
Nybyggere i noosfæren

af Eric S. Raymond

Dette er en oversættelse af Eric S. Raymonds *Homesteading the Noosphere*. Den engelske original findes på <http://www.catb.org/~esr/writings/homesteading/homesteading/>. Oversættelsen er foretaget af Jesper Laisen med kommentarer og kritik af Ole Michaelsen. Desuden tak til Donald Axel, Peter Brixen og Anders Drejer Nygaard for gode rettelsler. Yderligere forslag, kommentarer, rettelsler og kritik er yderst velkommen.

Efter at have konstateret en uoverensstemmelse mellem den 'officielle' ideologi, som den er defineret i open source-licenser, og den faktiske adfærd hos hackere, undersøger vi de faktiske skikke, som regulerer ejerskabet af og kontrol af open source-software. Vi opdager, at de antyder en underliggende teori om ejendomsret, der stemmer overens med Lockes teori om ejendomsbesiddelse. Vi sætter det i forbindelse med en analyse af hackerkulturen som 'gavekultur', hvor deltagerne konkurrerer om prestige ved at give deres tid, energi og kreativitet væk. Vi undersøger så følgerne af denne analyse for løsning af konflikter i kulturen og peger på nogle hævdvundne slutninger.

1. En indledende uoverensstemmelse

Enhver, som bare et øjeblik betragter den travle, enormt produktive verden inden for Internet open source-software, kan ikke undgå at bemærke en interessant modsigelse mellem, hvad open source-hackere siger, de tror på, og den måde de faktisk opfører sig på — mellem den officielle ideologi i open source-kulturen og den aktuelle adfærd.

Kulturer er tilpasselige maskiner. Open source-kulturen er et svar på en række kendte motiver og påvirkninger. Som sædvanlig viser kulturens tilpasning til dens omstændigheder sig både som bevidst ideologi og som implicit, ubevidst eller delvist ubevidst viden. Og, som det ikke er ualmindeligt, er de ubevidste tilpasninger delvist i modsætning til den bevidste ideologi.

I dette skrift vil vi undersøge den dybereliggende årsag til modsætningen og bruge det til at opdage disse motiver og tryk. Vi udleder nogle interessante ting om hackerkulturen og dens skikke. Vi vil afslutte med at foreslå måder, som kan give kulturens implicitte viden større indflydelse.

[Hårde parenteser] henviser til bibliografien/slutnoterne.

2. Hackerideologiens mangfoldighed

Internet open source-kulturens ideologi (hvad hackere siger, de tror) er et temmelig komplekst emne i sig selv. Alle medlemmer er enige om, at open source (det vil sige software, der er frit kan redistribueres, og som let kan udvikles og modificeres til at tilpasse forandrede behov) er en god ting og værd betydelig og kollektiv arbejdsindsats. Denne enighed definerer effektivt medlemskab i kulturen. Men de årsager, som individer og forskellige subkulturer giver til denne tro, varierer betydeligt.

Den ene variant er fanatisme; "Fri software er mit liv"! Jeg lever for at skabe brugbare, smukke programmer og ressourcer til information og så give dem væk". En person med mere moderat iver kunne sige "Open source er en god ting, som jeg er villig til at bruge en betydelig mængde tid til at hjælpe med at blive til noget". En person med lidt iver kunne sige "Ja, open source er nogen gange OK. Jeg leger med det og respekterer folk, som laver det".

En anden variant er fjendtlighed overfor kommerciel software og/eller de selskaber, som opfattes som dominerende på det kommercielle software-marked.

En meget antikommerciel person kunne sige "Kommerciel software er tyveri og hamstring. Jeg skriver fri software for at stoppe dette onde". En moderat antikommerciel person kunne sige "Kommerciel software i almindelighed er OK, fordi programmører fortjener at blive betalt, men firmaer lever højt på tarvelige produkter, og som dominerer urimeligt, er onde". En person, der ikke er antikommerciel, kunne sige "Kommerciel software er OK, jeg bruger og/eller skriver bare open source-software, fordi jeg bedre synes om det" (i betragtning af hvor meget open source-delen af industrien er vokset siden den første offentlige version af dette dokument, kan man også nu om dage høre "Kommerciel software er fint, så længe jeg kan få kilden, eller det gør, hvad jeg vil have det til").

Alle de ni holdninger, som antydes af det samlede produkt af ovenstående kategorier, er repræsenteret i open source-kulturen. Det er værd at påpege forskellene, fordi de antyder forskellige dagsordener og forskellig adfærd med hensyn til tilpasning og samarbejde.

Historisk set har den mest synlige og bedst organiserede del af hacker-kulturen været både meget fanatisk og meget antikommerciel. The Free Software Foundation stiftet af Richard Stallman (RMS) støttede en stor del af open source-udviklingen fra starten af 1980'erne og frem, inklusive værktøjer som Emacs og GCC, som stadig er grundlæggende for open source-verdenen på Internet, og det virker sandsynligt, at de bliver ved med at være det inden for en overskuelig fremtid.

I mange år var FSF det mest vigtige fokuspunkt for open source-hacking og producerede et enormt antal værktøjer, som stadig er kritiske for kulturen. FSF var også længe den eneste open source-støtte, der havde en institutionel identitet, der var synlig for dem, der observerede hacker-kulturen. De definerede effektivt begrebet 'fri software', og gav bevidst begrebet en konfrontationel vægtning [fri er oversat fra det engelske free, der både kan betyde gratis og fri] (som det nyere begreb 'open source' lige så bevidst undgår).

Derfor var der både inden for og uden for hacker-kulturen tendens til at identificere kulturen med FSF's fanatiske holdning og den tilsyneladende antikommercielle målsætning (RMS selv benægter, at han er antikommerciel, men hans program er blevet læst sådan af de fleste folk, inklusive mange af hans mest højlydte partifanatikere). FSF's voldsomme og udtrykkelige forsøg på at "uddrive al software-hamstring!" blev det, der var tættest på en hacker-ideologi, og RMS var det nærmeste, man kom på en leder af hacker-kulturen.

FSF's licensbetingelser, "General Public License" (GPL) udtrykker FSF's holdninger. De er meget almindeligt brugt i open source-verdenen. North Carolinas Metalav (tidligere Sunsite) er det største og mest populære software-arkiv i Linux-verdenen. I juli 1997 anvendte cirka halvdelen Sunsite software-pakkerne, der havde udtrykkelige licensbetingelser, GPL.

Men FSF var aldrig det eneste sted, hvor der skete noget. Der var altid et mere stille, mindre konfrontationssøgende og mere markedsvenlig kreds i hackerkulturen. Pragmatikerne var loyale ikke så meget over for en ideologi men over for en gruppe af udviklingstraditioner, der stammede fra tidlige forsøg inden for open source, og som gik forud for FSF. Disse traditioner inkluderede vigtigst af alt de sammenvoksede kulturer Unix og Internet før kommercialiseringen.

Den typiske pragmatiske holdning er kun moderat antikommerciel, og den største indvending mod forretningsverdenen er ikke 'hamstring' som sådan. Snarere er det den verdensperverse nægtelse af at adoptere overlegne metoder ved at indarbejde Unix, åbne standarder og open source-software. Hvis pragmatikkerne hader noget, så er det sandsynligvis ikke 'hamstrene' i almindelighed snarere software-etablissemantets nuværende stivhed; tidligere hos IBM, nu hos Microsoft.

GPL er vigtig for pragmatikere som et værktøj snarere end et mål i sig selv. Den hovedværdi er ikke som våben mod 'hamstring', men som et værktøj til at opmuntre til deling af software og vækst i udviklingsmiljøer efter basarmetoden*. Pragmatikeren foretrækker gode værktøjer og godt legetøj mere, end han har noget imod kommercialisme, og han kan bruge kommerciel software uden ideologisk ubehag. Samtidig har hans oplevelser med open source demonstreret teknologisk høje standarder, som kun meget lidt software med lukket kildekode kan nå.

I mange år kom den pragmatiske holdning kun til udtryk i hackerkulturen som en stædig modstand mod at overgive sig fuldstændig til GPL i særdeleshed eller FSF's dagsorden i al almindelighed. Gennem 1980'erne og de tidlige 1990'ere var der tendens til, at denne holdning var tilknyttet

* Se <http://www.laisen.dk/Katedralen.og.basare.1064.0.html>.

tilhængere af Berkeley Unix, brugere af BSD-licensen og de tidlige forsøg på at bygge open source-Unixer fra BSD-kildebasen. Det lykkedes dog ikke at bygge basarmiljøer af nogen betydelig størrelse omkring disse forsøg, og de blev stærkt fragmenteret og ineffektive.

Ikke før Linux-eksplosionen i starten af 1993-1994 fik pragmatismen en reel magtbase. Selv om Linus Thorvalds aldrig gjorde et nummer ud af at gå imod RMS, satte han et eksempel ved at se positivt på fremvæksten af en kommerciel Linux-industri ved offentligt at støtte brugen af kommerciel software af høj kvalitet til særlige opgaver, og ved forsigtigt at håne de mere puritanske og fanatiske elementer i kulturen.

En følge af Linux' hurtige vækst var fremkomsten af en stor mængde hackere, der primært var loyale over for Linux, og som synes, at FSF's dagsorden primært var af historisk interesse. Selv om den nye generation af Linux-hackere kan finde på at beskrive systemet som "GNU-generationens valg", har de fleste tendens til at efterligne Torvalds snarere end Stallman.

I stigende grad var det puritanerne, der kom i mindretal. Hvor meget tingene havde ændret sig, blev ikke klart før udtalelsen fra Netscape om, at de ville distribuere Navigator 5.0 i kildeform. Det øgede interessen for 'fri software' inden for forretningsverdenen. Den efterfølgende opfordring til hacker-kulturen om at udnytte denne mulighed, der var uden fortilfælde, og omdøbe produktet fra 'fri software' til 'open source' blev mødt med en øjeblikkelig godkendelse, som overraskede alle indblandede.

I en genforstærkende udvikling var den pragmatiske del af kulturen i midten af 1990'erne ved selv at blive polycentrisk. Andre halvuaafhængige miljøer med egen selvbevidsthed og karismatiske ledere begyndte at blomstre frem fra Unix/Internet. Af disse var Perl-kulturen under Larry Wall den vigtigste efter Linux. Mindre men stadig betydelige var traditionerne, der blev bygget op omkring sprogene Tcl (John Osterhout) og Python (Guido van Rossum). Alle disse tre miljøer hævdede ideologiske uafhængighed ved at frembringe deres egen licensaftale, der ikke var GPL.

3. Promiskuøs teori, puritansk adfærd

På trods af alle disse forandringer blev der ved med at være en bred enighed om, hvad 'fri software' eller 'open source' er. Det klareste udtryk for denne almindelige enighed kan findes i de forskellige open source-licenser, som alle har vigtige fællestræk.

I 1997 blev disse fællestræk skrevet ind Debians retningslinier for fri software, som blev til definitionen af open source (OSD)*. Efter retningslinierne som defineret af OSD skal en open source-licens beskytte enhver ubetingede ret til at modificere (og redistribuere modificerede versioner) af open source-software.

Det udtrykkelige følge af OSD (licenser, der følger OSD så som GPL, BSD-licensen og Perl's kunstneriske licens er således, at enhver kan hacke hvad som helst. intet forhindrer et halvt dusin mennesker i at tage hvilket som helst open source-produkt (som fx. Free Software foundations' gcc C-compiler), kopiere kildekoden, udvikle den i forskellige retninger, og alle hævde, at de er *produktet*.

I praksis sker den slags forgrening ikke. Opsplitning inden for store projekter har været sjældne, og er altid fulgt af en omdøbning samt energiske forsøg på offentlige selvforklaring. Det er tydeligt, at i sådanne tilfælde som opsplitningen GNU Emacs/XEmacs, opsplitningen gcc/egcs eller de forskellige spaltninger i BSD-udbydergrupper har udbyderne følt, at de gik imod en temmelig magtfuld norm i miljøet [SP].

Faktisk (og i modsætning til enigheden om enhver-kan-hacke-hvad-som-helst) har open source-kulturen en udførlig men stort set uindrømmet sæt af skikke omkring ejerskab. Disse skikke regulerer hvem, der kan modificere software, omstændighederne, som det kan modificeres under, og især hvem, der har ret til at *redistribuere modificerede versioner* til miljøet.

En kulturs tabuer sætter dens normer i skarpt relief. Det vil derfor være nyttigt senere, hvis vi sammenfatter nogle af de vigtige her.

(a) Der er et stærkt social pres mod forgening af projekter. Det sker kun under påberåbelse af streng nødvendighed, med megen offentlige selvforklaring og med en omdøbning.

* Se <http://www.opensource.org/docs/definition.php>.

(b) Der ses ned på distribuering af ændringer uden at det sker i samarbejde med de toneangivende undtagen i særlige tilfælde, som i realiteten er trivielle porteringsrettelser.

(c) Man fjerner absolut ikke en persons navn fra projekthistorien, kildeangivelsen eller vedligeholdelseslisten uden personens udtrykkelige tilladelse.

I resten af dette skrift undersøger vi grundigt disse tabuer og skikke om ejerskab. Vi vil ikke bare undersøge, hvordan de fungerer, men også hvad de afslører om den underliggende sociale dynamik og tilskyndelsesstruktur i open source-miljøet.

4. Ejerskab og open source

Hvad betyder 'ejerskab', når ejendom kan replikeres i det uendelige, er stærkt modulerbar, og når den omgivende kultur hverken har magtforhold præget af tvang eller en økonomi præget materiel knaphed?

Faktisk er det et let spørgsmål at svare på inden for open source-kulturen. Ejerne af et software-projekt er dem, der har en eneret, der generelt er anerkendt i miljøet, til at redistribuere modificerede versioner.

(Når jeg diskuterer 'ejerskab' i denne del, vil jeg bruge ental, som om alle projekter er ejet af n person. Det er dog underforstået, at projekter kan være ejet af grupper. Vi vil undersøge den indre dynamik i sådanne grupper senere i dette skrift.)

I følge standardlicenserne inden for open source er alle parter i det evolutionære spil lige. Men i praksis er der en velanerkendt skelnen mellem 'officielle' fejlrettelser, som er godkendt og integreret i det udviklende software af offentligt anerkendte vedligeholdere, og uautoriserede fejlrettelser af tredieparter. Uautoriserede fejlrettelser er usædvanlige, og der stoles i almindelighed ikke på dem [RP].

Det er let at se, at *offentlig* redistribution er det, som det fundamentalt handler om. Sædvanen opfordrer folk til at fejlrette software til personlig brug, når det er nødvendigt. Sædvanen er ligeglad med folk, som redistribuerer modificerede versioner inden for en lukket bruger- eller udviklingsgruppe. Det er kun, når modifikationer frigives til open source-miljøet i almindelighed for at konkurrere med originalen, at ejerskab kommer til debat.

Der er i almindelighed tre måder at få ejerskab over et open source-projekt. En, den mest åbenlyse, er at stifte projektet. Når et projekt kun har haft n vedligeholder siden starten, og vedligeholderen stadig er aktiv, tillader sædvanen ikke engang at stille *spørgsmål* om hvem, der ejer projektet.

Den anden måde er at få ejerskab over et projekt overdraget til dig af den forrige ejer (dette kaldes nogle gange at 'aflevere stafetten'). Det er alment anerkendt i miljøet, at projektejere har en pligt til at overdrage projekter til kompetente efterfølgere, når de ikke længere selv er villige eller i stand til bruge den nødvendige tid til at arbejde med udvikling eller vedligeholdelse.

Det er betydningsfuldt, at sådanne overdragelser af kontrol, når der er tale om større projekter, sædvanligvis annonceres højtlydt. Selv om det er uhørt, at open source-miljøet som sådan faktisk blander sig i ejerens valg af efterfølger, så er der i almindelig praksis indarbejdet en præmis om, at offentlig legitimitet er vigtig.

I mindre projekter er det sædvanligvis tilstrækkeligt at skrive om ændringen af ejerskabet i projektdistributionens ændringshistorie. Den klare antagelse er, at hvis den tidligere ejer faktisk ikke frivilligt har overdraget kontrollen, kan han eller hun miljøets opbakning tage kontrollen tilbage ved at protestere offentligt inden for en rimelig tid.

Den tredje måde at få ejerskab over et projekt er at gøre opmærksom på, at det trænger til en indsats, og at ejeren er forsvundet eller har mistet interesse. Hvis du vil det, så er det dit ansvar at gøre en indsats for at finde ejeren. Hvis det ikke lykkes, så kan du annoncere på et relevant sted (som en Usenet-nyhedsgruppe inden for applikationens område), at projektet synes at være forældreløst, og at du overvejer at tage ansvar for det.

Sædvanen kræver, at du lader nogen tid gå, før du følger op med en meddelelse om, at du har erklæret dig selv for den nye ejer. Hvis nogen anden i mellemtiden meddeler, at de faktisk har arbejdet på projektet, så vil deres krav overgå dit. Der regnes for at være gode manerer at offentligt gøre opmærksom på dine intentioner mere end en gang. Der er flere point for god opførsel, hvis du

meddeler det i mange relevante fora (beslægtede nyhedsgrupper, postlister); og endnu flere point hvis du tålmodigt venter på svar. Generelt kan man sige, at jo større synlig indsats du gør for at lade den forrige ejer eller andre, der gør krav, svare, jo bedre er dit krav, hvis der ikke kommer svar.

Hvis du åbenlyst har gjort dette i miljøet for projektets brugere, og hvis der ikke er nogen indvendiger, så kan du hævde ejerskab over det forældreløse projekt og notere det i projektets historiefil. Det er dog mindre sikkert end at få overbragt stafetten, og du kan ikke forvente at blive betragtet som fuldt legitim, før du åbenlyst for miljøets brugere har udført betydelige forbedringer.

Jeg har betragtet disse skikke i funktion i 20 år tilbage fra tiden før FSF i open source-softwarens gamle historie. Skikkene har adskillige interessante egenskaber. En af de mest interessante er, at de fleste hackere har fulgt dem uden at være helt klar over det. Faktisk er det ovenstående måske første bevidste og nogenlunde fuldstændige sammenfatning, der nogensinde er skrevet ned.

En anden er, at de af ubevidste skikke at være er blevet fuldt med bemærkelsesværdig (endda forbavsende) konsekvens. Jeg har bogstavelig talt fulgt hundredevis af open source-projekter, og jeg kan stadig påtø hænder tælle antallet af de betydelige overtrædelser, jeg har set eller hørt om.

Endnu en tredje interessant egenskab er, at efterhånden som disse skikke har udviklet sig over tid, og har de gjort det i en konsekvent retning. Den retning har været at opmuntre til mere offentlig ståen til regnskab, længere offentlig varsel og mere omhyggelighed med at bevare bidragerlisten og ændringshistorien til et projekt på en sådan måde, at det (blandt andet) etablerer de nuværende ejeres legitimitet.

Disse egenskaber antyder, at skikkene ikke er tilfældige, men er produkter af en implicit dagsorden eller frugtbar mønster i open source-kulturen, som er fuldstændig grundlæggende for den måde, kulturen fungerer på.

En tidlig læser gjorde opmærksom på, at det belyser det frugtbare mønster ganske godt at sætte Internet hacker-kulturen op over for cracker-/piratkulturen ("warez d00dz" centreret omkring cracke spil og pirat bulletin-board systems. Vi vender tilbage til kontrasten d00dz senere i skriftet.

5. Locke og adkomst til jord

For at forstå dette frugtbare mønster hjælper det at se en historisk analogi til disse skikke, som ligger langt uden for det domæne, som hackere sædvanligvis interesserer sig for. Som interesserede i rethistorie og politisk filosofi måske vil genkende, er teorien om ejendom, som de antyder, praktisk talt identisk med den anglo-amerikanske sædvaner om adkomst til jord!

I denne teori er der tre måde at erhverve ejerskab over jord.

Nær en grænse, hvor der findes land, som aldrig har haft en ejer, kan man erhverve ejerskab ved *nybyggeri*, det vil sige ved at begynde at arbejde med eller på den ejerløse jord, at sætte hegn omkring den og ved at forsvare den.

Den sædvanlige måde at overdrage land på i bosatte områder er *overdragelse af adkomst*, det vil sige at modtage skødet fra den forrige ejer. I denne teori er begrebet 'adkomstkæden' vigtig. Det ideale bevis på ejerskab er en kæde af skøder og overdragelser, der går tilbage til den gang, jorden oprindeligt blev nybygget.

Endelig anerkender teorien om sædvaner, at adkomst til jord kan mistes eller efterlades (for eksempel hvis ejeren dør uden arvinger, eller hvis de nødvendige optegnelser for at kunne bevise en adkomstkæde til efterladt land er væk). Et stykke jord, der er blevet forladt på denne måde, kan blive krævet af andre med henvisning til *manglende ejerskab* — man flytter ind, forbedrer den og forsvarer adkomsten, som om man var nybygger.

Denne teori er, på samme måde som hackerens skikke, udviklet organisk i en sammenhæng, hvor statsmagten var svag eller ikke eksisterende. Den udvikledes gennem en periode på tusind år ud af oldnordisk og germansk stammelov. Eftersom den blev systematiseret og rationaliseret i den tidlige moderne tidsalder af den engelske politiske filosof John Locke, kaldes den nogle gange 'Lockes teori om ejendom'.

Der har været tendens til, at teorier med en lignende indre logik er udviklet overalt, hvor ejendom har stor værdi eller betydning for overlevelse, og hvor ingen enkelt autoritet er magtfuld nok til at fremtvinge en central allokering af knappe ressourcer. Det gælder endda i jæger/samlerkulturer,

som romantisk ofte regnes for at være uden et 'ejendomsbegreb'. For eksempel hos !Kung San-buskmændene i Kgalagadiørkenen (tidligere 'Khalahari') er der traditionelt ikke nogen der ejer jagtområderne. Men der *er* nogen der ejer vandhuller og kilder i følge en teori, som er genkendeligt beslægtet med Lockes.

!Kung San-eksemplet er lærerigt, da det viser, at skikke med hensyn til ejendom, der ligner Lockes, opstår, hvor den forventede gevinst ved ressourcen overstiger den forventede omkostning ved at forsvare den. Jagtområder er ikke ejendom, fordi gevinsten ved jagt varierer, er stærkt uforudsigelig, og (selv om den er højt værdsat) ikke er nødvendig for den daglige overlevelse. Vandhuller derimod er vitale for overlevelsen og små nok til at forsvare.

'Noosfæren' i dette skrifs titel er ideernes territorium. Mængden af alle mulige tanker [N]. Det, vi ser antydnet i hackerens skikke om ejerskab, er en teori ligesom Lockes om ejendomsret i en delmængde af noosfæren, netop delmængden af alle programmer. Derfor 'nybyggeri i noosfæren', som er det, alle stiftere af et nyt open source-projekt gør.

Far Rideau [fare@tunes.org] påpeger korrekt, at hackere ikke just bevæger sig i de rene ideers territorium. Han hævder, at det, som hackere ejer, er *programmeringsprojekter* — bevidste fokuspunkter for materielt arbejde (udvikling, service, etc), som er påhæftet sådan noget som anseelse, pålidelighed, etc. Han hævder derfor, at den mængde, hackerprojekter er en del af, *ikke* er noosfæren men en fællesmængde, nemlig mængden af noosfæredforskende programmeringsprojekter (med et undskyldende nik til astrofysikerne derude, vil det være etymologisk korrekt at kalde denne fællesmængde for 'ergosfæren' eller 'arbejdssfæren').

I praksis er sondringen mellem noosfæren og ergosfæren ikke vigtig for formålet med dette skrift. Det er tvivlsomt, om 'noosfæren' i den rene forstand, som Far insisterer på, kan hævdes at kunne eksistere på nogen meningsfuld måde; man er næsten nødt til at være platonisk filosof for at tro på det. Og sondringen mellem noosfæren og ergosfæren er kun af praktisk betydning, hvis man ønsker at påstå, at ideer (noosfærens elementer) ikke kan ejes, mens deres instantieringer som projekter kan. Dette spørgsmål leder til spørgsmål med hensyn til teorien om intellektuel ejendom, som falder uden for dette skrift (men se [DF]).

For at undgå forvirring er det dog vigtigt at bemærke, at hverken noosfæren eller ergosfæren er det samme som totaliteten af virtuelle steder i de elektroniske medier, som some tider (til de fleste hackers væmmelse) kaldes 'cyberspace'. Der er ejendom reguleret gennem helt andre regler, der ligger tættere på dem om det grundlæggende materielle — i bund og grund er ejeren af en del af 'cyberspace' den person, som ejer mediet og maskinerne, på hvilke den del af 'cyberspace' ligger på.

Skikkens lockeske logik antyder stærkt, at open source-hackere overholder de skikke, de gør, for at forsvare en eller anden gevinst ved deres indsats. Gevinsten må være mere betydelig end indsatsen ved nybyggerprojekter: omkostningerne ved at vedligeholde versionshistorier, der dokumenterer 'adkomstkæden', omkostningerne i tid ved at give offentlige bekendtgørelser om en overtagelse af et projekt, og venteperioden, før man endelig kan overtage selve det forældreløse projekt på grund af manglende ejerskab.

Desuden må 'udbyttet' fra open source være noget mere end bare brugen af softwaren, noget andet som ville blive kompromiteret eller udvandet af forgrening. Hvis brugen var det eneste problem, ville der ikke være noget tabu mod forgrening, og ejerskab af open source ville ikke ligne besiddelse af jord. Faktisk er denne alternative verden (hvor brug er det eneste udbytte, og hvor forgrening er uproblematisk) den, der antydes af de eksisterende open source-licenser.

Vi kan eliminere nogle slags udbytte som kandidater med det samme. Da man ikke effektivt kan bruge tvang gennem en netværksforbindelse, er søgen efter magt helt udelukket. Ligeledes har open source-kulturen ikke meget, der ligner penge eller en indre knaphedsøkonomi, såhackere kan ikke stræbe mod noget, der minder særlig meget om materiel rigdom (fx. akkumulering af symboler på knaphed).

Der er dog en måde, hvormed aktivitet inden for open source kan hjælpe folk med at blive rigere — en måde, som giver et værdifuldt tegn på hvad, der virkelig motiverer dem. Ind imellem kan den anseelse, man får i hacker-kulturen, sive over i den virkelige verden på en måde, der har økonomisk betydning. Den kan skaffe dig et bedre jobtilbud, en konsulentaftale eller en bogkontrakt.

Den slags sidegevinst er dog for det meste sjælden og marginal for de fleste hackere; for sjælden

og marginal til at gøre det overbevisende som den eneste forklaring, selv hvis vi ignorerer de gentagne protester fra hackere om, at de ikke gør det for penge men af idealisme eller kærlighed.

Det er værd at undersøge måden sådanne sidegevinster formidles. Nedenfor vil vi se, at en forståelse af anseelsens dynamik inden for open source-kulturen selv har betragtelig forklarende evne.

6. Hackermiljøet som gavekultur

For at forstå den rolle anseelse spiller i open source-kulturen, er det en hjælp at flytte fra historie og længere ind i antropologien og økonomien og undersøge forskellen mellem *byttekulturer* og *gavekulturer*.

Mennesker har en medfødt trang til at konkurrere om social status; det er et resultat af vores evolutionshistorie. I 90% af den tid, der låfør opfindelsen af landbrug, levede vores forfædre i små nomadiske flokke af jæger/samlere. Individuer med høj status (de mest effektive til at underrette koalitioner og overtale andre til at samarbejde med dem) fik de sundeste mager og adgang til den bedste mad. Denne trang til status kommer til udtryk på forskellige måder stort set afhængigt af graden af knaphed på nødvendige varer.

De fleste måder mennesker organiserer sig på er tilpasninger til knaphed og behov. Hver måde indebærer forskellige måder at erhverve social status.

Den enkleste måde er *magthierakiet*. I magthierakier fordeles varer af en central autoritet, der styrer med magt. Magthierakier skalerer meget dårligt [Mal]; de bliver stadigt mere brutale og ineffective, efterhånden som de bliver større. Af denne grund er magthierakier større end en udvidet familie næsten altid parasiter på en større økonomi af en anden slags. I magthierakier er social status primært bestemt af adgang til tvangsmagt.

Vort samfund har primært en *bytteøkonomi*. Det er sofistikeret tilpasning til knaphed, som i modsætning til magtmodellen skalerer ganske godt. Fordeling af knappe varer sker decentralt gennem handel og frivillig samarbejde (og faktisk er den dominerende effekt ved konkurrerende behov at skabe samarbejdende adfærd). I en bytteøkonomi er social status primært bestemt af at have kontrol over ting (ikke nødvendigvis materielle ting) til at bruge eller handle med.

De fleste har forudindtagede mentale forestillinger om begge ovenstående, og hvordan de påvirker hinanden. Regeringen, militæret og organiseret kriminalitet (for eksempel) er magthierakier, der snylter på den bredere bytteøkonomi, som vi kalder 'det frie marked'. Der er dog en tredje model, der er radikalt forskellig fra dem begge og ikke generelt anerkendt undtagen af antropologer; *gavekulturen*.

Gavekulturer er en tilpasning ikke til knaphed men til overflod. De opstår i befolkninger, der ikke har betydelige materielle knaphedsproblemer med nødvendige varer. Vi ser gavekulturer i funktion i opindelige kulturer, der lever i økozoner med mildt klima og en overflod af mad. Vi kan også se dem i bestemte lag i vores eget samfund, især inden for showbusiness og blandt de meget rige.

Overflod gør magtforhold vanskelige at opretholde og bytteforhold til et næsten meningsløst spil. I gavekulturer er social status bestemt, ikke af hvad du har kontrol over, men *hvad du giver væk*.

Således Kwakiutl-høvdingens potlachfest. Således multimillionærens kunstfærdige og som regel offentlige filantropiske handlinger. Og således hackerens timelange indsats for at skabe open source-kode af høj kvalitet.

Når det ses på denne måde, er det helt klart, at miljøet af open source-hackere faktisk er en gavekultur. I det er der ingen alvorlig mangel på 'overlevelsesnødvendige' — diskplads, netværksbåndbredde, computerkraft. Software deles frit. Denne overflod skaber en situation, hvor det eneste tilgængelige mål for konkurrerende succes er anseelse blandt ens ligemænd.

Denne observation er dog ikke i sig selv helt tilstrækkelig til at forklare de observerede træk ved hacker-kulturen. Crackerne og warez d00dz har en gavekultur, som trives i det samme (elektroniske) medie som hackernes, men deres adfærd er meget anderledes. Gruppementaliteten i deres kultur er meget stærkere og mere eksklusiv end blandt hackere. De hamstrer hellere hemmeligheder end deler dem; det er mere sandsynligt, at man kan finde crackergrupper, der distribuerer programmer

uden kilde, der cracker software, end tips, som afslører, hvordan de gjorde det. (Se [LW] for et indre perspektiv på denne adfærd).

Det viser, hvis det ikke var åbenbart, at der er mere end en måde at køre en gavekultur. Historie og værdier har betydning. Jeg har opsummeret hacker-kulturens historie andetsteds i [HH]; de måder, som historien formede den nuværende adfærd, er ikke mystiske. Hackere har defineret deres kultur gennem et sæt af valg om den *form*, som deres konkurrence vil tage. Det er den form, som vi vil undersøge i resten af dette skrift.

7. Glæden ved at hacke

Det var forresten ikke min mening med denne analyse af 'anseelsespillet' at nedvurdere eller ignorere den rent kunstneriske tilfredsstillelse ved at designe smuk software og få det til at virke. Vi oplever alle den slags tilfredsstillelse og trives med den. Hvis det ikke er en betydelig motivation, bliver man overhovedet ikke hacker, ligesom folk, der ikke elsker musik, aldrig bliver komponister. Så måske skulle vi overveje endnu en model for hacker-adfærd, hvor den rene glæde ved håndværksmæssig dygtighed er den primære motivation. Denne 'håndværksmodel' vil skulle forklare hacker-skikke som en måde at maksimere både mulighederne for håndværk og kvaliteten af resultaterne. Er det i konflikt med eller antyder det andre resultater end modellen med 'anseelsespillet'?

Faktisk ikke. Når vi undersøger 'håndværksmodellen', vender vi tilbage til de samme problemer, som begrænser hacker-kulturen til at fungere som en gavekultur. Hvordan maksimerer man kvalitet, hvis der ikke er en målestok for kvalitet? Hvis der ikke er en knaphedsøkonomi, hvilken målestok findes der så udover vurdering blandt ligemænd? Det virker som om, alle håndværkskulturer i sidste ende må strukturere sig selv gennem et anseelsespil — og vi kan faktisk se nøjagtig denne dynamik i mange historiske håndværkskulturer fra middelalderens lav og frem.

På et vigtigt punkt er 'håndværksmodellen' svagere end modellen med en 'gavekultur'; i sig selv hjælper det ikke med at forklare den modsigelse, som vi indledte dette skrift med.

Endelig er 'håndværksmotivationen' måske ikke i sig selv psykologisk så langt væk fra anseelsespillet, som man måske gerne ville formode. Forestil dig dit smukke program låst nede i en skuffe og aldrig brugt igen. Forestil dig nu, at det bruges effektivt og med glæde af mange mennesker. Hvilken drøm giver dig tilfredsstillelse?

Ikke desto mindre vil vi holde øje med håndværksmodellen. Den er intuitivt tiltalende for mange hackere og forklarer nogle aspekter af individuel adfærd til strækkeligt [HT].

Efter jeg offentliggjorde den første version af dette skrift, kommenterede en anonym læser: "Du arbejder måske ikke for at fåanseelse, men anseelsen er en reel betaling med konsekvenser, hvis du gør jobbet godt". Dette er en skarpsindig og vigtig pointe. Den ansporende anseelse fortsætter at have betydning, uanset om en håndværker er klar over det; i sidste ende vil en hackers adfærd således, uanset om han forstår sin adfærd i lyset af spillet om anseelse, være formet af det spil.

Andre læsere satte anseelse blandt ligemænd og glæden ved at hacke i forbindelse med de øvre lag over subsistensbehovene i Abraham Maslovs velkendte 'behovspyramide' som model for menneskelig motivation [MH]. I følge dette synspunkt er glæden ved at hacke et selvaktualiserings- eller transcendensbehov, der ikke vedholdende vil blive udtrykt, før behovene fra de lavere lag (inklusive dem om fysisk sikkerhed og for 'tilhør' eller agtelse blandt ligemænd) er i det mindste minimalt tilfredsstillt. Anseelsespillet er således måske kritisk for skabe en social kontekst, hvor glæden ved at hacke faktisk kan *blive* individets primære motiv.

8. Anseelsens mange ansigter

Der er grunde, der generelt gælder i enhver gavekultur, til, at anseelse (prestige) blandt ligemænd er værd at betale for:

For det første og mest åbenlyst er anseelse blandt ens ligemænd en primær gevinst. Vi er tvunget til at opleve det på den måde på grund af de tidligere nævnte evolutionære grunde. (Mange mennesker lærer at omdirigere deres behov for prestige til forskellige sublimeringer, der ikke har nogen tydelig forbindelse til en synlig gruppe af ligemænd såsom "ære", "etisk integritet", "fromhed" etc.; dette ændre ikke på de underliggende mekanismer.)

For det andet er prestige en god måde (og i en ren gavekultur den eneste måde) at tiltrække opmærksomhed og samarbejde fra andre. Hvis man er kendt for generøsitet, intelligens, fair handlen, ledelsesevner eller andre gode kvaliteter, bliver det meget lettere at overbevise andre om, at de vil have fordel af tilknytning til n.

For det tredje hvis din gaveøkonomi er i kontakt med, sammenblandet med bytteøkonomi eller et magthieraki, kan dit rygte sive derover og skaffe dig højere status der.

Udover disse almene grunde gør de besynderlige forhold i hacker-kulturen gøre prestige endnu mere værdifuld, end den vil være i en 'gavekultur' i den virkelige verden.

Det 'besynderlige forhold', der er vigtigst, er, at det er meget komplekse artefakter, som man giver væk (eller tolket på en anden måde er energi og tid de synlige tegn på ens bidrag). Deres værdi er ikke engang lige så tydelige, som materielle gaver eller bytteøkonomiens penge er. Det er meget vanskeligere at skelne objektivt mellem en god gave og en dårlig gave. Derfor er successen af en givers ønske om status nøje afhængig af ligemænds kritiske vurdering.

En anden besynderlighed er open source-kulturens relative renhed. De fleste gavekulturer er kompromiteret — enten af forhold fra bytteøkonomien som handel med luksusvarer eller forhold fra magthierakiet som familie- eller klangrupperinger. Der findes ingen betydelige analogier hertil i open source-kulturen; der er således stort set ingen anden måde at få status på end gennem anseelse blandt ligemænd.

9. Ejendomsret og tilskyndelse til anseelse

Vi er nu i stand til at sammenfatte de forrige analyser til en sammenhænge beskrivelse af hackeres skikke om ejerskab. Vi forstår nu gevinsten ved nybyggeri i noosfæren; det er omdømme blandt ligemænd i hackeres gavekultur med alle de sekundær gevinster og bieffekter, som det indebærer.

Ud fra den forståelse kan vi analysere hacker-kulturens lockeske ejerskabsskikke som en metode til at *maksimere tilskyndelse til anseelse*; til at sikre, at anerkendelsen blandt ligemænd tilfalder den, som fortjener det, og ikke den, som ikke fortjener det.

De tre tabuer, vi bemærkede ovenfor, giver god fornuft i denne analyse. Ens anseelse kan lide uretfærdigt, hvis nogen misbruger eller ødelægger ens arbejde; disse tabuer (og beslægtede skikke) forsøger at forhindre det i at ske. (Eller for at sige det mere pragmatisk undlader hackere som regel at forgrene eller lave uautoriserede fejlrettelser til andres projekter for at være i stand til at benægte legitimiteten af denne adfærd, når de selv bliver udsat for den.)

(a) Forgrening af projekter er uheldig, fordi det udsætter bidragsydere fra før forgreningen for en risiko dårligt omdømme, som de kun kan kontrollere ved at være aktiv i begge delprojekter samtidig efter forgreningen. (Dette vil sædvanligvis være for forvirrende eller svært til at være praktisk.)

(b) At distribuere uautoriserede fejlrettelser (eller meget værre uautoriserede binære filer) udsætter ejerne for en uretfærdig risiko for deres anseelse. Selv hvis den officielle kode er perfect vil ejerne blive ramt af kritik som følge af fejl i fejlrettelserne (men se [RP]).

(c) En af de ultimative forbrydelser i en kulturel kontekst er at fjerne en eller andens navn fra et projekt. Det er tyveri af offerets gave for at fremstille den som tyvens egen.

Selvfølgelig er en forgrening eller en uautoriseret fejlrettelse også direkte angreb på den oprindelige udviklergruppes omdømme. Hvis jeg forgrener eller laver uautoriserede fejlrettelser til dit projekt, såsiger jeg: ”du tog en forkert beslutning (ved ikke at føre projektet i den retning, som jeg gør)”; og Enhver, som bruger min forgrenede variant støtter den udfordring. Men det i sig selv vil være en fair udfordring omend ekstrem; det er den skarpeste ende af ligemænds vurdering. Det er derfor ikke tilstrækkeligt til i sig selv at gøre rede for de tabuer, selv om det uden tvivl bidrager til at styrke dem.

Alle disse tre tabubelagte former for adfærd medfører såvel global skade på open source-miljøet såvel som lokal skade påofre(ne). Implicit skader det hele miljøet ved at nedsætte hver enkelt potentiel bidragsydners opfattede sandsynlighed for, at gave-/produktiv adfærd vil blive belønnet.

Det er vigtigt at bemærke, at der er alternative kandidater til forklaring af to af disse tre tabuer.

For det første forklarer hackere ofte deres antipati mod forgrening af projekter ved at klage over spildet ved duplikering af arbejdet, som følger af, at delprodukterne udvikles mere eller mindre

parallelt i fremtiden. De vil måske også bemærke, at forgrening betyder en opsplitning af medudviklarmiljøet, der efterlader begge delprojekter med færre hjerner at arbejde med end hovedprojektet.

En læser har gjort opmærksom på, at det er usædvanligt, at mere end et af forgreningens afkom overlever på lang sigt med en betydelig 'markedsandel'. Dette styrker tilskyndelsen til, at alle parter samarbejder og undgår forking, fordi det er svært at vide på forhånd, hvem der vil være på den tabende side, dermed se deres arbejde forsvinde fuldstændigt eller sygne hen i ubemærkethed.

Modvilje mod uautoriserede fejlrettelser forklares ofte ved at Bemærke, at de kan ofte komplicere fejlfinding enormt og medføre mere arbejde for vedligeholderne, som har rigeligt at gøre med at finde deres egne fejl.

Der er betydelig sandhed i disse forklaringer, og de kan med sikkerhed betyde en del for at forstærke ejerskabets lockeske logik. Men selv om de er intellektuelt tiltrækkende, forklarer de ikke, hvorfor så meget følelse og territorialitet bliver udvist i de få tilfælde, hvor tabuerne bliver bøjet eller brudt — ikke bare fra de forurettede parter men fra folk på sidelinien og betragtere, som ofte reagerer temmelig barskt. Nøgttern bekymring om dobbelt arbejde og besvær med vedligeholdelse forklarer simpelthen ikke den observerede adfærd særlig godt.

Desuden er der det tredje tabu. Det er svært at se, hvordan andet end analysen af anseelsespillet kan forklare det. Det faktum, at dette tabu sjældent analyseres meget mere dybtgående end "det ville ikke være fair", er afslørende i sig selv, som vi skal se i næste del.

10. Problemet med ego

I begyndelsen af dette skrift nævnte jeg, at det ubevidst tilpassede kendskab til en kultur ofte står i modsætning til dens bevidste ideologi. Vi har allerede set et hovedeksempel på dette i det faktum, at lockeske skikke om ejerskab har været fulgt almindelighed på trods af det faktum, at de strider mod standardlicensernes fastsatte hensigt.

Jeg har set et andet interessant eksempel på dette fænomen, når jeg diskuterede analysen af anseelsespillet med hackere. Det er, at mange hackere modsatte sig analysen og udviste en stærk modvilje mod at indrømme, at deres adfærd var motiveret af et ønske om anseelse blandt ligemænd eller det, som jeg dengang døbte 'egotilfredsstillelse'.

Dette illustrerer en interessant pointe om hacker-kulturen. Den mistror og foragter bevidst egoisme og egobaseret motivation; der er tendens til, at selvpromovering kritiseres skånselsløst, selv når miljøet synes at kunne drage fordel af det. Faktisk så meget, at det kræves af kulturens 'store mænd' og stammeældste, at de taler lavt, og at de hele tiden med humor forklejner sig selv for at beholde deres status. Hvordan denne holdning harmonerer med en tilskyndelsesstruktur, som tilsyneladende næsten udelukkende styres af ego, råber til himlen efter en forklaring.

En stor del af det stammer sikkert fra den generelle negative euro-amerikanske holdning til 'ego'. De fleste hackers kulturelle matrix lærer dem, at det er dårlig (eller i det mindste umodent) at motiveres af et ønske om tilfredsstillelse af ego; at ego i bedste tilfælde er en ekscentricitet, som kun kan tolereres hos primadonnaer, og ofte er et egentligt tegn på psykopatologi. Kun sublimerede og skjulte former som "anseelse blandt ligemænd", "selvopfattelse", "professionalisme" eller "stolthed over det opnåede" er alment accepterede.

Jeg kunne skrive et helt andet essay om de usunde rødder til denne del af vores kulturelle arv, og den forbavsende mængde af selvvidledende skade, vi udøver ved at tro (på trods af psykologiens og adfærdens beviser), at vi nogensinde har ægte 'uselviske' motiver. Måske ville jeg, hvis ikke Friedrich Wilhelm Nietzsche og Ayn Rand allerede har gjort et ganske kompetent indsats (uanset deres andre svagheder) med at dekonstruere 'altruismen' til ikke anerkendte form for egoisme.

Men jeg forsøger ikke at udøve moralfilosofi eller psykologi her, så jeg vil bare gøre opmærksom på en mindre form for skade som følge af holdningen, at ego er forkert, nemlig denne: den har gjort det følelsesmæssigt svært for mange hackere at bevidst forstå den sociale dynamik i deres egen kultur!

Men vi er ikke helt færdige med denne undersøgelsesretning. Den omgivende kulturs tabu mod synlig egostyret adfærd er så meget forstærket i hacker-(sub)kulturen, at man må mistænke det for at have en slags speciel tilpasselig funktion for hackere. I hvert fald er tabuet svagere (eller ikke eksisterende) i mange andre gavekulturer såsom teaterfolks eller de meget riges kultur for ligemænd!

11. Værdien af ydmyghed

Efter at have konstateret, at prestige som belønningsmekanisme er centralt i hacker-kulturen, har vi nu behov for at forstå, hvorfor det har virket så vigtigt, at dette faktum forblev halv-fordækt og stort set uindrømmet.

Kontrasten til piratkulturen er lærerig. I den kultur er søgen efter status åbenlys og endda højrystet. Disse crackere søger hyldest for at frigive "zero-day warez" (cracked software, der frigives på samme dag, som den originale ucrackede version frigives), men de fortæller ikke om, hvordan de gør det. Disse magikere kan ikke lide at forære deres tricks væk. Og som resultat vokser videnbasen i cracker-kulturen som helhed kun langsomt.

I modsætning hertil udtrykker man sig ved sit arbejde i hacker-miljøet. Der er et meget stærkt meritokrati (det bedste håndværk vinder), og der er et etos om, at kvalitet bør (endda skal) tale for sig selv. Det bedste pral er kode, der "bare virker", og hvor enhver kompetent programmør kan se, at koden er rigtig god. På den måde stiger hacker-kulturens videnbase hurtigt.

Tabuet mod egodrevet attitude øger derfor produktiviteten. Men det er en følgeeffekt: det er kvaliteten af miljøets information - systemet med vurdering blandt ligemænd - der direkte beskyttes her. Det vil sige, at pral eller selvoptagethed undertrykkes, fordi det er støj, der forstyrrer eksperimenteres vigtige signaler om kreative og samarbejdende adfærd.

Af samme grunde angriber man simpelthen ikke forfatteren i stedet for koden. Der er en interessant detalje her, som forstærker pointen; det er tilladt for hackere at kaste sig over hinanden af ideologiske og personlige årsager, men det er uhørt for en hacker offentligt at angribe en andens kompetence med hensyn til teknisk arbejde (selv privat kritik er usædvanlig og har tendens til at blive hvisket. Fejlfinding og kritik er altid møntet projektet og ikke personen.

Desuden bruges fortidige fejl ikke automatisk mod en udvikler; det faktum, at en fejl er blevet rettet, anses som regel for vigtigere end det faktum, at der har været en. Som en læser bemærkede, kan man fåstatus ved at rette 'Emacs-fejl' men ikke ved at rette 'Richard Stallman-fejl' — og det ville anses for ekstremt dårlige manerer at kritisere Stallman for gamle Emacs-fejl, der siden er blevet rettet.

Det er en interessant modsætning til mange steder i den akademiske verden, hvor nedgøring af andres formodet mangelfulde arbejde er en vigtig måde at få anseelse på. I hacker-kulturen er der et temmelig stærkt tabu mod sådan adfærd — faktisk så stærk, at fraværet af en sådan adfærd ikke gik op for mig, før en læser med et usædvanligt perspektiv gjorde opmærksom på det næsten et helt år efter, at dette skrift først var udgivet!

Tabuet mod angreb på kompetence (der ikke deles i den akademiske verden) er endnu mere afslørende end tabuet mod at føre sig frem, der deles med den akademiske verden, fordi vi kan relatere det til en forskel mellem den akademiske verden og hacker-kulturen i deres kommunikation og supportstrukturer.

Hacker-kulturens gavemetode er uåndgribelig, dens kommunikationskanaler er dårlige til at udtrykke emotionelle nuancer, og kontakt ansigt til ansigt er undtagelsen snarere end reglen. Det giver en lavere tolerance for støj end i de fleste andre gavekulturer, og giver en god forklaring på både kravet om offentlig ydmyghed og tabuet mod angreb på kompetence. Enhvert større angreb på hackeres kompetence kan gøre utålelig skade på kulturens pointsystem.

Den samme sårbarhed overfor støj forklarer det eksempel i offentlig ydmyghed, som hacker-kulturens stammeældste forventes at sætte. De måses at være fri for pral og føren sig frem, så tabuet mod farlig støj opretholdes. [DC]

Det er også en god id at tale stille og roligt, hvis man har ambitioner om at vedligeholde et succesfuldt projekt; man måoverbevise miljøet om, at man besidder god dømmekraft, da det meste af en vedligeholders arbejde vil være at bedømme andres kode. Hvem vil have lyst til bidrage med arbejde til en, som tydeligvis ikke kan bedømme kvaliteten af deres egen kode, eller hvis adfærd antyder, at de vil forsøge uden berettigelse at rage den anseelse til sig, som kommer fra projektet? Potentielle bidragere ønsker projektledere med tilstrækkelig ydmyghed og klasse til at være i stand til, når det objektivt set er passende, at sige "Ja, det virker bedre end min version, så jeg vil bruge det" — og give anerkendelse til den som fortjener det.

Det er endnu en grund til ydmyg adfærd, at man i open source-verdenen sjældent vil give det

indtryk, at et projekt er 'færdigt'. Dette kan føre til, at en potentiel bidrager ikke føler sig nødvendig. Man maksimerer sin indflydelse ved at være ydmyg med hensyn til programmets tilstand. Hvis man praler gennem sin kode og så siger: "Det er ikke en disse værd, det kan ikke gøre x, y og z, så det kan ikke være så godt", så vil fejlrettelser til x, y og z ofte følge hurtigt.

Endvidere har jeg personligt set, at den selvforklejnende adfærd hos nogle ledende hackere afspejler en reel (og ikke uberettiget) frygt for blive objekt for en personlighedskult. Linus Torvalds og Larry Wall leverer begge klare og talrige eksempler på sådan undgåelsesadfærd. En gang da jeg var på vej ud at spise på ved ud at spise med Larry Wall, sagde jeg for sjov "du er alpha-hackeren her — du får lov til at vælge restauranten". Han krympede sig hørligt. Og med rette; det har ødelagt mange miljøer ikke at adskille fælles værdi fra ledernes personligheder, et mønster, som han og Linus ikke kan undgå at være fuldt opmærksom på. På den anden side vil de fleste hackere elske at have Larrys problem, hvis de kunne få sig selv til at indrømme det.

12. Globale følger af modellen af anselesspillet

Analysen af anselesspillet har nogle flere implikationer, som måske ikke er umiddelbart indlysende. Mange af disse stammer fra det faktum, at man får mere prestige ved at stifte et succesfuldt projekt end ved samarbejde indenfor et eksisterende. Man får også mere prestige ved projekter, som er slående innovative i modsætning til mindre 'ogsåmig' forbedringer af software, som allerede eksisterer. På den anden side er software, som ingen anden end forfatteren forstår eller har behov for, ikke en god start i anselesspillet, og det er ofte lettere at tiltrække god opmærksomhed ved at bidrage til et eksisterende projekt, end det er at få folk til at lægge mærke til et nyt. Endelig er det meget sværere at konkurrere med et allerede succesfuldt projekt, end det er udfylde en tom niche.

Der er således en optimal afstand fra ens naboer (de mest lignende projekter). For tæt på og ens projekt er et "også mig!" af begrænset værdi, en dårlig gave (man ville være bedre tjent med at bidrage til et eksisterende projekt). For langt væk og ingen vil være i stand til at bruge eller forstå relevansen af ens indsats (igen en dårlig gave). Det skaber et mønster af nybyggeri i noosfæren, som temmelig meget ligner mønstret af bosættere, der spreder sig i et fysisk grænseområde — ikke tilfældigt men som en diffusions-hæmmet fraktal. Projekter har tendens til at blive startet for at udfylde funktionelle huller nær grænsen (se [NO] for en yderligere diskussion af det tiltrækkende ved det nye).

Nogle meget succesfulde projekter bliver 'kategoridræbere'; ingen vil nybygge i nærheden af dem, fordi det vil være for vanskeligt at konkurrere mod det etablerede grundlag om hackerens opmærksomhed. Folk, som måske ellers ville starte deres egne særskilte forsøg, ender i stedet for med at lave tilføjelser til disse store succesfulde projekter. Det klassiske eksempel på en 'kategoridræber' er GNU Emacs; dets varianter fylder den økologiske niche for en fuldt programmerbar editor så fuldt ud, at ingen konkurrent er kommet meget længere end et-mandsstadiet siden de tidlige 1980'ere. I stedet skriver folk funktioner til Emacs.

Globalt har disse to tendenser (udfyldelse af huller og kategoridræbere) skabt en i det store hele forudsigelig trend for start af projekter over tid. I 1970'erne var det meste af det open source, som fandtes, legetøj og demoer. I 1980'erne var fremdriften inden for udvikling og Internetværktøjer. I 1990'erne skiftede aktiviteten til operativsystemer. I hvert tilfælde blev et nyt og mere vanskeligt niveau af problemer angrebet, da mulighederne inden for det forrige område næsten var udtømt.

Denne trend har interessante implikationer for den nære fremtid. I starten af 1998 så Linux ud til at være en kategoridræber inden for nichen 'open source-operativsystemer' — folk, der måske ellers ville have skrevet konkurrerende OS'er, skriver nu i stedet for udstyrsdrivere og udvidelser. Og de fleste af de almindelige værktøjer, som kulturen nogen sinde har forstillet sig som open source, findes allerede. Hvad er der tilbage?

Applikationer. Som år 2000 nærmer sig, synes det risikofrit at forudsige, at udviklingsindsatsen inden for open source i stigende grad vil skifte mod det sidste uberørte område — programmer for de ikke-tekniske. En klar og tidlig indikator er udviklingen af GIMP*, det Photoshop-lignende

* <http://www.gimp.org/>

billedprogram, som er open sources første store applikation med den slags slutbrugervenlig grafisk brugergrænseflade, der betragtes som de rigueur inden for kommercielle applikationer i de sidste tiår. En anden er den opmærksomhed, der omgiver projekter som KDE* og GNOME**, der er værktøjssætsapplikationer.

En læser af dette skrift har gjort opmærksom på, at analogien med nybyggere også forklarer, hvorfor hackere reagerer med sådan fysisk vrede mod Microsofts politik om at ”omfavne og udvide” internetprotokollerne og dermed gøre dem mere komplekse og lukket. Hacker-kulturen kan sameksistere med det meste lukkede software; eksistensen af Adobe Photoshop for eksempel gør ikke territoriet omkring GIMP (dets open source-pendant) betragteligt mindre attraktivt. Men når det lykkes Microsoft at afkommercialisere en protokol [HD], så kun Microsofts egne programmører kan skrive software til det, skader de ikke bare kunderne ved at udvide deres monopol. De reducerer også mængden og kvaliteten af noosfære, der er tilgængelig for hackere til nybyggeri og opdyrkning. Det er ingen overraskelse, at hackere ofte kalder Microsofts strategi for ”protokolforurening”; de reagerer nøjagtig som farmere, der ser nogen forgifte floden, som de vander deres afgrøder med!

Endelig forklarer analysen af anselesspillet det ofte citerede udsagn, at man ikke bliver hacker ved at kalde sig hacker — man bliver hacker, når *andre hackere* kalder n hacker. En ’hacker’ set i det lys er en, der har vist (ved at bidrage med gaver), at han eller hun både har teknisk evne og forstår, hvordan anselesspillet fungerer. Denne vurdering handler for det meste om opmærksomhed og kulturindlæring, og den kan kun blive givet af dem, som allerede er godt inde i kulturen.

13. Hvor god en gave?

Der er et konsistent mønster i den måde, som hacker-kulturen værdsætter bidrag og giver agtelse blandt ligemænd tilbage for dem. Det er svært ikke at bemærke følgende regler:

1. *Hvis det ikke virker så godt, som man har ladet mig forvente, duer det ikke — ligemeget hvor smart og originalt, det er.*

Læg mærke til ’ladet mig forvente’. Denne regel er ikke et krav om perfektion; en beta og eksperimental software må gerne have fejl. Det er et krav om, at brugeren har mulighed for præcis at estimere risikoen ved projektets stadie og udviklerens fremstilling af det.

Denne regel ligger til grund for det faktum, at open source-software har tendens til at være i beta i lang tid og ikke engang får et 1.0 versionsnummer, før udviklerne er meget sikre på, at der ikke vil komme en masse grimme overraskelser. I closed source-verdenen betyder version 1.0 ”Hvis du er smart, lader du det ligge.”; i open source-verdenen betyder det snarere ”Udviklerne er villige til at sætte deres gode navn på spil med det.”

2. *Arbejde, der udvider noosfæren, er bedre end arbejde, der kopierer funktionalitet i eksisterende software.*

Den naive måde at sige det på ville have været: *Originalt arbejde er bedre end at kopiere eksisterende softwares funktioner.* Men så simpelt er det faktisk ikke helt. Det tæller lige så højt at kopiere eksisterende *lukket* softwares funktioner som originalt arbejde, især hvis man åbner en lukket protokol eller et lukket format og gør det territorium klart.

Til eksempel er et af de projekter med mest prestige i den nuværende open source-verden således Samba — den kode, som tillader Unix-maskiner at fungere som klienter eller servere for Microsofts proprietære SMB-protokol til fildeling. Der er meget lidt kreativt arbejde, der skal gøres her; det er mest et spørgsmål om at få udført reverse engineering-detalerne korrekt. Ikke desto mindre opfattes Samba-gruppen som helte, fordi de neutraliserede et forsøg fra Microsoft på at fastlåse hele grupper af brugere og afspærre en stor del af noosfæren.

3. *Arbejde, der kommer med i en stor distribution, er bedre end arbejde, der ikke gør det. Arbejde, der findes i alle store distributioner, er mest prestigefyldt.*

* <http://www.kde.org/>

** <http://www.gnome.org/>

De store distributioner er ikke kun de store Linux-distributioner som Red Hat, Debian, Caldera og S.u.S.E., men også andre samlinger, der anses for at have deres egen anseelse at opretholde og således implicit er garant for kvalitet — som BSD-distributioner eller Free Software Foundations samling af kildekoder.

4. Brug er den mest oprigtige form for smiger — og kategoridræbere er bedre end kopiprogrammer.

Det er grundlæggende for processen med gennemgang af ligemænd at stole på andre bedømmelse. Det er nødvendigt, fordi ingen har tid til at gennemse alle mulige alternativer. Så arbejde, der bruges af mange mennesker, betragtes som bedre arbejde end arbejde, der bruges af få.

Når arbejde gøres så godt, at ingen længere interesserer sig for at bruge alternativer mere, har man opnået en enorm prestige. Den størst mulige prestige kommer ved at have lavet arbejde, der er udbredt og populært, en kategoridræber og originalt, og som findes i alle større distributioner. Folk, som har opnået det, mere end en gang, kaldes delvist i alvor for 'halvguder'.

5. Fortsat hengivelse til hårdt, kedeligt arbejde (som at fejlfinde eller skrive dokumentation) er mere prisværdigt end kun at beskæftige sig med det morsomme og de lette hacks .

Denne norm handler om, hvordan miljøet belønner nødvendige opgaver, som hackere ikke af sig selv er tilbøjelig at tage sig af. Det modsiges i nogen grad af:

6. Udvidelser af funktioner, der ikke er trivielle, er bedre end mindre fejlrettelser og fejlfinding.

Det synes at virke på den måde, at det sandsynligvis giver større belønning i et enkeltstående tilfælde at tilføje en ny funktion end at rette en fejl — med mindre fejlen er exceptionelt ubehagelig eller obskur, så det at klare den i sig selv er en demonstration af usædvanlige evner og kløgt. Men udtrækkes den slags adfærd over tid, vil en person med en lang historie af at tage sig af og rette selv ordinære fejl, kan meget vel rangere højere end n, der har udøvet en tilsvarende indsats med at tilføje lette funktioner.

En læser har gjort opmærksom på, at disse regler spiller sammen på interessante måder og ikke nødvendigvis altid giver den størst mulige nytte. Spørg en hacker, om der er større sandsynlighed for, at han bliver mere kendt for et helt nyt og eget værktøj eller for udvidelser til en andens, og der vil ikke være tvivl om svaret "nyt værktøj". Men spørg om

(a) et helt nyt værktøj, som kun bruges usynligt af OS'et et par gange om dagen, men som hurtigt bliver en kategoridræber

mod

(b) flere udvidelse til et eksisterende værktøj, som hverken er særligt nyt eller kategoridræbere, men som bruges dagligt og dagligt synligt for en vældig stor gruppe brugere

og du vil sandsynligvis opleve nogen tøven, før hackeren bestemmer sig for (a). Disse alternativer er ligeligt fordelt.

Nævnte læser stillede dette spørgsmål til mig ved at tilføje "Eksempel (a) er fetchmail; eksempel (b) er dine mange udvidelser af Emacs som vc.el og gud.el." Og han har faktisk ret; jeg bliver sandsynligvis bedre kendt som 'forfatteren af fetchmail' end som 'forfatter af en masse Emacs-funktioner', endda selv om sidstnævnte sandsynligvis har haft en større total nyttevirkning over tid.

Det, som måske foregår her, er ganske enkelt, at arbejde med et nyt 'identitetsmærke' får mere opmærksomhed end arbejde, der lægges til et eksisterende 'mærke'. Yderligere undersøgelse til belysning af disse regler, og hvad de fortæller os om hacker-kulturens scoringsystem, ville være godt.

14. Noosfærisk ejendom og territoriets etologi

For at forstå årsager til og konsekvenserne af lockeske ejendomsskikke, vil det hjælpe os at se på dem fra en endnu en vinkel; den fra dyreetologi, specielt territoriets etologi.

Ejendom er en abstraktion af dyrenes territorialitet, som udvikledes til en måde at reducere vold inden for arten. Ved at markere sit territorium og respektere andre grænser mindsker en ulv sine chancer for at komme i slagsmål, der kunde gøre den svagere, dræbe den eller gøre den mindre reproduktiv. På samme måde har ejendom den funktion i menneskelige samfund at forhindre konflikter mellem mennesker ved at sætte grænser, som klart adskiller fredelig adfærd fra aggression.

Det er på mode i nogle cirkler at beskrive menneskelig ejendom som en tilfældig social konvention, men det er helt forkert. Enhver, der nogensinde har ejet en hund, der gøede, når fremmede kom nær dens ejers ejendom, har oplevet grundlæggende sammenhæng mellem dyreterritorialitet og menneskelig ejendom. Vores tamme fætter til ulven ved instinktivt, at ejendom ikke blot er en social konvention eller et spil, men en mekanisme af kritisk betydning udviklet til at udgå vold. (Det gør dem klogere end rigtig mange politiske teoretikere.)

Det er en udførende handling at gøre krav på ejendom (ligesom at markere et territorium), en måde at gøre opmærksom på hvilke grænser, der vil blive forsvaret. Støtte fra miljøet til kravet om ejendom er en måde at minimere friktion og maksimere samarbejdende adfærd. Det er stadig sandt, selv når "ejendomskravet" er meget mere abstrakt end et hegn eller en hunds gøen, selv når det bare er en udtalelse i form af projektvedligeholderens navn i en README-fil. Det er stadig en abstraktion af territorialitet, og (som andre former for ejendom) baseret på territoriale instinkter, der er udviklet til at hjælpe med konfliktløsning.

Denne etologiske analyse vil måske først virke meget abstrakt og svær at relatere til faktisk hacker-adfærd. Men den har nogle vigtige konsekvenser. En er at forklare populariteten af Internethjemmesider, og især hvorfor open source-projekter med hjemmesider virker så meget mere 'virkelige' og indholdsrige end dem uden.

Objektivt set virker det svært af forklare. Sammenlignet med indsatsen ved at finde på og vedligeholde selv et lille program er en hjemmeside let, så det er svært at lade en hjemmeside være tegn på substans eller usædvanlig indsats.

Ej heller er de funktionelle karakteristika ved Internet i sig selv tilstrækkelig forklaring. Kommunikationsfunktionen ved hjemmesider kan ligeså godt eller bedre laves med en kombination af en FTP-side, en postliste og posteringer på Usenet. Faktisk er det temmelig ualmindeligt, at et projekts rutinekommunikation foregår på Internet frem for på en postliste eller i en nyhedsgruppe. Hvorfor er hjemmesider så populære som hjemsted for projekter?

Metaforen i begrebet 'hjemmeside' giver et vigtigt spor. Selvom det at stifte et open source-projekt er et territorielt krav i noosfæren (og sædvanligvis anerkendt som sådan), er det ikke et særligt fængende krav på det psykologiske plan. Software har, når det kommer til stykket, ikke nogen naturlig placering, og det kan umiddelbart kopieres igen og igen med det samme. Det kan bringes til at ligne vores instinktive opfattelse af 'territorie' og 'ejendom' men først efter nogen indsats.

En projekthjemmeside konkretiserer et abstrakt nybyggeri i de mulige programmets rum ved at udtrykke det som territorielt 'hjem' i Internets mere rumligt organiserede verden. Vi når endnu ikke helt til den rigtige verdens hegn og gøende hunde ved at stige ned fra noosfæren til cyberspace, men det knytter det abstrakte ejendomskrav tættere til vores territoriale instinkt. Og det er derfor, projekter med hjemmesider virker mere 'virkelige'.

Denne pointe styrkes meget af hyperlinks og eksistensen af gode søgemaskiner. Et projekt med en hjemmeside vil med større sandsynlighed blive bemærket af nogen, der udforsker området i noosfæren; andre vil linke til det, og de, der søger, vil finde det. En hjemmeside er derfor en bedre reklame, en mere effektiv udførende handling, et stærkere territorielt krav.

Denne etologiske analyse opfordrer os også til at se nærmere på mekanismerne til at håndtere konflikter i open source-kulturen. Den lader os forvente, at udover at maksimere tilskyndelse til anseelse, bør ejendomsskikke også have en rolle med hensyn til at forhindre og løse konflikter.

15. Årsager til konflikt

Vi kan identificere fire større årsager til konflikt indenfor open source:

- (a) Hvem får lov til at tage de bindende beslutninger om et projekt?
- (b) Hvem for ære eller skyld for hvad?
- (c) Hvordan undgås det at fejlfinding medfører dobbeltarbejde og uautoriserede versioner?
- (d) Hvad er Det Rigtige, teknisk set?

Men hvis vi en gang til ser på spørgsmålet om "Hvad er Det Rigtige", så har det tendens til at forsvinde. For alle den slags spørgsmål er der enten en objektiv måde at afgøre det på for alle parter, eller også er der det ikke. Hvis der er, så er det afgjort og alle vinder. Hvis der ikke er det, reduceres det til "hvem bestemmer?".

Derfor er de tre problemer, som en teori om konfliktløsning skal afklare (A) hvem tager de endelige beslutninger om design, (B) hvordan bestemmes, hvilke bidragere skal krediteres og hvordan og (C) hvordan afholder man et projekt og et produkt fra at blive opdelt i mange grene.

Ejendomsskikkens rolle med hensyn til at løse spørgsmål (A) og (C) er klar. Skikkene bekræfter, at ejerne af projektet tager de bindende beslutninger. Vi har før set, at skikkene også over stort pres mod udtynding af ejerskab ved forgrening.

Det er lærerigt at bemærke, at disse skikke er fornuftige, selv hvis man glemmer spillet om anseelse og undersøger dem inde fra en ren 'håndværksmodel' af hacker-kulturen. I den vinkel har disse skikke mindre at gøre med udtynding af tilskyldelse til anseelse end med at beskytte en håndværkers ret til at udføre sin vision på den måde, han vælger.

Men håndværksmodellen er ikke tilstrækkelig til at forklare hackernes skikke omkring spørgsmål (B), hvem der får ære for hvad (fordi en ren håndværker, som ikke interesserer sig for spillet om anseelse, ikke ville have noget motiv til at bekymre sig. For at analysere disse er vi nødt til at følge den lockes teori et skridt videre og undersøge konflikter, og hvordan ejendomsret fungerer både *indenfor* projekter og *mellem* dem.

16. Projektstrukturer og ejerskab

Det trivielle tilfælde er det, hvor projektet har en enkelt ejer/vedligeholder. I det tilfælde er der ingen mulig konflikt. Ejeren tager alle beslutninger og får al ære og skyld. Den eneste mulige konflikt er over overdragelsesspørgsmål — hvem bliver den nye ejer, hvis den gamle forsvinder eller mister interessen. Miljøet har også en interesse under spørgsmål (C) i at forhindre forgrening. Disse interesser er udtrykt ved en kulturel norm, at en ejer/vedligeholder offentligt bør overgive ejerskabet til en anden, hvis han eller hun ikke længere kan vedligeholde projektet.

Det enkleste ikke-trivielle tilfælde er, når et projekt har mange medvedligeholdere, der arbejder under en enkelt 'venligsinde diktator', der ejer projektet. Sædvanen favoriserer den måde for gruppeprojekter; det har vist sig at virke på projekter så store som Linux-kernen eller Emacs, og løser problemet "hvem bestemmer" på en måde, som ikke er åbenlyst værre, end nogen af af alternativerne.

Typisk udvikler en organisation omkring en venligsinde diktator sig fra en organisation omkring en ejer-vedligeholder, efterhånden som stifteren tiltrækker bidragere. Selv hvis ejeren bliver ved med at være diktator, introducerer det nye niveauer af mulige uenigheder over hvem, der får ære for hvilke dele af projektet.

I denne situation giver sædvanen ejeren/diktatoren en forpligtelse til at give anerkendelse retfærdigt (gennem for eksempel passende nævnelse i README eller historiefiler). I henhold til den lockeske ejendomsmodel betyder det, at man får en del af æren ved at bidrage til et projekt (positiv eller negativ).

I følge denne logik ser vi, at en 'venligsinde diktator' faktisk ikke ejer sit projekt uden at være kvalificeret. Selv om han har ret til at tage bindende beslutninger, bytter han i realiteten andele af den totale anseelse væk for andres arbejde. Analogien med bortforpagtning af en farm er næsten uimodståelig, med undtagelse af at en bidragers navn bliver ved med stå i kreditlisten og fortsætter med at 'få' nogen anseelse, selv efter at bidragsyderen ikke længere er aktiv.

Efterhånden som projekter med venligsindede diktatorer får flere deltagere, har de tendens til at udvikle to lag af bidragere; almindelige bidragere og medudviklere. En typisk måde at blive medudvikler på er at tage ansvar for et større delområde af projektet. En anden er at påtage sig

rollen som 'den store fixer', beskrive og rette mange fejl. På den eller andre måder er medudviklere de bidragsydere, som yder en betydelig og vedvarende investering af tid til projektet.

Delområdets ejers rolle er særlig vigtig for vores analyse og fortjener yderligere undersøgelse. Hackere holder af at sige, at 'autoritet følger ansvarlighed'. En medudvikler, der accepterer et vedligeholdelsesansvar for et givet delsystem, får normalt lov til at bestemme over både implementeringen af det delsystem og dets grænseflade til resten af projektet kun under hensyntagen til korrektioner af projektlederen (der fungerer som arkitekt). Vi bemærker, at denne regel effektivt medfører indeholdte egenskaber efter den lockeske model indenfor projektet, og har nøjagtig den samme rolle til undgåelse af konflikter som andre ejendomsgrænser.

Ifølge sædvanen forventes 'diktatoren' eller projektlederen i et projekt at konsultere medudviklerne om vigtige beslutninger. Det gælder især, hvis beslutningen vedrører et delområde, som en medudvikler 'ejer' (det vil sige, har investeret tid i og taget ansvar for). En klog leder, der anerkender den funktion, som projektets interne ejendomsgrænser har, vil ikke ubekymret blande sig i eller omgøre beslutninger, som er taget af delsystemets ejere.

Nogle meget store projekter forkaster fuldstændigt modellen med en 'venligsinde diktator'. En måde at gøre det på er at lave en stemmekomitee (som i tilfældet Apache). En anden er et diktatur, der går på omgang, hvor kontrollen til tider afleveres fra et medlem til et andet inden for en kreds af seniorudviklere (Perl-udviklerne organiserer sig på denne måde).

Sådanne komplicerede arrangementer anses i almindelighed for at være ustabile og vanskelige. Denne opfattede vanskelighed er klart en funktion af de kendte farer ved design i komiteer og ved komiteer i det hele taget; dette er problemer, som hacker-kulturen bevidst forstår. Men jeg tror, at nogle af det fysisk ubehag, som hackere føler ved organisationer med komiteer eller formandskaber, der går på omgang, skyldes, at de er svære at få til at passe ind i den ubevidste lockeske model, som hackere bruger til at resonere med i de enkle tilfælde. Det er problematisk i disse komplekse organisationer at holde rede på ejerskab i form af kontrol eller på ejerskab i form af anseelse. Det er svært at se, hvor de indre grænser går, og således svært at undgå konflikt med mindre gruppen har et ekseptionelt højt niveau af harmoni og tillid.

17. Konflikt og konfliktløsning

Vi har set, at rollernes øgede kompleksitet inden for projekter viser sig ved en distribution af designautoritet og delvis ejendomsret. Selvom det er en effektiv måde at distribuere tilskyndelse, udvander det også projektlederens autoritet — vigtigst af alt udvander det lederens autoritet til at stoppe potentielle konflikter.

Selvom tekniske uenigheder om design måske synes at være den mest åbenlyse risiko for gensidigt ødelæggende konflikt, er de sjældent en alvorlig årsag til strid. Disse er som regel relativt let løste ved hjælp af den territoriale regel om, at autoritet følger ansvarlighed.

En anden måde at løse konflikter på er ved senioritet — hvis to bidragsydere eller grupper af bidragsydere er uenige, og hvis uenigheden ikke kan løses objektivt, og ingen ejer territoriet, der er uenige om, vinder den side, som har lagt mest arbejde i projektet som helhed (det vil sige, den side med størst ejendomsret i hele projektet).

(På samme måde taber den side med mindst på spil. Det er interessant, at dette faktisk er den samme heuristik, som mange relationelle databaser løser fastlåste situationer på. Når to tråde er fastlåste over ressourcer bliver den side med mindst på spil i den aktuelle transaktion offer for den fastlåste situation og lukkes ned. Det bliver sædvanligvis den længst kørende transaktion eller den, der er den ældste, der er vinderen.)

Disse regler er normalt tilstrækkelige til at løse de fleste projektuenigheder. Når de ikke er det, er projektlederens ordre tilstrækkelig. Uenigheder, der overlever begge disse filtre, er sjældne.

Konflikter er som regel ikke altid alvorlige med mindre disse to kriterier ("autoritet følger ansvarlighed" og "senioritet vinder") peger i hver sin retning, og projektlederens autoritet er svag eller fraværende. Det mest åbenlyse tilfælde, hvor det kan ske, er en uenighed om overtagelse efter, at projektlederen er forsvundet. Jeg har været i et slagsmål af den slags. Det var grimt, smertefuldt, langvarigt og blev kun løst, da alle parter var tilstrækkeligt udmattede til at overlade kontrollen til en udenforstående, og jeg håber oprigtigt, at jeg aldrig kommer i nærheden af den slags igen.

I sidste ende hviler alle disse mekanismer til konfliktløsning på det brede hacker-miljøes villighed til at håndhæve dem. De eneste mulige mekanismer til håndhævelse er ved at falde over dem og udstøde dem — offentlig fordømmelse af dem, som bryder skikkene, og vægring ved at samarbejde med dem, efter at de har gjort det.

18. Mekanismer til kulturindlæring og forbindelsen til den akademiske verden

En tidlig udgave af dette skrift stillede følgende forskningsspørgsmål: Hvordan informerer og lærer miljøet dets medlemmer miljøets skikke? Er skikkene selvindlysende eller selvorganiserende på et halvbevidst niveau, læres de gennem eksempler, læres de gennem eksplicit instruktion?

Læring gennem eksplicit instruktion er klart sjældent, ikke mindst fordi hidtil har kun få eksplicitte beskrivelser af kulturens normer eksisteret.

Mange normer læres gennem eksempel. For at nævne et meget enkelt tilfælde, er der en norm om, at al software skal have en fil kaldet README eller README, som indeholder indledende instruktioner om at orientere sig i distributionen. Denne konvention var været veletableret siden de tidlige 1980'ere; den har endda til tider været skrevet ned. Men man tilegner sig den normalt ved at have set den i mange distributioner.

På den anden side er nogle hacker-skikke selvorganiserende, når man har erhvervet en grundlæggende (måske ubevidst) forståelse for anseelsespillet. De fleste hackere behøver aldrig at blive undervist i de tre tabuer, som jeg beskrev i afsnit tre, eller de vil i det mindste hævde, at de er selvindlysende snarere end videregivet. Dette fænomen inviterer til nærmere analyse — og måske kan vi finde dets forklaring i den proces, som hackere erhverver viden om kulturen gennem.

Mange kulturer bruger skjulte spor (mere præcist 'mysterier' i den religiøse/mystiske betydning) som en mekanisme til kulturindlæring. Det er hemmeligheder, som ikke afsløres til udenforstående, men som det forventes, at den håbefulde nybegynder opdager eller resonerer sig til. For at blive accepteret i kredsen må man demonstrere, at man både forstår mysteriet og har lært det på en kulturelt acceptabel måde.

Hacker-kulturen gør usædvanlig bevidst og udbredt brug af sådanne spor eller tests. Vi kan se, at denne proces fungerer på mindst tre niveauer:

(a) Mysterier som adgangskontrol. Som eksempel er der en USENET-nyhedsgruppe, der hedder alt.sysadmin.recovery, der har en sådan, meget eksplicit hemmelighed; du kan ikke sende post til nyhedsgruppen uden at kende den, og at kende den betragtes som bevis på, at du er berettiget til at sende post til nyhedsgruppen. De faste har et stærkt tabu mod at afsløre denne hemmelighed.

(b) Nødvendigheden af optagelse inden for bestemte tekniske mysterier. Man må lære en stor mængde teknisk viden, før man kan give gaver, der er noget værd (dvs man må kende mindst et af de store computersprog). Dette krav fungerer i det store, ligesom skjulte spor fungerer i det små, som et kvalitetsfilter (såsom evne til abstrakt tænkning, vedholdenhed og mental fleksibilitet), som er nødvendige for at fungere i kulturen.

(c) Mysterier i social sammenhæng. Man bliver involveret i kulturen ved at knytte sig til specifikke projekter. Hvert projekt er en levende social sammenhæng af hackere, som den kommende bidrager er nødt til at undersøge og forstå socialt så vel som teknisk for at kunne fungere. (Konkret er en almindelig måde at gøre det på at læse projektets hjemmeside og/eller emailarkiver) Det er gennem disse projektgrupper, som de nye oplever eksempler på de erfarne hackers adfærd.

Mens man lærer disse mysterier, samler den, som vil være hacker, viden om baggrunden, som (efter et stykke tid) får de tre tabuer og andre skikke til at virke 'selvindlysende'.

Man kunne forresten hævde, at strukturen i hackerens gavekultur i sig selv er det centrale mysterie. Du betragtes ikke oplært i kulturen (konkret: ingen vil kalde dig for hacker), før du demonstrerer dyb forståelse for spillet om anseelse og de skikke, tabuer og den brug, som følger med. Men dette er trivielt; alle kulturer forlanger sådan en forståelse af kommende medlemmer. Desuden viser hackerkulturen intet ønske om at have dets indre logik og adfærd holdt hemmelig — eller i det mindste har ingen nogensinde faldet over mig for at afsløre dem!

Flere læsere af dette skrift, end jeg kan nævne, har gjort opmærksom på, at hackerskikke om ejerskab synes nært knyttet til (og stammer måske direkte fra) adfærden i den akademiske verden, især det videnskabelige forskningsmiljø. Dette forskningsmiljø har lignende problemer med at

udvinde et territorie af potentielt produktive ideer og udviser meget lignende tilpassede løsninger til de problemer med den måde, miljøet bruger vurdering blandt ligemænd og anseelse.

Da mange hackere har haft en dannende påvirkning fra den akademiske verden (det er almindeligt at lære at hacke, mens man går på universitetet), er det af mere end almindelig interesse at forstå i hvor høj grad den akademiske verden deler adaptive mønstre med hacker-kulturen, og hvordan disse skikke anvendes.

Den akademiske versen er rig på åbenlyse paralleller til 'hacker-gavekulturen', som jeg har karakteriseret den. Når en forsker opnår et embede, er der ingen grund til at bekymre sig om overlevelse (Begrebet embede kan sikkert spores tilbage til en tidligere gavekultur, hvor "naturfilosoferne" primært var velhavende herrer med overskud af tid, der kunne helliges forskning.) Når der ikke er problemer med overlevelse bliver øgning af anseelse den primære drivkraft, hvilket opfordrer til deling af nye ideer og forskning i tidsskrifter og andre medier. Det er objektivt set funktionelt, fordi videnskabelig forskning ligesom hacker-kulturen er stærkt afhængig af ideen om 'at stå skuldrene af de store' og af ikke at skulle genopdage basale principper igen og igen.

Nogle er gået så langt at foreslå, at hacker-skikke bare er en afspejling af forskningsmiljøets adfærd, og de enkelte hackere har faktisk (i de fleste tilfælde) lært skikkene der. Det er sikkert at overdrive tilfældet, om ikke andet så fordi, hacker-skikkene synes at blive lært let af intelligente gymnasieelever!

19. Gaver udkonkurrerer handel

Der er en mere interessant mulighed her. Jeg vil tro, at den akademiske verden og hacker-kulturen deler adaptive mønstre, ikke fordi de er genetisk beslægtede, men fordi de begge er udviklet af den mest optimale sociale organisering af det, som de prøver at udføre, givet naturlovene og menneskers instinktive indretning. Historiens dom synes at være, at kapitalismens frie marked er den globalt mest optimale måde at samarbejde på for at generere (og sikre!) kreativt arbejde af høj kvalitet.

Et stort antal psykologiske undersøgelser af samspillet mellem kunst og belønning støtter denne teori [GNU]. Disse undersøgelser har fået mindre opmærksomhed end de burde, delvist fordi dem, der har gjort opmærksom på dem, har haft en tendens til at overfortolke dem som generelle angreb på det frie marked og intellektuel ejendomsret. Ikke desto mindre antyder deres resultater, at nogle former for belønning i en mangeløkonomi nedsætter produktiviteten hos kreative arbejdere såsom programmører.

Psykologen Theresa Amabile fra Brandeis University, der forsigtigt har opsummeret resultaterne af en undersøgelse fra 1984 af motivation og belønning, bemærker "Det kan være at bestilt arbejde i almindelighed vil være mindre kreativt end arbejde, der udføres ud fra ren lyst. Amabile bemærker endvidere, at "Jo mere kompleks aktiviteten er, jo mere skades den af udefrakommende belønning." Det er interessant, at undersøgelserne antyder, at almindelig løn ikke er demotiverende, mens akkordarbejde og bonusordninger gør.

Det kan således være økonomisk fordelagtigt at give en præstationsafhængig bonus til folk, der vender burgere eller graver grøfter, men det er formentlig klogere ikke at sammenkoble aflønning og ydelse i en programmeringsafdeling og lade people selv vælge deres egne projekter (begge trends tager Open source-verdenen til deres logiske konsekvens). Faktisk antyder disse resultater, at det eneste tidspunkt, hvor det er en god id at belønne ydelse indenfor programmering, er, når programmøren er så motiveret, at han eller hun ville have arbejdet uden belønningen.

Andre forskere inden for området er villige til at pege direkte på områder som atonomy og kontrol over den kreative proces, som hackere går så meget op i. "I lige så høj grad som ens oplevelse af at være selvbestemmende er begrænset," siger Richard Ryan, viceprofessor ved universitetet i Rochester, "vil ens kreativitet også blive begrænset."

Generelt synes det at præsentere en hvilken som helst opgave som et middel snarere end som et mål i sig selv at være demotiverende. Endda det at vinde en konkurrence eller at få anseelse blandt ligemænd kan være demotiverende på denne måde, hvis det opfattes som arbejde for en belønning (hvilket kan forklare, hvorfor hackere kulturelt set er forhindret i udtrykkeligt at søge eller kræve denne anseelse).

For at gøre managementproblemet endnu mere kompliceret synes mundtlig feedback at være lige så demotiverende som akkordarbejde. Ryan fandt, at virksomhedsansatte, som blev fortalt, ”Godt, du gør som du skal” var ”signifikant mindre reelt motiveret end dem, som fik feedback i form af informationer.”

Det kan stadig være smart at tilbyde tilskyndelser, men de skal komme uden bindinger, så de ikke tilklitrer arbejdet. Der er en kritisk forskel (bemærker Ryan) mellem at sige: ”Jeg giver dig denne belønning, fordi jeg anerkender værdien af dit arbejde” og på at sige: ”Du får denne belønning, fordi du lever op til mine forventninger.” Den første udtalelse demotiverer ikke; det gør den anden.

På disse psykologiske observationer kan vi bygge en påstand om, at Open Source-udviklingsgrupper vil være betydeligt mere produktive (især i det lange løb, hvor kreativitet bliver en mere kritisk faktor for produktivitet) end en gruppe af closed source-programmører med samme størrelse og evner, men som bliver demotiveret af mangeløkonomiens belønninger.

Den antyder fra en lidt anden vinkel en af overvejelserne fra Katedralen og basaren; at i sidste ende var den industrielle, kapitalistiske måde at producere software på dømt til at blive udkonkurreret fra det øjeblik, kapitalismen begyndte at skabe en tilstrækkelig overflod af rigdom til, at mange programmører kunne leve i den gavekultur, der kommer efter knapheden.

Faktisk virker det som den opskrift på den højeste softwareproduktivitet næsten er et Zen-paradoks; hvis du vil have den højeste produktivitet, skal du holde op med at forsøge at få programmører til at producere. Sørg for deres eksistens, giv dem forstand og glem alt om tidsfrister. For den almindelige leder, lyder det ekstremt eftergivende og dømt til at fejle — men det er nøjagtig den opskrift, som open source-kulturen bruger til at dele kløud til dens konkurrenter.

20. Konklusion: Fra skik til sædvanelov

Vi har undersøgt skikkene, som styrer ejerskabet af og kontrollen over open source-software. Vi har set, hvordan de antyder en underliggende teori om ejendom, som er i overensstemmelse med Lockes teori om besiddelse af jord. Vi har relateret det til en analyse af hacker-kulturen som ’gavekultur’, hvor deltagerne konkurrerer om prestige ved at give tid, energi og kreativitet væk. Vi har undersøgt betydningen af denne analyse for konfliktløsning i kulturen.

Det næste logiske spørgsmål er at spørge, ”Hvorfor er det interessant?” Hackerne udviklede disse skikke uden bevidst analyse og har (indtil nu) fulgt dem uden bevidst analyse. Det er ikke umiddelbart klart, at vi har fået noget praktisk ud af den bevidste analyse — med mindre vi måske kan bevæge os fra beskrivelse til forskrift og udlede måder at forbedre den måde, skikkene fungerer på.

Vi har fundet en tæt logisk analogi til hacker-skikke i teorien om besiddelse af jord i følge sædvaneloven i den anglo-amerikanske traditionen. Historisk set [Miller] har de europæiske stammekulturer, som opfandt denne tradition, forbedret deres systemer til løsning af uoverensstemmelse ved at bevæge sig fra et system af uartikulerede halvbevidste skikke til et system med en eksplicit sædvanelov memoreret af stammens vismænd — og til sidst skrevet ned.

Måske er det ved at være tid til, efterhånden som vores befolkning vokser og kulturindlæringen af alle nye medlemmer bliver mere vanskelig, at gøre noget lignende — at udvikle skrevne regler for gode måder at løse de forskellige slags uoverensstemmelser, der kan opstå i forbindelse med open source-projekter, og en tradition med voldgift, hvor miljøets seniormedlemmer kan blive bedt om at mægle i uoverensstemmelser.

Analysen i dette skrift antyder omridset af hvordan, sådanne regler kunne se ud, idet den gør det eksplicit, som tidligere var implicit. Ingen sådanne regler kunne påtvunges oppefra; de skulle nødvendigvis adopteres frivilligt af stifterne eller ejerne af individuelle projekter. De måtte heller ikke være fuldstændigt stive, da påvirkningen af kulturen sandsynligvis vil ændre sig over tid. Endelig skulle reglerne være udtryk for en bred konsensus i hacker-stammen, så håndhævelsen af dem kunne lade sig gøre.

Jeg er begyndt at arbejde på sådan et regelsæt, som jeg forsøgtvis kalder Malvern-protokollen efter den lille by, hvor jeg bor. Hvis den generelle analyse i dette skrift bliver tilstrækkeligt bredt

accepteret, vil jeg gøre Malvern-protokollen offentligt tilgængeligt som en model til løsning af uoverensstemmelse. Hvis nogen er interesseret i at kommentere og udvikle dette regelsæt eller bare give feedback med hensyn til, om de synes, det er en god id eller ej, er de velkomne til at kontakte mig på e-mail.

21. Spørgsmål til yderligere undersøgelse

Kulturens (og min egen) forståelse for store projekter, som ikke følger modellen med en venligsindet diktator, er svag. De fleste sådanne projekter fejler. Nogle få bliver bemærkelsesfuldt succesrige og vigtige (Perl, Apache, KDE). Ingen forstår virkelig hvori forskellen består. (Der er en vag fornemmelse i omløb af, at ethvert sådant projekt er sui generis og står eller falder med gruppedynamikken blandt projektets enkelte medlemmer, men er det sandt, eller er der strategier, som kan efterlignes, og som en gruppe kan følge?)

22. Bibliografi og noter

[Miller] Miller, William Ian; *Bloodtaking and Peacemaking: Feud, Law, and Society in Saga Iceland*; University of Chicago Press 1990, ISBN 0-226-52680-1. En fascinerende undersøgelse af islandsk folkemoot-lov, som både belyser den lockeske teori om ejendoms herkomst og beskriver de senere stadier af en historisk proces, hvor skik blev til sædvanelov og derfra til skreven lov.

[Mal] Malaclypse the Younger; *Principia Discordia, or How I Found Goddess and What I Did To Her When I Found Her*; Loompanics, ISBN 1-55950-040-9. Der er meget oplysende fjol-lethed at finde discordianisme. Midt i den giver SNAFU-princippet en temmelig skarp analyse af hvorfor, magthierakier ikke skalerer særlig godt. Der findes en HTML-version på adressen <http://www.cs.cmu.edu/~tilt/principia/>.

[BCT] J. Barkow, L. Cosmides, and J. Tooby (Eds.); *The adapted mind: Evolutionary psychology and the generation of culture*. New York: Oxford University Press 1992. En glimrende introduktion til evolutionspsykologi. Nogle af disse skrifter har direkte relevans for de tre kulturelle typer, jeg omtaler (magt/bytte/gave), og de antyder, at disse mønstre er temmelig dybt indgroet i den menneskelige psyke.

[MHG] Goldhaber, Michael K.; *The Attention Economy and the Net* *. Jeg opdagede dette skrift efter version 1.7. Det har tydelige fejl (Goldhabers argument om, at økonomisk resonneren ikke egner sig til belysning af opmærksomhed, holder ikke til et nærmere eftersyn), men Goldhaber har ikke desto mindre morsomme og indsigtsfulde ting at sige om opmærksomhedssøgens rolle i organisationsadfærd. Den prestige eller anseelse blandt ligemænd, jeg har omtalt, kan ses som et særligt tilfælde af opmærksomhed, som han forstår det.

[HH] Jeg har opridset hacker-kulturens historie i *A Brief History Of Hackerdom*** . Den bog, som forklarer historien rigtig godt, mangler stadig at blive skrevet, sikkert ikke af mig.

[N] Begrebet 'noosfære' er et obskurt kunstbegreb inden for filosofien. Det udtales nu-u-svære (to u-lyde, en lang med betoning, en kort uden betoning op til svæ). Hvis skal være ulidelig korrekt med hensyn til ens retskrivning, skal det sikkert staves med en trykstreg over det andet 'o'.

Grundigere; dette begreb for "den menneskelige tænknings sfære" stammer fra det græske 'nous', der betyder 'bevidsthed', 'ånd' eller 'ånde'. Det blev opfundet af E. Leroy i *Les origines humaines et l'évolution de l'intelligence* (Paris 1928). Det blev først gjort alment kendt af den russiske biolog og økologipioner Vladimir Ivanovich Vernadsky, (1863-1945), og senere af den jesuittiske palæontolog/filosof Pierre Teilhard de Chardin (1881-1955). Det er med Chardins teori om den fremtidige menneskelige evolution som en form for ren bevidsthed, der kulminerer i forening med guddommen, at begrebet nu primært er tilknyttet.

[DF] David Friedman, en af de mest klare og tilgængelige tænkere inden for moderne økonomi, har skrevet en fortræffelig gennemgang*** den intellektuelle ejendomsrets historie og logik, hvis nogen skulle være interesseret i disse emner.

* Se <http://www.firstmonday.dk/issues/issue2.4/goldhaber>.

** Se <http://www.catb.org/~esr/writings/cathedral-bazaar/hacker-history/>.

*** http://www.best.com/ddfr/Academic/Course/Pages/L.and.E.LS.98/Why_Is_Law/Why_Is_Law_Chapter_11.html

[SP] En interessant forskel mellem Linux- og BSD-verdenen er, at Linux-kernen (og tilknyttede OS-kerneværktøjer) aldrig er blevet forgrenet, mens BSD's er blevet det mindst tre gange. Det er interessant, fordi den sociale struktur i BSD-gruppen er centraliseret på en måde med klart definerede autoritetslinier og med intentionen om at undgå forgrening, mens det decentraliserede og amorf Linux-miljø ikke tager sådanne forholdsregler. Det virker som om, at de projekter, som åbner mest for udviklingen, faktisk har mindst tendens til at forgrene sig!

Henry Spencer `jhenry@spsystems.net` foreslår, at i almindelighed er et politisk systems stabilitet omvendt proportionelt med graden af adgangsbarrierer til dets politiske proces. Hans analyse er værd at citere her:

En stor styrke ved et relativt åbent demokrati er, at de fleste potentielle revolutionære synes, det er lettere delvist at opnå deres mål ved at arbejde indenfor systemet end ved at angribe det. Denne styrke undermineres let, hvis etablerede parter sammen ”hæver adgangsgænsen” og gør det sværere for små utilfredse grupper delvist at opnå *nogle* af deres mål.

(Et lignende princip kan ses i økonomien. Åbne markeder har den stærkeste konkurrence og i almindelighed de bedste og billigste produkter. Derfor er det i høj grad i etablerede virksomheders interesse at gøre det vanskeligt at komme ind på markedet — til eksempel ved at overbevise regeringer om, at de skal kræve udførlige RFI-testning af computere eller ved at skabe ’konsensus’-standarder, der er så komplekse, at de ikke kan implementeres effektivt fra starten af uden store ressourcer. Markederne med de stærkeste adgangsbarrierer er dem, som angibes stærkest af revolutionære, fx. Internet og Det amerikanske Justitsministerium mod Bell System.)

En åben proces med lave adgangsbarrierer opmuntrer til deltagelse snarere end løsrivelse. Resultaterne er måske ikke så imponerende, som de ville være ved løsrivelse, men prisen er lavere, og de fleste mennesker betragter det som et acceptabelt bytte. (Da den spanske regering tilbagekaldte Francos antibaskiske love og tilbød de baskiske provinser deres egen skoler og begrænset lokal autonomi, forsvandt det meste af den baskiske frihedsbevægelse fra den ene dag til den anden. Kun hard-core marxister insisterede på, at det ikke var godt nok.)”

[RP] Der er nogle spidsfindigheder omkring uautoriserede fejlrettelser. Man kan dele dem i ’venlige’ og ’uvenlige’ typer. En ’venlig’ fejlrettelse er lavet, så den kan indarbejdes i projektets hovedlinje under vedligeholderens kontrol (uanset om indbejdningen faktisk sker); en ’uvenlig’ fejlrettelse er lavet for at rykke projektet i en retning, som vedligeholderen ikke billiger. Nogle projekter (især Linux-kernen selv) har temmelig afslappet holdning til venlige fejlrettelser og opmuntrer endda uafhængig distribution af dem som en del af deres beta testfase. På den anden side repræsenterer en uvenlig fejlrettelse en beslutning om at konkurrere med den originale, og det er alvorligt. At vedligeholde en hel række af uvenlige fejlrettelser har tendens til lede til forgrening.

[LW] Jeg er taknemmelig overfor Michael Funk `jmwfunk@uncc.campus.mci.net` for at gøre opmærksom på, hvor lærerig en kontrast til hackerne piratkulturen er. Linus Walleij har offentliggjort en analyse af deres kulturelle dynamik, som adskiller sig fra min (ved at beskrive dem som nødvendighedskultur) i A Comment on ’Warez D00dz’ Culture*.

Kontrasten varer måske ikke ved. Tidligere cracker Andrej Brandt `jandy@pilgrim.cs.net.pl` oplyser, at han tror, at cracker/warez d00dz-kulturen er ved at visne væk, da dens smarteste folk og ledere optages i open source-verdenen. Uafhængig bevis for dette synspunkt ses måske i den banebrydende handling, som cracker-gruppen, der kalder sig ’Cult of the Dead Cow’, udførte i juli 1999. De har frigivet deres ’Back Orifice 2000’ til at bryde Microsoft Windows sikkerhedsværktøjer med under GPL.

[HT] I evolutionære begreber kan en håndværkers trang selv (ligesom internaliserede etik) være et resultat af den høje risiko for og omkostning ved bedrag. Evolutionspsykologer har samlet eksperimentielt bevis [BCT] for, at mennesker har dele i hjernen, der er specialiseret til at opdage socialt bedrag, og det er temmelig let at se, hvorfor vores forfædre skulle være udvalgt for evne til at opdage snyd. Hvis man derfor ønsker at have et ry, der kan give fordele, men som er risikable og omkostningsfulde, kan det måske endda være en bedre taktik faktisk at have dette ry end at lade som om. (”Ærlighed varer længst”)

* Se http://www.df.lth.se/triad/papers/Raymond_D00dz.html.

Evolutionary psykologer har foreslået, at dette forklarer adfærd som værtshusslagsmål. Blandt yngre voksne mænd kan det være en fordel både socialt og (selv i vore dages feministpåvirkede klima) seksuelt at have et rygte for "at være hård". Det er dog ekstremt risikabelt at simulere det "at være hård"; det negative resultat at blive opdaget efterlader en i en værre situation, end hvis man aldrig har hævdet at have det træk. Omkostningerne ved bedraget er så høje, at det nogen gange er bedre at minimaksimere for at internalisere "hårdhed" og risikere alvorlige skader i et slagsmål for at bevise det. Parallele observationer er blevet udført omkring mindre kontroversielle træk som 'ærlighed'.

Selv om de primære meditationslignende belønninger ved kreativt arbejde ikke bør undervurderes, er håndværkerens trang formentlig i det mindste delvist netop sådan en internalisering (hvor det grundlæggende træk er 'evne til flittigt arbejde' eller noget lignende).

[MH] Et kortfattet sammendrag af Maslows pyramide og beslægtede teorier kan findes på Internettet her: Maslow's Hierarchy of Needs*

[DC] Men at kræve ydmyghed af ledere kan være et mere generelt karakteristika ved gave- eller overflodskulturer. David Christie jdc@netscape.com, fortæller om en tur gennem Fijis ydre øer: "Hos byhøvdingene på Fiji, så vi den samme slags selvudslettende lederstil med lav profil, som du tilskriver open source-projektledere. (...) Selv om de nyder stor respekt og har selvfølgelig al den aktuelle magt på Fiji, demonstrerede høvdingene, som vi mødte, ægte ydmyghed og ofte en helgenlignende accept af deres pligt. Dette er særlig interessant i lyset af, at det at være høvding er en arvelig rolle, ikke en valgt stilling eller en popularitetskonkurrence. På en eller anden måde er de trænet til det af kulturen selv, selv om de er født til det og ikke valgt af deres ligemænd." Han fortsætter og understreger, at han tror, at høvdingene fra Fijis karakteristiske stil skyldes, at de ikke har mulighed for tvingende samarbejde: en høvding har "ikke nogen gulerod eller stor pisk".

[NO] Det ses tydeligt, at dem, som stifter succesfulde projekter, får mere prestige end dem, der med rette kan siges at udføre den samme mængde arbejde ved at finde fejl og hjælpe til med succesfulde projekter. "En tidligere version af dette skrift stillede spørgsmålet "Er det en rationel vurdering af sammenlignelig indsats, eller er det en følgeeffekt ved den ubevidste territoriale model, vi har påberåbt her?" Adskillige læsere foreslog overbevisende og i bund og grund tilsvarende teorier. Den følgende analyse af Ryan Waldron jrew@erebor.com, formulerer det godt:

"I følge den lockeske teori om jord har den, som etablerer et nyt og succesfuldt projekt i bund og grund opdaget eller åbnet for et nyt territorie, som andre kan nybygge. For de fleste succesfulde projekter er der et mønster med faldende afkast, så efter et stykke tid er æren for bidrag til et projekt blevet så spredt, at det er svært fåbetydelig anseelse for en sen deltager uanset kvaliteten af hans arbejde.

Hvor godt et stykke arbejde ville jeg til eksempel være nødt til at udføre ved at lave modifikationer af perl-koden for at få bare en brøkdel af den anerkendelse for min deltagelse, som Larry, Tom, Randall og andre har fået.

På den anden side hvis et nyt projekt bliver stiftet (af en anden) i morgen, og jeg er en tidlig og hyppig deltager i det, er min mulighed for at få en del af den respekt, som følger af sådan et succesfuldt projekt, stærkt øget af min tidlige deltagelse deri (under antagelse af, at bidragene er af samme kvalitet). Jeg forestiller mig, at det er det samme som med dem, der investerede tidligt i Microsoftaktier, og dem der investerede i dem senere. Alle kan tjene på det, men de tidlige deltager tjener mere. Derfor vil jeg på et eller andet tidspunkt være mere interesseret i et nyt og succesfuldt selskab, der skal noteres på børsen, end jeg vil deltage i den fortsatte forøgelse af et eksisterende selskabs aktier." Ryan Waldrons analogi kan udvides. Stifteren af projektet udføre et missionerende salg af en ny id, som kan være acceptabel eller ej eller være brugbar for andre. Derfor udsætter stifteren sig for noget, der kan minde om risikoen ved en børsintroduktion (risiko for mulig skade på deres omdømme), endnu mere end andre, der hjælper til med et projekt, der allerede har opsamlet nogen accept blandt deres ligemænd. Stifterens belønning er konsistent på trods af det faktum, at assistenterne måske udfører mere arbejde i virkeligheden. Det kan let ses som analog til forholdet mellem risiko og belønning i en bytteøkonomi.

* Se <http://www.valdosta.peachnet.edu/whuitt/psy702/regsys/maslow.html>.

Andre læsere har bemærket, at vores nervesystem er indstillet til at bemærke forskelle, ikke regelmæssige tilstande. Den revolutionære forandring, som opleves ved skabelsen af et nyt projekt, er derfor meget mere bemærkelsesværdig end den akkumulerende effekt af konstant gradvist voksende forbedring. Således hædres Linus som Linux' fader, selv om nettoeffekten af tusinder af andre bidrageres forbedringer har bidraget mere til det succes som OS end en mands arbejde nogensinde kunne.

[HD] Udtrykket "afkommercialisere er en reference til the Halloween Documents* hvor Microsoft helt åbent taler om "afkommercialisering" som deres mest effektive langtidstrategi for at fastholde kunderne i et udbytende monopol.

[GNU] På Free Software Foundations hovedhjemmeside er der en artikel**, som opsummerer resultatet af disse undersøgelser. Citaterne i dette skrift er taget derfra.

23. Anerkendelser

Robert Lanphier [robla@real.com] har bidraget meget til diskussionen om jegløs adfærd. Eric Kidd [eric.kidd@pobox.com] belyste den rolle som værdsættelse af ydmyghed spiller for at undgå personlighedskulter. Afsnittet om de globale effekter var inspireret af kommentarer fra Daniel Burn [daniel@tsathoggua.lab.usyd.edu.au]. Mike Whitaker [mrw@entropic.co.uk] inspirerede hovedtideen i afsnittet om kulturindlæring. Chris Phoenix [cphoenix@best.com] gjorde opmærksom på vigtigheden af det faktum, at hackere ikke kan opnå anseelse nedgøre andre hackere.

24. Versionshistorie

Jeg er alene ansvarlig for hvad, der står i dette skrift, og for eventuelle fejl og misforståelser. Jeg har fået kommentarer og feedback, og jeg har brugt dem til at forbedre skriftet — en process, som jeg ikke forventer at afslutte på noget på nuværende fastsat tidspunkt.

10. april 1998: Version 1.2 offentliggjort på nettet.

12. april 1998: Version 1.3: Rettelser af slåfejl og svar på første runde af offentlige kommentarer. Første fire punkter i bibliografien. En anonymt bidraget bemærkning om, at ansproing til anseelse fungerer selv, når håndværkeren ikke er klar over det. Tilføjede instruktive kontraster til warez d00z, materiale om præmissen 'software bør tale for sig selv' og bemærkninger om at undgå personlighedskulter. Som resultat af alle disse forandringer voksede afsnittet 'Problemet med ego' og blev delt.

16. april 1998: Version 1.7: Nyt afsnit om 'Globale følger' diskuterer historiske trends i koloniseringen af noosphere og undersøger fænomenet 'kategoridræber'. Tilføjede endnu et forskningspørgsmål.

27. april 1998: Version 1.8: Tilføjede Goldhaber til bibliografien. Det er den version, der vil komme i Linux Expo-meddelelserne.

26. maj 1998: Version 1.9: Indarbejdede Far Rideaus skelnen mellem noosphere/ergosphere. Indarbejdede RMS' påstand om, at han ikke er antikommerciel. Nyt afsnit om kulturindlæring og den akademiske verden (tak til Ross J. Reedstrom, Eran Tromer, Allan McInnes, Mike Whitaker og andre). Mere om ydmyghed, ('egoløs adfærd') from Jerry Fass og Marsh Ray.

11. juli 1998: Version 1.10. Fjernede Far Rideaus reference til 'berømmelse' efter hans opfordring.

21. november 1998: Version 1.14: Mindre rettelser af korrektur og døde links.

8. august 1999: Større revision til O'Reilly-bogen. Indarbejdede nogle ideer om omkostningerne ved forgrening og uautoriserede fejlrettelser fra Michael Chastain. Thomas Gagne [tgagne@ix.netcom.com] opdagede ligheden mellem "senoritet vinder" og database heuristik. Henry Spencers politiske analogi. Ryan Waldron og El Howard [elhoward@hotmail.com] bidrog med betragtninger om nyhedens værdi. Thomas Bryan [tbryan@arlut.utexas.edu] forklarede hackerens modvilje mod "amfavn og udvid". Darcy Horrock inspirerede det nye afsnit "Hvor god en gave?".

* <http://www.catb.org/esr/halloween/>

** Se <http://www.gnu.org/philosophy/motivation.html>.

Andet nyt materiale om forbindelsen mellem Maslovs behovspyramide og tabuet mod angreb på kompetence.