

Michał Makarewicz
Wydział Matematyki, Informatyki i Mechaniki UW

Świat programistów

W ciągu ostatniego dziesięciolecia dało się zaobserwować zwiększone zainteresowanie otwartym oprogramowaniem komputerowym. Można zaryzykować próbę odgadnięcia jak przyszłość czeka ich twórców i użytkowników. Czy korporacje produkujące software upadną, a wszyscy będą mogli cieszyć się w końcu dopracowanym, łatwym w konfiguracji i ergonomicznym w użyciu zestawem różnego rodzaju oprogramowania, dostępnego dzięki wspólnemu wysiłkowi milionów programistów z całego świata? A może ludzie, zawiązani trudnościami z obsługą otwartych programów, nie będą mogli znaleźć praktycznych korzyści w fakcie, że kod systemu, który właśnie bezskutecznie próbują uruchomić, jest w pełni otwarty, będą powszechnie godzić się na ustępstwa względem wielkich koncernów¹, byle tylko dostać produkt w pełni odpowiadający ich oczekiwaniom, a kultura open-source pozostanie domeną wąskiej grupy miłośników programowania, i nielicznych osób silących się na oryginalność?

Nieustający rozwój technologiczny świata sprawia, że komputerowe urządzenia nie tylko mają coraz lepsze parametry techniczne, ale i koszty ich wytwarzania stają się coraz niższe. Znajdują coraz więcej zastosowań. W krajach rozwijających się taka tendencja również ma miejsce, choć na razie w mniejszym stopniu. Nie ma jednak podstaw by sądzić, że zjawisko to ulegnie zahamowaniu. W dłuższej perspektywie na Ziemi pozostaną tylko i wyłącznie ludzie, którzy przez całe życie na co dzień mieli do czynienia z urządzeniami komputerowymi. Popularność, przeradzająca się we wszechobecność tych urządzeń musi znaleźć odzwierciedlenie m. in. w szkolnej edukacji (dzięki zapotrzebowaniu na ich twórców na rynku pracy), ich obsługa, programowanie i znajomość zasad budowy będzie jedną z podstawowych umiejętności życiowych do opanowania. To, co obecnie jest wiedzą uchodzącą za zaawansowaną, przyswaną tylko przez osoby zainteresowane informatyką, stanie się znane każdemu. Firmy tworzące software będą się początkowo mnożyć i rozrastać, lecz w końcu okaże się, że ludzie nie potrzebują profesjonalnych korporacyjnych programistów, by ci nie wiadomo jak zmuszali komputery do wykonywania niesamowitych rzeczy, gdyż każdy będzie mógł sam dokonać tego samego. Przy takim poziomie wiedzy atrakcyjnym handlowo towarem staną się „otwarte” urządzenia elektroniczne, których budowa i zasady działania będą ujawniane, nie wymagające tajnych, komercyjnych sterowników. Będzie można wykorzystywać je w jak najwyższym stopniu, w pełni dopasowując do własnych potrzeb. Ludzie o podobnych zainteresowaniach będą współpracować przy realizacji bardziej złożonych projektów programistycznych, nie napotykając barier o charakterze techniczno-komputerowym, problemem będzie co chcą stworzyć, a nie jak. Pewne wspólne zasady, swojego rodzaju standardy pisania programów staną się zwyczajem, również nabywanym w szkole. Oczywiście będą one ewoluować w czasie. Zapewni to wzajemną zgodność większości programów, i umożliwi ich

¹ Na przykład akceptując sytuację w której produkt, za który płacą, nie staje się ich własnością, lecz jest im na swój sposób wypożyczany i może być odebrany (zablokowany) w każdej chwili. Albo nie mając pewności czy pracownicy kocemu nie mają możliwości uzyskania dostępu do prywatnych danych użytkownika przez tak zwane „tylne drzwi”. Lub pokornie kupując nowszy odtwarzacz płyt, gdy starszy nie współpracuje z najnowszym zabezpieczeniem oryginalnego nośnika przed kopiowaniem.

współpracę. Być może ludzie w głosowaniach podejmować będą decyzję dotyczące ich modyfikacji.

Przeglądarka „Firefox” może być ilustracją mających zajść zmian. Przebyła ona długą drogę - od zamkniętego i płatnego pakietu Netscape, do produktu otwartego, zyskującego na popularności, odbierającego użytkowników Internet Explorerowi nie dzięki darmowości, ale wyższej jakości, związanej z mnogością dodatków do niej, które tworzyć może każdy. Taką ewolucję przejdą inne grupy programów, ostatnie będą te, które oprócz programistów wymagają innego rodzaju ekspertów do ich powstania, na przykład grafików i muzyków w przypadku gier. Na końcu poddadzą się profesjonalne, dedykowane narzędzia, wykorzystywane jedynie przez nieliczne grupy specjalistów. Widać więc, że wykonalne jest przejście od komercyjnego produktu, mającego zarabiać pieniądze dla autorów, do dostępnych powszechnie wielu stron kodu, który można poznać, uzupełnić, ulepszyć, i wykorzystać w zupełnie nowy sposób. Możliwa jest droga w przeciwnym kierunku (Mac OS korzystający z systemu BSD), ale trudno nazwać to zagrożeniem dla upowszechniania otwartych programów. Tym bardziej że licencja GPL, stosowana jednak najczęściej, nie zezwala na zamykanie kodu. Istnienie komercyjnych dystrybucji Linuksa jest dobrym przykładem przydatności wolnego oprogramowania także do zarabiania pieniędzy, mimo że każda z nich ma otwarte źródła i istnieją ich powszechnie dostępne kompilacje, gotowe do zainstalowania, różniące się jedynie brakiem oznaczeń firmowych². Kto wie, czy właśnie taka przyszłość nie czeka różnych płatnych pakietów, na razie ciągle jeszcze będących symbolami świata „copyrightu”.

Z pewną dozą nieśmiałości pozwalam sobie podzielić się tak odważną i oderwaną od rzeczywistości wizją. Chciałbym, usprawiedliwiając się, zwrócić uwagę na fakt, że mowa o perspektywie wielu dziesiątek lat. Obecnie wiele czynników utrudnia dalsze upowszechnianie wolnego oprogramowania – niedoskonała organizacja większych projektów, brak spójności dążeń programistów, ich nastawienie bardziej na samorealizację, wdrażanie własnych pomysłów, dopracowywanie technicznych detali, a mniej na zaspokajanie oczekiwań potencjalnych odbiorców, „wychowywanie” sobie klientów przez korporacje atakujące szkoły i urzędy swoimi produktami, powszechna niechęć wobec nauk ścisłych i przekonanie o nadzwyczajnych trudnościach towarzyszących ich opanowaniu, brak zainteresowania zasadami działania używanych przedmiotów³. Na szczęście świat szybko się zmienia, fenomen Internetu 30 lat temu mógłby się wydawać nie mniej abstrakcyjny. Dlatego chyba nie ma powodów do zbytnej zachowawczości w przewidywaniu przyszłości.

² Na przykład komercyjny Red Hat Enterprise Linux, i darmowy CentOS

³ http://www.fuw.edu.pl/~szczytko/NT/Sprawozdania2008/Marcin_Switala_Ignorancja_konsumpcyjna_dla_nowych_tehnologii.pdf - to wręcz odwrócona w stosunku do mojej wizji, porównanie ich obu pozwala uzmysłowić sobie jak skrajnie różne wersje dalszego rozwoju sytuacji można rozpatrywać.