



# Systemek Kompis

2009-03-10



# Fri og åpen programvare

...og litt til, hvis vi får tid

Dag-Erling Smørgrav  
Seniorutvikler, Systemek AS

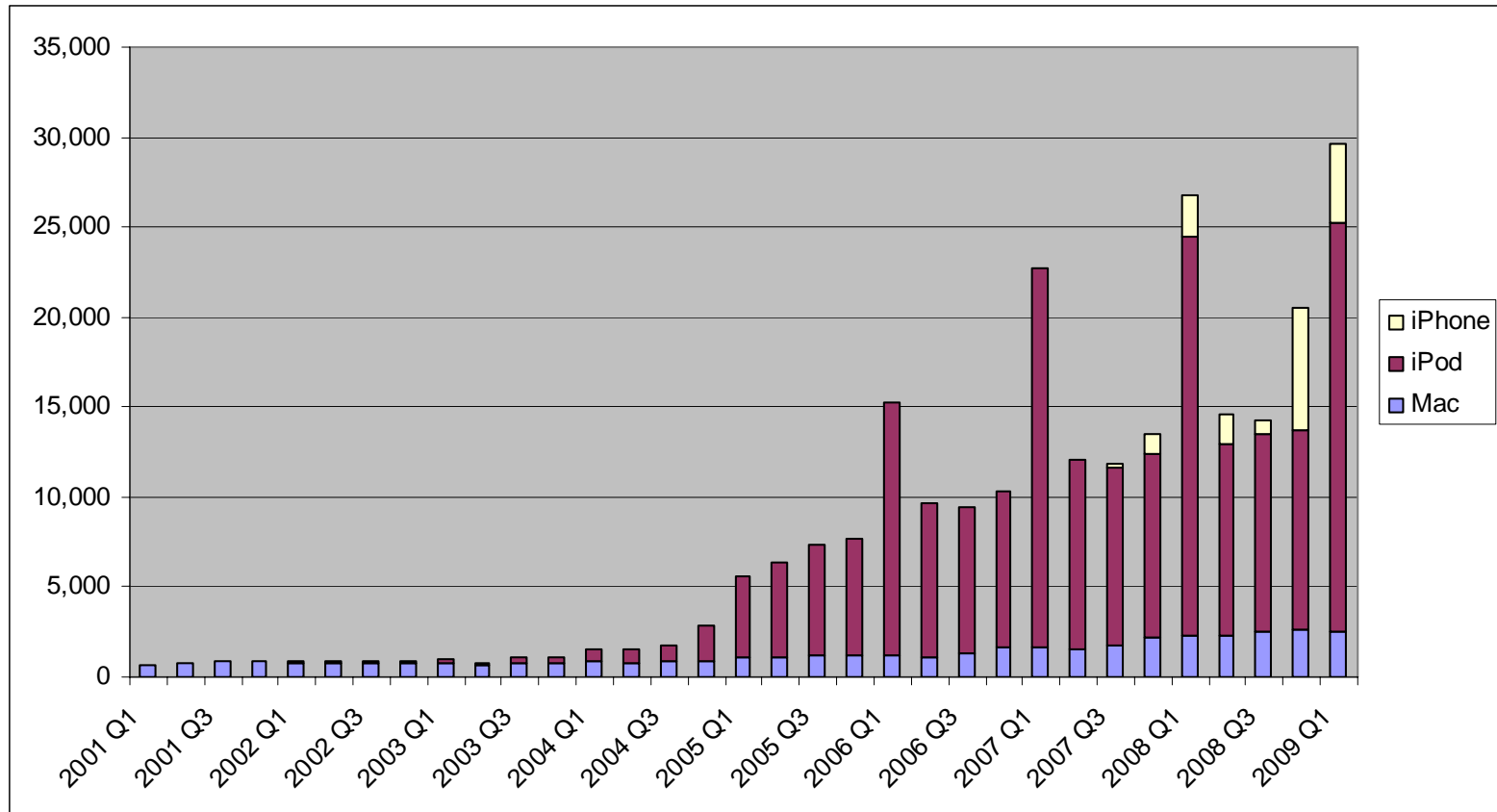
[dag-erling.smorgrav@systemek.no](mailto:dag-erling.smorgrav@systemek.no)



# Fri og åpen programvare er overalt



## Apple: 255M enheter solgt siden 2001





# Opphavsrett

Åndsverksloven (åvl)  
oppsummert av en legmann

(spør advokaten din om detaljene)



# Opphavsrett

- Hovedregel: opphavsmann har enerett til å «råde over verket» (§ 2)
  - Kopiere
  - Stille ut
  - Fremføre
- Opphavsmann kan overdra hele eller deler av denne retten (§ 39)
  - Det kan knyttes betingelser til overdragelsen (§ 39a)
  - Overdragelsen gir ikke uten videre rett til å endre verket (§ 39b)



## Verkshøyde

- Ikke nedfelt i åvl, men i rettspraksis
- Noe ullent
- Forutsetter at det «foreligger mulighet for variasjon»
- Tilsvarende prinsipp i amerikansk rett har vært avgjørende i bl.a. *USL v. BSDi* og *SCO v. Novell*
- Best å styre godt klar av skillelinjen



**lisen's m1** (lat. *licentia* av *licet* '(det) er tillate')

løyve; m a: løyve til å bruke radio- el. Fjernsyns-  
apparat; dispensasjon; rett til å nytte ei oppfinning  
e I ein annan har patent på *tannlækjarl- / våpenl- /*  
*produsere på l- / søkje l-*

lisensavgift *betale l-*



# Lisens

I denne sammenhengen:

Tillatelse til å bruke et åndsverk  
under de betingelser  
som er satt av opphavsmann  
(utover det åvl sikrer)



# Begreper

En rask oversikt – vi kommer til å se nærmere på detaljene senere.



# Begreper



Copyleft

Free Software

Open Source Software

Public Domain

Open Content



Copyleft

Fri programvare

Åpen programvare

Offentlig eiendom

Åpent innhold



# Copyleft

Ordspill på «Copyright», gjerne fulgt av andre ordspill som «All rights reversed» eller «All wrongs deserved». Varierende betydning i historiens løp.





# Fri programvare

Lisenser som tillater modifikasjon og videre-distribusjon under samme eller tilsvarende lisens. Hensikten er å sikre brukerens rett til å inspisere og tilpasse programvaren til sine egne behov. Frontes av Free Software Foundation (FSF).

<http://www.fsf.org/>



**FREE SOFTWARE**  
F O U N D A T I O N



# Åpen programvare

Lisenser som tillater modifikasjon og videredistribusjon under varierende betingelser, herunder også fri programvare. Frontes av Open Source Initiative (OSI).

<http://www.opensource.org/>





# Offentlig eiendom

Et verk er i offentlig eiendom dersom:

- vernetiden har utløpt (åvl § 40, § 41, § 41a)
- opphavsmann har sagt fra seg opphavsretten
- (det ikke har verkshøyde)

Det kan da brukes fritt (men se åvl § 3, § 48 om «opphavsmannens anseelse og egenart» og «almene kulturinteresser»).



## Åpent innhold

Tilsvarende konsept som åpen programvare, men for innhold (tekst, bilder, lyd, film m.m.)

Eksempler:

- Wikipedia: GFDL
- Wikimedia Commons: GFDL, Creative Commons
- Flickr: utstrakt bruk av Creative Commons
- Mye av dokumentasjonen for fri og åpen programvare



# Litt historie

OK, mye historie



## Tiny BASIC

1975: Dennis Allison skriver en artikkelserie i Bob Albrechts *People's Computer Company* om en minimal BASIC-implementasjon, og oppfordrer lesere til å skrive og sende inn sine egne:

*«Let us stand on each others' shoulders;  
not each others' toes»*



# Dr Dobb's Journal

Januar 1976: Allison & Albrecht publiserer disse i *Dr Dobb's Journal of Tiny BASIC Calisthenics & Orthodontia* (som senere blir til *Dr. Dobb's Journal of Computer Calisthenics*).

Juni 1976: DDJ publiserer Li-Chen Wangs Tiny BASIC-implementasjon merket «@COPYLEFT». Wang bruker ordet i betydningen av å frasi seg all opphavsrett til programmet.



# Open Letter to Hobbyists

Januar 1976: I frustrasjon over utstrakt kopiering av Altair BASIC skriver Bill Gates et åpent brev til dataamatører: «Who can afford to do professional work for nothing?»

Juli 1976: Som et tilsvarende til dette beskriver DDJ-redaktør Jim Warren tanken om fri programvare i SIGPLAN.



## MIT AI Laboratory

1973: Richard Greenblatt og Thomas Knight starter MITs *Lisp Machine Project* som i løpet av de neste årene utvikler først CONS-maskinen og deretter CADR-maskinen.

1979: Russell Noftsker foreslår å kommersialisere CADR-maskinen

1980: Etter mye intern krangling starter Noftsker Symbolics, Inc. mens Greenblatt starter Lisp Machines, Inc.



# Richard Stallman

- 1971: Stallman begynner ved MIT AI Lab, og jobber bl.a. med operativsystemet til CONS- og CADR-maskinene.
- 198x: Flere hendelser, bl.a. relatert til krigen mellom Symbolics og Lisp Machines, overbeviser Stallman om at tilgang til kildekode er en menneskerett. Dette er utgangspunktet for fri programvare: brukeren har rett til å kunne tilpasse programvaren til sine behov.



## GNU

1983: Stallman annonserer sin plan om å skrive et fritt Unix-derivat (GNU's Not Unix).

1985: Stallman begynner å bruke ordet «copyleft» i betydningen av å bruke opphavsretten til å befeste programvarens (ikke brukerens) frihet.

1985: GNU Emacs blir gitt ut.

1987: GNU C Compiler blir gitt ut.

1989: Stallman skriver GNU Public License (GPL).



# GNU/Linux

1991: Linus Torvalds får en PC og begynner å skrive en Unix-aktig kjerne for den, inspirert av Minix.

1992: GNU er stort sett komplett, men mangler en kjerne.

1992: Torvalds utgir Linux under GPL. Kombinasjonen av Linux-kjernen og GNU-verktøyene blir til det vi i dag kjenner som Linux (egentlig GNU/Linux).

Arbeidet med å skrive en egen kjerne for GNU fortsetter dog under navnet Hurd.



# Berkeley Software Distribution

- 1977: BSD oppstår ved University of California at Berkeley (UCB) som en samling av Unix-verktøy.
- 1978: UCB får en VAX, men det medfølgende OSet, Unix/32V, støtter ikke virtuelt minne. Studenter begynner å skrive om kjernen.
- 1979: Tredje utgave av BSD (3BSD) er en full Unix-distribusjon basert på 32V, med den nye kjernen og de eksisterende BSD-verktøyene.



# BSD-stakken

1983: CSRG utgir 4.2BSD med egen TCP/IP-stakk, i stedet for den offisielle implementasjonen fra BBN.

1986: CSRG utgir 4.3BSD; BSD-stakken har nå overtatt BBN-stakkens plass som den offisielle TCP/IP-implementasjon.

1989: CSRG utgir nettverkstakken alene som 4.3BSD Net/1; første bruk av BSD-lisensen. Resten av systemet inneholder fortsatt store deler AT&T-kode, og kan kun distribueres til institusjoner som allerede har en AT&T-lisens.



# USL v. BSDi

- 1991: CSRG forsøker å erstatte all AT&T-kode i BSD med nyskrevet kode og utgir dette (under BSD-lisens) som 4.3BSD Net/2.
- 1992: BSDi begynner å selge BSD/386. BSDi og UCB blir saksøkt av Unix System Laboratories (USL).
- 1994: Minnelig ordning i BSDis og UCBS favør. CSRG utgir 4.4BSD-Lite, helt fri for AT&T-kode.
- 1995: CSRG utgit 4.4BSD-Lite2, og oppløses.



# Free, Net, Open

1993: FreeBSD og NetBSD blir grunnlagt i skyggen av rettsaken. Førstnevnte fokuserer på Intel 386, sistnevnte er multiplattform.

1995: OpenBSD blir grunnlagt av Theo de Raadt etter at han blir kastet ut av NetBSD-prosjektet.

2003: DragonFly BSD blir grunnlagt av Matt Dillon etter at han blir kastet ut av FreeBSD-prosjektet.



## X Window System

1983: MIT starter Project Athena.

1984: X blir utviklet for å dekke Project Athenas behov for et enhetlig grafisk miljø i en heterogen maskinpark.

1988: X11 gis ut under (det som nå kalles) MIT-lisensen.

1988: X Consortium dannes for å forvalte og videreutvikle X11.



## XFree86

1991: Thomas Roell skriver en X-server for PC, X386. Den blir inkludert i X11R5 under MIT-lisensen.

1992: Roell velger å videreutvikle X386 som et kommersielt produkt. Andre fortsetter arbeidet med den MIT-lisensierte versjonen under navnet XFree86 i løst samarbeid med X Consortium.



# The Open Group

1997: X Consortium overlater X11 til The Open Group (TOG), som også forvalter UNIX-varemerket og Single UNIX Specification.

1998: TOG utgir X11R6.4 under en ny, restriktiv lisens. XFree86-prosjektet truer med å fortsette å utvikle X11 uavhengig av The Open Group. TOG trekker tilbake den nye lisensen.



## X.org

1999: TOG danner X.org, som får ansvaret for X11. I realiteten eksisterer X.org bare på papiret.

2003: XFree86-prosjektet kollapser etter flere år med intern maktkamp og intrigemakeri. En utbryter-gruppe danner det kortlivede Xouvert-prosjektet.

2004: XFree86 svarer med å bytte til en mer restriktiv lisens. De fleste gjenværende utviklere hopper av, og gjenoppliver X.org med støtte fra TOG. XFree86 går snart i glemmeboken.



# Free Software Foundation

Ideell organisasjon stiftet i 1985 av Stallman, dels for å kunne samle inn penger til å finansiere utviklingen av GNU-prosjektet, og dels som en juridisk person som kan eie rettighetene til koden.



# Open Source Initiative

Stiftet i 1998 av Bruce Perens og Eric Raymond for å fremme åpen programvare. Opererer med en egen «Open Source Definition» på ti punkter og en prosess for sertifisering av lisenser.



# Hvordan bruke fri og åpen programvare

og hvordan *ikke* bruke det



## Grunnregler

1. Vit hva du bruker - en enkelt pakke kan inneholde kode under forskjellige lisenser.
2. Vit hva lisensen innebærer.
3. Unngå programvare dekket av tvetydige, uklare, eller kompliserte lisenser.
4. Hvis du selv skal utgi programvare under en fri / åpen lisens, bruk en av de kjente og etablerte lisensene – og la for all del være å skrive din egen!



## Scenarier

1. Bruk av eksisterende fri / åpen programvare internt i organisasjonen
2. Bruk av eksisterende fri / åpen programvare som komponent i et system
3. Bruk av eksisterende fri / åpen programvare som del av egen programvare
4. Utgivelse av egen programvare under fri / åpen lisens



# Bruk av eksisterende fri / åpen programvare internt i organisasjonen

Eksempel: bruk av et fritt / åpent  
CRM-system



## Ikke distribusjon

Det er organisasjonen, ikke den individuelle ansatte, som er part i lisensen. Intern bruk og spredning av programvaren, modifisert eller ei, utgjør dermed ikke distribusjon.

- Lisenser som setter begrensninger for *hvem* som kan bruke programvaren, eller *hva* den kan brukes til, er ikke frie / åpne.



## Bruk av eksisterende fri / åpen programvare som komponent i et system

Eksempler: bruk av en fri / åpen  
databasemotor i en web-portal; bruk av  
en fri / åpen databasemotor i en  
nøkkelferdig løsning levert til kunde.



## Pass på detaljene

De fleste lisenser tillater dette uten videre, men detaljene varierer:

- Noen setter ingen begrensninger eller betingelser
- Andre krever at kildekoden *til komponenten* gjøres tilgjengelig for de som systemet distribueres til.

Noen lisenser har et utvidet distribusjonsbegrep.

- Jfr. ASP-smutthullet.



# Bruk av eksisterende fri / åpen programvare som del av egen programvare

Herunder også distribusjon av fri /  
åpen programvare med egne  
modifikasjoner



## Derivatverk

Dette er det mest kompliserte tilfellet.

- Du eier rettighetene til *din* kode / *dine* endringer.
- Den originale lisensen gjelder fortsatt for den originale koden.
  - Den du distribuerer derivatverket til har dermed – i teorien – rett til å videredistribuere de delene du ikke selv har skrevet.
  - Noen lisenser krever at *hele* derivatverket distribueres under samme lisens som originalen, eller en annen kompatibel lisens.
  - Noen lisenser legger overhodet ingen begrensninger.
  - Stort sett alle krever at du krediterer opphavet til det originale verket.

Se omtale av de enkelte lisensene senere i presentasjonen.



## Utgivelse av egen programvare under fri / åpen lisens

Eksempel: publisering av et  
egenutviklet funksjonsbibliotek til et  
åpent fagforum.



## Tenk deg godt om

- Ha en klar formening om hvorfor du gjør dette.
- Forstå brukernes forventninger:
  - Bygge- og installasjonsrammeverk
  - Et minimum av dokumentasjon
  - Mulighet til å rapportere feil
  - Anledning til å foreslå forbedringer
  - En følelse av at ovenstående blir tatt alvorlig
- Velg lisens med omhu.
- Pass på å ikke blande inn tredjeparts kode som du har fått under en ukompatibel lisens.



# Lisenser

Med anbefalinger



# Permissive



## MIT

Den «åpneste av de åpne»:

- Programvaren kan brukes, modifiseres og distribueres fritt, så lenge copyright- og lisensteksten ikke fjernes.
- Dette innebærer at programvaren kan inkluderes i annen programvare distribuert under en annen lisens.



## BSD

Sannsynligvis inspirert av MIT-lisensen. Original-utgaven har to klausuler i tillegg:

- Alt reklamemateriell som omtaler programvaren må kreditere opphavsmannen.
  - Ikke OSI-godkjent, ikke GPL-kompatibel, opphevet av UCB i juli 1999.
- Opphavsmannens navn kan ikke brukes for å promotere derivater.
  - OSI-godkjent
  - «Simplified BSD» utelater denne – også OSI-godkjent



# ISC

Skrevet av Internet Systems Consortium.  
Tilsvarende MIT-lisensen (og Simplified BSD).  
Kortere og enklere skrevet.



# MIT, BSD og ISC

- *Bruk gjerne programvare dekket av disse lisensene.*
  - Men unngå helst «reklameklausulen».
- *Bruk den gjerne selv:*
  - BSD med tre klausuler – hvis du er bekymret for at ditt navn kan misbrukes til å promotere derivater.
  - MIT, BSD med to klausuler, eller ISC – ingen nevneverdige forskjeller, velg det som passer best.
    - ISC er den enkleste



## zlib / libpng

Enda enklere enn MIT / BSD / ISC: copyright- og lisens-teksten skal alltid følge med kildekoden, men ikke med binærkode.

Godt egnet for referanseimplementasjoner av åpne standarder, og annen kode som man ønsker at andre skal kunne bruke uten å måtte tenke seg om.

*Bruk gjerne programvare dekket av den – uten forbehold.*

*Bruk den gjerne selv, hvis hensikten din er at programvaren skal spres så vidt som mulig.*



# GNU



## GNU GPL 2

GPLv2 er sannsynligvis den mest utbredte lisensen, og det fremste uttrykket for fri programvare slik Stallman og FSF definerer det.

*Bruk gjerne programvare dekket av den, men vær klar over konsekvensene for eventuelle modifikasjoner.*

*Bruk den gjerne selv, hvis du ikke ønsker at andre skal modifisere i koden din uten å dele endringene.*



# GNU Lesser GPL 2

En variant av GPLv2 beregnet på biblioteker. Et program som bruker biblioteket slik det er ment, uten å integrere det i sin egen kode, trenger ikke selv distribueres under LGPLv2.

L-en stod opprinnelig for *Library*. Navnet ble endret for å gjenspeile en dreining i FSFs syn på den. FSF ønsker at den kun brukes til programvare som ikke har noen annen kompetitiv fordel.

*Bruk gjerne programvare dekket av den, med samme begrunnelse som for GPLv2.*

*Bruk den gjerne selv, med samme begrunnelse som for GPLv2; biblioteket ditt vil potensielt få større utbredelse under GPL enn under LGPL.*



# ASP-smutthullet

Yoyodyne, Inc. lager en web-portal basert på fri programvare. I forbindelse med dette arbeidet gjør Yoyodyne store forbedringer i den underliggende programvaren. Programvaren blir tilgjengeliggjort for, men ikke distribuert til, brukerne av portalen. Dermed trår ikke lisensens betingelser for videre-distribusjon i kraft, og Yoyodyne kan holde modifikasjonene sine for seg selv.



# GNU Affero GPL 1

Utviklet av Affero i samarbeid med FSF for å tette ASP-smutthullet. Den er identisk med GPLv2, med en ekstra klausul som krever at *brukere*, ikke bare mottakere, skal ha tilgang til kildekoden; denne klausulen utgjør en «ytterligere begrensning» som gjør den ukompatibel med GPL.

*Bruk gjerne programvare dekket av den, men vær klar over konsekvensene for eventuelle modifikasjoner.*

*Bruk den gjerne selv, hvis ASP-smutthullet bekymrer deg.*



# Tivoisering

TiVo kjører Linux (distribuert under GPLv2), og brukere kan laste ned kildekode fra [tivo.com](http://tivo.com) eller få den tilsendt på forespørsel. Produsenten etterkommer derved lisensens *bokstav*.

Maskinvaren nekter dog å kjøre programvare som ikke er kryptografisk signert av produsenten. Dette strider mot lisensens *ånd*, som er at brukeren skal kunne tilpasse programvaren til sine behov.



# Novell-Microsoft-avtalen

I 2006 inngikk Novell og Microsoft en avtale om gjensidig lisensiering av hverandres patenter og om bedre kompatibilitet mellom Microsoft-programvare og fri / åpen programvare. Microsoft støtter for eksempel utviklingen av Moonlight, som er en fri reimplementasjon av Silverlight.

Denne avtalen var svært kontroversiell, og flere Novell-ansatte sa opp i protest. Det fremste problemet er at det er uklart hvorvidt denne avtalen dekker alle som bruker fri programvare distribuert av Novell, eller bare betalende kunder. Gode intensjoner teller lite så lenge de ikke er festet til papir.



## GNU GPL 3

GPLv3 er en videreutvikling av GPLv2 med klausuler som skal forhindre både tivoisering og krysslisensiering.

Den ble svært dårlig mottatt. Linus Torvalds gikk hardt ut mot den, og fortsetter å distribuere Linux under GPLv2. BSD-prosjektene vedtok restriksjoner mot innføring av GPLv3-programvare i kodebasen. Apple begynte å finansiere utviklingen av BSD-lisensierte kompilatorer, siden GNU-kompilatorene heretter vil distribueres under GPLv3.

*Unngå programvare dekket av den, fordi tilleggsklausulene er innfløkte og vanskelige å forholde seg til.*

*Ikke bruk den selv, fordi du risikerer at mange vil nekte å bruke programvaren din.*



## GNU LGPL 3

LGPLv3 er en videreføring av LGPLv2 basert på GPLv3, med samme klausul om unntak for programvare som bruker koden som et selvstendig bibliotek.

*Unngå programvare dekket av den, med samme begrunnelse som for GPLv3.*

*Ikke bruk den selv, med samme begrunnelse som for GPLv3.*



# GNU Affero GPL 2

En meningsløs lisens som hadde til hensikt å fungere som en bro mellom AGPLv1 og AGPLv3 (se neste slide).

For alle praktiske formål kan programvare distribuert under AGPLv2 håndteres som om den var distribuert under AGPLv3.



# GNU Affero GPL 3

AGPLv3 er en videreføring av AGPLv1 basert på GPLv3, med samme tilleggsklausul om tilgjengeliggjøring for brukere som AGPLv1. I tillegg har både AGPLv3 og GPLv3 klausuler som eksplisitt tillater å kombinere begge lisensene i samme verk.

*Unngå programvare dekket av den, med samme begrunnelse som for GPLv3.*

*Ikke bruk den selv, med samme begrunnelse som for GPLv3.*



# Øvrige



# Apache License 1.0

En lite gjennomtenkt utvidelse av den originale BSD-lisensen (*inkludert* den problematiske «reklameklausulen») med to ekstra klausuler:

- Derivater kan ikke bruke Apache-navnet
- Redundant krav om kreditering

Det finnes en variant (Apache License 1.1) hvor «reklameklausulen» er fjernet.

*Bruk gjerne programvare dekket av den, men vær klar over konsekvensene av «reklameklausulen».*

*Ikke bruk den selv, fordi det finnes bedre lisenser som dekker behovene dine.*



## Apache License 2.0

En lang mer gjennomtenkt erstatning for den forrige Apache-lisensen.

Kan betraktes som en langt mer detaljert utgave av BSD-lisensen, med noen ekstra klausuler (blant annet om patenter) hentet fra GPLv2.

Tillater bruk i lukket programvare.

*Bruk gjerne programvare dekket av den, men pass på at den har veldig presise krav til kreditering.*

*Bruk den gjerne selv, men vær klar over konsekvensene av patentklausulen.*



# Adaptive Public License

En merkelig «à la carte»-lisens: over ti sider med ugjennomtrengelig legalesisk og et avkrysnings-skjema som angir hvilke deler av den foregående teksten som faktisk gjelder.

*Unngå programvare dekket av den, fordi du kan aldri være helt trygg på hvilke bestemmelser som gjelder.*

*Ikke bruk den selv, fordi det finnes bedre lisenser som dekker behovene dine.*



## Common Public Attribution License

Hensikten er i utgangspunktet å tette ASP-smutthullet, men den krever i tillegg at forfatteren krediteres i brukergrensesnittet. Det største problemet er imidlertid en tvetydig formulering angående bruk av varemerker.

*Unngå programvare dekket av den, fordi den kan potensielt brukes mot deg.*

*Ikke bruk den selv, fordi det finnes bedre lisenser som dekker behovene dine.*



## Artistic License

- Finnes i to utgaver. Begge er like dårlige.

*Bruk gjerne programvare dekket av den, det er problemfritt så lenge du ikke modifiserer den.*

*Ikke bruk den selv, fordi det finnes bedre lisenser som dekker behovene dine.*



# OpenSSL / SSLey

En forferdelig sammensetning av Apache-lisensen med en maltraktert variant av BSD-lisensen. Brukes kun av OpenSSL (tidligere SSLey).

*Unngå for enhver pris.*

– med det forbehold at det er trygt å bruke OpenSSL-biblioteket som følger med operativsystemet. Vurder gjerne å bruke GnuTLS i stedet, men pass på at enkelte deler av GnuTLS er under GPL, ikke LGPL.



# The End