

# Open Source

*...men hvad med software patenter?*

Peter Toft  
<pto@sslug.dk>  
SSLUG

# Peter Toft

- ★ Civilingeniør, PhD
- ★ Algoritme- og software-udvikler
- ★ Bestyrelsesformand for SSLUG på 3. år
  - ★ *Skåne Sjælland Linux User Group*
  - ★ Nok verdens største Linux gruppe
  - ★ Snart 6000 medlemmer, 28 emailister
- ★ 6 Linuxbøger, artikler og foredrag
- ★ Anvendt Linux dagligt siden 1994
- ★ Senior Design Engineer hos Nokia

# Oversigt

- ★ Hvad betyder Open Source
- ★ Software udviklet under Open Source
- ★ Linux development model (LDM)
- ★ Patenter

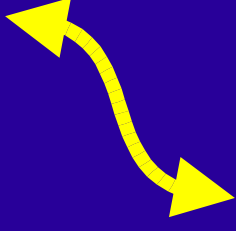
*Baggrund: EPO ønsker at tillade patenter på computer software*

Software

Internet

Programmer

Operativsystem



# Hvordan udvikler man software til computere?

- ★ Operativsystem
  - ★ Windows 98/2000/NT
  - ★ Linux, FreeBSD, MacOS, UNIX
- ★ Programmer
  - ★ Word, Netscape, epost-program
- ★ Internet
  - ★ Hjemmesider og indhold

# Hvad er kildekode?

```
RGB_Image* loadRGB(char *filename)
{
    int ii,jj,MN,kk,ll;
    FILE *RGBfile;

    RGB_Image *myrgb;
    unsigned char *imar;

    if (debug>0)
        printf("Reading from %s\n",filename);
    imar = (unsigned char *)malloc(3*ImageM*ImageN*sizeof(char));
    myrgb = (RGB_Image *)malloc(ImageM*ImageN*sizeof(RGB_Image));
    RGBfile=fopen(filename,"rb");
    if (!RGBfile) {
        printf("Image corrupt\n");
        exit(1);
    }
    MN = fread(imar,sizeof(unsigned char),3*ImageM*ImageN,RGBfile);
```

Det er det en  
programmør  
skriver



# Fra kildekode til program

★ Oversæt den læsbare kildekode til det maskinen kan forstå.

★ Man compiler/oversætter programmet

```
if (debug>0)
    printf("Reading from %s\n", filename);
imar = (unsigned char*)malloc(3*ImageM*ImageN*sizeof(char));
myrgb = (RGB_Image*)malloc(ImageM*ImageN*sizeof(RGB_Image)
```

01001010010101010010100111101111  
00001010101010001011100101001010

★ Programmet kan køres men ikke læses

# Closed Source

- ★ Man kan ofte gemme smarte konstruktioner ved ikke at give kildekoden ud
- ★ Microsoft bruger Closed Source

```
if (debug) printf("Reading from %s\n", filename);  
printf("Reading from %s\n", filename);  
image = (unsigned char*)malloc(sizeof(char));  
myrgb = (RGB_Image*)malloc(sizeof(RGB_Image));  
01001010010101010010100111101111  
00001010101010001011100101001010
```

- ★ Apple er ved at gå til Open Source
- ★ UNIX er ved at blive Linux

# Open Source

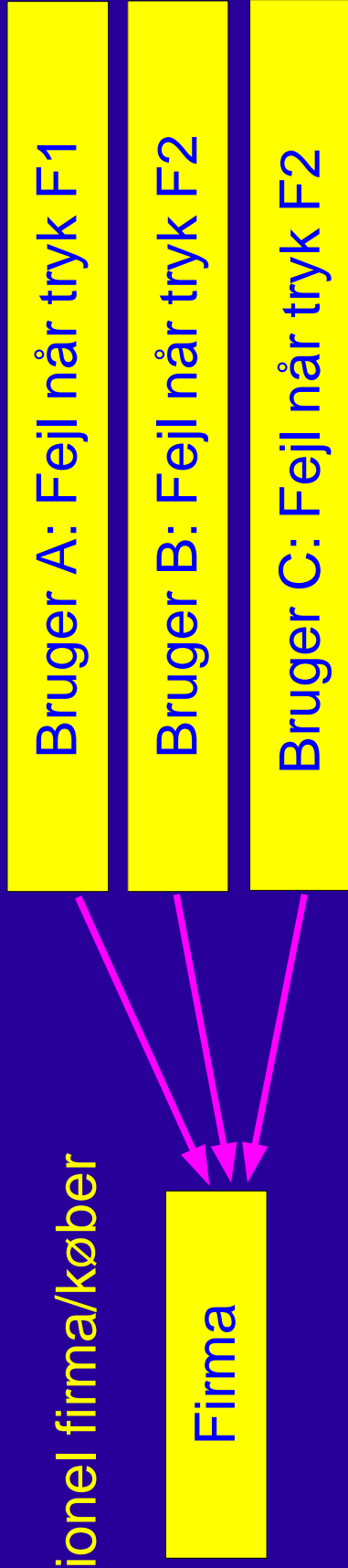
- (1) Du må lave kopier af programmet –husk dog alle copyrights
  - *Du må godt tage et vederlag for at lave kopien*
- (2) Du må ændre i programmets kildekode og lave nye versioner under samme licens
- (3) Du skal tilbyde kildekoden med sammen med programmet

# Linux Development Model

- ★ Hurtige releases
- ★ Peer review af kildekoden
- ★ Alle kan deltage
- ★ Bedste løsninger vinder
- ★ Dårlig kode dør ud

# Linux Development Model (II)

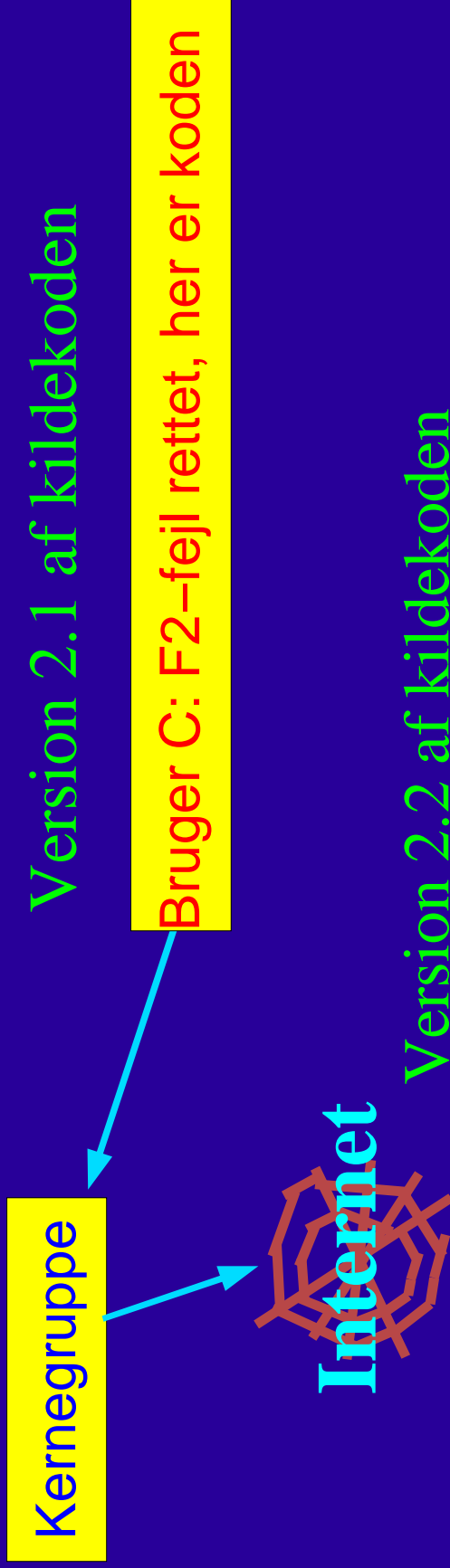
Traditionel firma/køber



Linux development model



# Linux Development Model (III)



```
if (debug>0)
    printf("Reading from %s\n",filename);
imar = (unsigned char*)malloc(3*ImageM*ImageN*sizeof(char));
myrgb = (RGB_Image*)malloc(ImageM*ImageN*sizeof(RGB_Image))
```

*Kildekoden lægger vi på Internet – alle kan se hvad vi laver*

## Hvad driver Open Source

- ★ *”Vi har selv fået så meget af andre – nu giver vi tilbage”*
- ★ *”Det er sjovt at være med til at skabe ting andre er glade for at bruge”*
- ★ *”Kan få programmet eksakt som vi ønsker”*
- ★ *”Friheden til at vælge”*

## Brugeren vinder ved Open Source

- ★ Brugeren kan vælge leverandør
- ★ Brugeren kan vælge en ny leverandør
- ★ Brugeren er i mindre grad afhængig af producent
- ★ Prisen på Open Source software er ofte meget lav ...
  - ★ *Eller gratis*
  - ★ SERVICE er fokus

# Open Source og softwarepatenter

- ★ EPO ønsker at tillade patentering af software
  - ★ EPO = Europæiske Patent Office
- ★ USA tillader patentering .... af alt ... SUK!
  - ★ Det er løbet helt af sporet
  - ★ Vi skal harmonisere med USA (TRIPS-aftalen)
  - ★ Chance for at overtræde software patenter er stor
- ★ Open Source bliver klemt!
  - ★ Vores kildekode er på Internet
  - ★ Closed source

# US patent: Windowing

- \* Windowing: Bestemmelse af 4-cifret årstal ud fra 2-cifret årstal
  - \* Hvis  $\text{år} > 30$  så  $\text{år} = \text{år} + 1900$
  - \* Ellers  $\text{år} = \text{år} + 2000$
- \* Eks: 86  $\Rightarrow$  1986 og 00  $\Rightarrow$  2000
- \* United States Patent 5,806,063
- \* Dickens September 8, 1998

# US patent: Windowing (II)

Date formatting and sorting for dates spanning the turn of the century

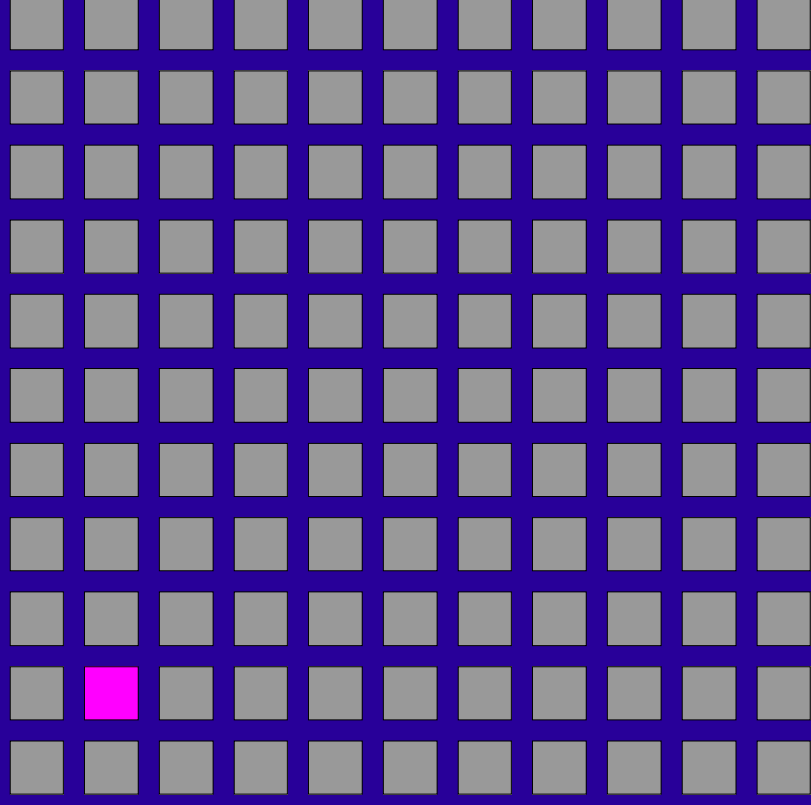
## Abstract

Dates stored in symbolic form in a database are reformatted to permit easy manipulation and sorting of date-related information. Each date in M.sub.1 M.sub.2, D.sub.1 D.sub.2, and Y.sub.1 Y.sub.2 format is converted to C.sub.1 C.sub.2, Y.sub.1 Y.sub.2, M.sub.1 M.sub.2, and D.sub.1 D.sub.2 format. To accomplish the conversion, a 10-decade window starting on Y.sub.A Y.sub.B is defined that encompasses all dates in the database. The value of C.sub.1 C.sub.2 is determined by the relative values of Y.sub.1 Y.sub.2 and Y.sub.A Y.sub.B. The reformatted date information is particularly useful when the reformatting is in C.sub.1 C.sub.2 Y.sub.1 Y.sub.2 M.sub.1 M.sub.2 D.sub.1 D.sub.2 format, because sorting by date is accomplished using a pure numerical-value sort.

# Jeg udtager et patent

■ Min patenterede ide

■ Andre oplagte  
programdele



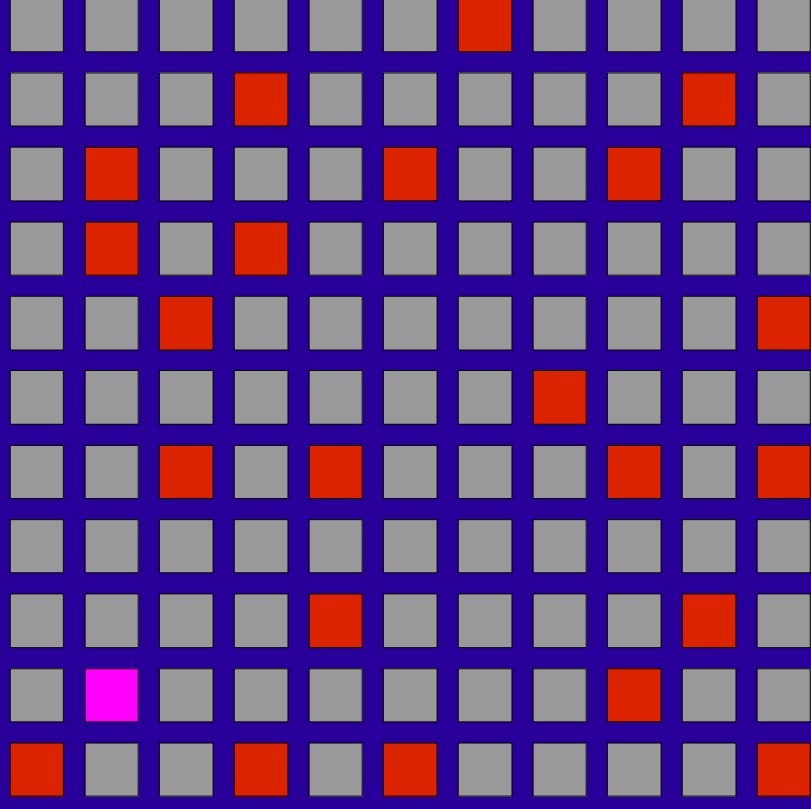
$$\text{Profit} = \text{Salg} - \text{Udvikling} + \text{P--Indtjening}(\text{■}) - \text{P--Udgift}(\text{■})$$

# Krig på software patenter

■ Min patenterede ide

■ Andre oplagte  
programdele

■ Risiko for patent  
sagsanlæg



$$\text{Profit} = \text{Salg} - \text{Udvikling} + \text{P} - \text{Indtjening}(\text{■}) - \text{P} - \text{Udgift}(\text{■}) - \text{P} - \text{udgift}(\text{■})$$

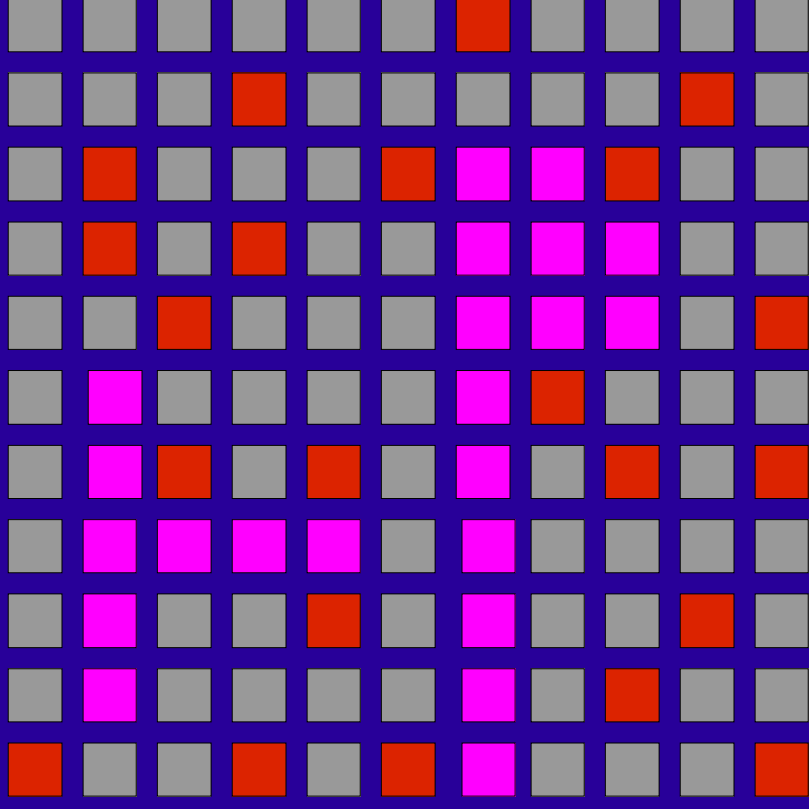
# Vinde på software patenter

■ Mine patenterede  
ideer

■ Andre oplagte  
programdele

■ Risiko for patent  
sagsanlæg

*Man skal være stor så man kan  
bytte patenter med andre*



**Profit=Salg-Udvikling+P-Indtjening(■)-P-Udgift(■)-P-udgift(■)**

# Vindere og tabere ved software patentering

Vindere bliver

- ★ Store firmaer *Bytter*
- ★ Patentagenter – producerer ikke. *Handler!*
- ★ USA *Forspring og lav standard*

Tabere

- ★ Små firmaer *I praksis*
- ★ Open Source verdenen
  - ★ *Små fisk uden penge – kan ikke bytte*
  - ★ *Hvad ville du gøre hvis der anlægges sag?*
- ★ Samfundet – dyrere software

# Patenter

## Closed Source

- \* Ingen kildekode
- \* Kildekode fra firma som tjener penge
- \* Næsten umuligt at checke patentbrud
- \* Penge bag ved program

## Open Source

- \* Vi publiserer fuld kildekode på Internet
- \* Tjener oftest ikke penge på at udvikle kildekode
- \* Alle kan se patentbrud
- \* Ofte er der ikke penge bag udvikleren

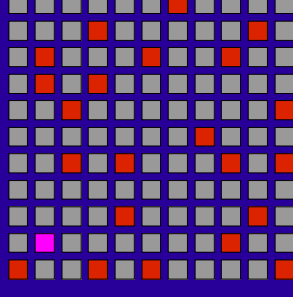
## Kan man vurdere software patenter?

- ★ (Artikel 52) European patents shall be granted for any inventions which are susceptible of industrial application, which are *new* and which involve *an inventive step*.
- ★ Undtagelse; *computer programs*

# Prior art og opfinderhøjde

- ★ Patent-agenter *KAN IKKE* kontrollere for prior art
- ★ Patentkontor kan søge i egne databaser – ikke kildekode
- ★ Patentkontor kan HØJST søge i 1% af prior art
  - ★ Open Source skriver kildekode ikke patent-ansøgninger
- ★ Det er ikke vores opgave at være censor for patentkontoret!

# Europa



- \* 60% af BNP kommer fra små og mellemstore firmaer i Europa
- \* *EU-kommisær Erkki Liikanen*
- \* Meget stor del af programmer som skabes er Open Source

## Hvad vil vi med software patenter ?

- ★ Patenter er en aftale mellem samfund og opfinder
- ★ Opfinder får eneret til at udnytte patentet i tyve år
  - ★ *20 år i IT er mere end 100 år i samfundet*
- ★ Samfundet får indsigt i opfindelsen
  - ★ *IFO-rapport –Næsten ingen videnspredning*
- ★ Software patenter giver ikke videnspredning
  - ★ *IFO-rapporten støtter dette – specielt små og mellemstore firmaer*
- ★ Hvem skal betale?
  - ★ Kunder, samfund og konkurrenter

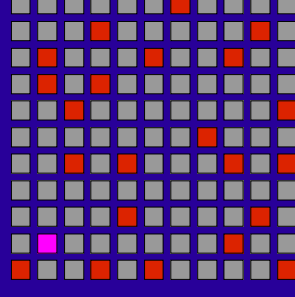
# Fangernes dilemma

A +	2	0
A -	5	4

B+ B-

Antal år for fange A i spjældet alt efter om fange A og B samarbejder eller modarbejder

- ★ + er samarbejde og - er modarbejde
- ★ Fange A bør ikke samarbejde
- ★ Du skal udtage patentet – *egoistisk holding*
- ★ Fange B vil så heller ikke samarbejde
- ★ Begge taber ...
- ★ Patentsystemet lægger op til at *jeg* skal med



# Patenter giver os penge og sikkerhed for vores R&D

- ★ Efter USA (ca. 1985) indførte softwarepatenter:
  - ★ Antal software patenter voksede eksplosivt
  - ★ Blandt de største patentholdere i USA faldt R&D udgifter i forhold til salg
  - ★ Signifikant bremse for videreudvikling
  - ★ *James Bessen and Eric Maskin, MIT*

## Fra plov til information

- \* Patent på en plov svarer ikke til patent på software!
- \* Plov-patent rammer ikke et kopimaskine-patent
- \* Software patent kan ofte ramme *MANGE* fagområder
- \* Ofte meget svært eller umuligt at omgå software patentet
- \* Tænk musik og bøger *ikke* søm og skruer

# Alternativer

- \* Closed Source gemmer meget information
- \* Open Source kan/vil ikke gemme information
- \* Ophavslovgivning
- \* Langt nemmere at sammenligne to produkter
- \* First-move
- \* Vær først og bedst med det rette produkt
- \* Pjat at man ikke kan klare sig uden patenter – *Microsoft*