

Nyílt forráskódú szoftverek

Buday Gergely

2010. november 10.

1 A nyílt forráskódú szoftver

2 Licenzek

A forráskód

```
#include <stdio.h>

int main(int argc, char **argv)
{
    printf("Hello World!\n");
    return 0;
}
```

Lefordítjuk, futtatjuk

```
$ vi hello.c  
$ gcc hello.c  
$ ./a.out  
Hello World!  
$
```

Open source példák

- München, városháza: OpenOffice
- Amazon.com: Linux, Sun gépek és Solaris operációs rendszer helyett
- Apache webszerver a legnépszerűbb
- IBM sok pénzt költött az Eclipse szoftverfejlesztőrendszerre
- Enterprise Resource Planning (ERP) és Customer Relationship Management (CRM) rendszerek

Előnyök

- Nincsenek licenstdíjak - megtakarítás
- Kisebb support költségek
- Kisebb integrációs költségek
- Gyártófüggés elkerülése (avoiding vendor lock-in). Példa: .doc formátum
- Ezernyi programot használhatunk
- Többet ér az IT a cégnek

Felelősségek

- Az IT személyzetnek meg kell tanulnia a nyílt forráskódú szoftverek installálását és konfigurációját
- Több programozást igényel
- Nyílt forráskódú projektek kiértékelésében is jártasnak kell lenni
- Jobban meg kell érteni, milyen műszaki követelményei vannak a cégnek
- Jártasnak kell lenni a nyílt forráskódú licenzek területén, különösen, ha a cég szoftvereket is árul/terjeszt

Az open source vita

- A nyílt forráskód túl nagy jelentőségű ahhoz, hogy félresöpörjük
- Régen: fejlesztőeszközök, operációs rendszerek — informatikusoknak
- Ma: szoftverek teljes skálája, a fentiekén kívül web és alkalmazásszerverek, adatközpont-szoftverek, irodai alkalmazások, vállalati alkalmazások, ERP, CRM
- Egy dologra fókuszálódik a vita: nyereséges-e nyílt forráskódot használni?

Pro és kontra

- Programozók és rendszergazdák gyakran rajongóként azzal reklámozzák a nyílt forráskódot, hogy ingyen van
- A menedzserek ezt gyakran gyanakodva szemlélik, azt látva, hogy az IT szakemberek újabb kütyüket találtak a saját szórakoztatásukra, anélkül, hogy számba vették volna, ez milyen költséggel jár: kiértékelés, tesztelés, installálás, konfigurálás, testreszabás, frissítés, működtetés és support.
- Ezen kívül: hiányzik pár dolog, ami kereskedelmi szoftvereknél megvan: support, karbantartás, automatizálás és dokumentáció.
- Jó, ha van valaki, akit le lehet szűrni, akit be lehet perelni
- De: sok ezer cég sikerrel vette ezeket az akadályokat

A képességek és a termékjelleg hiánya

- Egy kritika: hosszú képzés kell ahhoz, hogy egy IT szervezet jól tudjon bánni a nyílt forráskóddal
- Termékjelleg (Productization): gyakran a nyílt forráskódú szoftverek félkészek, a konfigurációjuk nehézkes, hackereket igényel. A felhasználók jobb szeretik az ablakos-klikkelős konfigurációs felületet, mint szövegfájlok szerkesztését

Sikeres open source stratégia

- Mi az a probléma, amit meg akarunk oldani?
- Hogy tudja ezt a nyílt forráskód megoldani?
- Van-e olyan OSS (open source software), ami részben vagy teljesen megoldja a problémát?
- Hogyan tudjuk megítélni a megtalált szoftver érettségét és stabilitását?
- Milyen képességek és tudás kell a szoftver üzemeltetéséhez?
- Birtokában van-e a cég ezen képességeknek? Ha nem, hogy lehet megszerezni és intézményesíteni ezeket?
- Valóban olcsóbb-e a nyílt forráskódú opciót használni, összehasonlítva más alternatívákkal?

Innovációs stratégiák

- Innovátorok: a legújabb technológiákat használják
- Óvatos duhajok: olyan technológiákat használnak, amit már mások kipróbáltak, de azért figyelik, milyen újdonságok vannak
- Bevett gyakorlatot követők: olyan technológiákat használnak, ami bevett gyakorlat — amit már sokan használnak
- Egy Nagy Cégre mondta egy alkalmazottja: olyan technológiákat használunk, amit már senki nem használ
- Ez utóbbi viccnek tűnik, de jól meg lehet belőle élni, ld. Cobol programok bankoknál, IBM nagygépes tudás

Richard Stallman és a szabad szoftver (free software)

- Négy szabadság
- A program futtatásának a szabadsága, bármilyen célból
- Szabad legyen tanulmányozni és módosítani a programot. Ennek előfeltétele, hogy hozzáférhető legyen a forráskód.
- Szabad terjeszteni a programot — így segítve a felebarátot
- Szabad javítani a programon és a javításokat publikálni a köz javára

Nyílt forráskód kontra szabad szoftver

- Richard Stallman és a Free Software Foundation: free software. Politikai mozgalom is
- Open Source: Bruce Perens. Inkább technológiai kifejezés, a bajtársi (peer-to-peer) fejlesztési módszert hangsúlyozza
- "Open source is a community-based, iterative, incremental and evolutionary software development methodology that emphasizes experimentation and experience over planning and formal design"
- "A nyílt forráskód közösségi alapú, iteratív, inkrementális és evolutív szoftverfejlesztési módszer, ami a kísérletezést és a tapasztalatot többre tartja, mint a tervezést és a formális terveket"

Honnan jön a nyílt forráskód?

- "An itch to scratch"
- Linus Torvalds, a Linux kernel megalkotója: jó lenne, ha lenne Unix Intel processzoros számítógépeken
- Larry Wall: kellene egy programozási nyelv, amivel rendszergazdai feladatokat lehet automatizálni: Perl
- Az inspirációt kemény munka követi
- Nincs követelmény-specifikáció. Nincs piackutatás
- Ami van: néhány okos és motivált fejlesztő, aki tudja, mit akar és meg is valósítja
- Ahol az igazi nyílt forráskód kezdődik: a publikálás

Motiváció

- Státusz és elismerés
- Közösségi etika
- Racionális önérdék: piacra dobható képességek fejlesztése, csapathoz tartozás, segítség másoktól

Hogyan nő egy OSS projekt?

- Nincs jogi szervezet
- Olyan, mint egy rockzenekar
- "Fork" - elágazás

GNU GPL

- GNU is Not Unix
- GNU General Public License
- Free Software Foundation
- Richard Stallman
- A nyílt forráskódú projektek 85%-a, becslések szerint
- Linux, MySQL, C++ fordítóprogramok stb.

GNU GPL

- Mindenkinek joga van használni, lemásolni és terjeszteni
- Copyleft: bármilyen módosítás is a GNU GPL hatálya alá tartozik, azaz szabadon hozzáférhetővé kell tenni
- Nem lehet GPL-s szoftvert módosítva saját terméket csinálni úgy, hogy a módosítást nem adjuk ki
- A kritikusok ezt *vírusos licenznek* nevezik.

GNU LGPL

- GNU Lesser Generic Public License
- Programkönyvtárakra, pl. van egy adatbáziskezelőnk, annak egy lekérdező-könyvtára
- Az ezt a könyvtárt használó programot nem kell publikálnunk

BSD Licenz

- Berkeley Software Distribution
- Motiváció: a szoftver adófizetők pénzéből készült egy állami egyetemen, ezért legyen mindenki számára hozzáférhető
- Lehetséges a BSD kódot módosítani és nem nyilvános forráskódú terméket fejleszteni
- Elemei: Copyright, Disclaimer és felhasználási feltételek
- A Mac OS X-hez a FreeBSD nevű operációs rendszert használták fel

BSD Disclaimer

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.