

Descobrim les possibilitats del programari lliure Menorca, 11-2007

Copyright © 2006 Benjamí Villoslada Gil (benjami@bitassa.cat)

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU Free Documentation License".

Què és el programari lliure

- **Les quatre llibertats segons la FSF**
 1. La llibertat d'executar el programa per qualsevol propòsit (llibertat 0).
 2. La llibertat de veure com funciona el programa i adaptar-lo a les necessitats pròpies (llibertat 1). L'accés al codi font és un requisit.
 3. La llibertat de redistribuir còpies (llibertat 2)
 4. La llibertat de millorar el programa i de distribuir-lo de nou amb les millores realitzades, per tal que tota la comunitat se'n pugui beneficiar (llibertat 3). Igual que a la llibertat 1, l'accés al codi font és un requisit.
- **En diem privatiu** del programari que no té les 4 llibertats, perquè priva 1, 2, 3 i 4. Són privacions importants.

Com va sorgir

- L'experiència del **Richard Stallman**
- El projecte **GNU**
- La protecció **legal**
 - La **FSF** per aconseguir fons i donar forma legal al projecte
 - La llicència **GNU** per a protegir la creació
- Detalls del projecte GNU i la FSF a <http://www.culturalliure.cat/livre/programari-lliure-societat-lliure/>

Un caprici de bojos pels ordinadors?

- **El programari és un bé bàsic.**

- **És l'eina del moment per a llegir, escriure, veure i escoltar:**
 - Forma part de l'oci i l'entreteniment
 - De l'ensenyament
 - De l'art
 - De la investigació
 - La literatura
 - Dels tràmits amb l'administració
 - Amb els bancs
- Philip G. Armour va escriure:
 - *«despite common practice, developing software is not a product manufacturing activity. Rather, it is a knowledge acquisition activity»* --«Not-Defect: The Mature Discipline of Testing». The Business of Software. Communications of the ACM, octubre de 2004.
- Armour també escriu a la mateixa revista «The Five orders of Ignorance» (octubre del 2000) i l'anterior «The Case for a New Business Model (agost del mateix any)». Als dos articles hi observem els mateixos arguments, encara que més desenvolupats, sobre el programari com a dipositori predilecte del coneixement humà. Armour distingeix **cinc maneres conegudes d'emmagatzemar el coneixement** –ADN, el cervell i la memòria, eines i aparells, llibres i programari-- i n'analitza les característiques, avantatges i inconvenients de cadascun.
 - **ADN:** És el primer mètode per emmagatzemar el coneixement. Conté la informació necessària per a crear vida. El coneixement hi està profundament incrustat, passar cap a un altre grau és obligatori per a la supervivència de les espècies. El coneixement és persistent, però s'actualitza poc a poc. No tenim la capacitat de canviar-lo –encara, o sí... intencionadament. El DNA pot fer créixer un objecte físic que interactua i modifica l'entorn.
 - **Cervell:** És un «experiment» gairebé exclusiu de l'espècie humana: emmagatzemar més coneixement, al cervell, del que s'hereta amb el DNA. Fem servir el nostre cervell per a guardar el

coneixement que adquirim i va ser el segon mètode per a emmagatzemar-lo que varem conèixer. El coneixement és molt volàtil, però podem canviar-lo ràpidament i a voluntat. Podem usar els coneixements per afectar i modificar el món.

- **Màquines i eines:** El valor més important d'una eina no és ella en sí, sinó com va ser creada i modificada. El coneixement del creador de les eines és allò que marca les diferències. Sovint també se'n parla com «coneixement sòlid» i va ser la tercera manera d'emmagatzemar el coneixement que varem conèixer. El coneixement de les eines és força persistent i mal d'actualitzar. És intencional i existeix per afectar el món exterior.
- **Els llibres:** Han permès nous mètodes per a dipositar i fer accessible un coneixement que fins aleshores estava al cervell. Van veure que el coneixement fos portable en el temps i l'espai. És molt persistent però d'actualització lenta. Encara que els llibres són intencionals, no tenen capacitat per a canviar el món.
- **Programari:** És la darrera forma coneguda -des de tant sols 50 anys-- per a emmagatzemar el coneixement. Després d'uns inicis dubitatius, creix a una velocitat vertiginosa. Una gran quantitat de persones hi fan feina per a obtenir informació des de les fonts més diverses, comprendre-la, classificar-la i traslladar-la al medi programari, amb el que intenten validar tot el coneixement traslladat. Hi ha una raó que fa invertir-hi tant esforç: aquest medi té les característiques que volem i que no són presents a d'altres medis: es intencional, persistent, d'actualització senzilla i ràpida.
Però sobretot és actiu.

Ha de tenir amo el programari?

- Està fet amb una matèria primera revolucionària, els bits. Amb els bits és important l'autor del primer exemplar que va servir per a fer-ne còpies infinites a baix cost. **Però si l'autor fa d'amo, aleshores tenim dilemes socials:**
 - Sabem que passar una còpia és fàcil, però **els fabricants de programari privatiu ens diuen «pirates»** amb aquest argument *«El benefici per al veí, pel fet de donar-li una còpia d'un programa, queda contrarestat pel perjudici provocat al propietari»*.
 - Acceptar què és «pirateria» significa dir que **el perjudici i el benefici tenen la mateixa magnitud**. Nosaltres li donam més importància a les qüestions socials, això és, que no hi hagi entrebancs per una cosa tant senzilla com compartir coneixement en forma de programari --fer-ne còpies-- i que així tothom pugui tenir els programes que necessita.
 - **Per a entendre-ho**, imaginem que existissin màquines per a fer còpies de pa, amb un cost marginal proper a zero per còpia, i apliquem-li la frase anterior: ***«El benefici per al veí, pel fet de donar-li una còpia del nostre pa, queda contrarestat pel perjudici provocat al forner»***
 - No traslladem el model dels àtoms als bits:
 - «Si et dono el meu pinzell, em quedo sense. Però **si et dono una còpia del meu programa de dibuix, tots dos tenim pinzells informàtics»**.

La defensa dels amos

- **Ninguna metàfora no és perfecte.** Algú pot pensar que hi ha **diferències entre el pa i el programari:** basta amb un pa inicial, però en el cas del programari cal la feina de creació de molts originals.

- **Analitzem què ens diuen els amos:**
 - **Argument emocional:** *«Aquest programa porta la meva suor, el meu cor i la meva ànima. L'he fet jo, és meu!»*
 - Respostes:
 - Molts informàtics estan disposats a oblidar la seva propietat a canvi d'un sou.
 - Molts informàtics estan disposats a fer feina només per la utilitat dels seus serveis, sense cap argument emocional sobre la propietat.
 - **Argument Econòmic:** Amb el to d'un consell savi i amistós: *«Vull ser ric (sovint dissimulat amb un 'guanyar-me la vida') i, si no deixeu que em faci ric programant, aleshores no programaré. Tothom pensa com jo, de manera que ningú no programarà mai. I aleshores us trobareu que no tindreu cap programa»*
 - Resposta: És una fal·làcia que obliga a pensar que ha d'haver-hi programari privatiu o absència de programari. L'experiència mostra que no és imprescindible la presència d'un amo per a desenvolupar programari.

Què en perdem: quatre perjudicis

1. Menys gent usa programari

- Prohibir la còpia fa que sigui més complicat accedir-hi
- Menys qualitat de vida a la Societat de la Informació i el Coneixement.

2. Hi ha problemes socials (necessitat de permisos, culpabilitat, prohibit compartir, falta de contacte amb els autors, programadors cínicos)

- Necessitam permís per a llegir els nostres propis treballs
- L'autor té un poder desmesurat sobre els usuaris, encara que el programa sigui «gratis»:
 - *«[...] Però per parlar assenyadament, la subjecció a un amo de qui ningú no en pot estar mai segur que sigui bo, perquè té sempre el poder de ser dolent quan vulgui, és una extrema dissort. I tenir molts amos és, com més se'n tinguin, ser més vegades extremadament dissortat»* -«La servitud voluntària», Étienne de la Boétie (1530-1563).
- Queda prohibit compartir, ajudar al veïnat. Hi ha culpabilitat quan es fa.
- No tenim contacte amb els autors: com els hi demanarem res si la nostra còpia és «pirata»?
- El cinisme dels programadors.
 - 3 Escola cínica: Escola filosòfica grega que aspirava a l'autosuficiència moral de l'individu, lligada a una crítica de la civilització i de les convencions socials i a una apologia de l'estat natural. -el Fabra.
 - Imaginem un programador que descriu amb gran entusiasme la seva feina perquè la troba tècnicament apassionant. Aleshores, quan li preguntem «que podré fer servir el programa i veure tot el que expliques?» el posem en un problema que no pot superar si no és amb cinisme, ço çes,. sent crític amb la convenció i potser tirant de l'estat natural del bit: «et dono una còpia il·legal, no ho diguis, i tal dia farà un any».

3. Ningú no el pot millorar

- Normalment el propietari manté en secret el codi per por que algú en pugui aprendre alguna cosa. Aquesta situació ja és prou negativa per

a una societat que serà millor en proporció a la quantitat de coneixement que hi circuli.

- Prohibir les millores significa renunciar a un dels grans avantatges respecte a tecnologies anteriors basades en àtoms en comptes de bits: és tant fàcil millorar un programa! –quan s'és programador, clar.
 - Ens ha de preocupar si no sabem programar? Tant com si el nostre govern ens permet viatjar a l'estranger, encara que no tinguem pensat fer-ho.
- Estem en mans de l'autor:
 - – Com canvio aquesta recepta per treure'n la sal?
 - Com goses posar en dubte la meva recepta, la creació del meu cervell i el meu paladar, prenent manipular-la? No tens el judici suficient com per modificar la meva recepta i fer que funcioni correctament!
 - Però el metge em diu que no prengui sal! Què puc fer? Vostè podria treure-hi la sal per a mi?
 - Ho faria encantat: els meus honoraris són de només 12.000 Euros. Ara, però, no tinc temps. Estic ocupat amb un encàrrec per dissenyar unes galetes pel Carrefour. Potser d'aquí a un parell d'anys podré estar per tu.

4. En perjudica el desenvolupament

- Antigament, fins a la darreria dels 70, el desenvolupament de programari era un procés evolutiu. Algunes parts del programa s'aprofitaven com a base per a altres programes. L'existència de propietaris impedeix aquest tipus d'evolució. Cada cop que es desenvolupa un programa cal partir de zero.
- Els nouvinguts no poden estudiar els programes existents i aprendre-hi tècniques útils. Hi ha estudiants d'informàtica que acaben la carrera i mai no han vist el codi d'un programa gran.
- Newton deia: «Si és que he pogut veure més enfora que els altres, ha estat gràcies a que he pogut colcar damunt les espatlles dels gegants que m'han precedit.» És una frase que sovint s'usa en parlar de ciència. En el cas del programari privatiu, caldria canviar-la per «Si és que he pogut veure més enfora que els altres, ha estat gràcies a que he pogut colcar damunt les espatlles dels empleats de la mateixa empresa que m'han precedit».

N'hi ha prou que els informàtics es guanyin la vida

- La pregunta «com podem pagar els programadors?» se simplifica quan ens adonem que **no es tracta de pagar-los una fortuna. N'hi ha prou amb que es guanyin la vida.**
 - Hi ha la idea que la informàtica fa rics, però és falsa des de fa temps.
 - Si miram al nostre voltant, hi ha més possibilitats de fer-se ric amb la **loteria** que no pas fent programari privatiu.
 - Més camps amb poques possibilitats on hi fa feina gent perquè els apassiona, **ho troben divertit i que té valor social:**
 - Lògica matemàtica
 - Música clàssica
 - Arqueologia
 - Art
 - Si descartem la possibilitat d'enriquir-se enormement, al cap d'un temps, quan ens en fem la idea, aleshores recuperarem el gust de treballar en programari just pel plaer de la realització personal, perquè programar és divertit.
- **No comporta fer cap vot de pobresa.**
 - Hi ha molts oficis que no són sinònim de fer-se rics. Es basen en aconseguir feines, cobrar-les i cedir-les a qui les va encarregar, sense restriccions.
 - No proposem cap model de negoci estrany. Milions de professionals es guanyen la vida en condicions semblats:
 - **Advocats** que fan feina amb codi públic: les lleis.
 - Qui construí i condicionà aquest local, que ara pot modificar qualsevol altre.
 - El programari lliure també serveix per a fer grans negocis: **Red Hat.**
- **Quan tot el programari sigui lliure, aleshores** sorgiran noves oportunitats de negoci avui gens freqüents:
 - Avui la relació entre els usuaris i fabricants de programari privatiu està distorsionada: la majoria de clients són «pirates» i s'amaguen davant els autors.

Desafiaments (avui) per a usar programari lliure

- **Problemes tècnics:** cal instal·lar-lo
 - Els ordinadors nous arriben amb el Windows
 - No tot el maquinari funciona
 - És en vies de solució, però **no depèn de nosaltres**, excepte la pressió que puguem fer com a consumidors.
- **Problemes socials:** cal anar contracorrent
 - Usar el GIMP en comptes del Photoshop
 - Usar l'OpenOffice.org en comptes del Microsoft Office
 - Etc.
 - Llocs on obliguen a usar programes no lliures
 - Feina
 - Comerç
 - Sanitat
 - Les revistes en parlen poc, però les comunitats a Internet ho compensen
 - **Depèn de nosaltres**
- **Beneficis tècnics**
 - N'hi ha molts, però no importa esmentar-los: la llibertat no hauria de menester justificacions tècniques. Per això, al llarg de tota la xerrada només he parlat dels motius socials.