

IDŹ DO

PRZYKŁADOWY ROZDZIAŁ



SPIS TREŚCI

KATALOG KSIĄŻEK

KATALOG ONLINE

ZAMÓW DRUKOWANY KATALOG

TWÓJ KOSZYK

DODAJ DO KOSZYKA

CENNIK I INFORMACJE

ZAMÓW INFORMACJE
O NOWOŚCIACH

ZAMÓW CENNIK

CZYTELNIA

FRAGMENTY KSIĄŻEK ONLINE

Ubuntu. Oficjalny podręcznik

Autorzy: Benjamin Mako Hill, Jono Bacon,
Corey Burger, Jonathan Jesse, Ivan Krstić

Tłumaczenie: Adam Bąk, Przemysław Szeremiota

ISBN: 978-83-246-0846-1

Tytuł oryginału: [The Official Ubuntu Book](#)

Format: B5, stron: 424



Poznaj i zrozum fenomen Ubuntu

Ubuntu to dystrybucja Linuksa, która przebojem wdarła się na rynki oprogramowania, zyskując coraz liczniejsze grono zwolenników. Ubuntu jest systemem operacyjnym przeznaczonym głównie dla użytkowników domowych, zbudowanym w oparciu o sprawdzoną i stabilną dystrybucję Debian. Podobnie jak inne dystrybucje Linuksa, Ubuntu dostępny jest nieodpłatnie. Został tak zaprojektowany, że pozwala na uruchomienie i efektywne wykorzystanie systemu bezpośrednio po instalacji. Dzięki możliwości współpracy z pakietami instalacyjnymi dystrybucji Debian użytkownik może dostosować system do własnych potrzeb i upodobań.

Książka „Ubuntu. Oficjalny podręcznik” to napisany przez twórców tej dystrybucji przewodnik, który pozwoli Ci dołączyć do społeczności użytkowników oprogramowania. Podczas lektury poznasz historię Ubuntu, dowiesz się, jak go zainstalować, skonfigurować, uruchomić i jak wykorzystać jego możliwości. Nauczysz się administrować systemem, instalować nowe oprogramowanie, zarządzać kontami użytkowników i zabezpieczać komputer przed utratą danych i atakami hakerów. Przeczytasz również o przyczynach i rozwiązaniach typowych problemów z Ubuntu oraz graficznym środowisku pracy Kubuntu.

- Historia dystrybucji Buntu
- Pobieranie wersji instalacyjnych
- Instalacja i konfiguracja systemu
- Korzystanie z WWW i poczty elektronicznej
- Zmiana wyglądu pulpitu
- Instalowanie i usuwanie pakietów
- Administracja systemem
- Drukowanie
- Uruchamianie serwera sieciowego w oparciu o Ubuntu
- Praca w środowisku Kubuntu
- Współpraca ze społecznością Ubuntu





Spis treści

Przedmowa	15
Wstęp	19
Podziękowania	21
O autorach	23
Wprowadzenie	25
Witamy w „Ubuntu. Oficjalny podręcznik”!	25
Informacje o książce	25
Zakres książki	26
Menu	26
Rozdział 1. Wprowadzenie do Ubuntu	29
Dzika jazda	30
Wolne oprogramowanie, oprogramowanie o otwartych źródłach, GNU/Linux	31
Wolne oprogramowanie i GNU	31
Linux	32
Oprogramowanie o otwartych źródłach — open source	33
Krótka historia Ubuntu	34
Mark Shuttleworth	35
Warthogs, czyli Guźce	36
Co oznacza Ubuntu?	37
Utworzenie Canonical	38
Czym jest Ubuntu?	39
Czym jest dystrybucja?	39
Ekosystem dystrybucji	41

Debian i wszechświat wolnego oprogramowania	42
Spółeczność Ubuntu	43
Cele i obietnice Ubuntu	44
Założenia filozoficzne	44
Kodeks Postępowania Ubuntu	46
Cele techniczne	47
Błąd #1	49
Canonical i Fundacja Ubuntu	50
Canonical Ltd.	50
Usługi świadczone przez Canonical oraz wsparcie techniczne	51
Bazaar i Launchpad	52
Fundacja Ubuntu	53
Podprojekty Ubuntu, dystrybucje pochodne i inne	54
Podsumowanie	55
Rozdział 2. Instalacja Ubuntu	57
Wybór odpowiedniej wersji Ubuntu	58
Inne dystrybucje Ubuntu	59
Czy to wciąż Ubuntu?	59
Zdobywanie Ubuntu	60
Nagrywanie płyty CD	60
Instalacja z płyty desktop CD	62
Wybór języka	63
Konfigurowanie układu klawiatury	63
Lokalizacja	64
Identyfikacja	65
Miejsce na dysku	67
Czynności końcowe	70
Instalacja z użyciem alternate CD	71
Instalacja sieci	72
Tworzenie partycji	73
Konfigurowanie konta użytkownika	76
Kończenie instalacji	76
Czynności poinstalacyjne	77
Konfigurowanie ekranu logowania	77
Konfigurowanie drukarki	78
Podsumowanie	79
Rozdział 3. Używanie Ubuntu na komputerach biurkowych	81
Pierwsza przejażdżka z Ubuntu	82
Uruchamianie aplikacji i wyszukiwanie	84
Odnajdywanie plików i folderów	86
Konfigurowanie systemu	87
Ikony dowiązań	87

Aplety	88
Obszar powiadamiania	88
Zegar	89
Pasek zadań	89
Wyłączanie komputera i wylogowywanie użytkownika	90
Używanie aplikacji	92
Przeglądanie internetu za pomocą Firefoksa	92
Tworzenie dokumentów za pomocą OpenOffice.org	96
Evolution — poczta elektroniczna i terminarz	98
Tworzenie grafiki za pomocą programu GIMP	103
Komunikator Gaim	109
Ekiga — zaawansowany technologicznie komunikator głosowy	111
Eksploracja Ubuntu	118
Zakładki i wybieranie plików	126
Ubuntu w różnych językach	128
Dostosowywanie wyglądu systemu Ubuntu	128
Zmiana tła pulpitu	129
Zmiana motywu	129
Konfigurowanie wygaszacza ekranu	130
Zarządzanie plikami	131
Przechowywanie i organizacja plików w Linuksie	131
Wybieranie, kopiowanie i przenoszenie plików i folderów	134
Używanie panelu paska bocznego	135
Używanie symboli	136
Ubuntu i multimedia	136
Instalowanie kodeków	136
Odtwarzanie plików audio	138
Odtwarzanie i zgrywanie płyt CD	140
Odtwarzanie plików wideo	141
Podsumowanie	143
Rozdział 4. Zaawansowane zarządzanie i korzystanie z Ubuntu	145
Dodawanie i usuwanie programów i pakietów	146
Wykorzystanie opcji Dodaj/Usuń	146
Terminologia	148
Instalacja oprogramowania z wykorzystaniem Synaptica	149
Aktualizowanie systemu	153
Instalowanie aktualizacji	153
Szczegółowe informacje o aktualizacjach	154
Użycie Synaptica do sprawdzania aktualizacji	155
Aktualizacja do nowszego wydania systemu	156
Aktualizacja systemu do najnowszej wersji	156

Korzystanie z urządzeń zewnętrznych i mediów	157
Używanie pamięci USB	157
Nagrywanie płyt CD	158
Używanie stacji dyskieta	159
Korzystanie z aparatów cyfrowych Ubuntu	159
Konfigurowanie drukarki w Ubuntu	160
GNOME CUPS Menedżer — łatwiej i prościej	160
Kilka informacji o używaniu GNOME	161
Gromadzenie informacji	161
Uruchamianie kreatora konfiguracji	161
Drukowanie zdalne	164
Misja zakończona	164
Okno drukarki	164
Zdalny graficzny dostęp do plików	166
Terminal	167
Intensywny kurs obsługi terminala	168
Współpraca z Windows	171
Uruchamianie aplikacji	172
Korzystanie z plików znajdujących się na partycjach Windows	172
Podsumowanie	173

Rozdział 5. Serwer Ubuntu 175

Czym jest Ubuntu Server?	176
Instalowanie serwera Ubuntu	178
Kilka sztuczek instalatora	179
Partycjonowanie	179
Historia RAID	180
Zakładanie macierzy RAID	182
Historia LVM	184
Ustawianie LVM	186
Prawie gotowe — ale precz od konta root!	188
Zarządzanie pakietami w Ubuntu	188
Archiwum Ubuntu	189
Źródła i repozytoria APT	189
dpkg	190
Ręczne instalowanie pakietu	191
apt-get i apt-cache	193
Aktualizacja wydania dystrybucji	195
aptitude	198
Sztuczki i kruczki	198
Bezpieczeństwo serwera Ubuntu	199
Zarządzanie kontami użytkowników	200
Bezpieczeństwo systemu plików	201
Reglamentowanie zasobów systemowych	203

Pliki dzienników systemowych	204
Słowo o bezpieczeństwie sieci	205
Ostatnie słowa na temat bezpieczeństwa	206
Podsumowanie	206

Rozdział 6. Wsparcie techniczne i typowe problemy 209

System	211
Ubuntu nie chce się uruchomić!	211
Podczas uruchamiania systemu logo Ubuntu jest uszkodzone	212
Po uruchomieniu komputera zamiast trybu graficznego dostępny jest tylko tryb tekstowy ...	213
Czcionki z dokumentów Worda i PowerPointa wyglądają źle	214
Jak zainstalować pakiety?	215
Instalacja aplikacji niedostępnych w programie Synaptic	216
Wyczerpane repozytoria	217
Nautilus działa bardzo powoli — jak go przyspieszyć?	217
Dodawanie czcionek typu True Type	218
Jak sprawdzić poprawność pliku iso?	219
Pobrałem autopakiet, ale nie wiem, jak go uruchomić	220
Jak skompilować program?	220
Nie widzę ukrytych plików i folderów w menedżerze plików	221
Jak przywrócić usunięte z menedżera plików elementy?	222
System graficzny się zawiesił — co robić?	222
Rozdzielczość ekranu jest niewłaściwa	223
Automatyczne logowanie bez konieczności podawania nazwy użytkownika i hasła	223
Podczas próby aktualizacji systemu pojawia się komunikat o błędzie	223
Kończy się miejsce na dysku, jak odzyskać go choć trochę?	224
Pomimo usunięcia plików przez menedżera plików ilość wolnego miejsca nie uległa zwiększeniu	225
Jak zaktualizować system do nowej wersji Ubuntu?	225
Programy	226
Używanie ikony zamykania programu nie przynosi efektu	226
Jak wyłączyć komunikat o dostępnych aktualizacjach?	226
Poszerzanie możliwości Nautilusa	226
Podczas korzystania z Firefoksa okazało się, że brakuje wtyczki Macromedia Flash	228
Instalacja Javy	228
Podobno przeszukiwanie komputera jest super — jak to zainstalować?	228
Poczta elektroniczna nie działa w Evolution	230
Multimedia	231
Pobrałem pliki multimedialne, ale nie mogę ich odtworzyć	231
Brak możliwości odtwarzania płyt DVD	231
Obraz z płyt DVD jest niestabilny	232
Podczas uruchamiania aplikacji Ubuntu wyświetla komunikat o braku dostępu do /dev/dsp	232
Mój mikrofon nie działa	233

Jak można zmienić motyw graficzny systemu?	234
Gdzie znaleźć i skąd pobrać nowe tematy i tła pulpitu?	235
Jak zamienić komputer w MythTV Box?	236
Sieć	237
Brak dostępu do sieci	237
Problemy z serwerami nazw	238
Jak używać ssh do bezpiecznego przesyłania plików poprzez sieć?	238
Zdalny graficzny dostęp do aplikacji za pośrednictwem ssh	239
Moja karta bezprzewodowa nie działa	239
Korzystanie ze standardów WPA i WPA-PSK	241
Sprzęt	242
Ubuntu nie wykrył mojej starej karty dźwiękowej	242
Urządzenie cardbus nie zostało rozpoznane	244
Po podłączeniu pamięci USB nic się nie dzieje	245
Skopiowane na pamięć USB pliki są niewidoczne po podłączeniu do innego komputera	246
Napęd CD/DVD nie działa	246
Napęd CD/DVD nie chce się otworzyć	247
Kupiłem urządzenie, ale nie działa ono w Ubuntu	247
Brak pamięci	248
Jak skopiować do Ubuntu zdjęcia z telefonu komórkowego za pomocą Bluetooth?	248
Mogę odczytywać pliki z zewnętrznego nośnika pamięci USB, ale nie mogę ich tam zapisywać	248
Zabawy z systemem plików	249
Jak sformatować dysk?	250
Problemy z klawiaturą	251
Mysz podłączona do portu szeregowego nie działa	251
Rolka przewijania na myszy nie działa	252
Zdalne sterowanie nie działa	252
Gdzie sprawdzić, czy sprzęt, który zamierzam kupić, będzie działał w Ubuntu?	253
Administrowanie systemem	253
Jak zaplanować wykonywanie czynności?	253
Jak skopiować pliki z jednego komputera na drugi?	256
Wiem, że aplikacja jest dostępna w Ubuntu, ale nie mogę jej znaleźć w Synapticu	256
Korzystam z Ubuntu na starszym komputerze i chciałbym przyspieszyć działanie systemu	256
Po reinstalacji Windows Ubuntu nie jest dostępne	258
Jak naprawić dysk po awarii zasilania?	259
Ubuntu zużywa zbyt wiele przestrzeni dyskowej na moim starym komputerze	260
Mój komputer działa bardzo powoli — jak mogę sprawdzić, co jest tego przyczyną?	261
Gdzie mogę znaleźć listę dostępnych opcji dla poleceń?	261
Jak odzyskać konto superużytkownika?	261
Zapomniałem swojego hasła do systemu, co robić?	262
Jak uzyskać dostęp do partycji systemu Windows?	263
Ubuntu wolno działa na moim procesorze AMD K7	264
Jak dodać użytkownika?	264

Inne	264
Uruchamianie w Ubuntu innego systemu operacyjnego	265
Mam zgłosić raport o błędach, ale nie wiem jak	266
Jak mogę sprawdzać pogodę?	268
Jak zrobić chleb Ubuntu?	268
Jak zapobiec bólowi dłoni podczas pisania na klawiaturze?	270
Podsumowanie	270
Rozdział 7. Kubuntu	271
Wprowadzenie do Kubuntu	272
Historia KDE	273
Historia Kubuntu	274
Poruszanie się po Kubuntu	275
Wyłączanie komputera i kończenie sesji	276
Instalowanie Kubuntu	277
Gdzie znaleźć Kubuntu?	277
Czy można zamienić zainstalowane już Ubuntu w Kubuntu?	278
Instalacja z przewodnikiem	278
Instalowanie Kubuntu z płyty Desktop CD	279
Korzystanie z sudo	282
Dostosowywanie Kubuntu	283
Dostosowywanie pulpitu	283
GHNS, czyli dostawa towaru	284
Dostosowywanie apletów i panelu	285
Dostosowywanie menu K	286
Administrowanie systemem	287
Instalowanie nowych pakietów	287
Zarządzanie repozytoriami	291
Instalowanie pakietu	293
Uaktualnianie Kubuntu	296
Jak dbać o aktualność systemu	296
Ustawienia systemowe	297
Zarządzanie plikami w Kubuntu	303
Wprowadzenie do Konquerora	303
Wyszukiwanie plików i katalogów	305
Zgrywanie płyt CD-Audio	305
Dostęp do partycji systemu Windows	306
Dostęp do napędów USB	307
Zarządzanie zbiorami muzycznymi	307
Popularne aplikacje	308
OpenOffice.org	309
Przeglądanie WWW	309
Przeglądanie WWW za pomocą Firefoksa	311
Nagrywanie płyt CD (z muzyką i danymi)	311

KAudioCreator	312
Komunikatory internetowe	314
Kontakt	315
Krita	319
Oglądanie filmów i odtwarzanie płyt muzycznych	321
IRC	322
Tryb Kiosku	323
Wycieczka po krainie Kubuntu	325
Sztuczki i kruczki	327
Szukanie pomocy i współpraca ze społecznością	329
Szukanie pomocy	329
Podręczniki pomocy Kubuntu	330
Współpraca ze społecznością	331
Podsumowanie	331

Rozdział 8. Społeczność Ubuntu 333

Kanały	335
Listy dystrybucyjne poczty elektronicznej	335
IRC (ang. Internet Relay Chat)	339
Lista kanałów IRC	340
Fora WWW	343
Wiki	346
Serwis The Fridge	347
Konferencje i spędy	349
Planet	350
Zespoły, procesy i nadzór	352
Zespoły	353
Reprezentacje społeczności lokalnych	353
MOTU	354
Rada Społeczności Ubuntu	355
Rada Techniczna	356
Samozwańczy Łaskawy Dożywotni Dyktator Ubuntu	357
Ubunteros i członkowie Ubuntu	358
Jak się przyłączyć?	358
Orędownictwo	359
Wsparcie	359
Pomysły i propozycje	359
Dokumentacja	360
Oprawa	360
Tłumaczenia i lokalizacje	360
Zapewnianie jakości	361
Programowanie i przygotowywanie pakietów	361
Podsumowanie	362

Rozdział 9. Projekty Ubuntu	363
Projekty partnerskie	364
Kubuntu	364
Edubuntu	365
Dystrybucje pochodne	367
Guadalinex	368
Xubuntu	368
Nexenta	369
nUbuntu	369
Ufficio Zero	370
The Open CD	370
Baltix	370
ImpiLinux	371
Launchpad	371
Soyuz	372
Rosetta	373
Malone	374
Inne komponenty infrastruktury	375
Bazaar	375
Podsumowanie	376
 Dodatek A Witamy w wierszu poleceń	 377
Uruchamianie terminala	377
Poruszanie się po systemie plików	378
Manipulowanie plikami i katalogami	379
Polecenia informacji o systemie	380
Przeszukiwanie i edytowanie plików tekstowych	381
Obsługa kont użytkowników i grup	382
Pomoc w wierszu poleceń	383
Przeszukiwanie stron dokumentacji systemowej	384
Stosowanie symboli wieloznacznych	384
Uruchamianie wielu poleceń	385
Uruchamianie sekwencyjne	385
Przekazywanie wyjścia	386
Zaawansowane zastosowania wiersza poleceń	386
Książki i strony WWW	386
 Dodatek B Dokumenty Ubuntu	 387
Kodeks Postępowania Ubuntu	387
Wstęp	388
Kodeks Postępowania Ubuntu	388
Listy i fora dyskusyjne	390

Filozofia Ubuntu	390
Wolne oprogramowanie i otwarte źródła	390
Wolne oprogramowanie	391
Otwarte źródła	391
Komponenty Ubuntu	392
Komponent „main”	392
Komponent „restricted”	393
Komponent „universe”	393
Komponent „multiverse”	394
Polityka Licencyjna Ubuntu	394
Polityka Licencyjna Komponentów „main” i „restricted” Ubuntu	396
Dokumentacja, firmware i sterowniki	397
Oprogramowanie instalowane domyślnie	398
Dodatek C Licencja	399
Dodatek D Odpowiedniki programów dla Windows w Ubuntu	405
Ubuntu na biurku	405
Przetwarzanie tekstu	405
Arkusze kalkulacyjne	406
Prezentacje	406
Bazy danych	406
Przeglądarki WWW	406
Poczta elektroniczna	406
Odtwarzacze mediów	407
Edycja fotografii	407
Komunikatory internetowe	407
Telefonia internetowa (VoIP)	407
Aplikacje dodatkowe	408
Biuro i finanse	408
Modelowanie i rysunki	408
Gry i edukacja	409
Skorowidz	411

Rozdział 1

Wprowadzenie do Ubuntu



- **Dzika jazda**
- **Wolne oprogramowanie, oprogramowanie o otwartych źródłach, GNU/Linux**
- **Krótką historia Ubuntu**
- **Czym jest Ubuntu?**
- **Cele i obietnice Ubuntu**
- **Canonical i Fundacja Ubuntu**
- **Podprojekty Ubuntu, dystrybucje pochodne i inne**
- **Podsumowanie**

Niniejszy rozdział jest wprowadzeniem do projektu Ubuntu: dystrybucji i procesu jej tworzenia, zawiera też trochę faktów z historii, dzięki którym to wszystko stało się możliwe. Jeżeli jednak chciałbyś pominąć te informacje i od razu rozpocząć przygodę z Ubuntu, powinieneś przejść do rozdziału 2., „Instalowanie Ubuntu”. Jeśli jednak zależy Ci mimo wszystko na tym, aby najpierw dowiedzieć się, skąd wzięło się Ubuntu i dokąd zmierza, w rozdziale tym znajdziesz informacje będące dobrym wstępem.

Dzika jazda

W kwietniu 2004 roku Mark Shuttleworth zwołał na burzę mózgów tuzin programistów projektów Debian, GNOME i GNU Arch. Zadał im pytanie, czy możliwy jest lepszy typ systemu operacyjnego (ang. *operating system* — OS). Odpowiedź brzmiała — tak. Następnym pytaniem Shuttlewortha było, jak powinien on wyglądać. Poprosił też o opis społeczności, która mogłaby utworzyć taki system. Cała grupa pracowała nad opracowaniem odpowiedzi na postawione pytania, po czym zdecydowano, aby spróbować zamienić teoretyczne odpowiedzi w praktyczne wykonanie. Grupa przybrała nazwę Warthogs¹ i dała sobie sześć miesięcy na zbudowanie systemu operacyjnego potwierdzającego zasadność przyjętej tezy. Pierwsze wydanie nosiło nazwę Warty Warthog, ponieważ rozsądnie przyjęto, że ten produkt może mieć usterki². Tak oto zaistnieli w biznesie.

Trudno uwierzyć — zwłaszcza tym, którzy mieli przywilej znaleźć się pośród pierwszych Guźców — że od pierwszego spotkania dotyczącego projektu Ubuntu upłynęło mniej niż dwa lata. Piegowaty Guziec okazał się mieć mniej piegów, niż zakładano; przekroczył nasze najbardziej optymistyczne założenia i wszystkie nasze najśmielsze oczekiwania. W ciągu sześciu miesięcy stał się numerem jeden pod względem popularności w kilku rankingach dystrybucji GNU i Linuksa. Ubuntu zanotowało najbardziej dynamiczny wzrost w dotychczasowej historii, a pierwszy rok istnienia można zaliczyć do najbardziej imponujących początków jakiegokolwiek projektu wolnego oprogramowania w historii.

To zdumiewające, że po niespełna dwóch latach miliony ludzi używają Ubuntu. Tysiące z tych użytkowników codziennie ulepszają dystrybucję poprzez opracowywanie dokumentacji, tłumaczeń czy kodów. Tysiące współuczestniczą w rozwijaniu i wspieraniu społeczności, zarówno wirtualnej, jak i tej zupełnie realnej. Rozwój Ubuntu pozostaje niepowstrzymany. Lista projektów wywodzących się z Ubuntu (zawierająca wchodzące właśnie w fazę dojrzałości Kubuntu i Edubuntu) rozszerza się, osiągając cele podstawowego projektu na nowych obszarach.

¹ Warthog to po angielsku guziec; ssak parzystokopytny z rodziny świniowatych występujący w Afryce — *przyp. tłum.*

² Wart to po angielsku brodawka, narośl, ale też przeszkoda, problem — stąd pojawia się w nazwie Warty. W Polsce przyjęło się tłumaczenie nazwy kodowej Warty Warthog jako Piegowaty Guziec — *przyp. tłum.*

Tymczasem miliony wytłoczonych płyt CD z Ubuntu zostały bezpłatnie dostarczone na uniwersytety, do kawiarenek internetowych, sklepów komputerowych i różnego rodzaju organizacji na całym świecie. Przyjazną brązową tapetę i paski tytułowe Ubuntu można znaleźć niemal wszędzie tam, gdzie ludzie korzystają z komputerów. Autor osobiście widział ludzi korzystających z Ubuntu w pociągach w Hiszpanii, bibliotekach w Bostonie, muzeach w Chorwacji, szkołach w Meksyku i w wielu innych miejscach, których jest zbyt wiele, aby je tutaj wymienić.

W ciągu dwóch lat Ubuntu zaczęło dojrzewać. Ubuntu 6.06 Dapper Drake to dopracowane wydanie z długoterminowym wsparciem zarówno dla komputerów biurkowych, jak i serwerów. Pomimo tego, że Ubuntu zaczyna zagnieżdżać się na dłużej w komputerach, dystrybucja nie straciła nic z młodzieńczego wigoru, ambitnego podejścia, oddania zasadom i ukierunkowania na społeczność. Wraz z rozwojem jasne staje się, że projekt wyciąga wnioski zarówno z porażek, jak i sukcesów, dzięki czemu może się rozwijać bez kompromisów w dziedzinie stabilności. Przebyliśmy długą drogę, ale wciąż jesteśmy na jej początku.

Wolne oprogramowanie, oprogramowanie o otwartych źródłach, GNU/Linux

Podczas gdy tysiące osób na różne sposoby uczestniczy w Ubuntu, sam projekt korzysta z wkładu kolejnych wielu tysięcy osób, które położyły techniczne, społeczne i ekonomiczne podwaliny pod jego sukces. Informacje na temat wolnego oprogramowania, oprogramowania o otwartych źródłach i GNU/Linux można znaleźć w wielu miejscach, jednak żadne wprowadzenie do Ubuntu nie będzie kompletne bez choćby krótkiego omówienia tych koncepcji, stojących za nimi ludzi i historii. To te idee oraz społeczność dała motywację i przyczyniła się do narodzin Ubuntu. To dzięki nim Ubuntu istnieje.

Wolne oprogramowanie i GNU

W serii wydarzeń, które dzięki ciągłemu powtarzaniu stały się już niemal legendą, w roku 1983 Richard M. Stallman stworzył pojęcie „wolnego oprogramowania”. Stallman dorastał wraz komputerami w latach 60. i 70., kiedy używanie tych maszyn oznaczało konieczność zakupu ogromnych i niesłychanie drogich urządzeń typu *mainframe*, a na jeden komputer przypadło wielu programistów. Oprogramowanie było postrzegane wówczas jako dodatek do komputera, a każdy użytkownik posiadał umiejętność oraz prawo do modyfikowania i przepisywania programów, mógł też swobodnie się nimi dzielić z innymi. W latach 70. komputery zaczęły być coraz tańsze i liczniejsze, a producenci oprogramowania zaczęli dostrzegać wartość oprogramowania jako takiego. Producenci komputerów zaczęli

argumentować, że programy stanowią własność intelektualną i należy je objąć prawami autorskimi, podobnie jak muzykę, filmy czy książki. Rozpoczęto wówczas dystrybuowanie oprogramowania z licencjami, które ograniczały prawa użytkowników do jego wykorzystywania, redystrybuowania czy modyfikowania kodu źródłowego. We wczesnych latach 80. restrykcyjne licencje dołączane do oprogramowania były już normą.

Stallman, wówczas programista w MIT Artificial Intelligence Laboratory, coraz bardziej zaczął skupiać się na tym, co postrzegał jako ryzyko utraty wolności, którą dotychczas cieszyli się programiści i użytkownicy oprogramowania. Zajmował się też umiejętnością tworzenia przez samych użytkowników komputerów dobrosąsiedzkich relacji pomiędzy współuczestnikami sprawnej i działającej zgodnie z przyjętą etyką społeczności. Aby walczyć z negatywnymi tendencjami, Stallman przedstawił wizję społeczności, która tworzy wolny kod czy, używając jego słów, „wolne oprogramowanie”. Wedle jego definicji, za wolne oprogramowanie można uznać takie, które spełnia cztery warunki (oznaczone od zera do trzech — jest to zabawne nawiązanie do języków programowania):

- wolność uruchamiania programów w dowolnym celu (wolność 0.),
- wolność w zapoznawaniu się z mechanizmem działania programu oraz swoboda w jego dopasowywaniu do własnych potrzeb (wolność 1.),
- wolność w rozpowszechnianiu oprogramowania, tak aby móc pomagać innym (wolność 2.),
- wolność do ulepszania oprogramowania i dzielenia się poprawionymi wersjami z innymi, tak aby cała społeczność czerpała z tego korzyści (wolność 3.).

Dostęp do kodu źródłowego — do każdego jego fragmentu w postaci czytelnej dla ludzi, w odróżnieniu od wersji kodu czytelnej tylko dla maszyn, a w takiej postaci dystrybuowana jest większość oprogramowania — to warunki wstępne do wolności 1. i 3. Jako uzupełnienie ogłoszonej definicji wolnego oprogramowania Stallman ogłosił projekt utworzenia całkowicie „wolnego” systemu operacyjnego, który zastąpiłby — popularny w latach 80. — UNIX. Projekt został ogłoszony w roku 1984 i nazwany „GNU”; był to kolejny żart językowy, ponieważ rozwinięciem GNU jest *GNU's Not Unix* (GNU to Nie UNIX).

Linux

We wczesnych latach 90. Stallman i inni programiści pracujący nad GNU zaprojektowali niemal kompletny system operacyjny, który mógł być darmowo rozpowszechniany. Jedynym, czego brakowało, był ostateczny i podstawowy element w postaci jądra (ang. *kernel*) — kompletnego zestawu poleceń, które znajdują się w centrum każdego systemu operacyjnego. W roku 1991 Linus Torvalds napisał

wczesną wersję jądra opublikował ją na „wolnej” licencji pod nazwą „Linux”. *Kernel* Linusa został połączony z narzędziami opracowywanymi w ramach projektu GNU oraz z graficznym systemem okien zwanym X. Dzięki temu zestawowi powstał kompletny system operacyjny: wolny zarówno od wszelkich opłat, jak i zgodny z zasadami wyznawanymi przez Stallmana.

Wszystkie dzisiejsze systemy, o których mówi się „Linux”, opierają się na wspomnianym połączeniu różnych elementów. Pod względem technicznym termin „Linux” odnosi się tylko do jądra systemu. Wielu programistów oraz współuczestników GNU uważa, że system operacyjny powinien właściwie nazywać się „GNU/Linux”, co podkreślałoby wkład projektu GNU oraz zasługi w propagowaniu wolności oprogramowania, w czym niekoniecznie uczestniczył Torvalds. Jednak dla wielu taka nazwa jest dość osobliwa i dlatego wolą prostszą nazwę Linux. Innym rozwiązaniem umożliwiającym uniknięcie kontrowersji związanych z nazwą (tak jest na przykład w Ubuntu) jest używanie tylko samej nazwy projektu.

Oprogramowanie o otwartych źródłach — open source

Dyskusje o nazwach nie zakończyły się tylko na kwestiach nazwy i kombinacji słów GNU i Linux. Ponieważ lista uczestników GNU i Linuksa powiększa się, świat nowych projektów wolnego oprogramowania rozszerza się, w dużej części dzięki coraz powszechniejszemu dostępowi do internetu. Wraz z rozwojem społeczności i jej dywersyfikacją coraz więcej osób zaczyna dostrzegać niezamierzony efekt idei wolnego oprogramowania Stallmana. Ponieważ jest ono otwarte, każdy może współuczestniczyć w jego rozwoju poprzez czytanie kodu, odnajdywanie błędów i ich usuwanie. Dzięki temu, że wolne oprogramowanie jest testowane przez dużą liczbę programistów, oferuje lepszą jakość, wydajność i więcej funkcji aniżeli podobne oprogramowanie tworzone według standardowych mechanizmów tworzenia programów. W wielu sytuacjach model tworzenia oprogramowania stojący za wolnym oprogramowaniem daje w efekcie lepsze efekty aniżeli tradycyjne metody.

Kiedy przemysł informatyczny i komputerowy wchodził w erę boomu tzw. „dot-comów”, jedna z grup twórców wolnego oprogramowania, na czele której stało dwóch programistów i rzeczników ruchu, Eric S. Raymond i Bruce Perens, dostrzegła możliwości oferowane przez model pracy oparty na pracy wolontariuszy czy współpracy między różnymi firmami w tworzeniu oprogramowania. Zmartwieciem był jednak sam termin „wolne oprogramowanie” i to przynajmniej z dwóch powodów. Po pierwsze, z powodu jego dwuznaczności — angielskie słowo „free”

oznacza zarówno „bezpłatny”, jak i „wolny” w sensie wolności właśnie. A po drugie, istniała obawa, wyrażona przez Raymonda, że wszystkie rozmowy o „wolności” odstraszą od projektów osoby podejmujące decyzje w dużych firmach. Decydentów do wolnego oprogramowania mógł przekonać tylko odniesiony sukces.

Aby rozwiązać ten problem, grupa utworzyła nowy termin: oprogramowanie o otwartym kodzie (*open source*) oraz powołała nową organizację — Open Source Initiative. Jako podstawę swej działalności grupa przyjęła definicję *open source* zazębiającą się z czteroczęściową definicją Stallmana, a także innymi, również bazującymi na osiągnięciach tego samego Stallmana.

Aby nieco rozjaśnić różnicę pomiędzy wolnym oprogramowaniem a oprogramowaniem o otwartym kodzie, można posłużyć się analogią do schizmy. Schizma religijna ma miejsce wówczas, kiedy kościoły nie współpracują z powodu drobnych różnic w przekonaniach, interpretacji czy motywacji. I tak protestanci i katolicy zgadzają się niemal we wszystkim, ale dzieli ich kilka drobnych kwestii, które jednak są nie do pogodzenia. W przypadku wolnego oprogramowania i oprogramowania o otwartym kodzie istnieje podstawowa różnica w przypadku motywacji i przekonań obydwu ruchów. Jedna grupa skupiona jest na wolności, druga na kwestiach bardziej pragmatycznych. Wolne oprogramowanie najlepiej opisuje termin grupy społecznej, natomiast oprogramowanie o otwartym kodzie to raczej metodologia tworzenia. Jednakowoż obie te grupy nie mają kłopotów ze wspólną pracą nad różnymi projektami.

Niemniej jednak w zakresie motywacji i celów wolne oprogramowanie i oprogramowanie o otwartym kodzie różnią się wyraźnie, natomiast w zakresie licencjonowania czy używania projektów nie ma różnic. Dla części osób ten konflikt jest dziwny: uczestnicy projektu Ubuntu nie widzą żadnego konfliktu. Ludzie często identyfikują się z obiema grupami, rzadziej tylko z jedną. W niniejszej książce często wykorzystuje się oba terminy dla opisanego różnych projektów, a współtwórcy Ubuntu często łatwiej identyfikują się z jednym lub drugim projektem. Na potrzeby niniejszej książki postanowiliśmy używać obu terminów zamiennie i tak należy je rozumieć, chyba że wyraźnie powiedziano inaczej.

Krótką historia Ubuntu

Ponieważ Ubuntu narodził się w kwietniu 2004 roku i w chwili pisania niniejszej książki miał dwa lata, pisanie historii systemu może wydawać się przedwczesne. Jednak ostatnie dwa lata obfitowały w wydarzenia. Dynamiczny wzrost utrudnia śledzenie najważniejszych momentów nawet dla osób znajdujących się bardzo blisko samego projektu. Co ważne, w projekcie uczestniczą pewne ważne postaci, a poznanie ich historii jest istotne dla pełnego zrozumienia Ubuntu. Niniejsze

krótkie podsumowanie ma za zadanie przedstawić Czytelnikowi najważniejsze momenty i daty w historii Ubuntu, stanowiące niezbędne tło do zrozumienia, skąd właściwie wzięło się Ubuntu.

Mark Shuttleworth

Żadna historia Ubuntu nie będzie kompletna bez historii Marka Shuttlewortha. Jest on — niezaprzeczalnie — najważniejszą i najbardziej widoczną osobą w projekcie. Co ważniejsze, z punktu widzenia historii, Shuttleworth jest również pomysłodawcą i inicjatorem projektu: to on rozpoczął coś, co można nazwać efektem śnieżnej kuli, a co ostatecznie przekształciło się w Ubuntu.

Shuttleworth urodził się w 1973 roku w Welkom, w prowincji Free State, w Republice Południowej Afryki. Uczęszczał do college'u diecezjalnego i uzyskał stopień naukowy w dziedzinie finansów i systemów informatycznych na uniwersytecie w Cape Town. Właśnie w tym okresie stał się gorliwym zwolennikiem nauk informatycznych, wtedy też miał do czynienia ze społecznościami zajmującymi się wolnym oprogramowaniem i oprogramowaniem o otwartych źródłach. Był również zaangażowany w projekty Apache i Debian i zasłynął tym, że jako pierwsza osoba umieścił Apache'a — być może jedną z najważniejszych części oprogramowania GNU/Linux dla serwerów — w repozytoriach Debiana.

Na wczesnym etapie rozwoju sieci dostrzegł drzemiące możliwości i założył firmę o nazwie Thawte, zajmującą się bezpieczeństwem w internecie. Po kilku latach Thawte była drugą po Verisign firmą pod względem wielkości zajmującą się certyfikatami w sieci. W tym okresie produkty i usługi świadczone przez Thawte niemal w całości oparte były na wolnym oprogramowaniu. W grudniu 1999 roku Shuttleworth sprzedał Thawte firmie Verisign za nieujawnioną sumę, która jednak szacowana jest na setki milionów dolarów.

Z fortuną zarobioną w tak młodym wieku Shuttleworth mógł oddać się czerpaniu przyjemności z życia — co prawdopodobnie wziął pod uwagę. Najpierw postanowił zrealizować swoje marzenie o podróży w kosmos. Po zapłaceniu około 20 milionów dolarów i spędzeniu prawie roku na przygotowaniach (włączając w to naukę rosyjskiego oraz siedmiomiesięczny pobyt w Gwiezdnym Mieście), spełnił swoje marzenie i jako cywilny kosmonauta wziął udział w rosyjskiej misji kosmicznej na pokładzie Sojuza TM-34. Shuttleworth spędził dwa dni w rakiecie Sojuz oraz osiem na pokładzie Międzynarodowej Stacji Kosmicznej, gdzie współuczestniczył w eksperymentach nad AIDS oraz poszukiwaniem genomu. Na początku maja 2002 roku Shuttleworth powrócił na Ziemię.

W roli dodatku do eksploracji kosmosu i nieco mniej widowiskowego wypadu na Antarktykę, Shuttleworth pełnił aktywną rolę jako filantrop i inwestor. W 2001 roku założył fundację (TSF), niedochodową organizację z siedzibą w RPA. Zadaniem

fundacji jest projektowanie, finansowanie i wdrażanie społecznych innowacji w dziedzinie edukacji. Oczywiście w swoich działaniach TSF często korzysta z osiągnięć ruchu wolnego oprogramowania. Dzięki tym projektom organizacja stała się jedną z najbardziej widocznych spośród zajmujących się wolnym oprogramowaniem w RPA, a nawet na świecie. W obszarze inwestycji Shuttleworth pracował nad rozwojem badań, projektowaniem i przedsięwzięciami w RPA poprzez dotacje finansowe dla nowych firm w ramach programu HBD. Nazwa to akronim od wyrażenia „*Here be dragons*”, co w wolnym tłumaczeniu oznacza „Tu są smoki”. W tym okresie Shuttleworth był również zajęty opracowywaniem nowego projektu, który ostatecznie przerodził się w Ubuntu.

Warthogs, czyli Guźce

Nie brakowało projektów, których celem było umieszczenie GNU, Linuksa i innych fragmentów wolnego oprogramowania i oprogramowania z otwartym kodem w schludne, łatwe w użyciu i przyjazne dla użytkownika pakiety. Mark Shuttleworth, podobnie jak wielu innych, wierzył, że filozoficzne i pragmatyczne korzyści oferowane przez wolne oprogramowanie gwarantują mu sukces. Jednak żadna z dotychczasowych propozycji nie była szczególnie imponująca. Wszystkim czegoś brakowało. W tym właśnie Shuttleworth dostrzegł szansę. Jeżeli ktoś mógłby zbudować wspaniałą, darmową dystrybucję, która pomogłaby umieścić GNU/Linuksa w głównym nurcie, mógłby w ten sposób osiągnąć pozycję o ogromnym znaczeniu.

Shuttleworth, podobnie jak wielu innych ludzi o technicznych inklinacjach, był zagorzałym fanem projektu Debian (omówionego nieco dalej). Jednakże w samym Debianie było wiele elementów, które nie mieściły się w wizji idealnego systemu operacyjnego według Shuttlewortha. Przez pewien czas Shuttleworth rozważał możliwość poprowadzenia projektu jako lider, co umożliwiłoby zreformowanie Debiana od środka. Z upływem czasu coraz wyraźniej widoczne było, że umieszczenie GNU/Linuksa w głównym nurcie, przy korzystaniu wprost z Debiana, nie jest najlepszym sposobem, choć w wielu sytuacjach faktycznie mogłoby być. Zamiast tego Shuttleworth wołał powołać nowy projekt, który działałby w symbiozie z Debianem, aby zbudować nowy, lepszy system GNU/Linux.

Aby rozpocząć projekt, w kwietniu 2004 roku Shuttleworth zaprosił tuzin (mniej więcej) programistów Debiana do swojego mieszkania w Londynie. To właśnie podczas tego spotkania (patrz: pierwszy akapit niniejszego wstępu) położone zostały podwaliny pod projekt Ubuntu. Od tej chwili wielu spośród zaproszonych zapaliło się do możliwości stworzenia nowego projektu. Podczas opisywanego spotkania zgromadzeni — którzy z czasem utworzyli rdzeń projektu — pracowali w trakcie burzy mózgów listę rzeczy, które chcieliby umieścić w swojej wizji idealnego systemu operacyjnego. W chwili obecnej ta lista jest dobrze znana wszystkim użytkownikom Ubuntu. Wiele z poniższych cech zostanie omówione nieco później. Oto, czego chciały zgromadzone osoby:

- przewidywalnych i częstych wydań,
- silnego ukierunkowania na lokalizację i dostępność,
- silnego ukierunkowania na łatwość użytkowania dla użytkowników komputerów biurkowych,
- silnego ukierunkowania na Pythona, jako na pojedynczy język programowania, na którym można oprzeć i rozwijać cały system,
- współpracy ze społecznościami pracującymi nad wolnym oprogramowaniem przez cały czas, a nie tylko w okresie wydawania nowej edycji,
- nowego zestawu narzędzi zaprojektowanych do procesu tworzenia dystrybucji, które umożliwią programistom sprawne działanie w ekosystemie różnych projektów, a użytkownikom współpracę w takiej formie, jaka będzie dla nich możliwa.

Grupa zaakceptowała powyższe cele, a ponieważ do jej członków bardziej przemawiają czyny niż słowa, obyło się bez upowszechniania tej informacji publicznie. Zamiast tego przyjęto nieprzekraczalny termin — sześć miesięcy. Shuttleworth zgodził się finansować projekt oraz płacić programistom pełnoetatowe wynagrodzenia za pracę nad projektem. Po pół roku chcieli ogłosić swój projekt i przedstawić pierwsze efekty swej pracy. Opracowano listę celów, które miały być osiągnięte w ustalonym terminie, a członkowie grupy podjęli się konkretnych zadań. Grupa przyjęła nazwę Warthogs, czyli Guźce.

Co oznacza Ubuntu?

Od tego momentu Guźce miały świetny zespół, listę celów oraz rozsądne pomysły, jak osiągnąć większość z nich. Z drugiej strony brakowało jeszcze nazwy dla nowego projektu. Shuttleworth przedstawił poważne argumenty za tym, aby projekt nazywać „Ubuntu”.

Ubuntu to idea i słowo zaczerpnięte z kilku południowoafrykańskich języków, między innymi Zulu i Xhosa. Odsyła ono do południowoafrykańskiej ideologii czy etyki, którą trudno oddać w języku angielskim, a którą można przetłumaczyć jako „humanitaryzm (człowieczeństwo) ukierunkowane na innych” czy też „istnienie, ponieważ istniejemy”. Inni opisują ubuntu jako „wiarę w uniwersalną więź, która łączy całą ludzkość”. Arcybiskup Desmond Tutu, południowoafrykański orędownik praw człowieka, objaśnia ubuntu następująco:

„osoba z ubuntu jest otwarta i dostępna dla innych, lubiana przez innych, nie czuje zagrożenia faktem, że inni są zdolni czy dobrzy, ponieważ posiada właściwą samoocenę oraz poczucie przynależenia do większej całości i jest ograniczana, kiedy inni są poniżani lub ograniczani, dręczeni czy uciskani”.

Ubuntu odgrywało ważną rolę jako podstawowa zasada w postapartheidowskiej Republice Południowej Afryki i pozostało ideą znaną większości dzisiejszych mieszkańców tego kraju.

Shuttleworth lubi pojęcie ubuntu, przyjęte jako nazwa dla nowego projektu, z kilku powodów. Po pierwsze, jest to afrykańska idea. Podczas gdy większość ludzi pracujących nad Ubuntu nie pochodzi z południowej Afryki, to jego korzenie tam właśnie się znajdują, a nazwa wybrana przez Shuttlewortha unaocznia to. Po drugie, projekt kładzie nacisk na definicję indywidualności w warunkach stosunków z innymi i proponuje głęboki typ społeczności i współdzielenia, czyli dokładnie taki, który jest rdzeniem współpracy i współdzielenia w społecznościach zajmujących się wolnym oprogramowaniem. Termin reprezentuje tę stronę wolnego oprogramowania, którą zespół chce dzielić ze światem. Po trzecie, idea osobistych relacji opartych na wzajemnym szacunku i związku opisuje podstawową zasadę wysoce funkcjonalnych społeczności, a takie właśnie chce zbudować zespół Ubuntu. Ubuntu to idea obrazująca, skąd pochodzi projekt, dokąd zmierza i jak to osiągnąć. Nazwa jest idealna. Nic dodać, nic ująć.

Utworzenie Canonical

Aby móc płacić programistom za pełnoetatową pracę nad Ubuntu, potrzebna była firma, która by ich zatrudniła. Shuttleworth chciał wybrać najlepszych ludzi spośród członków wszystkich społeczności zajmujących się wolnym oprogramowaniem i *open source*. Społeczności te, co mogło być pewną niedogodnością, nie znają granic geograficznych. Zamiast sprowadzać wszystkich w jedno miejsce i do jednego biura, Shuttleworth podjął decyzję o zatrudnieniu programistów w „wirtualnej firmie”. Pomimo wielu wad wynikających z dużych opóźnień na łączach o niskiej przepustowości, różnych stref czasowych i jeszcze wielu innych czynników, przyniosło to również pewne korzyści w wielu poszczególnych aspektach projektu. Z jednej strony rozczłonkowany sposób zatrudnienia oznaczał, że można angażować pracowników bez konieczności przeprowadzania ich do nowego kraju. Ale co ważniejsze, oznaczało to, że praca każdej osoby w firmie zależy od IRC-a, listy mailingowej i mechanizmów pracy w sieci. Bez wyraźnego zamiaru, choć automatycznie, rozwiązano problem, który był plagą wielu innych zbiorowych przedsięwzięć opartych na wolnym oprogramowaniu, mianowicie nadmiernego odosobnienia programistów podczas pracy, co do tej pory udawało się zwalczyć tylko podczas spotkań przy dystrybutorze wody.

Otóż, programiści woleli osobiście i swobodnie rozmawiać o swojej pracy i kompletnie odciąć od rozmów społeczność i wszystkich innych, którzy nie pracowali w biurze. Przez pierwszy rok miejscem, któremu było najbliższe do biura firmy Canonical, było mieszkanie Shuttlewortha w Londynie.

Z czasem firma przybrała nazwę Canonical. Nazwa koresponduje z optymistycznymi założeniami projektu, który ma się stać „kanonem”³ dla usług i wsparcia wolnego oprogramowania i *open source*. Słowo „kanon” odnosi się do czegoś, co jest uznawane za autorytatywne. To powszechne słowo w leksykonach komputerowych. Ważne jest, aby pamiętać, że „kanon” to inaczej „standard”, pozbawiony jednak przymusu. Kiedy coś staje się kanonem, oznacza to odniesienie w pewnym sensie sukcesu, ale inaczej niż w przypadku posiadania monopolu, kanon nie może zostać rozbity, ani też nie jest wyłączny. Inne firmy mogą wspierać Ubuntu i tworzyć systemy operacyjne w oparciu o niego, ale dopóki Canonical będzie dobrze wykonywać swoją pracę, będzie pełniło główną rolę.

Czym jest Ubuntu?

Celem Guźców i flagowym założeniem Canonical jest Ubuntu. Czytelnicy, którzy dotarli do niniejszego fragmentu, zapewne mają już pewne wyobrażenie o tym, co to oznacza. Kolejne akapity zawierają trochę informacji, które są pomocne w zrozumieniu, czym dokładnie jest Ubuntu i jakie przyswiecają jej cele.

Czym jest dystrybucja?

Dla większości osób jasne jest, że Ubuntu to system operacyjny. Jednak pełna historia jest nieco bardziej skomplikowana. Ubuntu jest tym, co nazywamy dystrybucją GNU/Linux. Zrozumienie, co to naprawdę oznacza, wymaga — ponownie — trochę historii. W początkach GNU i Linuksa użytkownicy musieli posiadać sporą wiedzę techniczną. Zwykle byli to geekowie. Nie istniał wówczas system operacyjny Linux w sensie, w jakim używamy tego terminu dzisiaj: nie było jednej płyty CD (czy zestawu kilku płyt), którą można było wykorzystać do instalacji. Zamiast tego oprogramowanie dostępne było jako setki osobnych programów, tworzonych przez różnych programistów i rozpowszechnianych oddzielnie. Instalowanie każdej aplikacji z osobna pochłaniało niesłychanie dużo czasu. W wielu przypadkach wzajemna niekompatybilność i konieczność stosowania różnych sztuczek, aby ominąć przeszkody, znacząco utrudniały zainstalowanie systemu GNU/Linux na dysku. Do zainstalowania i uruchomienia systemu niezbędna była spora porcja wiedzy. W rezultacie tylko bardzo niewielu ludzi niebędących programistami używało wczesnych systemów GNU/Linux.

Wczesne dystrybucje były projektami gromadzącymi w jednym miejscu wszystkie niezbędne programy, łącząc je w całość, której instalacja był prostsza i w przypadku których dokonywano wstępnej konfiguracji.

³ Słowo *canonical* oznacza „kanoniczny”, jednak aby uniknąć innych skojarzeń, dalej będziemy używać pojęcia „kanon”, które ma szerszy kontekst znaczeniowy — *przyp. tłum.*

Celem tych dystrybucji było uczynienie GNU/Linuksa wygodniejszym w użytkowaniu oraz dostarczenie go szerszej grupie użytkowników. Dzisiaj niemal nikt nie używa GNU/Linuksa bez korzystania z jakiejś dystrybucji. W efekcie tego nazwy poszczególnych dystrybucji stały się powszechnie znane. Ubuntu jest właśnie takim projektem. Inne popularne dystrybucje to Red Hat, SuSE firmy Novell, Turbo-Linux, Linspire, Gentoo i Debian.

Większość dystrybucji zawiera zbliżone zbiory oprogramowania. Wszystkie na przykład zawierają podstawowe elementy GNU oraz jądro Linux. Niemal wszystkie korzystają z systemu graficznego X, a w ich skład wchodzi zestaw aplikacji, na początku którego jest przeglądarka internetowa, środowisko graficzne i pakiet biurowy. Początkowo dystrybucje zawierały tylko podstawowe elementy systemu operacyjnego, teraz rozrosły się i obejmują również szeroki zakres dodatkowych aplikacji. Współczesne dystrybucje zawierają całe oprogramowanie, które system operacyjny „powinien zawierać”: kilka płyt CD lub DVD, na których znajduje się niemal wszystko, czego potrzebować może współczesny użytkownik i co jest prawnie dopuszczone do rozpowszechniania.

Ubuntu, podobnie jak inne współczesne dystrybucje, zawiera program ułatwiający instalację oraz szkielet zawierający oprogramowanie, a także adresy serwerów, z których można pobrać dodatkowe oprogramowanie, kiedy podstawowy system zostanie już uruchomiony. Posiada też standardowe metody konfigurowania programów oraz ujednolicone metody zgłaszania przez użytkowników wykrytych w programach błędów i wiele więcej. Dystrybucje często zawierają też ogromne repozytoria oprogramowania zgromadzonego na serwerach, a dostępnego poprzez internet. Aby mieć pojęcie o skali, warto wiedzieć, że Ubuntu zawiera 17 000 różnorodnych programów zgromadzonych na centralnych serwerach. Każda aplikacja jest przetestowana i dostosowana do współpracy z innymi elementami systemu. Ta liczba zwiększa się z każdym dniem.

Ważne jest, aby zdawać sobie sprawę z tego, że dystrybucja — w większości przypadków — nie tworzy oprogramowania, z którego korzysta użytkownik. Zespół Ubuntu nie napisał Linuksa, podobnie jak nie napisał GNU, chociaż jego członkowie przyczynili się do powstania obu projektów. Zamiast tego, zespół Ubuntu przetestował i zintegrował wiele aplikacji GNU, Linuksa i innych, tak aby były dostępne za pomocą jednego instalatora. Ubuntu to spoiwo, dzięki któremu można z jednej płyty CD zainstalować setki oddzielnych aplikacji pracujących później razem jako jeden spójny system operacyjny. Jeżeli sięgnąć po płytę CD z innej dystrybucji (Debiana, Red Hata czy Novella), zainstalowane oprogramowanie będzie niemal identyczne z zamieszczonym w Ubuntu. Różnice można znaleźć w sposobie instalacji programów, usług serwisowych, uaktualnieniach i sposobie ich zintegrowania z pozostałymi aplikacjami znajdującymi się w systemie.

Ekosystem dystrybucji

W dniu dzisiejszym istnieją setki aktywnych dystrybucji. Wystarczy rzut oka na bazę znajdującą się w witrynie Distrowatch (www.distrowatch.com), aby odnaleźć zdumiewającą — i wciąż powiększającą się — liczbę dystrybucji. Jedną z pierwszych nosiła nazwę Softlanding Linux System, w skrócie SLS. Z wielu różnych powodów programista Patrick Volkerding pomyślał, że mógłby ulepszyć SLS. Ponieważ SLS było wolnym oprogramowaniem, mógł swobodnie utworzyć pochodną oryginalnego systemu i ją rozprowadzać. Volkerding użył oryginalnego kodu SLS i wykorzystał go jako szkielet czy model, na którym oparł własny wariant, nazwany przez niego Slackware. Później Slackware stał się pierwszą dystrybucją GNU/Linuksa, która odniosła wielki sukces i jest utrzymywana do dziś.

Z czasem krajobraz dystrybucji GNU/Linux zaczął ulegać zmianom. Ważna rola różnorodnych odmian, która umożliwiła powstanie Slackware, pozostała nienaruszona i nadal kształtuje aktualną sytuację. Dzisiaj setki dystrybucji GNU/Linux służą całej rzeszy użytkowników do realizacji miliardów różnorodnych zadań. Są dystrybucje przeznaczone specjalnie dla dzieci, dentystów, naukowców, przedsiębiorców i dostępne w wielu różnorodnych językach. Są dystrybucje przeznaczone na serwery, na komputery kieszonkowe, dla organizacji pożytku publicznego, dla muzyków i dla niezliczonych innych grup.

Pomimo tej różnorodności korzenie większości istniejących dystrybucji można odnaleźć w dwóch „rodzicielskich” dystrybucjach: Red Hat i Debian. Nie jest niezbędne zrozumienie wszystkich różnic pomiędzy nimi, ale warto wiedzieć, że zarówno Red Hat, jak i Debian oferują dwie przykuwające uwagę, choć często różne, platformy. Każdy projekt posiada swoje mocne, jak i słabe strony. Niemal dla każdej grupy tworzącej własną dystrybucję opartą na GNU/Linux jeden z wymienionych projektów jest punktem odniesienia (z wyjątkiem kilku przypadków, jak na przykład Gentoo).

Jednak chociaż cały proces tworzenia dystrybucji pochodnych dostarcza systemów operacyjnych na różne platformy i zaspokajających różnorodne potrzeby, to jednak sam w sobie, w większości przypadków, działa tylko w jedną stronę. Nowe dystrybucje oparte o Red Hata — jak Mandriva czy SuSE firmy Novell — traktują go lub dostępne w nim technologie jako punkt wyjścia, dalej ich drogi rozchodzą się. Bardzo niewiele zmian dokonanych w nowych projektach znajduje odzwierciedlenie w dystrybucji bazowej, z czasem zaś dochodzą do punktu, w którym różnice między nimi stają się nie do pogodzenia. Podczas gdy oprogramowanie umieszczone w każdej z dystrybucji jest w dużej mierze zbliżone, to sposób, w jaki jest umieszczane w pakietach, prezentowane, instalowane i konfigurowane staje się coraz częściej odmienny. Powoduje to, że współdzielenie i współpraca są coraz trudniejsze.

Powiększająca się rozbieżność wskazuje na bardziej podstawowy problem, z którym mierzą się główne dystrybucje. Często użytkownicy GNU/Linuxu znajdują błędy i składają o nich raporty w używanym oprogramowaniu. Wielokrotnie błędy te są usuwane. Takie błędy czasem znajdują się w nowych dystrybucjach, ale zwykle istnieją w samych aplikacjach, a opracowane przez ich twórców poprawki można wprowadzić do każdej dystrybucji. Co jest niezwykle i niestety zdecydowanie rzadsze, to umieszczenie poprawek dla błędów znalezionych w danej aplikacji w samych programach, tak aby każdy użytkownik danej aplikacji czy każdej dystrybucji mógł z nich skorzystać. Brak współpracy rzadko kiedy wynika ze złej woli, niekompetencji czy strategicznych decyzji podejmowanych przez programistów czy ich współpracowników. Powód jest znacznie prostszy: otóż, śledzenie i monitorowanie zmian w różnych dystrybucjach i w odniesieniu do oryginalnych pakietów jest skomplikowane i trudne. Powszechnie znanym faktem jest to, że takie zmiany czasami powodują więcej kłopotów. Usterki są produktem ubocznym procesu tworzenia dystrybucji, przyjętej polityki, a także narzędzi używanych do tego procesu i wchodzących później w skład systemu — ale nie są one wynikiem ogólnie pojętego procesu jako działania w ekosystemie.

Jak wiele innych dystrybucji, Ubuntu pochodzi od Debiana. W odróżnieniu jednak od innych, w przypadku tej dystrybucji za jedną z podstawowych zasad przyjęto sprawdzenie możliwości ulepszenia procesu tworzenia pochodnych dystrybucji od Debiana zarówno z pakietami samego Debiana, jak i Ubuntu, a także jego pochodnych. Bardziej szczegółowe omówienie Debiana pozwoli wyjaśnić pozycję Ubuntu w świecie wolnego oprogramowania.

Debian i wszechświat wolnego oprogramowania

Debian jest dystrybucją opartą na pracy ochotników, ma 1000 oficjalnych członków i znacznie, znacznie więcej wolontariuszy oraz współpracowników. Rozrósł się do ponad 17 000 pakietów wolnego oprogramowania i aplikacji o otwartym kodzie oraz dokumentacji. Historia tego projektu oraz jego struktura czynią go bardzo dobrym przykładem. Debian ma dobrą reputację z powodu zintegrowanego systemu zarządzania pakietami oraz obszernej listy dostępnych programów. Z drugiej jednak strony, jako ogromna, oparta na pracy ochotników organizacja bez hierarchicznej struktury ma również swoje słabe strony. Częste i rzetelne wydania, odpowiedzialność i wsparcie dla firm, przemysłowy układ aplikacji biurkowych to cechy, których Debian raczej nie zaoferuje swoim użytkownikom.

Każda dystrybucja istnieje z jakiegoś powodu. Tworzenie nowej dystrybucji, nawet w oparciu o już istniejącą, nie jest łatwe. Ubuntu w dużej mierze istnieje dzięki sukcesowi projektu Debian, który poradził sobie z dużą ilością problemów, z którymi się mierzył. Celem zaś jest utworzenie syntetycznej całości dla użytkownika, który wcześniej nie miał styczności lub chęci używania Debiana.

Korzystając z pracy włożonej w Debiana, GNU, Linuksa i inne projekty związane z Debianem, realizatorzy Ubuntu chcieli zbadać nowy styl tworzenia pochodnych dystrybucji skupiony na ścisłych związkach pomiędzy projektami w ekosystemie różnorodnych programistów. Co więcej, Ubuntu nie mogłoby istnieć bez Debiana, jego ogromnej bazy ochotników i oprogramowania oraz stale dostarczanych pakietów wysokiej jakości. Symbioza, w jakiej znajdują się Debian i Ubuntu, odzwierciedla też zależność, w jakiej pozostają one od projektów, takich jak GNU i Linux wytwarzających wspaniałe oprogramowanie, bez których nie mogłyby istnieć. Symbioza to naturalna relacja pomiędzy Ubuntu, Debianem i innymi projektami.

Jednak układ pomiędzy Ubuntu i Debianem nie jest prosty i bezbolesny, wymaga cierpliwości i zaangażowania z obu stron. Z czasem znaleziono jednak metody współpracy, które zdają się oferować większe korzyści od tradycyjnego modelu „utwórz i zapomnij”. Polegają one na całej serii technicznych, socjalnych, a nawet politycznych procesów — wiele z nich zostanie opisanych w dalszej części rozdziału — w ten sposób Ubuntu ma przyczynić się do stworzenia lepszego sposobu tworzenia wolnego oprogramowania.

Spółeczność Ubuntu

Czytelnicy, którzy dotarli do tego miejsca, zapewne zauważyli pewien temat, który przenika projekt Ubuntu na kilku poziomach. Historia wolnego oprogramowania i oprogramowania o otwartych źródłach to właściwie historia skutecznych społeczności. Podobnie jak w przypadku budowania dystrybucji GNU/Linux, projekt Ubuntu jest ukierunkowany na model ekosystemu — układ organizacji — innymi słowy: społeczność. Nawet samo słowo ubuntu odnosi się do ludzi działających w społeczności.

Nie jest zaskoczeniem, że „wewnętrzna” społeczność wywiera poważny wpływ na sposób, w jaki tworzone jest Ubuntu. Podczas gdy wersja 4.04 (Warthy Warthog) została utworzona przez niewielką liczbę osób, to już szeroki sukces Ubuntu był możliwy dzięki współpracy znacznie większej grupy programistów, twórców dokumentacji, ochotników i użytkowników. Canonical zatrudnia kilkudziesięciu współpracowników na stałe, jednak od pierwszego dnia zachęcano wszystkich do aktywnego udziału, nagradzając ich tym, że byli powszechnie rozpoznawani przez członków społeczności. Starano się też, zamiast płacić pracownikom zewnętrznym, zatrudniać ich w firmie i nadal traktować jak wszystkich członków społeczności. Muszą oni wnieść rozpoznawalny wkład w społeczność Ubuntu, jak wszyscy inni. Wszelkie debaty (oprócz tych dotyczących bezpośrednio firmy) są publiczne i dostępne dla społeczności. Członkowie społeczności mają dwa miejsca w dwóch najważniejszych radach Ubuntu: technicznej (zajmującej się zgodnie z nazwą wszelkimi aspektami technicznymi) i tej dotyczącej bezpośrednio społeczności,

która z kolei zajmuje się przyjęciami nowych członków oraz rozstrzygnięciem dyskusji. Stanowiska w radach są zatwierdzane przez odpowiednie grupy: programistów — w przypadku Rady Technicznej i przez członków Ubuntu — w przypadku Rady Społeczności.

Próbując utrzymać ład i jednocześnie zachęcać do aktywnego wkładu w działanie społeczności, w projekcie dąży się do zbalansowania ważnej roli, jaką odgrywa Canonical ze znaczeniem pojedynczych członków społeczności. Projekt Ubuntu opiera się na fundamentalnym założeniu, że świetne oprogramowanie można tworzyć, wspierać i zarządzać nim tylko w oparciu o silne relacje z osobami, które go używają. W ten sposób, popierając i rozwijając aktywną społeczność, Ubuntu może osiągnąć znacznie więcej, aniżeli tylko płacąc programistom za ich pracę. Uczestnicy społeczności wierzą, że podczas gdy wkład Canonical i Shuttlewortha był katalizatorem procesu, który doprowadził do powstania Ubuntu, to właśnie ich działanie przyniosło dystrybucji sukces. Wierzą też, że tylko poprzez zwiększanie znaczenia społeczności można zagwarantować dalsze sukcesy. Nie chcemy wydawać pieniędzy na oprogramowanie własnościowe, jako społeczność możemy znacznie więcej.

Istota społeczności Ubuntu została opisana w rozdziale 8., który w całości został poświęcony temu tematowi. Na końcu warto jeszcze zauważyć, że chociaż niniejsza książka nosi miano „oficjalnej”, to żaden z jej autorów nie jest pracownikiem firmy Canonical. Ta książka, podobnie jak cała reszta Ubuntu, jest wytworem społeczności.

Cele i obietnice Ubuntu

Do tej pory omówiono prehistorię, historię oraz kontekst projektu Ubuntu. Następne rozdziały skupiają się już na samej dystrybucji. Jednak zanim to nastąpi, ważne jest, aby zrozumieć motywy i cele przyświecające projektowi.

Założenia filozoficzne

Najważniejsze założenie jest natury filozoficznej. Najistotniejsze tezy tej filozofii można znaleźć w serii dokumentów umieszczonych na stronie internetowej projektu, są one jednak na tyle istotne, że wiele z nich w całości zamieściliśmy w dodatku B. W głównym dokumencie podsumowano najważniejsze filozoficzne założenia:

Ubuntu to projekt ukierunkowany na społeczność, którego celem jest utworzenie systemu operacyjnego i pełnego zestawu aplikacji w oparciu o wolne oprogramowanie i oprogramowanie z otwartym kodem. Rdzeniem Filozofii Ubuntu o Wolności Oprogramowania są następujące idee:

1. Każdy użytkownik komputera powinien mieć swobodę uruchamiania, kopiowania, dystrybuowania, zmieniania i ulepszania oprogramowania oraz dzielenia się nim w każdym celu i bez płacenia opłat licencyjnych.
2. Każdy użytkownik komputera powinien móc używać oprogramowania w dowolnie wybranym języku.
3. Każdy użytkownik komputera powinien mieć możliwość korzystania z oprogramowania, nawet jeżeli jest niepełnosprawny.

Pierwszy punkt powinien być już zrozumiały. To jedynie podsumowanie definicji wolnego oprogramowania autorstwa Stallmana, omówionej wcześniej. Podkreśla ona, że każdy użytkownik oprogramowania powinien mieć wolność określoną przez ideę wolnego oprogramowania. Jest to ważne z wielu powodów. Po pierwsze: oferuje użytkownikom wszystkie korzyści płynące z używania oprogramowania, działającego lepiej i szybciej, i które łatwiej jest dostosować do własnych potrzeb. Co ważniejsze, daje każdemu możliwość wykroczenia poza rolę użytkownika końcowego i konsumenta oprogramowania. Projekt Ubuntu zakłada, że oprogramowanie umożliwia użytkownikowi taki sposób pracy, jakiego on oczekuje. Każdy użytkownik powinien mieć możliwość upewnienia się, że program pracuje właśnie dla niego. Aby to było możliwe, oprogramowanie musi być wolne, dlatego w Ubuntu uczyniono z tego wymóg i obietnicę.

Oczywiście podstawowe założenia nie kończą się na definicji wolnego oprogramowania. Oprócz tego w projekcie głoszone są dwie inne, choć równie ważne idee. Pierwsza z nich mówi o tym, że każdy użytkownik komputera powinien móc korzystać z programów w języku, który wybierze. Jest to nawiązanie do tego, że większość ludzi na świecie nie posługuje się językiem angielskim, chociaż zdecydowana część aplikacji dostępna jest tylko w tym języku. Aby mogły być użyteczne, komentarze do kodu programu, języki programowania, dokumentacja, teksty, menu aplikacji muszą być pisane również w innych językach. Prawdopodobnie wybór najbardziej międzynarodowego języka jest rozsądną decyzją, ale z drugiej strony nie jest to język, którym posługują się wszyscy, dlatego język angielski jest nieprzydatny dla tych, którzy go nie znają. Komputer może być wspaniałym narzędziem umożliwiającym wiele również w dziedzinie edukacji, ale tylko pod warunkiem, że użytkownik może zrozumieć słowa pojawiające się na jego monitorze. Dlatego właśnie projekt Ubuntu przyjął za zadanie umożliwić każdemu użytkownikowi korzystanie z komputera w języku, który jest dla niego najwygodniejszy.

Możliwość dokonywania modyfikacji — warunek wolnego oprogramowania i pierwsza z zasad Ubuntu — dzięki temu można dokonywać tłumaczeń. Niniejsza książka jest swoistym przykładem. Pomaga zapoznać się z Ubuntu stosunkowo niewielkiej grupie osób władających językiem angielskim, ale ten język został wybrany po to, aby umożliwić jak najszersze działanie. Co najważniejsze: książka została opublikowana na licencji umożliwiającej tłumaczenie, wprowadzanie zmian oraz redystrybucję. Autorzy nie mogli jej napisać we wszystkich językach świata ani

też w więcej niż jednym. Zamiast tego postanowili usunąć wszelkie prawne ostrzeżenia i inne przeszkody, które mogłyby utrudniać społeczności pracę nad przekładem.

Żadna osoba nie powinna mieć ograniczanych możliwości dostępu do komputera tylko dlatego, że nie zna podstawowego języka; żaden użytkownik nie może mieć zablokowanego dostępu do komputera z powodu jakichś dysfunkcji. Ubuntu musi być dostępne dla wszystkich użytkowników, niezależnie od tego, czy są niepełnosprawni motorycznie, mają problemy ze wzrokiem czy słuchem. System musi zapewniać możliwość dwukierunkowej komunikacji z użytkownikiem w każdej z wymienionych sytuacji. Znaczący procent najbardziej inteligentnych i kreatywnych ludzi na świecie to osoby niepełnosprawne. Dostępności Ubuntu nie można ograniczać tylko do pewnej części społeczeństwa, podczas gdy może być dostępny dla wszystkich. Co ważniejsze, Ubuntu powinno być dostępne dla nich, aby jako członkowie społeczności mogli tworzyć lepsze i bardziej wydajne oprogramowanie.

Kodeks Postępowania Ubuntu

O ile filozoficzne zasady Ubuntu opisują założenia projektu Ubuntu, to Kodeks Postępowania traktuje o tym, jak to osiągnąć. Kodeks Postępowania (w skrócie KP) jest prawdopodobnie najważniejszym dokumentem społeczności: po pierwsze dlatego, że ustanawia podstawowe zasady współpracy w ramach projektu, a po drugie dlatego, że stosuje się go praktycznie codziennie. Wyraża zgodę na przyjęcie tych warunków jest jedynym kryterium koniecznym, aby zostać oficjalnie rozpoznawanym aktywistą Ubuntu — czyli „Ubuntero” — i zarazem niezbędnym krokiem, aby zostać członkiem projektu. Podpisywanie Kodeksu Postępowania Ubuntu oraz procedura przyjęcia w poczet członków Ubuntu opisane są w rozdziale 6.

Kodeks Postępowania (który można znaleźć w Dodatku B) „dotyczy sposobów zachowania się członków wspólnoty Ubuntu na forach, listach mailingowych, stronach internetowych, kanałach IRC, publicznych zgromadzeniach czy w prywatnej korespondencji pomiędzy członkami wspólnoty”⁴. KP rozwija się w kilku zagadnieniach, które można podsumować następującymi nagłówkami:

- Bądź rozważny
- Zachowaj szacunek
- Współpracuj
- Kiedy jesteś odmiennego zdania, konsultuj się z innymi
- Kiedy nie masz pewności, poproś o pomoc
- Odchodząc, pomyśl o innych

⁴ Tekst zaczerpnięty ze strony <http://ubuntu.pl/manifest.html> — przyp. tłum.

Większość z nich zdaje się brzmieć zupełnie zwyczajnie, czy też dotyczyć podstawowych zasad dobrego wychowania — jest to zabieg celowy. Żaden fragment Kodeksu Postępowania nie jest kontrowersyjny czy radykalny i nigdy nie miał takim być.

Trudność jednak wynika stąd, że nie da się tych zasad wprowadzić w życie, czy też przekonać do nich siłą, zwłaszcza że rozumienie pojęć, takich jak odpowiedzialne działanie, szacunek dla innych czy współpraca często nieco się różni. Zawsze może przydarzyć się poważne nieporozumienie, czy nawet ktoś może się obrazić. Dlatego wymienione zasady są tylko pewnymi — choć akceptowanymi — skrótami. KP nie był pisany w celu kodyfikowania wszelkich możliwych działań, zakazów itd. Zamiast tego ma on raczej służyć pomocą w ustalaniu zasad oraz przypominać, że rozsądna dyskusja i szacunek dla partnerów zapewniają zdrowe podstawy i rozwój projektu. W sytuacji, kiedy zaistnieje poważna różnica zdań lub członek społeczności będzie szkodził innemu (czy też kodowi programu), Rada Społeczności — organ szerzej omówiony w rozdziale 8. — jest powołana do podjęcia arbitrażu lub też wszelkich innych niezbędnych działań.

Nikt z uczestników projektu Ubuntu, włączając Shuttlewortha i jakiegokolwiek innego członka Rady Społeczności, nie stoi ponad Kodeksem Postępowania. Nie można stosować jego zasad wybiórczo czy opcjonalnie. Oczywiście, nie można było napisać go w taki sposób, aby wyeliminować wszelkie konflikty czy różnice zdań. Kłótnie w Ubuntu są tak samo powszechne, jak i w innych projektach czy społecznościach internetowych. Jednak istnieje powszechna zgoda na to, że wszelkie dyskusje powinny toczyć się w warunkach wzajemnego szacunku i współpracy. Dzięki temu zarówno dyskusje, jak i ich rezultaty są lepsze, a przy okazji mniej jest złych emocji i mniej posiniaczonych ego.

Zdarza się czasem, że Kodeks Postępowania jest niewłaściwie używany jako kontrargument przeciw oponentowi podczas dyskusji. Należy go jednak traktować raczej jako punkt odniesienia, który umożliwia osiągnięcie porozumienia w społeczności Ubuntu. Najczęściej jednak jest tak, że gdy grupa dostrzeże, że jeden z członków działa w sposób, który narusza zasady, wówczas delikatnie przypomina się mu, często prywatnie, że obowiązuje go KP. W większości sytuacji to wystarczy, aby zażegnać konflikt czy zapobiec niewłaściwemu rozwojowi sytuacji. Bardzo niewiele naruszeń KP znalazło swój finał przed Radą Społeczności.

Cele techniczne

Podczas gdy wzajemny szacunek członków społeczności oraz stosowanie się do założeń filozoficznych dostarcza niezbędnej ramy, w której działa cały projekt, Ubuntu jest i pozostanie projektem technicznym. Oprócz założeń filozoficznych i zasad działania (czy raczej jako ich uzupełnienie), Ubuntu posiada również cele techniczne.

Pierwszym celem technicznym i prawdopodobnie najważniejszym jest koordynowanie regularnych i przewidywalnych nowych wydań systemu. W kwietniu 2004 roku podczas spotkania Guźców przyjęto jako cel wydawanie kolejnych wersji co pół roku. Po części z powodu oszałamiającego sukcesu projektu, a po części po to, aby wpisać się w harmonogram wydawania kolejnych edycji GNOME, zespół pozostał przy przewidywalnych i regularnych (co 6 miesięcy) wydaniach nowych wersji. W chwili obecnej można zmienić kalendarz wydań tylko o 6 tygodni i tylko za zgodą społeczności. Częste wydania są ważne, ponieważ umożliwiają użytkownikom korzystanie z najlepszego i najnowszego darmowego oprogramowania, a jest to istotne w szybko zmieniającym się i dążącym do ciągłego poprawiania kodu środowisku programistów wolnego oprogramowania. Przewidywalność kolejnych wydań jest równie ważna, szczególnie dla firm, które dzięki temu mogą opracowywać plany z uwzględnieniem Ubuntu. Dzięki konsekwentnej realizacji planu nowych wydań Ubuntu dostarcza platformy umożliwiającej zarówno firmom, jak i twórcom dystrybucji pochodnych, planowanie własnego rozwoju.

Równie ważne jak solidne i częste wydania nowych wersji jest wsparcie dla już istniejącego oprogramowania. Ubuntu, podobnie jak wszystkie inne dystrybucje, musi radzić sobie z faktem, że oprogramowanie ma błędy. Większość z nich to drobne usterki, ale naprawianie ich może spowodować poważniejsze kłopoty. Dlatego właśnie usuwanie usterek w już udostępnionym oprogramowaniu należy przeprowadzać z największą ostrożnością — albo wcale. W projekcie Ubuntu tylko poważne zmiany (w tym również poprawki błędów) są wprowadzane wyłącznie w okresie pomiędzy kolejnymi wydaniem, kiedy jest czas na ich dokładne sprawdzenie. Niektóre z usterek niosą ze sobą ryzyko utraty danych przez użytkownika czy też poważne zagrożenie bezpieczeństwa. Są one naprawiane natychmiastowo, a dostępne są w postaci aktualizacji dla systemu. W Ubuntu pracuje się ciężko przed kolejnym wydaniem nad wykrywaniem i usuwaniem błędów wszelkiego rodzaju i odnosi się sukces w przypadku najpoważniejszych usterek. Jednak ponieważ zawsze istnieje ryzyko wykrycia kolejnych błędów, każde wydanie Ubuntu ma zapewnione wsparcie przez 18 miesięcy od daty wydania. W przypadku Ubuntu 6.06 (Dapper Drake) posunięto się nawet dalej i zdecydowano, że to wydanie będzie wspierane przez pełne trzy lata w przypadku komputerów przeznaczonych na biurka i przez pięć lat w konfiguracjach serwerowych.

To dwustronne podejście do komputerów przeznaczonych na biurka i dla serwerów wyznacza kolejne techniczne zobowiązanie projektu Ubuntu: osobne wsparcie dla obu wymienionych rodzajów komputerów, z naciskiem jednak na równoległy tryb ich udostępniania. Choć Ubuntu jest bardziej znane i zapewne popularniejsze w konfiguracji przeznaczonej dla komputera biurowego, to warto wiedzieć, że istnieje zespół programistów, których działanie ukierunkowane jest zarówno na użytkowników serwerów, jak i desktopów. W projekcie przyjęto, że oba rodzaje komputerów są istotne i dlatego przygotowuje się instalacyjne płyty CD dla obu rodzajów systemów. Ubuntu dostarcza przetestowane oprogramowanie

(dodatkowo posiadające wsparcie) najbardziej nadające się do standardowych zadań dla obu środowisk, a także dokumentację dla każdego z nich. Niniejsza książka zawiera informacje o uruchamianiu Ubuntu zarówno na komputerach przeznaczonych na biurko, jak i na serwery.

I na koniec: Ubuntu jest zaangażowane również w umożliwienie swoim klientom wyjścia z roli użytkowników oprogramowania i korzystania z możliwości oferowanych przez każdy z głównych punktów naszej filozofii. W rezultacie Ubuntu stara się skupić uwagę swoich programistów na używaniu i promowaniu jednego języka programowania: Pythona. Wszyscy pracują nad tym, aby Python stał się językiem szeroko stosowanym w systemie. Kiedy uda się doprowadzić do sytuacji, w której większość aplikacji graficznych, tekstowych czy konsolowych będzie napisana w tym języku, wówczas użytkownik będzie mógł nauczyć się tylko jednego języka, znając go, który pozwoli mu na pełniejsze korzystanie z systemu poprzez automatyzację czynności czy dostrajanie poszczególnych jego elementów.

Błąd #1

Oczywiście celem całego projektu Ubuntu nie jest tylko zbudowanie systemu operacyjnego zgodnie z naszą filozofią czy założeniami technicznymi — chociaż zapewne będziemy szczęśliwi, kiedy uda się nam to osiągnąć. Naszym ostatecznym celem (nadrzędnym i mającym wpływ na wszystkie inne) jest rozprzestrzenianie naszego wspólnego oprogramowanie w częstych wydania systemu oraz propagowanie zasad wolności zamieszczonych w naszych założeniach światopoglądowych dla tak wielu użytkowników komputerach w tak wielu krajach, jak to tylko możliwe. Podstawowym celem Ubuntu nie jest wcale stać się najbardziej rozpowszechnioną dystrybucją GNU/Linux na świecie, ale stać się najczęściej używanym systemie operacyjnym na świecie.

Pierwszy błąd umieszczony w Malone (systemie śledzenia błędów używanym w Ubuntu) ilustruje ten fakt. Błąd, określony przez Shuttlewortha jako „krytyczny”, wciąż się tam znajduje i można znaleźć go pod adresem: <https://launchpad.net/distros/ubuntu/+bug/1>. Jego tekst w przekładzie brzmi następująco:

Microsoft posiada większość udziałów na rynku. Nie-wolne oprogramowanie wstrzymuje innowacje w przemyśle informatycznym, ograniczając dostęp do IT dla małej grupy światowej populacji i zmniejszając globalnie szanse programistów na osiągnięcie pełnego potencjału. Ten błąd jest powszechnie znany w przemyśle producentów komputerów.

Czynności do wykonania:

1. Odwiedzić lokalny sklep z komputerami.

Jak jest?

1. Da się zaobserwować, że większość sprzedawanych komputerów posiada preinstalowane nie-wolne oprogramowanie.
2. Da się zaobserwować, że bardzo niewiele komputerów posiada preinstalowane wolne oprogramowanie.

Jak powinno być?

1. Większość sprzedawanych komputerów powinna być wyposażona tylko w wolne oprogramowanie, takie jak Ubuntu.
2. Ubuntu powinno być prezentowane w taki sposób, aby wszystkie jego niesamowite możliwości i płynące z nich korzyści były powszechnie znane.
3. Z upływem czasu system powinien być coraz bardziej przyjazny użytkownikowi.

Wielu w ciągu ostatnich dwóch lat opisywało sukces Ubuntu jako niesamowity. Dla nowych dystrybucji GNU/Linux poziom i szybkość sukcesu faktycznie są bezprecedensowe. W tym okresie w pełni czerpało ze swoich filozoficznych i technicznych założeń, osiągnęło wiele z założonych celów i zbudowało żywą społeczność użytkowników i współpracowników, którzy również odnieśli wielkie sukcesy, współpracując w duchu szacunku i zgodnie z zasadami Kodeksu Postępowania Ubuntu. Jednakże Błąd #1 pokazuje wyraźnie, że projekt Ubuntu będzie mógł ogłosić pełny sukces dopiero wówczas, gdy propagowane przezeń standardy wolności, doskonałości technicznej oraz zasady staną się normą w całym programistycznym świecie.

Canonical i Fundacja Ubuntu

Ubuntu jest organizacją, której motorem napędowym jest społeczność, znajduje się w niej jednak kilka innych jednostek, które pełnią ważną rolę w jej strukturze. Pośród nich znajduje się Canonical Ltd., komercyjne przedsiębiorstwo omówione we wstępie do historii Ubuntu, oraz Fundacja Ubuntu, która omówiona jest poniżej.

Canonical Ltd.

Jak wspomniano wcześniej, Canonical Ltd. jest firmą założoną przez Shuttlewortha, a jej podstawowym celem jest projektowanie i wspieranie dystrybucji Ubuntu. Wielu z kluczowych programistów Ubuntu — chociaż obecnie nie jest to już większość — pracuje na pełnym (lub niepełnym) etacie dla Canonical Ltd. Dzięki temu dystrybucja Ubuntu może oferować wsparcie na obecnym poziomie. Ubuntu może deklarować wydawanie regularnie co pół roku nowych edycji, dlatego że właśnie za to Canonical płaci programistom. To jedna ze słabości Debiana, który

jako organizacja całkowicie oparta na pracy wolontariuszy nie może ściśle określać terminów i ich dotrzymywać. Ochotnicy są czasami bardziej zajęci lub też mają narzucone nieprzekraczalne terminy w pracy, która jest ich źródłem utrzymania. Oferując programistom płatną pracę, Canonical może zaoferować wsparcie dystrybucji oraz ustalać konkretne terminy kolejnych wydań, a co ważniejsze — dotrzymywać ich.

W ten sposób Canonical zapewnia realizację głównych zobowiązań Ubuntu. Oczywiście firma nie może zapewnić ani finansowania wszystkich prac, ani ich wykonania. Canonical może publikować dystrybucję co sześć miesięcy, ale sama dystrybucja będzie dużo lepsza i bardziej użyteczna, jeżeli swój wkład wniesie w nią społeczność użytkowników. Większość funkcji, podobnie jak większość nowych programów, niemal całość tłumaczeń i dokumentacji, powstała poza firmą Canonical Ltd. W zamian firma zapewnia dotrzymywanie terminów oraz wykonanie zasadniczych prac, niezależnie od tego, czy są one przyjemne, czy też nie.

Canonical Ltd. została założona na wyspie Man, małym skrawku ziemi położonym pomiędzy Irlandią a Walią, znanym szerzej jako raj dla międzynarodowych przedsiębiorców. Ponieważ pracownicy firmy są rozproszeni po całym świecie i właściwie nie ma potrzeby posiadania biura, wyspa Man wydawała się tak samo dobra jak każde inne miejsce, aby tam rozpocząć działalność.

Usługi świadczone przez Canonical oraz wsparcie techniczne

Dla wielu będzie to zaskakująca informacja, jednak mniej niż połowa pracowników Canonical zatrudniona jest przy projekcie Ubuntu. Reszta z nich zajmuje się czymś innym: rozwojem firmy, wsparciem i administracją oraz wsparciem programistycznym projektów Bazaar i Launchpad.

Osoby zajmujące się rozwojem firmy pomagają tworzyć strategie kontaktów z innymi firmami oraz wspierają certyfikowanie programów; ma to związek głównie z Ubuntu. Jednym z przykładów takich działań jest współpraca z innymi firmami, która ma na celu doprowadzenie do sytuacji, gdy produkowane przez te firmy aplikacje (zwykle prawnie zastrzeżone) są tworzone i certyfikowane do uruchamiania na Ubuntu. Canonical współpracuje z IBM, aby umożliwić uruchamianie popularnych baz danych DB2 w Ubuntu. Kiedy to zostanie osiągnięte, następnym krokiem będzie uzyskanie certyfikatu dla Ubuntu jako platformy dedykowanej dla DB2. Na podobnych zasadach Canonical współpracuje z HP, aby umożliwić klientom w Europie zakup komputera przenośnego, dla którego jako opcjonalny system operacyjny będzie można wybrać Ubuntu. Trzecim przykładem jest niniejsza książka, opublikowana przez Pearson Prentice Hall, a będąca owocem współpracy z Canonical.

Canonical odgrywa również istotną, wspierającą rolę w projekcie Ubuntu i to na trzy sposoby. Po pierwsze, wspiera proces tworzenia Ubuntu. To działanie obejmuje na przykład obsługę i utrzymanie przez administratora sieci serwerów wykorzystywanych w procesie tworzenia dystrybucji. Po drugie, Canonical może bezpośrednio wspierać użytkowników Ubuntu. Obejmuje to ofertę pomocy technicznej przez telefon i za pośrednictwem poczty elektronicznej oraz poprzez organizowanie kontraktów na udzielanie wsparcia technicznego dla dużych firm i organizacji. Takie wsparcie jest czymś więcej niż to darmowe, oferowane przez społeczność. Komercyjne wsparcie jest oferowane za opłatą wnoszoną bądź w charakterze abonamentu, bądź też jako wynagrodzenie za wykonaną usługę. Wreszcie ostatni punkt: pomoc udzielana innym organizacjom, które same zajmują się udzielaniem wsparcia. Canonical nie szuka, ani też nie zamierza uzyskać monopolu w świadczeniu pomocy technicznej dla Ubuntu. Z dumą przedstawia w witrynie internetowej listę setek organizacji zajmujących się świadczeniem pomocy technicznej dla Ubuntu. Zamiast tego Canonical oferuje coś, co nosi nazwę pomocy technicznej drugiej warstwy (ang. *second-tier*). Ponieważ Canonical zatrudnia wielu kluczowych programistów Ubuntu, ma ułatwioną pozycję w rozwiązywaniu wielu trudniejszych problemów, jakie mogą napotkać firmy świadczące wsparcie. Dzięki skoncentrowanej wiedzy i umiejętnościom firma może skutecznie wspierać innych.

Bazaar i Launchpad

Jako dodatek do wspierania i programowania Ubuntu, Canonical Ltd. sfinansowała utworzenie Bazaar-NG, narzędzia do kontroli wersji oraz projekt Launchpad. Bazaar-NG to narzędzie do projektowania oprogramowania szeroko wykorzystywane w Ubuntu, odgrywa ono też bardzo ważną rolę w technicznych procesach, dzięki którym powstaje Ubuntu. Jednak to narzędzie, którego funkcja podobna jest do tej spełnianej przez inne, podobne programy (CVS, Subversion, BitKeeper) jest przydatne również w wielu innych projektach. Co ważniejsze, Bazaar-NG pracuje jako koń pociągowy projektu Launchpad.

Więcej niż połowa pracowników technicznych firmy Canonical pracuje przy projekcie Launchpad. Launchpad to ambitny, działający w przeglądarce internetowej program, czy raczej nakładka na kilka, w wysokim stopniu zintegrowanych narzędzi. Ten program odgrywa główną rolę w procesie tworzenia Ubuntu, ale jest też wykorzystywany w procesie tworzenia innych dystrybucji, zwłaszcza tych opartych na Ubuntu. Na Launchpad składają się następujące aplikacje:

■ Rosetta

Działający na serwerze WWW system ułatwiający tłumaczenie niemal każdego fragmentu wolnego oprogramowania z języka angielskiego na niemal każdy inny. Nazwa pochodzi od kamienia z Rosetty, który umożliwił lingwistom złamanie kodu starożytnych egipskich hieroglifów.

■ Malone

Używany w Ubuntu system śledzenia błędów. Umożliwia ich śledzenie w różnych edycjach i pozwala obserwować ich status również w innych dystrybucjach czy pakietach autorskich. Malone to odwołanie do musicalu „Bugsy Malone”.

■ Soyuz

Ta część pakietu do zarządzania dystrybucją odpowiedzialna jest za procesy pakietowania, testowania i migracji pomiędzy różnymi częściami dystrybucji. Sojuz to nazwa rosyjskiej rakiety, która wyniosła Marka Shuttlewortha w przestrzeń kosmiczną. Słowo „sojuz” po rosyjsku oznacza „związek”.

Pakiet Launchpad wraz z jego składnikami został szerzej omówiony w rozdziale 9. Znaczenia Launchpada dla Ubuntu nie można przecenić. Oprócz obsługi błędów, tłumaczeń i budowania dystrybucji system ten wykorzystywany jest też do autoryzacji i kodyfikowania informacji o członkach zespołu Ubuntu. To miejsce, w którym gromadzone są wszystkie informacje o pracach nad Ubuntu. Każdy członek społeczności Ubuntu, każda osoba, która wnosi swój wkład w dystrybucję — w odpowiednim momencie — zakłada konto w systemie Launchpad.

Fundacja Ubuntu

Oprócz firmy Canonical i społeczności, projekt Ubuntu wspierany jest dodatkowo przez fundację Ubuntu założoną przez Shuttlewortha, który dodatkowo wyposażył ją na początek w kapitał w wysokości 10 milionów dolarów. Fundacja, podobnie jak Canonical, ma swoją siedzibę na wyspie Man. Fundację wspiera Rada Społeczności Ubuntu.

Odmienne niż firma, fundacja nie bierze aktywnego udziału w codziennym życiu Ubuntu. W chwili obecnej jej główną rolą jest pełnienie funkcji worka z pieniędzmi, aby wspomagać finansowo i zapewnić Ubuntu przyszłość. Ponieważ Canonical jest młodą firmą, wielu osobom i przedsiębiorstwom trudno zaufać i uwierzyć w to, że będzie ona w stanie zapewnić wsparcie dla dystrybucji w określonych ramach czasowych (na przykład w ciągu 5 lat). Fundacja istnieje po to, aby rozwiewać takie wątpliwości.

Gdyby coś złego przydarzyło się Shuttleworthowi czy Canonical Ltd., tak że nie byłiby w stanie wspierać rozwoju Ubuntu, wówczas Fundacja ma zapewnić kontynuację wielu podstawowych działań w przyszłości. Dzięki istnieniu Fundacji, Ubuntu może podejmować długoterminowe zobowiązania i obiecywać ich wypełnienie.

Jednym z działań, które może podejmować Fundacja (i które podejmuje), jest pozyskiwanie darowizn na rzecz Ubuntu. Te środki finansowe — i tylko one — są następnie uruchamiane w ramach projektu zgodnie z życzeniami zespołu programistów i rady technicznej. W większej części są one przeznaczane na nagrody dla członków społeczności, którzy osiągnęli ważne cele projektu Ubuntu.

Podprojekty Ubuntu, dystrybucje pochodne i inne

Żadne wprowadzenie do Ubuntu nie będzie kompletne, jeżeli nie zostanie zamieszczona w nim lista podprojektów i dystrybucji pochodnych. Choć sama dystrybucja pochodzi od Debiana, w trakcie dwóch lat istnienia doczekała się sporej liczby dystrybucji, które wykorzystują ją samą jako podstawę do swojego rozwoju.

Pierwszą i podstawową pośród nich jest Kubuntu — czyli wersja Ubuntu wykorzystująca środowisko KDE zamiast GNOME. Kubuntu szerzej omówiono w rozdziale 7., dlatego nie będziemy skupiać się nad tą wersją tutaj. Warto jednak zdawać sobie sprawę z faktu, że relacje pomiędzy Ubuntu a Kubuntu są zgoła inne od tych, które łączą Debiana i Ubuntu. Jeżeli przyjmiemy techniczny punkt widzenia, to Kubuntu zawiera się w Ubuntu. Również organizacyjnie zespół Kubuntu pracuje w ramach zespołu Ubuntu.

Podobna organizacja ma miejsce w przypadku projektu Edubuntu, który pomaga projektować konfigurację Ubuntu tak, aby można było łatwo i skutecznie umieszczać tę dystrybucję w szkołach. Projekt skupia się na dwóch aspektach: oprogramowaniu edukacyjnym i przeznaczonym dla szkół oraz na projekcie LTSP (ang. *Linux Terminal Server Project*), który umożliwia uruchamianie stanowisk komputerowych w szkołach z wykorzystaniem jednego (lub więcej) wydajnego serwera i podłączanych do niego uproszczonych terminali. Ten stosunkowo prosty zabieg techniczny pozwala na ogromne oszczędności dla szkół.

Choć Edubuntu i Kubuntu są blisko związane z Ubuntu, a prace nad nimi są częściowo wspierane przez Canonical, to istnieje wiele dystrybucji pochodnych, które zaczęły pojawiać się, korzystając z innego modelu niż ten, który miał zastosowanie w przypadku dwóch omówionych dystrybucji. Pierwszą „zewnętrzną” dystrybucją pochodną był Guadalinex, system stworzony i zarządzany przez władze lokalne Andaluzji w Hiszpanii. Inne dystrybucje to mniej wymagające sprzętowo wersje Ubuntu, przeznaczone na wolniejsze komputery (Xubuntu czy nUbuntu) czy przeznaczone do testów bezpieczeństwa sieci (Network Ubuntu). Wciąż trwają prace nad głównym systemem zarówno w ramach projektu, jak i poza nim.

W ten sposób i dzięki dystrybucjom pochodnym oraz celom przyświecającym Ubuntu projekty współpracują i krystalizują się. Tylko poprzez ruch wolnego i otwartego oprogramowania i przy założeniu wolnego dostępu do kodu Ubuntu mogło być stworzone. I podobnie jest z dystrybucjami pochodnymi — tylko dzięki kontynuowaniu tych idei przez Ubuntu inne dystrybucje mogą zeń czerpać. Sama będąc dystrybucją pochodną i postrzegając się jako istniejącą w ekosystemie, nie traktuje procesu tworzenia dystrybucji pochodnych jako zniewagi czy krytyki. Jesteśmy dalecy do tego. Uważamy raczej, że derywacja to wyższy stopień komplementu.

Poza Ubuntu Canonical skupia się na pracy nad oprogramowaniem takim jak Launchpad i Bazaar, które zostały zaprojektowane do ułatwiania właśnie takiego rodzaju tworzenia dystrybucji pochodnych. Ten proces, dobrze sprawdzony, jest jednym, który opisuje ekosystem programowania przynoszący korzyści wszystkim — tworzonej dystrybucji, Ubuntu i autorom pakietów. Tylko poprzez proces derywacji wszyscy otrzymują to, czego oczekują.

Prawidłowo przeprowadzony proces tworzenia dystrybucji pochodnej pozwala grupom rozchodzić się, kiedy to konieczne i pracować razem, kiedy to możliwe. Ostatecznie prowadzi to do wykonania większej ilości pracy, większej liczby zadowolonych użytkowników i całościowo do jeszcze lepszej współpracy. Poprzez stale poprawioną współpracę techniczne i filozoficzne założenia Ubuntu mogą zostać osiągnięte. To właśnie poprzez głębokie zaangażowanie wspólnoty Błąd #1 może zostać usunięty. To poprzez głęboką współpracę, wewnątrz i na zewnątrz projektu, ten niesamowity rozwój Ubuntu w ciągu pierwszych dwóch lat istnienia będzie podtrzymany w następnych dwóch i następnych dwudziestu.

Podsumowanie

Niniejszy rozdział to wprowadzenie do fenomenu, jakim jest Ubuntu. Rozpoczyna się od historii wolnego oprogramowania i oprogramowania o otwartych źródłach, następnie omówiono historię Ubuntu. W dalszej kolejności opisano Ubuntu jako produkt, jego założenia filozoficzne, przyjęte cele oraz związek pomiędzy Canonical Ltd. i Fundacją Ubuntu. Na końcu omówiono związane z Ubuntu podprojekty, dystrybucje pochodne i inne, korzystające z jej osiągnięć.