

Bjarte Johansen

# Fri programvare og åpen kildekode

## *Fra vitenskapelige idealer til essensielle rettigheter*

**Fri programvare og åpen kildekode går utover den avgrensede retten til å bruke programvaren vi installerer på maskinene våre: Det sentrale i disse ideene er brukerens rett til å endre programvaren etter eget behov, og ikke minst retten til å distribuere disse endringene. Ettersom programvare blir en stadig større del av livene våre, så er fri programvare den beste måten å sikre retten til privatliv og friheten til egne data.**

Helt fra de tidligste datamaskinene har fri programvare og åpen kildekode vært sentrale ideer i programmeringsverdenen. De første kommersielle maskinene fra 1950- og 60-tallet ble som oftest levert med brukermanualer og full kildekode til operativsystem og kompilatorer som kjørte på maskinen. De første brukerne av disse maskinene var også for det meste universiteter, og i tråd med vitenskapelige idealer ble endringer og forbedringer delt bredt med andre brukere på tvers av universitetene. Senere, på 1970- og 80-tallet, begynte imidlertid bedriftene som utviklet datamaskinene og andre bedrifter å oppdage verdien av programvare. Dermed ble det vanligere å ikke gi tilgang til kildekoden, og i stedet tilby konsulenttjenester til brukerne av datamaskinen.

En av de største profilene innen fri programvare er Richard Stallman. Han er en av grunnleggerne av GNU prosjektet, som har som målsetning å lage et helt fritt operativsystem fra kjernen av og helt opp til alle programmene som kjører i brukergrensesnittet. Dette innebærer at all programvare, fra den som styrer prosessoren og kontrollerer lagring, til den som for eksempel styrer nettleseren, skal kunne inspiseres, endres og distribueres av brukerne. Det gjelder også for kompilatoren,

som er programvaren som tar kildekode og gjør den om til maskinkode.

Stallman er også forfatteren av GNU Manifestet, som beskriver de fire essensielle frihetene brukerne av programvare skal ha: 1) friheten til å kjøre et program uansett formål; 2) friheten til å studere koden og foreta endringer; 3) friheten til å distribuere programvaren; og 4) friheten til å distribuere endringer.

Stallman skrev mange av de viktigste komponentene selv gjennom sitt arbeid med GNU prosjektet. Han står blant annet bak GNU C Compiler (GCC) og GNU Debugger (GDB), komponenter som trengs for å kompilere og lettere finne feil i programvare. På grunn av det toneangivende arbeidet til Stallman, forventer de fleste utviklere som i dag lærer å programmere at kompilatorer og annen støtte til programmeringsspråk er fritt og åpent tilgjengelig.

Det er nesten utelukkende i nisjesituasjoner at man finner "lukkede" kompilatorer. Microsoft har for eksempel lenge vært tilbakeholdende med sine programmeringsspråk, og har generelt ikke vært vennlig innstilt til fri og åpen programvare. Men selv de har etterhvert frigjort deler av sin infrastruktur. Det var for eksempel svært overraskende for store deler av programmeringsmiljøet at Microsoft frigjorde The Common Language Runtime (CLR) i 2015, programvaren som ligger bak mange

---

**Bjarte Johansen**, universitetslektor, Institutt for informasjons- og medievitenskap, Universitetet i Bergen.

av Microsofts programmeringsspråk. Dette i sterk kontrast til Steve Ballmers tid som CEO for Microsoft, der han blant annet omtalte det største frie og åpne operativsystemet GNU/Linux som en “kreftsvulst”.

## Fri programvare versus åpen kildekode

Ved siden av GNU prosjektet, har Stallman også grunnlagt The Free Software Foundation (FSF)<sup>1</sup>, som er en viktig grunnpilar og støtte-spiller for samfunnet rundt fri programvare og åpen kildekode. Dette gjør de ved å bidra til å finansiere utviklingen av fri programvare, og ved å tilby juridisk assistanse til utviklere som kommer på kant med “patenttroll”, eller får andre problemer som resultat av at de jobber med fri programvare. FSF jobber dessuten også aktivt mot patenter for programvare og digital rettighetsadministrasjon, som de anser som trusler mot fri programvare.

Stallmans verdier er i prinsippet de samme som ligger til grunn for ideen om åpen kildekode, men det er mange som føler at FSF – og kanskje spesielt Richard Stallman – er for dogmatiske i sin tolkning av fri programvare som et politisk og etisk valg på den ene siden, og proprietær programvare som uetisk på den andre. Her kan et eksempel på proprietær kode være driveren til datamaskinens grafikkort, altså programvaren som gjør det mulig å tegne bilder på skjermen.

Som en reaksjon på FSFs rigide tolkning av fri programvare har begrepet “åpen kildekode”, promotert gjennom Open Source Initiative (OSI)<sup>2</sup>, vokst fram som et mer pragmatisk alternativ. For tilhengere av åpen kildekode er det for eksempel akseptabelt å tillate proprietær kode i GNU/Linux, slik at operativsystemet skal virke på flere plattformer. Motsvarende vil nok Stallman og forkjemperne for fri programvare mene at dette er uetisk, og heller gå inn for å tvinge alle som vil benytte denne programvaren til også å dele sin kode. Begge parter er altså enige i om at brukeren skal ha frihet, men de har forskjellige ideer om hvordan man best kan nå dette målet.

## Fri og åpen programvare blir stadig viktigere

I dag føles fri programvare og åpen kildekode nærmest som en selvfølge. Vi ser at store selskaper som Facebook, Apple og Google både anvender, utvikler og promoterer store og viktige prosjekter som er åpne for alle. Vi ser dette hos Facebook som frigjorde React, et rammeverk for å lage brukergrensesnitt for vevsider. Tilsvarende har Apple frigjort LLVM, som er en samling kompilatorer for ulike språk, og Webkit, motoren som driver nettleseren Safari. Google har på sin side frigjort Tensorflow, en plattform for dyp læring og kunstig intelligens. Siden 2005 har dessuten Google drevet programmet Google Summer of Code<sup>3</sup>, der prosjektkoordinatorene for fri og åpen programvare kan søke penger til å belønne studenter som ønsker å jobbe på prosjektene deres.

Vi ser også at meningene til de som jobber profesjonelt med programmering er i endring. På midten av 80-tallet var flesteparten av de som var opptatt av fri programvare personer som Stallman. Dette var folk som vokste ut av hacker-miljøet ved MIT og lignende miljøer av akademikere og entusiaster som var opptatt av å lære, hjelpe andre og vise fram ferdighetene sine. Over tid og i takt med utbredelsen av internett har disse (semi-)isolerte miljøene utviklet seg til et verdensomspennende meritokratisk samfunn av programvareutviklere og likesinnede. Selv om det meritokratiske idealet kanskje ikke alltid fungerer som intendert, er det uttrykk for en felles forståelse og et sett med delte verdier som definerer samfunnet rundt fri programvare og åpen kildekode.

Det kan virke som vi i dag er vitne til begynnelsen på en eksponentiell vekst i tilgjengeligheten av fri og åpen programvare. Det blir stadig mer populært å bruke plattformer som Github for å gjøre tilgjengelig små og store prosjekter. Slike plattformer fungerer som sosiale nettverk for fri deling av kildekode, og kan fungere som både skytevegg og portefølje for potensielle arbeidsgivere. Samtidig legger disse plattformene også til rette for å gi noe tilbake til miljøet, noe dagens utviklere anser som svært viktig.

Jeg vil påstå at det er svært få utviklere av programvare i dag som ikke i det hele tatt benytter seg av åpen kildekode eller fri programvare. Særlig gjelder dette om man jobber med utvikling for veven, ettersom de aller fleste teknologiene man trenger for at veven skal fungere i utgangspunktet er åpne. I tillegg er populære og lett tilgjengelige nettlesere som Firefox basert på åpen kildekode. Det samme gjelder de mest populære verktøyene for å utvikle interaktive nettsider, for eksempel Facebooks React, som alle og enhver kan gjøre hva de vil med.

Selv ble jeg trukket inn i dette miljøet da jeg lastet ned en modul til GNU Emacs, en teksteditor for programmerere som med tiden er det prosjektet jeg har jobbet mest med. Det var noen småting som irriterte meg, og siden jeg hadde kildekode og kjente programmeringsspråket, tenkte jeg at jeg kunne forsøke å fikse det selv heller enn å vente på at noen andre gjorde det. Så jeg rettet feilene og sendte inn en endringsforespørsel. Da prosjekteieren aksepterte endringene, ble jeg virkelig stolt! Dette var første gang jeg hadde bidratt til et prosjekt som ble aktivt brukt av andre enn meg selv – og som til og med var viktig for

andres arbeid. Siden den gang har jeg forbedret og utviklet flere moduler som hjelper brukeren til å håndtere kildekode i ulike programmeringsspråk. Dette arbeidet har ikke bare ført til stolthet. Jeg har også hatt noen flau opplevelser, hvor jeg har introdusert feil i moduler jeg har arbeidet med og gjort det vanskelig for andre. Men det er slik man lærer, og det viktigste for meg er dog å kunne dra nytte av kunnskapen, bygge videre på den og gi tilbake det jeg kan.

Jeg håper dessuten at fri og åpen programvare skal bli viktig også for de som ikke regner seg som programmerere og utviklere. Ettersom programvare blir en stadig større del av livene våre, mener jeg fri og åpen programvare er den beste måten å vinne tilbake retten til privatliv og råderetten over egne data. Dette bidrar også til økt ansvarlighet, siden hvem som helst kan etterprøve at programvaren fungerer som den skal – samt hva den faktisk gjør. Nettopp det er kanskje den viktigste grunnen til at jeg mener forskningsarbeid alltid skal utføres med fri programvare. Da kan forskningsprosjekter testes, etterprøves og alle antagelser sjekkes, i tråd med de akademiske idealene for vitenskapelig arbeid.

## Noter

1. Se Free Software Foundation (u.å.).
2. Se Open Source Initiative (u.å.).
3. Se Google Summer of Code (u.å.).

## Referanser

- Free Software Foundation (u.å.). About Free Software Foundation. [Online.] Tilgjengelig fra <https://www.fsf.org>. Besøkt 03.05.2018.
- Google Summer of Code (u.å.). About Google Summer of Code. [Online.] Tilgjengelig fra <https://summerofcode.withgoogle.com/about>. Besøkt 03.05.2018.
- Open Source Initiative (u.å.). About Open Source Initiative. [Online.] Tilgjengelig fra <https://opensource.org/>. Besøkt 03.05.2018.