici je NdisWrapper 1.50 pro běh ovladačů z Windows.



Linux Mint 4.0 Beta Fluxbox a Debian

Vyšla zatím vůbec první verze Linux Mint ve variantě⁽⁶⁷⁾ s Fluxboxem – cílem je získat ještě odlehčenější systém, a tak i více potenciálních uživatelů. Na CD je typická sada nástrojů Linux Mintu a známé programy jako Mozilla Firefox, Mozilla Thunderbird, AbiWord, Gnumeric, Exaile! a další.

Druhou věc, kterou vývojáři udělali, je to, že (experimentálně) vzali Debian Testing a přepočtovali na něj svou distribuci – výsledkem⁽⁶⁸⁾ je neinstalovatelné živé CD, které předvádí, jak by distribuce vypadala po přechodu na Debian (z Ubuntu). Podle slov vývojářů by distribuce byla rychlejší, nabízela by novější software, ale mohly by být potíže se stabilitou nebo doladěností některých součástí – další postřehy čtěte v poznámkách k vydání⁽⁶⁹⁾.

Distribuční rada: rychlé převody mezi časovými pásmy v konzoli

Máte se dostavit na pokec s přáteli na IRC, ale čas je specifikován v nějakém cizí časovém pásmu? Konzole vám nabízí jedno snadné řešení pro převod času pomocí příkazu `date`. Dejme tomu, že máme stanoven čas na 15. 1. 2008 20.00 hodin v pacifickém časovém pásmu (PST) a chceme vědět, kolik to bude u nás.

```
$ date -d '2008-1-15 20:00 PST'  
St led 16 05:00:00 CET 2008
```

`date` nám pomůže ještě v jedné situaci. Kamarád bydlí v Číně a budeme na něj chtít zapůsobit rychlým převodem času pro jeho zemi. Pomůže nám hodnota proměnné prostředí `TZ` – pozor, pokud zadáme neexistující zónu, program nevypíše žádnou chybu.

```
$ TZ='Asia/Shanghai' date -d '2008-1-15 20:00 PST'  
St led 16 12:00:00 CST 2008
```

■

Distribuční novinky – 3/2008

Poslední dění okolo Gentoo: Obnova Gentoo Foundation; Návrat buletinu; V čem spočívá krize? Mandriva a Turbolinux se stávají partnery. Neoficiální vývojový plán KDE 4.1. Qt je nyní i pod GNU GPLv3. Linux dostává šanci v ruském školství. Zenwalk Linux 5.0. openSUSE 11.0 Alpha 1. Distribuční rada: dodatečné nastavení některých druhů zrcadel (Gentoo).

Luboš Doležel

Letem světem

Je tu [FreeBSD 6.3](#) ⁽⁷⁰⁾ s celou řadou novinek. Foresight Linux 2.0 Alpha 3 má nový způsob instalace, který urychlí celou proceduru na méně než 10 minut. [Trinity Rescue Kit 3.3 Beta, Build 310](#) podporuje nové síťové karty Intelu a má novou verzi nástroje pro změnu hesla ve Windows. ⁽⁷¹⁾ [Scientific Linux 5.1](#) ⁽⁷²⁾ usnadňuje vytváření odvozených distribucí a přidává balíčky, které se nedostaly do předchozí verze 5.0. [pfSense 1.2 RC4](#) ⁽⁷³⁾ opravuje chyby a stabilizuje systém. [TinyMe Test 7](#) ⁽⁷⁴⁾ má TinyCC (ovládací centrum) a nabízí novější verze programů. [StartCom MultiMedia Edition 5.0.7 RC1](#) ⁽⁷⁵⁾ přechází na jádro s nízkou latencí.

Poslední dění okolo Gentoo

Obnova Gentoo Foundation

[Ztráta zakládací listiny Gentoo Foundation](#) ⁽⁷⁶⁾ rozhýbala diskuze o tom, co se vlastně s Gentoo v poslední době děje, co je příčinou takové situace a jak by se dala napravit. V první řadě se začalo pracovat na nápravě formalit okolo existence Gentoo Foundation – [této práce se ujal Grant Goodyear](#) ⁽⁷⁷⁾. Grant Goodyear je nyní jedním ze dvou zplnomocněnců organizace, kteří nerezignovali, a toto postavení měl dokonce i v době zakládání organizace. Ve svém e-mailu uvádí, že mu není jasné, proč k celé věci došlo, když potřebné materiály minulý rok v červenci úřadům zaslal. Vzhledem k tomu, že šek, který v obálce poštou také putoval, nebyl vyplacen, se nabízí možnost, že se vše někam zatoulalo. Naštěstí se dá vše snadno napravit, stačí poslat úřadům informace, které chtějí, a celý problém zmizí, jako kdyby nikdy neexistoval. Nyní se zvažuje, že organizace přejde pod právní křídla Software Freedom Law Center. SFLC má sice trochu obavy, protože Gentoo je věc podstatně rozsáhlejší než ostatní projekty, kterým pomáhají s právními otázkami, a na straně Gentoo bude zase třeba připravit spoustu dokumentů, které SFLC potřebuje, nicméně můžeme doufat, že se věc podaří.

Návrat buletinu

V uplynulém týdnu jste se dále mohli dozvědět o [zrušení vydání Gentoo 2007.1](#) ⁽⁷⁸⁾. V období od září do prosince 2007 bylo odhaleno rozsáhlé množství chyb v různých programech a spolu se skutečností, že tým Gentoo zodpovědný za vydávání je oslabený, to způsobilo, že se verze 2007.1 nestihla vydat. V podstatě to znamená, že další vydání bude mít označení 2008.0. Dále se začalo řešit, co s Gentoo Weekly Newsletterem (GWN), který už nějakou dobu nevychází. Výsledek je ten, že 19. ledna se vydalo speciální poslední číslo GWN (v němž jsou pouze statistické informace), a dále už bude následovat nový Gentoo Monthly Newsletter.

V čem spočívá krize?

Donnie Berkholz (vývojář Gentoo) vydal na LWN.net článek „[Je Gentoo v krizi?](#)“ ⁽⁷⁹⁾, kde vyjadřuje svůj pohled na celou situaci. Shrnuje všechny události a odkazuje se na blogový zápisek Daniela Robbinse,

podle kterého je problém v komunikaci mezi vývojáři a komunitou okolo Gentoo. Vývojáři Gentoo nadále tvrdě pracují a je vidět pokrok, nicméně se přestali zabývat tím, aby svět informovali o výsledcích. Bude třeba, aby někdo nebo něco začalo tuto komunikaci zajišťovat. Donnie zakončuje článek větou: „*Gentoo neumírá, ale jeho vývojáři to musí světu říct.*“

Mandriva a Turbolinux se stávají partnery

Mandriva a Turbolinux začaly v říjnu 2007 spolupracovat v laboratořích nazvaných Manbo-Labs, ale **věc byla oznámena až nyní** ⁽⁸⁰⁾, když jsou na světě první výsledky. V Manbo-Labs se připravuje společný základ systému pro obě distribuce, který se skládá z GCC, glibc, RPM, jádra, binutils, mkinitrd a udev. Pracuje na něm více než 10 lidí z Francie, Japonska a Brazílie, přičemž díky tomuto sdílení prostředků budou firmy moci investovat více do jiných oblastí (vývoj technologií a zajišťování kvality produktu). Zároveň se zlepší hardwarová kompatibilita a naváží se lepší vztahy s nezávislými vývojáři hardwaru a softwaru. Výsledky budou uvolněny pod **GNU GPL** ⁽⁸¹⁾ a ostatním distribucím postaveným na RPM je nabízena spolupráce.

Neoficiální vývojový plán KDE 4.1

Ačkoliv ještě nebyl časový plán pro KDE 4.1 oficiálně připraven, **v zákulisí probíhají diskuze** ⁽⁸²⁾ o podobě vývojového cyklu. Většinu vývojářů by se líbil pevný 6měsíční cyklus, v rámci kterého by se do dalšího vydání dostalo vždy to, co bylo připraveno v době zmrazení. Předběžný plán vypadá tak, že KDE 4.0.1 vyjde ještě v lednu a verze KDE 4.1.0, na kterou mnoho lidí čeká, by se mohla objevit v červenci. Červenec vypadá jako vhodný čas pro některé distribuce, které staví hlavně na KDE. Druhou diskutovanou věcí jsou plány na obsah KDE 4.1. Jedním z hlavních cílů je pravděpodobně vylepšená Plasma se stabilním API, což do značné míry závisí na Qt 4.4. Zařazení do KDE 4.1 plánují i další projekty jako KDE-PIM, **Decibel** ⁽⁸³⁾ nebo **KDevelop**.

Qt je nyní i pod GNU GPLv3

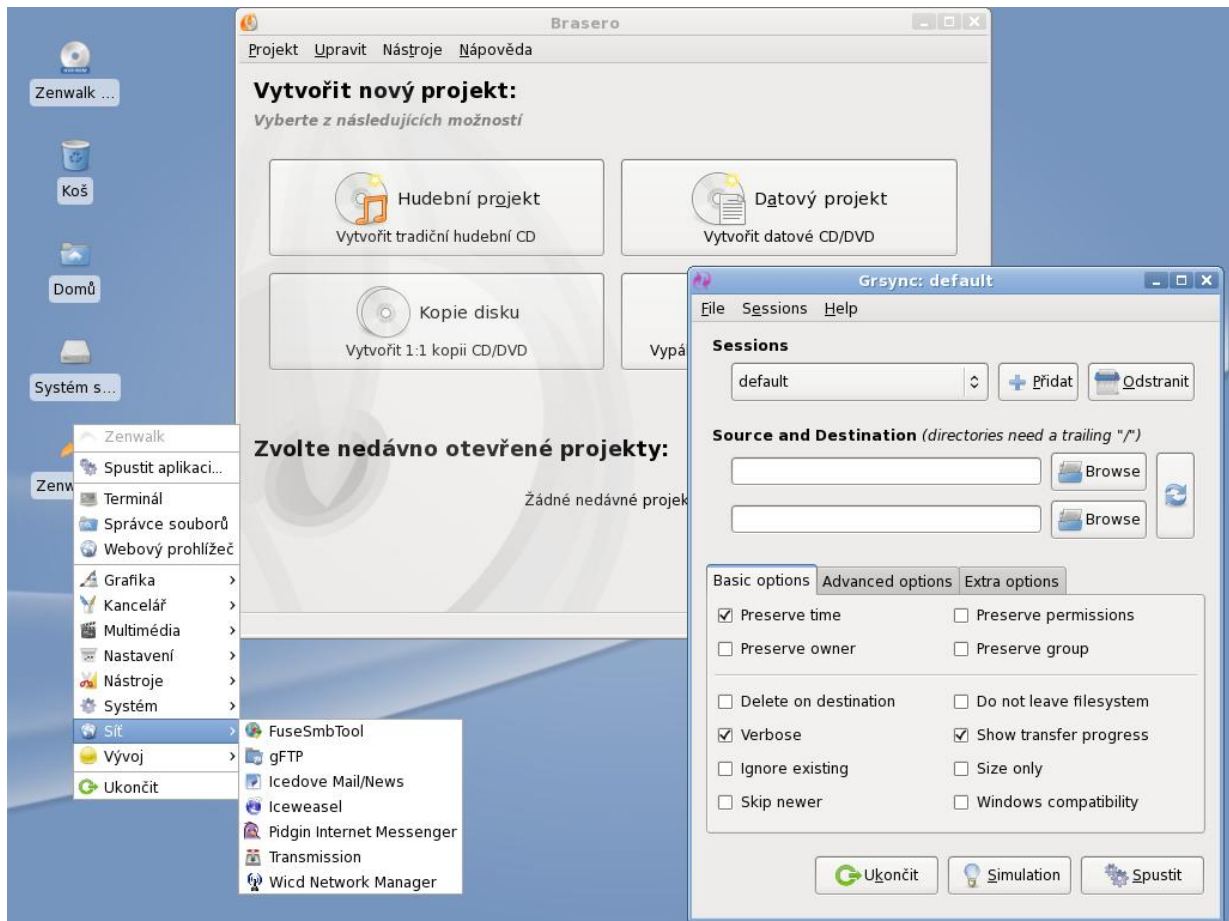
Trolltech se rozhodl **licencovat Qt 4 pod GNU GPLv3** ⁽⁸⁴⁾ (zároveň s původní GPLv2). K tomuto rozhodnutí se došlo po konzultaci s KDE e.V. a Free Software Foundation a vliv na to měl i tlak komunity, která se Trolltechu ptala *kdy to konečně bude* už od doby, kdy byla GPLv3 dokončeno. Výsledek je ten, že pod GPLv3 jsou snapshoty Qt 4.3.4 a 4.4.0 a verze Qt 3.3.8b. V případě Qt 3.3.8b jde čistě jen o doplnění licence do všech edicí Qt kromě edice pro Windows, u Qt 4 snapshoty předcházejí verzím Qt 4.3.4 a 4.4.0 beta 1. Další změnou je hladší přechod na budoucí možné verze GNU GPL – v licenční hlavičce se nyní uvádí, že jakékoliv další verze licence můžete použít, pokud to veřejně schválil Trolltech a KDE Free Qt Foundation. KDE Free Qt Foundation je organizace, která má právo vydat poslední verzi Qt pod licencí BSD, pokud by Trolltech někdy přestal uvolňovat open source verze Qt.

Linux dostává šanci v ruském školství

V Rusku se stupňují snahy o snížení nákladů na IT (i když je tam značný podíl softwaru provozován nelicencovaně) a omezení závislosti na zahraničních softwarových firmách. Díky tomu se zde dostává ke slovu Linux spolu s místními distribucemi a výsledkem je také podpoření místních firem a programátorů. Nyní se rozšířila informace, že projekt s cílem dostat Linux a OSS do škol **završil první fázi** ⁽⁸⁵⁾. Projekt vede skupina firem Armada Group a druhou fází je pilotní nasazení ve vybraných oblastech: Perm, Tomsk a Tatarstán. Získané zkušenosti a komentáře uživatelů pak usnadní rozšíření projektu na další oblasti. Předností open source softwaru je pro školy i skutečnost, že systém dobře běží na starých počítačích.

Zenwalk Linux 5.0

Zenwalk Linux⁽⁸⁶⁾ se zaměřuje na jednoduchost a optimální výkon – i verze 5.0⁽⁸⁷⁾ by měla být v tomto duchu. Protože se jedná o „velkou“ verzi, přichází několik zásadních inovací. Zaprvé je nasazen Freedesktop ♦HAL⁽⁸⁸⁾, který nahrazuje dřívější vlastní systém Zenwalku pro připojování hardwaru za běhu. První změnou, které si díky tomuto budete moci všimnout, jsou ikony všech připojených vyměnitelných zařízení na ploše. Programy s podporou HALu navíc umí reagovat na příslušné události (např. vykonáním operací dle přání uživatele). Další velkou změnou je přechod na X.org 7.3, což přináší nové ovladače a funkce. Na závěr stojí za zmínku prostředí Xfce 4.4.2 a aktualizované programy Iceweasel a Icedove (spolu s celou řadou dalších).



openSUSE 11.0 Alpha 1

A je tu openSUSE 11.0 Alpha 1⁽⁸⁹⁾. Od předchozí alphy z prosince bylo aktualizováno 1026 balíčků, další vývojová verze vyjde začátkem února. Hlavních změn je hned několik: byl integrován Sat Solver (knihovna, která řeší závislosti balíčků pro libzypp), instalátor byl přeportován na Qt 4 (a má zároveň nový kabátek), je přítomno KDE 4 (zda bude hlavním prostředím v konečné verzi, se ještě uvidí) a dále najdete zvukový server PulseAudio, NetworkManager 0.7, glibc 2.7, Perl 5.10 nebo CUPS 1.3.5. Jedná se o vývojovou verzi, takže má v sobě systém mnoho nedostatků a chyb – při zkušební instalaci z KDE CD se mi ani správně nenakonfiguroval zavaděč.



Distribuční rada: dodatečné nastavení některých druhů zrcadel (Gentoo)

Soubory stahované při instalaci balíčků mohou mít různá umístění – většinu najdete na zrcadlech Gentoo, ostatní na SourceForge a jiných serverech. V případě těch ostatních se obvykle jedná o zrcadlo určené v podobě `mirror://oznaceni/cesta`. U nich se emerge podívá do `/usr/portage/profiles/thirdpartymirrors` a postupně začne zkoušet uvedená zrcadla s tímto označením.

Pro nás může být vhodné nastavit preferenci pro zrcadla, která jsou geograficky bližší. To můžeme učinit v souboru `/etc/portage/mirrors`. Pokud bychom chtěli upřednostnit vybraná zrcadla pro data ze SourceForge, GNU a Debianu, nastavili bychom:

```
sourceforge http://dfn.dl.sourceforge.net/sourceforge
gnu http://ftp.sh.cvut.cz/MIRRORS/gnu/pub/gnu/
debian http://debian.sh.cvut.cz/debian
```

■

Distribuční novinky – 4/2008

Lotus Notes 8.5 bude mít plnou podporu pro Ubuntu. Hewlett-Packard uvolnil nástroj pro správu OSS. Poslední novinky ze světa svobodných grafických ovladačů. Plány na Gentoo 2008.0. Německé Ministerstvo práce nasadilo Linux. Parsix GNU/Linux 1.0r0. GoblinX 2.6 „Micro“. Mandriva Linux 2008.1 Beta 1. Distribuční rada: přednačítání souborů s readahead-list.

Luboš Doležel

Letem světem

Ubuntu 6.06.2⁽⁹⁰⁾ je aktualizovaná verze Ubuntu s dlouhou podporou (LTS). SystemRescueCd 0.4.3⁽⁹¹⁾ aktualizuje jádro, přidává ovladače ATI a NVIDIA, pbzip2 (paralelní komprese bzip2), lzma-utils (komprese LZMA), bootovací volbu dodebug a opravuje všelijaké drobnosti. Elive 1.5.0⁽⁹²⁾ je vývojové vydání distribuce, ve kterém je kompletně přepsán bootovací systém. Dreamlinux 3.0 Beta 2⁽⁹³⁾ vylepšuje vlastní nástroje, nabízí správce oken z Xfce i GNOME, odstraňuje proprietární kodeky plus další změny. Linux Mint 4.0 RC2 „KDE“⁽⁹⁴⁾ se více přibližuje hlavní edici distribuce.

Lotus Notes 8.5 bude mít plnou podporu pro Ubuntu

IBM během konference Lotusphere 2008 oznámilo, že prostřednictvím Open Collaboration Client nabídne plnou podporu pro Ubuntu Linux⁽⁹⁵⁾ u produktů Lotus Notes 8.5 a Lotus Symphony. Je to prý odpověď na zájem zákazníků – Lotus Notes 8.0.1 má podporu Ubuntu jen omezenou a lidé chtějí více. A nejde o nic malého: některé požadavky na zavedení podpory přišly od velkých firem. Podle slov marketingového ředitele IBM Lotus šlo o organizace, kde na Ubuntu Linux přechází až 100 000 uživatelů. Započalo to malými firmami a došlo to až k firmám velkým, kde na Linux přecházelo nejdříve nějakých 500 uživatelů a později to dosáhlo 2000 uživatelů a pokračovalo to dál až k současným číslům. Po dobu posledních šest let jsme mohli slyšet, že Linux je konečně připraven pro velká firemní nasazení, ale IBM věří, že tento rok to tu doopravdy je.

Hewlett-Packard uvolnil nástroj pro správu OSS

Hewlett-Packard spustil projekt⁽⁹⁶⁾ FOSSology⁽⁹⁷⁾ – ten slouží pro monitorování použití svobodného softwaru v IT prostředí. Web projektu popisuje software takto: *Potřebovali jsme nástroj, který by rychle a přesně popsál, jak je konkrétní open source projekt licencován. Namísto shromažďování informací ze stránek a z dokumentace projektu tento nástroj zanalyzuje zdrojový kód a chytře oznámí, jaké licence jsou používány.* FOSSology je dostupné pod GNU GPLv2 a mělo by fungovat na většině distribucích. Nejedná se o něco, co by mělo nahradit kompletní právní analýzu, ale poskytuje to dobrou představu. V budoucnu by se měly objevit ještě další funkce jako informace o opravách chyb a znovupoužití kódu. HP dále za spolupráce s Linux Foundation rozběhl diskuzní web FOSSBazaar⁽⁹⁸⁾. FOSSBazaar bude sloužit pro vzdělávání, protože se zde lidé mohou dozvědět o open source, licencích, bezpečnosti, intelektuálním vlastnictví atd. Poslední novinkou jsou konzultační služby „Open Source Health Check“ – HP nabídne možnost vytvoření posudku kódu a pomůže při migraci na open source software.

Poslední novinky ze světa svobodných grafických ovladačů

Pro uživatele NVIDIA se zatím nepodařilo zjistit žádné informace o údajných plánovaných open source aktivitách⁽⁹⁹⁾ firmy NVIDIA, nicméně vyšly alespoň nové ovladače `nv`. Jde konkrétně o xf86-video-nv 2.1.7

pro X.Org 7.2 a novější a o xf86-video-nv 2.0.3 pro starší řady X.Org. Je přidána podpora pro GeForce 8800GT a Quadro FX 3700 plus několik dalších oprav a vylepšení. Podrobnosti najdete v [oznámení](#) ⁽¹⁰⁰⁾.

AMD RS690 je integrovaný grafický procesor, který můžeme najít na některých základních deskách – je oblíbený především v počítačích typu [HTPC](#) ⁽¹⁰¹⁾. RS690 je model, jenž kombinuje nastavování režimů z procesorů řady R500 spolu s odlehčeným 3D jádrem z řady R400. Nyní se dočkal počáteční open source podpory, která je na úrovni řady RS400 – funguje glxgears a několik dalších OpenGL aplikací, ale zatím nestačí ani na [Compiz](#). Pokud chcete ovladač vyzkoušet, musíte si zkompileovat hned několik věcí z [git](#) ⁽¹⁰²⁾ repozitáře – [více najdete na Phoronixu](#) ⁽¹⁰³⁾.

Plány na Gentoo 2008.0

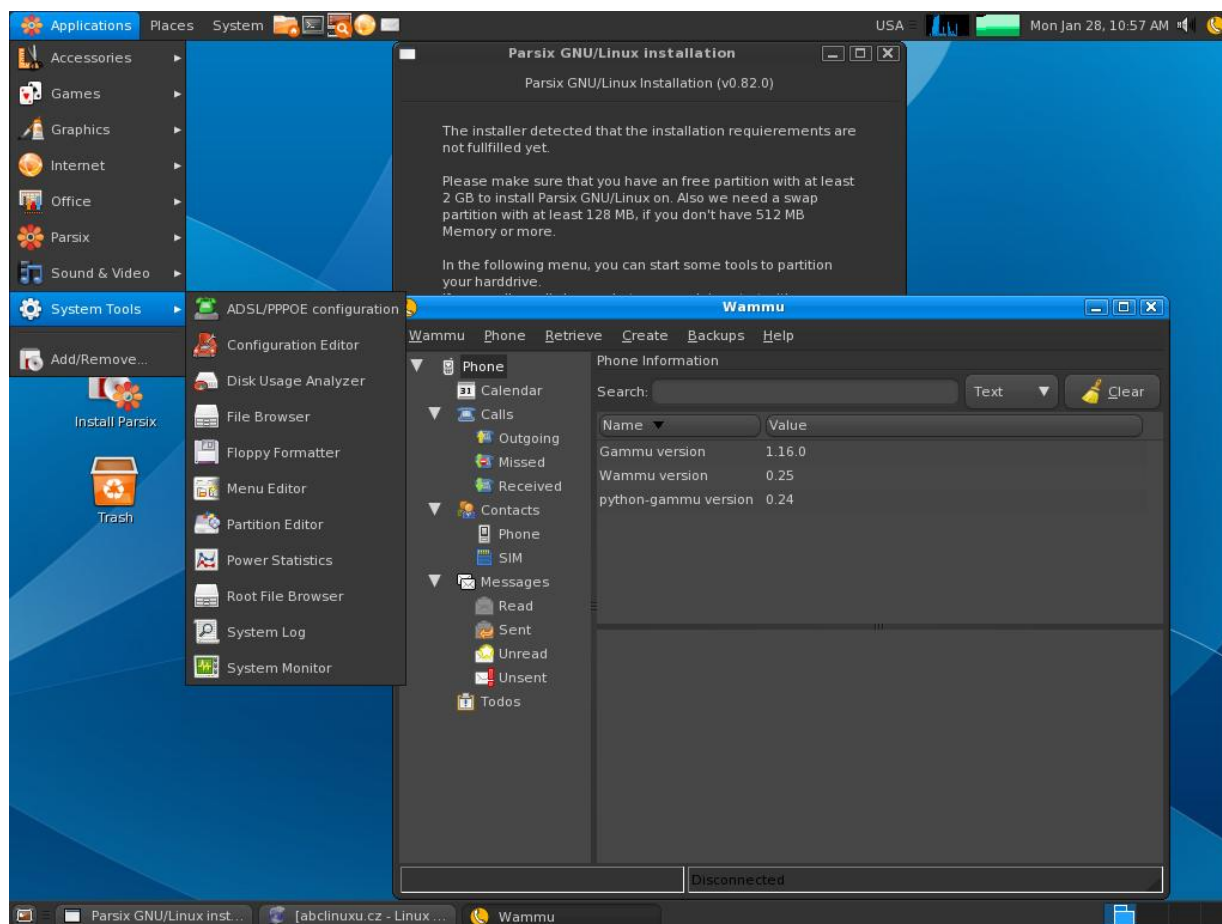
Vypadá to, že poprask kolem úpadku Gentoo zafungoval jako účinný stimul. Domovská stránka Gentoo ožila zprávami, později především těmi, které se týkají příštího vydání 2008.0. Hned v pondělí vyšel [slibovaný Gentoo Monthly Newsletter](#) ⁽¹⁰⁴⁾ – tam se mj. dozvíte, že distribuce má nyní 275 aktivních vývojářů a že za dobu existence na ní pracovalo vývojářů celkem 628. Později byl zveřejněn [vývojový plán Gentoo 2008.0](#) ⁽¹⁰⁵⁾, ve kterém je 17. březen určen jako den vydání. Aby se do celého procesu více zapojila komunita, hned další den [byli uživatelé přizváni](#) ⁽¹⁰⁶⁾ do mailing listu gentoo-releng, kde mohou navrhnout věci, které by v ISO obrazech Gentoo rádi viděli jinak, než jak tomu bylo v předchozích verzích.

Německé Ministerstvo práce nasadilo Linux

Německé Ministerstvo práce udělalo významný krok směrem k používání open source. Všech svých 13 000 veřejných terminálů [převedli z Windows NT na Linux](#) ⁽¹⁰⁷⁾. Vybranou distribucí je openSUSE 10.1 a [Mozilla Firefox](#) funguje jako náhrada Internet Exploreru. Stanice bootují přes síť díky PXE, což znamená, že stačí udržovat pouze centrální repozitáře na serveru. Přejít zabral devět měsíců a nevyžadoval spolupráci s externími firmami. Důvodem pro změnu tímto směrem je nejen cena a bezpečnost (s čímž se pojí extrémně nízké riziko nakažení systému viry), ale také snadná a automatizovaná údržba spojená s flexibilitou konfigurace.

Parsix GNU/Linux 1.0r0

Parsix GNU/Linux je živá (instalovatelná) distribuce založená na KANOTIXU a Debianu. Výchozím prostředím je [GNOME](#), přičemž zvláštností je podpora perské klávesnice a další věci související s tímto jazykem (fonty, slovník, ...). [Parsix GNU/Linux 1.0r0](#) ⁽¹⁰⁸⁾ nese kódové označení Ramon. Běží na jádře 2.6.23.14 (s [CK](#) ⁽¹⁰⁹⁾ a TuxOnIce), X.Org 7.2.0 a prostředí GNOME 2.20.3. Software pochází z repozitářů Debianu Testing z 24. ledna 2008. Bylo vyřešeno několik chyb, [Rhythmbox](#) nahradilo [Exaile!](#) a na základě žádostí uživatelů se do distribuce vrátilo několik programů (iptraf, rdesktop, [Wammu](#) a vncviewer). Přibyl konfigurační nástroj X.Org (portovaný z Ubuntu) a kombinace SquashFS s kompresí LZMA znamená menší velikost ISO obrazu spolu s rychlejší dekompresí.

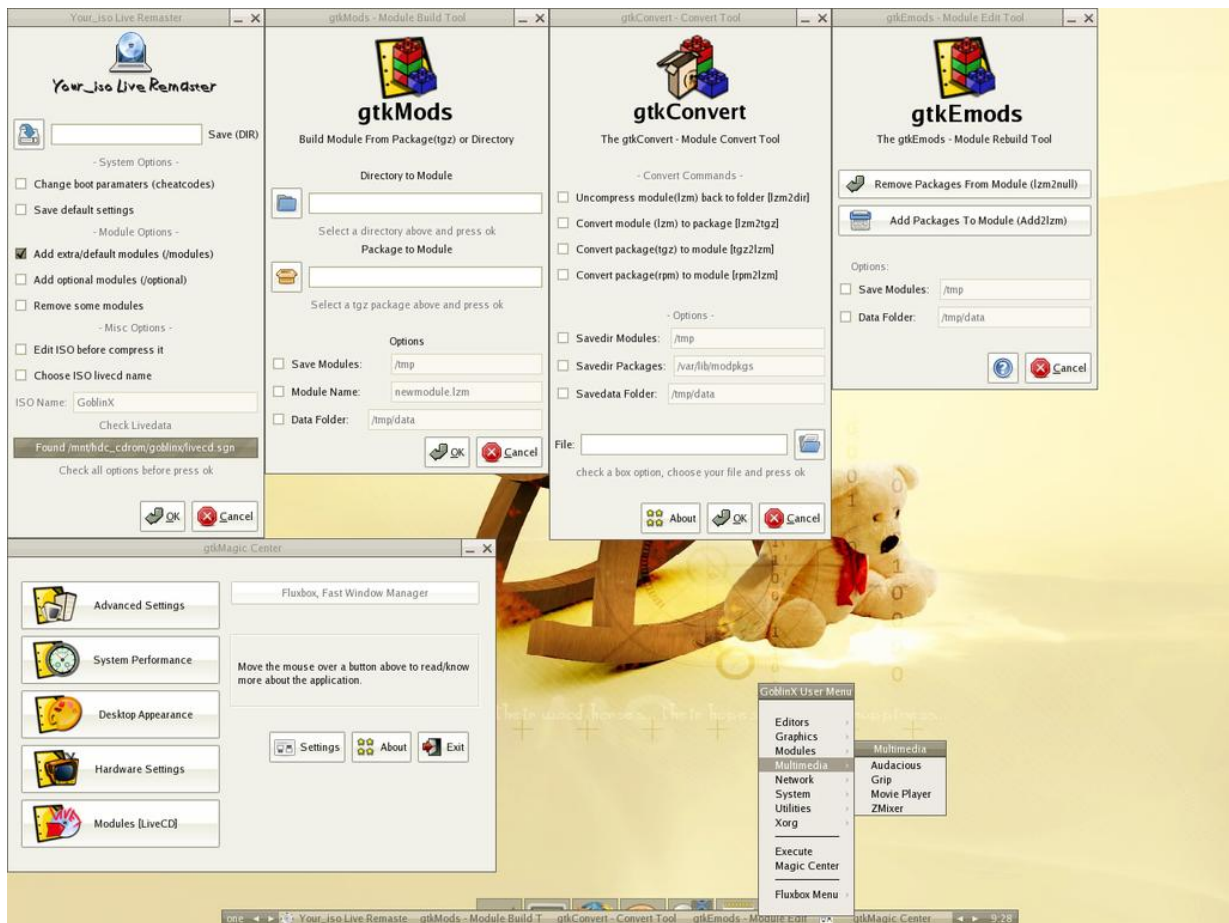


GoblinX 2.6 Micro

[GoblinX Micro Edition](#) ⁽¹¹⁰⁾ je drobná distribuce postavená na Slackwaru. Nabízí [Fluxbox](#) a veškeré aplikace jsou stavěné na [GTK+ 1/2](#). Distribuci si můžete jednoduše rozšířit pomocí přichystaných nástrojů, navíc je k dispozici repozitář s tisíci programů. Systém využívá transparentní dekomprese souborů. V [poslední verzi](#) ⁽¹¹¹⁾ přibyl firewall ze Slaxu a v bootovacím menu jsou nové položky. Obraz CD má lehce pod 100 MB.

Mandriva Linux 2008.1 Beta 1

Vyšel [Mandriva Linux 2008.1 Beta 1](#) ⁽¹¹²⁾ s celou řadou zajímavých novinek. První změnou je přidání systému pro poskytování informací o balíčcích založeného na XML. I nadále jsou základní informace o balíčcích v repozitářích přítomné v souboru [synthesis.hdlist](#), ale soubor [full.hdlist](#) nahradí soustava XML souborů, které budou stahovány podle potřeby. Dále byla rozšířena podpora multimediálních klávesnic – jednoduché funkce jako změna hlasitosti by měly fungovat rovnou, bez další konfigurace. Přítomny jsou aktuální proprietární grafické ovladače: [NVIDIA 169.09](#) ⁽¹¹³⁾ a [ATI fg1rx 8.45.2 \(Catalyst 8.1\)](#). V repozitáři [contrib](#) se nalézá konečná verze [KDE 4.0.0](#) ⁽¹¹⁴⁾. [Nautilus](#) je vybaven podporou pro nový virtuální souborový systém [GVFS](#), který nahrazuje starší [gnome-vfs](#). Jako jádro slouží [Linux 2.6.24](#) ⁽¹¹⁵⁾ RC8, k němuž se pojí [ALSA 1.0.16 RC1](#) ⁽¹¹⁶⁾.



Distribuční rada: přednačítání souborů s readahead-list

`readahead-list` je program, který umožňuje vytvořit seznam souborů, ke kterým se přistupovalo, a získat tak kandidáty vhodné pro přednačítání. To může na systémech s dostatkem RAM urychlit spouštění.

Vytvoříme si adresář, kam informace budeme ukládat, a rovnou nějaké zachytíme:

```
mkdir /var/lib/readahead
cd /var/lib/readahead
readahead-watch -o nejaky-nazev
```

Jakmile se poslední příkaz provede, necháme spustit operaci, u které chceme používání souborů vysledovat (např. se přihlásíme do grafického prostředí). Na závěr program ukončíme:

```
killall readahead-watch
```

Výčet souborů prohlédneme, případně odstraníme některé nechtěné věci. Nyní zařídíme, aby se soubory přednačítaly – přednačítání můžeme například umístit do startovacího skriptu. Soubory se tak načtou dříve, než se stihneme přihlásit do systému. Hlavní kód takového skriptu by vypadal takto:

```
for soubor in /var/lib/readahead/*; do
    readahead-list $soubor
done
```

Gentoo ebuild – 6 (závislosti, sloty, restrikce)

Jak řešit závislosti jednotlivých ebuildů, včetně závislostí ob-
rácených a těch, které jsou určeny USE flagy. Jak umožnit
instalaci více verzí stejného ebuildu.

David Watzke

Závislosti

Závislosti jsou softwarové požadavky ebuildu. Existuje několik druhů závislostí.

Závislosti potřebné k sestavení

Do proměnné `DEPEND` patří všechny závislosti, které jsou třeba pro rozbalení, záplatování, sestavení a instalaci balíčku (s výjimkou implicitních, vizte níže).

Závislosti potřebné ke spuštění

Proměnná `RDEPEND` by měla obsahovat závislosti, které jsou potřeba pro běh programu. Pokud tuto proměnnou nezádáte, nastaví se automaticky na `DEPEND`. Při instalaci z binárního balíčku (vytvořeného např. pomocí `qpkg` nebo `quickpkg`) se kontroluje pouze `RDEPEND`, takže je třeba tam přidat i závislosti, které už jsou i v `DEPEND`.

Poinstalační závislosti

Do proměnné `PDEPEND` patří závislosti, které je třeba nainstalovat až po instalaci cílového balíčku. To je užitečné například pro zásuvné moduly instalovaného balíčku. Zpravidla je vhodné se `PDEPEND` vyhýbat a raději používat `RDEPEND`, vyjma případů, kde by to způsobilo kruhové závislosti.

Implicitní systémové závislosti

Všechny balíčky mají implicitní závislost na systémových věcech (vizte soubory `packages*` v adresáři a podadresáři `$PORTDIR/profiles`), jako je třeba `gcc` a `libc`. Tyto balíčky mezi závislosti nedávejte, pokud nepotřebujete konkrétní verzi nebo balíček (např. `glibc` místo `uclibc`).

Syntaxe proměnných `*DEPEND`

Vždy je třeba uvést alespoň kategorii a název balíčku, například `games-fps/quake3`. Přidáním patřičného operátoru před kategorií balíčku lze upřesnit požadovanou verzi.

	Popis
<	Nižší než daná verze
>	Vyšší než daná verze
<=	Nižší nebo rovno dané verzi
>=	Vyšší nebo rovno dané verzi
~	Libovolná revize dané verze

Může to vypadat například takto:

```
DEPEND=">=app-i18n/enca-1.9
        <dev-lang/perl-5.10.0"
```

Při použití (samotného) operátoru `=` je možné zobecnit požadavek na verzi. Následující příklad se spokojí s jakoukoliv verzí knihovny Qt třetí řady.

```
=x11-libs/qt-3*
```

Dále lze také závislosti negovat, čili určit balíčky (třeba i jejich verze), které v systému být nesmí, aby se mohl nainstalovat ten cílový. Stačí přidat vykřičník před název kategorie.

```
!app-text/pdftohtml
!>=app-accessibility/speech-tools-1.2.95_beta
!<app-text/texlive-2007
```

Tolik tedy k určování verzí. Dále lze dát ebuildu na výběr z několika závislostí, které například poskytují stejnou funkčnost. Kupříkladu potřebuje-li váš ebuild **Konqueror** a vy víte, že jej poskytuje monolitický balíček `kde-base/kdebase` i rozdělený `kde-base/konqueror`, můžete napsat toto:

```
|| ( kde-base/konqueror kde-base/kdebase )
```

Kulaté závorky určují blok, ke kterému se vztahuje výraz před nimi. V tomto případě to bylo `||`, což určuje, že je vyžadován jen jeden balíček z bloku. Pro některé podobné případy, jako je vyžadování některé z implementací JRE, JDK, GhostScript atp., může být lepší záviset na patřičném virtuálním balíčku (`$PORTDIR/virtual`), který se o to postará za vás.

Tímto se dostáváme k závislostem podmíněným USE flagy. Zápis je podobný předchozí ukázce, jen dáme před závorku místo výrazu `||` USE flag a za něj otazník. Pokud naopak chcete, aby se některé závislosti nainstalovaly, jen pokud je daný USE flag vypnutý, přidejte před něj ještě vykřičník.

```
gif? ( media-libs/giflib )
# pokud nejsme na architektuře amd64* a máme povolený
# USE flag 'cdf', potřebujeme sci-libs/cdf
!amd64? ( cdf? ( sci-libs/cdf )
```

* USE flag architektury systému je nastaven automaticky.

Všechno vyjmenované lze různě kombinovat. Následující řádek si v případě, že je zapnutý USE flag `rar`, vyžádá balíček `rar` nebo `unrar` (vybrat podle preference si můžete tak, že jeden z balíčků nainstalujete ručně).

```
rar? ( || ( app-arch/rar app-arch/unrar ) )
```

V ebuildu balíčku `poppler-bindings` najdeme tyto závislosti:

```
RDEPEND="~app-text/poppler-${PV}
        cairo? ( >=x11-libs/cairo-1.4 )
        gtk? (
            >=x11-libs/gtk+-2.6
            >=gnome-base/libglade-2
        )
        qt3? ( =x11-libs/qt-3* )
        qt4? ( =x11-libs/qt-4* )"
DEPEND="${RDEPEND}
        dev-util/pkgconfig
        >=sys-devel/automake-1.9.6"
```


Balíček `poppler-bindings` potřebuje k běhu `poppler` stejné verze, což se dá obecně vyřešit použitím `$PV`, jak si lze všimnout na prvním řádku. Není třeba znovu vypisovat stejné závislosti, proto klidně použijte již nastavenou proměnnou v té následující (v příkladu vidíme použití `RDEPEND` v `DEPEND`).

Sloty

Slotování přijde vhod ve chvíli, kdy chceme umožnit instalaci více verzí jednoho ebuildu, například KDE verze 3.x a 4.x, knihovnu GTK+ verze 1.2 a 2.x, Qt verze 3.3 a 4.x či zdrojové kódy jádra. Většina balíčků slotování nepotřebuje a v jejich ebuildech se nastavuje `SLOT="0"`. To není totéž jako uvedení prázdného slotu, což vypne slotování úplně a není doporučováno. Portage umožňuje mít nainstalovanou jednu verzi od každého slotu.

Restrikce

V [prvním díle](#) ⁽¹¹⁷⁾ jsem slíbil, že si později povíme více o proměnné `RESTRICT` a jejích možnostech. Jednotlivé vlastnosti v proměnné oddělujte mezerou. Následuje jejich vysvětlení.

Pro ovlivnění automatického nastavení zrcadel existují restrikce `mirror`, `primaryuri` a `fetch`; `mirror` zakáže hledání souboru na oficiálních Gentoo zrcadlech, `primaryuri` změní pořadí hledání tak, že se soubor nejdříve hledá na zrcadlech zadaných v proměnné `SRC_URI` a teprve poté na těch oficiálních. A `fetch` se používá tehdy, když licence neumožňuje stažení potřebných souborů přímo a software je třeba nejdříve ručně uložit do `$DISTDIR`. V takovém případě je třeba do proměnné `SRC_URI` vypsát pouze názvy souborů a případně definovat funkci `pkg_nofetch()`, která uživateli vysvětlí, odkud (a který) balíček stáhnout, kam jej uložit a proč. Výchozí definice vypadá následovně:

```
pkg_nofetch()
{
    [ -z "${SRC_URI}" ] && return

    echo "!!! The following are listed in SRC_URI for ${PN}:"
    local x
    for x in $(echo ${SRC_URI}); do
        echo "!!!  ${x}"
    done
    return
}
```

Restrikce `test` zakáže spuštění `src_test()`, vizte [druhý díl](#) ⁽¹¹⁸⁾ tohoto seriálu. Chcete-li zabránit automatickému spouštění `strip` na binárky, nastavte stejnojmennou restrikci. To je vhodné například při instalaci předkompilovaných balíčků, kde by Portage jinak vypisovala varování, protože to řeší po svém. Restrikce `userpriv` vynucuje pro celý proces kompilace rootovská práva. To je výchozí chování Portage a změna nastane jen u uživatelů, kteří si tuto vlastnost explicitně povolili pomocí proměnné `FEATURES`. Balíčku, který neobsahuje žádné binárky, je vhodné nastavit restrikci `binchecks`, která zakáže QA kontroly. Toto není určeno k obcházení QA kontrol, o tom se zmíním v příštím díle. Distribuci binárních balíčků lze zakázat pomocí restrikce `bindist`. Kupříkladu binární balíčky Firefoxu smíte z licenčního hlediska šířit jen pokud aktivujete USE flag `bindist` nebo `iceweasel`. Program `quickpkg` vás na toto upozorní, alternativa `qpkg` (z balíčku `portage-utils`) zatím ne. Restrikce `installsources` zakáže stejnojmennou vlastnost, i když je uživatelem explicitně povolena pomocí proměnné `FEATURES`. Jde o vlastnost, která nainstaluje zdrojové kódy do `/usr/src/debug/$CATEGORY/$PF`, pokud je nainstalovaný `debugedit` a v `CFLAGS` nastavený nějaký debugovací přepínač, např. `-ggdb`.

Gentoo Linux – Portage Overlay

Overlay se říká neoficiálnímu stromu s ebuildy pro balíčkovací systémy v Gentoo. V tomto článku se dozvíte, jak si vytvořit a spravovat vlastní Portage Overlay.

David Watzke

Poznámka: Pokud nerozumíte některým věcem souvisejícím s ebuildy, přečtěte si obsah [seriálu o ebuildch](#) ⁽¹¹⁹⁾.

K čemu to je

Běžně se doporučuje nainstalovat software mimo balíčkovací systém. To samozřejmě platí i na Gentoo. Když narazíte na software, který si chcete nainstalovat a není pro něj ebuild v oficiálním stromu, jednou z možností je napsat si ebuild vlastní (vizte seriál [Gentoo ebuild](#) ⁽¹²⁰⁾). Tyto ebuildy ovšem nemůžete dát do hlavního stromu, protože by byly během další synchronizace vymazány. Proto existují overlaye.

Jak začít

Pro začátek si pro overlay vytvořte nějaký adresář, řekněme třeba `/usr/local/portage` (dále jen overlay).

```
mkdir -p /usr/local/portage
```

Poté editujte konfigurační soubor `/etc/make.conf`. Do proměnné `PORTDIR_OVERLAY` je třeba zadat zvolený adresář. V případě, že máte více overlayí, odděluje cesty mezerou – v cestě tedy nesmí být žádné mezery.

```
PORTDIR_OVERLAY="/usr/local/portage"
```

Nyní si v overlayi vytvořte adresář `profiles` a do něj přidejte soubor `repo_name`, který bude obsahovat název vaší nové overlaye. Název smí obsahovat pouze malá a velká písmena bez háčeků a čárek, číslice, pomlčky a podtržítka.

```
cd /usr/local/portage
mkdir profiles
echo overlay-watzke-cz > profiles/repo_name
```

Struktura a kategorie

Struktura overlaye je stejná jako u hlavního stromu, s tím rozdílem, že v overlayi *musí* být jen opravdu nezbytné soubory a adresáře (správně umístěné ebuildy):

```
overlay
|-- kategorie
    '-- program
        |-- Manifest
        |-- files
        |   '-- program-1.2-link.patch
        '-- program-1.2-r5.ebuild
```

Kategorii pro program, který chcete přidat, si vyberte z `profiles/categories` v hlavním stromu Portage (tj. obvykle `/usr/portage/`) nebo si můžete vymyslet vlastní a dopsat ji do `/etc/portage/categories`, ale tomu bych se spíše vyhýbal (dost zbytečná komplikace).

Eclass

Můžete si napsat vlastní eclass, což je shellový skript, který nastavuje pomocné proměnné a obsahuje často používané funkce pro danou činnost. Eclass patří do adresáře `eclass` relativně k overlayi. Lze je poté načíst v ebuildu pomocí funkce `inherit` (tj. něco jako shellový `source`, vizte článek [Gentoo ebuild – 2 \(funkce, eclass a příkazy\)](#) ⁽¹²¹⁾).

Eclass v overlayi je ovšem třeba používat opatrně, protože Portage nejenže neaktualizuje cache, když se ebuild změní, ale neaktualizuje ji ani když se změní eclass z hlavního stromu, kterou vaše eclass načítá. Při práci s eclass v overlayi můžete také narazit na upozornění „illegal inherit“, přestože jste neudělali chybu (a nenačítali eclass podmíněně). Pro jistotu můžete po úpravě eclass ručně spustit `touch` na všechny relevantní soubory overlaye.

Patche

Pokud chcete v ebuildu patchovat, tak vězte, že patche patří do adresáře `kategorie/program/files`. V ebuildu pak načtete eclass `utils` a patch aplikujete např. takto:

```
epatch "${FILESDIR}/${P}-link.patch"
```

Pokud provozujete veřejnou overlay a máte nějaký větší patch (v hlavním stromu je limit 20 kB) nebo rovnou sadu patchů, může být vhodné je zkomprimovat a uložit na vlastní mirror. Zkomprimované patche do overlaye nepatří.

Maskování balíčků

Různé alpha, beta nebo rozbité verze je vhodné maskovat přidáním do souboru `profiles/package.mask` (relativně k overlayi). Nezapomeňte tam specifikovat verzi, protože pokud je stejnojmenný balík i v hlavním stromu, zamaskovali byste jej také. Do souboru tedy přidáte například:

```
~kategorie/program-1.0_alpha2
```

Totéž je třeba přidat do `/etc/portage/package.unmask` pro lokální odmaskování.

Live ebuildům ⁽¹²²⁾ je dobré nedávat žádný keyword (a nechat proměnnou prázdnou), protože nelze zaručit, že se poslední revize vůbec zkompileje, natož pak že bez problémů poběží na určitých architekturách. Pro lokální odmaskování takového ebuildu je třeba přidat do `/etc/portage/package.keywords` něco jako:

```
~kategorie/program-9999 **
```

Dříve se takovým ebuildům dával keyword `-*` a odmaskovávaly se pomocí:

```
~kategorie/program-9999 -*
```

ale to už je minulost, takže to nepoužívejte.

Stabilním (od upstreamu) ale neotestovaným verzím je dobré dát `~arch` keywordy (pouze architektury, na kterých víte, že se software nainstaluje a funguje). V overlayi bych si s označováním ebuildů za stabilní ani moc nelámal hlavu,

Poznámka: operátor tilda (~) na začátku znamená, že pravidlo zahrnuje i revize, takže platí i pro verzi 9999-r1, atd.

Záznamy změn

Pokud chcete, můžete si u balíčků vést záznam o tom, jak a kdy se ebuildy měnily nebo kdy se objevily nové verze. Formát těchto záznamů si prohlédněte v hlavním stromu Portage – jde o soubory s názvem **ChangeLog** (v adresáři s ebuildy). Vývojáři používají nástroj `echangelog`, který podporuje repozitáře **GIT**, **CVS** a **SVN**. Nejprve změníte, přidáte a odeberete soubory pomocí utilit pro ovládání vašeho repozitáře, poté spustíte:

```
echangelog "zpráva do changelogu"
```

a pak změny odešlete na server.

Na co si dát pozor

Je tu hned několik věcí, na které je třeba dát si pozor. Většina z nich platí, především když provozujete veřejnou overlay, kterou používá více lidí.

Nejdůležitější je dávat pozor, abyste za žádných okolností nehlásili chyby, které produkují vaše nebo jiné ebuildy, které nejsou z hlavního stromu. V případě, že si nejste opravdu jisti, že jde o problém Portage, nic nehláste, protože vývojáři pak nebývají moc nadšení.

Pokud máte v overlayi ebuild se stejnou kategorií, názvem i verzí, pak vězte, že má prioritu před tím z hlavního stromu. Je to dobrý způsob, jak si například dočasně opravit ebuild, který je v hlavním stromu poškozený nebo obsahuje chyby, než jej vývojáři opraví. Ovšem pak je třeba dávat pozor, aby vás to nezmátlo a vy jste nenahlásili chybu ebuildu mimo hlavní strom, jak je popsáno o odstavce výše.

Dávejte si pozor na kolizi názvů `eclass`. Když si do overlaye přidáte `eclass` se stejným názvem, jako je v hlavním stromu, starší verze Portage nic nenamítnou a použijí `eclass` z overlaye i pro balíčky z hlavního stromu, což může způsobit neplechu. Novější verze Portage si to hlídají a docela důrazně vás v takovém případě upozorní. Pokud opravdu víte, co děláte, varování lze zrušit přidáním `PORTAGE_ECLASS_WARNING_ENABLE="0"` do `/etc/make.conf`.

Pokud ebuild pouze neupravujete a soubory, které stahuje, tedy nejsou na Gentoo mirrorrech, je dobré přidat `RESTRICT="mirror"`, aby tam nebyly zbytečně hledány.

■

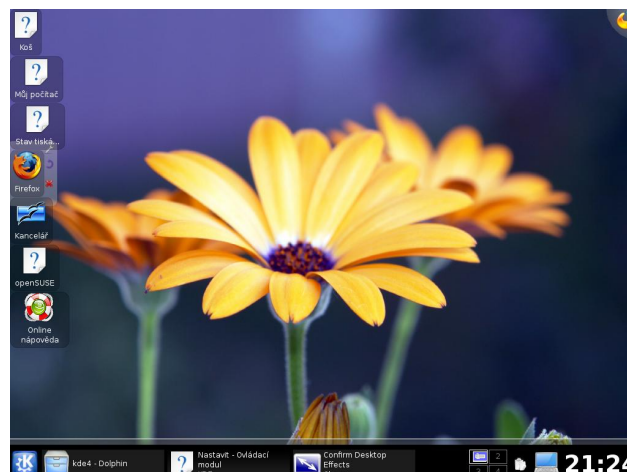
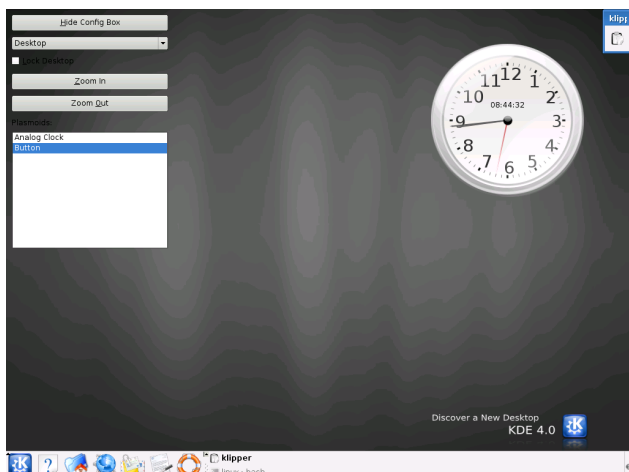
Jaké je KDE 4.0.0

KDE 4.0.0 představuje první ostrou verzi, o níž vývojáři usoudili, že je vhodná pro široké testování. Pojdme se tedy podívat na to, jak vypadá a co (ne)umí.

Michal Vyskočil

S KDE4 jsme se už na stránkách AbcLinuxu setkali mnohokrát. Naposledy u příležitosti vydání první alfa verze, která představovala první náhled na to, které [nové technologie](#) ⁽¹²³⁾ jsou ohlášeny a jak by mohlo celé [prostředí vypadat](#) ⁽¹²⁴⁾. Od té doby vykonali vývojáři a grafici mnoho práce, aby prostředí přiblížili svojí vizi dokonalého desktopu.

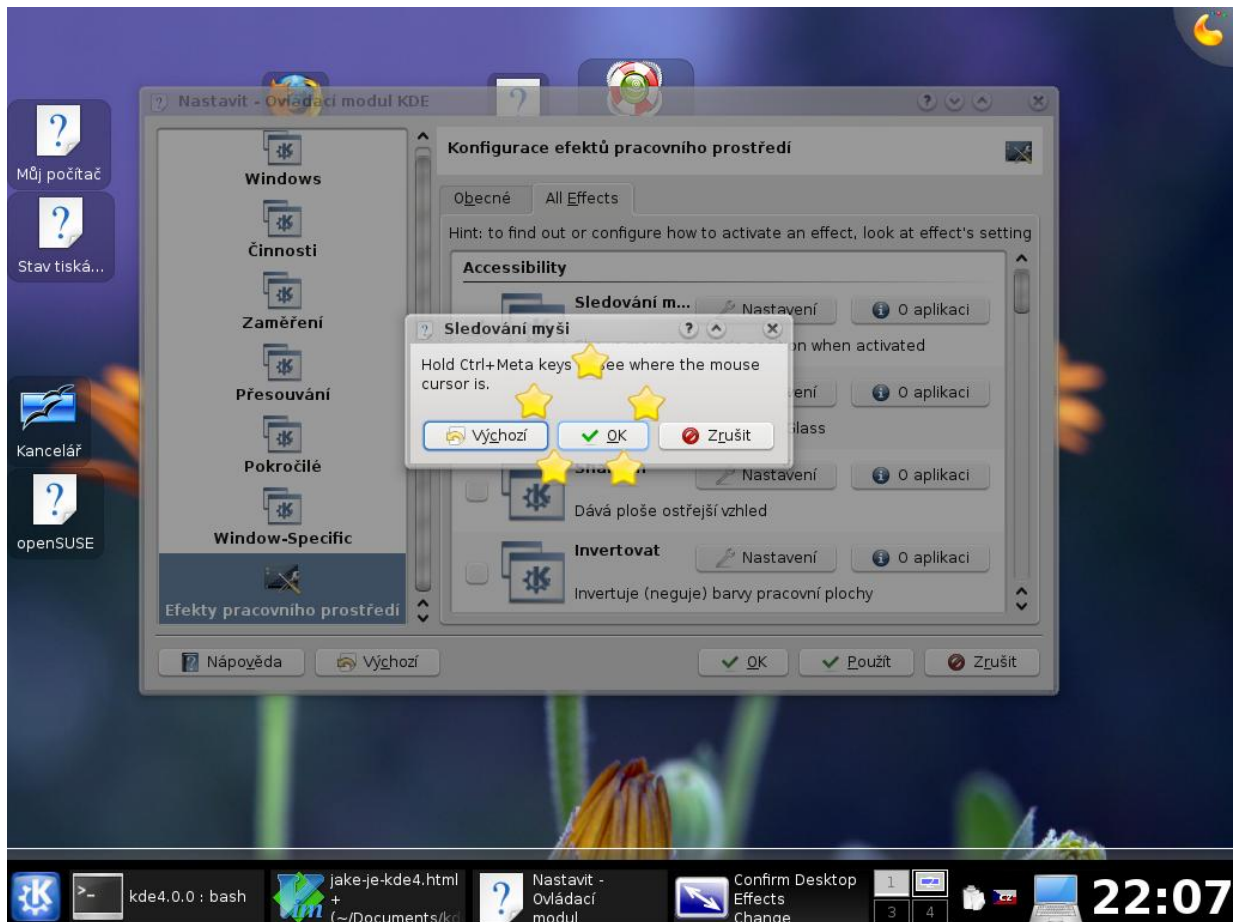
Nebude asi zcela od věci začít od možná trochu povrchního tématu, leč zcela určitě zajímavého, totiž vzhledu. Vizuální stránka prostředí doznala mnohých změn a je tak tou první věcí, která uživateli prozrazuje – pozor, jsem nové [KDE4](#) ⁽¹²⁵⁾. Stačí porovnat vzhled ve verzi alfa2 a 4.0.0:



Vizuální stránka prostředí je dotvářena dvěma základními prvky. Na jednu stranu je to téma [Oxygen](#) ⁽¹²⁶⁾, které zahrnuje ikony, styl ovládacích prvků, dekoraci oken a barevné schéma. Subjektivně hodnoceno se jedná asi o nejlepší a nejvíce propracovaný grafický styl pro KDE, který jsem doposud viděl. Novému [KDE](#) ⁽¹²⁷⁾ dává zcela odlišnou tvář. Na straně druhé stojí [Plasma](#) ⁽¹²⁸⁾ a její nejvýznačnější grafický prvek, kterým je spodní panel (a který na jistém webu považují za kopii Windows Vista). Z mého laického úhlu pohledu se zdá, jako by velký a černý panel se světlým a elegantním stylem Oxygen vůbec nešel dohromady. Nicméně nejsem grafik, takže pravděpodobně je současný stav přesně takový, jaký ho vývojáři/grafici KDE chtěli mít.

Akcelerovaný KWin

S nástupem nového KDE rovněž končí doba, kdy se museli příznivci KDE trápit s [Compiz/Beryl/Compiz Fusion](#). A závistivě sledovat, že jsou do [Gnome](#) tyto programy integrovány přeci jenom lépe. [Luboš Luňák](#) ⁽¹²⁹⁾ [[rozhovor](#) ⁽¹³⁰⁾], navzdory tomu, že nebyl docela přesvědčen o smysluplnosti tohoto kroku, vylepšil KWin tak, že se stal i z něj kompozitní správce. K mé smůle jsem si efektů příliš neužil. V testovacích verzích mi KWin na notebooku většinou s efekty nenajel vůbec. V ostré verzi již problémy se stabilitou netrpí, ale s překreslováním je to horší. Ovšem z efektů, které jsem měl tu čest vidět, mě neskutečně dostal ukazatel myši. Posuďte sami:



Jinak KWin nabízí celou řadu efektů, včetně lupy, kreslení po obrazovce, průhlednosti, stínů a ztmavování oken. Vzhledem k tomu, že titulek aktivního i neaktivního okna vypadá v tématu Oxygen stejně, zdá se, že vývojáři KDE ponechali část informací přímo na správci oken a jeho efektech.

Stinnou stránkou zůstává fakt, že je současná verze dle [poznámek k vydání](#) ⁽¹³¹⁾ stabilní, ale kompozitní vlastnosti jsou zatím v experimentálním stádiu. Pokud je třeba vypnout kompozitní vlastnosti, je možné to nejjednodušeji provést příkazem

```
writeconfig --file kwirc --group Compositing --key Enabled false
```

Plasma

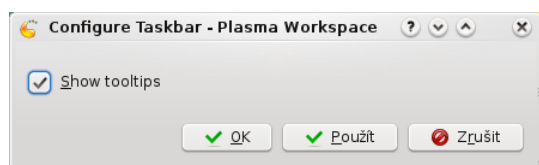
Posledním významným vizuálním prvkem je Plasma. Ta sjednocuje kdesktop a Kicker z předchozí verze. Celé je to zabaleno do SVG a technologií Qt4, které s tím umí pracovat. Plasma se projevuje několika věcmi. Pokud spustíme KDE4 poprvé, tak to vše, co vidíme, je Plasma. Ikony jsou malé plasmoidy a onen oranžový flek v pravém horním rohu obrazovky je panel sloužící k přidávání vlastních plasmoidů na plochu. A dokonce i panel ve spodním rohu je plasmoid. Vše, co vidíme a pomocí čeho ovládáme KDE, je Plasma.

Nápad je to jistě vynikající, ale líbit se vám bude jen do té doby, než vás padnuvší Plasma nechá kompletně bez desktopu. Ba co víc, chyby a nedostatky právě v ní jsou potom připočteny celému KDE4, což si zbytek komponent ne vždy zaslouží. Naštěstí je prostředí v nynější verzi schopno Plasmu při pádu alespoň restartovat, což je velice žádoucí.

Práce s plasmoidy je teď intuitivnější. Podobá se práci v prostředí Squeak, protože namísto kontextového menu jsou veškeré dostupné akce umístěny po obvodu prvku. A stejně jako ve Squeaku i u Plasmu je možné popustit uzdu fantazii a měnit velikost daného prvku nebo s ním otáčet. A jelikož jsou ikony na desktopu rovněž plasmoidy, dají se stejné operace provádět i s nimi. Osobně bych ocenil ještě funkci pro návrat do původního stavu.

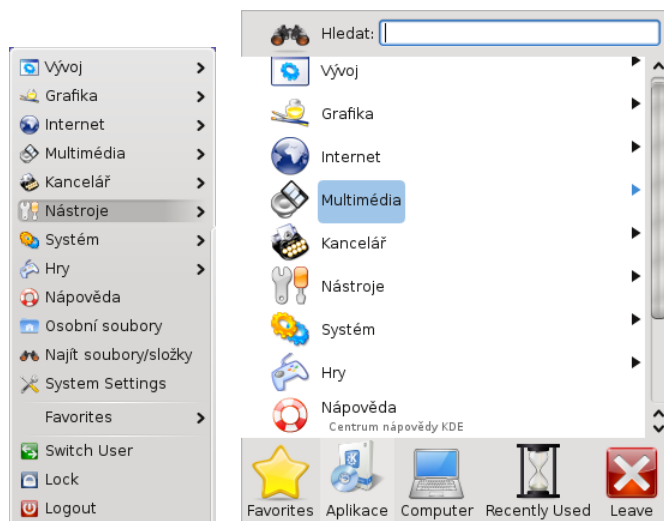


Největší nevýhoda Plasmy tak v současné době tkví v jediné věci – není zcela hotová a nedokáže plně nahradit stávající programy, především pak Kicker. Je to trochu škoda, protože se jedná o klíčovou komponentu nového KDE (pro většinu lidí je prakticky synonymem). Naštěstí se zdá, že její potenciál je značný a za nějaký čas dokáže plně uspokojit i ty největší hračky.



Spousta dotazů byla již (pravděpodobně nespokojenými uživateli) dávno položena, takže pokud nejste spokojeni s tím, že chybí milovaná funkce, přečtěte si nejprve [FAQ](#) ⁽¹³²⁾ projektu.

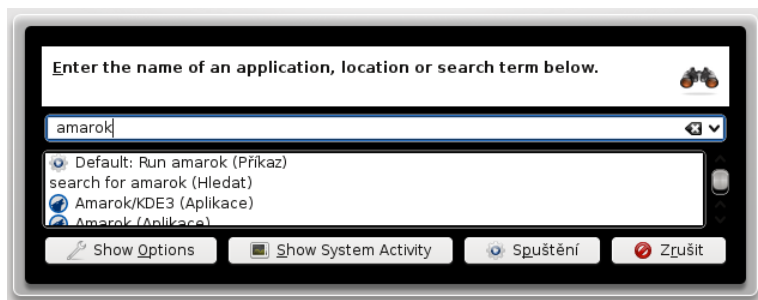
KickOff a Krunner



KickOff je nové menu, které bylo převzato z distribuce OpenSUSE. Je rozděleno do pěti sekcí: Favorites, Applications, Computer, Recently Used a Leave. Každá z nich obsahuje jiný typ položek. První *Favorites* slouží pro uložení oblíbených položek. *Applications* obsahuje seznam aplikací. Část označená jako *Computer* zpřístupňuje nastavení a důležité adresáře a část *Recently Used* položky, které byly použity naposledy. Nakonec *Leave* obsahuje příkazy pro ukončení, přepnutí uživatele nebo jeho odhlášení. KickOff rovněž disponuje funkcí pro vyhledávání:

...podobně jako Krunner, což je spouštěč aplikací. Na první pohled rozhodně zaujme jeho černý vzhled, ovšem protože je povídání o vzhledu již za námi, budu hodnotit především funkčnost.

Dokáže nabízet aplikace, které odpovídají zadanému řetězci. Na rozdíl od svého předchůdce dokáže rozlišovat a spouštět aplikace z obou verzí prostředí KDE. Ovšem ne vždy správně, protože se mi pomocí stisku Alt+F2, napsání řetězce **konsole** a stisknutí Enter podařilo několikrát spustit verzi z KDE3. Stejně jako jeho předchůdce umí **počítat aritmetické výrazy** ⁽¹³³⁾ a pracovat se zkratkami prohlížeče Konqueror.



Obsahuje rovněž zkratky pro zobrazení grafického správce procesů a možnost zadat podrobnější informace pro spuštění. Bohužel kombinace šedého písma u popisů checkboxů, černého podkladu a špatného pozorovacího úhlu na LCD měla na použitelnost až překvapivě velký a negativní vliv.

Do KDE4 bylo, na četné žádosti uživatelů, přeportováno i klasické K menu, takže jim nebrání nic v tom, aby používali to, které chtějí.

AplIKace

Dolphin

Správce souborů Dolphin byl už představen v [článku o alfaverzi](#) ⁽¹³⁴⁾. Od té doby zůstal prakticky stejný, jen obsahuje kompletní sadu ikon Oxygen a je lokalizovaný. Stále se jedná o odlehčený Konqueror (sdílející s ním podstatnou část kódu), který slouží výhradně jako jedno až dvoupanelový správce souborů.

Konqueror

Konqueror dospěl do větší stability. Na rozdíl od Dolphinu ještě prošel změnou vzhledu, která je jednoznačně k lepšímu. Nové výchozí nastavení Konqueroru z něj dělá především prohlížeč, což je pravděpodobně zásluha Dolphinu. Veškeré ovládací prvky jsou teď na jednon panelu a nezabírají tak místo. Starší Konqueror bylo třeba k podobné věci nejprve nakonfigurovat. Přehledný dialog pro nastavení naštěstí zůstal na svém místě.



Okular

Okular je nástupce **KPDF**, k němuž přidává podporu daleko většího množství formátů a nahrazuje tak různé specializované čtečky dokumentů. Mimo PDF podporuje EPS, bitmapové formáty, nativní formát **Gimpu**, DVI soubory **TeXu**, **Djvu** a navíc také **CHM** a **ODT**. Podpora **ODT** pravděpodobně souvisí s **KOffice** a je založena na stejném kódu. Zajímavé by určitě bylo přidat podporu pro dokumenty z **MS Office** a odstranit tak nutnost používat prohlížeče ve **wine**.

Okular rovněž podporuje anotace. Ty nejsou součástí dokumentu (alespoň u formátu PDF), ale jsou uloženy někde v konfiguraci prohlížeče.

Trochu jako popelka působí **Ligature**. Už proto, že standardně vybírá pouze **gz** a **bz2** soubory a alespoň v **OpenSUSE** neumí otevřít takřka nic.

Gwenview

Prohlížeč obrázků, který jako by svým názvem do KDE nepatřil. Existoval už ve verzi 3.5, ale v současné verzi nabízí přehlednější rozhraní. Posuďte sami:

System Settings

Mnohými proklínané **System Settings** z **Kubuntu** jsou teď nedílnou součástí KDE. Svým uspořádáním se silně inspirovaly prostředím **Mac OS X**. Osobně nevidím žádný velký rozdíl mezi použitelností starého a nového uspořádání, ale experti svůj názor pravděpodobně mají. Ač je změna výchozí obrazovky velice výrazná, tak obrazovky pro nastavení prakticky odpovídají těm z **KDE3**. Což je trochu škoda, protože sympatické změny uživatelského rozhraní se jednotlivých modulů příliš nedotkly.

Příkladem může být dialog pro nastavení rozložení klávesnice, jehož horní polovinu zbytečně zabírá několik checkboxů. Část pro vlastní nastavení, která je tvořena mnohem více widgety, se proto musela vměstnat do spodní poloviny.

Ostatní

O ostatních aplikacích už jenom stručně.

Amarok sice nepatří do **KDE4**, ale jedná se o jednu z nejdůležitějších uživatelských aplikací. Nová verze je zatím ve stádiu prealfaverze a uživatelé jsou dokonce žádáni, aby ani nehlásili chyby. Ale jednoduché přehrání adresáře zvládl bez problémů, ostatní funkce jsem netestoval. Potěšující je skutečnost, že už nemá problém se vejít i do menšího prostoru.

Podobně jako na plochu, i na **Amarok** je možné přidávat applety a rozhraní si tak přizpůsobovat.

VNC a **RPD** prohlížeč **krdc** doznal změn ve vzhledu. Mně vadila nefunkčnost historie a skutečnost, že při špatném zadání adresy prostě smaže celý záznam. Jinak je poměrně stabilní a **OpenSUSE** dokonce už nenabízí jeho starší variantu.

Více obrázků (zejména her a výukových aplikací) naleznete v **KDE4 visual guide** ⁽¹³⁵⁾.

Technologie

Na počátku **KDE4** byla skupina technologií a pojmů jako **Arthur**, **Phonon** ⁽¹³⁶⁾ a další [viz článek **Novinky v KDE 4** ⁽¹³⁷⁾]. Co z toho nakonec **KDE 4.0.0** obsahuje?

Qt4

Trolltech pracuje na verzi 4.4, která by měl mj. obsahovat i **WebKit** ⁽¹³⁸⁾. Současné **KDE4** v **Kubuntu** jej již **experimentálně obsahuje** ⁽¹³⁹⁾, ačkoli se zdá, že to ještě není funkční. Nová verze by měla navíc přinést vyšší výkon vykreslování, čehož bude využívat právě **Plasma**.

Plasma

Z konceptu nového desktopu se za přibližně rok a půl posunula do dnešní podoby. V současném stavu nemůže stoprocentně nahradit staré technologie (například Kicker), ale vývojáři slibují, že budou ve vývoji pokračovat. Regresí je spousta, ale Plasma slibuje i lepší udržitelnost kódu (zatímco kód Kickeru sahá až do dob KDE2).

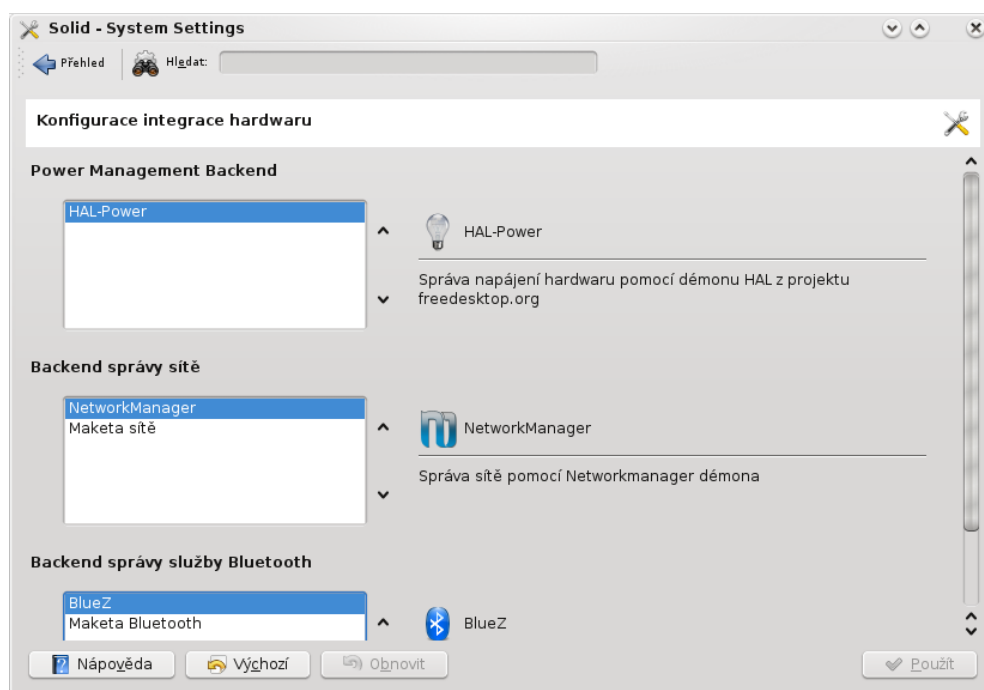
Phonon

Alias multimediální framework – co o něm říct. Funguje a jediná vizuální zmínka o něm je v nastavení KDE:

Na druhou stranu o něčem svědčí fakt, že KDE4 obsahuje pouze **KMPlayer** a ne některý z přehrávačů založených na **xine**, ať už **Kaboodle** nebo **Kaffeine** (ten ovšem nikdy nepatřil přímo do KDE).

Solid

Další z technologií, které jsou tiché a nijak se vizuálně neprojevují. O tom, že něco takového existuje, napovídá opět obrazovka v nastavení nebo plasmoid, který ukazuje naposledy připojená výměnná zařízení.



Decibel

O jeho existenci vypovídá stejnojmenný balíček a stránka decibel.kde.org⁽¹⁴⁰⁾ vypovídá o vývojářské aktivitě. V současnosti by jej měla využívat aplikace **kcall**⁽¹⁴¹⁾. Jednotlivé aplikace používající Decibel by měly mít konfigurační soubor v `/usr/share/Decibel/components/`.

Sonnet

Náhrada za **kspell2** není v KDE 4.0.0 přítomna. Trochu znepokojivý je fakt, že poslední zápisek v blogu **autora**⁽¹⁴²⁾ pochází z února loňského roku. V SVN projektu zmínky o Sonnetu jsou, ale jinak se mi nepodařilo najít něco, kde by se Sonnet mohl projevit.

Kross

Čili univerzální framework pro podporu skriptování v různých jazycích je součástí KDE4. **Současný stav**⁽¹⁴³⁾ je poměrně uspokojivý. Díky integraci kódu z KOffice do kdelibs projekt pokročil, a to dokonce i v takových věcech, jako je **podpora jazyka R**⁽¹⁴⁴⁾ v **KSpread** (**Programovací jazyk R**⁽¹⁴⁵⁾).

Nepomuk

Čili sémantický desktop, projekt sponzorovaný společností Mandriva. Současný stav ukazuje [video](#) ⁽¹⁴⁶⁾.

Strigi

Technologie pro desktopové vyhledávání. Je integrována do KRunneru a poskytuje jednoduché Qt rozhraní.

Závěrem

Celá tato recenze vznikla při několikadenním používání KDE 4.0.0. Celkový dojem zůstává mírně rozpačitý. Jistě, nějaké ty chyby se daly při té spoustě nového kódu očekávat a surové KDE4 v porovnání s vyladěným KDE 3.5 nemělo šanci tak jako tak. Pády aplikací či jejich uvíznutí v nekonečné smyčce (odkud je neuměl vypnout ani kwin) byly častější, než by člověk od ostré verze očekával. Mnoho technologií v systému sice je, ale viditelně nedokončených, na což doplácí především Plasma.

Na druhou stranu je vidět obrovský potenciál. Mnoho spíše podpovrchových změn (například Kross a podpora jazyka R) dává tušit, že budeme zásobováni zajímavými novinkami ze světa KDE4 velice často. Samotní vývojáři pokládají současné vydání pouze za první stupeň na cestě ke KDE4, takže jim jenom můžeme přát, ať se jim práce daří. Ovšem otázku, zda je KDE 4.0.0 skutečně stabilní a vhodné pro práci, si musí každý uživatel zodpovědět sám.

■

Knihy: OpenOffice.org WRITER a CALC

V této recenzi se podíváme na dvě knihy od vydavatelství Computer Media. Obě jsou to knihy o kancelářském balíku OpenOffice.org, konkrétně o Writeru a Calcu. Jde o stručné příručky, které mají své mouchy, ale přesto mohou být užitečné.

Luboš Doležel

Ty odborné výrazy...

Hned první věci, které si můžete povšimnout i na webu nakladatelství, je značný zmatek v terminologii – Writer je označován jako „textový editor“ či dokonce vysloveně mylně jako „tabulkový procesor“. V případě Calcu se podobné nejasnosti nevyskytují.

Obě knihy se snaží být vhodné i pro uživatele, kterým dosud chybí zkušenosti s používáním kancelářského softwaru. Autoři však celou věc zatahují do extrému, takže na prvních několika stranách se dozvíte, k čemu slouží posuvník, zaškrtačací políčko (checkbox), přepínač, přepínací nabídka (spin box), rozevírací nabídka (combo box), a aby byl nepřítel dostatečně zmaten, textové pole je v celé knize označováno jako dialog. Toto výrazivo je v případě knihy o OpenOffice.org Writer pro jistotu na začátku probíráno hned dvakrát – pravděpodobně někdo omylem při přípravě knihy dvakrát vložil stejný text do dokumentu.

OpenOffice.org se nevyhne porovnání s Microsoft Office – mezi výhodami je čtenář např. upozorněn na to, že cena nemusí být u softwaru jednoznačným ukazatelem kvality. Výčet nevýhod odkazuje na problémy s kompatibilitou s MS Office, později také na horší vzhled nebo „absenci vychytávek“. Čtenáři je vysvětlen i [OpenDocument Format](#) ⁽¹⁴⁷⁾, jakožto „standardizovaný formát, který podporuje bezproblémové používání dokumentů XML“. Ponechám bez komentáře.

Linux?

Autoři knihy velmi důsledně zapírají funkčnost OpenOffice.org pod Linuxem – na začátku knihy se nachází upozornění, které říká, že OpenOffice.org může být nainstalováno pod různými verzemi Windows, což může vést k jinému vzhledu titulku okna a tlačítek. I dále v knize autoři často zmiňují Windows a některé drobnosti práce s ním, i když se nejedná o nic pro Windows specifického a ani mi to nepřipadá jako obsah, který by do takové knihy patřil – tím narážím především na odstavec, kde se popisuje zobrazování oken na hlavním panelu a přepínání pomocí Alt-Tab.

Linux se dostává ke slovu až v kapitole o exportu do PDF. PDF je zde označen jako formát dokumentů nezávislý na operačním systému, díky čemuž je možno PDF soubory otevírat i na Linuxu, Solarisu a Mac OS. Celá kapitola o PDF je velice zajímavá – dozvíte se, že jiný formát než PDF se na různých počítačích může zobrazit různě v závislosti na nainstalovaných ovladačích tiskáren. Za druhou zásadní výhodu PDF je označeno to, že jej podporují všechny známé prohlížeče webových stránek.

Po přečtení knih se už opravdu nebudu nikdy divit nesmyslům, které se někdy šíří mezi méně zdatnými uživateli, protože budu vědět, že dosti zvláštní informace najdete i na poličce v knihkupectví. Odborné znalosti autorů tedy pravděpodobně nejsou nijak zvlášť vysoké, ale pokud to v jiných částech knihy nakonec povede k lepšímu chápání textu ze strany nezkušených uživatelů, snad bych to i odpustil.

OpenOffice.org Writer

V knize o Writeru se začíná u základů práce s písmem a zarovnáním textu. Další kapitoly zahrnují práci s odstavci, styly, seznamy, přípravu dokumentu pro tisk a práci s vloženými objekty – tedy žádné velké

překvapení se nekoná, úvodu do práce není co vytknout. Velká část knihy se věnuje práci s grafikou (rastrovou i vektorovou) – ta se zdá být dobře propracovaná a dobře odráží potřeby běžného uživatele. Pozitivní je také zahrnutí popisu používání funkce sledování změn či hromadné korespondence, které se mohou upotřebit obzvláště ve firemním prostředí a jsou často opomíjeny.

Nepříliš kladně bych ohodnotil odsunutí vkládání ručních zalomení textu až někam mezi ostatní funkce. Každý se už určitě setkal s dokumentem, kde někdo řeší zalomení stránky vytvářením desítek prázdných odstavců. Třeba tak někdo činí právě proto, že nepovažoval obsah kapitol někde ke konci knihy za důležitý pro kvalitní výsledek. Sám přitom vkládání ručních zalomení používám poměrně často.

Za kvalitní bych považoval také kapitoly o automatických opravách, rozdělování slov a slovníkové kontrole. Tisk dokumentu a potřebné úkony pro tuto činnost jsou vysvětlovány také. Na závěr je v knize popis práce s nápovědou a konfigurace programu. Tato kapitola se v knize o Calcu nenachází, což sice není závažný nedostatek, ale působí to dojmem nevyrovnanosti obsahu knih.

Název:	OpenOffice.org WRITER
Autor:	Michal Jiříček, Jan Pomichálek
Vydavatelství:	Computer Media
ISBN:	80-86686-75-2
Počet stran:	134
Cena: Kč	139
Web:	http://www.computermedia.cz/knihy/openoffice-writer.html ⁽¹⁴⁸⁾

OpenOffice.org Calc

S touto knihou je to poněkud horší. Systém řazení informací je značně nelogický a zmatený. Chvillemi jsem měl dokonce pocit, že se někde při přípravě kapitoly zamíchaly. Popis tak často používané operace, jako je ukládání souborů, je skryt až v sekci pokročilých funkcí programu. Nepopírám, že pro program samotný je tato operace programově náročná, avšak uživateli se bude pravděpodobně hodit už někdy ze začátku. To samé se týká práce se šablonami – v první knize je tato kapitola na začátku, zde je odsunuta na konec.

Poněkud nevhodné je rozdělení výkladu vzorců do dvou částí, přičemž ani v jedné z nich jsem nenalezl důkladný výklad toho, co vlastně vzorec je a jak se chová. To mě docela překvapuje, pokud přihlédnou k tomu, že kniha dokonce radí i v takových oblastech, jako je spouštění programu přes nabídku Start. Popis práce s grafy mi připadá dostačující a dá se říci, že čtenáře na práci s nimi připraví dobře – pozitivní jsou i různé postřehy doplňující výklad.

Jedna z kapitol pokrývá i psaní matematických výrazů (pomocí OpenOffice.org Math), bohužel velmi povrchně – odkazuje se především na nápovědu, kde se nachází příslušná dokumentace. Nechci shazovat nápovědu, ale trochu obsáhlejší úvod by byl vhodný. Na konci knihy se v podobě přílohy nachází pěkný přehled nejčastějších funkcí pro použití ve vzorcích – v tabulce jsou uváděna počestěná jména z Excelu, ale také screenshoty s příkladem použití.

Hodnocení

Pozitivní stránkou knih je důsledné doplňování všech screenshotů o šipky a jiné ukazatele. Často totiž vídám uživatele, kteří zoufale nemohou najít patřičné tlačítko ani na sebepřehlednějším místě. Nejslabší stránkou zůstávají odborné znalosti autorů a někdy nelogické pořadí kapitol. Klasickému „linuxákovi“ kniha moc zajímavostí nepřinese, je to opravdu materiál určený spíše pro ty, kteří počítač stále považují za nutné zlo, a ne za milého pomocníka.

Sazba knihy je až na jeden uříznutý obrázek dobrá, na chyby v textu (ve smyslu chybějících čárek nebo překlepů) jsem prakticky nenarážel.

Název:	OpenOffice.org CALC
Autor:	Michal Jiříček, Pavel Navrátil
Vydavatelství:	Computer Media
ISBN:	80-86686-74-4
Počet stran:	120
Cena: Kč	139
Web:	http://www.computermedia.cz/knihy/openoffice-calc.html ⁽¹⁴⁹⁾

■

Viac či menej otvorene grupvérové systémy

Dozrievanie informačných technológií v rámci firiem a organizácií prebieha niekedy veľmi podobne. Zamestnanci zvládnu používanie e-mailu, iniciatívnejší „skrotia“ kalendár. Postupne rastie potreba synchronizovať kalendáre a kontakty, stupňujú sa požiadavky na spoločné plánovanie úloh a projektov. To už žiada celkom nový prístup.

Peter Tuharský

Čas pre grupvér

Grupvér [groupware] je softvér pre podporu práce v skupinách. Typicky ide o balík pomerne jednoduchých ale veľmi efektívnych aplikácií. Jeho prekvapujúca sila spočíva v ich vzájomnom prepojení, rozmanitom využití raz vložených údajov, koordinovaní spoločných úloh. Pomáha zamestnancom uvedomiť si, že sú súčasťou tímu, zviditeľňuje sa ich úloha a zodpovednosť. Informácie sa centralizujú, čím stúpa ich kvalita, aktuálnosť a dostupnosť. Dobre nasadený grupvér znamená kvalitatívny skok vo využívaní informačných technológií. Prináša nové uhly pohľadu na firemné postupy, stimuluje ich sprehľadnenie, zlepšenie komunikácie a produktivity práce. Firmy, ktoré ho používajú, si to už bez neho asi nevedia predstaviť.

Avšak jeho nasadenie môže byť aj prekvapivo namáhavé a zložité, najmä ak sa zanedbali školenia zamestnancov, alebo ak firma nemá jasne zadefinované svoje postupy a počas nasadzovania grupvéru sa vlastne odohráva vnútorný audit.

Prínosy sú dlhodobé a nemusia byť hneď viditeľné. Niektoré súčasti sa ukážu užitočnejšie ako iné, hľadá sa optimálny rozsah a režim ich využívania. Od začiatku musíme počítať s nedodržaním časového plánu zavedenia, pretože grupvér sa viac či menej dotýka mnohých oblastí fungovania firmy, a aj tvorba potrebnej bázy údajov je časovo náročná.

Komerčné riešenia

Keď sa povie „grupvér“, väčšina ľudí znalých vecí si zrejme predstaví niektoré veľké, osvedčené riešenie – napríklad Lotus alebo MS Exchange. Ponúkajú dlhoročnú tradíciu a záštitu veľkej firmy. Ich nevýhodou môže byť cena a uzavretosť. Ak sa k cene licencií a infraštruktúry pridá ešte závislosť na ďalších komerčných produktoch, stanú sa náklady na aktualizácie nemalou každoročnou súčasťou rozpočtu.

V súčasnosti podľa prieskumov vyše polovica IT projektov končí neúspechom vo fáze nasadenia a aj úspešné z troch štvrtín meškajú. Preto najmä vo finančne citlivejších firmách môže byť investícia prijímaná s nedôverou, a v prípade neúspechu sa naozaj stáva neefektívnou. Ak si navyše vyžiada úspory na nesprávnych miestach, napríklad na školení, riziko neúspechu sa ešte zväčšuje. Pri nasadení preto môže vznikáť napätie, akokoľvek je produkt kvalitný.

Finančné nároky môžu brániť nasadeniu v potrebnom rozsahu, alebo spôsobiť odkladanie. V takom prípade môže byť otvorené riešenie lepšou voľbou. Aj solventná firma by mala najprv zmapovať svoje skutočné požiadavky a zobrať do úvahy nielen funkčnosť, ale aj mieru technologickej otvorenosti jednotlivých produktov, aby nedošlo k prílišnej závislosti na jednom dodávateľovi.

Otvorené riešenia

Grupvér už nie je trhom vyhradeným pre komerčné produkty veľkých firiem. Postupne sa na ňom začínajú udomáčať lacnejšie, viac či menej otvorené alternatívy. Svet slobodného softvéru, ako zvyčajne, ponúka

riešenia, ktoré sú už dostatočne použiteľné pre bežnú prácu. V posledných rokoch urobili výrazný pokrok, ale ešte nejaký čas potrvá, kým vystúpia z tieňa „veľkých“, renomovaných komerčných riešení a zaujmú svoje náležité miesto na trhu. Pritom ich azda jediným skutočným problémom je, že zatiaľ nie sú dostatočne známe.

Keď hovoríme o otvorenosti, tento pojem môže mať viacero podôb a stupňov:

- otvorené, štandardné technológie ako základ produktu
- z toho vyplývajúca multiplatformnosť
- licencia umožňujúca bezplatné používanie
- licencia umožňujúca neobmedzené šírenie
- otvorený zdrojový kód
- licencia umožňujúca ľubovoľné úpravy a ich ďalšie šírenie

Zrejme jediným riešením, ktoré je úplne a po všetkých stránkach otvorené, je eGroupWare, ktorý nadväzuje na starší PHPGroupWare. Bezplatnú podporu zabezpečuje komunita, komerčnú podporu zase niekoľko firiem.

Okrem toho je na trhu ešte niekoľko produktov, ktoré po niektorej stránke ponúkajú určitú otvorenosť, vo väčšej či menšej miere, čomu sa budeme podrobnejšie venovať v ďalších častiach seriálu. Celkovo sa vyznačujú mierne rôznymi rozsahmi služieb za rozdielnu cenu. Medzi takéto produkty patria OpenXchange5, Zimbra, Zoho, Sun Collaboration Suite, Mindquarry, pričom tento zoznam nemusí byť konečný. Licenčný model týchto produktov je rôzny, ale zvyčajne umožňuje bezplatné používanie základného balíka aplikácií a multiplatformnosť prinajmenšom na strane klienta, pričom prídavné moduly, komunikačné rozhrania pre prepojenie s inými programami, ako aj servisná a inštalčná podpora, bývajú viazané na platenie licenčných poplatkov.

Potenciál trhu nízkonákladových grupvérových riešení je očividne veľmi zaujímavý, o čom svedčí aj fakt, že aj Microsoft tu ponúka svoj takmer bezplatný produkt Sharepoint Team Services (platiť treba licencie za Windows server). Širšej otvorenosti sa v tomto prípade určite nedočkáme.

Cieľom nášho seriálu je postupne zmapovať a predstaviť viacero produktov, podľa miery ich otvorenosti. Zaujímavé sú najmä tie produkty, ktoré umožňujú prácu cez webové rozhranie, čím odpadá nutnosť inštalácie klientských programov. Webové rozhranie, v závislosti od použitej technológie, je zároveň akou-takou garanciou multiplatformnosti na strane klienta, čo považujeme za základnú črtu otvoreného softvéru.

Budúcnosť je na webe

Na webové aplikácie sa pripájame cez internetový prehliadač ako na bežnú webovú stránku. To je veľká výhoda, pretože každý kancelársky počítač v sieti je pripravený na okamžitú prácu s nimi, netreba nič inštalovať. Ak je aplikácia napísaná dobre, máme zaručenú plnú slobodu voľby operačného systému na strane klienta alebo aj servera. Je potom už len na nás, či používame Windows, Linux, Mac, Solaris, ... Takáto webová aplikácia potom môže byť dostupná aj mimo vnútrofirnej siete, cez Internet, takže si môžeme svoj grupvér „brať so sebou“ na služobné cesty, rokovania, pracovné stretnutia...

Spoločná charakteristika

Grupvérové riešenia sú si v základoch dosť podobné, princípy sú už rokmi overené a ustálené. Líšia sa v ponuke „nadštandardných“ aplikácií a vschopnostiach prepojenia s inými bežnými aplikáciami, napríklad poštovými klientami a plánovačmi, a ešte synchronizáciou s mobilnými zariadeniami a ich kalendármi. Prístupnosť cez bežné webové rozhranie však považujeme samo osebe za mimoriadne pružnú črtu, pretože aj keby konkrétny produkt zaostával v schopnosti synchronizácie s práve tým typom mobilného zariadenia, ktoré používame, tak cez bežné webové rozhranie sa dostaneme priamo na aplikácie a „živé“ údaje v databáze grupvéru.

Platí pritom, že čím väčšie rozlíšenie a kvalitu ponúka displej zariadenia, tým pohodlnejšiu prácu s webovou aplikáciou môžeme očakávať. Keďže rôzne variácie na tému miniatúrnych počítačov práve zaznamenávajú ďalšie obdobie rozmachu a na trh prichádzajú čoraz zaujímavejšie modely, netreba dopredu odsudzovať ani tento štýl práce.

Napokon, ak skutočne potrebujeme synchronizáciu kalendárov, kontaktov a pošty s natívnymi aplikáciami mobilného zariadenia, máme k dispozícii niekoľko uznaných štandardov, takže by s ňou nemali byť problémy, ak pravda nejde o neúnosne uzavreté a neštandardné zariadenie či aplikáciu.

Oplatí sa?

Bezplatné alebo nízkonákladové riešenia približujú plnohodnotný grupvér úplne každému. Záleží pritom na našom rozhodnutí, či je dôležitejšia nízka (prípadne až nulová) cena, funkčné možnosti, konkrétna funkcia, ktorá sa nám javí ako kľúčová, alebo je podstatná otvorenosť a dôraz na štandardy. V tomto ohľade sa líšia aj priority autorov jednotlivých produktov, pričom je zaujímavé, že napríklad autori eGroupWare považujú za najdôležitejšie práve posledné menované kritérium, pretože dáva istotu, že naše údaje sa nikdy nestanú rukojemníkom dodávateľa.

Už samotná nízka či dokonca nulová cena má niekoľko priaznivých „vedľajších účinkov“. Tým, že nie sme vystavení riziku veľkej a potenciálne neefektívnej investície, získavame nebývalú voľnosť. Bez zbytočného stresu a tlaku nasadíme jednotlivé grupvérové aplikácie podľa potrieb firmy, školení a schopností používateľov. Môžeme si dopriať to, čo je vzácnosťou pri nasadzovaní veľkých IT projektov – dostatok času.

Napríklad najprv neoficiálne zapojíme skúsenejších používateľov, ktorí čoskoro objavia, aké silné nástroje majú k dispozícii, a sami začnú žiadať viac. Najlepšie bude, ak začnú pozitívne vplyvať na firemné procesy. Azda niet lepšieho spôsobu nasadenia IT projektu než „revolúcia zdola“, metodicky usmernená a s podporou vedenia.

Keď postupne objavíme možnosti, optimálny režim používania, predstavu o našich skutočných požiadavkách, dokážeme už porovnať ostatné riešenia na trhu. A čo ďalej? Môžeme:

1. Zostať pri otvorenom grupvéri a ťažiť z jeho pozvoľného vylepšovania.
2. Investovať, podporiť jeho vývoj. Oslovíme schopnú vývojársku firmu, ktorá ho dokáže prispôbiť naším potrebám – odstrániť chyby, pridať nové funkcie alebo aj celú aplikáciu. Alebo si zaplatíme komerčnú podporu.
3. Vyskúšať iný otvorený produkt.
4. Kedykoľvek môžeme kúpiť licencie na produkt, ktorý viac vyhovuje našim požiadavkám – či už „veľký“ komerčný grupvér, alebo „ešte väčší“ integrovaný podnikový systém.

Vďaka otvorenému grupvéru sme zatiaľ ušetrili na licenčných poplatkoch, zmapovali skutočné potreby, sprehľadnili firemné toky, zamestnanci sú vyspelejší. Nasadenie komerčného produktu už bude podstatne jednoduchšie. Dodávateľ by mal vykonať migráciu údajov, ktoré sme už do grupvéru vložili.

Nabudúce sa pozrieme na eGroupWare, úplne otvorený a bezplatný grupvérový balík.

■

Inkscape – 6 (vizitka)

Jako správný linuxový guru se jistě potřebujete umět „prodat“. Základem toho, aby si vás zapamatovali, je slušná vizitka neboli navštívenka. Pojďme si v Inkscape jednu opravdu reprezentativní zpracovat.

Vítězslav Válka

Základní pravidla

Ještě než otevřete **Inkscape**, měli bychom si zopakovat pravidla, která pro vizitky obecně platí. Základem je rozměr 90 × 50 mm, který striktně dodržujte. Pokud budete chtít v tomto případě experimentovat, buďte se do běžné složky na vizitky ta vaše nevejde, nebo tam bude působit jako chudý příbuzný. Stejně tak je to s použitím papíru o minimální gramáži 200 g/m (lépe 300 g/m) a volbou tiskárny, která by rozhodně neměla být inkoustová (není nic horšího, než podávat rozmazanou vizitku na pomačkaném kancelářském papíru o gramáži 80 g/m).

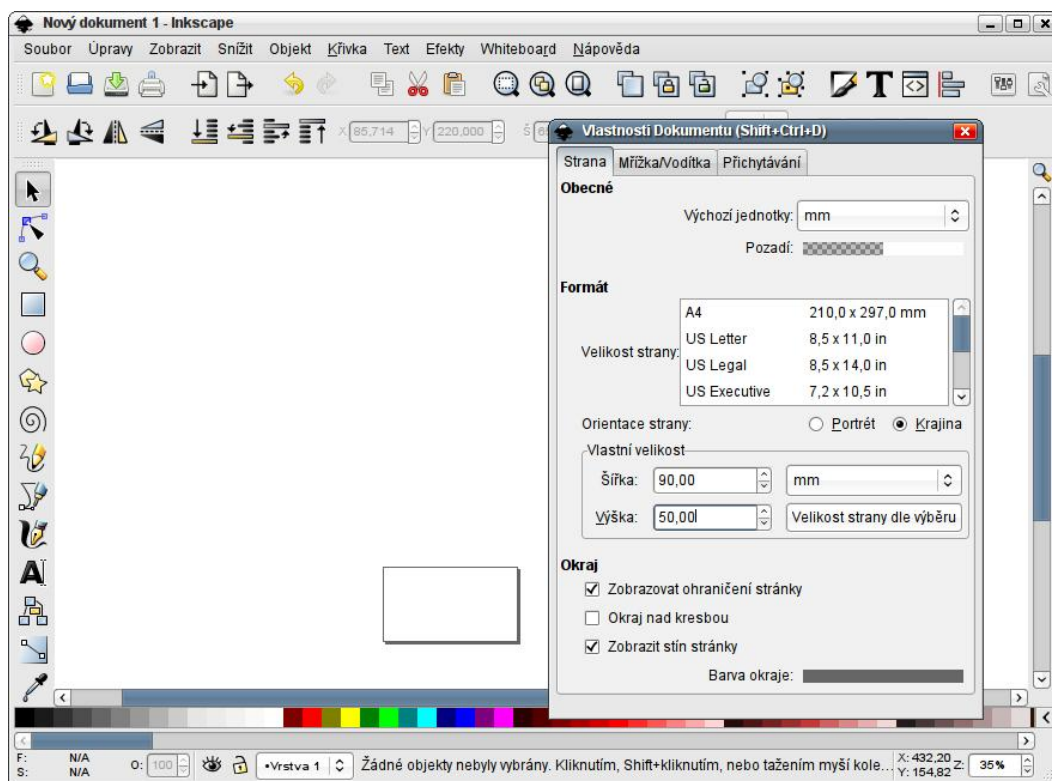
Přípravné práce

Rozmyslete si, co chcete na vizitce propagovat. Jestli firmu nebo vaši osobnost. V ideálním případě to bude jedna věc. Propagovat současně „Odbornou znalost Linuxu“ a „Prodej ovoce“ nepůsobí právě skvěle. Lépe uděláte, když se budete prezentovat jako specialista na jednu věc (pokud opravdu prodáváte přes den ovoce a večer se zdokonalujete v Linuxu, udělejte si dvě sady vizitek). Snadno můžete uplatnit staré dobré pravidlo: „Méně je více“.

Obsah vizitky

Máme tedy jasný rozměr a víme, co chceme na vizitce mít. Základní sada informací je: titul, jméno, pozice (zaměření), telefon, email, web a případně logo. Pokud máte pocit, že příjemce ocení i Jabber, ICQ či Skype, můžete to přidat také. Mnohdy stačí, když tyto „další“ kontakty uvedete až na svém webu nebo v patičce emailu. Co se týče poštovní adresy, tu bych uváděl pouze v případě, že „k vám“ chcete někoho pozvat. Vhodné je to pro firmy, které mají reprezentativní kanceláře, ale pokud u vás doma klávesnice volně přechází ve svačinu, mohli byste být takovou návštěvou nepříjemně překvapeni.

Otevřete si **Inkscape**, změňte v nastavení dokumentu jednotky na „milimetry“ a nastavte formát na 90 mm šířky a 50 mm výšky. Vizitka na výšku není problém, ale opravdu profesionální vizitky na výšku nebývají. Je to proto, že se používají držáky, složky, podavače, které jsou koncipovány výhradně na šířku.



Připravte si tedy text podobný tomuto:

Ing. Vladimír Tux
 Poradce v oblasti Linuxu

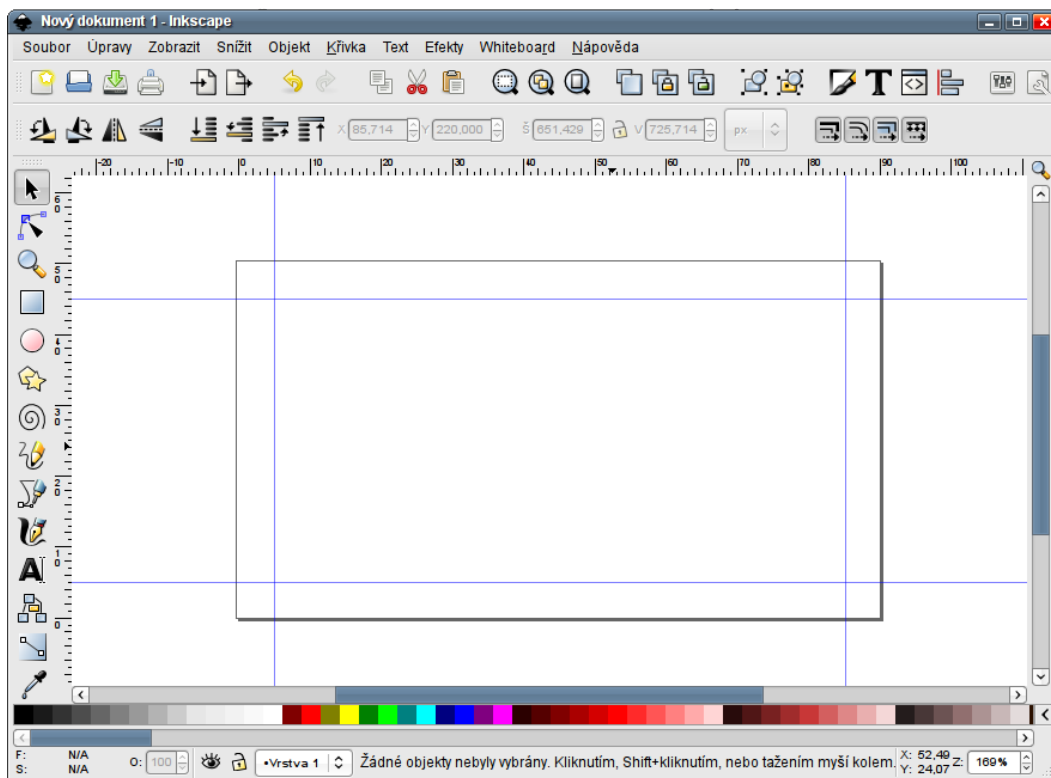
Tel.: +420 609 112 777
 Email: vladimir.tux@abclinuxu.cz
 Web: <http://www.abclinuxu.cz/blog/tux>

Doporučuji dodržet dělení telefonního čísla po třech cifrách. Pomáhá to přehlednosti a když pak číslo z vizitky opisujete, jde to lépe.

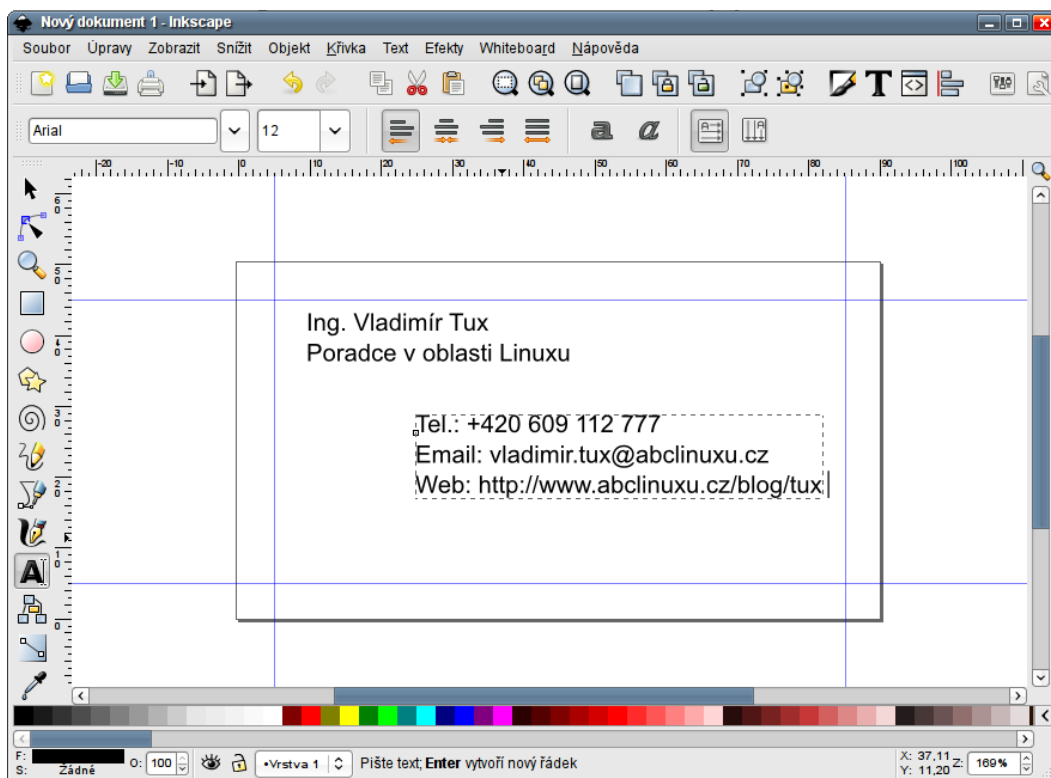
Samotná sazba vizitky

Máme správný formát a připravený text. Pojdme to dát dohromady. Doporučuji zachovat odsazení obsahu nejméně 4 - 5 mm od okraje vizitky. Může se jí náhodou ohnout okraj nebo může být v držáku s rámečkem. Velikost písma by neměla v běžných případech přesáhnout 14 - 16 bodů velikosti písma. Stejně tak by neměla klesnout pod 9 bodů, protože vizitku může číst někdo s horším zrakem. Zvolený font nechám na vás; pokud však nechcete ohrozit svou důvěryhodnost, nepoužívejte nečitelná písma, ručně psaná písma a obecně příliš neexperimentujte – opět je cílem, aby bylo možné vizitku použít k účelu, k němuž je určena.

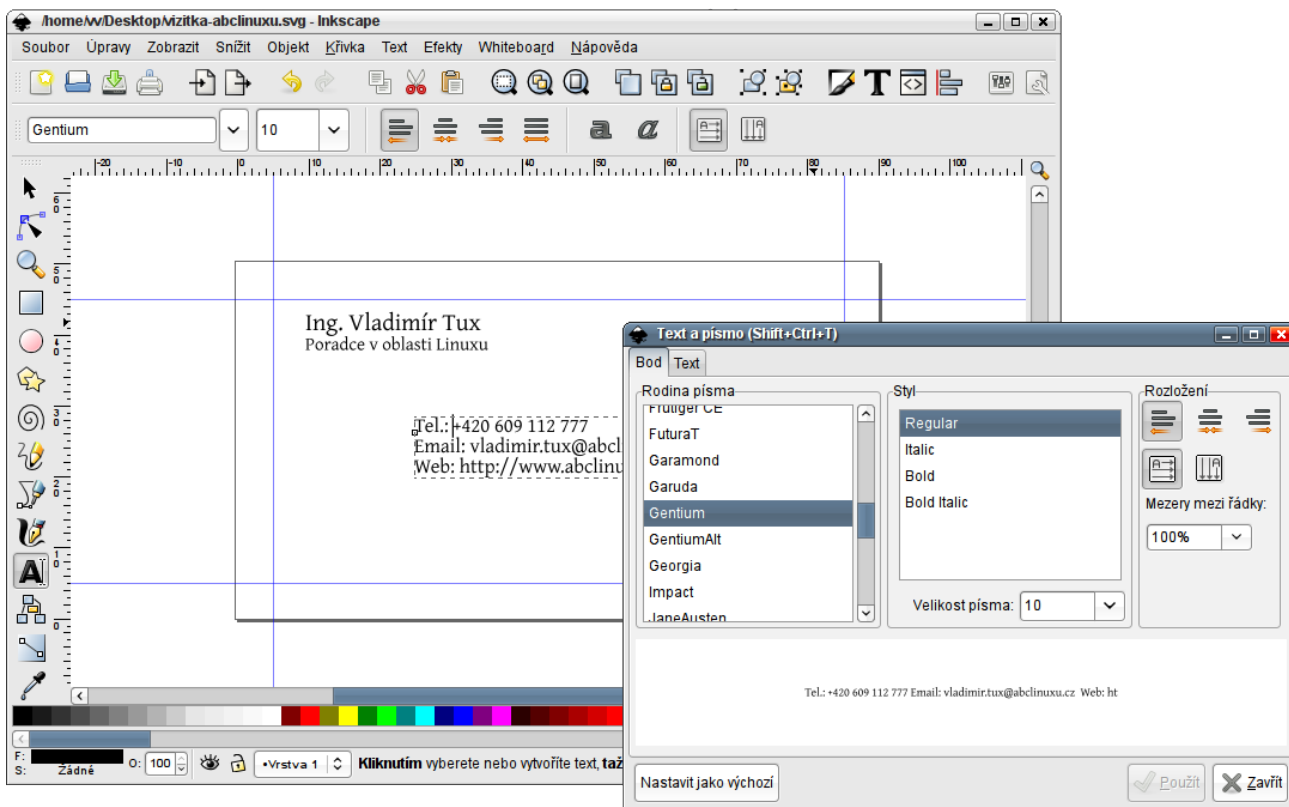
Okraje si vyčleníte snadno pomocí vodítek, které „vytahujete“ kliknutím a tažením myši z postranních pravitěk (Pokud máte pravitka skryta, stačí zmáčknout **Ctrl+R** nebo hledat v menu **Zobrazit**). Pokud některá vodítka nechcete, snadno je zase „zatáhněte do“ pravitěk.



Zvolte nástroj *Text* v levém panelu (ikona „A“ resp. klávesa **F8**) a klikněte někam do plochy vizitky. Připravený text si přesuňte ze schránky nebo rovnou napište ten váš. Já jsem zvolil rozdělení textu na dvě poloviny. V první mám jméno a zaměření. Zbytek tvoří kontakty.

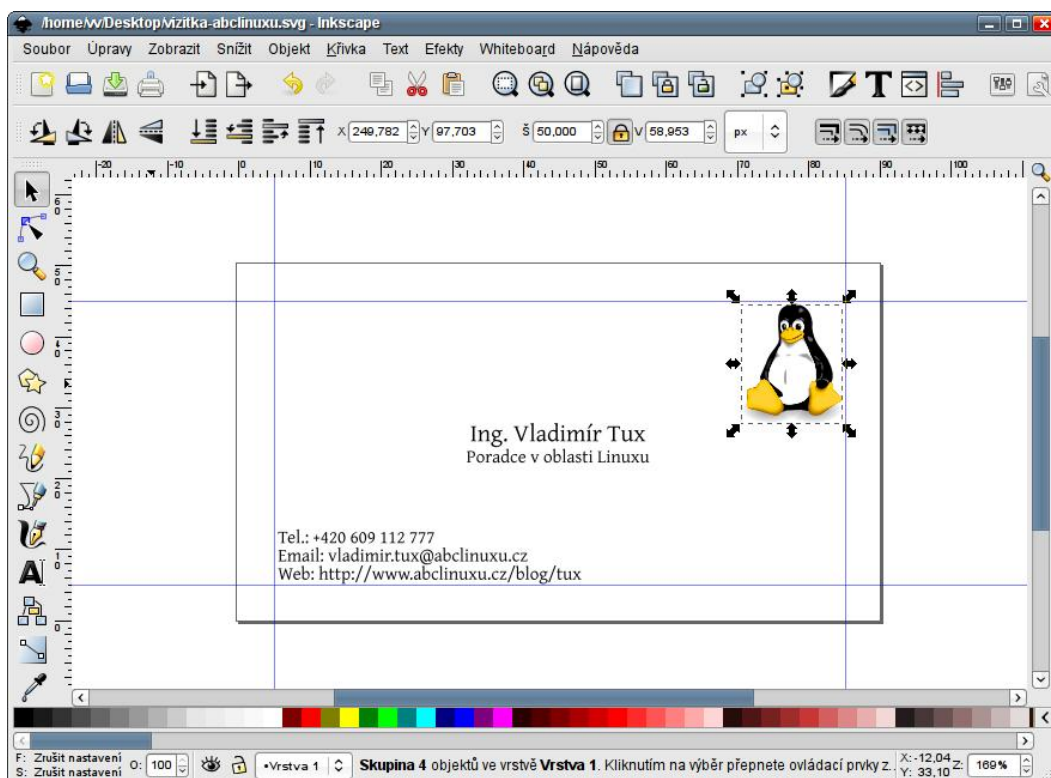


Volbou vhodného fontu a nastavením jeho velikosti (označte textový objekt nebo některé slovo v něm a zmáčkněte **Ctrl+Shift+T** nebo použijte opět nástroj *Text* v levém panelu resp. klávesu **F8**) dosáhnu požadovaného výsledku. Další práci s textem se můžete naučit ve čtvrtém díle seriálu, který byl věnován textu (*Inkscape – 4 (text)*⁽¹⁵⁰⁾). Pak už jen zarovnáme textové objekty (**Ctrl+Shift+A** nebo ikona v horní nástrojové liště) tak, aby výsledek odpovídal naší představě.



Ještě logo

Pouze textová vizitka nepůsobí právě elegantně. Je vhodné ji obohatit o vaše logo. Pokud žádné nemáte, klidně můžete použít obrázek Tuxe z Wikipedie (<http://cs.wikipedia.org/wiki/Tux>⁽¹⁵¹⁾); který je tam přímo ve formátu SVG, který Inkscape používá jako svůj nativní. Stažený obrázek pomocí **Ctrl+I** vložíte přímo do vašeho rozpracovaného souboru s vizitkou.



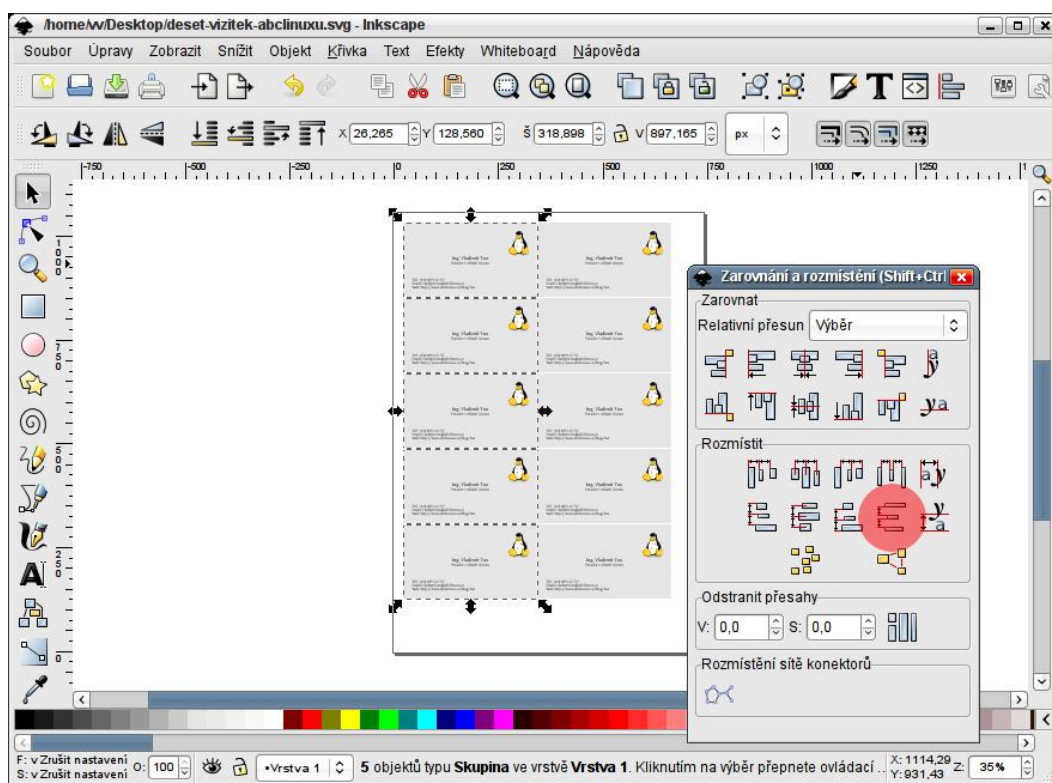
A je to! Teď už vizitku jen uložte jako PDF, protože formát SVG pro tyto účely není běžný. A pošlete výsledek do tiskárny na rohu.

Tip – tisk vizitek na vlastní tiskárně

Pokud chcete ušetřit nebo věříte tomu, že vaše domácí tiskárna je to nejlepší na světě, můžete se svým dílem ještě chvíli pracovat tak, aby jej pak bylo snadné vytisknout doma v teple.

Pod vizitku si vložte světle šedý obdélník (přesun objektu do spodku lze provést pomocí klávesy `Page Down`, nahoru zase `Page Up`), přesně velikosti vizitky (90 × 50 mm) a ten zarovnejte na střed stránky pomocí `Ctrl+Shift+A`. Vše označte a seskupte pomocí `Ctrl+G`. Otevřete si úplně nový soubor, který má standardně rozměry A4 (210 mm × 297 mm ale standardně uvedeno v pixelech). Pomocí schránky si do nového okna přesuňte zpracovanou vizitku (`Ctrl+C` a `Ctrl+V`).

Umístěte ji do levého horního rohu a pomocí `Ctrl+D` zduplikujte. Duplikát pak pomocí levého tlačítka myši a přidržení `Ctrl` posuňte vodorovně doprava, přesně vedle originálu. To stejné pak 5× udělejte s oběma vizitkami, ale ve svislém směru dolů. Vejde se vám tam 10 vizitek s tím, že něco zbylo na okraje. Označte levý sloupec vizitek (ideálně obdélníkovým označením) a otevřete okno s nástrojem zarovnání `Ctrl+Shift+A`, kde kliknete na srovnání mezer mezi vizitkami (pasáž Rozmístit).



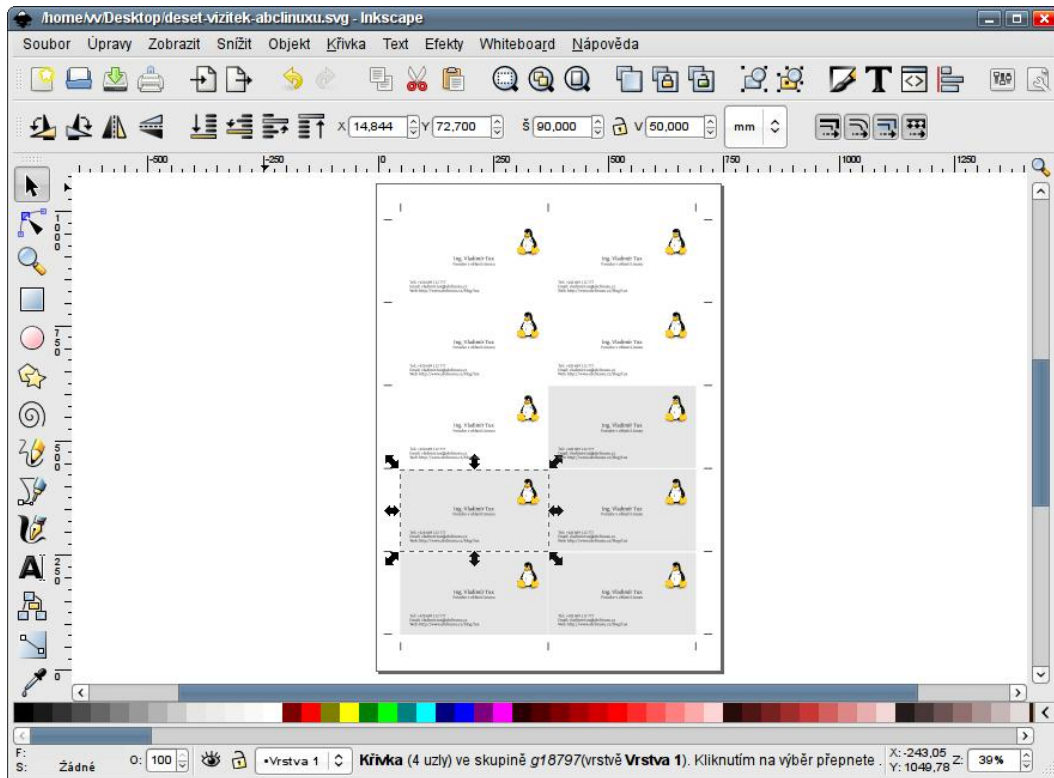
To stejné uděláte s pravým sloupcem vizitek. Následně je všechny označte, seskupte pomocí `Ctrl+G` a opět nástrojem zarovnání dejte zarovnat, tentokrát na střed stránky.

Ořezové značky

Amatérům by stačily okraje obdélníků kolem vizitek a nůžky. Pokud však chcete dosáhnout maxima, připravte si ořezové značky tak, aby na okrajích vizitky nezůstaly vytištěny nehezské čáry.

Vytvořte si černý obdélník o rozměru 5 mm na 0,3 mm. A zarovnejte jej s horní hranou vizitek. Duplikujte jej a přesuňte za pomocí `Ctrl` vodorovně doprava (tak, aby se okraje vizitek ani v jednom případě nedotýkal). Obdélníky seskupte a duplikujte pomocí `Ctrl+D`. Pomocí `Ctrl` přesuňte svisle dolů a zarovnejte se spodní

hranou vizitek. Následně 4× duplikujte jednu z těchto dvojic a pak je všechny označte, ale neoznačte vizitky. Opět pomocí nástroje zarovnání (**Ctrl+Shift+A**) sjednoťte mezery mezi dvojicemi a potom tyto ořezové značky seskupte (**Ctrl+G**) a vyrovnejte na střed strany (**Ctrl+Shift+A**). Postupujte podle tohoto odstavce obdobně i při tvorbě svislých značek.



Pak už jen pomocí **Ctrl** a levého tlačítka myši označte a smažte jednotlivé šedé podklady vizitek (**Ctrl** umožňuje označit objekt v rámci skupiny). Výsledek vytiskněte na své tiskárně a potom pomocí ocelového pravítka a ostrého řezáku rozřezejte na jednotlivé vizitky. Nemusíte se bát, že si jednotlivé značky odříznete, pokud uděláte nejprve svislé a potom vodorovné řezy a budete pevně tlačit ocelové pravítko k podložce, nic takového nehrozí.

Zdrojové dokumenty

[Linuxová vizitka ve formátu SVG](#) ⁽¹⁵²⁾

[Sada deseti vizitek ve formátu SVG](#) ⁽¹⁵³⁾

■

Bankovníctví Komerční banky v Linuxu

24. listopadu 2007 nastala změna, ve kterou leckdo už ani ne-doufal (a třeba už dlouho před tímto dnem změnil peněžní ústav) – Komerční banka upravila své internetové bankovníctví MojeBanka tak, že by mělo fungovat (téměř) nezávisle na operačním systému a prohlížeči. Podívejme se, jak to s tímto bankovníctvím vypadá v Linuxu.

Lukáš Jelínek

Historie bankovníctví KB

Komerční banka spustila provoz internetového bankovníctví v roce 2001. Vznikly dva různé produkty: čistě webové bankovníctví *MojeBanka* (pro provoz stačí prohlížeč) a dále produkt *ProfiBanka*, u kterého jsou data ukládána částečně i do databáze běžící na počítači, kde se s bankovníctvím pracuje. Později vznikla ještě třetí varianta – „odlehčené“ bankovníctví *Expresní linka Plus* (nižší stupeň zabezpečení, méně funkcí, nízké limity).

Obecně nejzajímavější službou je MojeBanka (ProfiBanka se hodí pro střední a velké firmy, Expresní Linka Plus je navržena jako doplněk k jiným komunikačním kanálům), zbývající část článku se bude týkat této služby.

Při svém vzniku bylo bankovníctví MojeBanka pevně připoutáno na softwarovou platformu tvořenou operačním systémem Microsoft Windows a webovým prohlížečem Microsoft Internet Explorer. Bylo to dáno tím, že se zpočátku používala komponenta technologie ActiveX. Další potřebnou technologií byla Java, ovšem pouze implementace Javy od Microsoftu, tedy už v té době notně zastaralá.

Později bylo bankovníctví upraveno. Podpora Javy byla rozšířena i na novější verze od Sun Microsystems a především byla odstraněna závislost na ActiveX. Závislosti na operačním systému a prohlížeči se však MojeBanka nezbavila.

K další změně došlo v srpnu roku 2006. Zabezpečení bankovníctví bylo totiž příliš slabé. Byla sice možnost používat čipovou kartu, využil toho však málokdo. Většina klientů využívala základní verzi zabezpečení, tedy elektronické podepisování příkazů založené na soukromém klíči chráněném heslem. Protože došlo k útokům na účty některých klientů (trojský kůň na jejich počítačích odeslal útočnickovi soubor s klíčem a současně zaznamenal stisky kláves při zadávání hesla), bylo zabezpečení posíleno povinnou autorizací prostřednictvím jednorázového kódu zasílaného jako SMS na mobilní telefon.

Komerční banka byla dlouhá léta bombardována e-mailovými i telefonickými dotazy, kdy a zda vůbec bude bankovníctví upraveno tak, aby fungovalo i v jiných prohlížečích a operačních systémech. Konkurenční peněžní ústavy mezitím vesměs podporu implementovaly, nebo ji měly k dispozici již od počátku provozu internetového bankovníctví. Pouze Komerční banka odolávala a na dotazy odpovídala způsobem, který sice přinášel určitou naději, současně ale vůbec nic nesliboval.

Ke zlomu došlo až v roce 2007. Koncem října banka oznámila, že od 24. listopadu nastanou značné změny v internetovém bankovníctví. Jednou z hlavních změn měla být podpora dalších operačních systémů a prohlížečů. Podpora byla rozštěpena do dvou úrovní – jedna znamená plnou podporu (včetně asistence klientské linky při nastavování), druhá podporu na funkční úrovni (tedy že by to mělo běžet, ale každý si to musí nastavit sám).

Onoho dne skutečně ke změně došlo. Samozřejmě se mohly vyskytnout různé „dětské nemoci“, proto něco nemuselo zpočátku fungovat úplně nejlépe. Nyní ale už uplynula dostatečně dlouhá doba, aby bylo možno funkčnost bankovníctví podrobit testu v různých prostředích.

Do banky v Linuxu

Test funkčnosti bude zaměřen pouze a jen na GNU/Linux. Testovací distribucí je openSUSE 10.3 ve 32bitové verzi. Testoval jsem i na distribuci Fedora Core 6 (opět dvaatřicetibitové) s naprosto stejnými výsledky. Výsledky pro 64bitové prostředí uvedu samostatně.

Co se týká Javy, testoval jsem jak Javu 5.0 (konkrétně 1.5.0_14-b03), tak i verzi 6.0 (Update 4). V obou případech byly výsledky stejné. Technické podmínky služby MojeBanka sice uvádějí konkrétní podporované verze, nicméně používání starých verzí (v nichž mohou být bezpečnostní chyby) není zrovna dobrý nápad. Proto jsem vždy používal (a v tomto testu tomu bylo nejinak) aktuální verzi Javy.

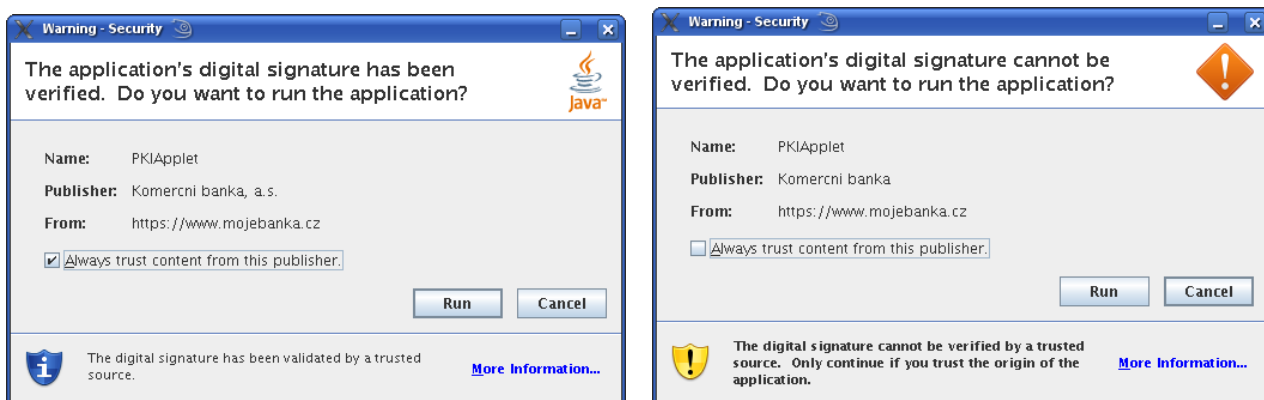
Mozilla Firefox

Firefox je nejpoužívanější prohlížeč pro GNU/Linux, proto přijde na řadu první, a to ve verzi 2.0.0.10. Při přechodu na adresu <https://www.mojebanka.cz/> ⁽¹⁵⁴⁾ se objeví stránka s informacemi o tom, jak bude Java v následujících chvílích „vyhrožovat“ a co se s tím má dělat. Stránka je stejná i pro další prohlížeče.

Následně je potřeba mít nainstalován plugin pro Javu. Pokud je plugin k dispozici, proběhne kontrola transparentně, v opačném případě skončíte na stránce zvané Konfigurační průvodce. Tam se dostanete i ve všech dalších případech, kdy není splněna některá nutná podmínka funkčnosti bankovníctví (např. není povolena Java).

Poznámka: Mezi podmínkami uváděnými průvodcem je také přítomnost podporovaného PDF prohlížeče. Bankovníctví bude fungovat i bez něj. Pouze nebude možné zobrazovat výstupy generované ve formátu PDF, případně (pokud prohlížeč k dispozici je, ale není propojen s prohlížečem) zobrazovat půjdou, ale v samostatném okně apod.

Při načítání přihlašovací stránky vyskočí dva potvrzovací dialogy. Každý se týká jednoho JAR archivu s javovými třídami. U prvního jde jen o potvrzení souhlasu s privilegovanými operacemi. Druhý ale navíc hlásí, že podpis archivu není důvěryhodný. Java nezná autoritu, která podepsala certifikát, jímž byl podepsán archiv appletu. Jenže je to poměrně podivné, protože onou autoritou je přímo firma Sun Microsystems, takže by v instalačním balíku Javy měl být potřebný certifikát přítomen. Těžko říct, v čem je problém...



Po odsouhlasení dialogů (lze zvolit, aby se příště nemusely znovu potvrzovat), výběru souboru s certifikátem a přihlášení do aplikace lze s bankovníctvím pracovat. Všechny funkce pracují bez problémů.

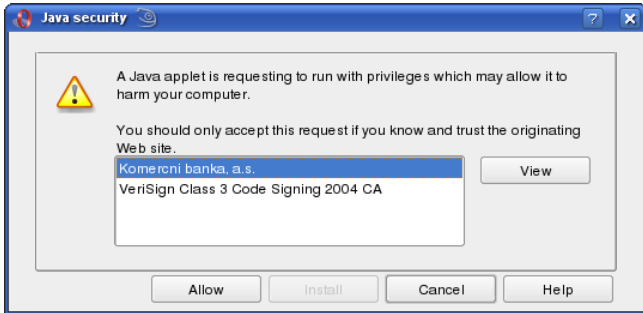
Opera

Dalším testovaným prohlížečem je Opera – a sice verze 9.25. Tady je to s varovnými dialogy ještě o něco složitější. Jsou totiž hned tři (dva se týkají archivů podepsaných certifikátem, který byl podepsán autoritou VeriSign), třetí je pak pro druhý archiv, stejně jako v případě prohlížeče Firefox.

Nepříjemné ale je, že se tyto dialogy zobrazí vždy, nelze je potvrdit natrvalo. Asi by to bylo řešitelné přes ruční nastavení v souborech Javy ([java.policy](#), [java.security](#)), ale nejde rozhodně o triviální věc. Proto

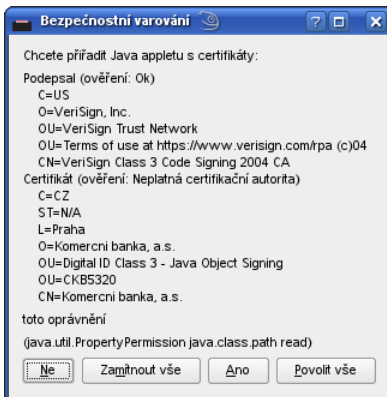
kdo chce používat pro bankovníctví tento prohlížeč, nezbyvá mu, než se s odklikáváním dialogů smířit (s tím, že by si správně měl vždycky zkontrolovat, jestli nejde o pokus o útok).

Po překonání této nepříjemnosti už to jde jako po másle. Stačí (při první návštěvě) vybrat cestu k certifikátu, přihlásit se a normálně pracovat. Všechno funguje, jak má.



Konqueror

Test funkčnosti završím prohlížečem Konqueror 3.5.7. Vše se velmi podobá oběma předchozím prohlížečům – tentokrát to ale nezná ani autoritu VeriSign. Při pokusu provést operaci vyžadující větší práva se však Java dotazuje jednotlivě – tedy pro každé konkrétní oprávnění, které je potřeba (např. `java.util.PropertyPermission java.class.path read` – což značí povolení ke čtení systémové vlastnosti obsahující cesty k javovým třídám).



Lze „odklepávat“ jednotlivá povolení (a vidět, čeho konkrétně se applet domáhá) nebo stisknout *Povolit vše* a víc to neřešit. Dotazy (resp. skupiny dotazů) budou celkem dva, jeden pro každý JAR archiv. Nastavená povolení se uloží a příště už nebude uživatel znovu obtěžován.

Při pokusu zadat cestu k certifikátu ovšem nastane problém. Někde je chyba (v implementaci appletu nebo v knihovnách Javy) – nelze totiž vybrat cestu v souborovém dialogu. Přesněji řečeno ji vybrat lze, ale místo ní se v poli appletu objeví text „undefined“. Jedinou možností je napsat cestu ručně nebo ji odněkud překopírovat. Chybu už jsem ohlásil Komerční bance, takže třeba bude v budoucnu odstraněna.

Tím bohužel problémy nekončí. Kdo nemá povolena vyskakovací okna, dočká se pouze toho, že se mu objeví varování o pokusu otevřít takové okno. Pokud není otevření povoleno, vůbec se nezobrazí přehled účtů a s bankovníctvím nelze pracovat (při kliknutí na libovolnou položku z menu se pouze zobrazí zpráva, že načítání ještě není dokončeno). I v tomto případě jde o chybu, ovšem pravděpodobně v Konqueroru – javascriptové zobrazení do jiného rámu je asi vnímáno jako otevření pop-up okna. Prozatímním řešením je povolit si vyskakovací okna pro doménu `mojebanka.cz`.

To je naštěstí ohledně potíží všechno a další práce s bankovníctvím je již zcela bezproblémová. Všechno správně funguje a rovněž vzhled není nijak „rozhozený“ nebo jinak neobvyklý. Pouze může působit trochu divně, že dotaz na zavření okna se zobrazí dvakrát (a pak se teprve okno zavře).

Další testy

Certifikační průvodce

Tzv. certifikační průvodce je webová aplikace doplňující bankovníctví MojeBanka a slouží k práci se souborem obsahujícím klíče a certifikát. Prostřednictvím průvodce lze prodloužit platnost certifikátu (resp. před skončením platnosti vytvořit nový), změnit e-mailovou adresu, zkontrolovat soubor s certifikátem atd.

Podle provedených testů fungoval průvodce ve všech třech testovaných prohlížečích. Podmínky funkce jsou stejné jako pro samotné bankovníctví – i průvodce je založen na javových appletech.

64bitový Linux

64bitové systémy se již stávají běžnou záležitostí. Proto leckoho zajímá, jak na nich fungují i takové věci, jako je právě internetové bankovníctví. Vyzkoušel jsem tedy funkčnost bankovníctví i s 64bitovým Linuxem, konkrétně na distribuci Kubuntu 7.10.

Na 64bitových systémech má člověk dvě možnosti, jak provozovat aplikace. Jednak v nativním (64bitovém) režimu, a pak také v režimu kompatibility (32bitové aplikace). Pokud se využije kompatibilní režim, bude se jednat o 32bitový prohlížeč a taktéž 32bitovou Javu, čili se situace převádí na tu, která byla na 32bitovém systému (pomineme-li případné chyby zaviněné vývojáři některých použitých knihoven – ale dnes již nejsou příliš časté).

Zajímavější je zjištění situace v čistě 64bitovém prostředí. Tam to vypadá tak, že existuje 64bitová verze Konqueroru a Firefoxu. Opera zatím takovou stabilní verzi nemá – lze použít verzi 9.50 beta, ale ta si s Javou příliš nerozumí a velmi často padá.

Také samozřejmě potřebujeme ještě 64bitovou Javu. Ta je již delší dobu k dispozici, nicméně v ní zatím chybí plugin potřebný pro některé prohlížeče. S Firefoxem, který tento plugin používá, tedy nepochodíme. Konqueror plugin nepotřebuje, vystačí si se samotným prostředím JRE.

Zkusil jsem v daném prostředí (tj. 64bitový GNU/Linux + 64bitový Konqueror + 64bitová Java) internetové bankovníctví MojeBanka a výsledek byl naprosto stejný jako v případě 32bitového prostředí. Kdo se tedy smíří s výše popsanými neduhy fungování v Konqueroru, může bankovníctví bez dalších potíží používat.

Zhodnocení

Internetové bankovníctví Komerční banky v GNU/Linuxu lze považovat za funkční a poměrně dobře použitelné. Kdo používá 32bitový Linux a prohlížeč Mozilla Firefox, pravděpodobně nebude muset řešit žádné problémy. Téměř stejně dobře poslouží i Opera, pokud se smíříme s otravnými dialogy. U Konqueroru je to mírně složitější, ale jen poprvé – při dalším použití je to již stejné jako u Firefoxu.

Co zatím nelze označit za uspokojivé, je nativní podpora na 64bitovém Linuxu. Zde v tuto chvíli připadá v úvahu jen Konqueror, brzy možná přibude i Opera. Uvidíme. Na použití bankovníctví KB však může

naprosto zapomenout ten, kdo by chtěl používat čipovou kartu (místo autorizačních SMS) – ta v Linuxu podporována není a nejspíš hned tak nebude.

Obecně se tedy dá říct, že kdo chce nyní používat službu MojeBanka pod Linuxem, téměř nic mu v tom nebrání. Zjištěné nedostatky jsou jen zanedbatelné. Problémy na 64 bitech jsou způsobeny závažnými příčinami ležícími jinde a jejich vyřešení by pomohlo i v mnoha jiných případech. To už je ale úplně jiné téma.

■

Regulární výrazy

Regulární výraz (regular expression, dále jen regexp) slouží k vyhledání části řetězce, kterou předem (úplně) neznáme nebo která může mít více podob. Používá se v programovacích a skriptovacích jazycích.

David Watzke

Obecné regulární výrazy

Mezi obecné regexpy, které podporují všechny možné implementace (Bash, Perl, grep, GNU sed, GNU awk, glibc /regex.h/, atd.), patří množiny znaků. Ty reprezentují jeden znak a používají se pro označení části řetězce, která může mít různé podoby (různé znaky v určité části). Zapisují se mezi hranaté závorky.

```
b[au]f # Odpovídá řetězci baf nebo buf.

[a-zA-Z] # Množina rozsahu znaků. Odpovídá jednomu výskytu písmene
# A až Z (malá i velká písmena,
# pouze bez diakritiky).

[:alpha:] # Totéž, ale zahrnuje i písmena s háčky a čárkami
# (pokud jsou nastavené české locales).

[:alnum:] # Jako předchozí, ale navíc zahrnuje číslice.

[aeiouy] # Množina malých písmen (a, e, i, o, u, y).

[0-9] # Množina všech číslic.
[:digit:] # Jiný zápis téhož.

[123] # Množina číslic 1, 2 a 3.

[:space:] # Množina "whitespace" znaků

. # Množina všech znaků. Odpovídá jednomu znaku.
\. # Odpovídá tečce.
\\ # Odpovídá zpětnému lomítku.
\* # Odpovídá hvězdičce.
```

Přidáte-li za výraz (např. množinu či znak) hvězdičku, změní se jeho význam. Místo jednoho výskytu se bude hledat libovolný počet výskytů; žádný až nekonečno. Například `a*` odpovídá libovolnému počtu písmen `a` za sebou. Výraz `.*` odpovídá úplně všem řetězcům.

Množiny znaků lze přidáním `^` na začátek negovat tak, že znaky v nich obsažené se v daném řetězci nesmějí vyskytovat. Přidáte-li znak `^` jinam, než na začátek, stane se prostým znakem z množiny.

Regexp vždy odpovídá své nejdelsí možné variantě v řetězci (v Perlu toto chování lze ovlivnit, vizte níže), takže použijete-li například o odstavci výše zmíněný výraz `.*` a ihned za něj dáte například `[0-9]*` (libovolný počet číslic), tak si všimněte, že výraz `.*` využil své moci a označil vše, co jen mohl a na množinu `[0-9]*` nezbylo nic, místo toho aby se `.*` při prvním výskytu číslice zastavil, jak by se někdo mohl mylně domnívat. Zmiňuji se o tom zde proto, že tyto znegované množiny znaků jsou univerzálním

řešením tohoto problému. Použijete-li místo původního výrazu `.*[0-9]*` výraz `^[^0-9]*[0-9]*`, tak výraz `[0-9]*` označí celé první číslo, na které narazí, jelikož `[^0-9]*` se před číslicí zastaví. Pro lepší pochopení jsem níže (do části článku o GNU sedu) přidal dva ilustrující příkazy, ale nepředbíhejme. Následuje ukázka znegovaných množin znaků:

```
[^0-9]      # Odpovídá jednomu znaku, který není číslice.
[^[:digit:]] # Jiný zápis téhož.
```

Dále mezi obecné regexpy patří symboly začátku a konce řádku, které mají ovšem tento význam jen tehdy, jsou-li správně umístěny (začátek na začátku a konec na konci výrazu).

```
^      # Začátek řetězce.
$      # Konec řetězce.
```

Několik ukázek obecných regexpů:

```
^$      # Odpovídá prázdnému řetězci (začátek a hned konec).

^a.     # Odpovídá řetězci, který začíná písmenem 'a'
        # a je následován jedním libovolným znakem.

^[^0-9]* # Odpovídá libovolnému počtu nečíselných znaků
        # od začátku řetězce

b$      # Odpovídá řetězci, který končí písmenem 'b'.

[0-9].[:alpha:] # Odpovídá řetězci, který obsahuje číslici,
                # poté libovolný znak a za ním písmeno.
```

Rozšířené regulární výrazy

Rozšířené (*extended*) regexpy značně rozšiřují možnosti těch obecných. Často je třeba je explicitně povolit. Bash je nepodporuje, Perl a GNU `awk` je používají běžně, `grep` potřebuje přepínač `-E` (příp. `-P` pro použití perlových regexpů; tento přepínač je dostupný, jen pokud byl `grep` zkompileován s podporou libpcre), GNU `sed` přepínač `-r` a při použití `regex.h` v C/C++ musíte předat funkci `regcomp()` flag `REG_EXTENDED`.

Rozšířené jsou možnosti pro zachycení řetězce, ve kterém se určité části několikrát za sebou opakují. Tyto znaky se zapíší za určitou část regexpu.

```
?      # Volitelný výskyt (jeden nebo žádný).
+      # Jeden a více výskytů.
{n}    # Za n se dosadí požadovaný počet výskytů.
        # Toto nepodporuje GNU awk.
{n,}   # Odpovídá n a více výskytům.
{n,m}  # Odpovídá n až m výskytům.
```

Například:

```
a?     # Jedno nebo žádné písmeno a.
[ao]+  # Jeden nebo více výskytů písmen z množiny.
[^ ]{3,5} # Tři až pět znaků různých od mezery.
[0-9]* # Libovolný počet číslic.
```

Kromě množin znaků jsou dostupné i množiny řetězců resp. výrazů, které se zapisují do kulatých závorek a oddělují pomocí znaku `|` (pipe).

```
(str|pwd)    # Odpovídá řetězci str nebo pwd.
(foo)+      # Jeden a více výskytů řetězce foo.
(foo|bar)*  # Libovolný počet řetězců foo nebo bar.
(gentoo){3,} # Odpovídá třem a více výskytům řetězce gentoo.
((li){2}e)? # Volitelný výskyt řetězce lilie.
```

Jak jsem naznačil, množina řetězců není úplně správný název, protože zrovna tak může jít o regexpy. Následující množina je složená ze dvou výrazů. Hned za množinou následuje otazník, který znamená, že výskyt všech výrazů z množiny je volitelný. První výraz značí jeden nebo více výskytů znaků *a* a *b* za sebou a druhý výraz odpovídá pěti libovolným číslicím.

```
([ab]+|[0-9]{5})?
```

Pokud chcete regexp odpovídající například sudému počtu řetězců, lze použít toto:

```
^((linux){2})*$
```

Vysvětlení: Máme regexp na označení dvou výskytů řetězce *linux* za sebou. Když se celý výraz obalí dalšími závorkami, hvězdička za ním zajistí označení též čtyř, šesti, osmi, deseti (atd.) výskytů. Pokud chcete minimálně dva výskyty, použijte místo hvězdičky plus.

GNU sed

Kromě toho, že `sed` podporuje nahrazování řetězců odpovídajících regexpům, má zajímavou vlastnost, a tou je podpora referencí. Když si regexpem vyberete určitou část řetězce a chcete ji použít i v nové (nahrazující) části, máte možnost použít příkazy podobné následujícím. Předávám `sed` u řetězec přes `echo`, abyste si to pro lepší představu mohli rovnou vyzkoušet.

```
# Ukázka výše (u znegovaných množin) zmiňovaného problému.
# Na výraz [0-9]+ zbude jen poslední číslo, tedy 1.
echo retezec4241 | sed -r 's/.*([0-9]+)/\1/'
# ... a ukázka řešení pomocí znegované množiny.
# Výraz [^0-9]* se před první číslicí zastaví; vypíše se 4241.
echo retezec4241 | sed -r 's/[^0-9]*([0-9]+)/\1/'

# Výraz začínající číslicí bude vložen mezi hvězdičky.
echo "nahodne cislo: 11.256 :-)" | sed -r 's/([0-9][^ ]+)*\1*/'
```

```
# Je možné použít i více referencí. Zkusme třeba označit
# řetězec "lol" s libovolným počtem 'o' a za ním nějaké číslo.
echo "nejdriv loool a pak cislo 123 ..." | \
sed -r 's:.*(lo+1)[^0-9]+([0-9]+).*:\\1 je \\1\\n\\2 je \\2:'
```

Za reference jsou považovány pouze regexpy v kulatých závorkách, tedy množiny výrazů.

Perl

Perl	standardní
<code>\d</code>	<code>[[[:digit:]]</code>
<code>\D</code>	<code>[^[:digit:]]</code>
<code>\w</code>	<code>[[[:alnum:]]</code>
<code>\W</code>	<code>[^[:alnum:]]</code>
<code>\s</code>	<code>[[[:space:]]</code>
<code>\S</code>	<code>[^[:space:]]</code>

Perlové regexpy obsahují několik vlastních rozšíření. Jedním z nich jsou množiny. Tyto zkrácené zápisy množin se používají úplně stejně jako ty běžné; `\d+` – ekvivalent `[0-9]+` či `[[[:digit:]]+`, `\s{3,}` – tři a více whitespace nebo třeba `\w*` – libovolný počet znaků, které se nevyskytují v množině `[[[:alnum:]]`. Pokud chcete, aby Perl do množin, jako je `\w`, zahrnoval i písmena s diakritikou, kromě českých locales je třeba v programu nastavit jejich použití: `use locale;`

Perl poskytuje možnost kontroly nad tím, zda se bude označovat nejkratší nebo nejdelší možná varianta výrazu. Pokud chceme tu nejkratší, za daný výraz přidáme otazník; nejdelší je výchozí chování regexpů (nejen perlových).

```
my($string) = "popokatepetl";

# označí (a přiřadí do $1) celý řetězec
$string =~ m/(p.*[oel])/;

# označí řetězec "po"
$string =~ m/(p.*?[oel])/;
# totéž, tentokrát obecně
$string =~ m/(p[^\oel]*[oel])/;
```

Toto zdaleka není vše, co Perl s regulárními výrazy umí – popsat vše by vydalo na dost dlouhý seriál. Jako zdroj dalších informací můžete použít například manuálovou stránku [perlre\(1\)](#) ⁽¹⁵⁵⁾.

Regulární výrazy v praxi

Asi neuškodí, když si ukážeme nějaké to využití regulárních výrazů v praxi.

```
# Vypíše ze souboru (/etc/fstab) unixové cesty bez mezer.
egrep -o '(/[^\ ]*)*' /etc/fstab

# Vypíše názvy skupin v systému.
egrep -o '^[^:]*' /etc/group
sed 's/.*//' /etc/group

# Rozparsuje z unixové cesty adresář a soubor.
echo /etc/fstab | sed -r 's:(.*)/(.*):adresář "\1", soubor "\2":'

# Získá tagy z Ogg a přiřadí je do proměnných ($ALBUM, $ARTIST, $TITLE).
eval $(ogginfo Floor-Fee-La.ogg | sed -r -e \
  '/^[[[:space:]]+(TITLE|ARTIST|ALBUM)=/!d' -e 's:=":" -e 's:$:"')'
```

Šikovnou pomůckou je i stránka s ukázkami [Užitečné jednořádkové skripty pro sed](#) ⁽¹⁵⁶⁾. Regexpy se jeví složitější, než ve skutečnosti jsou. Ovšem až na ty perlové, ty složité opravdu jsou :-). Po nějakém čase už píšete rozšířené regexpy bez velkého přemýšlení. Svoje zkušenosti můžete prověřit třeba interaktivními testy na [lexmasterclass.com](#) ⁽¹⁵⁷⁾.

Zpravodaj o Víně – 339

Dostane se Wine do Mac OS X Leopard? Statistika autorů za uplynulé roky. Audit Bugzilly. Konec projektu Alky. Podpora Run/RunOnce. Změny v AppDB: Změny od správců programů; Změny od uživatelů.

Luboš Doležel

Poznámka redakce: Dneškem počínaje vám budeme přinášet překlady „Wine Weekly Newsletter“. Dozvíte se tu o novinkách ve vývoji Wine, nově podporovaných aplikacích a dalších zajímavostech, které se projektu Wine týkají.

Posledním vydáním Wine je [verze 0.9.53](#) ⁽¹⁵⁸⁾ z 11. ledna 2008.

- Položky RunOnce a Run jsou nyní spouštěny při startu.
- Pracuje se na podpoře emulovaných diskových zařízení.
- Došlo k mnoha vylepšením widgetu Richedit.
- Dialog pro výběr barev vypadá lépe.
- Spousta různých oprav.

Roderick Colenbrander (aka Thunderbird) přidal do wiki [stránku o DirectDraw](#) ⁽¹⁵⁹⁾ a souvisejících klíčích v registrech. Těm, kteří si chtějí s podobným nastavením hrát, se doporučuje, aby si tuto stránku přečetli, než se začnou ptát na #winehq.

Dostane se Wine do Mac OS X Leopard?

Seth Weintraub z computerworld.com připravil hromadu [předpovědí](#) ⁽¹⁶⁰⁾ souvisejících s budoucností Leopardu – ta bude představena na blížícím se Mac Worldu. Většina jeho článku s projektem Wine nesouvisí, ale nakonec se dostanete k zajímavé zmnice:

Leopard zvládne nativně spouštět programy pro Windows (ve fázi Beta). Ale ne všechny. Nyní má Leopard vestavěné knihovny Wine. Můžete hrát některé hry, používat IE7, Outlook 2007 – součást Leopardu 10.5.2.

Kdyby se to splnilo, bylo by to určitě zajímavé.

Statistika autorů za uplynulé roky

Ve starším čísle Wine Weekly Newsletteru vyšel přehled počtu commitů do Wine za jednotlivé roky. Tentokrát připravil Michael Stefaniuc podobnou statistiku s počty autorů:

```
$ for y in `seq 2002 2007`; do
  echo -n "Number of authors in $y: "
  git shortlog -s --since="$y-01-01 00:00" --until="$y-12-31 24:00" | wc -l
done
```

```
Number of authors in 2002: 185
Number of authors in 2003: 167
Number of authors in 2004: 183
Number of authors in 2005: 212
Number of authors in 2006: 195
Number of authors in 2007: 218
```

Audit Bugzilly

Před několika týdny napsal Vijay Kiran Kamuju zprávu, ve které doporučuje, aby bylo provedeno nějaké pročištění Bugzilly.

Jak jsme minulý rok udělali audit všech nahlášených chyb v Bugzille, myslím si, že letos bychom měli udělat to samé. Množství nahlášených chyb hodně roste a starší chyby se nechávají ležet ladem. Jen jsem chtěl říct, že bychom měli projít všechny staré otevřené chyby a ověřit je/zeptat se uživatele na stav. Pokud uživatel neodpoví, uzavřeli bychom chybu jako opuštěnou. Zároveň program sami vyzkoušejte na poslední verzi z [gitu](#). Tohle je třeba udělat a potřebujeme na to dobrovolníky. Potřebujeme dobrovolníky na pročištění bordelu (starých opuštěných chyb) v Bugzille.

Dan Kegel se ozval se informacemi o stavu tohoto auditu:

Od té doby, co Vijay před dvěma týdny navrhl, abychom udělali audit nahlášených chyb, se množství otevřených chyb snížilo⁽¹⁶¹⁾ o nějakých 300 kusů, tedy o 10 %. To je celkem solidní kus práce!

Pořád zbývá 1800 chyb, na které nikdo za posledních 90 dnů nesáhnul (137 z nich jsem bohužel nahlásil já), takže nás čeká ještě spousta ověřování a testování.

Dan Kegel také připravil [graf](#)⁽¹⁶²⁾ ukazující vývoj počtu nahlášených chyb.

Konec projektu Alky

Projekt Alky⁽¹⁶³⁾ nedávno skončil⁽¹⁶⁴⁾. Jednou z důležitých věcí je to, co hodlají udělat s kódem, který stihli napsat:

Nicméně, každý konec znamená nové příležitosti, a ačkoliv s lítostí oznamujeme náš odchod, jsme stejně nadšení vám sdělit okamžitou dostupnost VEŠKERÉHO zdrojového kódu projektu Alky!

Toto pochopitelně rozvířilo diskuzi na wine-devel. Stefan Dösinger vysvětlil, že většina kódu bohužel nebude Wine k ničemu dobrá.

Když se dívám na jejich implementaci D3D10, musím říct, že je to srovnatelné s Andreasovým kódem z Google SoC s tím rozdílem, že Alky má pár řádek kódu pro vytvoření kontextu GL a odesílání vertextů, ale Andreas zase napsal hlavičkový soubor D3D10.idl. Přidání kódu z Alky do Andreasova kódu je otázkou hodin, pokud to uděláme rychlým a nečistým způsobem. Celkově vzato si myslím, že podpora D3D10, kterou Wine má a zatím neintegrovalo, je pokročilejší.

Pokud jde o ostatní knihovny, tak tam toho celkem není moc k vidění [...]

I když se nám ten kód nemusí k ničemu hodit (nebo komukoliv jinému, protože je tu přece Wine), vypadá to, že Cody napsal hodně kódu. Nemusí to být porovnatelné s Wine, do kterého byla vložena práce téměř tisíce vývojářů a je skoro 15 let staré, nicméně je to docela úspěch na to, že je to projekt stvořený jedním (nebo dvěma?) vývojáři. (Každopádně to nemohu porovnat se začátky Wine. Tehdy jsem měl jen Gameboye :-))

Podpora Run/RunOnce

Alexandre Julliard před nedávnem [commitnul](#)⁽¹⁶⁵⁾ [patch](#)⁽¹⁶⁶⁾, který zajišťuje, že wineboot je spuštěno při startování nového procesu Wine. Jedním z hlavních následků této změny je skutečnost, že spolu s Wine jsou spuštěny programy z Run a RunOnce.

Co je nového v této verzi:

- Položky RunOnce a Run jsou nyní spouštěny při startu

Tak tohle asi bude mít nějaké následky ;-)

Chtěl jsem to napsat už dřív, ale zapomněl jsem na to. V mé velké instalaci Wine se nyní spouští:

- 2 instance červa pro Windows
- Google Talk (nefunkční)

- Skype (havarující)
- pccam.exe z nějakého CD od digitálního fotoaparátu
- obtěžující aktualizace Steam.exe a dialog požadující koupi předplatného
- snad ještě další věci

a tohle vše dokonce i při `make check` :(

Asi teď budeme potřebovat nějaký nástroj pro úpravu položek Run a RunOnce.

Většina se shodla na tom, že je třeba nějaký nástroj pro řízení těchto věcí. Bylo navrženo několik možností:

Steven Edwards: Asi je třeba nějaký nástroj jako msconfig pro správu služeb, nebo nějaká stránka ve winecfg.

Vincent Povirk: Startup Control Panel⁽¹⁶⁷⁾ od Mika Lina mi na Windows připadá docela užitečný. Vypadá to, že pod Wine funguje, i když to vypadá, že nemám žádné položky v Run a RunOnce, takže si nemohu být jistý.

Brian Vincent: Dokud někdo nepřijde s rozšířením winecfg pro úpravu klíčů Run a RunOnce (a opravdu to vypadá, že to potřebujeme), měli bychom pravděpodobně detekovat změny v těchto klíčích a při dalším spuštění wineboot ukázat dialog s textem: „Klíče registru Run a RunOnce se změnil, asi kvůli tomu, že jste nainstaloval nový program. Nyní budou tyto klíče spuštěny, takže se připravte na to, že něco exploduje a ukradne vám kapesné. Problémy můžete vyřešit spuštěním „wine regedit“ a úpravou HKLM\Software\Microsoft\Windows\CurrentVersion\Run.“

Pak se objevila zajímavá diskuze o msconfig z ReactOS:

Steven Edwards: Winecfg opravdu není vhodným místem pro takovou věc. Lidé zběhlí v používání Windows očekávají msconfig... je škoda, že si nemůžeme vzít ten z ReactOS, protože mají funkční náhradu pod `◇LGPL`⁽¹⁶⁸⁾.

Austin English: Proč ne? Ono to je to snad dílo zpětného inženýrství? Nebo je tu snad něco jiného, co brání integraci?

Steven Edwards: Naším všeobecným postojem je, že z ReactOS už kód nepřijímáme, i když taskmgr a editor registru byly vyvinuty vývojáři ReactOS. Je docela zjevné, že je to čistě napsaný kus kódu, takže možná by Alexandre udělal výjimku a kód přijal. Tady vážně není důvod, proč by se cokoli dělalo zpětným inženýrstvím, a Colinova verze je pod vhodnou licencí, takže nevidím žádný důvod, proč by to mělo být zamítnuto.

Změny v AppDB

Změny od správců programů

Program	Původní stav (verze)	Nový stav (verze)	Změna
Wizardry 8 1.2.4	Silver (0.9.52)	Gold (0.9.53)	+1
Babylon-Pro Babylon Pro 6	Bronze (0.9.22)	Silver (0.9.52)	+1
Dungeon Siege II 2.x	Bronze (0.9.47)	Gold (0.9.53)	+2
Touhou Kaeizuka ~ Phantasmaqoria of Flower View 1....	Garbage (0.9.51)	Silver (0.9.53)	+2
Super EF2000 1.0	Gold (0.9.46)	Platinum (0.9.53)	+1
IDA Pro 5.2.0.908	Platinum (0.9.50)	Silver (0.9.53)	-2
Europa 1400: The Guild Gold: 1.05 Beta 3	Silver (0.9.42)	Bronze (0.9.52)	-1
Star Wars: Jedi Knight - Mysteries of the Sith 1.0	Garbage (0.9.52)	Gold (0.9.53)	+3
Leisure Suit Larry: Love for Sail! 7, Love For Sai...	Platinum (0.9.36)	Gold (0.9.47)	-1
Lord of the Rings Online: Shadows of Angmar 1.05	Silver (0.9.51)	Gold (0.9.52)	+1
Widelands 11.2	Gold (0.9.44)	Silver (0.9.53)	-1
jetAudio JetAudio 7.0	Garbage (0.9.46)	Gold (0.9.52)	+3
Star Trek: Away Team 1.0	Platinum (0.9.49)	Gold (0.9.52)	-1
Star Wars: Jedi Knight - Dark Forces II 1.01	Garbage (0.9.52)	Silver (0.9.53)	+2
Star Wars: Jedi Knight - Dark Forces II 1.0	Garbage (0.9.52)	Silver (0.9.53)	+2
Microsoft Money 2003	Gold (0.9.36)	Platinum (0.9.53)	+1
The Lord of the Rings: The Battle for Middle-Earth...	Bronze (0.9.52)	Silver (0.9.53)	+1
Cellková změna			+14

Změny od uživatelů

Program	Původní stav (verze)	Nový stav (verze)	Změna
FlatOut 1.0	Silver (0.9.35)	Platinum (0.9.52)	+2
StarCraft 1.x	Garbage (0.9.50)	Silver (0.9.52)	+2
Half-Life 2 Episode 2	Bronze (0.9.47)	Gold (0.9.52)	+2
The Witcher US demo	Garbage (0.9.52)	Bronze (0.9.53)	+1
Supreme Commander: Forged Alliance 1.0	Bronze (0.9.51)	Garbage (0.9.53)	-1
S.T.A.L.K.E.R. : Shadow of Chernobyl 1.x	Silver (0.9.51)	Gold (0.9.52)	+1
SDL TRADOS 2007 SDL Trados 2007 (Professional)	Garbage (0.9.49)	Bronze (0.9.53)	+1
Battlefield 1942 1.x	Silver (0.9.49)	Platinum (0.9.52)	+2
StarCraft Brood War: 1.x	Silver (0.9.47)	Gold (0.9.52)	+1
Guild Wars All Versions	Gold (0.9.52)	Silver (0.9.53)	-1
Age of Empires II The Age of Kings: 2.x	Silver (0.9.50)	Garbage (0.9.52)	-2
Half-Life 2 Retail (32-bit)	Platinum (0.9.50)	Gold (0.9.53)	-1
Half-Life 2 Episode One: Retail	Silver (0.9.50)	Gold (0.9.52)	+1
Dreamweaver 8.x	Platinum (0.9.46)	Silver (0.9.48)	-2
Gothic 1.0x	Bronze (0.9.52)	Gold (0.9.53)	+2
EasyUO 1.50 [Build 0118] Beta	Gold (0.9.39)	Silver (0.9.52)	-1
Silkroad Online 1.x	Platinum (0.9.51)	Gold (0.9.52)	-1
EAC - Exact Audio Copy 0.99 prebeta 3	Gold (0.9.49)	Silver (0.9.53)	-1
The Longest Journey Build 161	Bronze (0.9.49)	Garbage (0.9.52)	-1
Steam All Versions	Gold (0.9.51)	Silver (0.9.52)	-1
The Nations 1.0	Bronze (0.9.20)	Silver (0.9.49)	+1
Cellková změna			+4

Tento text je distribuován v souladu s podmínkami [GNU GPL](#) ⁽¹⁶⁹⁾.

Jaderné noviny – 12. 12. 2007

Aktuální verze jádra: 2.6.24-rc5. Citáty týdne: Ingo Molnár, Ted T'so, Linus Torvalds. Jednodušší syslety. Přiškrcování zápisu nečistých stránek na disk [writeout]. Nové a staré chyby.

Robert Krátký

Aktuální verze jádra: 2.6.24-rc5

Aktuální předverze je (k 12. 12. 2007) 2.6.24-rc5, vydaná⁽¹⁷⁰⁾ 10. listopadu. Linus Torvalds⁽¹⁷¹⁾ k tomu napsal:

Věci se zpomalily, i když bych samozřejmě lhal, kdybych tvrdil, že jsme se postarali o všechny regrese a že je máme pod kontrolou. Ale pracuje se na tom a seznam se krátí. Pokud bych si však měl tipnout, tak před Vánoci 2.6.24 určitě nevyjde – pokud nám Santa nepošle pár elfů, aby s těmi regresemi pomohli.

Seznam oprav je pořád docela dlouhý; byla také doplněna velká aktualizace FireWire stacku. Krátký changelog je součástí Linusova mailu; všechny podrobnosti najdete v [dlouhém](#)⁽¹⁷²⁾. Od vydání -rc5 se do hlavního git repozitáře dostala jen hrstka patchů.

Citáty týdne: Ingo Molnár, Ted T'so, Linus Torvalds

Nechci předbíhat, aniž bych si nejprve prohlédl nějaké profily, ale myslím, že ta regrese týkající se výkonu SLUBu⁽¹⁷³⁾ dává najevo, že následující je omyl: „SLAB může být implementován mnohem jednodušeji při zachování stejného výkonu“. U SLABu nebylo nic, co bych mohl označit za „nepotřebný kód“.

– Ingo Molnár⁽¹⁷⁴⁾

Předpokládám, že pokud by NSA měla ve sklepě na plný výkon 20 000 2Ghz procesorů po deset let, tak 50 % času *po té*, co by crackli stav random poolu, by mohli získat posledních 80 bitů generovaných z /dev/random. Ale pokud už uvažujeme o takové síle, která by dokázala získat pool i add_ptr, tak se mi zdá, že by se s tím dalo udělat spousta užitečnějších věcí – třeba vymyslet trojského koně pro ten software a sebrat přímo soukromý klíč.

– Ted Ts'o⁽¹⁷⁵⁾

Nic mě nezastaví. Na m0j3 c00l h4ck3rské sch0pn0s1i nikd0 n3má.

– Linus Torvalds⁽¹⁷⁶⁾

Jednodušší syslety

Syslety⁽¹⁷⁷⁾ jsou navrhovaný mechanismus, který by umožnil, aby bylo kterékoli systémové volání spuštěno asynchronně; tato technika slibuje ucelenější a jednodušší asynchronní I/O mechanismus a ještě mnohem více – jakmile by se podařilo vyřešit všechny ty nepříjemné drobnosti. Před nedávnem převzal vývoj sysletů Zach Brown⁽¹⁷⁸⁾ a od té doby bylo v té oblasti poměrně ticho. Teď však Zach přišel s novou verzí patche⁽¹⁷⁹⁾, která ukazuje, kam se tato myšlenka ubírá.

Tato verze odstraňuje většinu funkčnosti, kterou nabízely předchozí verze. Možnost nahrávat do jádra jednoduché programy pro asynchronní spouštění je pryč, stejně jako mechanismus threadletů pro asynchronní spouštění uživatelských funkcí. Místo toho se syslety vrátily ke kořenům: mechanismus pro spuštění jediného systémového volání bez blokování.

Jak už bylo naznačeno v předchozích diskuzích, syslety teď využívají mechanismus systémového volání `indirect()`⁽¹⁸⁰⁾. Aplikace, která chce provést asynchronní systémové volání, vyplní strukturu `syslet_args`,

kteřá popisuje, jak má být asynchronní spuštění řešeno; potom aplikace zavolá `indirect()`, které to provede. Pokud může systémové volání proběhnout bez blokování, `indirect()` prostě vrátí konečný stav. Je-li zapotřebí blokování, tak se jádro (stejně jako u předchozích verzí patche) vrátí v samostatném procesu do uživatelského prostoru, zatímco původní proces čeká na dokončení. Po dokončení je koncový stav uložen v uživatelské paměti a aplikace je na to zajímavým způsobem upozorněna (vizte dále). Struktura `syslet_args` vypadá takto:

```
struct syslet_args {
    u64 completion_ring_ptr;
    u64 caller_data;
    struct syslet_frame frame;
};
```

Pole `completion_ring_pointer` obsahuje ukazatel na kruhový buffer uložený v uživatelském prostoru. Hlavička bufferu je definována takto:

```
struct syslet_ring {
    u32 kernel_head;
    u32 user_tail;
    u32 elements;
    u32 wait_group;
    struct syslet_completion comp[0];
};
```

`kernel_head` je index další položky v kruhovém bufferu označující dokončení [next completion ring entry], kterou má jádro vyplnit, a `user_tail` je další položka, která bude předložena aplikaci. Pokud jsou ty dvě položky stejné, je kruh prázdný. Pole `elements` říká, kolik položek může být v kruhu uloženo; musí jít o mocninu dvou. Jádro používá `wait_group` jako interní způsob lokalizace čekací fronty, když aplikace čeká na dokončení sysletu; [Jonathan Corbet](#) ⁽¹⁸¹⁾ odhaduje, že tato část API do finální verze nepřežije.

A konečně vlastní hodnoty koncových stavů jsou uloženy v poli struktur `syslet_completion`, které vypadá takto:

```
struct syslet_completion {
    u64 stav;
    u64 caller_data;
};
```

Po dokončení sysletu je koncový návratový kód uložen do `stav`, kdežto pole `caller_data` je nastaveno na hodnotu poskytnutou polem stejného jména ve struktuře `syslet_args`, když provádění sysletu začalo.

Zbývá jedno pole `syslet_args`, o kterém se ještě nemluvílo: `frame`. Definice této struktury se liší na jednotlivých architekturách; pro x86 vypadá takto:

```
struct syslet_frame {
    u64 ip;
    u64 sp;
};
```

Tyto hodnoty jsou použity, když syslet skončí. Po té, co jádro uloží koncový stav v kruhovém bufferu, zavolá funkci, jejíž adresa je uložena v `ip`, a to pomocí stackového ukazatele, který je v `sp`. Toto volání slouží jako jistý druh okamžitého asynchronního upozornění pro aplikaci, že syslet už skončil. Stojí za zmínku, že toto volání je prováděno v rámci původního procesu – toho, ve kterém byl spuštěn syslet –

místo v tom novém, který je použit pro návrat do uživatelského prostoru, když se syslet blokuje. Tato funkce nemá, co by vracela, a proto by po splnění úkolu měla prostě skončit.

Takže, abychom si zopakovali, jak budou aplikace používat syslety k provádění asynchronních systémových volání:

- V uživatelském prostoru je založen a inicializován „kruh dokončení“ [completion ring].
- Pro upozorňovací funkci je alokován stack a struktura `syslet_args` je vyplněna relevantními informacemi.
- Zavolá se `indirect()`, čímž se syslet nastartuje.
- Pokud je dané systémové volání schopno dokončit bez blokování, předá se z `indirect()` návratová hodnota rovnou zpátky do uživatelského prostoru a je hotovo.
- Pokud se systémové volání blokuje, přepne se provádění do nového procesu, který se vrátí do uživatelského prostoru. V takovém případě je vrácena chyba `ESYSLETPENDING`.
- Jakmile je systémové volání dokončeno, uloží jádro návratovou hodnotu do kruhu dokončení a zavolá v původním procesu upozorňovací funkci.

Pokud by si aplikace přála zastavit a počkat na dokončení všech rozběhnutých sysletů, může využít nové systémové volání:

```
int syslet_ring_wait(struct syslet_ring *ring, unsigned long user_idx);
```

`ring` je ukazatel na kruh dokončení a `user_idx` je hodnota indexu `user_tail`, jak ji vidí proces. Předání `user_tail` jako parametru do `syslet_ring_wait()` předchází problémům se souběhem, které by mohly nastat, kdyby syslet skončil poté, co se aplikace rozhodla čekat. Toto volání skončí ve chvíli, kdy je v kruhu alespoň jedno dokončení.

Skutečným účelem této sady patchů je pokus o stanovení uživatelského API sysletů; je zřejmé, že ještě zbývá dost práce. Například v tuto chvíli není možné, aby aplikace použila `indirect()` k současnému spuštění sysletu a předání dalších parametrů cílovému systémovému volání (což byl původní účel `indirect()`). Ono je vůbec velmi obtížné rozlišit, která z těch dvou akcí vlastně probíhá. Zach už zmínil, že volací konvenci bude potřeba změnit, aby se jasně vymezilo, kdy se používají syslety a kdy jde o přidávání parametrů.

Navíc je ještě potřeba nalézt odpovědi na několik dalších otázek – Zach jich pár načrtl už při představení patche. Není jasné, jak bude fungovat spolupráce s `ptrace()`, je dost problémů se správou zdrojů a tak dále. Zach by byl zjevně rád za nějaké podněty ohledně těchto věcí:

Především bych rád slyšel od lidí, kteří se snaží syslety používat ve svých aplikacích. Bude to zatím vyžadovat nechutné berličky místo glibc volání a váš počítač u toho možná vybuchne, ale snad vám bude možnost ovlivnit podobu sysletů za tu námahu stát.

Takže pokud vás zajímá, jak bude toto rozhraní vypadat, mějte teď oči otevřené.

Příškrcování zápisu nečistých stránek na disk [writeout]

Předcházení zatuhávání při zápisu nečistých stránek na disk [writeout deadlocks] je téma, které se čas od času v konferencích objeví. Většina linuxových systémů si s tím umí poradit bez větších problémů. Občas se však systém může dostat do stavu, kdy nemá paměť a musí nějaké stránky zapsat na disk, než je možné další paměť alokovat. Pokud však samotný akt zápisu stránek na disk vyžaduje alokaci paměti, může systém zatuhnout. Nejnáchylnější jsou systémy, které používají komplikované blokové I/O – mapovač zařízení, síťová úložiště, souborové systémy v uživatelském prostoru atd.

Tento problém se už snažilo řešit docela dost patchů; jedním z nich byl i patch pro [chytřejší příškrcování zápisu](#) ⁽¹⁸²⁾, o kterém jsme psali v srpnu. Jde však o těžko řešitelný problém; vypadá to, že nás bude doprovázet ještě dlouho. Pokud se ovšem neosvědčí nový a dosti agresivně prosazovaný patch, který poslal Daniel Phillips: [příškrcování zápisu nečistých stránek](#) ⁽¹⁸³⁾.

Danielův patch je v jádru docela jednoduchý. Jeho přístup k zamezování zatuhávání spojených se zápisem nečistých stránek je následující:

- Založit paměťovou rezervu, ze které smí stránky alokovat pouze kód, který provádí zápis stránek. Tato rezerva už vlastně existuje, protože máme rezervovanou paměť pro procesy označené příznakem `PF_MEMALLOC`.
- Stanovit limit na množství paměti, které je možno využít pro zápis stránek na každé zařízení.

Patch se nesnaží přímo sledovat množství paměti, které bude použito pro každý požadavek na zapsání stránek; místo toho ukládá ovladačům na blokové úrovni, aby počítaly, kolik bude využito „jednotek“. K tomuto účelu přidává do každé fronty požadavků `atomic_t` proměnnou (nazývanou `available`) a ukazatel na funkci (`metric()`). Když se odchozí požadavek dostane do `__generic_make_request()`, je předán na `metric()`, aby se získal odhad objemu zdrojů potřebných pro zpracování daného požadavku. Přesahuje-li odhadovaný objem požadovaných zdrojů hodnotu `available`, proces se blokuje, dokud se nedokončí požadavek a `available` nebude navýšena na dostatečnou úroveň.

Funkci `metric()` by měl poskytovat blokový ovladač nejvyšší úrovně, který má na starosti frontu požadavků. Pokud je sám ovladač zodpovědný za to, aby se data dostala na fyzické médium, bude odhadování požadovaných zdrojů poměrně jednoduché. Potíže se zatuháváním se však objevují, když musí jít I/O požadavky přes několik vrstev ovladačů; představte si RAID postavený nad síťovými úložnými zařízeními. V takovém případě bude muset horní úroveň získat odhad z nižších úrovní, což je problém, který tento patch zatím neřeší.

Andrew Morton⁽¹⁸⁴⁾ navrhl alternativní přístup⁽¹⁸⁵⁾, při jehož použití by bylo sledováno využití paměti každého blokového zařízení. Několik háčků [hooks] v kódu pro alokaci stránek by poskytlo rozumný odhad o tom, kolik paměti je v kteroukoliv dobu vyhrazeno pro čekající I/O požadavky; tyto háčky by také mohly být použity k odhadu toho, kolik paměti bude potřebovat každý nový požadavek. Bloková vrstva by pak mohla využít tento odhad a aktuální stav, aby zajistila, že zařízení nepřekročí svůj povolený limit využití paměti. Daniel tento přístup od začátku odmítal⁽¹⁸⁶⁾, protože by podle něho bylo příliš riskantní sledovat aktuální stav využití paměti. Mohlo by se stát, že by nějaké zařízení obsluhovalo I/O požadavky, které by, ještě než budou dokončeny, vyžadovaly více paměti, než bylo doposud alokováno. V takovém případě by nakonec mohla být skutečná spotřeba paměti o hodně vyšší. Proto Daniel tvrdí, že je lepší provést konzervativní výpočet na začátku.

Patch vůbec neřeší otázku paměťové rezervy; místo toho spoléhá na stávající mechanismus `PF_MEMALLOC`. Daniel říká, že bylo nezbytné `PF_MEMALLOC` „upřednostnit“ před některými dalšími procesy, které pomáhají s procesem zápisu stránek, ale nic více. Kromě toho také tvrdí, že pro nejlepší výsledky je nutné z jádra odstranit většinu stávajícího kódu cíleného na zabraňování zatuhávání při zápisu stránek. Shrnuje to takto:

Rád bych to zakončil tím asi úplně nejpodstatnějším: přiložený kód byl vystaven náročnému testování a používání po mnoho měsíců. Není tedy nic teoretického na tom, když říkám, že funguje – a patch, co se týče korektnosti, mluví sám za sebe. Rád bych k tomu v nepříliš vzdálené budoucnosti dodal, že jsme z jádra odstranili stovky řádků kódu a přitom zachovali stabilitu a zlepšili výkon.

Od té doby se v kódu našlo několik problémů, což trochu narušuje auru korektnosti. Ale nikdo nepoukázal na závažný problém s myšlenkou kódu. Zjišťování skutečné účinnosti a příprava pro to, aby fungoval s větším množstvím úložných konfigurací, však bude vyžadovat čas a další úsilí. Pokud se však zadaří, mohlo by se jádro zbavit jednoho z nepříjemných problémů.

Nové a staré chyby

S blížícím se vydáním 2.6.24 roste snaha zbavit se regresí. Jak je vidět na aktuálním seznamu⁽¹⁸⁷⁾ (těsně před vydáním 2.6.24-rc5), pořád ještě zbývá nějaká práce. Seznam je tak dlouhý, že při vydání -rc5 Linus podotkl, že tradiční vánoční vydání tentokrát asi nebude.

Jednou z těchto regresí je to, že s jádrem 2.6.24-rc nefunguje určitý model DVD mechanik; s 2.6.23 fungují bez problémů. Pohled na příslušnou stránku v bugzille⁽¹⁸⁸⁾ ukazuje, že na odhalení tohoto bugu bylo

vynaloženo docela dost úsilí (jak ze strany vývojářů, tak testerů). V době vydání originálu tohoto článku (12. 12. 2007) ještě nebylo jasné, jak problém opravit.

Ví se však, který patch to způsobil. Tejun Heo to [popisuje](#)⁽¹⁸⁹⁾ takto: **Je to kvůli nastavení velikosti přenášeného kusu u ATAPI [ATAPI transfer chunk size] na skutečnou velikost, což je obecně správně.**

Aktuální vývojový kód (určený pro 2.6.25) s tímto zařízením funguje bez potíží, ale to by v této fázi vývojového cyklu byl příliš velký patch pro 2.6.24. Takže Tejun společně s ostatními hledá jednodušší opravu. Má také záložní plán:

Pokud se nám včas nepodaří najít řešení, vždycky tu aktualizaci můžeme stáhnout. Sice kvůli tomu nebudou fungovat některé jiné věci, ale nebude se aspoň jednat o regrese. A opravu pak můžeme začlenit společně s ostatními změnami v 2.6.25.

Tento plán si vysloužil okamžitý [nesouhlas](#)⁽¹⁹⁰⁾ [Alana Coxe](#)⁽¹⁹¹⁾, který připomněl, že kvůli vyřazení této opravy by přestalo fungovat hned několik zařízení, která byla konečně zprovozněna, ale pomohlo by to jen jedinému, o němž se ví, že má s novým kódem problémy. Řekl, že tato změna **...nedává smysl a není obecně ku prospěchu.**

Alan by raději strpěl, že jedno zařízení nebude fungovat, když to znamená, že by pak větší počet jiných zařízení konečně fungovat začal. Pokud by to bylo potřeba, mohla by být ta nefunkční mechanika dána po dobu 2.6.24 na speciální blacklist.

Proti takovému postupu se však [ostře ohradil](#)⁽¹⁹²⁾ Linus:

Taková jedna regrese je pravděpodobně jen špička ledovce. Pokud něco přestane fungovat jednomu testerovi, tak to nejspíš znamená, že je dalších tisíc lidí, kteří vůbec vývojové jádro netestovali, nebo s ním měli problém, a tak se prostě vrátili k předchozí verzi.

Naproti tomu vrácení nějaké změny určitě nebude znamenat takové problémy, protože všimnout by si toho mohli jedině ti, kterým to stejně nikdy nefungovalo. Žádný „tichý dav“ v takovém případě není.

Spolu s narůstající komplexitou jádra (a [obavami o jeho kvalitu](#)⁽¹⁹³⁾) si vývojářská komunita začala dávat na regrese mnohem větší pozor. Jak podotkl Linus, regrese způsobují hmatatelné problémy lidem, jejichž systémy dříve fungovaly; to je dobrá cesta, jak přijít o testery (a nakonec i o uživatele). Na druhou stranu, pokud něco nikdy nefungovalo, a pořád to nefunguje, tak to uživatelům život neztěžuje. Z toho důvodu se potlačování regresí stalo jednou z nejvyšších vývojových priorit.

Existuje ještě jeden příbuzný důvod: zmíněné obavy o kvalitu jádra. Můžeme se ptát, jestli se kvalita jádra zlepšuje, nebo ne, ale opravdová odpověď na tuto otázku se nehledá snadno. Lepší jádro by mohlo přilákat více uživatelů, což by vyústilo ve větší počet hlášení o chybách; stejně tak by mohlo chybovější jádro odradit testery, takže by výsledkem bylo méně hlášených chyb. Nedá se pouze podívat na seznam chyb, o kterých víme, a podle toho posuzovat, jestli je jádro horší nebo lepší než jiné.

Můžeme však dávat pozor na to, aby všechno, co funguje teď, fungovalo i v dalších verzích. Pokud nám věci nepřestávají fungovat, tak za předpokladu, že se čas od času nějaká chyba opraví, můžeme vyvodit, že se jádro zlepšuje. Pokud bychom tolerovali regrese, neměli bychom žádné srovnání. Regrese jsou tedy to nejlepší, co máme (z hlediska objektivního posuzování kvality dané verze jádra). Není tedy divu, že když se kvalitě jádra věnuje více pozornosti, vede to k zaměření na regrese.

Kdokoliv ví něco o práci Alana Coxe, ten také ví, že mu na kvalitě jádra velmi záleží. Tentokrát je však [přesvědčen](#)⁽¹⁹⁴⁾, že tažení proti regresím už zašlo příliš daleko:

Slepě tvrdit, že na regresích tolik záleží, to je někdy (třeba v tomto případě), jako kdybyste tvrdili: „tahle dálnice už není kompatibilní s koňským povozem“, takže tu dálnici zase změním na hliněnou cestu.

Pokud by se pro popisovaný problém nenašlo řešení (pozn.: [už se našlo](#)⁽¹⁹⁵⁾), tak to vypadá, že by asi koně a povozy tentokrát zvítězili. Ti, kdo potřebují pořádnou dálnici, by si museli počkat na verzi kompatibilní s koňmi, která přijde v 2.6.25.

■

Jaderné noviny – 19. 12. 2007

Aktuální verze jádra: 2.6.24-rc5. Citáty týdne: Andrew Morton, Ingo Molnár. Krátké zmínky: kerneloops, read-mostly, prodlevy I/O portů. revoke() se vrací.

Robert Krátký

Aktuální verze jádra: 2.6.24-rc5

Aktuální verze jádra je i nadále 2.6.24-rc5; v minulém týdnu nevyšly žádné nové předverze. Do hlavního repozitáře však i nadále proudí opravy.

Aktuální verze -mm stromu je [2.6.24-rc5-mm1](#)⁽¹⁹⁶⁾. Mezi nedávné změny patří výrazné změny modelu zařízení; kvůli konfliktům patchů bylo z této verze vynecháno dost stromů subsystémů.

Aktuální stabilní jádro řady 2.6 je [2.6.23.10](#) [2.6.23.11](#) [2.6.23.12](#). Velký patch [2.6.23.10](#)⁽¹⁹⁷⁾, který vyšel 14. prosince, obsahoval několik desítek oprav. [2.6.23.11](#)⁽¹⁹⁸⁾ (14. prosince) a [2.6.23.12](#)⁽¹⁹⁹⁾ (18. prosince) obsahují malé opravy problémů způsobených verzí 2.6.23.10.

Starší jádra: [2.6.22.15](#)⁽²⁰⁰⁾ vyšlo 14. prosince se slušnou řádkou oprav.

[2.4.36-rc1](#)⁽²⁰¹⁾ bylo vydáno 17. prosince s několika bezpečnostními opravami. [2.4.35.5](#)⁽²⁰²⁾ obsahuje stejné opravy.

Citáty týdne: Andrew Morton, Ingo Molnár

Pro představu... mám:

- Asi 1 400 otevřených hlášení v bugzille.
- 719 emailů uložených ve složce chybové-hlášení-emailem, které je všechny třeba projít a požádat o další otestování a případné opětovné nahlášení, pokud ještě není chyba opravena.
- Velký a ošklivý email nadepsaný „2.6.24-rc5-git1: hlášené regrese oproti 2.6.23“ přímo v inboxu.

Což všechno znamená, že je trochu nevhodné uvažovat o nových funkcích, které vypadají, že budou mít velký dopad.

Takže mi to celé pošlete aplikovatelné na rc5-mm1 a já to tam vrazím – uvidíme, co přestane fungovat.

– Andrew Morton⁽²⁰³⁾

OK, a vzhledem k časovému posunu a ročnímu období si večer sednu a budu sledovat, jak padá sníh, a spokojeně snít o koťátkách s uzi samopaly s municí s nukleárními hroty, které loví super-elitní vývojáře bezdrátového síťování, aby do nich vtloukli trochu lidskosti a pochopení, OK?

– Ingo Molnár⁽²⁰⁴⁾

Krátké zmínky: kerneloops, read-mostly a port 80

Kerneloops

Významnou součástí práce vývojáře je stanovování priorit [triage]. Tak velký a používaný projekt jako jádro bude mít pořád více hlášení o chybách, než kolika by bylo reálně možné se v dostupném čase věnovat. Vývojáři tedy musí zjistit, které problémy stojí za to, aby jim věnovali svou pozornost. Někdy to rozhodování usnadní přítomnost našťavaného zákazníka. Jindy je však třeba uhádnout, které chyby se dotýkají největšího počtu uživatelů. A to se často pozná podle toho, kolik různých hlášení o daném problému přišlo.

Počítání hlášení však není nic jednoduchého – zvláště pokud nejsou na stejném místě. Ve snaze celý proces zjednodušit, Arjan van de Ven [oznámil](#) ⁽²⁰⁵⁾ nové stránky [kerneloops.org](#) ⁽²⁰⁶⁾. Arjan dal dohromady software, který na určitých stránkách a konferencích vyhledává poslané výstupy jaderných `oops` ⁽²⁰⁷⁾; každý nalezený pád je nacistán do databáze. Pak se program pokusí propojit hlášení pomocí verze jádra a obsahu výpisu [call trace]; z toho lze pak vytvořit seznam nejoblíbenějších způsobů, jak shodit systém. V tuto chvíli to vypadá, že nejvíce v kurzu mezi oopsy vývojových jader je `ieee80211_tx()`. Kromě toho se ukládají i některé další informace; konkrétně je možné zjistit, která nejstarší verze jádra ještě daným oopsem trpí.

Je to nepochybně užitečný zdroj informací, ale je tu ještě pár problémů, které situaci komplikují. Jedním z nich je skutečnost, že konec výpisu nebývá nijak jednoznačně označen, takže skripty nevědí, kde přestat kopírovat text. Druhá potíž je v tom, že vícenásobná hlášení o stejném oopsu mohou uměle navýšit zaznamenávaný počet u daného pádu. Řešením obou problémů je umístění značky na konec výpisu, která by obsahovala náhodné UUID generované při bootu. Patche, které by to zařídily, jsou již v oběhu, i když se ukazuje, že dostat do výpisů to náhodné číslo je trochu obtížnější, než by se dalo čekat. U 2.6.24 může být „náhodné“ číslo složeno ze samých nul, takže vyřešeno to nakonec bude asi až v 2.6.25.

Read-mostly

Když se někdo aspoň chvilku štourá v kódu jádra, tak si všimne mnoha proměnných deklarovaných následujícím způsobem:

```
static int __read_mostly ignore_loglevel;
```

Atribut `__read_mostly` [většinou čteno] říká, že přístupy k dané proměnné jsou většinou (ale ne vždy) operace typu čtení. Nedávno se objevily dotazy o tom, proč se toto označování provádí. Odpověď je, že jde o důležitou optimalizaci, i když ne vždy má takový účinek, v jaký vývojáři doufali.

Jak je popsáno v [What every programmer should know about memory](#) ⁽²⁰⁸⁾, správné používání paměťové keše je pro optimální výkon klíčové. `__read_mostly` seskupuje proměnné, které jsou měněny jen výjimečně, aby mohly sdílet řádky keše [cache lines], které by nemusely na víceprocesorových systémech lítat mezi jednotlivými procesory. Dokud proměnnou označenou jako `__read_mostly` někdo nezmění, může klidně sedět ve sdíleném řádku keše s ostatními takovými proměnnými a být po ruce v keši na všech procesorech systému.

Atribut `read-mostly` obvykle funguje dobře a poskytuje měřitelné zlepšení výkonu. Existují však obavy, že by tato funkce mohla být využívána až příliš. [Andrew Morton](#) ⁽²⁰⁹⁾ to [vyjádřil](#) ⁽²¹⁰⁾ takto:

Takže... až přeměníme všechny většinou čtené proměnné na `__read_mostly`, co nám zbude v `bss`? Všechny často zapisované proměnné. Všechny hezky namačkané pohromadě, aby se maximalizovalo sdílení řádků keše.

Kombinování často zapisovaných proměnných do sdílených řádků keše je dobrý způsob, jak maximalizovat přehazování těchto řádků mezi procesory – což by mělo na výkon špatný vliv. Příliš agresivní segregace `read-mostly` proměnných, aby se minimalizovalo přehazování řádků keše, může mít úplně opačný účinek: výkon jaderné keše by mohl poklesnout.

Podle Andrewa by bylo lepší vytvořit atribut „`read often`“ [často čteno], který by se přiřazoval proměnným často využívaným v režimu pouze pro čtení. To by umožnilo, aby početné málo čtené proměnné vyplnily místo mezi často zapisovanými proměnnými. Zatím však tuto myšlenku nikdo nepřevlel do patche.

Prodlevy I/O portů

Mezi funkcemi, které jádro poskytuje pro přístup k I/O portům, jsou už verze, jež vkládají prodlevy. Ovladač by za normálních okolností přečetl bajt z portu pomocí `inb()`, ale pokud by po operaci bylo potřeba vložit krátkou prodlevu, může se použít `inb_p()`. Pohled na ovladače ve stromu jádra prozradí, že tyto přístupy s prodlevou jsou využívány docela často, i když k tomu v mnoha případech není důvod.

Prodleva je (na architekturách x86) implementována pomocí zápisu na I/O port 80. Na tomto portu většinou žádný hardware nenaslouchá, takže zápis pouze způsobí zdržení procesoru, zatímco se sběrnice marně snaží operaci provést. Tato operace má rozumně definovanou sémantiku a funguje už v Linuxu hodně let.

Až na to, že to teď **vypadá** ⁽²¹¹⁾, jako by tato technika na malé části x86_64 systémů nefungovala. Zápis na port 80 občas způsobí úplné zatuhnutí systému, což má za následek poněkud delší prodlevu, než bylo v plánu. Šlo by si představit komplikovaný mechanismus pro restartování I/O operací po té, co uživatel restartuje systém, ale vývojáři jádra se raději pustili do hledání jiných způsobů, jak implementovat I/O prodlevy.

Skoro ve všech ostatních případech poslouží jako alternativní způsob docílení prodlevy volání `udelay()`. Největší potíž je však v tom, že `udelay()` funguje jako smyčka; nemůže vědět, kolikrát se má smyčkou projít, dokud není kalibrována rychlost procesoru. K této kalibraci dochází poměrně záhy v rámci procesu bootování, ale předtím je ještě potřeba provést několik věcí – včetně operací s I/O porty. Problém je zatím obcházen odstraňováním operací s prodlevou z kódu, který je spouštěn příliš brzy, ale někteří vývojáři **mají obavy** ⁽²¹²⁾, že se nikdy nepodaří je přesunout všechny. Padl návrh, že až do chvíle, kdy proběhne kalibrace, by jádro mohlo prostě předpokládat, že běží na nejrychlejším možném procesu. Ale kromě toho, že to není moc elegantní, tak by to mohlo výrazně snížit rychlost bootu na pomalejších strojích – které všechny se stávajícím kódem fungují dobře.

Skutečným řešením by bylo se prostě zbavit skoro všech těch operací s prodlevou. Jen velmi málo z nich bude pravděpodobně potřeba pro použití s hardwarem, který se ještě používá. V některých případech je navíc skutečným účelem prodlev kamuflování chyb v ovladačích. Pouhé odstranění prodlev by asi způsobilo nestabilitu na neočekávaných místech, čemuž by se vývojáři raději vyhnuli. Bude se to tedy muset provést opatrně a pomalu. V 2.6.24. zůstane využívání portu 80 asi beze změny.

revoke() se vrací

Naposledy jsme se na patch `revoke()`, který napsal Pekka Enberg, dívali v červenci 2006. Účelem tohoto volání je kompletně odpojit všechny procesy od určitého souboru, což umožní nějakému novému procesu exkluzivní přístup. Taková funkčnost má mnoho možných uplatnění, například zajištění, že nově přihlášený uživatel bude mít jako jediný přístup ke zdrojům spojeným s konzolí – třeba ke zvukovému zařízení. Jsou také vývojáři jádra, kteří si čas od času výhrůžně bručí pod vousy o neopravitelných bezpečnostních problémech způsobených absencí možnosti zrušit otevřené popisovače souborů – i když nikdo neví, proč se nechtějí podělit o podrobnosti ohledně těchto chyb. Také jakákoliv opravdová aplikace pro **hledání malwaru** ⁽²¹³⁾ bude potřebovat mít možnost zrušit přístup k souborům, o kterých zjistí, že obsahují Zlé Věci.

Pekka nedávno poslal **novou verzi patche** ⁽²¹⁴⁾, takže si ji prohlédneme. První věc, které si všimnete, je to, že systémové volání `revoke()` je pryč; místo toho vypadá nové volání takto:

```
int revokeat(int dir_fd, const char *filename);
```

Toto volání je tedy modelováno podle mnoha dalších, poměrně nových systémových volání typu `*at()`. V tomto případě je `filename` název souboru, k němuž má být zrušen přístup; pokud jde o absolutní cestu, pak je `dir_fd` ignorován. Jinak je `dir_fd` otevřený popisovač souboru ukazující na adresář, který se má použít jako počáteční bod při hledání `filename`. Hodnota `AT_FDCWD` určuje aktuální pracovní adresář volajícího procesu. Pokud je volání `revokeat()` úspěšně dokončeno, budou pro `filename` platné jen popisovače souborů, které byly vytvořeny až po volání. Patch také přidává nového člena do `file_operations`:

```
int (*revoke)(struct file *filp);
```

Funkce má na starosti zajistit, aby byly dokončeny všechny zbývající I/O operace s daným souborem – je-li to potřeba, tak i s chybovým stavem. Zatím jediná implementace je obecná verze pro souborové systémy; celá vypadá takto:

```
int generic_file_revoke(struct file *file)
{
    return do_fsync(file, 1);
}
```

Aby bylo volání `revokeat()` v budoucnu užitečné, bude potřebovat podporu alespoň v nějakých ovladačích.

Odpojení přístupu k obyčejným popisovačům souborů je poměrně jednoduché; systémové volání prostě prochází seznamem otevřených souborů na příslušném zařízení a nahrazuje strukturu `file_operations` novou sadou, která vrací `EBADF` za každý pokus o operaci (OK, skoro každý pokus – čtení ze socketů a souborů zařízení vrací místo toho nulu). Nepříjemné je pouze to, že seznamem souborů se musí projít vícekrát, než budou všechny otevřené soubory odstraněny; jinak by mohly nastat souběhy [race conditions] s jinými systémovými voláními, která vytvářejí nové popisovače souborů ve stejný okamžik, kdy jsou ty staré rušeny.

Složitější je to s mapovanou pamětí. Ve většině případů jde o nalezení všech oblastí virtuální paměti (VMA), které jsou se souborem spojeny. Pak se nastaví příznak `VM_REVOKED` a zavolá `zap_page_range()`, aby se pročistily příslušné záznamy v tabulce stránek. Příznak `VM_REVOKED` zajistí, že jakýkoliv pokus o vrácení stránek [fault pages back] vyústí v signál `SIGBUS` – to může být nepříjemné překvapení pro jakýkoliv proces, který by se pokoušel k té oblasti přistupovat.

Ještě horší je to v případě privátních copy-on-write (COW) [kopírování při zápisu] mapování, která mohou vzniknout, když se proces rozdělí [fork]. Obyčejné pročištění těchto mapování by mohlo být účinné, ale mohlo by to také způsobit konec procesů, které není potřeba zabíjet. Je však důležité si ohlídat, aby COW mapování nepředstavovalo způsob, jak ztrácet data zapsaná do souboru po volání `revokeat()`. COW mapování jsou tedy od sebe oddělena jednoduchým (ale nákladným) voláním `get_user_pages()`, které vytvoří privátní kopie všech relevantních stránek.

O patchi se zatím moc nediskutovalo – možná, že příslušní vývojáři už odjeli na vánoční dovolené a linux-kernel přestali sledovat. Jde však o důležitý patch s mnoha náročnými nízkourovňovými operacemi; to je částečně důvod, proč jeho příprava trvala tak dlouho. Než se bude uvažovat o začlenění do hlavního jádra, bude potřeba jej pečlivě zkontrolovat. Vzhledem k povaze problému by nebylo překvapivé, kdyby byla potřeba ještě jedna nebo dvě další verze.

■

Jaderné noviny – 2. 1. 2008

Aktuální verze jádra: 2.6.24-rc6. Citát týdne: David Miller.
SLUB v nesnázích. Racionalizace scatter/gather řetěů.

Robert Krátký

Aktuální verze jádra: 2.6.24-rc6

Aktuální předverze je (2. 1. 2008) 2.6.24-rc6, vydaná⁽²¹⁵⁾ 20. prosince. Linus⁽²¹⁶⁾: Seznam regresí se zmenšuje, takže zatím to vypadá, že by se mělo podařit vydat jádro 2.6.24⁽²¹⁷⁾ na počátku ledna. Za předpokladu, že se všichni během prázdnin nepřejíme, takže by nikdo nic neudělal. Ale všichni víme, že prázdniny jsou ve skutečnosti doba, kdy si dáme pohov od „opravdové práce“, a můžeme tak strávit 24 hodin denně hackováním jádra, že jo?

Dlouhý changelog⁽²¹⁸⁾ obsahuje všechny podrobnosti.

Aktuální verze -mm stromu je 2.6.24-rc6-mm1⁽²¹⁹⁾. Nedávné změny v -mm (kromě toho, že nefunguje na i386 systémech) zahrnují několik změn v nízkourovňovém modelu ovladačů, přepracování tmpfs, aktualizace ext4 a začátek odstraňování atributu `fastcall` z funkcí.

Starší jádra: 2.4.36⁽²²⁰⁾ vyšlo 1. ledna s několika opravami.

Pro blázny: 0.01⁽²²¹⁾ vydal Abdel Benamrouche také 1. ledna.

Citát týdne: David Miller

Musím říct, že vánoční prázdniny jsou to nejlepší období pro hackování. Většina blbečků je na dovolené, takže ten nepřetržitý proud vyrušování a opravdu hloupých emailů je pryč. Zkrátka můžete pracovat na věcech, na kterých chcete pracovat pořád, ale nikdy na ně nemáte čas.

– David Miller⁽²²²⁾: 31. 12. 2007⁽²²³⁾

SLUB v nesnázích

SLUB alokátor⁽²²⁴⁾ je nová implementace nízkourovňového jaderného alokátoru stránek; jde o náhradu dlouho používaného slab alokátoru. SLUB byl začleněn do 2.6.22⁽²²⁵⁾ a v 2.6.23⁽²²⁶⁾ byl nastaven jako výchozí alokátor. Do budoucna bylo v plánu, aby SLUB nakonec úplně nahradil slab. Tak to nakonec ještě může dopadnout, ale SLUB se teď potýká s potížemi.

První problém se týká nižšího výkonu v několika specifických situacích. Ukázalo se, že `hackbench`⁽²²⁷⁾, který měří výkon plánovače, běží pomaleji, když se používá SLUB. Výkon může být dokonce o polovinu nižší, což je dost na to, aby se to lidem nelíbilo. Tento výsledek byl mnoha lidmi potvrzen; objevily se i zprávy o horším výkonu s proprietárním testem `TPC-C`⁽²²⁸⁾, které už tak snadno reprodukovat nešlo. V obou případech to však vypadalo, že vývojáři SLUBu Christophu Lameterovi trvalo opravení chyb až příliš dlouho; koneckonců, reakci na zhoršení výkonu testů je normální poslat hned, když jsou zrovna vánoční prázdniny.

Když se k tomu Christoph dostal, poslal dlouhou analýzu⁽²²⁹⁾, ve které tvrdí, že rozsah problému je ve skutečnosti docela malý. Uzavřel to slovy: **s ohledem na všechna omezení při daném porovnávání si myslím, že nestojí za to se tím zabývat.**

To nebyla odpověď, v jakou doufal Linus⁽²³⁰⁾:

Je to fakt úplně jednoduché. Buď řekneš „Jasně, já to opravím“, nebo půjde SLUB pryč. Jiná možnost není. Když říkáš „nestojí za to se tím zabývat“, tak pro mě je to jasně a zřetelně „tak dejme SLUB pryč“.

Tou dobou se objevilo řešení problému v reakci na poznámku, kterou poslal Pekka Enberg⁽²³¹⁾. Vyšlo najevo, že podle profilů to vypadá, jako by interní funkce SLUBu nazývaná `add_partial()` stála za většinou toho spotřebovaného času. SLUB alokátor funguje tak, že rozděluje stránky do objektů stejných velikostí, přičemž v rámci těch stránek nemá žádná svá metadata. Když jsou alokovány všechny objekty ze stránky, SLUB na stránku úplně zapomene. Ale když je jeden z těch objektů uvolněn, musí SLUB danou stránku označit jako „částečnou“ a přidat ji do své fronty dostupné paměti. Vypadá to, že k tomuto přidávání částečných stránek docházelo mnohem častěji, než by mělo.

Nástroj `hackbench` pracuje tak, že rychle posílá data mezi procesory a sleduje, jak plánovač reaguje. V rámci tohoto procesu je vynuceno množství alokačních a uvolňovacích operací, což způsobovalo vytváření spousty částečných stránek. Konkrétní problém byl v tom, že částečná stránka byla po vytvoření přidána na začátek seznamu, takže další alokace alokovala ten jediný objekt, který byl na stránce k dispozici, a z částečné stránky se opět stala plná. A SLUB na ni tedy zapomněl. Když došlo k dalšímu uvolnění, celý cyklus se zopakoval.

Jakmile na tohle Christoph přišel, byla oprava⁽²³²⁾ na jeden řádek: částečné stránky by měly být přidávány na konec seznamu, ne na začátek. To dá stránce více času nashromáždit další volné objekty, než z ní zase bude zdroj pro alokace, a minimalizuje se tak přidávání a odstraňování částečných stránek. Výsledky⁽²³³⁾ přišly rychle: `hackbench` regrese byla opravena. Nebyly zveřejněny žádné výsledky testování s TPC-C (licence této sady benchmarků nemá ráda zveřejňování výsledků), ale předpokládá se, že problém byl napraven i tam.

Mezitím se začalo mluvit o jedné poněkud opožděné stížnosti na SLUB: pro SLUB alokátor neexistuje ekvivalent `/proc/slabinfo`. Soubor `slabinfo` může být velmi efektivním způsob, jak zjišťovat, co se děje s pamětí, kterou jádro alokuje; rychlé a účinné znázornění aktuální alokační situace. Nástroj `slabtop` tyto informace zpřístupňuje ještě více. Skutečnost, že při použití SLUBu `slabtop` nefunguje, některým vývojářům vadí; zdá se pravděpodobné, že až si SLUB najde cestu do distribučních jader, bude si stěžovat ještě více lidí. Uživatelé Linuxu obecně požadují více informací o tom, jak jádro pracuje; odstranění užitečného zdroje informací je nepotěší.

Někteří vývojáři dokonce tvrdili, že soubor `slabinfo` je součástí uživatelského ABI, a proto musí být zachován navždy. Je však těžké soudit, nakolik může být takové rozhraní opravdu tesáno do kamene; jde o poměrně přímý pohled na vnitřnosti jádra, které se časem mění. Takže argumentace s ABI se asi daleko nedostane, ale potřeba mít možnost dotazovat se na alokační situaci jádra zůstává.

Pracuje se na dvou řešeních tohoto problému. První připravuje Pekka Enberg: `patch pro náhradu slabinfo`⁽²³⁴⁾ pro SLUB, který poskytuje dostatek informací na to, aby `slabtop` fungoval. Ale skutečným zdrojem těchto informací bude v budoucnu účtyhodná sada souborů v `/sys/slab`. Hrabat se v tom adresáři ručně nelze doporučit, zvláště s ohledem na to, že existuje lepší způsob: soubor `slabinfo.c`, který je součástí zdrojů jádra (v `Documentation/vm`). Z něj může být zkompileován nástroj, který poskytuje stručné a užitečné informace o aktuálním použití slabu. Distributoři ten nástroj snad začnou dodávat (měl by se stát součástí sady `util-linux`); zatím je nutné kompilovat.

Poslední zbývající problém je už skoro tradiční: obávaná zpráva⁽²³⁵⁾ od Al Vira⁽²³⁶⁾ o tom, jak jsou pravidla životnosti souborů v `/sys/slab` nastavena úplně špatně. Ukázalo se, že i vývojář jako Christoph, který umí hackovat kód pro správu paměti a hezky vyladit systémy o 4096 procesorech, má potíže se `sysfs`⁽²³⁷⁾. Stejně jako v podstatě každý, kdo s tím kódem pracuje. Kolují patche, které by `sysfs` trochu racionalizovaly, což možná odstraní problémy v budoucnu. SLUB ale potřebuje rychlejší opravu. Je-li to však poslední zbývající problém, pak se zdá, že status Jediný Správný Alokátor je skoro na dosah.

Racionalizace `scatter/gather` řetězů

API pro řetězení `scatterlist`⁽²³⁸⁾ byla patrně nejušivější novinka v 2.6.24, ačkoliv se jedná o relativně malé množství kódu. Toto API jadernému kódu umožňuje řetězit `scatter/gather` seznamy pro DMA I/O

operace, takže je maximální velikost těchto operací větší. To vede k lepšímu výkonu, zvláště v subsystému blokového I/O. Řetězení scatterlistů je považováno za dobrou věc, ale objevily se stížnosti na stávající implementaci. V současné době musí kód, který by chtěl pracovat se zřetězenými scatterlisty, sestavit řetězy sám – v tom se snadno udělá chyba.

Jeden z přístupů ke zlepšení situace je `sg_ring API` ⁽²³⁹⁾, které navrhuje [Rusty Russell](#) ⁽²⁴⁰⁾. Tento patch odstraňuje současný přístup ke zřetězování; scatterlisty už nemají žádné položky (které jsou ve skutečnosti kamuflované řetězové ukazatele). Místo toho Rusty zavádí `struct sg_ring`:

```
struct sg_ring
{
struct list_head list;
unsigned int num, max;
struct scatterlist sg[0];
};
```

Je zřejmé, že řetězení se přesunulo ze scatterlistů a je z něj teď explicitní linkovaný seznam. Proměnné sledují aktuální a maximální velikosti seznamu, což pomáhá se snížením zátěže na jiných místech. Některé verze patche také přidávají celočíselné pole `dma_num`, které má obsahovat počet mapovaných scatter/gather položek – ten se může lišit od čísla původně stanoveného ovladačem. `sg_ring` s daným počtem scatterlist položek může být deklarována pomocí následujícího makra:

```
DECLARE_SG_RING(name, max);
```

Pak by měl být jedním z těchto způsobů inicializován kruh:

```
void sg_ring_init(struct sg_ring *ring, unsigned int max);
void sg_ring_single(struct sg_ring *ring, const void *buf,
unsigned int buflen);
```

Druhá možnost je zkratka pro případy, ve kterých musí být s daným bufferem založen kruh s jedinou položkou.

Sestavení kruhu s více položkami je záležitost alokování tolika samostatných `sg_ring` položek, kolik je potřeba, a explicitního zřetězení těchto položek pomocí pole `list`. Pro procházení všech položek kruhu je připraveno pomocné makro, jež skrývá hranice mezi jednotlivými scatterlisty:

```
struct sg_ring *sg;
int i;

sg_ring_for_each(ring, sg, i) {
/* *sg is the scatterlist entry to operate on */
}
```

Rusty poslal patche, které převádějí na toto API části SCSI subsystému. Jak poznamenal, převod odstraňuje dost práce spojené se sestavováním a rušením scatterlistů.

[Jens Axboe](#) ⁽²⁴¹⁾, autor původního kódu pro řetězení scatterlistů, [reagoval](#) ⁽²⁴²⁾ s tím, že aktuální API bylo psáno tak, aby se minimalizoval dopad na ovladače v 2.6.24. Není to, jak říká, dokončený produkt a situace už se zlepšuje. Pohled na jeho [git repozitář](#) ⁽²⁴³⁾ ukazuje v API doplňky, které mají podobný cíl jako Rustyho práce.

Jensův přístup ponechává stávající zřetězovací mechanismus, ale balí jej do struktury a několika pomocných funkcí, aby s ním byla snazší práce. Ovladače by pak pracovaly se `struct sg_table`:

```
struct sg_table {
```

```
struct scatterlist *sgl; /* seznam */
    unsigned int nents;      /* počet mapovaných položek */
    unsigned int orig_nents; /* původní velikost seznamu */
};
```

`sg_table` se sestaví pomocí:

```
int sg_alloc_table(struct sg_table *table, unsigned int nents,
                  gfp_t gfp_flags);
```

Tato funkce nealokuje strukturu `sg_table`, která musí být předána jako parametr. Místo toho alokuje paměť, která se použije pro pole `scatterlist`, a zařídí jejich zřetězení. Ovladači, který potřebuje sestavit velkou scatter/gather operaci, teď stačí provést jediné volání `sg_alloc_table()` a pak běžným způsobem projít seznamem položek scatterlistu. Když je operace dokončena, volání

```
void sg_free_table(struct sg_table *table);
```

uvolní alokovanou paměť.

Někdy v době začátku vývoje 2.6.25 bude potřeba rozhodnout o dalším směru API pro řetězení scatterlistů. Možná to nebude ta největší událost, ale stanoví se tím, jak bude v budoucnu psán kód řešící vysoce výkonné I/O. Coby autor začleněného zřetězovacího kódu má Jens asi trochu výhodu, jistě však není nic.

Jaderné noviny – 1/2008

2.6.24-rc4, „trošku pozdě“. 2.6.24-rc5, konečná verze do Vánoc nebude. Citát: Vyhýbání se OOM killerovi. Sbíráání dat z jaderných oops. Oprášení jádra 0.01. Vydání stabilního jádra 2.4.36. Citát: Proti tomu se postavím. Ospalý Linux. Citát: Výhody TuxOnIce. 2.6.24-rc7, spousta malých změn. Citát: Menšina. Dekódování oops.

Jirka Bourek

2.6.24-rc4, trošku pozdě

4. prosinec, [originál](#) ⁽²⁴⁴⁾

Mezi jednotlivými vydáními -rc bychom měli mít týdenní interval, ale během Děkuvzdání jsem byl týden pryč (stejně jako mnoho dalších vývojářů jádra), takže tohle je trochu opožděné. Zpoždění začíná být spíše pravidlem než výjimkou, ale slibuji, že se polepším...

Linus Torvalds ⁽²⁴⁵⁾ takto začal oznámení o vydání jádra 2.6.24-rc4. ⁽²⁴⁶⁾ *Není v něm moc vzrušujících změn, ale děje se v něm o hodně víc změn, než bych doufal ve stádiu, kdy máme -rc4. V diffstatech vynikají Blackfin, MIPS a Power, ale v ARM a x86 došlo také k nějakým aktualizacím.*

Něco se dělo také u ACPI (příškrcování [throttling] procesoru atd.), společně s tím proběhly různé aktualizace ovladačů: ATA, IDE, infiniband, SCSI, USB a síťové ovladače. U souborových systémů cifs, NFS, ocfs2 a proc. Uf. Je toho příliš.

Diff od -rc3 má skoro 36000 řádek a to je ten menší pro git, kde se přejmenování ukazují jako přejmenování (není to ten, který nahrávám jako patche na kernel.org, ten je dělán tak, aby ho mohli používat lidé s GNU patch a dalšími starodávnými programy, které používají diffy). Částečně za to může to dvoutýdenní okno, ale i tak je to trochu skličující. Opravdu doufám, že budeme zpomalovat a že -rc5 bude o hodně menší.

Zkrátka, žádná ze změn není opravdu *vzrušující* nebo opravdu děsivá. Také jsme opravili spoustu regresí, i když víc jich určitě zbývá.

2.6.24-rc5, konečná verze do Vánoc nebude

12. prosinec, [originál](#) ⁽²⁴⁷⁾

Je to týden, co jsem slíbil, že budu hodný chlapec a budu dodržovat pravidla o vydávání verzí, takže tady je další -rc, psal Linus Torvalds v [oznámení jádra 2.6.24-rc5](#) ⁽²⁴⁸⁾ *Dodal k němu: Věci se zpomalily, ale lhal bych, kdybych řekl, že máme všechny regrese ošetřené a pod kontrolou. Pracuje se na nich a seznam se zmenšuje, ale můj odhad je, že určitě nebudeme mít finální verzi 2.6.24 hotovou před Vánoci, pokud tedy Santa nezaměstná víc elfů, aby na těch regresích zapracovali. Takže všichni elfové tam venku – prosím, pokračujte v práci.*

Linus dodal, že v tomto nejnovějším release candidate nebyly žádné velké změny, takže nemá cenu posílat diffstat. S ohledem na délku textu je na pohled výrazný revert PA-RISC a aktualizace defconfigu u powerpc. A SPI ovladač pro Blackfin. Zbytek je z velké části náhodný šum v různých subsystémech (ovladače/síť, souborový systém xfs a aktualizace architektur jsou některé z oblastí, kde se objevuje víc změn).

Citát: Vyhýbání se OOM killerovi

Kdykoliv začne OOM killer střílet, se systémem je něco špatně a je mnohem produktivnější to řešit, než si přát přesnějšího OOM killera.

Dan Kegel, zpráva ze 7. prosince⁽²⁴⁹⁾ na Linux Kernel mailing list.

Sbírání dat z jaderných oops

17. prosinec, originál⁽²⁵⁰⁾

Arjan van de Ven⁽²⁵¹⁾ oznámil novou webovou stránku: <http://www.kerneloops.org>⁽²⁵²⁾ sbírá z různých mailových konferencí a bugzill zprávy z jaderných oops a varování.

Přiložil shrnutí deseti nejčastějších oops zpráv za posledních 7 dní a poznamenal: **Toto je první takové shrnutí, které zasílám. Dejte mi, prosím, vědět, jestli je to užitečné, nebo ne.**

Odezva byla kladná. Andrew Morton⁽²⁵³⁾ poznamenal: **To mohla být sranda naprogramovat.**

Steven Richter vyjádřil obavy, že tento nástroj bude započítávat stejnou zprávu o chybě několikrát, pokud ji najde na několika místech. S tím Arjan souhlasil: **To je pravda, nicméně je to... složitá věc. Je opravdu složité rozlišit duplicitní zprávu od dvou zpráv o stejné chybě.**

Objevila se také další obava, že budou odděleny oops vygenerované jádrem 2.6.X-rcY a 2.6.X.rcY-mmZ. Arjan poznamenal: **Zjištění, že které verze jádra ten oops pochází je... překvapivě složité. A upřímně, chyby proti -mm nás také dost zajímají, protože se přece jenom v budoucnu dostanou do hlavní řady.**

Oprášení jádra 0.01

2. leden, originál⁽²⁵⁴⁾

Abdel Benamrouche oznámil⁽²⁵⁵⁾, že aktualizoval původní jádro verze 0.01 tak, aby šlo přeložit gcc řady 4.x a mohlo tedy běžet v emulátorech jako je QEMU a Bochs.

Po aplikování sady malých patchů, vysvětluje Abdel, může být jádro 0.01 přeloženo na systému, který běží na linuxovém jádře 2.6. Dodal, že úspěšně portoval bash-3.2, část coreutils-6.9, dietlibc-0.31 (místo glibc), bin86-0.16.17, make-3.81, ncurses-2.0.7 a vim-7.1, aby mohly na tomto modifikovaném jádře fungovat.

Vydání stabilního jádra 2.4.36

3. leden, originál⁽²⁵⁶⁾

Nový rok, nové jádro: Linux 2.4.36 je konečně hotové a bylo prověřováno dost dlouho na to, abychom ho mohli vydat. Od 2.4.35 byla opravena spousta chyb, chyb při překladu a bezpečnostních problémů, ale všechny tyto opravy byly začleněny do 2.4.35-stable,

oznámil⁽²⁵⁷⁾ správce jádra 2.4 Willy Tarreau.

Měl bych říct, že jsem docela spokojen s modelem vydávání s dvěma větvemi, který velmi úspěšně odděluje rychlé opravy od změn, které potřebují podrobnější testování.

Potom dodal: **Co se budoucích verzí týče, nezbyvá už nic, co by bylo potřeba udělat. Budu pokračovat s 2.4.36.X, až přijdou opravy chyb. 2.4.37 založím jenom v případě, že dostanu něco, co nebudu považovat za vhodné pro zařazení do 2.4.36.X.**

Předchozí stabilní jádro 2.4.35⁽²⁵⁸⁾ bylo vydáno v červenci 2007. Změny na úrovni zdrojových kódů lze prohlédnout přes [gitweb rozhraní linux-2.4](#)⁽²⁵⁹⁾.

Citát: Proti tomu se postavím

Proti čemu se budu hodně stavět, je přístup „nevíme, co je špatně, ale necháme to špatně, protože nás přece nikdo nemůže otravovat s tím, abychom to řešili.“

Linus Torvalds, zpráva ze 2. ledna ⁽²⁶⁰⁾ na Linux Kernel mailing list.

Ospalý Linux

4. leden, originál ⁽²⁶¹⁾

Současné verze Linuxu umí přejít do režimu usnutí do paměti [suspend-to-RAM] docela dobře, ale mohou to udělat jenom na přímý příkaz. Uspávání do paměti je ale důležité, spotřeba energie v porovnání s nezatíženým [idle] systémem je asi 10 %. Ruční uspávání není příliš vhodné, začal Pavel Machek ⁽²⁶²⁾ popis svého nápadu, který nazval **Ospalý Linux**. ⁽²⁶³⁾

Ruční uspávání nepřichází v úvahu na víceuživatelských strojích a ani na jedinouživatelských to není tak jednoduché: 1) Stáhni tenhle velký kus něčeho v Mozille a pak se uspi; 2) Zkompiluj tohle a pak se uspi; 3) Můžeš se uspat hned, ale v 8:30 mě probudí mp3 přehrávačem. Pavel poskytl jednoduchý a ne zcela funkční patch, potom popsal navrhované řešení: Dnešní hardware je většinou schopen fungovat lépe: se správně nastaveným probouzením může stroj spát a přitom předstírat, že nespí, tak, že se probudí pokaždé, když se stane něco zajímavého. To je samozřejmě jednodušší na strojích, které nejsou připojeny do sítě, a na noteboocích.

Citát: Výhody TuxOnIce

Možná to také pomůže v mé snaze, pokud si na nějakou najdu čas, přesvědčit Andrewa, že TuxOnIce má skutečně významné výhody oproti (u)swsusp a na kexec založené hibernaci.

Nigel Cunningham, zpráva z 1. ledna ⁽²⁶⁴⁾ na Linux Kernel mailing list.

2.6.24-rc7, spousta malých změn

7. leden, originál ⁽²⁶⁵⁾

Linus Torvalds oznámil vydání ⁽²⁶⁶⁾ linuxového jádra 2.6.24-rc7 ⁽²⁶⁷⁾: Od -rc6 uplynuly dva týdny, ale přiznejme si, že mezi Vánoci a Novým rokem (a narozeninami) nebylo mnoho pracovních dní, takže rozdílový patch z -rc6 je poloviční oproti tomu z rc5 na rc6. Budu shovívavý a budu tvrdit, že je to díky tomu, že se všechno stabilizuje, ne proto, že jsme všichni během svátků byli zpití do němoty.

Linus stručně shrnul změny: Zkrácený log (přiložen dole) je krátký a vcelku informativní. Je to prostě jenom spousta malých změn. Diffstat zobrazuje spoustu jedno- a dvouřádkových změn, větší pozornost budí jen pár ovladačů (a platforma Cell.) Více vyniká jenom podpora `/proc/slabinfo` ve SLUB.

Citát: Menšina

Musím říct, že chyby, které skutečně zmizí poté, co uživatel přestane používat nvidia/fglrx/ndiswrapper/atd., tvoří v celkovém počtu menšinu.

Andrew Morton, zpráva z 9. ledna ⁽²⁶⁸⁾ na Linux Kernel mailing list.

Dekódování oops

9. leden, **originál** ⁽²⁶⁹⁾

Tento týden bylo hlášeno celkem 49 oops hlášení a varování, minulý týden to bylo 53, napsal Arjan van de Ven, když posílal seznam **nejčastějších 10 kernel oops** ⁽²⁷⁰⁾ z tohoto týdne. Al Viro ⁽²⁷¹⁾ navrhl: Lidé, kteří si stěžují, že s prací v jádře teprv začínají a že pro ně proto není dost práce, by mohli dekodovat tato hlášení na „tohle místo v téhle funkci, funkce byla volána <odtud>, hodnota té a té proměnné byla <taková>“ a posílat výsledky.

Následovalo mnoho požadavků na návod, jak takový oops dekodovat. Linus Torvalds odpověděl: Ve skutečnosti to není vždy tak triviální, zvláště když nemáte hluboké znalosti o kódu generovaném pro danou architekturu (a i tak jsou některé oops složitější než jiné kvůli inliningu, volání funkce na konci jiné [tail-call], atd.) Pokud oops nastane u kernelu, který jste si přeložili sami, je to většinou jednoduché, zvláště pokud jste při konfiguraci řekli „y“ u „Generovat ladící informace“ [Generate debugging info].

Linus pokračoval detailnějším postupem, jak hledat chybu v oops hlášení, které náhodně vybral v `◊lkml` ⁽²⁷²⁾. Obecně musíte disasemblovat šestnáctkovou sekvenci uvedenou v oops hlášení (řádka „Code“) a pokusit se ji spojit se zdrojovým kódem tak, abyste zjistili, co se děje.

Potom uvedl několik tipů, jak toho dosáhnout nejlépe, a pokračoval příkladem, ve kterém prošel jeden z nahlášených oops. Al Viro se přidal **popisem svých vlastních metod** ⁽²⁷³⁾, jak dosáhnout stejného výsledku, přičemž prošel jiné hlášení a chybu našel.

■

Odkazy

Distribuční novinky – 52/2007

- [1] <http://www.abclinuxu.cz/zpravicky/debian-4.0r2-etch>
- [2] <http://www.debian.org/News/2007/20071228>
- [3] <http://www.abclinuxu.cz/zpravicky/zenwalk-linux-5.0-beta>
- [4] <http://www.parsix.org/html/News-article-sid-65.html>
- [5] http://grml.org/2007/12/29/#release_1.1-rc1
- [6] http://www.pclinuxos.com/index.php?option=com_content&task=view&id=46&Itemid=2
- [7] <http://vectorlinux.com/article.php?sid=7>
- [8] <http://www.musix.org.ar/wiki/index.php/Musix1.0r3test4-announce>
- [9] <http://www.abclinuxu.cz/zpravicky/sco-konecne-vyrazeno-z-nasdaq>
- [10] <http://www.abclinuxu.cz/slovník/olpc>
- [11] <http://www.abclinuxu.cz/zpravicky/technologicka-reditelka-olpc-opousti-svuj-post>
- [12] <http://dev.laptop.org/%7Earjs/cp1.JPG>
- [13] <http://lwn.net/Articles/263542/>
- [14] <http://www.abclinuxu.cz/zpravicky/vietnam-by-mel-uprednostnovat-open-source-software>
- [15] <http://www.gobolinux.org>
- [16] <http://gobolinux.org/index.php?page=doc/articles/gobohide>
- [17] http://gobolinux.org/index.php?page=release_notes_014
- [18] <http://lists.freebsd.org/pipermail/freebsd-stable/2007-December/039334.html>
- [19] <http://www.freebsd.org/platforms/xbox.html>
- [20] <http://lists.freebsd.org/pipermail/freebsd-stable/2007-December/039352.html>
- [21] <http://www.endian.com/>
- [22] <http://www.endian.com/en/community/about/features/>
- [23] https://sourceforge.net/project/shownotes.php?release_id=565051&group_id=132104
- [24] <http://annvix.org/>
- [25] <http://blog.annvix.org/index.php?/archives/93-Annvix-3.0-RELEASE-Freya-now-available.html>
- [26] <http://packages.qa.debian.org/u/unp.html>

Distribuční novinky – 1 a 2/2008

- [27] <http://www.abclinuxu.cz/zpravicky/druha-alfaverze-mandriva-linux-2008-spring>
- [28] <https://lists.ubuntu.com/archives/ubuntu-devel-announce/2008-January/000367.html>
- [29] <http://www.abclinuxu.cz/zpravicky/pclinuxos-minime-2008>
- [30] http://wiki.contribs.org/SME_Server_7.3
- [31] <http://www.mepis.org/node/14188>
- [32] <http://www.abclinuxu.cz/zpravicky/trixbox-ce-2.4.0>
- [33] <http://www.ultimalinux.com/forum/viewtopic.php?p=4930>
- [34] <http://www.darkstarlinux.ro/content/darkstar-linux-20081-0>
- [35] <http://www.frugalware.org/news/83>
- [36] <http://www.abclinuxu.cz/zpravicky/gnu-octave-3.0>
- [37] <http://www.parsix.org/html/News-article-sid-67.html>
- [38] <http://vortalinux.sicurezza.com/?p=34>
- [39] http://www.goblinx.com.br/en/index_news.htm
- [40] <http://www.abclinuxu.cz/zpravicky/mcafee-ma-obavy-z-gnu-gpl>
- [41] <http://www.abclinuxu.cz/slovník/gnu-gpl>
- [42] <http://www.abclinuxu.cz/zpravicky/spor-busybox-vs-xterasys-vyresen-mimosoudne>
- [43] <http://www.becta.org.uk/>
- [44] <http://www.abclinuxu.cz/zpravicky/oss-dostane-vetsi-sanci-v-britskem-skolstvi>
- [45] <http://www.abclinuxu.cz/slovník/odf>
- [46] <http://www.abclinuxu.cz/clanky/ruzne/distribucni-novinky-29-2007#olpc-ma-noveho-clena-intel>
- [47] <http://www.abclinuxu.cz/slovník/olpc>
- [48] <http://www.abclinuxu.cz/slovník/classmate-pc>
- [49] money.cnn.com/2008/01/04/technology/kirkpatrick_negroponte_fortune/index.htm?postversion=2008010416
- [50] <http://www.abclinuxu.cz/zpravicky/intel-opousti-spravni-radu-olpc>
- [51] http://en.wikipedia.org/wiki/Daniel_Robbins
- [52] <http://blog.funtoo.org/2008/01/and-it-gets-worse.html>
- [53] <http://blog.funtoo.org/2008/01/here-my-offer.html>
- [54] http://en.wikipedia.org/wiki/E%2ATRADE_Financial
- [55] <http://forums.gentoo.org/>
- [56] <http://lwn.net/Articles/253875/>
- [57] <http://www.abclinuxu.cz/zpravicky/amd-uvolnilo-dalsi-davku-dokumentace-ke-svym-gpu>

- [58] <http://www.abclinuxu.cz/slovník/nda>
- [59] <http://www.abclinuxu.cz/slovník/nouveau>
- [60] <http://www.phoronix.com/scan.php?page=article&item=966>
- [61] <http://www.abclinuxu.cz/zpravicky/chysta-vidia-odezvu-na-open-source-strategii-amd>
- [62] <http://www.abclinuxu.cz/clanky/rozhovory/andy-ritger-vidia>
- [63] <http://www.abclinuxu.cz/zpravicky/desktopbsd-1.6>
- [64] <http://desktopbsd.net/index.php?id=89>
- [65] <http://www.kanotix.com>
- [66] <http://www.kanotix.com/modules.php?op=modload&name=News&file=article&sid=234&feedlang=eng&topic=7>
- [67] <http://www.linuxmint.com/forum/viewtopic.php?f=0&t=8248>
- [68] <http://www.linuxmint.com/forum/viewtopic.php?f=0&t=8250>
- [69] http://www.linuxmint.com/rel_daryna_debian.php

Distribuční novinky – 3/2008

- [70] <http://www.abclinuxu.cz/zpravicky/freebsd-6.3>
- [71] http://trinityhome.org/Home/blog_comments.php?front_id=15&blog_cat_id=2&b_node=2&blog_id=231
- [72] <https://www.scientificlinux.org/news/sl51>
- [73] <http://blog.pfsense.org/?p=164>
- [74] <http://tinyme.mypclinuxos.com/forums/index.php?topic=1566.0>
- [75] <https://forum.startcom.org/viewtopic.php?f=10&t=1415>
- [76] <http://www.abclinuxu.cz/clanky/ruzne/distribucni-novinky-1-a-2-2008#potize-ve-vedeni-gentoo-foundation>
- [77] <http://www.abclinuxu.cz/zpravicky/gentoo-foundation-se-obnovuje>
- [78] <http://www.abclinuxu.cz/zpravicky/gentoo-2007.1-zruseno-bulletin-obnoven>
- [79] <http://www.abclinuxu.cz/zpravicky/jak-vidi-krizi-gentoo-donnie-berkholz>
- [80] <http://www.abclinuxu.cz/zpravicky/manbo-labs-partnerstvi-mandriva-a-turbolinux>
- [81] <http://www.abclinuxu.cz/slovník/gnu-gpl>
- [82] <http://www.abclinuxu.cz/zpravicky/vydanie-kde-4.1-naplanovane-na-jul>
- [83] <http://www.abclinuxu.cz/slovník/decibel>
- [84] <http://www.abclinuxu.cz/zpravicky/qt-i-pod-gplv3>
- [85] <http://www.abclinuxu.cz/zpravicky/ruske-skoly-budou-vybaveny-linuxem-a-open-source>
- [86] <http://www.zenwalk.org/>
- [87] <http://www.abclinuxu.cz/zpravicky/zenwalk-linux-5.0>
- [88] <http://www.abclinuxu.cz/slovník/hal>
- [89] <http://www.abclinuxu.cz/zpravicky/opensuse-11.0-alpha-1>

Distribuční novinky – 4/2008

- [90] <http://www.abclinuxu.cz/zpravicky/ubuntu-6.06.2-lts>
- [91] <http://www.sysresccd.org/Changes-x86>
- [92] <http://www.elivecd.org/Main/News/78>
- [93] <http://dream.linux-hardcore.com/index.php/topic,199.0.html>
- [94] http://www.linuxmint.com/rel_daryna_kde.php
- [95] <http://www.computerworlduk.com/technology/applications/enterprise/news/index.cfm?newsid=7193&print>
- [96] <http://www.abclinuxu.cz/zpravicky/hp-uvolnilo-nastroje-pro-rizeni-foss-a-jeste-vice>
- [97] <http://www.fossology.org/>
- [98] <https://fossbazaar.org/>
- [99] <http://www.abclinuxu.cz/clanky/ruzne/distribucni-novinky-1-a-2-2008#nvidia>
- [100] <http://lists.freedesktop.org/archives/xorg/2008-January/032394.html>
- [101] <http://cs.wikipedia.org/wiki/HTPC>
- [102] <http://www.abclinuxu.cz/slovník/git>
- [103] http://www.phoronix.com/scan.php?page=news_item&px=NjMwNQ
- [104] <http://www.gentoo.org/news/en/gmn/20080121-newsletter.xml>
- [105] http://www.gentoo.org/news/20080123_releng_beta.xml
- [106] http://www.gentoo.org/news/20080124_releng-feature-request.xml
- [107] <http://www.abclinuxu.cz/zpravicky/nemecke-ministerstvo-prace-preslo-na-linux>
- [108] <http://www.parsix.org/html/pnWikka-tag-ReleaseNotes10r0.html>
- [109] <http://www.abclinuxu.cz/clanky/recenze/patchsety-pro-kernel#patchset-ck>
- [110] http://www.goblinx.com.br/en/index_micro.htm
- [111] http://www.goblinx.com.br/en/index_news.htm
- [112] <http://www.abclinuxu.cz/zpravicky/prvni-beta-mandriva-linux-2008.1-spring-opphrys>
- [113] <http://www.abclinuxu.cz/zpravicky/nvidia-ovladace-169.09>
- [114] <http://www.abclinuxu.cz/clanky/recenze/jake-je-kde-4.0.0>
- [115] <http://www.abclinuxu.cz/zpravicky/linux-2.6.24>
- [116] <http://www.abclinuxu.cz/zpravicky/alsa-1.0.16-rc1>

Gentoo ebuild – 6 (závislosti, sloty, restrikce)

- [117] <http://www.abclinuxu.cz/clanky/navody/gentoo-ebuild-1-zaciname-nazvy-ebuildu-a-promenne>
- [118] <http://www.abclinuxu.cz/clanky/navody/gentoo-ebuild-2-funkce-eclass-a-prikazy>

Gentoo Linux – Portage Overlay

- [119] <http://www.abclinuxu.cz/serialy/gentoo-ebuild>
- [120] <http://www.abclinuxu.cz/serialy/gentoo-ebuild>
- [121] <http://www.abclinuxu.cz/clanky/navody/gentoo-ebuild-2-funkce-eclass-a-prikazy>
- [122] <http://www.abclinuxu.cz/clanky/navody/gentoo-ebuild-3-live-ebuildy>

Jaké je KDE 4.0.0

- [123] <http://www.abclinuxu.cz/clanky/recenze/co-prinasi-kde-4-technologie>
- [124] <http://www.abclinuxu.cz/clanky/recenze/co-prinasi-kde-4-alfaverze-party-a-d-bus>
- [125] <http://www.abclinuxu.cz/slovník/kde4>
- [126] <http://www.abclinuxu.cz/slovník/oxygen>
- [127] <http://www.abclinuxu.cz/slovník/kde>
- [128] <http://www.abclinuxu.cz/slovník/Plasma>
- [129] [/Profile/10292](#)
- [130] <http://www.abclinuxu.cz/clanky/rozhovory/rozhovor-lubos-lunak-o-kde>
- [131] <http://techbase.kde.org/index.php?title=Projects/KWin/4.0-release-notes>
- [132] <http://techbase.kde.org/Projects/Plasma/FAQ>
- [133] <http://www.abclinuxu.cz/clanky/ruzne/distribucni-novinky-47-2007#distribucni-rada-rychle-vypocty-v-kde>
- [134] <http://www.abclinuxu.cz/clanky/recenze/co-prinasi-kde-4-technologie>
- [135] <http://www.kde.org/announcements/4.0/guide.php>
- [136] <http://www.abclinuxu.cz/slovník/phonon>
- [137] <http://www.abclinuxu.cz/clanky/recenze/novinky-v-kde-4>
- [138] <http://www.abclinuxu.cz/slovník/webkit>
- [139] <http://www.aselabs.com/img.php?id=2187&asesessid=b92a6bfd70648ce4f8db9db7687be6774e6628eb>
- [140] <http://decibel.kde.org>
- [141] <http://basysblog.org/index.php/archives/kcall-for-kde4-using-decibel>
- [142] <http://blog.jacobrideout.net/>
- [143] <http://www.kdedevelopers.org/node/3187>
- [144] http://techbase.kde.org/Development/Tutorials/KSpread_Scripting#The_R_Project
- [145] [http://en.wikipedia.org/wiki/R_\(programming_language\)](http://en.wikipedia.org/wiki/R_(programming_language))
- [146] <http://video.google.de/videoplay?docid=-2542844895415903403&hl=de>

Knihy: OpenOffice.org WRITER a CALC

- [147] <http://www.abclinuxu.cz/slovník/odf>
- [148] <http://www.computermedia.cz/knihy/openoffice-writer.html>
- [149] <http://www.computermedia.cz/knihy/openoffice-calc.html>

Viac či menej otvorene grupvérové systémy

Inkscape – 6 (vizitka)

- [150] <http://www.abclinuxu.cz/clanky/navody/inkscape-4-text>
- [151] <http://cs.wikipedia.org/wiki/Tux>
- [152] <http://www.abclinuxu.cz/images/clanky/valka/inkscape-6-vizitka-abclinuxu.svg>
- [153] <http://www.abclinuxu.cz/images/clanky/valka/inkscape-6-deset-vizitek-abclinuxu.svg>

Bankovní Komerční banky v Linuxu

- [154] <https://www.mojebanka.cz/>

Regulární výrazy

- [155] <http://www.perl.com/doc/manual/html/pod/perlre.html>
- [156] <http://sed.sourceforge.net/sed1line.cz.html>
- [157] <http://www.lexmasterclass.com/exercises/regex/index.html>

Zpravodaj o Víně – 339

- [158] <http://www.abclinuxu.cz/zpravicky/wine-0.9.53>
- [159] <http://wiki.winehq.org/DirectDraw>
- [160] http://blogs.computerworld.com/macworld_2008_predictions
- [161] <http://bugs.winehq.org/reports.cgi?product=-All-&datasets=NEW%3A&datasets=UNCONFIRMED%3A>
- [162] http://bugs.winehq.org/graphs/-All-_NEW_UNCONFIRMED.png
- [163] <http://www.fallingleafsystems.com/>
- [164] <http://www.abclinuxu.cz/zpravicky/projekt-alky-je-open-source>
- [165] <http://www.abclinuxu.cz/slovník/commit>
- [166] <http://www.abclinuxu.cz/slovník/patch>
- [167] <http://mlin.net/StartupCPL.shtml>
- [168] <http://www.abclinuxu.cz/slovník/gnu-lgpl>
- [169] <http://www.abclinuxu.cz/slovník/gnu-gpl>

Jaderné noviny – 12. 12. 2007

- [170] <http://lwn.net/Articles/261668/>
- [171] <http://www.abclinuxu.cz/kdo-je/linus-torvalds>
- [172] <http://kernel.org/pub/linux/kernel/v2.6/testing/ChangeLog-2.6.24-rc5>
- [173] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-11.-4.-2007#slub-alokator>
- [174] <http://lwn.net/Articles/261868/>
- [175] <http://lwn.net/Articles/261869/>
- [176] <http://lwn.net/Articles/261870/>
- [177] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-28.-2.-2007#threadlets>
- [178] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-30.-5.-2007#navrat-sysletu>
- [179] <http://lwn.net/Articles/261325/>
- [180] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-20.-11.-2007>
- [181] <http://www.abclinuxu.cz/kdo-je/jonathan-corbet>
- [182] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-15.-8.-2007#chytřejši-priskrcování-zapísu>
- [183] <http://lwn.net/Articles/261271/>
- [184] <http://www.abclinuxu.cz/kdo-je/andrew-morton>
- [185] <http://lwn.net/Articles/261633/>
- [186] <http://lwn.net/Articles/261634/>
- [187] <http://lwn.net/Articles/261714/>
- [188] http://bugzilla.kernel.org/show_bug.cgi?id=9346
- [189] <http://lwn.net/Articles/261722/>
- [190] <http://lwn.net/Articles/261723/>
- [191] <http://www.abclinuxu.cz/kdo-je/alan-cox>
- [192] <http://lwn.net/Articles/261725/>
- [193] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-14.-11.-2007>
- [194] <http://lwn.net/Articles/261727/>
- [195] http://bugzilla.kernel.org/show_bug.cgi?id=9346#c66

Jaderné noviny – 19. 12. 2007

- [196] <http://lwn.net/Articles/262012/>
- [197] <http://lwn.net/Articles/262174/>
- [198] <http://lwn.net/Articles/262209/>
- [199] <http://lwn.net/Articles/262592/>
- [200] <http://lwn.net/Articles/262173/>
- [201] <http://lwn.net/Articles/262361/>
- [202] <http://lwn.net/Articles/262362/>
- [203] <http://lwn.net/Articles/262741/>
- [204] <http://lwn.net/Articles/262742/>
- [205] <http://lwn.net/Articles/262555/>
- [206] <http://kerneloops.org>
- [207] <http://www.abclinuxu.cz/slovník/oops>
- [208] <http://lwn.net/Articles/250967/>
- [209] <http://www.abclinuxu.cz/kdo-je/andrew-morton>
- [210] <http://lwn.net/Articles/262557/>
- [211] <http://lwn.net/Articles/262559/>
- [212] <http://lwn.net/Articles/262562/>
- [213] <http://lwn.net/Articles/260918/>
- [214] <http://lwn.net/Articles/262377/>

Jaderné noviny – 2. 1. 2008

- [215] <http://lwn.net/Articles/262978/>
- [216] <http://www.abclinuxu.cz/kdo-je/linus-torvalds>
- [217] <http://www.abclinuxu.cz/zpravicky/linux-2.6.24>
- [218] <http://kernel.org/pub/linux/kernel/v2.6/testing/ChangeLog-2.6.24-rc6>
- [219] <http://lwn.net/Articles/263078/>
- [220] <http://lwn.net/Articles/263545/>
- [221] <http://lwn.net/Articles/263562/>
- [222] <http://www.abclinuxu.cz/kdo-je/david-s.-miller>
- [223] <http://vger.kernel.org/davem/cgi-bin/blog.cgi/2007/12/31#holidays2007>
- [224] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-11.-4.-2007#slub-alokator>
- [225] <http://www.abclinuxu.cz/zpravicky/jadro-2.6.22>
- [226] <http://www.abclinuxu.cz/zpravicky/vyslo-dlouho-ocekavane-jadro-2.6.23>
- [227] <http://devresources.linux-foundation.org/craiger/hackbench/>
- [228] <http://www.tpc.org/tpcc/>
- [229] <http://lwn.net/Articles/263332/>
- [230] <http://lwn.net/Articles/263333/>
- [231] <http://lwn.net/Articles/263334/>
- [232] <http://lwn.net/Articles/263335/>
- [233] <http://lwn.net/Articles/263336/>
- [234] <http://lwn.net/Articles/263337/>
- [235] <http://lwn.net/Articles/263338/>
- [236] <http://www.abclinuxu.cz/kdo-je/alexander-viro>
- [237] <http://lwn.net/Articles/263339/>
- [238] <http://www.abclinuxu.cz/clanky/jaderné-noviny/jaderné-noviny-31.-10.-2007#api-pro-retezeni-scatterlistu>
- [239] <http://lwn.net/Articles/263203/>
- [240] <http://www.abclinuxu.cz/kdo-je/rusty-russell>
- [241] <http://www.abclinuxu.cz/kdo-je/jens-axboe>
- [242] <http://lwn.net/Articles/263345/>
- [243] <http://git.kernel.dk/?p=linux-2.6-block.git;a=shortlog;h=sg>

Jaderné noviny – 1/2008

- [244] http://kerneltrap.org/Linux/2.6.24-rc4_A_Bit_Late
- [245] <http://www.abclinuxu.cz/kdo-je/linus-torvalds>
- [246] <http://kerneltrap.org/mailarchive/linux-kernel/2007/12/4/472625>
- [247] http://kerneltrap.org/Linux/2.6.24-rc5_No_Final_Before_Xmas
- [248] <http://kerneltrap.org/mailarchive/linux-kernel/2007/12/11/496613>
- [249] <http://kerneltrap.org/mailarchive/linux-kernel/2007/12/7/493388>
- [250] http://kerneltrap.org/Linux/Collecting_Kernel_Oops_Data
- [251] <http://kerneltrap.org/mailarchive/linux-kernel/2007/12/14/507875>
- [252] <http://www.kerneloops.org/>
- [253] <http://www.abclinuxu.cz/kdo-je/andrew-morton>
- [254] http://kerneltrap.org/Linux/Dusting_Off_the_0.01_Kernel
- [255] <http://kerneltrap.org/mailarchive/linux-kernel/2008/1/1/532268>
- [256] http://kerneltrap.org/Linux/2.4.36_Stable_Release
- [257] <http://kerneltrap.org/mailarchive/linux-kernel/2008/1/1/531560>
- [258] <http://kerneltrap.org/node/14003>
- [259] <http://git.kernel.org/?p=linux/kernel/git/wtarreau/linux-2.4.git;a=summary>
- [260] <http://kerneltrap.org/mailarchive/linux-kernel/2008/1/2/534188>
- [261] http://kerneltrap.org/Linux/Sleepy_Linux
- [262] <http://www.abclinuxu.cz/kdo-je/pavel-machek>
- [263] <http://kerneltrap.org/mailarchive/linux-kernel/2007/12/25/523664>
- [264] <http://kerneltrap.org/mailarchive/linux-kernel/2008/1/1/532418>
- [265] http://kerneltrap.org/Linux/2.6.24-rc7_A_Lot_of_Rather_Small_Changes
- [266] <http://kerneltrap.org/mailarchive/linux-kernel/2008/1/6/543164>
- [267] <http://kerneltrap.org/mailarchive/git-commits-head/2008/1/6/543305>
- [268] <http://kerneltrap.org/mailarchive/linux-kernel/2008/1/9/550838>
- [269] http://kerneltrap.org/Linux/Decoding_Oops
- [270] <http://kerneltrap.org/mailarchive/linux-kernel/2008/1/5/540557>
- [271] <http://www.abclinuxu.cz/kdo-je/alexander-viro>
- [272] <http://www.abclinuxu.cz/slovník/lkml>
- [273] <http://kerneltrap.org/mailarchive/linux-kernel/2008/1/8/546884>