

Programvarepolitikk for fremtiden

Teknologiske strategier for et åpnere samfunn

Programvarepolitikk for fremtiden

Teknologiske strategier for et åpnere samfunn

ISBN 82-92447-01-6

Utgitt: Oslo, november 2004

Omslag: Enzo Finger Design AS

Copyright © Teknologirådet

Elektronisk publisert på: www.teknologiradet.no

Innholdsfortegnelse	Side
Forord	
Sammendrag	1
Kapittel 1 Om programvarepolitikk	5
1.1 <i>Hvorfor programvarepolitikk?</i>	5
1.1.1 <i>Programvare som portvakt til informasjon</i>	5
1.1.2 <i>Programvaremarkedet er et spesielt marked</i>	6
1.1.3 <i>Det offentlige er ikke en vanlig kunde</i>	6
1.2 <i>Grunnlaget for en god programvarepolitikk</i>	7
1.3 <i>Oppsummering</i>	8
1.4 <i>Rapportens struktur</i>	8
Kapittel 2 Spesielle forhold ved programvaremarkedet	9
2.1 <i>Markedsdominans</i>	9
2.1.1 <i>Nettverkseffekter</i>	10
2.1.2 <i>Monopolisering</i>	12
2.2 <i>Historisk riss: Fra vertikal integrasjon til lagdeling</i>	13
2.3 <i>Konkurransemyndighetene griper inn</i>	15
2.3.1 <i>Amerikanske myndigheters inngripen mot Microsoft</i>	15
2.3.2 <i>EU-kommisjonens tiltak mot Microsoft</i>	16
2.4 <i>Oppsummering: markedsdynamikk</i>	16
Kapittel 3 Programvarepolitikk i Norge	18
3.1 <i>Innkjøpspolitikk</i>	18
3.2 <i>Ny norsk konkurranselov</i>	19
3.2.1 <i>Økt fokus på markedssvikt</i>	19
3.3 <i>Norsk politikk i forhold til åpen kildekode</i>	20
3.3.1 <i>Nærings- og handelsdepartementet</i>	20
3.3.2 <i>Arbeids- og administrasjonsdepartementet – modernisering med IKT</i>	21
3.3.3 <i>Utdannings- og forskningsdepartementet og Skolelinux</i>	21
3.3.4 <i>Barne- og familiedepartementet – forbrukerpolitikk</i>	21
3.4 <i>Norsk politikk på åpne standarder</i>	21
3.4.1 <i>Moderniseringsdepartementet</i>	22
3.4.2 <i>Miljøverndepartementet</i>	22
3.4.3 <i>Justissektoen</i>	22
3.5 <i>Status i norske kommuner</i>	23
3.6 <i>Øvrig offentlig arbeid med standardisering</i>	23
3.6.1 <i>Altinnprosjektet</i>	24
3.7 <i>Bruk av åpne standarder i norsk offentlig kommunikasjon</i>	24
3.8 <i>Lukkede standarder i åpne medier</i>	25
3.9 <i>Oppsummering: programvarepolitikk blir stadig viktigere</i>	25
Kapittel 4 Teknologiske strategier for et åpnere samfunn	26
4.1 <i>Åpne standarder</i>	26
4.1.1 <i>Historikk</i>	27
4.1.2 <i>Forslag til en god definisjon av åpne standarder</i>	28

4.1.3	<i>Kan åpne standarder hemme innovasjon?</i>	30
4.1.4	<i>Industristandarder</i>	33
4.2	<i>Åpen kildekode</i>	33
4.2.1	<i>Hva bør være definisjonen på åpen kildekode?</i>	35
4.2.2	<i>Åpen kildekode og innovasjon</i>	35
4.3	<i>Programvarepatenter</i>	39
4.3.1	<i>Argumenter for programvarepatenter</i>	39
4.3.2	<i>Argumenter mot programvarepatenter</i>	40
4.3.3	<i>Virkningen av programvarepatenter</i>	40
4.4	<i>Tekniske sperrer</i>	42
4.5	<i>Oppsummering: teknologivalg avgjør åpenhet</i>	43
Kapittel 5	<i>Offentlige strategier for et åpnere samfunn</i>	45
5.1	<i>Fire hovedstrategier for det offentlige</i>	45
5.2	<i>La markedet råde</i>	46
5.3	<i>Aktiv konkurransepolitikk</i>	47
5.4	<i>Preferansepolitikk</i>	47
5.4.1	<i>Preferansepolitikk for åpne standarder</i>	47
5.4.2	<i>Hva kan Norge gjøre for å fremme åpne standarder?</i>	49
5.4.3	<i>Preferansepolitikk for åpen kildekode</i>	53
5.5	<i>Pilotprosjekter med åpen kildekode</i>	55
5.6	<i>Oppsummering – tilgjengelige strategier i programvarepolitikken</i>	57
Kapittel 6	<i>Modernisering – en åpnere forvaltning med IKT</i>	59
6.1	<i>IT i offentlig forvaltning</i>	59
6.2	<i>Digital lovanvendelse</i>	59
6.2.1	<i>Automatisering av tolkning og vilkårsprøving</i>	60
6.2.2	<i>Innhenting og utveksling av saksopplysninger</i>	60
6.2.3	<i>Behov for klare begrepsdefinisjoner</i>	61
6.2.4	<i>Klagerett ved digital saksbehandling</i>	62
6.2.5	<i>Rettslige beslutninger i det skjulte</i>	62
6.2.6	<i>Omfattende konsekvenser ved medhold i klagesaker</i>	63
6.3	<i>Offentlighetsprinsippet og IKT-politikken</i>	64
6.3.1	<i>Ny offentlighetslov og innsyn i digitale saksopplysninger</i>	64
6.3.2	<i>Effektivisering av innsynsrettighetene</i>	65
6.3.3	<i>Dokumentasjon av rettslige systemavgjørelser</i>	66
6.3.4	<i>Oversikt og rettigheter i en digital forvaltning</i>	66
6.4	<i>Oppsummering – IKT i forvaltningen: lov, formater og konkurranse</i>	67
Kapittel 7	<i>Kunnskapsutvikling i Norge og internasjonalt</i>	68
7.1	<i>Teknologisk og samfunnsmessig endring – trender i skolen</i>	68
7.2	<i>IKT i skolen</i>	69
7.2.1	<i>Digital kompetanse</i>	70
7.2.2	<i>En inkluderende skole</i>	70
7.2.3	<i>Programvare i grunnskolen</i>	71
7.3	<i>Tidligere erfaringer med IKT i skolen</i>	71
7.3.1	<i>Lærdommer av offentlige "IT-skandaler"</i>	74
7.4	<i>Er Skolelinux et reelt alternativ?</i>	75
7.4.1	<i>Mulige hindre for utbredelse av Skolelinux</i>	75

7.4.2	<i>Pedagogisk programvare</i>	76
7.4.3	<i>Gjenbruk av eldre datamaskiner ved hjelp av Skolelinux</i>	77
7.4.4	<i>Utfordringene for Linux i skolen</i>	77
7.5	<i>Programvare i forskning og høyere utdanning</i>	77
7.6	<i>Bistandspolitik, programvare og åpenhet</i>	79
7.6.1	<i>Lokal tilpasning av programvare</i>	79
7.6.2	<i>Samarbeid på tvers av landegrenser</i>	80
7.7	<i>Oppsummering: IKT i skolen – en samfunnsmessig utfordring</i>	81
	Ordforklaringer	83
	Forkortelser	85
	Vedlegg 1 – Om Prosjektet	87
	Vedlegg 2 – Standardiseringsorganisasjoner	90
	Vedlegg 3 – Ulike typer standarder	91

Forord

Teknologirådet er et uavhengig, rådgivende organ som skal stimulere til debatt om de muligheter og konsekvenser som ny teknologi skaper for samfunnet og for det enkelte menneske. Resultatene av vårt arbeid skal formidles til Stortinget, øvrige myndigheter og samfunnet generelt.

Våren 2003 nedsatte Teknologirådet en ekspertgruppe for å vurdere demokrati- og konkurransespørsmål i informasjonssamfunnet, særlig knyttet til programvare. Gjennom å sette fokus på programvarepolitikk ønsker vi å signalisere at dette er en viktig, men lite diskutert del av offentlig IKT-politikk.

Det er store utfordringer knyttet til at IT trer inn i kommunikasjonen mellom det offentlige, borgerne og næringslivet. Vi har derfor valgt å stort sett bruke begrepet IKT (informasjons- og kommunikasjonsteknologi) i denne rapporten. For ordens skyld vil vi påpeke at vi ikke dekker alle relevante programvarepolitiske områder. For eksempel finnes det viktige momenter knyttet til sikkerhet som vi har latt ligge denne gang.

Teknologirådet jobber prosjektbasert. Det vil si at vi involverer nye kompetansepersoner for hvert nytt tema vi tar for oss. Bak denne rapporten står en ekspertgruppe med bred kompetanse. I tillegg har vi involvert andre viktige aktører, eksperter og berørte parter i en egen høring. Ekspertgruppen for programvarepolitikk har hatt følgende medlemmer:

Professor Dag Wiese Schartum, Institutt for rettsinformatikk ved Universitetet i Oslo,
Førsteamanuensis Mikael Snaprud ved Høgskolen i Agder,
Sjeføkonom Lars Sørgard i Konkurransetilsynet (professor ved NHH frem til 01.09.04),
Teknologidirektør Håkon Wium Lie fra Opera Software,
Forsker Tanja Storsul ved Universitetet i Oslo og
Daglig leder Grethe Helene Viksaas fra Basefarm as;
Professor Einar J. Aas, Teknologirådet og Inst. for elektronikk og telekom. ved NTNU og
Professor Magne Espedal, Teknologirådet og Matematisk inst. ved Universitetet i Bergen.

Prosjektleder i hoveddelen av prosjektforløpet har vært dr. polit. Trond Arne Undheim fra Teknologirådets sekretariat. Siviling. Christine Hafskjold har vært prosjektleder fra 01.09.04.

Denne rapporten inngår i Teknologirådets arbeid med informasjonsteknologi. Vi har også en stor satsing på personvern og vil følge opp med flere prosjekter innen informasjonsteknologi i tiden fremover. Alle dokumenter kan lastes ned fra www.teknologiradet.no.

På vegne av Teknologirådet vil jeg takke medlemmene av ekspertgruppen for solid arbeid og fruktbare diskusjoner. Vårt håp er at rapporten kan stimulere til en aktiv og åpen diskusjon om programvarepolitikk. Vi tror at tiden er overmoden for det.

Tore Tennøe
Sekretariatsleder
Teknologirådet

Sammendrag

IKT og programvare spiller en stadig viktigere rolle i samfunnslivet. I Regjeringens program for modernisering av offentlig sektor er IKT det helt sentrale verktøyet. Utnæring av dokumenter og tjenester på nettet blir stadig vanligere; e-post, nettbank og e-handel er allerede utbredt, og gjennom Personlig internettportal vil også offentlig informasjon og saksbehandling få et digitalt grensesnitt mot befolkningen. Det er derfor på høy tid at myndighetene tar en aktiv holdning til programvare.

Det er spesielt tre forhold som gjør at det er viktig med en bevisst programvarepolitikk:

- Valg av programvare handler om kontroll av informasjonsstrømmene i hverdagen. Mange nettbanks, kringkastere som NRK og TV2, og også regjeringens direkte sendte pressekonferanser krever i dag at brukerne har programvare fra Microsoft for å få enkel tilgang til tjenestene. Dersom informasjon låses til bestemte formater, blir porten til informasjonssamfunnet smal, og den demokratiske tilgangen til informasjon svekkes. Vi kan si at programvaren fungerer som *portvakt* til informasjon.
- Programvaremarkedet er et spesielt marked, som er preget av det vi kaller *nettverks-effekter*. Kort fortalt går dette ut på at markedet favoriserer det produktet som de fleste bruker. Dette medfører at det kan oppstå tilnærmede monopolsituasjoner i de ulike segmentene i programvarebransjen.
- Det offentlige har en spesiell posisjon som viktigste kunde og regulerende myndighet i programvaremarkedet. Fordi offentlige valg legger premissene for alle borgerne i informasjonssamfunnet, har det offentlige et spesielt ansvar for de valgene som tas.

I rapporten peker vi på at en god programvarepolitikk bør være demokratisk, effektiv og fremtidsrettet. To teknologiske strategier som på ulike måter kan fremme disse hensynene, er åpne standarder og åpen kildekode. Det er viktig å skille mellom disse to strategiene.

Åpne standarder

Åpne standarder for programvare er regler og retningslinjer for lagring og utveksling av data som kan fremme både demokratisk tilgjengelighet og effektiv konkurranse. Disse retningslinjene bør være tilgjengelige for liten eller ingen kostnad, slik at alle har en reell mulighet til å få tilgang til dem. Det er viktig å presisere at selv om et selskap følger åpne standarder, er det ikke nødt til å avsløre alle spesifikasjonene i produktet. Åpne standarder gir rom for forretningshemmeligheter, samtidig som man sikrer at data og informasjon kan utveksles uavhengig av programvareleverandør.

Et eksempel på en standard de fleste har et forhold til, er GSM-standard for mobiltelefoni. Fordi vi har en standard, er det mulig for mange ulike leverandører å produsere telefoner som kan brukes i alle GSM-nett. Forbrukerne har mulighet til både å bytte mobiltelefon og operatør når de måtte ønske det, og det er også mulig å ta med telefonen til de fleste andre land og ha glede av den der. HTML er en åpen standard for dokumenter på web som kan sies å ha mye av æren for at internett og world wide web har fått den utbredelsen det har i dag. Dersom man hadde hatt en tilsvarende standard for tekstbehandlingsdokumenter, ville forbrukerne kunne bytte både operativsystem og tekstbehandlingsprogram uten å måtte tenke på om det ville bli mulig å lese eller utveksle dokumenter problemfritt.

Det er også viktig at en åpen standard forvaltes av ikke-kommersielle organisasjoner eller organisasjoner med et internasjonalt mandat. Dersom en kommersiell organisasjon skulle forvalte en standard, kan man tenke seg at standarden kunne endres på kort varsel, på en slik måte at det ga en konkurransefordel for den organisasjonen som forvalter standarden.

IKT-politikken i Norge har i liten grad omhandlet programvare eller områder som åpne standarder og åpen kildekode. I praksis er det lukkede formater som dominerer offentlige systemer og offentlig datautveksling. I andre land i Europa føres det en mer aktiv politikk på dette feltet, blant annet har både Danmark og Storbritannia en klar politikk for å fremme åpne standarder i det offentlige.

Teknologirådets ekspertgruppe mener åpne standarder har en viktig demokratisk funksjon, fordi de fremmer ytringsfrihet og tilgjengelighet gjennom fri utveksling av informasjon. Gruppen anbefaler at norske myndigheter fører en aktiv preferansepolitikk for åpne standarder, og foreslår blant annet følgende konkrete tiltak:

- Gjør offentlige dokumenter som utveksles eller arkiveres tilgjengelige på åpne formater (for eksempel HTML, XML eller PDF), i tillegg til evt. lukkede formater.
- Still krav til at allmennkringkastere skal bruke åpne standarder.
- Vurder om all programvare det offentlige kjøper inn skal følge åpne standarder. Stat og kommune bør etterspørre programvare som støtter åpne standarder, samt vurdere åpen kildekode ved offentlige anbud. Programkode betalt av og utviklet for det offentlige bør også eies av det offentlige.
- Vurder aktive forhandlinger med enkelte større programvareleverandører for å få dem til å åpne sine formater.
- Øk bestillerkompetansen i offentlig sektor. Valg av programvare og tildeling av utviklingsoppdrag har en avgjørende betydning for bruk av standarder, og det er viktig å ha oversikt over hvilke valgmuligheter som finnes og konsekvenser av valg som tas.

Åpen kildekode

Åpen kildekode kjennetegnes ved at den er åpen for innsyn av alle, kan modifiseres av alle, kan gjenbrukes i annen programvare og fritt distribueres til andre. Åpen kildekode må sies å være en fremtidsrettet strategi, i det den stimulerer en innovativ kunnskapskultur som innebærer deling av kunnskap og desentralisert utviklingskompetanse. Slik deling må kunne ses på som et langsiktig samfunnsgode.

Teknologirådets ekspertgruppe anbefaler en noe forsiktigere politikk for åpen kildekode enn for åpne standarder. En viss stimulering av leverandører av programvare med åpen kildekode, som konkurrenter til rådende markedsledere, kan ha en gunstig effekt både på innovasjonsarbeidet og prispolitikken hos de dominerende aktørene. Det anbefales derfor at Norge følger Storbritannias og Danmarks eksempel og iverksetter *statlige pilotprosjekter* på programvare basert på åpen kildekode. Slike prosjekter kan gi verdifulle erfaringer, særlig knyttet til levetidskostnader, brukerbehov og funksjonalitet.

I rapporten ser vi også på to andre områder knyttet til programvare: *Programvarepatenter og tekniske sperrer*.

Teknologirådets ekspertgruppe mener det er mange grunner til at programvare ikke bør kunne patenteres. Det pekes blant annet på at programvare er annerledes enn andre felt hvor patentering er mulig: Det er mindre ressurskrevende å komme fram til nye produkter enn for eksempel i farmasøytisk industri, og etter patentets utløpstid på 20 år vil de fleste programvareprodukter være utdatert.

Patenter er ment å tjene både oppfinner og samfunnet for øvrig. Ekspertgruppen mener at programvarepatenter synes å ha få positive virkninger for samfunnet. Det finnes derimot flere viktige negative sider: Programvarepatenter er en trussel mot åpen kildekode, åpne standarder og fri konkurranse.

Tekniske sperrer kan brukes til å frata forbrukere rettigheter de har i følge norsk lov. Teknologirådets ekspertgruppe mener det er viktig å opprettholde leseretten til digitale data man har rettmessig tilgang til. Innføring av InfoSoc-direktivet i Norge bør derfor revurderes.

I rapporten pekes det også på et annet hensyn som må ligge til grunn for en demokratisk programvarepolitikk, nemlig *gjennomsiktighet*. Utfordringene knytte til dette kan oppsummeres som følger:

- Å sikre fortsatt åpenhet når programmer tar over lovanvendelsen.
- Å revidere offentlighetslov og forvaltningslov og forskrifter med IKT for øye.
- Å sikre at saksbehandlingsrutiner i den digitale forvaltningen blir oversiktlige og tilstrekkelig beskrevet til at brukerne kan forstå sin rolle i den samlede rutinen.
- Å legge betydelig større vekt på å lage digitale saksbehandlingsrutiner som sikrer etterlevelse av lovbestemte krav til utøvelse av forvaltningsmyndighet når det gjelder brukeres rettigheter og forvaltningens plikter.

Til slutt i rapporten behandles programvare i undervisning og forskning. Programvare i skolen bør bidra til å gi elevene innsikt i informasjonsteknologi slik at de kan bli aktive deltakere i informasjonssamfunnet, både som brukere og utviklere. I tillegg bør programvare i undervisning være så nøytral som mulig i forhold til å binde den enkelte utdanningsinstitusjon og elev opp til bestemte leverandører.

Skolelinux er et prosjekt med norsk opprinnelse, basert på åpen programvare. Teknologirådets ekspertgruppe mener at mer Linux i skoleverket kan bidra til at norske skolebarn blir mer fleksible, innovative og kompetente databrukere, og anbefaler derfor at støtten til Skolelinux utvides.

Utviklingsland med en beskjedne IKT-industri og -infrastruktur er spesielt sårbare for å bli låst til bestemte leverandører. Norge bør engasjere seg i bistands- og samarbeidsprosjekter med åpen kildekode. Basert på Skolelinux og erfaringer fra FAIR og FN-organet GRID i Arendal, kan Norge ta initiativ til å samarbeide med fattige land basert på felles utvikling av åpen kildekode.

Merknad fra ekspertgruppens mindretall (Lars Sjørgard)

I arbeidet med rapporten oppsto det en dissens i ekspertgruppen. Mindretallet (Lars Sjørgard) er uenig i flertallets hovedsynspunkter, og tar forbehold om de deler av innholdet som leder fram til flertallets konklusjoner (dette gjelder spesielt 0 og Kapittel 5).¹

Sjørgard mener at sterke føringer fra det offentlige i forhold til bruk av åpne standarder kan ha uheldige effekter. For det første kan det være problematisk på kort sikt for dagens brukere, rett og slett fordi en svært stor andel av dagens programvare i offentlig sektor ikke følger åpne standarder. For det andre kan det være uheldig på lang sikt, fordi det kan hindre en fremtidig bruk av programvare som kan vise seg å ha mange gode egenskaper, men som ikke er basert på åpne standarder. Det advares derfor mot å føre en streng preferansepolitikk hva angår åpne standarder.

Selv om Sjørgard finner grunn til å advare mot å innføre en streng preferansepolitikk for åpne standarder, er han enig i en rekke av de forslag som flertallet fremmer:

- Det kan være gode grunner for at det offentlige presser på for å få frigitt sentrale kildekoder for visse programvarer.
- Det kan også være grunn til å støtte tiltak som fremmer arbeidet med utvikling av åpne standarder.
- Oppsigelsen av statens innkjøpsavtale med Microsoft er et eksempel på et tiltak som kan stimulere til nytenkning i ulike deler av offentlig sektor, der en tenker i retning av alternative løsninger som ikke innbefatter Microsoft.
- Prosjektet Skolelinux er et annet positivt tiltak, som bidrar til at de kommende generasjoner får større kunnskap om alternative løsninger, og det i seg selv bereder grunnen for at alternativer til Microsoft vokser frem.
- Alle offentlige dokumenter som legges ut på nettet skal være tilgjengelig i åpne formater (som PDF, HTML og XML). Allmennkringkastere som NRK og TV2, samt andre nettmedier, bør være lett tilgjengelig i åpne multimedieformater (for eksempel MPEG eller MP3). Det gir synlighet for at det finnes alternative løsninger, og gjør alternative løsninger mer levedyktige.

¹For en helhetlig og samlet fremstilling av mindretallets syn, se Sjørgard, Lars (2004): *Markedsdynamikk i dataindustrien*, SNF arbeidsnotat 38/04 tilgjengelig på ftp://ftp.snf.no/Arbeidsnotat/2004/A38_04.pdf



Kapittel 1 Om programvarepolitikk

Programvare er en teknologi som kan brukes til mye og som påvirker samfunnet i stadig sterkere grad. Det er nå over 40 år siden IT ble en vesentlig disiplin innen norsk forskning. IT-bedrifter, spesielt Norsk Data, men også Kongsberg Våpenfabrikk og Tandberg Data, begynte å utvikle datamaskiner og programvare for salg. Utover 1970-tallet ble IT mye mer enn et forskningsverktøy. Fra 1980-tallet og fremover ble IT tatt i bruk av næringslivet, og på 1990-tallet kom PC-en inn i norske hjem. På slutten av 1990-tallet kom internett for full fart. Internett har allerede endret våre oppfatninger og rutiner rundt arbeid, informasjon og kunnskapsflyt, og IKT påvirker i dag hele samfunnsutviklingen.

1.1 Hvorfor programvarepolitikk?

Hvorfor skal myndighetene involvere seg sterkere i programvare enn i andre typer varer? Programvare er en relativt ny arena for politikk. Særlig den omfattende utvekslingen av dokumenter og tjenester over nettet har skapt behov for at politiske myndigheter tar en aktiv holdning til programvare.

1.1.1 Programvare som portvakt til informasjon

Valg av programvare handler blant annet om kontroll av informasjonsstrømmene i hverdagen. En viktig forutsetning for bruk av databaserte informasjons- og kommunikasjons-systemer er at programvaren på de ulike nivåene i kommunikasjonsprosessen kan lese og bearbeide informasjonen.

For 20 år siden var programvare og data uløselig knyttet til hverandre og til den enkelte leverandør: Norsk Data hadde sine dataformater og programmer og IBM hadde sine. Informasjon kunne ikke uten videre utveksles mellom leverandørenes systemer, og lite digital informasjon fra denne tidsperioden er tilgjengelig i dag. Dersom offentlig elektronisk informasjon skulle vært lagret på bare ett format, ville man i praksis ha tvunget hele befolkningen til å kjøpe programvare fra den aktuelle leverandøren, eller svekket informasjons-tilgangen for mange.

Dette viser hvordan programvare kan fungere som portvakt i elektroniske informasjons- og kommunikasjonsprosesser. Dette var ikke bare tilfelle i den digitale teknologiens barndom, men er i høyeste grad fortsatt aktuelt. Vi skal her ta for oss to aspekter ved slike portvakter, nemlig betydningen de kan ha for tilgangen til informasjon og tjenester, og betydningen de kan ha i forhold til hvem som kan gjøre informasjon og tjenester tilgjengelige.

Hindringer for å motta informasjon

De som ønsker å hente ned TV-programmer fra NRK eller TV2 over web må i dag ha programvare fra en bestemt leverandør for å gjøre dette². Mange nettbanker forutsetter dessuten at man benytter en bestemt nettleser (som regel Microsofts Internet Explorer). For de som bruker andre nettlesere er dette irriterende og kan føre til at man ikke benytter nettbank eller ser TV på nettet, eventuelt at man blir tvunget til å skifte nettleser (eller bank).

Dersom offentlig informasjon kun gjøres tilgjengelig gjennom formater som forutsetter en bestemt programvare, er dette mer problematisk. Hvis informasjon fra skatteetaten, trygdeetaten eller kommunen kun kan lastes ned i Microsofts doc-format, avstenges de som ikke har Microsoft Word fra å få denne informasjonen. Dette er alvorlig, men det finnes eksempler på dette i en del norske kommuner.

Slik ser vi at de programvarevalgene som gjøres får betydning for informasjonstilgang og deltakelse i informasjonssamfunnet. Dersom informasjon låses til bestemte formater, blir porten inn til informasjonssamfunnet smal og den demokratiske tilgangen til informasjon svekkes.

Hindringer for å sende informasjon

Programvare kan også fungere som portvakt for den som skal sende informasjon. Dette gjelder for eksempel når forlag, bedrifter eller offentlige myndigheter ønsker informasjon oversendt i bestemte formater (ofte Microsofts doc- eller xls-formater). I slike tilfeller er det kun de som har tilgang til slik programvare som kan meddele seg elektronisk. Dette er igjen særlig problematisk for folks deltakelsesmuligheter dersom det offentlige stiller slike krav.

Problemstillingen er også aktuell for profesjonelle tjeneste- og innholdsleverandører som vil nå ut til en større del av befolkningen. Mange leverer tjenester og innhold over flere plattformformater som web, mobile- og håndholdte terminaler, fjernsyn osv. Dersom disse plattformformene benytter lukkede formater, og kanskje flere inkompatible lukkede formater, vil tilpasning bli svært krevende og kostbart. Dette gjelder for eksempel markedet for digitale fjernsynsdekodere.

1.1.2 Programvaremarkedet er et spesielt marked

Vi har nettopp sett at programvare ikke er en tradisjonell vare. Markedet denne varen opptrer i, er heller ikke et tradisjonelt marked: Det er preget av *nettverkseffekter*, en mekanisme vi skal gå grundig inn på i Kapittel 2.

Kort fortalt går nettverkseffekten ut på at markedet favoriserer det produktet flest bruker. Som kunde har man ikke mye glede av et tekstbehandlingsprogram som ikke kan utveksle dokumenter med andre. Sannsynligheten er derfor stor for at de fleste vil kjøpe det programmet som er mest utbredt. På grunn av denne effekten oppstår det lett tilnærmede monopolsituasjoner i de ulike segmentene i programvarebransjen.

1.1.3 Det offentlige er ikke en vanlig kunde

Det offentlige har mange roller i programvarepolitikken – som innkjøper, som formidler av kunnskap og som leverandør av offentlige tjenester. Fordi offentlige valg legger premissene

²NRK påpeker imidlertid at det mulig å bruke Linux med en viss egeninnsats:
<http://www.nrk.no/informasjon/hjelp/3157548.html>

for alle borgere i det norske informasjonssamfunnet, har det offentlige et spesielt ansvar i de valgene som tas.

Fordi man har behov for en effektiv forvaltning, settes det stadig i gang store IKT-prosjekter i offentlig regi. I tillegg er kommuner og etater storforbrukere av kontorstøttesystemer. Det offentlige er med andre ord en aktør med enorm innkjøpsmakt, noe som gjør det mulig å påvirke utviklingen i programvaremarkedet.

Moderniseringsdepartementet har påbegynt oppgaven med å gjøre offentlige tjenester og offentlig informasjon mer tilgjengelig for den enkelte borger. IKT, og standardisering av systemer og grensesnitt, er et viktig virkemiddel for å greie dette. Å stille krav til standardisering er en viktig del av en aktiv programvarepolitikk.

1.2 Grunnlaget for en god programvarepolitikk

Vi legger i denne rapporten til grunn følgende hovedhensyn for en god programvarepolitikk: Demokrati, effektivitet og fremtidsrettethet.

For det første bør norsk programvarepolitikk være *demokratisk*. Kommunikasjonsteknologier med programvare som plattform er i ferd med å bli et nødvendighetsgode, all den tid stadig mer offentlig informasjon og saksbehandling blir papirløs. Det må derfor legges vekt på følgende punkter:

- *Ytringsfrihet*: Det er viktig å sikre at alle har gode ytringsmuligheter også i den digitaliserte offentligheten.
- *Tilgjengelighet og brukervennlighet*: Den enkeltes kommunikasjon med offentlige myndigheter krever at teknologier er universelt utformet og legger til rette for en kontinuerlig dialog med brukere, borgere og kunder av det offentlige.
- *Gjennomsiktighet*: Hvilke lover og regler ligger til grunn ved automatisert saksbehandling og hvordan endres innsynet i disse ved innføring av en digital forvaltning? Rettighetsperspektivet må stå sentralt for enhver reform av offentlig forvaltning.
- *Demokratisk kontroll* over teknologier som er viktige for samfunnet. Åpenhet og innsyn i teknologier er et krav som bør kunne veie tungt.

For det andre bør programvarepolitikken ivareta hensyn til *effektivisering*

- *For brukerne*; ved enklere tilgang til offentlige tjenester.
- *For myndighetene*; som en innsatsfaktor til å effektivisere offentlig forvaltning gjennom samarbeid, koordinering, informasjons- og ressursdeling.
- *For næringslivet*; ved at staten sikrer stabile rammevilkår og standardisering slik at systemer, prosesser og mennesker kommuniserer optimalt (interoperabilitet). Dette krever arbeid med IKT-standardisering.

For det tredje bør programvarepolitikken være *fremtidsrettet*:

- Programvare har blitt en viktig teknologi for borgere, myndigheter og næringsliv og må derfor bidra til forutsigbare vilkår for disse gruppene.
- Den raske utviklingen innen informasjons- og kommunikasjonsteknologi fordrer et aktivt arbeid med kunnskapsutvikling innen skole, utdanning og forskning. Åpenhet er en sentral kvalitet dersom vi ønsker en aktiv utvikling av programvare også i Norge.
- *Innovasjon*: Det offentlige må legge til rette både slik at eksisterende næringsliv kan innovere, og slik at nyskaping skjer både på tvers av bransjer og i nye foretak.

Programvare er derfor en flersidig politisk utfordring som berører konkurransepolitikk, statlig innkjøps- og standardiseringspolitikk samt kunnskaps- og utdanningspolitikk.

1.3 Oppsummering

I dette kapitlet har vi pekt spesielt på tre forhold som gjør det viktig med en bevisst programvarepolitikk: Programvare kan fungere som portvakt til informasjon, markeds-mekanismene som virker i programvaremarkedet er spesielle, og det offentlige har en spesiell posisjon som viktigste kunde og regulerende myndighet i programvaremarkedet.

Vi har også presentert 3 hovedhensyn for en god programvarepolitikk: *Demokrati, effektivitet og fremtidsrettethet*.

1.4 Rapportens struktur

Videre i rapporten vil vi se mer på programvarepolitikk i Norge og andre land, på mulige teknologiske strategier for å tilfredsstille de ovennevnte hensynene, og på mulige politiske strategier for en offensiv programvarepolitikk.

I Kapittel 2 utdyper vi hvordan programvaremarkedet fungerer, og går nærmere inn på de spesielle mekanismene som gjelder i dette markedet.

Kapittel 3 omhandler dagens programvarepolitikk i Norge.

I 0 definerer vi *åpne standarder*, *åpen kildekode* og *programvarepatenter*, og diskuterer hvordan disse teknologiske strategiene tilfredsstiller hensynene til demokrati, effektivitet og fremtidsrettethet, herunder hvordan de påvirker innovasjon.

I Kapittel 5 presenterer vi fire hovedstrategier det offentlige kan følge når det gjelder åpne standarder og åpen kildekode, og vi gir våre anbefalinger til valg av strategi.

Kapittel 6 fokuserer spesielt på et aspekt ved hensynet til demokratiet – nemlig *gjennomsiktighet* i de systemer som brukes til automatisert saksbehandling og borgernes "selvbetjening" mot offentlig sektor.

Vi har pekt på det offentlige som en viktig kunde og regulerende myndighet. I tillegg har det offentlige en viktig rolle som ansvarlig for kunnskapsutvikling, og i bistand til andre land. Kapittel 7 bruker vi derfor til å se nærmere på områder knyttet til dette, med spesielt fokus på IKT i skolen.



Kapittel 2 Spesielle forhold ved programvaremarkedet

Programvareindustrien er en næring preget av kontinuerlig produktutvikling, med til dels dramatiske skift fra en teknologi til en annen. Teknologiske skift innenfor informasjons-teknologien har i løpet av de siste tiårene gitt flere eksempler på at en dominerende aktør har mistet sitt hegemoni og at andre har overtatt. IBM dominerte dataindustrien til langt ut på 90- tallet, da Windows for alvor slo igjennom som operativsystem, og Microsoft fikk en dominerende posisjon.

I dette kapitlet skal vi se nærmere på de markedsmekanismene som gjør at programvare-markedet har så lett for å bli dominert at noen få, store aktører til enhver tid.

2.1 Markedsdominans

Markedsdominans kan ha ulike årsaker. Tradisjonelt har det blitt forklart med forhold på tilbudssiden. Det dreier seg om både etableringshindringer på grunn av eiendomsrett, store faste kostnader, strategisk tilpasning som sikrer dominans, kostnadsdrivende konkurranse mellom få aktører – eller om nettverkseffekter.

Eksempler på ulike mekanismer som kan gi markedsdominans

DeBeers kontroll med diamantressurser

En aktør kan ha kontroll over viktige ressurser som nyttes i produksjonen av en aktuell vare. Sørafrikanske De Beers er et eksempel på et slikt selskap. Gjennom eiendomsrett til noen av de største forekomstene av diamanter har selskapet oppnådd en dominerende posisjon i det globale markedet for diamanter.

Jernbanenettverk – kostbart å skifte system

Høye faste kostnader kan også forklare dominans. Hvis det er slik at en må bygge et stort produksjonsanlegg for i det hele tatt å kunne produsere, vil det være mulig å få dekket disse faste kostnadene kun dersom det er et fåtall tilbydere i markedet. Et eksempel er jernbane-nettverk, der det av naturlige grunner er sjelden vi ser at det bygges parallelle nettverk.

Kelloggs – stort antall frokostblandinger

Det kan også være tale om at en aktør foretar en strategisk tilpasning som sikrer dominans. For eksempel har Kelloggs i USA valgt å utvikle et stort antall varianter av frokostblandinger, noe som i neste omgang gjør det vanskelig for andre å finne noen nisjer de kan etablere seg i uten å måtte konkurrere direkte med Kelloggs.

Farmasøytisk industri – høye FoU-kostnader

Alternativt kan det være en kostnadsdrivende konkurranse mellom et fåtall aktører. Innen farmasøytisk industri er FoU helt sentralt og betraktes som en fast kostnad. Konkurransen mellom selskapene bidrar til høye FoU-kostnader, med det resultat at kun et fåtall aktører er i stand til å få dekket disse kostnadene.

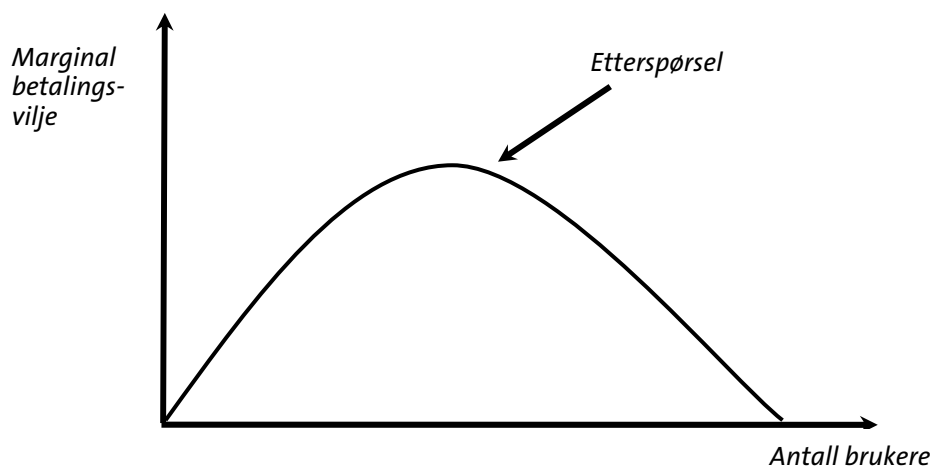
2.1.1 Nettverkseffekter

Nettverkseffekter kan sannsynligvis også forklare mye av Microsofts globale dominans. Den grunnleggende mekanismen forbundet med en nettverkseffekt er at en ny bruker har genuin nytte av at det er andre brukere. Dette er utvilsomt relevant innen programvareindustrien. Nytten av å bruke et tekstbehandlingssystem er knyttet opp mot hvor mange andre som benytter det samme systemet. Hvis samtlige som samarbeider og utveksler dokumenter benytter Microsofts doc-format (Word), kan det skape problemer for den som benytter et annet tekstbehandlingsprogram dersom disse programmene ikke kan forholde seg til samme standard for informasjonsutveksling.

Eksempel: Telefoni

For å illustrere hva vi mener, la oss betrakte tjenesten telefoni. Hvem hadde interesse av å kjøpe den første telefon i verden? Kan et slikt produkt ha noen som helst verdi så lenge ingen andre har et slikt produkt? Verdien av å ha telefon er bestemt av hvor mange andre som har telefon. Hvis ingen andre har telefon, vil min verdsettelse av å ha en telefon være lik null. På den annen side vil det ifølge tradisjonell markedsteori finnes et metningspunkt for hvor mange som er villige til å kjøpe et produkt selv om prisen for produktet etter hvert blir svært lav.

En etterspørselskurve som beskriver et slikt fenomen beskrives gjerne som pervers sammenlignet med hva en opplever i tradisjonelle markeder. Se Figur 1.³



Figur 1. Etterspørsel i et marked med nettverkseffekter

Den horisontale aksen angir antall brukere. Den vertikale aksen angir betalingsviljen til en ny bruker, det vil si den marginale betalingsviljen i markedet. Som forklart over, vil betalingsviljen for den første brukeren være lik null. Men straks det kommer til andre brukere, vil nytten for hver enkelt bruker øke. Dette er illustrert i figuren ved at den

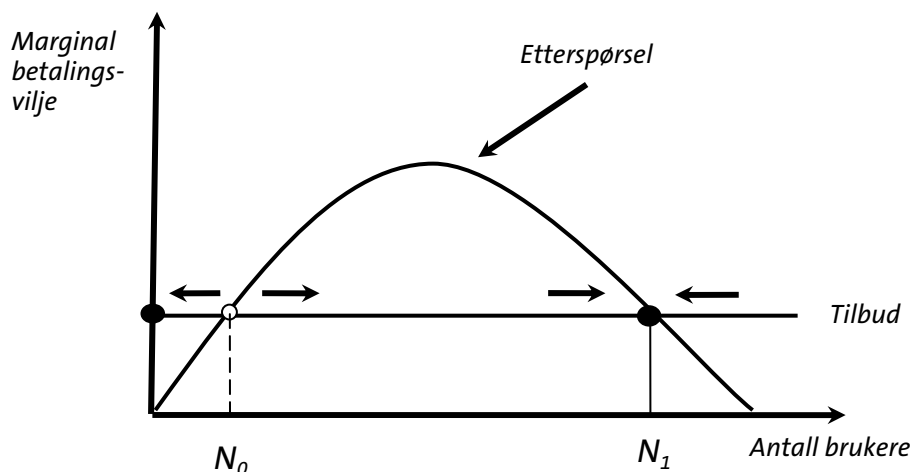
³Resonnementet i forbindelse med figur 1 og 2 er hentet fra Shapiro, C og H. Varian (1998), *Network Effects*, <http://www.inforules.com/>

marginale betalingsviljen øker i form av antall brukere, det vil si at kurven som viser marginal betalingsvilje er stigende.

Når antallet brukere kommer over et visst nivå, vil det gradvis oppstå en metning i markedet, det vil si at hver ny bruker har lavere marginal betalingsvilje. Da vil en ha en mer normal form for utvikling i markedet, ved at for hver ny bruker vil betalingsviljen være fallende.

Kurven som viser den marginale betalingsviljen er i figuren betegnet etterspørselskurven. Første del av etterspørselskurven er som nevnt stigende. Sammenlignet med en tradisjonell etterspørselskurve, karakteriseres dette som en "pervers" form.

Hva vil skje i et marked med en slik etterspørsel? I Figur 2 har vi angitt en tilbudskurve, som vi antar er horisontal. Da ser vi at det er tre mulige likevektspunkter.



Figur 2. Likevekt i et marked med nettverkseffekter

En mulighet er at det rett og slett ikke oppstår noe marked for dette produktet. Det vil i så fall si at $N = 0$ (ingen brukere). Men det er i tillegg to mulige tilpasninger hvor etterspørsel er lik tilbud. For det første er det en situasjon med det vi kan karakterisere som lav forventning, hvor antallet brukere er lik N_0 . Da er etterspørselen relativt begrenset. For det andre kan det tenkes at det er høy forventning, og at antallet brukere blir lik N_1 .

Hvis vi ser mer i detalj på markedssituasjonen, vil vi se at situasjonen med lav forventning ikke kan forventes å bli en likevekt i markedet. For å forstå dette, er det naturlig å se på hvilke krefter som trekker i retning av henholdsvis flere eller færre brukere. Dette er illustrert med piler i figur 2. Hvis det er færre brukere enn N_0 , vil marginal betalingsvilje være lavere enn den pris tilbyderne kan tilby (vist ved kurven tilbud). Det vil ventelig bety at enda flere brukere velger ikke lenger å ha dette produktet, og vi beveger oss mot en situasjon med null brukere. Hvis det derimot er noen flere brukere enn N_0 , vil betalingsviljen for en ny bruker overstige den prisen tilbyderne kan akseptere. Da kan det lede til at nye brukere kommer til, og det blir mer attraktivt for enda flere nye brukere. Denne dynamikken vil fortsette helt til

den nye brukeren igjen har en betalingsvilje lik den pris tilbyderne kan tilby. Dette er tilfellet når det er N_1 antall brukere.

Vi ser at dersom antallet brukere kommer over en viss terskel, vil markedet "ta av" og det blir en selvforsterkende etterspørsel som leder til at vi får et stort antall brukere. For at dette skal skje, må antallet brukere over en kritisk masse, nærmere bestemt må en ha N_0 brukere.

På etterspørselssiden vil nettverkseffekten forsterke stordriftsfordelene. Desto flere som bruker et aktuelt produkt, desto mer nytte vil jeg ha av å benytte det samme produktet. Innen dataindustrien er det en rekke eksempler på produkter med såkalte nettverkseffekter. Desto flere som er knyttet opp mot Microsoft Networks e-postsystem, desto mer nytte vil andre kunder ha av å benytte det samme systemet. Desto flere som benytter Microsofts tekstbehandlingssystem Word, desto større verdi vil det ha å benytte samme produkt, fordi det gir større mulighet til å utveksle dokumenter.

Vi ser at dersom antallet brukere kommer over en viss terskel, vil markedet "ta av" og det blir en selvforsterkende etterspørsel som leder til at vi får et stort antall brukere.

2.1.2 Monopolisering

Stordriftsfordeler på etterspørselssiden kan innebære at markeder tipper mot monopol. Denne tendensen synes å være relevant innen dataindustrien. I Tabell 1 har vi vist graden av dominans i ulike markeder innen dataindustrien.

Produkt	Leder på verdensmarkedet	Markedsandel
PC operativsystem	Microsoft	94%
PC mikroprosessor	Intel	82%
Internett rutere	Cisco	80%
Auksjoner på internett	eBay	80%
Databasestyring	Oracle	50%

Tabell 1. Global markedsdominans i dataindustrien (Kilde: Evans og Schmalensee, 2001)

Vi ser at selv i et globalt perspektiv har én bedrift en betydelig markedsandel i flere av de produktkategoriene som er listet opp i tabellen. For enkelte av disse er det åpenbart at nettverkseffekter spiller en viktig rolle. For eksempel vil et auksjonssted på Internett slik som eBay oppleve en selvforsterkende etterspørsel. Dersom det er mange selgere på nett-auksjonen vil det tiltrekke seg mange kjøpere og omvendt.

Denne selvforsterkende effekten kan forsterkes ytterligere av forhold på tilbudssiden. Den dominerende aktøren vil i kraft av sin omsetning ha mulighet til å videreutvikle sine programmer og utvikle stadig nye spesialsydde programmer. Andre aktører vil da ha problemer med å investere i FoU i samme takt, fordi deres omsetning ikke er stor nok til å dekke tilsvarende høye faste kostnader. Dette gjør det enda mer vanskelig å ta opp konkurransen. Den dominerende aktøren har kommet over den kritiske masse av brukere, har markedskreftene på sin side, og kan føre an til tilnærmet monopol er oppnådd.

En studie av bruk av programvare i norske kommuner og fylker illustrerer problemene som oppstår når en liten aktør skal konkurrere mot en aktør med markedsdominans. I Teleplans

rapport om åpen kildekode⁴ er det gjengitt en undersøkelse av bruk av åpen programvare i norske kommuner og fylker, med særlig vekt på bruk av Linux som et alternativ til programvare fra Microsoft. Studien finner blant annet at en viktig grunn til manglende utbredelse av Linux er mangel på såkalte fagapplikasjoner som kan kjøres på Linux operativsystem. Fagapplikasjoner er spesialsyddde programmer for et bestemt formål (for eksempel saksbehandling). I svært mange tilfeller er disse spesialprogrammene skreddersydd for bruk med Microsofts applikasjoner (Office). Microsoft nyter dermed godt av en selvforsterkende etterspørsel – sterk dominans gjør det naturlig for produsenter av fagapplikasjoner å skreddersy programmene for Microsoft.

2.2 Historisk riss: Fra vertikal integrasjon til lagdeling

Som forklart i foregående avsnitt, er dominans bortimot uunngåelig i næringer med nettverkseffekter. Men dominansen varer ikke nødvendigvis evig. Tvert i mot er det en rekke eksempler fra dataindustrien på dramatiske skifte i dominans. Hvorfor oppstår slike skift? Er det grunn til å tro at slike skift vil inntreffe i dataindustrien med større sannsynlighet nå enn før? For å drøfte dette vil vi sammenligne situasjonen i dataindustrien i henholdsvis 1980 og 1995⁵.

I 1980 var dataindustrien karakterisert ved stor grad av vertikal integrasjon. Den enkelte produsent tilbød kunden et komplett tilbud, som inkluderte alt fra datamaskin til programvare. Kunden hadde ikke selv muligheten til å kombinere ulike løsninger, for eksempel kjøpe datamaskinen fra én produsent og programvaren fra en annen. Hvis en først hadde tatt et valg, var en som kunde innelåst hos denne produsenten. Følgelig var det en konkurranse om å vinne markedet. I tillegg til selskapene som er nevnt i Figur 3, kan vi fra et norsk perspektiv tilføye produsenten Norsk Data, som et klassisk eksempel på vertikal integrasjon. Var du først kunde hos Norsk Data, hadde du små muligheter for å kunne kommunisere med andre systemer.

Salg og distribusjon	IBM	DEC	Sperry Univac	Wang
Applikasjoner og programvare	IBM	DEC	Sperry Univac	Wang
Operativsystem	IBM	DEC	Sperry Univac	Wang
Maskinvare	IBM	DEC	Sperry Univac	Wang
Brikker (chips)	IBM	DEC	Sperry Univac	Wang

Figur 3. Dataindustrien i 1980

⁴Teleplan (2003): *Åpen programvare i Norge – status, effekter, hindringer og drivere*. Rapport skrevet på oppdrag av Arbeids- og administrasjonsdepartementet og Nærings- og handelsdepartementet

⁵Figur 3 og Figur 4 ble første gang fremstilt av Andrew S. Grove i 1996, den gang administrerende direktør i Intel. Diskusjonen her er basert på fremstillingen i Bresnahan (2002): *The Right Remedy*, Working Paper 233, Stanford University.

15 år senere var situasjonen endret dramatisk, se Figur 4.

Salg og distribusjon	Detaljist	Forhandler	Fjernhandel	Andre
Applikasjoner og programvare	MS Office			Andre
Operativsystem	DOS og Windows		OS/2	Mac Unix
Maskinvare	Compaq	Dell	HP	IBM Andre
Brikker (chips)	Intel arkitektur		Motorola	RISC

Figur 4. Dataindustrien i 1995

Vi ser at dataindustrien ikke lenger er vertikalt integrert, men lagdelt. På hvert nivå kan en velge mellom alternative produkter fra ulike produsenter. En kan for eksempel først velge en datamaskin, og deretter hvilken programvare en vil benytte på denne datamaskinen. Ingen selskaper kontrollerer hele kjeden fra databrikke til ulike programvarer.

Imidlertid kan vi se at det i noen tilfeller er knyttet tette bånd mellom enkelte aktører, som for kunden kan gi en tilsvarende effekt som eierskap. Et eksempel på dette kan være båndene mellom maskinvare- og programvareprodusentene, hvor både IBM og HP i sine annonser anbefaler operativsystemer Microsoft XP Professional. I motsetning til på 80-tallet er det ikke umulig å bytte ut programvaren, men det kan være tungvint og lite økonomisk gunstig for forbrukeren.

Fra IBM til Microsoft

CP/M var det rådende operativsystemet på 70-tallet. Det fantes på samtlige datamaskiner. Men så introduserte IBM et nytt og bedre operativsystem, som innebar et skift fra 8 til 16 bit og mulighet for farger. Dermed ble IBMs maskin plattformen de andre aktørene bygget på, ene og alene på grunn av en drastisk teknologisk innovasjon, dvs. et såkalt eksternt sjokk. IBM oppnådde markedsdominans. Konkurransemyndighetene i USA gikk til sak mot IBM for å få selskapet splittet opp, men tapte. Som kjent mistet IBM likevel sin dominans. Igjen kom en drastisk innovasjon (eksternt sjokk), men denne gangen fra andre enn IBM.

IBM og Microsoft samarbeidet på begynnelsen av 1980-tallet om å utvikle operativsystemet OS/2. Etter hvert valgte de å gå hver sin vei. IBM videreutviklet OS/2 og Microsoft satset på Windows. Windows kom i sin første versjon i 1984. En viktig årsak til Windows sin suksess var at Microsoft hadde en rekke applikasjoner som var skreddersydd for sitt operativsystem. Word og Excel var sentrale applikasjoner, og Microsoft kjøpte også opp Powerpoint. Det var en utbredt oppfatning at Apple i utgangspunktet hadde et mer brukervennlig system, Macintosh, men at Apple manglet flere applikasjoner som Microsoft kunne tilby i sin pakkedøsning. Det store gjennombruddet for Windows kom med versjon 3.0, som ble introdusert i juni 1990. 3.0-versjonen inneholdt mange av de brukervennlige egenskapene til Macintosh.

Fra WordStar til Word via WordPerfect

Et annet eksempel er skiftene i dominans innen tekstbehandling. Da IBM overtok operativsystemet CP/M, og ble den dominerende aktøren på markedet, skapte det store problemer for det daværende dominerende tekstbehandlingssystemet WordStar. Dette åpnet veien for tekstbehandlingssystemet WordPerfect. Da Microsoft kom med Word 2.0, som med sitt grafiske brukergrensesnitt var tilpasset deres Windows-plattform, mistet WordPerfect sin dominerende posisjon.

Årsaken til dette skiftet var et eksternt sjokk (se faktaboks). Eksterne sjokk har altså implikasjoner for skifte av plattform, så vel som for dominans innen hvert enkelt lag eller segment. Dataprodukter er såkalt komplementære, og økt bruk av ett produkt fører gjerne til økt bruk av et annet og komplementært produkt. Dersom en har god kunnskap i bruk av andre Microsoft Windows applikasjoner, vil en ha stor nytte av å anvende en applikasjon som for eksempel Excel regneark som er skreddersydd for Windows-plattformen.

2.3 Konkurransemyndighetene griper inn

Både amerikanske og europeiske konkurransemyndigheter har forsøkt å gripe inn mot Microsofts dominerende stilling i programvaremarkedet. Microsoft er kritisert for følgende:

- Teknisk innlåsing: Det er vanskelig for brukeren å bytte leverandør fordi programvaren inngår i en større sammenheng hvor alle produktene er tilpasset hverandre.
- Produktsammenbinding ("bundling") av nettleser (Microsoft Internet Explorer) og mediaavspiller (Windows Media Player), noe som er et brudd på antitrustlovgivning i både USA og EU.
- Urettmessig å benytte sin dominerende rolle i PC-markedet til å vinne fordeler i servermarkedet, særlig ved at Windows operativsystem ikke fungerer sammen med øvrige aktørers servere.
- Aggressiv atferd overfor konkurrenter gjennom oppkjøp og trusler og ved å underby sine egne produkter ved store kontrakter, noe som diskriminerer mellom kjøpere på markedet.

2.3.1 Amerikanske myndigheters inngripen mot Microsoft

Amerikanske myndigheter førte på 90-tallet en rettssak mot Microsoft. Først ville de dele opp selskapet i to – et for operativsystem og et for applikasjoner. Dette lyktes ikke. Dette forsøket på å splitte Microsoft i to selskaper hadde nettopp sin bakgrunn i ønsket om å legge til rette for konkurranse mellom plattformer. Det sentrale var å skape konkurranse mellom ulike deler av dataindustrien, slik at selskapene la seg i selen for å utvikle nye og bedre produkter. Forslaget var å la ett selskap få ansvaret for operativsystemet Windows, mens et annet fikk ansvar for applikasjonene. Dette ville ifølge amerikanske myndigheter kunne stimulere næringen til ytterligere produktutvikling.

Som kjent ble det ingen oppsplitting av Microsoft. Men det betyr ikke at det ikke ble i verk satt tiltak mot Microsoft i USA. Selskapet ble blant annet pålagt å offentliggjøre informasjon om dets programvare, slik at det ble lettere for andre serverbaserte operativsystemer å kommunisere med operativsystemet Windows. Det sentrale var å åpne opp og offentliggjøre noen av operativsystemets koblingspunkter, kalt Application Programming Interfaces (API).

API-ene gjør at det er mulig å installere og kjøre ulike applikasjoner, eksempelvis Word eller Excel. Gjennom API-ene kan en applikasjon benytte programkode som allerede ligger i operativsystemet. Dette letter arbeidet for programvareutviklerne. Ved å gi andre operativsystemer tilgang til noen av Microsofts API-er, vil det bli lettere for andre serverbaserte operativsystemer å fungere sammen med maskiner som benytter Microsofts operativsystem. Tiltakene synes imidlertid å ha hatt liten praktisk effekt.

2.3.2 EU-kommisjonens tiltak mot Microsoft

EU-kommisjonen har på sin side iverksatt andre tiltak enn amerikanske myndigheter.⁶ EU-kommisjonen ila i mars 2004 Microsoft en bot på om lag fire milliarder kroner for misbruk av dominerende stilling. Begrunnelsen var at Microsoft hadde benyttet sin dominerende posisjon i markedet for PC-operativsystemer til å øke sin innflytelse i henholdsvis markedet for operativsystemer for servere og i markedet for avspilling av bilde og lyd. Microsoft ble pålagt å gjøre åpent tilgjengelig noen av sine koder knyttet til grensesnitt, slik at andre serversystemer skal kunne kommunisere med Windows operativsystem for PC. Dessuten ble Microsoft pålagt å lage en versjon av Windows som ikke inneholdt MediaPlayer, programvaren for avspilling av lyd og bilde.

Microsoft har anket avgjørelsen og EU-kommisjonen har besluttet å vente med å sette i verk sanksjonene mot Microsoft, inntil rettsvesenet har sagt sitt i saken. Ankeprosessen vil ventelig ta minst et par år.⁷

Det er visse prinsipielle forskjeller mellom den oppsplitting som USA opprinnelig foreslo og de tiltak som EU-kommisjonen har pålagt Microsoft. USA valgte ikke å gå inn i selve produktene til Microsoft og be selskapet om å gjøre endringer. Fremfor oppsplitting av produktene valgte amerikanske konkurransemyndigheter å foreslå en oppsplitting av selskapet. EU-kommisjonens løsning er blitt kritisert av amerikanske konkurransemyndigheter, som hevder at denne type inngrep kan ha uante konsekvenser for innovasjonen.⁸ Amerikanske myndigheters kritikk av EU-kommisjonens avgjørelse illustrerer det dilemmaet myndighetene står overfor når en skal vurdere å gripe inn i denne typen næring. På den ene siden er det viktig å for dem å sikre en reell trussel om at en ny plattform kan overta hegemoniet i markedet. På den annen side er det viktig å sikre at det enkelte selskap har incentiver til innovasjon og stadig produktutvikling. Det siste tilsier at det faktisk er lov å vinne og bli dominerende.⁹ Men hvordan kan en sikre innovasjon og utvikling og samtidig unngå at et selskap benytter sin markedsdominans til å hindre konkurrenter i å komme inn på markedet?

2.4 Oppsummering: markedsdynamikk

I dette kapitlet har vi sett på hvordan strukturen i både utvikling, salg og distribusjon av datasystemer har endret seg fra 1980 frem til i dag (Figur 3 og Figur 4). Fra å være en vertikal leveransekjede, der én leverandør sto for alt fra maskinvare til salg og distribusjon til

⁶http://europa.eu.int/rapid/start/cgi/guesten.ksh?p_action=gettxt=gt&doc=IP/04/382|0|RAPID&lg=en&display=

⁷<http://www.aftenposten.no/nyheter/nett/article818272.ece>

⁸Se http://www.usdoj.gov/atr/public/press_releases/2004/202976.htm

⁹Pro og contra i saken mot Microsoft er blant annet drøftet i D. S. Evans, F. F. Fisher, D. Rubinfeld og R. Schmalensee (2000), *Did Microsoft Harm Consumers? Two Opposing Views*, AEI – Brookings Joint Center for Regulatory Studies, hvor henholdsvis Microsofts og Department of Justices økonomer begynner sine synspunkter.

sluttkundene, er markedet nå mer horisontalt, og leverandører bygger på hverandres produkter.

Dynamikken i programvaremarkedet skaper ofte dominerende aktører på grunn av nettverkseffekter – det er større nytte av programmer som mange bruker. Dette er i utgangspunktet et gode, både for brukerne og for de som klarer å etablere seg som større aktører i IKT-bransjen.

For å motvirke negative sider ved slike monopoltendenser er det viktig å iverksette tiltak som kan virke konkurransefremmende. Som vi har sett, har det vist seg vanskelig å skape et ideelt marked på programvareområdet. Mye av den samme effekten kan imidlertid oppnås gjennom å bevare trusselen om at monopolen er midlertidig. En mulig strategi for dette, er aktiv bruk av konkurranseloven for å motvirke misbruk av markedsrett. Dette vil være på linje med EUs strategi, slik vi har sett i kapittel 2.3.2. Andre strategier kan være krav om standardisering og åpne formater fra det offentliges side. Slike strategier beskrives nærmere i Kapittel 5.



Kapittel 3 Programvarepolitikk i Norge

Kravet om et koordineringsdepartement for IT (1993-95) førte ansvaret for IT-styring fra Statsministerens kontor (SMK) via Samferdselsdepartementet (SD) og det tidligere Planleggings- og samordningsdepartementet (PSD), til IT-politisk avdeling i Nærings- og handelsdepartementet (NHD), og i juni 2004 videre til Arbeids- og administrasjonsdepartementet.¹⁰ Moderniseringsminister Meyer er slik sett en IKT-minister, men de andre departementene har sektoransvar for IKT-spørsmål. Programvarespørsmål er et aspekt ved IKT-politikken som berører både den overordnede IKT-politikken og sektorene.

Regjeringens politikk er først og fremst uttrykt i eNorge-planen, særlig eNorge 2005¹¹ og *Strategi for IKT i offentlig sektor* (2003).¹² IKT er også et sentralt tema i så godt som alle samfunnsområder og sektordepartementer, som for eksempel forsvar, justis, samferdsel, helse, utdanning og kultur. Eksempler på temaer på dagsorden er: personvern, data-kriminalitet, sårbarhet, norsk innhold, bredbånd, næringsutvikling, standardisering, offentlige løsninger og opphavsrett. Relevante politiske satsinger på IKT i Norge i 2003-2004 har vært bredbåndsmeldingen (2003), elektronisk signatur (PKI), sikkerhetssatsing, Strategi for IT i offentlig sektor (2003), Arkitektur for elektronisk samhandling i offentlig sektor (forprosjekt, 2004), samt oppfølgingen av diverse EU-direktiver (telekommunikasjon, tilgang til offentlig innhold, digital opphavsrett og e-handel).

Folks tilgang til offentlige tjenester endres fundamentalt gjennom den pågående digitaliseringen av forvaltningen. Regjeringens politikk på IKT består i å utforme gode rammebetingelser for verdiskaping, sikre tilgjengelighet og sikkerhet, samt bidra til endringskompetanse i næringsliv og offentlig sektor. Politikken skal dessuten sikre reell konkurranse og brukermedvirkning.

3.1 Innkjøpspolitikk

Sommeren 2002 valgte daværende statsråd Victor D. Norman i Arbeids- og administrasjonsdepartementet å ikke fornye innkjøps- og serviceavtalen mellom staten og elleve store leverandører. Selv om tiltaket først ble lansert som et tiltak mot Microsoft og andre monopolleverandører,¹³ kan tiltaket etter hvert bedre tolkes som et tiltak for desentralisering av beslutninger.

¹⁰Arbeids- og administrasjonsdepartementet er fra 01.10.04 omgjort til et Moderniseringsdepartement med hovedfokus på modernisering av offentlig sektor.

¹¹All IT-politikk legges nå ut på nettstedet eNorge (<http://www.enorge.org/>)

¹²<http://odin.dep.no/aad/norsk/aktuelt/presse/002001-070128/index-dok000-b-n-a.html>

¹³<http://www.wired.com/news/antitrust/0,1551,53898,00.html>

Et viktig moment er hvordan norsk statsforvaltning og norske kommuner praktiserer *Lov om offentlige anskaffelser* (1999), der det i §5 blant annet heter at "oppdragsgiver skal ... sikre at det ikke finner sted forskjellsbehandling mellom leverandører. En anskaffelse skal så langt det er mulig være basert på konkurranse." Videre skal oppdragsgiver ikke "... bruke standarder og tekniske spesifikasjoner som et virkemiddel for å hindre konkurranse".

Relevant for programvareinnkjøp er det også at man i følge §6 skal "... under planleggingen av den enkelte anskaffelse ta hensyn til livssyklus kostnader og miljømessige konsekvenser av anskaffelsen".¹⁴

3.2 Ny norsk konkurranselov

Norge fikk ny konkurranselov 1. mai 2004.¹⁵ Den nye loven styrker Konkurransetilsynets¹⁶ arbeid mot karteller og monopolvirksomhet, blant annet ved å gi tilsynet adgang til å utstede bøter.

Den nye konkurranseloven er en ren forbudslov – ikke en lov om inngrep.¹⁷ Konkurranseloven fremmer to sentrale forbud:

§10 om *konkurransebegrensende avtaler mellom foretak og*

§11 om *utilbørlig utnyttelse av dominerende stilling.*

I § 14 om *konkurransefremmende tiltak* står det at: "Dersom det er nødvendig for å fremme konkurransen i markedene, kan Kongen ved forskrift gripe inn mot vilkår, avtaler og handlinger som begrenser eller er egnet til å begrense konkurransen i strid med lovens formål.

Konkurransetilsynet kan ikke instrueres av Arbeids- og administrasjonsdepartementet i enkeltsaker. Vedtak og avgjørelser kan imidlertid påklages til departementet, med unntak av overtredelsesgebyr som må bringes inn for domstolen. Det skal tas særlig hensyn til forbrukernes interesser ved anvendelse av loven.¹⁸ Ved forhold som kan påvirke samhandelen i EØS i henhold til artikkel 53 og 54 i EØS-avtalen,¹⁹ gjelder EØS-konkurranseloven.²⁰

¹⁴I § 10-2 av forskriften til loven står det følgende om kriterier for valg av tilbud:

"(1) Tildelingen skal skje enten på basis av hvilket tilbud som er det økonomisk mest fordelaktige, eller utelukkende ut fra hvilket tilbud som har den laveste prisen.

(2) Dersom tildelingen av kontrakten skjer på grunnlag av det økonomisk mest fordelaktige tilbud, skal alle kriterier som vil bli lagt til grunn oppgis enten i konkurransegrunnlaget eller i kunninggjøringen. Kriteriene skal om mulig angis i prioritert rekkefølge. Det kan benyttes slike kriterier som pris, kvalitet, teknisk verdi, estetiske og funksjonsmessige egenskaper, service og teknisk bistand, oppfyllestid mv." <http://www.lovdatab.no/for/sf/nh/th-20010615-0616-017.html#10-2>

¹⁵<http://www.konkurransetilsynet.no/internett/index.asp?strUrl=1004853i&topExpand=&subExpand=>

¹⁶Ny konkurranselov i Norge i 2004:

<http://www.dep.no/aad/norsk/aktuelt/pressem/002001-070178/index-dok000-b-n-a.html>

<http://www.dep.no/aad/norsk/publ/otprp/002001-050019/index-dok000-b-n-a.html>

¹⁷ Dette betyr at den gamle § 3.10 forsvinner (inngrepslov) og blir erstattet av § 11 (Utilbørlig utnyttelse av dominerende stilling) – men dette kan ikke straffes og er å betrakte som overtredelsesgebyr.

¹⁸ Det er uklart hvor viktig denne formuleringen, som kom med under Odelstingsbehandlingen, blir.

¹⁹ Rådsforordning (EF) 1/2003 handler om desentralisert håndheving

²⁰ <http://www.konkurransetilsynet.no/internett/index.asp?strUrl=1004854i>

3.2.1 Økt fokus på markedssvikt

Konkurransetilsynet har ikke tidligere vært spesielt oppmerksomme på en eventuell markedssvikt i det norske programvaremarkedet. På Teknologirådets høring i 2003²¹ uttrykte Konkurransetilsynet støtte til en mer offensiv strategi for åpne standarder, fordi dette etter tilsynets oppfatning vil kunne bedre konkurransen i markedet ved at det blir vanskeligere for dominerende aktører å opprettholde sin posisjon.²²

3.3 Norsk politikk i forhold til åpen kildekode

Åpen kildekode er enkelt forklart kildekode som er tilgjengelig og kan modifiseres av alle. I kapittel 4.2 gjør vi en grundig drøfting av definisjonen på åpen kildekode.

Allerede i 2001 gjorde Statskonsult en studie av åpen kildekode i det offentlige.²³ Statskonsult mente det var et stort potensial, ”særlig der de totale IT-kostnadene i hovedsak er knyttet til kjøp av lisenser”²⁴ og utfordret til norsk satsing på området. Etter dette har imidlertid politiske myndigheter sagt og gjort relativt lite på området åpen kildekode.

3.3.1 Nærings- og handelsdepartementet

En rapport om programvarepolitikk fra Teleplan²⁵ utført på oppdrag fra Nærings- og handelsdepartementet, utredet enkelte sider ved programvare basert på åpen kildekode, særlig knyttet til kommunene og skolesystemet.

Rapporten peker på at 25 norske skoler bruker Linux-programvare i en eller annen form²⁶ (s.6), men at utvalget av pedagogisk programvare er begrenset. Kommunene bruker også programvare basert på åpen kildekode i noen grad, men ikke kontorstøttesystemene. De mangler også fagapplikasjoner som kan kjøres på Linux operativsystem (s.6). Teleplans rapport signaliserer flere gevinster ved å fremme programvare basert på åpen kildekode, både økonomiske besparelser og stimulering av konkurransen i programvaremarkedet. De mange høringsinnspillene²⁷ i etterkant viste at temaet provoserer.

På konferansen *Informasjonssamfunnet for alle*, arrangert av Teknologirådet og Nærings- og handelsdepartementet høsten 2003, sa Statssekretær Oluf Ulseth:

”Åpen kildekode er et høyaktuelt tema både for oss og i andre land. [...]Regjeringen er også opptatt av at det skal være virksom konkurranse i markedet. Vi vil at det skal finnes alternativer. Selv om mange først og fremst assosierer åpen kildekode med gratis programvare, så er vi vel så opptatt av åpenheten i programvaren – hvor åpen plattformen er for utvikling av nye produkter og tjenester, i hvilken grad produktene bruker åpne standarder som gir bedre tilgang og friere konkurranse, foruten hvor åpen for innsyn selve

²¹<http://www.teknologiradet.no/html/508.htm>

²²<http://www.teknologiradet.no/files/konkurransetilsynet.pdf>

²³<http://www.statskonsult.no/prosjekt/itavd/2001/openprogrv.htm>

²⁴Statskonsults rapport fra 2001 (<http://www.statskonsult.no/publik/rapporter/2001/2001-07.pdf>)

²⁵<http://www.enorge.org/FileArchive/246/Endelig%20rapport%20Open%20Source%203112003.pdf>

²⁶Pr. Oktober 2004 er dette tallet ca 100.

²⁷http://www.enorge.org/modules/module_111/news_item_view.asp?iNewsId=2406&iCategoryId=153&iSubjectId=

kildekoden er. Også åpen programvare har kostnader, som vi alle vet, enten det er knyttet til salg av produktet, kompetanse eller tjenester.”²⁸

I *Status for elektronisk innhold 2003*²⁹ fra Nærings- og handelsdepartementet heter det at: ”Å kunne levere produkter som tilfredsstiller internasjonale krav til tilgjengelighet (WAI³⁰) er i ferd med å bli et viktig konkurransefortrinn” (pkt. 3.3.2). WAI, eller *Web Accessibility Initiative*, er et sett retningslinjer som *the World Wide Web Consortium* (W3C) anbefaler for å gjøre nettsteder og nettlesere tilgjengelige for alle, mest mulig uavhengig av teknisk plattform eller funksjonshemming.

3.3.2 Arbeids- og administrasjonsdepartementet – modernisering med IKT

I Strategi for IKT i offentlig sektor (2003-2005)³¹ fra Arbeids- og administrasjonsdepartementet går det frem at det er i tråd med moderniseringsprogrammet i offentlig sektor å støtte ”den internasjonale utvikling av åpen kildekode”. Lenger fremme i strategien heter det imidlertid at ”ut i fra innrettingen av den foreliggende strategien, tas området ikke med blant de prioriterte” (s.34) selv om saken også må ses i sammenheng med forslaget om en ”sentral standardiseringsfunksjon”.

3.3.3 Utdannings- og forskningsdepartementet og Skolelinux

I utdanningssektoren har Norge gjennom Skolelinuxprosjektet vært et foregangsland for å få skoleelever til å bli mer aktive, kompetente og bevisste IKT-brukere. Statskonsults rapport *Erfaringer fra bruk av Skolelinux*³² viser at prosjektet har vært en suksess og bør videreføres. Skolelinux er hittil bygd på dugnad og finansiert av sponsorer fra privat næringsliv med stiftelsen Norwegian Unix User Group Foundation (NUUGF) som sentral støttespiller. Det offentlige har bidratt noe til oversettelse av kontorstøttepakken OpenOffice.org til nynorsk og samisk.

3.3.4 Barne- og familiedepartementet – forbrukerpolitikk

I forbrukerpolitikken har de nordiske forbrukerorganisasjonene etablert et eget informasjonsnettsted for programvare basert på åpen kildekode³³, med aktiv støtte fra Barne- og familieminister Laila Dåvøy. Statssekretær Ruth Stenersen uttalte 13. oktober 2003 i sitt foredrag ved utdelingen av Norwegian Unix User Groups (NUUG) pris 2003 at ”nordiske forbrukere nå kan minske avhengigheten av dominerende programvareprodusenter, og bygge videre på den språklige og kulturelle særstilling de nordiske landene har”.³⁴ I sin høringsuttalelse til Teleplans rapport om åpen programvare³⁵ understreket Barne- og familiedepartementet denne holdningen.

²⁸Statssekretær Oluf Ulseths tale 30-09.03:

<http://www.odin.dep.no/nhd/norsk/aktuelt/taler/024101-090045/index-dok000-b-n-a.html>

²⁹<http://www.handel.no/FileArchive/145/StatusInnhold2003.pdf>

³⁰<http://www.w3.org/WAI/>

³¹<http://odin.dep.no/aad/norsk/aktuelt/presse/002001-070128/index-dok000-b-n-a.html>

³²Statskonsult: *Rapport 2003:24 Erfaringer fra bruk av Skolelinux – Bruk av åpen programvare i fire norske skoler.*

³³<http://www.nordicos.org>

³⁴http://odin.dep.no/bfd/norsk/aktuelt/taler/taler_politisk_ledelse/004051-090234/index-dok000-b-n-a.html

³⁵Teleplan (2003): *Åpen programvare i Norge – status, effekter, hindringer og drivere.* Rapport skrevet på oppdrag av Arbeids- og administrasjonsdepartementet og Nærings- og handelsdepartementet

3.4 Norsk politikk på åpne standarder

Det finnes en utførlig drøfting av mulige definisjoner på åpne standarder i kapittel 4.1.

Spørsmålet om tiltak for å fremme åpne standarder og protokoller var oppe i Stortingets spørretime 15. oktober 2003.³⁶ Representanten Jørund Leknes spurte daværende Arbeids- og administrasjonsminister Victor Norman:

”Den norske statsforvaltningen benytter i dag i stor utstrekning programvare basert på lukkede standarder og protokoller. Dette er konkurranseevridende, og vanskeliggjør overgang til alternativer basert på åpen kildekode. Det finnes nå gode alternativer til Microsoft som er gratis, særlig kan kontorpakken OpenOffice og operativsystemet Linux nevnes. Hva vil statsråden foreta seg for å øke bruken av programvare, som er basert på åpne standarder og protokoller?”

Barne- og familieminister Laila Dåvøy svarte på vegne av Norman at valg av programvare er et lokalt ansvar, men at bruk av åpne formater er viktig når informasjon skal utveksles med publikum og næringslivet. Hun pekte på at det også var en konkurransepolitisk problemstilling, og at Arbeids- og administrasjonsdepartementet i samarbeid med Nærings- og handelsdepartementet har laget en undersøkelse på ”området åpen programvare”. Hun sa at tiltak ville komme i etterhånd av denne rapporten.³⁷

I regjeringens innholdsstrategi fra april 2002³⁸ heter det blant annet:

”På det tekniske planet, i produksjonen av innhold, må det vere openheit overfor ikkje-proprietære opne standardar, som har vore grunnleggjande for utviklinga av mellom anna Internett. Her har det offentlege sjølv eit ansvar for ikkje å utelukke opne standardar internt eller i samkvem med brukarane. Sjølv om det kan vere unntak, vil manglande konkurranse som oftast bremse utviklinga av ein marknad” (s.15)

3.4.1 Moderniseringsdepartementet

Arbeids- og administrasjonsdepartementet (nå Moderniseringsdepartementet) har startet et ambisiøst og omfattende arbeid for å samordne IKT innenfor offentlig sektor. Samordning av grunndata, integrerte løsninger for saksbehandling og økt bruk av selvbetjening fra borgernes side er viktige områder det skal arbeides med. Departementet har utarbeidet en forprosjektrapport: *Arkitektur for elektronisk samhandling i offentlig sektor*.³⁹ Her peker man på standardisering som et viktig virkemiddel for å nå de fastsatte målene, og man ønsker å vektlegge bruk av åpne standarder for å sikre konkurransen.

3.4.2 Miljøverndepartementet

Regjeringens satsing på universell utforming,⁴⁰ at produkter skal være så fleksible at alle brukere får like muligheter, er nært knyttet til spørsmålet om åpne standarder.⁴¹ Nettsteder som følger åpne standarder, særlig WAI-kravene, er lette å tilpasse til alle typer brukere.

³⁶<http://odin.dep.no/aad/norsk/aktuelt/002001-991070/>

³⁷<http://odin.dep.no/aad/norsk/aktuelt/002001-991070/>

³⁸<http://www.dep.no/archive/nhdvedlegg/01/02/eNorg059.pdf>

³⁹<http://odin.dep.no/filarkiv/216451/Arkitektur.pdf>

⁴⁰<http://www.universell-utforming.miljo.no/>

⁴¹<http://www.universell-utforming.miljo.no/artikkel.shtml?id=60>

Dette gjelder funksjonshemmede, så vel som unge og eldre. Det er langt fra slik at alle produkter tilgjengelig i Norge følger disse anbefalingene, både grunnet manglende innsikt hos bestillere og fordi sterkere sanksjonsmidler er mangelvare.

3.4.3 Justissektoren

Høringsnotatet til ny IKT-strategi i justissektoren 2004-2007 omtaler utfordringer og tiltak på området teknisk infrastruktur på følgende måte: "De siste årene har en ny teknologi kalt WEB-Services fått økende betydning. WEB-Services er basert på SOA⁴², er sterkt distribuert og basert på åpne internasjonale standarder." Videre sier de om strategiske tiltak at:

"Standardisering vil både lette samarbeidet internt i justissektoren og medføre lavere IT-kostnader i sektoren som helhet. Dersom det er mulig skal det velges samme teknologi som hovedtyngden av offentlig sektor. Det skal gis følgende teknologiske rammer: Format for informasjonsutveksling. Det standardiseres på XML⁴³ som basis for informasjonsutveksling mellom systemer. For de viktigste meldingstyper etableres det standarder i XML. WEB Services. Denne teknologien tas suksessivt i bruk for å frigjøre fra avhengigheter til teknologi."

Samlet kan vi si at IKT-strategi i justissektoren 2004-2007 omtaler og nevner åpne standarder som viktig, men presiserer ikke nøyaktig hva dette skulle bety for strategien eller for praksis. Symptomatisk nok finnes høringsnotatet på Justisdepartementets hjemmesider kun i Microsofts doc-format.⁴⁴

3.5 Status i norske kommuner

For kommunene er IKT et sentralt verktøy i fornyelse, omstilling og effektivisering. IKT benyttes innenfor de fleste virksomhetsområder. IKT har størst utbredelse innenfor administrasjon og i teknisk virksomhet, og minst anvendelse innenfor områdene pleie og omsorg, særlig barnehagesektoren.

To av tre kommuner har utarbeidet konkrete planer for bruk av IKT. Ca. syv av ti kommuner har slike planer for ett eller flere virksomhetsområder. Få norske kommuner har tatt i bruk programvare basert på åpen kildekode eller har en politikk for å fremme åpne standarder, ifølge rapporten *Åpen programvare i Norge – status, effekter, hindringer og drivere*, utarbeidet av Teleplan for Nærings- og handelsdepartementet i 2003. Manglende informasjon eller kompetanse, kobling mot eksisterende fagapplikasjoner samt bytte- og opplæringskostnader ser ut til å være viktige hindre.

Norge.no er en felles inngang til all offentlig informasjon på nettet. Virksomheten arbeider med å bidra til økt kvalitet på kommunale og statlige nettsted. Sentralt i dette står arbeidet med å kvalitetsvurdere statlige og kommunale nettsteder basert på kvalitets-

⁴²Service-oriented Architecture - arkitektur for å skape løse koblinger mellom ulike elementer i et komplekst IT-system, slik at de enkelte delene kan byttes ut uten at resten av systemet blir påvirket.

⁴³Extensible Markup Language – en måte å representere strukturert informasjon på.

⁴⁴http://odin.dep.no/jd/norsk/aktuelt/hoeringssaker/paa_hoering/012101-080049/index-dok000-b-n-a.html

kriteriene for offentlige nettsteder.⁴⁵ Det har vært gjennomført vurderingsrunder i 2001 og 2003, og en ny runde gjennomføres høsten 2004.

3.6 Øvrig offentlig arbeid med standardisering

I Strategi for IT i offentlig sektor (2003) fra Arbeids- og administrasjonsdepartementet var det et punkt om å styrke arbeidet med standardisering – for å styrke målene om *brukerretting, effektivisering og forenkling*. I mars 2004 ble det oppnevnt en tverrdepartemental gruppe med deltakere fra 12 ulike departement samt Kommunenes Sentralforbund for å se nærmere på saken.

Det offentliges oppfølging av IKT-standardisering skjer i dag også gjennom Standard Norge,⁴⁶ som særlig deltar i komiteer for interoperabilitet i forbindelse med informasjonstjenester.⁴⁷ Deres arbeid er foreslått styrket, blant annet av Arbeidsgruppen funksjonshemmede og ny teknologi (2001),⁴⁸ som påpeker at: "Oppfølgingen av det internasjonale standardiseringsarbeidet kan synes vanskeliggjort ved at ressursene til det norske standardiseringsarbeidet på IKT-området er skåret ned med 70 % i år 2001". Utfordringen for det internasjonale IKT-standardiseringsarbeidet og i direktiver (EU) er at "... hensynet til funksjonshemmedes behov blir tatt med i hele prosessen allerede fra starten. Dersom disse hensyn kommer til uttrykk i standarder som sikrer universell utforming, vil produktene og tjenestene kunne brukes av mange flere mennesker" (pkt 6.8).

Rapportering fra kommuner til Statistisk Sentralbyrå skjer ved hjelp av KOSTRA som kun er utviklet for Windows. Saksbehandlingssystemer (for eksempel eSak), binder staten til Windows.

3.6.1 Altinnprosjektet

Altinn er et samarbeid mellom Brønnøysundregistrene, Lånekassen, Statistisk Sentralbyrå (SSB) og Skattedirektoratet om digital innrapportering. Den tekniske løsningen er delvis basert på den åpne standarden XML.⁴⁹

3.7 Bruk av åpne standarder i norsk offentlig kommunikasjon

På tross av at offisiell politikk både i Norge og i EU fremhever tilgang for alle og bruk av åpne standarder i informasjonssamfunnet, er det fremdeles en rekke grensesnitt mellom myndigheter og mellom borgere og myndigheter som ikke følger åpne standarder og som forutsetter produkter som kun leveres av en leverandør:

- Leverandører av fagsystemer, for eksempel til økonomi eller arkivering, støtter nesten uten unntak⁵⁰ kun grensesnitt mot Microsoft-produkter.
- Statens reiseregning krever for tiden Windows operativsystem.⁵¹

⁴⁵<http://www.norge.no/kvalitet/kvalitet2004/>

⁴⁶<http://www.standard.no/imaker.exe?id=3617&visdybde=1&aktiv=3617>

⁴⁷<http://www.niso.org/international/SC4/index.html>

⁴⁸<http://www.dep.no/nhd/norsk/publ/rapporter/024101-220017/index-dok000-b-n-a.html>

⁴⁹Per 19. april 2004 støtter Altinn imidlertid kun Explorer, Netscape og Opera nettlesere. Dette vil sannsynligvis utbedres (<https://www.altinn.no/cms/1044/altinn/brukerstotte/sporsmal/424b9643-b8de-4741-b721-9e542586a555.htm>)

⁵⁰Ergo Ephorma har også begynt å støtte OpenOffice.org med fagsystemet ePhorte.

- Dokumenter i Word-format er utbredt i både intern og ekstern kommunikasjon, særlig i norske kommuner.

Dette betyr at den typiske norske bruker, enten han/hun er borger, skoleelev, bedriftsleder eller pensjonist, for alle praktiske formål må være Microsoftkunde.

På tross av at mange offentlige utredninger har deltakelse som ideal, tar altså ikke det offentliges IT-systemer og digitale satsing høyde for utfordringene knyttet til åpne standarder.

Noe av dette kan være i ferd med å endres, i og med det er et stadig sterkere fokus på åpne standarder så vel som åpen kildekode i statsforvaltningen. Både Nærings- og handelsdepartementet, Utdannings- og forskningsdepartementet og Moderniseringsdepartementet er i ferd med å ta inn over seg problemstillingen. Det må også understrekes at Riksarkivet, som en av de få offentlige institusjoner, tar åpne standarder alvorlig. Problemet er at de kommer inn *for sent* i prosessen og dokumentutveksling foregår lenge før dokumentet lagres for ettertiden.

3.8 Lukkede standarder i åpne medier

En rekke medier bruker Internett for å formidle sine programmer, journalistiske reportasjer og andre tjenester. Noen nettmedier bruker åpne standarder, men i sterk grad er standardene lukket. Til og med NRK og TV2, hvis fjernsynssendinger skal være tilgjengelige for alle, benytter lukkede standarder. NRKs web har Microsofts Windows media player som foretrukket avspiller for web-TV, mens TV2 har dette som eneste alternativ.⁵²

3.9 Oppsummering: programvarepolitikk blir stadig viktigere

Som vi pekte på i Kapittel 1, blir programvarepolitikk stadig viktigere. IKT og programvare spiller også en stadig viktigere rolle i samfunnslivet. Regjeringen, med Moderniseringsdepartementet som "implementeringspartner", setter IKT sentralt i moderniseringen av Norge, og setter dermed også programvarepolitikk på dagsordenen.

I denne rapporten setter vi spesielt fokus på *åpne standarder* og *åpen kildekode*. Status på dette området er at relativt lite er gjort i Norge – i praksis er det lukkede formater som dominerer offentlige systemer og offentlig datautveksling. Imidlertid finnes det enkelte uttrykk for at spørsmålet oppfattes som viktig, blant annet har det kommet signaler fra moderniseringsdepartementet om at man ønsker å stille krav om bruk av åpne standarder

⁵¹<http://odin.dep.no/aad/norsk/regelverk/rutiner/000001-990486/index-dok000-b-n-a.html>

⁵²<http://www.nrk.no/informasjon/hjelp/2884911.html>. NRKs begrunnelse for valget av Windows Media Player: <http://www.nrk.no/informasjon/hjelp/2402392.html> NRK påpeker imidlertid at det mulig å bruke Linux med en viss egeninnsats: <http://www.nrk.no/informasjon/hjelp/3157548.html>



Kapittel 4 Teknologiske strategier for et åpnere samfunn

Teknologivalg kan være viktig for å bidra til at samfunnet blir mer demokratisk, effektivt og fremtidsrettet.

Vi har tidligere pekt på problemet med programvare som portvakt til informasjon. Dette kapittelet vil diskutere strategiske valg knyttet til programvare som kan motvirke dette problemet. Vi vil spesielt diskutere *åpne standarder* og *åpen kildekode*.

For å gi et mer utfyllende bilde, vil vi også se på to teknologiske strategier som kan betraktes som en trussel mot åpenhet: *Programvarepatenter* og *tekniske sperrer*.

Innledningsvis kan det være nyttig å gi en kort innføring i begrepene *standard*, *kildekode* og *format*, som er begreper som vil bli mye brukt videre i rapporten:

Standard: En standard er et dokument som gir felles retningslinjer for hvilke krav som skal settes til et produkt (varer og tjenester) eller en arbeidsprosess. En standard inneholder hensiktsmessige spesifikasjoner, fremgangsmåter og terminologi.⁵³ Hensikten med standarder er å gjøre samarbeid enklere fordi man skaper forutsigbarhet. Eksempler på standarder de fleste forholder seg til daglig uten å tenke særlig over det er: A4-ark, CD-plater og støpselet på elektriske artikler (de fleste har sikkert også opplevd hvilke problemer det kan skape når man er et sted som ikke følger den samme standarden man selv er vant til).

Kildekode: Et programs kildekode er en instruks som forteller hva programmet skal gjøre. Kildekoden kan leses og endres av en programmerer.

Format: Ordning av data på et datamedium.⁵⁴ Det vil si at dataene er lagret på en bestemt og forhåndsdefinert måte (etter en bestemt *standard*). Ofte brukes begrepene *utvekslingsformater* og *lagringsformater* for filformater som brukes til hhv. utveksling av data mellom ulike systemer, og lagring av data i et system.

4.1 Åpne standarder

Åpen kildekode og *åpne standarder* er to begreper som ofte blandes sammen. Sammenblandingen har to årsaker: begge begreper har ordet *åpen* i seg, og programvare basert på åpen kildekode støtter ofte åpne standarder. I denne seksjonen skal vi definere og diskutere begrepet *åpne standarder*, og i neste seksjon vil vi gjøre det samme med *åpen kildekode*.

⁵³Definisjonen er hentet fra Standard Norge, <http://www.standard.no>

⁵⁴Definisjonen er hentet fra Bokmålsordboka, <http://www.dokpro.uio.no/ordboksoek.html>

Begrepet *standard* brukes, som vi har sett, på mange andre områder enn programvare og angir gjerne regler og retningslinjer for en aktivitet eller resultatet av denne aktiviteten. Denne rapporten er kun opptatt av programvare, og fokuserer spesielt på utveksling av informasjon mellom datamaskiner samt lagring av informasjon. Standarder på andre områder innen programvare (f.eks. syntaks for programmeringsspråk, ytelsesmålinger og kallgrensesnitt) er av mindre interesse, og det samme gjelder informasjon som kun lagres midlertidig på lokale maskiner. Det er heller ikke naturlig å diskutere programvare for telefonsentraler og mobilnett, selv om sentralene i dag for en stor del består av datamaskiner.

Hovedformålet med standarder for programvare er å gjøre det mulig for forskjellige programmer å utveksle informasjon. Dersom en produsent forsvinner, vil annen programvare fortsatt kunne lese informasjonen og forhindre at den går tapt. Standarder styrker også konkurransesituasjonen ved å gjøre det enklere for kunder å bytte leverandør av programvare, ved at man hindrer at kunder låses til en enkelt leverandør ("lock-in").

På programvarefeltet snakker vi oftest om *åpne standarder*. Betegnelsen *åpen* indikerer et fortrinn framfor andre standarder, men begrepet brukes litt forskjellig. For å kunne bruke begrepet på en effektiv måte i offentlig politikk er det derfor viktig å definere hva som menes og impliseres når man snakker om åpne standarder.

4.1.1 Historikk

I denne rapporten er det standarder for utveksling og lagring av digital informasjon som er i fokus. Standarder på dette feltet har tradisjonelt blitt utviklet av standardiseringsorganisasjoner som ISO og ITU-T⁵⁵. ISO består av mange nasjonale standardiseringsorganisasjoner. Norge er representert gjennom *Standard Norge*, mens ANSI, som først standardiserte ASCII, representerer USA. ITU-T, tidligere kjent under navnet CCITT, har tradisjonelt hatt medlemmer fra telekombransjen.

Eksempel: Standardisering av skrifttegn

Omkring 1840 utviklet Samuel Morse en metode for å kommunisere bokstaver og tall over en elektrisk linje. Metoden består av lange og korte signaler som en trent menneskelig operatør kan tyde. Omtrent 120 år senere, i 1960, ble en variant av den samme oppfinnelsen utviklet for datamaskiner: American Standard Code for Information Interchange (ASCII), utviklet av Robert Bemer fra IBM. ASCII gjorde det mulig for datamaskiner av forskjellig fabrikat å utveksle bokstaver, tall og andre enkle symboler. ASCII ble senere publisert som en standard av ANSI, som er en av flere organisasjoner som arbeider med åpne standarder for programvare. ASCII konsentrerer seg om bokstavene i det engelske alfabetet, og andre land sine bokstaver, inkludert våre egne æ, ø og å, er ikke med. Først med Unicode-standard, påbegynt omkring 1990, ble det mulig å utveksle og lagre dokumenter med tegn fra alle levende, samt en del historiske, språk.

Fremveksten av internett markerte et skille for hvordan standarder utvikles. Mens ISO og ITU-T utviklet standarder bak lukkede dører og solgte standardene i papirform når de var ferdigutviklet, ble standardene bak internett utviklet i IETF, der alle kunne delta fysisk og

⁵⁵Alle organisasjonene som presenteres her blir beskrevet nærmere i vedlegget *Standardiseringsorganisasjoner* bak i rapporten

elektronisk, og hvor standardene lå fritt tilgjengelig på nettet både under utvikling og etter ferdigstilling.⁵⁶

Rundt 1990 var det stor motstand i tradisjonelle telekom-miljøer mot Internett sine tekniske standarder og måten disse ble utviklet på. Det ble hevdet at internett i beste fall var et interessant eksperiment, men at utviklingen av pålitelige nettjenester ville kreve en modell som ISO representerer. X.25 og X.400 var sentrale ITU-T standarder som henholdsvis standardiserte dataoverføring og elektronisk post. IETF, på sin side, utviklet TCP/IP og SMTP som hadde lignende funksjonalitet. I dag er det klart at IETF sine spesifikasjoner vant frem. Åpenheten som IETF sto for var en viktig faktor for dette.

En tredje type organisasjon som har vokst fram er industrikonsortiene. Blant de mest kjente er MIT sitt X Consoritum, World Wide Web Consortium (W3C), Unicode Consoritum, og OASIS. Modellen er basert på samarbeide mellom medlemsbedrifter som finansierer og utvikler standarder. Industrikonsortiene er likevel ofte åpne for allmenn diskusjon og deltagelse da dette regnes å styrke standardiseringsarbeidet.

Alle de tre nevnte utviklingsmodellene kan lage teknisk gode standarder for programvare. Økt åpenhet i utviklingen av en standard ser imidlertid ut til å virke positivt inn på kvaliteten og gjennomslagskraften for en standard.

4.1.2 Forslag til en god definisjon av åpne standarder

Før en kan diskutere hvorvidt bruk av åpne standarder skal anbefales eller ikke, er det viktig å ha en definisjon av hva en åpen standard er. Teknologirådets ekspertgruppe mener begrepet *åpen standard* bør brukes om standarder som er utviklet og blir vedlikeholdt etter veldefinerte kriterier, og vi legger følgende kriterier til grunn i denne rapporten:

1. Åpne standarder er beskrevet i et dokument som er publisert og er tilgjengelig for liten eller ingen kostnad.
2. Åpne standarder utvikles og vedlikeholdes av ikke-kommersielle organisasjoner hvor alle kan delta, eller i en organisasjon med et internasjonalt mandat.
3. Åpne standarder har ingen restriksjoner på bruk, for eksempel i form av avgifter eller begrensninger på hvilke plattformer standarden kan brukes.

Punktene over kan kritiseres både for å definere åpne standarder for strengt og for svakt. Det kan også hende at åpne standarder godt kan utvikles av andre, f.eks. private bedrifter eller universiteter, gitt at det er en viss forutsigbarhet og innsyn knyttet til utviklingen. Vi går gjennom argumentene og begrunnelsen for definisjonen i de to følgende seksjoner.

Er definisjonen for streng?

Det kan hevdes at en spesifikasjon kvalifiserer til betegnelsen *åpen* når den er tilgjengelig for alle (jf. punkt 1) og at punkt 2 og 3 ikke er nødvendige for å kunne kalle en standard

⁵⁶IETF og en del andre organisasjoner bruker ikke ordet *standard* om sine spesifikasjoner for ikke å komme i direkte konflikt med ISO. For å forenkle terminologien noe brukes ordet *standard* i denne rapporten også om spesifikasjoner fra andre organisasjoner

åpen. Vi mener de to siste punktene er viktige for å sikre at standarder forblir tilgjengelige, og får størst mulig gjennomslagskraft.

Punkt 2 diskuterer organisasjonsformen. En ikke-kommersiell organisasjon vil ikke endre planer like raskt som private bedrifter ofte gjør. Enkeltbedrifter kan eksempelvis fjerne eller endre dokumenter på nettsidene, mens dette vanskelig kan gjøres i organisasjoner som har etablerte vedtekter og publikasjonsprosesser. Videreutvikling av en standard er også et viktig argument mot enkeltbedrifters kontroll. Dersom en enkelt bedrift styrer utviklingen av en standard på egen hånd vil bedriftens produktutvikling kunne koordineres med utviklingen av en standard. Dette vil gi egne produkter et forsprang overfor konkurrerende produkter.

Kravet om at alle kan delta i standardiseringsorganisasjoner betyr ikke nødvendigvis at alle skal kunne delta gratis. Det koster penger å drive organisasjoner og det er naturlig å ta betalt av medlemmene. Det bør likevel være mulighet for allmenn deltagelse i elektroniske fora uten å betale full medlemsavgift. Eksempelvis bør alle ha mulighet til å sende tilbakemelding på en standard, kunne delta i et diskusjonsforum om standarden, samt ha lese-tilgang til endringsloggen.

I tillegg til organisasjoner der alle kan delta, finnes det også organisasjoner som tradisjonelt har krevd mandat fra et lands myndigheter for deltagelse. Fremst blant disse er ISO, men også enkelte FN-organ engasjerer seg i standardiseringsarbeid. Selv om disse organisasjonene krever mandat for deltagelse er de likevel basert på demokratiske grunnprinsipper som gjør dem egnet til utvikling av åpne standarder.

Punkt 3 omhandler restriksjoner på bruk av åpne standarder. Det vil typisk være enkelte restriksjoner på bruk av innholdet. Eksempelvis vil det kunne kreves at man refererer til standarden dersom man siterer fra den. Det må likevel ikke være restriksjoner som kan svekke en standards gjennomslagskraft. Lisensavgifter som må betales ved implementering er en slik restriksjon, da slike avgifter typisk vil gjøre det umulig å støtte standarden i åpen kildekode. Dette punktet betyr ikke at standarden må være patentfri, men at eiere av patenter kan ikke kreve lisensbetaling eller sette andre restriksjoner.

Er definisjonen for svak?

Definisjonen kan også kritiseres for å være for svak. Mange standarder går lengre enn definisjonen i sin åpenhet: I tillegg til selve dokumentet som beskriver standarden, publiseres ofte en referanseimplementasjon og et testsett ("test suite"). Disse blir laget for å hjelpe programmerere å tolke spesifikasjonen, samt å teste hvorvidt et program opptrer i henhold til standarden. Referanseimplementasjoner og testsett styrker en standard, men bør ikke være noe absolutt krav. Mange standarder interagerer med andre standarder på en kompleks måte, og det vil være vanskelig å teste dem uten også å kreve støtte for andre standarder. Å utvikle referanseimplementasjoner og testsett tar også mye ressurser, og dette vil kunne forsinke utviklingen av åpne standarder.

Det kan også hevdes at en standard ikke er åpen dersom det finnes patenter som helt eller delvis beskriver standarden, og at et fjerde punkt om patentfrihet burde vært tatt inn. Det er vanskelig å ha full oversikt over patenter og patentsøknader. Det er også vanskelig å vurdere hvorvidt et patent virkelig dekker samme område som en standard. Vi mener derfor det ikke er hensiktsmessig å ta inn et slikt krav, men dersom et patent befester sin rolle gjennom

rettssystemet eller dersom ledende industriaktører velger å betale patentlisenser, er det likevel ikke naturlig å regne den som en åpen standard, da dette vil stride mot punkt tre om avgifter.

4.1.3 Kan åpne standarder hemme innovasjon?

Standarder etablerer felles fundament der flere programvareleverandører kan møtes. Det tar tid og krefter å utvikle slike fundament og ofte må man finne kompromissløsninger underveis. Når fundamentet først er bygget, blir det gjerne stående en stund. Det kan derfor hevdes at standarder er en trussel mot innovasjon; de kreftene som blir brukt på standardisering kunne vært brukt på å lage nye, innovative funksjoner.

Dersom Microsoft ble tvunget til benytte åpne standarder, ville det kanskje bidra til mer konkurranse. Debatten mellom Microsoft og selskapets utfordrere, hvorav flere ønsker åpne standarder, er helt i tråd med lignende debatter om standardisering i andre situasjoner.⁵⁷ Den aktøren som har en dominerende posisjon vil ofte ha interesse av å velge en lukket standard, fordi den i utgangspunktet vil ha tilgang til en stor kundemasse som følge av at markedet har tippet i dens favør (se diskusjon rundt nettverkseffekter i kapittel 2.1.1.) Potensielle konkurrenter vil dermed ha interesse av en åpen standard, fordi det vil innebære at de får tilgang til en eksisterende kundemasse som de ellers ikke hadde fått samme tilgang til. Sånn sett vil en åpen standard sikre konkurranse i markedet.

På den annen side kan det være grunn til å frykte at et pålegg om en åpen standard vil redusere innovasjonstakten for den dominerende aktøren, og kanskje også for hele programvaremarkedet. En åpen standard vil føre til mindre grad av proprietære rettigheter for aktøren som utviklet programvaren, men ikke medføre at de bortfaller helt. En ulempe for utvikleren av standarden vil være at det kan være vanskelig å gjennomføre endringer, da man ikke lenger suverent styrer programvaren/grensesnittet, men må ta hensyn til at den er en standard for andre. Dessuten vil en forbedring av produktet komme også andre aktører til gode, og dermed ikke fullt ut tilfalle den aktøren som foretar produktutviklingen av denne delen av programvaren.

Da de fleste funksjoner i et program ikke har noe med standarder å gjøre vil det uansett være rom for innovasjon. Eksempelvis vil Microsoft kunne forbedre ytelse, stavekontroll, og utseende på Microsoft Word uten å gjøre endringer i lagringsformatet.

Et eksempel på en vellykket standard er GSM-standard for mobiltelefoni. Denne standarden sikrer at brukerne kan ringe med sin mobiltelefon alle steder hvor standarden er i bruk, uavhengig av land og operatør. Dersom hver operatør skulle operert med sin egen, proprietære standard, ville nytteverdien av en mobiltelefon være svært begrenset. Til tross for at alle operatører må basere seg på samme standard (eller kanskje nettopp derfor?), ser vi at det stadig dukker opp nye tjenester innen mobiltelefoni.

Innen programvareindustrien er det eksempler som illustrerer både vellykkede åpne standarder og vellykkede lukkede standarder. Når det gjelder dokumentstandards har en for eksempel alt fra åpen standard (HTML), delvis åpen standard (PDF) til en lukket standard (Doc). HTML er dokumentformatet som benyttes i forbindelse med internett, og det er blitt

⁵⁷ Shapiro og Varian (1999) er det drøftet hvordan bedrifter som er i ulike situasjoner kan finne det lønnsomt å velge henholdsvis en åpen eller lukket standard.

trukket frem som et eksempel på en åpen standard som ga tilgang til mange og dermed mulighet for å komme over en kritisk masse med hensyn på antall brukere. I så henseende bidro den åpne standarden til internetts suksess i betydningen rask utbredelse.

To eksempler på standardisering og konsekvenser for innovasjon

ASCII

La oss tenke oss et eksempel (A) der norske myndigheter i 1964 hadde innført ASCII som format for utveksling og lagring av informasjon. La oss videre tenke oss at norske myndigheter i dag, 40 år senere, fortsatt insisterer på bruk av ASCII – og kun ASCII – i offentlig sektor. En slik standardiseringspolitikk ville vært skadelig da den hadde hindret offentlig sektor i å ta del i utvikling og innovasjon siden den gang. Eksempelet er ikke realistisk, men viser at overdreven satsning på enkelte standarder kan virke hemmende. Samtidig er det en kjensgjerning at ASCII-filer fra 1964 fortsatt kan leses av dagens maskiner.

Et noe modifisert eksempel (B) begynner også med innføring av ASCII i 1964. Deretter foretar myndighetene jevnlig, for eksempel hvert tredje år, en gjennomgang av standardiseringspolitikken. Når nye åpne standarder med støtte for norske tegn blir utviklet vil man kunne velge å ta disse i bruk.

En litt annen hypotetisk variant (C) starter også med ASCII i 1964. I 1965 kommer så firma X med et patentert program Y som gjør det mulig å støtte norske tegn. Firma X tilbyr forvaltningen å ta i bruk den nye oppfinnelsen for en rimelig penge på betingelse av at offentlig sektor binder seg til en innkjøpsavtale med X så lenge patentet gjelder.

Forskjellen mellom B og C er at B bruker åpne standarder som basis for teknologivalg. Innovasjon får innpass, men det tar kanskje litt lengre tid enn i eksempel C. Til gjengjeld vil antagelig B være billigere fordi flere leverandører konkurrerer om kontraktene. I et historisk perspektiv (og historien inntreffer raskt på dette området) vil B være å foretrekke framfor C, da lagringsformatet som program Y bruker kanskje ikke lar seg lese.

Sammenligningen mellom eksempel B og C baserer seg på at man i eksempel B klarer å velge rett standard. I ettertid er det lett å se at ASCII var et riktig valg i 1964 (framfor EBCDIC som var alternativet), men den gang var det kanskje vanskeligere. Til en viss grad må beslutningstakere velge mellom forskjellige åpne standarder, men som oftest vil vinnerne peke seg ut ganske raskt. I løpet av en treårsperiode vil det erfaringsmessig være klart om en åpen standard vil overleve og tas i bruk eller glemmes.

World Wide Web

Et eksempel som ligger litt nærmere i tid enn ASCII, er utviklingen av World Wide Web som startet omkring 1990. De tre standardene som la grunnlaget for weben (HTML, HTTP, URL) ble alle utviklet som åpne standarder.

La oss gjøre et nytt tankeeksperiment: Hva hadde skjedd dersom CERN hadde valgt å ikke utvikle disse som åpne standarder? I stedet for å kommunisere og diskutere spesifikasjonene i åpne fora, ville CERN måttet utvikle spesifikasjonene selv, for deretter å utvikle programvaren selv (D). Alternativt kunne CERN lisensiert idéene til programvareprodusenter som så kunne videreutviklet dem, hver på sin måte (E).

Ville D eller E gitt oss et bedre internett med flere tjenester og mer informasjon? Dersom en går tilbake i meldingsarkivene fra diskusjonene den gang (arkivene er fortsatt tilgjengelige) ser man at en del av de opprinnelige idéene forsvant etter debatt på nettet. Noen av disse ville kanskje overlevd i scenario D og E. En kan argumentere at åpne diskusjoner dermed førte til at funk

sjonalitet som ellers hadde kommet brukerne til gode ble fjernet. I dag synes det likevel klart at idéene som forsvant ikke var blant de beste og at de nye idéene som kom til på grunnlag av diskusjonene (f.eks. formularer, tabeller og stilsett) var viktige bidrag til webens popularitet. De nye idéene ble også utviklet bemerkelsesverdig raskt. I dette tilfellet ser det derfor ut som om åpenhet tiltrekker seg dyktige mennesker som bidrar innovativt.

Selv om weben har vært en suksess, er det ikke dermed sagt at en ikke kunne fått tilsvarende, eller kanskje enda bedre, suksesshistorier i scenario E. Firmaene som lisensierte CERN sitt arbeide kunne hver hatt dyktige medarbeidere som bidro med egne idéer. Kanskje man kunne gjort HTTP raskere og HTML mer robust. Resultatet ville sannsynligvis vært at flere inkompatible versjoner av weben konkurrerte mot hverandre. Selv om de hver for seg kunne tilby gode tekniske løsninger, ville ekstraarbeidet for forfattere som måtte lage flere versjoner av sidene vært betydelig. Å lage sider for de forskjellige systemene ville også vært vanskelig da spesifikasjonene ikke nødvendigvis ville være tilgjengelige for alle. Det er rimelig grunn til å frykte at ingen av systemene ville oppnådd kritisk masse og weben, som i dag er vertskap for en stor mengde innovative tjenester og informasjon, ikke ville blitt etablert.

I eksemplene som har vært diskutert i denne seksjonen har åpne standarder blitt trukket fram som suksesshistorier. En kan også enkelt finne mange åpne standarder som ikke har vært vellykket. Det at en standard er åpen er ingen garanti for at den vil lykkes.

Åpne standarder har vesentlige fordeler i form av mulighet for deltakelse, lesbarhet og økt konkurranse, noe som også går klart frem av eksemplene. Utvikling av internett og web hadde ikke vært mulig uten åpne standarder, og kommersiell programvare har fritt kunnet implementere idéer hentet fra disse miljøene nettopp fordi standardene var åpne. Det er derfor lite ønskelig med en situasjon hvor det offentlige beveger seg tilbake til en verden med leverandørspesifikke formater.⁵⁸

⁵⁸Ekspertgruppens mindretall, (Lars Sjørgard), fremhever at sterkt offentlig krav om åpne standarder kan komme til å svekke innovasjonsevnen i deler av programvaremarkedet. Ulike bedrifter vil finne ulike løsninger. Sett i et slikt perspektiv vil for eksempel et generelt pålegg om kun å bruke åpne standarder lukke døren for fremtidig innovasjon basert på lukkede standarder. Det vil dessuten innebære at videreutvikling av programvare som i dag er basert på en lukket standard vil kunne hemmes.

Sjørgard finner derfor grunn til å advare mot et generelt pålegg om åpne standarder. Microsoft er en aktør som benytter lukkede standarder, men langt fra den eneste. Ofte vil det være tale om programvare skreddersydd for en bestemt brukergruppe. Ett spesifikt eksempel er programvaren *Scientific Workplace*, som har blitt benyttet av mange innenfor universitets- og høyskolemiljøene til å skrive artikler med matematikk. Apple, med sine Mac-produkter, er et annet eksempel på en aktør som benytter lukkede standarder.

Det kan være særlig grunn til å være bekymret for hva et generelt pålegg om åpne standarder vil bety for potensialet for at nye bedrifter kan utvikle ny programvare. Som beskrevet tidligere, kan en lukket standard bidra til at en bedrift kan kapre den fremtidige gevinsten forbundet med at et

4.1.4 Industristandarder

På områder der det ikke finnes åpne standarder utviklet i standardiseringsorganisasjoner, vil formater utviklet av enkeltleverandører til en viss grad kunne fylle rollen til en åpen standard. En forutsetning er at spesifikasjonen publiseres åpent uten krav om avgifter og begrensninger. Et eksempel er PDF-formatet fra Adobe. PDF er nå under standardisering av ISO, men Adobe styrte lenge utviklingen av formatet selv. Spesifikasjonen var åpent tilgjengelig og Adobe og andre leverandører har gjort programvare tilgjengelig på flere plattformer. Siden det ikke fantes alternativer, var PDF derfor et akseptabelt format for utveksling og lagring av informasjon, selv om formatet ikke faller inn under en streng definisjon av åpne standarder.

Microsoft sitt doc-format, som brukes av Microsoft Word, vil derimot ikke kunne betraktes som en åpen standard. Microsoft har aldri gjort spesifikasjonen åpent tilgjengelig, og annen programvare som ønsker å lese og skrive doc-filer vil måtte forske seg fram på egen hånd (noe OpenOffice og andre har gjort med varierende hell). Microsoft har heller ikke lansert programvare for å kunne lese doc-filer på konkurrerende plattformer som Linux.

I praksis er det ofte markedet som bestemmer hva som skal gjelde som standard. Industri-standarder, også kalt de facto standarder, får ofte sette rammer for hvordan ting gjøres fordi de løser spesielle industrielle utfordringer. Microsofts Word-format er for eksempel den ledende standard for tekstbehandlingsdokumenter selv om denne ikke er åpen.

4.2 Åpen kildekode

Begrepet *åpen kildekode* ("open source") er sentralt i diskusjoner omkring åpenhet og programvare. Begrepet er i stor grad relatert til en utviklingsfilosofi og kunnskapskultur som er vanlig blant programmerere. Det handler om at mange datastudenter tar med seg et tenkesett som er bygd på informasjonsdeling og at programvare er et fellesgode som tilhører alle som programmerer. Deres lojalitet går derfor i stor grad til fellesskapet og de som lager *vakre* programmer.

Sentralt i åpen kildekode-kulturen står fri utveksling av kildekode, fagfelleevaluering (under prinsippet om at da blir de fleste feil oppdaget) og estetiske kvalitetskrav som går på enkelhet, renhet og perfektjon. Dette ideologiske fundamentet tar mange programmerere

produkt oppnår suksess i betydningen stor utbredelse. Hvis en avskjærer muligheten for at noen bedrifter kan vokse og lykkes på grunnlag av en lukket standard, vil en da risikere at programvare som ellers ville blitt utviklet ikke vil bli utviklet? Hvis en går 20 år tilbake i tid, hva ville skjedd hvis en da innførte et generelt pålegg om en åpen standard? Det er vanskelig å si, men det ville i hvert fall ikke vært mulig for Microsoft – og en rekke andre selskaper som har valgt en lukket standard – å vokse frem slik vi har sett.

Derimot ser Sørgard gode grunner til at myndighetene griper selektivt inn mot de mest uheldige sidene ved en proprietær løsning. Da er det særlig grunn til å fokusere på de områder hvor den dominerende aktøren kan tenkes å legge hindringer i veien for at en ny plattform skal vokse frem. De tiltak Microsoft iverksatte mot Netscape på siste halvdel av 90-tallet er ett eksempel der det var grunn til å gripe inn. På tilsvarende måte er det grunn til å gripe inn mot tiltak Microsoft har iverksatt for å legge hindringer i veien for at servere blir en ny plattform. I den forbindelse er det en må betrakte noen av tiltakene som ble iverksatt mot Microsoft i både USA og nå nylig i EU. Det er tale om pålegg som gjør at Windows på visse områder kommuniserer bedre med andre operativsystemer, og dermed en selektiv form for bevegelse i retning av en åpen standard. Da har en unngått å gripe alvorlig inn i selve produktutviklingen, samtidig som en har fokusert mot de områder hvor en frykter at Microsoft har lagt hindringer i veien for en ny plattform.

med seg fra universitetet, og åpenhet blir noe som kjennetegner deres inntreden også i næringslivet. De fortsetter i de fleste tilfeller å bidra inn til det globale programmerings-fellesskapet, ofte med velsignelse fra arbeidsgiver. Samtidig er det klart at når de jobber for en arbeidsgiver begynner etter hvert næringslivets normer å sive inn, og mange må forholde seg til tidsfrister, et mer pragmatisk krav om *godt nok* og en viss beskyttelse av bedrifts-hemmeligheter.

Når en programmerer lager et program, skriver han eller hun *kildekode* i et språk som både kan leses av andre programmerere og datamaskiner. For å kunne kjøre programmet effektivt på en datamaskin, må kildekoden oversettes (*kompileres*) til *binærkode*. Binærkode er lesbar for datamaskiner, men ikke for mennesker. Når man kjøper eller laster ned programvare, får man vanligvis bare binærkoden til programmene. Ved også å distribuere kildekoden til programvare, vil andre programmerere kunne se hvordan programmet fungerer og gjenbruke kildekoden. Kommersielle programvareselskap distribuerer derfor sjelden kildekode, da dette kan gi konkurrenter innsyn i styrker og svakheter ved produktet, samt at andre kan lage lignende produkter.

Historikk

Gjennom datamaskinens relativt korte historie har det vært vanlig å distribuere kildekode til programvare. På 50- og 60-tallet var det datamaskinene, ikke programvare, som var hovedproduktet, og kildekode ble ofte distribuert sammen med maskinene. På 70-tallet ble de første personlige datamaskinene utviklet og entusiastene utvekslet programvare, inkludert kildekode, seg imellom. Operativsystemet Unix ble utviklet av telefonselskapet AT&T og kildekoden var enkelt tilgjengelig den første tiden.

Internett vokste fram som et universitetsnett på 80-tallet og universitetsmiljøene som ble knyttet sammen hadde en liberal holdning til utveksling av informasjon. GNU/Linux-prosjektet, som ble startet av Richard Stallman ved Massachusetts Institute of Technology i 1984, har lagt mye av det moralske og rettslige fundamentet for åpen programvare; kildekode var i følge GNU/Linux ikke lenger bare greit å ha, det var en rettighet man ikke kunne ta fra noen.

Rundt 1990 startet to prosjekter som har hatt stor betydning for framveksten av åpen kildekode: Ved forskningsinstitusjonen CERN utenfor Geneve startet Tim Berners-Lee *World Wide Web*-prosjektet som hadde som mål å gjøre utveksling av informasjon mellom forskerne enklere. For å gjøre det mulig for andre å sette opp nettstedet, ble kildekoden publisert og CERN oppfordret andre til å gjenbruke koden. Blant dem som fulgte oppfordringen var Marc Andreessen og Eric Bina som arbeidet ved forskningsinstitusjonen NCSA (The National Center for Supercomputing Applications) i USA. De utviklet programmet Mosaic som for alvor gjorde weben populær utenfor akademiske miljøer. Kildekoden til Mosaic ble også distribuert.

Omtrent samtidig skrev finske Linus Torvalds operativsystemet Linux. Han publiserte kildekoden på nettet, og andre programmerere kunne derfor hjelpe ham med utviklingen. Linux har siden vokst til å bli et stabilt operativsystem som også støttes av kommersielle bedrifter, i tillegg til entusiastene. Et eksempel er Linuxdistributøren Red Hat som er notert på den amerikanske teknologibørsen Nasdaq. Norske TrollTech har markert seg internasjonalt ved å utvikle Qt som gir Linuxprogramvare et innbydende grafisk grensesnitt. Det norske Skolelinuxprosjektet har oversatt Linuxprogramvare til bokmål og nynorsk og satt sammen en distribusjon tilpasset norske skoler.

4.2.1 Hva bør være definisjonen på åpen kildekode?

Mangfoldet innen utvikling av åpen kildekode er stort. Et nettsted som tilbyr lagringsplass for åpen kildekode-prosjekter, SourceForge, er vertskap for mer enn 88 000 prosjekter. Juridisk er likevel åpen kildekode oversiktlig, med kun en håndfull lisenser i bruk. En lisens beskriver, i juridiske termer, hvilke rettigheter og plikter brukere av åpen kildekode har. Blant de mest populære lisensene finner vi GNU Public License (GPL), Mozilla Public License (MPL), og MIT sin lisens. Felles for alle lisensene er at kildekoden

- er åpen for innsyn av alle;
- kan modifiseres av alle;
- kan gjenbrukes i annen programvare og
- fritt kan distribueres til andre.

Vi mener punktene over til sammen utgjør en god definisjon av begrepet *åpen kildekode* fordi de er allment aksepterte, og knyttet til bruk og distribusjon, så vel som innsyn. Det er nettopp aktiv utvikling som kjennetegner kunnskapskulturen rundt åpen kildekode. Merk at det ikke er noe krav om at åpen kildekode skal være gratis, ei heller at koden skal støtte åpne standarder (selv om dette ofte er tilfelle). Programvare som ikke kvalifiserer som åpen kildekode, kalles lukket kildekode, eller alternativt proprietær kildekode.

Programvareprodusenter kan dele lukket kildekode med andre, uten at koden dermed blir til åpen kildekode. Eksempelvis vil store kunder kunne kreve innsyn i kildekoden for å gjøre kvalitetskontroll. Microsoft har lansert begrepet *shared source* der myndigheter og enkelte andre grupper kan få tilgang til Microsofts kildekode. Det vil da knytte seg restriksjoner til tilgangen, som at kildekoden ikke kan distribueres til andre. *Shared source* kan derfor ikke betraktes som åpen kildekode.

4.2.2 Åpen kildekode og innovasjon

I dette kapittelet skal vi ta for oss en del momenter som vi mener er viktige i forbindelse med åpen kildekode og innovasjon.

Kunnskapsutvikling

Tankegangen bak åpen kildekode har sterke røtter i forskningsmiljøer. Den grunnleggende verdien i forskningsmiljøer er å dele kunnskap. I en forskningsverden er kunnskap i seg selv valutaen: Å gjøre sin kunnskap og løsninger tilgjengelig for andre "betales" ved at en får tilbake ny kunnskap og nye løsninger som en selv kan bygge videre på.

Dersom det selskapet som utvikler programvaren har eiendomsrett og dermed eksklusiv rettighet til programvaren, blir dette gjerne karakterisert som en *proprietær løsning*. En slik løsning kjennetegnes ved at selskapet holder selve kildekoden for programvaren hemmelig, eller ved at selskapet tar patent på programvaren. Proprietær programvare utvikles av selskaper som har som grunnleggende verdi å *beskytte* kunnskap for å bruke den som et konkurransefortrinn. Uten mulighet til å skape konkurransefortrinn (som man i neste instans kan tjene penger på) faller grunnlaget for kommersiell virksomhet bort. Det finnes imidlertid flere selskaper som har som forretningsmodell å generere verdier ved bruk av

åpen kildekode, som f.eks. Linux-distributøren Red Hat, eller databaseleverandøren MySQL AB i Sverige.

Det er et langsiktig samfunnsgode at kunnskap er allemannseie. Her spiller både åpen kildekode og åpne standarder en viktig rolle. Spesielt for skole- og utdanningsmiljøene er dette en sentral ressurs.

Markedsbehov

Kommersielle selskaper utvikler sine produkter basert på analyser av behov i markedet. Det brukes ofte store ressurser på å gjøre disse analysene. Resultatet er at produktene som utvikles er nyttige for mange.

Åpen kildekode har den fordel at den kan fungere som en byggekloss i utvikling av skreddersydde produkter, slik at god funksjonalitet blir tilgjengelig også for mindre brukergrupper, som svaksynte eller minoritetsspråklige, til en overkommelig kostnad.

Ofte profitterer utviklingen av åpen kildekode på den markedskunnskapen som er ervervet gjennom den kommersielle produktutviklingen, ved at funksjonalitet fra de kommersielle systemene kopieres i programvare basert på åpen kildekode. Men også det motsatte skjer – at proprietære systemer kopierer åpen kildekode.

Kostnadsbildet

De fleste bransjer har et element av skalafordeler, enten ved at produksjonsprosesser effektiviseres gjennom læring, produksjonsutstyr utnyttes mer effektivt eller komponenter i produktene blir billigere når man kjøper større volum. Programvarebransjen er ekstrem i så måte: Når en programvare først er utviklet, er distribusjonen billig. Og jo mer standardisert programvaren er, desto nærmere gratis er det å levere til ytterligere en kunde. I ytterste konsekvens er marginalkostnaden pr. solgte enhet kun knyttet til kostnaden ved å fakturere kunden.

I næringer hvor produktutvikling er viktig, vil det i utgangspunktet kunne være gode argumenter for å tillate proprietære løsninger. Ofte blir det vist til farmasøytisk industri, hvor utviklingskostnadene for å lage nye medisiner er svært store. For å sikre bedriftenes insentiver til å drive med forskning og utvikling, er det nødvendig at bedriften kan høste gevinstene fra nye produkter som utvikles. Av den grunn får bedriftene en beskyttelse for sitt produkt, ofte i form av patenter som varer i en viss periode. Sett i et slikt perspektiv bør en også kunne tillate proprietære løsninger innen programvareindustrien, nettopp fordi produktutvikling har vist seg å være viktig og sannsynligvis også vil være viktig i fremtiden.

Men det er grunn til å være varsom med å trekke en parallell mellom farmasøytisk industri og programvareindustri.⁵⁹ I farmasøytisk industri vil en ofte betrakte en oppdagelse som en diskret endring, der en plutselig har funnet en oppskrift for et nytt produkt. Vi har tidligere

⁵⁹Langlois, R. N. (2001): *Technological Standards, Innovation, and Essential Facilities: Towards a Schumpeterian Post-Chicago Approach*, i Jerry Ellig (red.): *Dynamic Competition and Public Policy: Technology, Innovation, and Antitrust Issues*, New York: Cambridge University Press, 2001, pp. 193-228. Her foretas det en drøfting av argumenter for og mot proprietære løsninger, med særlig fokus på effekten av standardisering.

vært inne på eksterne sjokk og plutselige skift i programvareindustrien. Denne type utvikling hører imidlertid til unntakene – som oftest er det tale om en kumulativ utvikling. Det innebærer at teknologien endres gradvis, der nye utviklere bygger videre på det andre har gjort. Dette er et argument for at flest mulig har tilgang til programmet, slik at de kan bidra til å utvikle det videre. Slik blir programmet utviklet, bit for bit.

I et marked med sunn konkurranse er sluttbrukerprisen en funksjon av utviklings- og produksjonskostnader. Det er en fordel for sluttbrukerne at utviklingskostnaden kan deles på mange. Fordelene forsvinner imidlertid dersom leverandøren blir en de facto monopolist.

Kostnadsbildet med åpen kildekode som utgangspunkt er annerledes. Utviklingskostnadene er spredt på flere aktører – ofte til og med brukeren selv – som på sett og vis utvikler produktet på dugnad. Basisfunksjonaliteten kan være gratis og mulighetene for påbygging etter egne behov er mange. Anskaffelseskostnaden er lav, mens vedlikeholds- og videreutviklingskostnadene kan trekke totalkostnaden opp.

Det klassiske eksempelet på dette er, som tidligere nevnt, operativsystemet Linux. En ikke-proprietær organisering av utviklingen har uten tvil vært vellykket, og dette er et eksempel på at innovasjon innen programvareindustrien ikke nødvendigvis krever proprietære løsninger. På den annen side innebærer ikke dette at all virksomhet knyttet til åpen kildekode er ukommersiell. Det har vokst frem en rekke selskaper som benytter åpen kildekode som grunnmodell og selger komplementære tjenester og dermed tilbyr kommersielt (mot betaling) en produktpakke hvor åpen kildekode er en viktig ingrediens.

Programvareindustrien

Standardiserte grensesnitt er avgjørende for programvareindustrien. Det gjør det mulig å gjenbruke programkode på tvers av operativsystemer og maskinvare, og gir programvareleverandøren et større nedslagsfelt og dermed lavere enhetskostnader. Dette gjelder selv-sagt enten det dreier seg om proprietære eller åpen kildekode-baserte utviklingsmiljøer.

Ett av Microsofts suksesskriterier har vært å sørge for at programvareleverandører bruker deres utviklingsplattform som byggekloss i å utvikle nye applikasjoner for sluttbrukerne. Dermed øker bruksområdene for Microsofts egne produkter. Programvareutvikleren på sin side får et stort antall potensielle brukere av sitt program.

Faren ved dette, sett med programvareindustriens øyne, oppstår når en aktør får *for* stor markedsmakt og finner det opportunt å konkurrere innenfor et område som tidligere har vært en arena for "påbyggere". Det er enkelt for en slik aktør å lage barrierer for sine nye konkurrenter, i form av endrede programmeringsgrensesnitt eller lukkede dataformater. Dermed virker de konkurrerende produktene dårligere enn aktørens egne. Dette er noe av det Microsoft har blitt anklaget for å gjøre.

Koordinering

De ulike komponentene i programvaren må fungere godt sammen. Dette kan være et argument for at én bedrift bør koordinere utviklingen og ha en proprietær rettighet og dermed full kontroll over utviklingen av programvaren. Det faktum at mange er involvert i utviklingen, det kumulative elementet ved programvareteknologi, kan delvis løses ved at denne bedriften går i allianse og partnerskap med andre aktører i forbindelse med produkt-

utviklingen. På den måten vil den ene bedriften styre utviklingen, men samtidig dra veksler på andre krefter som kan bidra til den skrittvis utviklingen mot en bedre programvare.

Problemet er å avgjøre når bedriften ikke bare iverksetter tiltak for å gjøre kundene mer tilfredse, men samtidig opptrer slik at det blir vanskelig for andre, potensielle konkurrenter å nå opp i konkurransen til skade for forbrukerne.

Konkurransen mellom Microsoft og Netscape illustrerer hvor vanskelig det er å skille mellom produktutvikling som er gunstig for kundene og produktutvikling som bidrar til å befeste en dominerende posisjon og dermed ikke nødvendigvis er gunstig på lang sikt for kundene. Den tette koblingen mellom dokumentformatet Doc og Windows 98 inneholdt en rekke gunstige egenskaper sett fra ståstedet til brukere av de to typene programvare. Samtidig var den tette koblingen et viktig element i Microsofts strategi for å unngå konkurranse.

Optimal innovasjon

Det er ikke nødvendigvis ønskelig med mest mulig innovasjon, men snarere optimal i stedet for maksimal innovasjon. Introduserer et selskap som Microsoft nye versjoner av sine programvarer for ofte? Problemet kan være at det blir for lite innovasjon fra de potensielle konkurrentene, fordi den dominerende bedriften har lyktes til de grader i å sikre sin dominans. Dermed er det nærmest nytteløst å prøve å utfordre den etablerte. I så fall er det grunnleggende problemet at den dominerende aktørens atferd reduserer innovasjon fra andre potensielle aktører. Den samlede innovasjonsaktiviteten skulle i så fall ideelt sett vært dreid i retning av mindre innovasjon fra den dominerende aktøren og mer enn tilsvarende økt innovasjon fra de potensielle utfordrerne.⁶⁰

På den annen side er det blitt påpekt at dersom en aktør virkelig oppnår en total dominans, uten frykt for å bli erstattet av en potensiell konkurrent, kan det lede til at den dominerende aktøren driver mindre FoU enn hva samfunnet er best tjent med. Grunnen er at et nytt produkt fra aktøren kun vil erstatte det eksisterende produktet, og dermed er det begrenset med incentiver for en ekte monopolist til å utvikle og introdusere nye produktvarianter.⁶¹ Hvis det derimot var to aktører som kjempet om dominans, ville hver av dem hatt sterke incentiver til å utvikle nye produkter for dermed å stjele kunder fra konkurrenten. Sett i et slikt perspektiv kan svært omfattende FoU iverksatt av Microsoft tolkes som at de frykter at de uten slik satsning vil mistet sitt hegemoni. I så fall tyder det på at den potensielle konkurransen er virksom i denne næringen.

Kan det også tenkes at den dominerende aktøren iverksetter *feil* type innovasjon? Vi kan tenke oss innovasjonen som et virkemiddel for å kunne selge "samme" produktet til kundene flere ganger (i form av nye versjoner), uten at det nødvendigvis er kundenes behov som driver utviklingen. Vi kan også tenke oss at selskapet lager bindinger mellom ulike programvarer ut fra ønsket om å gjøre det vanskelig for andre selskaper å nå opp i konkurransen, og ikke ut fra hensynet til hva kundene primært ønsker.

⁶⁰Segal, I. og M. D. Whinston (2003): *Antitrust in Innovative Industries* drøftes samspillet mellom dominerende aktør og potensielle utfordrere. <http://www.ios.neu.edu/papers/f2a3.pdf>

⁶¹K. J. Arrow (1961) var den første som drøftet dette aspektet, og dette er beskrevet i blant annet Motta, M. (2004): *Competition Policy. Theory and Practice*, Cambridge University Press, Cambridge, kapittel 2.4.

Den type forsøk på å hindre alternative plattformer å vokse frem er det myndighetene primært må fokusere på. Både hos amerikanske myndigheter og i EU er det eksempler på at det har vært grepet inn mot at Microsoft har prøvd å hindre operativsystemer som er knyttet til servere å vokse frem som alternative plattformer. Det har skjedd ved at en har pålagt Microsoft å frigjøre spesifikk informasjon som gjør kommunikasjon mellom ulike systemer mulig. Det er i realiteten å gå i retning av en åpen standard på visse selektive områder.

4.3 Programvarepatenter

Et patent er en kontrakt mellom en oppfinner og samfunnet forøvrig. I bytte mot å få enerett på utnyttelsen av en oppfinnelse for omlag 20 år, beskriver oppfinneren oppfinnelsen på en måte som gjør det mulig for andre å kopiere oppfinnelsen når patentet utgår. Hensikten med patentlovgiving er å gi oppfinnere et insentiv til idéskapning (i form av monopol), samtidig som også samfunnet får noe igjen (nye produkter basert på gode idéer, samt en beskrivelse av idéene som andre senere kan bruke). Sammen med opphavsrett på åndsverk er patenter en viktig del av feltet som på norsk kalles immaterielle rettigheter ("intellectual property").

Patenthistorien er full av interessante, men også kontroversielle patenter. I Norge er vi stolte av at Johan Vaaler i 1899 fikk patent på en papirklemme av metall, og at Thor Bjørklund i 1925 patenterte ostepølsen. Færre kjenner George Seldens patent på bensinbilen. Selden var patentadvokat og søkte om sitt patent da det virket klart at bensinbiler ville bli en suksess. Henry Ford var den eneste amerikanske bilprodusenten som nektet å lisensiere Selden sin "oppfinnelse", og Ford klarte å annullere patentet etter mange års kamp i rettsalen. Det pågår nå en debatt om programvare: Skal man kunne patentere data-programmer? Svaret på dette spørsmålet vil kunne være avgjørende for fremtiden til åpne standarder og åpen kildekode.

4.3.1 Argumenter for programvarepatenter

Før 1980 var det ikke mulig å patentere programvare. I 1981 slo amerikansk høyesterett fast (Diamond v. Diehr) at selv om programvare var en del av en oppfinnelse, skulle det være lov å patentere oppfinnelsen. Siden da har praksis på området blitt stadig mer liberal, og programvarepatenter utstedes i dag rutinemessig i USA og delvis i Europa.

I utgangspunktet virker det rimelig at oppfinnere som utvikler programvare skal kunne patentere sine oppfinnelser. En høyt utdannet programmerer burde ha samme mulighet for å beskytte sine gode idéer som en farmasøyt eller mekanisk ingeniør. Og siden både farmasøytisk og mekanisk industri har gjort store framskritt under et patentregime virker det rimelig å anta at også programvareindustrien vil blomstre med patenter.

I den vestlige verden utgjør landbruk og tradisjonell produksjonsindustri en synkende del av den økonomisk aktiviteten. Stadig mer av verdiskapningen er basert på kunnskapsindustri hvor idéer står i fokus. Patenter er den beste måte å sikre enerett til idéer, og patentering er derfor en måte å sikre at verdiene kommer idéskaperne til gode. Dette vil oppmuntre programmerere til fortsatt å tenke nytt.

Patenter skaper økonomisk aktivitet og bidrar derfor til å opprettholde økonomisk vekst. Det koster penger å søke om et patent, det koster penger å forsvare et patent, det koster penger å forsvare seg mot et patent, og det koster penger å lisensiere et patent. Et patent er et

produkt som kan kjøpe og selges i tråd med markedsliberalistisk tankegang. Å unnta programvare fra patentering vil kunne svekke vår økonomiske vekst.

4.3.2 Argumenter mot programvarepatenter

Et patent er et privat monopol. Det kan argumenteres for at monopol svekker konkurransen, da det hindrer andre å lage lignende produkter.

Det er mange grunner til at programvare ikke bør kunne patenteres. Mange av argumentene bygger på at programvare er annerledes enn andre felt hvor patentering er mulig. I farmasøytisk industri arbeider forskere i dyre laboratorier for å komme fram til et kjemisk preparat som kan beskrives på en måte som gjør det enkelt å reproducere. Programvare fungerer annerledes. En 17-årig entusiast med en PC på rommet kan ha mange gode idéer. Terskelen for å produsere gode idéer er langt lavere enn i andre industrier, og 20 år er ekstremt lang tid i et programvareperspektiv.

Det finnes en rekke eksempler hvor store aktører har tatt ut patenter på meget generelle «oppfinnelser» som brukes til å blokkere etablering av konkurrerende nye selskaper.⁶² Slike patenter kan lede til en ytterligere forsterkning av det digitale klases skillet og permanent utelukke utviklingsland fra å delta i informasjonssamfunnet.

Heller enn å tenke på et dataprogram som en oppfinnelse, kan en betrakte et dataprogram som en matematisk formel. Dette er i tråd med hvordan datamaskiner fungerer teknisk: Binærkode er et sett matematiske instruksjoner. Matematiske formler har aldri vært patenterbare og derfor bør heller ikke programvare kunne patenteres.

Å skrive programvare kan også sammenlignes med å skrive en symfoni: En bruker mange noter og idéer som har vært brukt tidligere. Dersom europeiske myndigheter hadde tillatt musikalske patenter på 1700-hundre tallet kunne Haydn, eller kanskje Mozart, patentert symfonien. Da hadde verden kanskje aldri fått hørt Beethovens symfonier.

Et for noen like tragisk scenario hadde vært om tekstbehandling eller regneark hadde blitt patentert omkring 1975. Den gang var det ikke mulig å få innvilget slike patenter, men med dagens amerikanske regler ville det sannsynligvis vært enkelt.

Et annet argument som ofte brukes mot programvarepatenter, er at kvaliteten på innvilgede patenter ofte er lav. Patentene er bevisst skrevet på en måte som gjør det vanskelig å forstå hva som egentlig er patentert og "oppfinnelsene" som skjuler seg bak tungt språk er ofte trivielle. Eksempelvis har idéen om en "elektronisk handlevogn" [EP19970201881 19961224] og et "elektronisk ordresystem" blitt patentert [US 6,356,874]. Programvarepatenter som favner vidt, fører også til at mange patenter er i konflikt med hverandre.

Det bør også bemerkes at kreativiteten blant programmerere alltid har vært høy og at patenter neppe er nødvendige for å sikre nyskaping. Tvert om, patenter vil hindre skapertrang da programmerere vil måtte bruke tid på å søke gjennom stor mengde patenter for å se om idéen de har fått allerede er patentert.

⁶²<http://ffii.org/>

4.3.3 Virkningen av programvarepatenter

Åpen kildekode, åpne standarder, og programvarepatenter er bygd på samme grunntanke: Ved å publisere en teknisk beskrivelse av hvordan programvare fungerer/kommuniserer, får både utvikler og samfunnet forøvrig fordeler. Tabellen under beskriver virkninger og insentiver:

	Hva publiseres?	Fordeler for utvikler	Fordeler for samfunnet
Åpen kildekode	Kildekode	Ære og berømmelse	Innsyn i og gjenbruk av koden
Åpne standarder	Spesifikasjon	Større marked	Økt konkurranse, bedre utveksling og lagring av informasjon
Programvarepatenter	Beskrivelse	20 års monopol	Nye produkter, en beskrivelse som andre kan kopiere etter 20 år.

Tabell 2: Virkningen av ulike teknologiske strategier

Den store forskjellen mellom programvarepatenter og de to andre mekanismene, er at eieren av et patent får et monopol på utnyttelsen av en idé i 20 år. Dette er i direkte konflikt med målsettingen til åpen programvare og åpne standarder hvor gjenbruk og interoperabilitet står i fokus.

Åpen kildekode vil særlig kunne hemmes av programvarepatenter, da prosjektene sjelden har penger til verken å søke patenter eller forsvare seg mot patentsøksmål. Således vil eksempelvis Linux være sårbar mot angrep fra kommersielle aktører som oppfatter Linux som en brysom konkurrent.

Åpne standarder vil også kunne bli skadelidende av programvarepatenter. Til en viss grad vil skade unngås ved at patentert teknologi ikke oppnår nødvendig støtte for å bli en åpen standard. Men i mange tilfeller er ikke patentet kjent når en teknologi standardiseres. Spesielt har bildestandarder blitt rammet av patenter i ettertid. Mosaic støttet to bildestandarder: GIF og JPEG. Patentkrav har senere blitt framsatt mot begge formatene. GIF har fått et patentfritt alternativ i form av PNG-formatet, mens JPEG fortsatt ikke har noe alternativ. For å unngå slike situasjoner i framtiden har World Wide Web Consortium (W3C) derfor satt krav til sine deltagere: Alle som er med på utvikling av W3C-standarder må love at eventuelle patenter en deltager måtte ha eller få på området skal lisensieres til andre uten kostnad.

Det er også eksempler på at programvarepatenter hindrer bruk av ny innovativ teknologi. Bildeformatet JPEG2000, som oppnår bedre komprimeringsresultater enn JPEG vha. såkalt "wavelet"-teknologi, er ikke i utstrakt bruk. Hovedgrunnen til dette er at algoritmene i JPEG2000 er patenterte og derfor ikke kan implementeres fritt.

EU vurderer i disse dager å justere regelverket for patenter på programvare. EU-parlamentet har under tvil akseptert at programvarepatenter skal tillates, men gjort flere viktige unntak. Eksempelvis vil det være tillatt å bryte et patent dersom det er nødvendig for å få data-systemer til å kommunisere. EU-kommisjonen mener slike unntak ikke er nødvendig og går – med støtte fra Nokia, Philips, Microsoft og andre markedsledere – inn for en liberal

patentpraksis som ligger tett opptil den amerikanske. Dette er omstridt og kan vise seg å befeste de maktstrukturer og den kontroll som visse programvareselskaper allerede har.

Det vil eksempelvis være uheldig om Microsofts monopoler blir ytterligere befestet med patenter. En ser i dag tegn til økende konkurranse innenfor områder der Microsoft står spesielt sterkt (operativsystemer for PC, nettlesere, kontorstøttesystemer), men patentvåpenet kan brukes mot nye konkurrenter. Microsoft har til nå kun brukt patenter mot andre i forsvarsøyemed. Mange patenter har blitt lisensiert uten kostnad i bytte mot løfte om patentfred. Microsoft har også måttet bruke mye penger på forlik for å unngå rettssaker i patentspørsmål, og har således selv mye å tape dersom programvarepatenter styrkes. Imidlertid har selskapet nylig signalisert endringer i patentstrategien ved bl.a. å åpne for lisensiering av sine patenter. En slik lisensiering vil neppe være effektivt uten at trusselen om søksmål fra Microsoft virker reell. Det er derfor grunn til å tro at Microsoft i fremtiden vil bruke patentvåpenet mer aggressivt, og dette vil kunne svekke konkurransen.

Patenter er ment å være incentiver til innovasjon. På området programvare viser forskning at det motsatte er tilfelle: Programvarepatenter blir en *erstatning* for forskning og utvikling i stedet for å føre til økt satsning på forskning og utvikling.⁶³

Patenter er ment å tjene både oppfinner og samfunnet forøvrig. Programvarepatenter synes å ha få positive virkninger for samfunnet. Det finnes derimot flere viktige negative sider: programvarepatenter er en trussel mot åpen kildekode, åpne standarder og fri konkurranse.

4.4 Tekniske sperrer

Programvarepatenter er juridiske sperrer for programvare. Programvare kan også begrenses av tekniske sperrer som hindrer tilgang til informasjon. Tekniske sperrer har to hovedformål: å hindre lesing av informasjon for uvedkommende, og å hindre kopiering av informasjon.

En type teknisk sperre som mange forbrukere møter, er kontroll ved avspilling av DVD-plater. For å kunne styre når filmer slippes i forskjellige markeder har filmindustrien utviklet et system med sonekontroll. Dersom du bor i en sone (f.eks. Europa) kan du ikke spille av en DVD-film kjøpt i en annen sone fordi produsenten av DVD-spillere har lagt inn en sperre mot å lese DVD-filmer fra andre soner. Dette er et eksempel på en lesesperre.

Mange nye CD-plater er ikke merket med det tradisjonelle CD-symbolet, men sier i stedet at CD-platen har en innebygd kopisperre. Produsenten av platen har da kodet innholdet på en måte som gjør den vanskelig lesbar i datamaskiner (som enkelt kan kopiere innholdet), men lesbar i de fleste CD-spillere. Dette tiltaket har vært omdiskutert, da flere kunder ikke har kunnet spille av lovlig kjøpte plater på sine stereoanlegg, mens kopisperren for datamaskinen har vist seg å være enkel å omgå.

Både lesesperrer og kopisperrer er styrt av programvare. En type programvare for å kunne styre i detalj hvordan informasjonen kan brukes er "Digital Rights Management" (DRM). I et DRM-system kan distributøren bestemme hvem som skal kunne lese informasjonen, om informasjonen skal kunne kopieres, eller kanskje hvor mange kopier som skal kunne lages. DRM representerer en mulighet for håndtering av de utfordringer flyten av elektroniske

⁶³J. Bessen, R. M. Hunt (2003/2004): *An Empirical Look at Software Patents*
<http://www.researchoninnovation.org/swpat.pdf>

dokumenter gir, særlig når det gjelder kommunikasjon av konfidensiell informasjon for eksempel av strategisk, teknologisk eller finansiell karakter. Brukere som mottar et dokument der DRM er aktivert, vil ikke kunne benytte seg av informasjonen ut over det som er forutsetningene satt av den som eier informasjonen.

I utgangspunktet virker det rimelig at den som har opphavsrett til et åndsverk skal kunne kontrollere hvordan åndsverket spres. Dette er en av to hovedideer i lov om opphavsrett til åndsverk. Den andre idéen er at også forbrukerne skal ha rettigheter i forhold til åndsverk. Eksempelvis er rettighetsperioden begrenset, og loven gir adgang til kopiering til privat bruk. DRM-systemer, samt andre lese- og kopisperrer, er derfor sperrer som fratar forbrukere rettigheter de har i følge norsk lov. En kan ha sympati overfor opphavsmenn som er engstelige for virkningen av ulovlig fildeling på internett, men tekniske sperrer fortjener en videre debatt før introduksjon.

EU sitt InfoSoc-direktiv (også kjent som EUCD-direktivet) omhandler tekniske sperrer. Direktivet introduserer en beskyttelse av tekniske sperrer. Det er ikke lenger bare åndsverket som er beskyttet av loven, men det blir også forbudt å åpne tekniske sperrer. Det virker sannsynlig at sonefrie DVD-spillere vil bli forbudt etter Infosoc-direktivet (selv om både norske og danske myndigheter gjort det klart at de ikke betrakter programvare for å spille av DVD-filmer på Linux-maskiner som et brudd på direktivet). InfoSoc-direktivet styrker ensidig rettighetene til opphavsmann på bekostning av forbrukernes rettigheter.

Elektronisk Forpost Norge (EFN) har lansert et lovforslag⁶⁴ om tekniske sperrer som fortjener seriøs behandling. Forslaget introduserer en leserett til digitale data man har rettmessig tilgang til. For eksempel, dersom man kjøper en CD-plate i et kryptert format, har man som forbruker lov å dekryptere innholdet selv om dette innebærer å bryte tekniske sperrer. Å bryte tekniske sperrer vil ofte være nødvendig for å kunne ta sikkerhetskopier eller overføre innholdet til neste generasjon lagringsmedium. På denne måten opprettholder leseretten rettigheter forbrukere har hatt før tekniske sperrer ble introdusert.

4.5 Oppsummering: teknologivalg avgjør åpenhet

Dersom vi skal vurdere åpne standarder og åpen kildekode ut fra hensyn til *demokrati, effektivitet og fremtidsrettethet*, ser vi at åpne standarder kan ha en viktig demokratisk funksjon. Slike standarder fremmer ytringsfrihet og tilgjengelighet fordi de sikrer fri utveksling av informasjon. Også effektivitetshensynet ivaretas ved åpne standarder, fordi man får enklere tilgang til offentlige tjenester, enklere koordinering og stabile rammevilkår.

I forhold til fremtidsrettethet har vi spesielt tatt for oss innovasjonsaspektet. Vi har sett eksempler og argumenter for at åpne standarder både kan fremme og hemme innovasjon. Flertallet i ekspertgruppen mener det finnes gode eksempler på at åpne standarder fremmer innovasjon, og at denne strategien også har vesentlige fordeler i form av økt konkurranse og lesarhet. Ekspertgruppens mindretall (Sørgard) mener at sterke føringer fra det offentlige i forhold til åpne standarder kan ha uheldige effekter.⁶⁵

⁶⁴<http://efn.no/lovforslag.html>

⁶⁵For en helhetlig og samlet fremstilling av mindretallets syn, se Sørgard, Lars (2004): *Markedsdynamikk i dataindustrien*, SNF arbeidsnotat 38/04 tilgjengelig på ftp://ftp.snf.no/Arbeidsnotat/2004/A38_04.pdf

Åpen kildekode er en alternativ utviklingsmodell for programvare. Den er demokratisk i den forstand at alle kan ha innsyn og bidra til koden. Den er effektiv og fremtidsrettet siden kode kan gjenbrukes nå og i fremtiden. Åpen kildekode kan også ses på som en konkurransemodell. I motsetning til proprietær programvare, er ikke kostnaden ved åpen kildekode knyttet til lisenser (fordi utviklingskostnadene er delt på "alle"), men til migrasjonskostnader og konsulenttenester.

Åpen kildekode må sies å være en fremtidsrettet strategi, i det at den stimulerer en kunnskapskultur som innebærer deling av kunnskap, slik at vi får en desentralisert utviklingskompetanse. Slik deling må kunne betraktes som et langsiktig samfunnsgode.

I tillegg til åpen kildekode og åpne standarder, har vi sett på programvarepatenter, som kan ses på som en juridisk sperre for programvare.

Det er mange grunner til at programvare ikke bør kunne patenteres. Teknologirådets ekspertgruppe peker blant annet på at programvare er annerledes enn andre felt hvor patentering er mulig: Det er enklere å komme fram til nye produkter, og etter 20 år vil de fleste programvareprodukter være utdatert. I tillegg er kvaliteten på innvilgede patenter er ofte lav.

Patenter er ment å tjene både oppfinner og samfunnet forøvrig. Ekspertgruppens flertall mener at programvarepatenter synes å ha få positive virkninger for samfunnet, mens det derimot er flere viktige negative sider: Programvarepatenter er en trussel mot åpen kildekode, åpne standarder og fri konkurranse.

Tekniske sperrer brukes ofte til å frata forbrukere rettigheter de har i følge norsk lov. Ekspertgruppen mener derfor at innføring av InfoSoc-direktivet i Norge bør revurderes.



Kapittel 5 Offentlige strategier for et åpnere samfunn

5.1 Fire hovedstrategier for det offentlige

Vi har i 0 gått gjennom de ulike teknologiske strategiene som ligger til grunn for ulike typer programvare – proprietær programvare, programvare basert på åpne standarder og programvare med åpen kildekode.

I dette kapitlet tar vi utgangspunkt i åpne standarder og åpen kildekode, og går nærmere inn på hvordan det offentlige kan forholde seg til disse teknologiske strategiene.

Det er i den sammenhengen viktig å ta i betraktning at det offentlige har mange roller i forhold til programvareutvikling: Som tjenestetilbyder for borgerne; som storkunde; som regulerende myndighet, blant annet for konkurranse; som tilrettelegger for næringsutvikling og innovasjon; og som utvikler av kompetanse. Dette medfører at myndighetene har flere hensyn å ta, knyttet til å bidra til et mer demokratisk, effektivt og fremtidsrettet samfunn.

Teknologirådets ekspertgruppe har valgt å gå nærmere inn på fire ulike strategier det offentlige kan velge blant:

- La markedet råde
- Aktiv konkurransepolitikk
- Preferansepolitikk
- Pilotprosjekter

Mens den første strategien betyr minimal inngripen i markedet, innebærer de siste tre strategiene ulike grader av offentlige tiltak. Disse tre intervensjonsstrategiene kan brukes hver for seg, eller i kombinasjon med hverandre.

Internasjonalt er programvare blitt satt høyere på agendaen i løpet av de siste to årene. Mange lands myndigheter er opptatt av marked og innovasjon, demokrati og offentlig tilgang til informasjon, samt å kommunisere på tvers av nye og gamle IT-systemer. Flere land har en klar politikk som enten sier at det offentlige kun skal sørge for å gjøre økonomisk sunne investeringer eller at de aktivt skal forhindre markedssvikt. I noen tilfeller, slik som Brasil og Peru, går land spesifikt inn for bestemte typer programvare. Vi skal her spesielt referere til nærliggende eksempler som EU, Danmark og Storbritannia.

5.2 La markedet råde

Å la markedet få råde vil si at en stoler på at markedskreftene bringer frem de beste teknologiske løsningene uten at det offentlige trenger å gripe inn utover vanlig konkurransepolitikk. Markedet selv skal styre, og det offentlige kan, som alle andre kunder, fritt velge det de mener er de beste løsningene basert på sine egne kriterier. Disse kriteriene må være tilgjengelig for alle markedsaktører, og det skal være reelle anbudsrunder.

Når man velger en slik strategi, kan man enten uttrykke eksplisitt at programvarevalg kun skal være basert på markedskreftene (noe som upresist kalles "value for money"), man kan velge ikke å ha noen politikk, og man kan avstå fra å gjøre inngrep i form av preferansepolitikk eller IKT-standardisering.

Dette innebærer at man ser på det offentlige primært som en kunde og at programvare er en hvilken som helst vare. Det offentlige bør i henhold til dette ikke mene noe om åpne standarder eller åpen kildekode, men først og fremst avveie brukershensyn for den enkelte institusjon.

Denne typen politikk blir typisk brukt av markedsliberale myndigheter, men også av myndigheter som ikke har tatt et aktivt grep om sin IKT-politikk generelt. De fleste multinasjonale IKT-selskap, så vel som bransjeorganisasjoner, argumenterer for denne typen ikke-intervensjon. Hovedargumentet er at samlede brukerkostnader ("Total Cost of Ownership") er det viktigste, og at rettferdige handelsvilkår er bra for markeder. Teknologinøytralitet er et grunnleggende prinsipp.

Et av målene med en slik offentlig politikk er å gi eksisterende industri et stabilt rammeverk for å kunne planlegge sin virksomhet og oppnå vekst. Det beste eksempelet på slik politikk finner vi i USA.⁶⁶

Det positive med denne politikken er at større programvareleverandører får stabile rammevilkår og forutsigbarhet ved at IKT-markedet ikke påvirkes negativt av skiftende offentlig politikk.

Det negative er for det første at markedssvikt, monopol- og kartellvirksomhet ikke reguleres i tilstrekkelig grad. Vi peker i kapittel 2.1.1 på at nettverkseffektene i programvaremarkedet vil medføre inngangsbarrierer for å nå inn på markedet, særlig for mindre aktører, og videre kan ende i monopoldannelse.

For det andre er det et viktig poeng at programvare *ikke* er en hvilken som helst vare, og at det offentlige *ikke* bare er en helt vanlig kunde. Tvert imot er det offentlige en privilegert kunde med et tungt ansvar for å treffe valg på vegne av borgerne. Disse valgene skal ikke bare gi gode løsninger for dagens behov, men også sikre fremtidig valgfrihet i et fungerende marked, og ikke minst tilgjengelighet til nødvendige offentlige tjenester på lengre sikt.

⁶⁶For en oversikt over amerikansk programvarepolitikk: <http://www.teknologiradet.no/html/572.htm>
Unntaket er Massachusetts, som har markert seg sterkt ved å gå inn for bruk av både åpne standarder og større bruk av programvare som benytter åpen kildekode. Se <http://www.state.ma.us/itd/ITAcquisitionPolicy.htm>

5.3 Aktiv konkurransepolitikk

Til nå har det offentlige strategier i stor grad vært å la markedet råde. Som beskrevet tidligere, er et resultatet av dette sterke monopoliseringstendenser i markedet der Microsoft har fått et tilnærmet monopol på operativsystemer. Dette er en problematisk situasjon, ikke minst for prisutvikling og innovasjon. Det er derfor en økt oppmerksomhet rundt konkurransepolitiske tiltak både nasjonalt og innen EU (se Kapittel 2 og Kapittel 3).

En aktiv konkurransepolitikk innebærer en aktiv strategi for å korrigere monopoliserings-tendenser i markedet for å bidra til å styrke konkurransen. Tiltak for en slik strategi vil i hovedsak være knyttet til å gripe inn mot fusjoner og samarbeid som truer konkurransen, og å gripe inn mot at dominerende aktører misbruker sin markedsrett. Men programvare-markedene er ikke nasjonale, de store aktørene er multinasjonale selskap, og konkurranse-fremmende tiltak på internasjonalt plan vil være nødvendig.

En aktiv konkurransepolitikk kan imidlertid vise seg utilstrekkelig dersom målet er å fremme åpne standarder. Virkemidler som preferansepolitikk og pilotprosjekter kan da være virksomme supplerende strategier.

5.4 Preferansepolitikk

Preferansepolitikk for det offentlige favoriserer visse typer programvare, standarder, forretnings- eller utviklingsmodeller. Preferansepolitikken varierer etter i hvilken grad den tvinges gjennom, er frivillig, eller kun er anbefalt. Vi kan definere fire strategier for preferansepolitikk:

- *Gulrot*; dvs. å premiere og gi insentiver til å følge mandatet.
- *Pisk*; dvs. å tvinge preferansen gjennom, enten i lovs form, i forskrift eller ved annet type vedtak.
- *Preken*; dvs. å formane, informere, etablere koordineringskomitéer og eventuelt ha websider med tjenester som understøtter politikken.
- Sammenlikning ("*Benchmarking*"), dvs. å satse på at når en aktør ser hva andre gjør, vil han ønske å være like god eller bedre.

Myk preferansepolitikk kan gjelde kun i enkelte sektorer eller på visse typer funksjoner (for eksempel servere etc.). Dette kan innebære en aktiv regulering av hvilke filformater det offentlige skal bruke i sin kommunikasjon og ellers aktivt sette standarder som IKT-bransjen må forholde seg til, enten overfor det offentlige som kunde, eller også for å operere i Norge.

En mer aktiv strategi vil være å ha preferanse for bestemte typer programvare eller måter å utvikle programvare på (utviklingsmodell), å føre rettssaker og/eller gi bøter for å slå ned på monopolist- og kartellvirksomhet. Dette er en modell EU ser ut til å gå inn for, noe kjennelsen i rettssaken mot Microsoft 24. mars 2004 viser.

5.4.1 Preferansepolitikk for åpne standarder

Preferansepolitikk for åpne standarder betyr å foretrekke, påby eller anbefale bruk av åpne standarder ved offentlig innkjøp og bruk av programvare.

Aller først vil dette gjelde formater for ekstern dokumentutveksling, særlig mot publikum, men også mellom etater. Mange land, som Argentina, Danmark⁶⁷, Frankrike, Tyskland, India, Italia, Nederland, Peru, Storbritannia⁶⁸ og Sør-Afrika har klargjort dette i en eller annen grad.⁶⁹

Som nevnt i kapittel 4.1, kan innføring av åpne standarder bidra til å korrigere markedssvikt og generelt fremme mulighetene for sunn konkurranse i programvaresektoren. Videre er det viktig for offentlige myndigheter å møte utfordringen med at IT-systemer ikke snakker sammen. Derfor blir *interoperabilitet* en nøkkelprioritet og preferansepolitikk for åpne standarder et viktig virkemiddel for å oppnå dette.

Britisk politikk på åpne standarder

Britene har satt standardisering høyt – og koordinerer dette gjennom *Head of eGovernment* og de såkalte *eChampions* (sterke skikkelser fra samfunnslivet som møtes jevnlig for å følge utviklingen og foreslå tiltak). Britene har etablert et rammeverk for interoperabilitet (kalt "e-GIF"). Dette rammeverket definerer teknologipolitikk og spesifikasjoner som styrer informasjonsflyt på tvers av parlament, regjering og offentlig sektor generelt. Spesifikasjonene dekker interoperabilitet, dataintegrasjon, tilgang til elektroniske tjenester og innholdsstyring. E-Gif har fått god mottakelse både fra industrien og andre brukere.

EUs politikk på åpne standarder

EU har definert som strategiske mål å bli den mest konkurransedyktige og dynamiske kunnskapsbaserte økonomien i verden, med bærekraftig økonomisk vekst, flere og bedre arbeidsplasser og bedre sosialt samhold innen 2010. Nøkkelen til å oppnå dette målet er eEurope-handlingsplanen. Planen fokuserer på tilgjengelighet og legger åpne standarder til grunn for et rammeverk for interoperabilitet, og oppfordrer til økt bruk av programvare basert på åpen kildekode.

Interoperabilitet defineres slik av Manuela Finetti, sjef for *Interchange of Data between Administrations* (IDA)-enheten i generaldirektoratet for næringsliv ("DG Enterprise"):

"Interoperabilitet betyr kommunikasjon mellom systemer, prosesser og mennesker for å gi sømløse og kundeorienterte offentlige tjenester".⁷⁰ Det handler altså ikke bare om programvaresystemer men også om å dele informasjon. Finetti henviser til at teknisk interoperabilitet forutsetter bruk av åpne standarder og viser til IDAs skille mellom:

- Teknisk interoperabilitet som handler om å knytte sammen IT-systemer og programvare, definere åpne grensesnitt, standarder, formater og protokoller.
- Semantisk interoperabilitet som handler om å sørge for at meningsinnholdet i den informasjon som utveksles bevares og forstås av de applikasjoner som er involvert.

⁶⁷<http://www.teknologiradet.no/html/573.htm>

⁶⁸<http://www.teknologiradet.no/html/565.htm>

⁶⁹For en oversikt pr juni 2004, se <http://www.teknologiradet.no/html/562.htm>

⁷⁰Interoperabilitet defineres slik av Manuela Finetti, sjef i IDA-enheten i EU:
<http://www.publicservice.co.uk/pdf/europe/autumn2003/manuelaf.pdf>

- Organisatorisk interoperabilitet som handler om å strømlinjeforme organisasjonsmessige prosesser og informasjonsarkitektur til organisasjonelle mål – ikke bare tekniske – og å avhjelpe samspillet mellom tekniske og organisatoriske hensyn.

Den nye eEurope-planen forutsetter utviklingen av et rammeverk for interoperabilitet for å fremme pan-europeisk tjenesteutvikling basert på åpne standarder og bruk av programvare basert på åpen kildekode. Dette kalles *European Interoperability Framework (EIF)*.⁷¹ EU holder nå på å utvikle dette rammeverket for å oppfylle kravet i *Handlingsplanen for eEurope 2005* som sier at EU-kommisjonen skal utgi et slikt rammeverk innen 2003 for å støtte leveransen av pan-europeiske digitale forvaltningstjenester ("e-government services") til borgere og bedrifter.

Rammeverket skal supplere en pan-europeisk dimensjon heller enn erstatte nasjonale initiativ, skal oppdateres regelmessig og skal dekke alle tre formene for interoperabilitet. Et førsteutkast av rammeverket ble gjort tilgjengelig på IDAs hjemmesider 20. januar 2004. Et sentralt prinsipp for dette rammeverket er å bruke åpne standarder, som defineres slik:

*"Kostnaden for bruk av standarden er lave og er ikke til hinder for å ha tilgang til den; standarden er publisert; standarden er adoptert på basis av en åpen avgjørelsesprosedyre (konsensus eller majoritetsbestemmelse), opphavsrettighetene til standarden ligger hos en ikke-kommersiell organisasjon som har en fullstendig fri tilgangs-policy og det er ingen begrensninger knyttet til gjenbruk av standarden" (s.11).*⁷²

I rapporten *Linking up Europe: the Importance of Interoperability for eGovernment Services* (2003) fra IDA-enheten i generaldirektoratet for næringsliv ("DG Enterprise") skisseres interoperabilitet som en uutnyttet kilde til økonomisk vekst. Det handler om å begynne å dele og gjenbruke offentlig informasjon, systemer og administrative prosesser både innen og på tvers av offentlig sektor i EU. Det kan være store synergieffekter også gjennom frivillige avtaler.

I den såkalte FLOSS-rapporten⁷³ fra 2002 anbefaltes åpne standarder for å sikre interoperabilitet, redusere avhengigheten av spesielle systemer, så vel som å redusere leverandøravhengighet ("vendor lock-in").

5.4.2 Hva kan Norge gjøre for å fremme åpne standarder?

Dersom Norge skulle drive aktiv preferansepolitikk for åpne standarder, kunne et tiltak være å følge Danmark og kreve at store leverandører åpner sine formater.

En forsiktig begynnelse som likevel ville gå rett til kjernen av problematikken er vår hovedanbefaling:

a) Gjør offentlige dokumenter som utveksles eller arkiveres tilgjengelig i åpne formater

Alle offentlige dokumenter som utveksles eller arkiveres skal være tilgjengelig i et åpent format (for eksempel HTML, XML eller PDF⁷⁴) – i tillegg til eventuelle lukkede formater.

⁷¹<http://europa.eu.int/ISPO/ida/jsps/index.jsp?fuseAction=showDocument&parent=news&documentID=2033>

⁷²EIF førsteutkast (20 januar 2004): <http://europa.eu.int/ISPO/ida/export/files/en/1674.pdf>

⁷³<http://www.infonomics.nl/FLOSS/report/>

For å sikre en effektiv omlegging, kan man lage enkle overvåkingsverktøy, eller gjøre bruk av eksisterende teknologi. For eksempel kan myndighetene kreve at offentlige nettsteder må passere World Wide Web Consortium sin valideringstest.⁷⁵ Vi kan tenke oss at Riksrevisjonen kan være kontrollorgan for å sjekke at påleggene følges.

b) Krav om åpne standarder til allmennkringkastere

Alle medier bør oppfordres til å bruke åpne standarder. Det bør i tillegg stilles særlige krav til allmennkringkasterne. NRK bør pålegges å gjøre sine web-sendinger og tjenester tilgjengelig på åpne standarder. Ved eventuell fornyelse bør krav om åpne standarder inkluderes i TV2 og Kanal24s konsesjoner.

c) En mer aktiv preferansepolitikk

En kan videre vurdere om all programvare det offentlige kjøper inn bør følge åpne standarder – men konsekvensene må nøye gjennomgås. En slik omlegging må for eksempel gi programvareleverandører tid til å tilpasse sine produkter til åpne standarder. Det kan også være aktuelt å begrense kravet om åpne standarder til å gjelde programvare som brukes i kommunikasjon ut fra offentlige etater.

d) Forhandlinger med enkeltaktører

Staten kan aktivt forhandle med enkelte større leverandører for å få dem til å åpne sine formater dersom de ønsker et fortsatt kundeforhold til det offentlige. Et mulig scenario ville da være følgende: *Å kreve at Microsoft dokumenterer formatene doc, xls og ppt, samt måten de bruker xml-standarder på.*

Microsoft sine formater er i utstrakt bruk i norsk offentlig sektor. For at informasjon som lagres i disse formatene skal kunne leses av annen programvare, er det nødvendig at Microsoft dokumenterer sine formater. Åpne formater gjør det lettere å utveksle dokumenter og stimulerer konkurransen i markedet for lesing og behandling av digital informasjon.

Et slikt krav ville være relativt vidtgående og det er tvilsomt om man vil vinne frem på kort sikt. På den annen side stilte danske myndigheter krav til Microsoft om å få åpnet XML-skjema (en type informasjon nødvendig for datautveksling) – og lyktes. Uten et pålegg om åpne standarder vil det være vanskelig å etterprøve om det industrielle aktører kaller "åpne formater" virkelig er åpne og dermed tilgjengelig slik at produkter fra ulike leverandører kan kommunisere med dem. Det kan oppstå forskjellige meninger om hva f.eks. HTML eller PDF er. I et slikt scenario kan en enkelt dominerende aktør i realiteten definere hva som skal være en standard.

e) Følge EU-sporet (interoperabilitet)

Dette betyr å satse bredt på interoperabilitet som en teknisk, organisasjonsmessig og semantisk utfordring.

Et tiltak kan være å *opprette et statlig websted* (linket fra Moderniseringsdepartementets hjemmesider) med oversikt over tre kategorier av standarder: *Grønne* (standarder som skal

⁷⁴PDF er utviklet og kontrolleres for en stor del av Adobe. Ekspertgruppen mener likevel PDF er egnet, jfr diskusjonen i 4.1.4

⁷⁵<http://validator.w3.org/>

brukes), *Gule* (nye standarder mottatt til klassifisering), *Røde* (avviste og gamle standarder).⁷⁶

Grønne standarder bør være obligatoriske for bruk i norsk offentlig forvaltning etter et nytt regjeringsvedtak, og bør nedfelles i forskrift til forvaltningsloven. Kun slik kan vi oppnå at alle sektordepartement samt kommuner forplikter seg til å følge disse standardene ved offentlig innkjøp og dokumentutveksling.

Norge bør også jobbe med *semantisk interoperabilitet*. I EU bruker man begreper som "life events/citizen events" (å ta førerkort, å gifte seg, å flytte, å besøke biblioteket, å søke skole/universitetsplass, å avgi politierklæring) og "business episodes" (å kjøpe bil, å rapportere om sin næringsvirksomhet, å starte et selskap). Vi kunne også snakke om "eGovernment episodes" (å gjøre offentlige innkjøp) eller eksplisitt fokusere på *borgernes* møter med offentlig forvaltning (å fylle ut selvangivelsen, å søke om trygd, å fylle ut offentlige informasjonsskjema/statistikk, å flytte til utlandet).

Det er også stort behov for slik interoperabilitet innen Norge. Det viktigste ville være å bli enige om hvilke typer data som det er behov for å utveksle mellom kommuner, eller hva som ville gi effektivitetsgevinster. Mye av problematikken rundt "life events" eller "business episodes" gjelder også ved flytting internt i Norge.

Det foregår noe arbeid innenfor dette området i dag, men ikke eksplisitt med interoperabilitet for øye. Det norske prosjektet LivsIT⁷⁷ arbeider med klassifikasjon av offentlig informasjon med utgangspunkt i livssituasjoner, og også arbeid knyttet til Kostra har ført til felles forståelse og beskrivelse av begreper. I *Arkitektur for elektronisk samhandling i offentlig sektor* foreslår man å utarbeide en enhetlig beskrivelse av informasjonselementer av stor betydning for det offentlige. Man bør også jobbe i tett samarbeid med EU, særlig *Interchange of Data between Administrations*-enheten i Generaldirektoratet for næringsliv ("DG Enterprise").⁷⁸

Teknisk sett er det behov for at informasjonsutvekslingen av semantiske data (data med meningsinnhold) skjer med utgangspunkt i XML metadata. Dette trenger både Norge og EU å jobbe videre med.

f) Økt bestillerkompetanse

Interoperabilitet krever gjennomført bruk av åpne standarder i flere ledd i offentlig forvaltning. Valg av programvare og tildeling av utviklingsoppdrag har en avgjørende

⁷⁶Jfr. tysk arbeid med interoperabilitet, men i noe forenklet utgave: (http://kbst.bund.de/Anlage304417/Saga_2_0_en_final.pdf), side 24.

⁷⁷<http://www.livs.it.info/>

⁷⁸Her er noen viktige formål med et slikt arbeid: 1. Speide etter viktige områder der Norge bør speile EUs standardisering, for eksempel (a) trygdesystemet ("social security"/mutualité), (b) statistikk-området - i samarbeid med SSB, (c) offentlig paneuropeisk arbeidsformidling via jobbsøkportaler etc. eller mer ambisiøst (d) - one-stop-shop for offentlige skjema i Europa samt (f) ulykke-/risikohåndtering vha. geografiske informasjonssystemer, (g) politisamarbeid og (h) språkstøtte; 2. implementere slik standardisering; 3. utvikle norske "best practices" som sparer skatte kroner pga. effektivisering eller nye tjenester; 4. foreslå nye typer EU-standardisering som frigjør borgere som beveger seg i EU og EØS-området (f.eks. mellom Spania og Norge), eller standardisering av brukergrensesnitt samt innhold/språk på offentlige nettsteder samt rutiner for toll- og politisamarbeid, velferds- og trygdeordninger etc.

betydning for bruk av standarder. Økt bestillerkompetanse vil være avgjørende for å gi bestillere oversikt over konsekvenser og valgmuligheter.

Kjøpsavtaler som brukes til offentlig innkjøp kan ha en avgjørende betydning. Statskonsult har allerede utviklet standard avtaler for IKT-anskaffelser til staten⁷⁹. Den store kjøpsavtalen⁸⁰ er ment å regulere blant annet at standard maskin- og programvare som passer sammen og passer til kundens eksisterende anlegg/opplegg (riktig konfigurering og grensesnitt). Foreløpig omtaler denne avtalen ingen konkrete standarder for interoperabilitet. En videre utvikling av en slik avtale kunne bygge på et interaktivt skjema som fylles ut av bestilleren i samråd med leverandøren. Skjemaet kan knyttes til praktiske råd og en oppdatert liste over statens anbefalte standarder. En slik applikasjon for generering av avtaler kan også brukes til å samle materiale for statistikk over offentlige innkjøp. Oppgaver som må løses for å realisere dette inkluderer:

- Sammenstille og vedlikeholde samling av aktuelle åpne standarder og anbefalinger for bruk i offentlig forvaltning i Norge, evt. Norden. Både standarder og retningslinjer for bruk av standarder vil utvikles over tid.
- Koordinere med standardiseringsorganisasjoner og andre land. Her finnes allerede en rekke organisasjoner som har en slik oppgave innen sitt område, f.eks. Post- og teletilsynet eller Sosial- og helsedirektoratet. Her bør koordineringen styrkes også i Norge.
- Sammenstille tilbakemeldinger fra brukere. I Danmark, England og Tyskland legges forslag til retningslinjer for bruk av standarder ut på høring. Både bedrifter og allmennheten oppfordres til å bidra med kommentarer. En liknende åpen prosess som sikrer bred forankring brukes f.eks. ved utviklingen av World Wide Web Consortiums standarder.
- Tilrettelegge verktøy for test, evt. sertifisering, av implementering av standarder. Utvikle løsninger for sertifisering av standardprodukter og skreddersøm. Det vil for datatrafikken tilsvare den funksjon som biltilsynet har for veitrafikken eller Det Norske Veritas for sjøfarten.

En rekke videre tiltak basert på incitamenter, direktiv og lovverk kan brukes til oppfølgingen.

- For å sikre at norske kommuner kan tilby gode interoperable løsninger bør digitale veibevevilgninger knyttes til krav om konformitet til standarder. Vi kan tenke oss at Riksrevisjonen kan brukes som kontrollorgan for dette.
- En annen vei er valgt av amerikanske myndigheter hvor all føderal offentlig informasjon skal utformes i henhold til Section 508. Avvik fra disse retningslinjer kan lede til søksmål mot det offentlige med erstatningskrav fra dem som ikke kan bruke informasjonen eller tjenesten.

⁷⁹<http://www.statskonsult.no/it/ssa/>

⁸⁰<http://www.statskonsult.no/it/ssa/storkjop.ht>

- Et videre virkemiddel kan være regelmessig kartlegning og publisering av rikets tilstand hva gjelder bruk av standarder og tilgjengelighet til offentlig informasjon og tjenester. Dette kan fremme en utviklende konkurranse mellom f.eks. kommuner.
- Priser for beste offentlige nettsted deles ut av Norge.no sammen med kommunal rapport. Alle Norske kommuner på nett ble rangert i 2001 og 2003 etter kvalitet på nettsidene.⁸¹ Statistisk Sentralbyrå har kartlagt noen aspekter ved den digitale norske forvaltningen, så som bruk av hjemmeside eller kommunal IKT-strategi.

5.4.3 Preferansepolitikk for åpen kildekode

Fra den politiske verktøykassen – med gulrot, pisk, preken eller sammenlikning som mulige verktøy – handler det også på dette området om å bestemme hvor strengt mandatet skal være, og hvordan det skal ta hensyn til det faktum at en stor del av programvaremarkedet er på proprietær plattform. Mandatet vil være knyttet til innkjøpspolitikken, og det kan da lyde som følger: "Det offentlige bør prioritere programvare med åpen kildekode i alle sine fremtidige innkjøp".

Å stimulere programvare basert på åpen kildekode reduserer sannsynligheten for å bli låst til enkeltleverandører ("vendor lock-in"), og kan brukes som et midlertidig tiltak, siden markedet i dag ikke gjør at aktørene konkurrerer på lik linje.

Videre blir det ofte henvist til at bruk av åpen kildekode kan spare det offentlige for store lisenskostnader og kostnader forbundet med stadige oppgraderinger.

Argumenter for preferanse for åpen kildekode

Noen argumenter for slik preferansepolitikk er:

- Det offentliges ansvar for å fremme fellesgoder (ikke-eksklusive og ikke-rivaliserende) løses best ved programvare med åpen kildekode. Stikkord kan være å beskytte opphavsrettigheter (som finnes for programvare uavhengig av lisens), men begrense patenter.
- Både myndigheter, brukere og industri bør ha mulighet til å fritt velge hvilke programvareløsninger, leverandører eller læremodeller de vil bruke. Slik er det ikke i dag, og derfor må åpen kildekode stimuleres slik at denne utviklingsmodellen kan komme på like fot, eller helst ta over for den proprietære.
- Vi må unngå avhengighet til utenlandske leverandører. Myndighetene må hjelpe små og mellomstore bedrifter som del av en nasjonal innovasjonsplan. Lokale behov kan også være knyttet til språk, tilpasning og nærhet til brukeren.
- Sikkerhet: Det finnes noe belegg for at Linux foreløpig er mindre utsatt for virusangrep, hacking etc.
- Brukerne får mer makt når de selv kan velge teknologiplattform.

⁸¹<http://www.norge.no/kvalitet/kvalitet2003/kvalitet.asp>

- Utdanningshensyn: Elever og studenter kan eksperimentere, velge mellom teknologier og lære hvordan systemer er oppbygd fra grunnen av.

Problematiske aspekter ved preferansepolitikk for åpen kildekode

Utfordringer ved en slik politikk kan være:

- Å gjøre rede for migrasjonskostnader (økonomiske).
- Å forstå brukernes behov (sosiologiske). Disse kan variere avhengig av hvilket segment du snakker om, deres IKT-kompetanse, hva slags nasjonal infrastruktur det er snakk om, og hvorvidt politikken inkluderer en arbeidspakke og et budsjett for opplæring av lærere, byråkrater, instruktører og konsulenter.
- Hva en skal gjøre med kontorstøttesystemer. Hvis brukere må leve med flere systemer samtidig (for eksempel Microsoft Office så vel som OpenOffice.org) – hvordan kan dette være brukervennlig?
- Sikre at leverandører av programvare basert på åpen kildekode er tilgjengelige, profesjonelle og mulige å kontakte for support. Det er riktig at det per i dag finnes mange slike leverandører, men de er ofte små, og uten en lang merittliste.⁸²
- Innovasjonsaspektet: Nye aktører med unik teknologi kan trenge å beskytte sine opphavsrettigheter. Mange konkurranseøkonomer vil også hevde at monopolister som har vært "flinke", fortjener sin ledende posisjon og ikke bør straffes unødige.

EUs politikk på åpen kildekode

EUs politikk på offentlig innkjøp, konkurranse og interoperabilitet har utviklet seg over en periode på mange år, og er innvevd i hele den europeiske politikken på informasjons-samfunnet og næringslivet ("information society and enterprise"), så vel som i konkurranse-politikken. POSS-rapporten⁸³ fra 2002 foreslår en strategi som går ut på at offentlig programvare utvikles som programvare med åpen kildekode for å fremme samarbeid mellom de ulike medlemslandene. Rapporten fastslår at all offentlig programvare bør være basert på åpen kildekode for å sikre gjenbruk og gi mulighet for lokale tilpasninger. Rapporten fremmer ikke bare deling av programvare men også av kunnskap og kompetanse.

Interchange of Data Between Administrations (IDA)-rapporten *Linking up Europe* oppfordrer videre til økt bruk av programvare basert på åpen kildekode i offentlig forvaltning for å stimulere vekst i Europa. På EU-toppmøtet om informasjonssamfunnet i Geneve i desember 2003 ble det understreket hvor viktig det er at programvare basert på åpen kildekode konkurrerer på lik linje med proprietær programvare.⁸⁴

I *Europeisk rammeverk for interoperabilitet (EiF)* knyttes spørsmålet om åpen kildekode relativt nært til spørsmålet om interoperabilitet. Å "vurdere nytten av programvare med åpen kildekode" er ett av åtte hensyn EU-kommisjonen mener bør gjelde for alle digitale

⁸²Noen unntak er: OpenOffice.org og MySQL som er produkter med stor brukermasse og hvor brukere kan få support. Ergo Ephorma har også begynt å støtte OpenOffice.org⁸² med fagsystemet ePhorte.

⁸³<http://europa.eu.int/ISPO/ida/export/files/en/1115.pdf>

⁸⁴http://www.itu.int/wsis/documents/doc_multi.asp?lang=en&id=1161|1160

forvaltningstjenester i paneuropeisk sammenheng.⁸⁵ De foreslår en preferansepolitikk som lyder slik "programvare med åpen kildekode [...] er i sin natur åpent tilgjengelige spesifikasjoner og det at kildekoden er tilgjengelig fremmer åpen, demokratisk debatt rundt spesifikasjonene, noe som gjør dem både mer robuste og mer interoperable. Programvare med åpen kildekode korresponderer i seg selv til dette rammeverket [om interoperabilitet, red. anm.] og bør vurderes og favoriseres ved siden av proprietære alternativer ("should be assessed and considered favourably alongside proprietary alternatives"), (s.11).

Dansk politikk på åpen kildekode – myk preferansepolitikk?

Det danske teknologirådets rapport *Open source software – i den digitale forvaltning (2002)*⁸⁶ fokuserte på mulige besparelser ved bruk av programvare med åpen kildekode i det offentlige. Rapporten ble kritisert for ikke å ha tatt i betraktning at å beregne IKT-kostnader over tid, samt at utgiftene ved migrasjon vris fra lisensbetaling til opplæring/konsulent-honorarer.

Teknologirådets vurdering av preferansepolitikk på åpen kildekode

Teknologirådets vurdering er at det foreløpig er uklart hvilke konsekvenser en sterk preferansepolitikk for åpen kildekode ville føre med seg. Det er blant annet derfor vi mener norske myndigheter bør sette i gang pilotstudier. På sikt gir en sterk preferansepolitikk for åpen kildekode imidlertid ikke rettferdig konkurranse i markedet. En viss stimulering av konkurrenter til rådende monopolister kan imidlertid forsvares, særlig om disse er viktige i nasjonal innovasjons- og verdiskaping.

5.5 Pilotprosjekter med åpen kildekode

Pilotprosjekter er viktig for å forstå effekten av nye idéer og satsninger. Bruk av pilotprosjekter tar utgangspunkt i spørsmålet det er vanskelig å gjøre rene teoretiske vurderinger av, og som må utredes gjennom erfaring med bruk. Hvordan vil det hele virke? Det viktigste når en vurderer å sette i gang slike pilotprosjekter er følgende:

- Hvor bredt/smalt skal en gå inn? – hvilke sektorer skal være med?
- Hva er det presise målet med pilotprosjektet?
- Hva skal piloten inneholde og hva er viktige vurderingsparametere?

Pilotprosjekter med programvare med åpen kildekode trengs slik at det offentlige kan skaffe seg erfaring med levetidskostnader ("Total Cost of Ownership"), brukerbehov og funksjonalitet. De beste eksemplene på slik politikk finner vi i Danmark⁸⁷ og i Storbritannia.⁸⁸ Andre steder der det offentlige har ønsket å vinne erfaringer med åpen kildekode gjennom pilotprosjekt er Brasil, Hong Kong, Italia og USA (Austin, Texas).

I prosjektene er det viktig å ta hensyn til brukerne og ikke presse store bekymringer over på dem. God tid til testing, erfaringsutveksling og tilbakemelding er viktig.

⁸⁵De øvrige er: tilgang, flerspråklighet, sikkerhet, personvern, subsidiaritet, bruk av åpne standarder samt bruk av multilaterale løsninger.

⁸⁶http://www.tekno.dk/pdf/projekter/p02_open-source-rapport.pdf

⁸⁷<http://www.teknologiradet.no/html/573.htm>

⁸⁸<http://www.teknologiradet.no/html/565.htm>

Britiske pilotprosjekter med åpen kildekode

Storbritannia kjører nå pilotprosjekter med programvare basert på åpen kildekode i sju offentlige forvaltningsenheter. De såkalte "OSS Trials" gjennomføres ved hjelp av IBM og Sun microsystems og koordineres av *The Office of Government Commerce* (OGC) og den britiske *Head of eGovernment*. Målet med pilotene er å se hvorvidt programvare med åpen kildekode er et godt alternativ til proprietær programvare – og potensielt bidra til bredere konkurranse, bedre "kost/nytte" og "øvrige goder".

Britene har implementert EU-initiativer som for eksempel eEurope-initiativet om å "fremme programvare basert på åpen kildekode" og har laget et omfattende policydokument på bruk av åpen kildekode.⁸⁹ Fire prinsipper ligger til grunn:

- "UK Government will consider OSS solutions alongside proprietary ones in IT procurements. Contracts will be awarded on a value for money basis."
- "UK Government will only use products for interoperability that support open standards and specifications in all future IT developments. "
- "UK Government will seek to avoid lock-in to proprietary IT products and services."
- "UK Government will consider obtaining full rights to bespoke software code or customisations of COTS (Commercial Off The Shelf) software it procures wherever this achieves best value for money."
- "UK Government will explore further the possibilities of using OSS as the default exploitation route for Government funded R&D software."

Danske pilotprosjekter med åpen kildekode – "desktop trials"

Målet for dansk programvarestrategi er: "Å øke konkurransen i programvaremarkedet og øke kvalitet og koherens i offentlige programvareløsninger. Å støtte opp under offentlige institusjoners muligheter til å kjøpe programvare ut fra prinsippet om maksimal verdi for pengene sikrer dette".⁹⁰

Dansk programvarestrategi tar følgende hensyn ved innkjøp:⁹¹ a) best og billigst uansett programvaretype, b) konkurranse, uavhengighet og valgfrihet og c) sammenheng og fleksibilitet.

Danskene har satt i gang seks piloter på "desktop-software, herunder open source".⁹² Målet for forsøkene er å "vurdere funksjonalitet og omkostningsnivå ved drift af PC arbeidsplasser i den offentlige sektor. "

Evalueringsmodellen er basert på tilbakemeldinger fra konsulenter, leverandørgruppen, en ekspertgruppe, en rekke fokusgrupper samt offentlig høring. Det som registreres er

⁸⁹<http://www.ogc.gov.uk/index.asp?id=2190>

⁹⁰<http://www.oio.dk/software/english>

⁹¹Ifølge <http://www.oio.dk/files/vtu.pdf>

⁹²<http://www.oio.dk/software/piloter>

omkostninger, brukertilfredshet og verdi av investeringen. En kokebok med erfaringer skal bli sluttresultatet.

Prosjektet blir gjennomført i samarbeid med de danske kommunenes interesseorganisasjon, KL,⁹³ og Amtsrådsforeningen. Roskilde kommune vil undersøke hvorvidt en open source basert skrivebordsapplikasjon kan fungere sammen med kommunens andre systemer. Aalborg Handelsskole vil utprøve Open Office i skolens administrasjon. Næstved kommune vil utprøve Star Office. Psykiatrisk Hospital, Århus Amt vil rulle ut kontorpakken Star Office fra Sun til over 2.000 brukere. Forbrugernes Hus vil utprøve Open Office-pakken. Justitsministeriet deltar med et prosjekt med oppgradering til nyeste versjon av Microsoft-programvare. Som hjelp i prosessen bruker danskene aktivt *Interchange of Data between Administrations (IDA) open source migration guidelines*, utarbeidet av EU-kommisjonen i 2003.⁹⁴

Norske pilotprosjekter på åpen kildekode – en mulig modell

Teknologirådets vurdering er at pilotprosjekter vil være et godt tiltak for Norge dersom disse kriterier blir oppfylgt: a) institusjonene har selv valgt å delta i pilotprosjektet, b) det er incentivmidler knyttet til prosjektet, c) IDA migration guidelines samt danske og britiske erfaringer benyttes aktivt og d) pilotene følges nøye av politisk ledelse i de aktuelle institusjoner og erfaringer blir satt i system.

I forhold til de danske pilotene ville det være ønskelig å få med noen litt større forvaltningsenheter og kommuner – for å få en realistisk sjekk av mulige effekter. Dette vil kreve større investeringer enn de 1.5 million danske kroner som var avsatt til hver pilot i Danmark.

5.6 Oppsummering – tilgjengelige strategier i programvarepolitikken

I dette kapittelet har vi sett på noen virkemidler det offentlige kan bruke i programvarepolitikken: Man kan la markedet råde, drive aktiv konkurransepolitikk, drive preferansepolitikk og gjennomføre pilotprosjekter.

Teknologirådets ekspertgruppe mener det offentlige bør ta et aktivt grep om programvarepolitikken, og anbefaler derfor preferansepolitikk for åpne standarder. Flertallet i ekspertgruppen mener all programvare det offentlige kjøper inn bør følge åpne standarder, mens mindretallet (Sørgard) advarer mot å føre en streng preferansepolitikk hva angår åpne standarder.

Interoperabilitet krever gjennomført bruk av åpne standarder i flere ledd i offentlig forvaltning. Valg av programvare og tildeling av utviklingsoppdrag har en avgjørende betydning for bruk av standarder. Økt bestillerkompetanse vil være avgjørende for å gi bestillere oversikt over konsekvenser og valgmuligheter.

Ekspertgruppens anbefaling er at Norge fører en myk, sektor- og funksjonsspesifikk preferansepolitikk for åpen kildekode (i skole-, utdannings- og forskningssektoren samt ved bistand til utviklingsland) særlig gjennom å stimulere til FoU rundt fagsystemer (for etater), regnskapssystemer og variert kontorprogramvare, så vel som stimulere til mer bruk av programvare med åpen kildekode.

⁹³<http://www.kl.dk>

⁹⁴http://www.oio.dk/files/OSS_migration_guideline.pdf

For å få til det siste mener Teknologirådets ekspertgruppe at det offentlige kan erverve verdifull erfaring og innsikt, særlig knyttet til levetidskostnader, brukerbehov og funksjonalitet, gjennom å kjøre pilotprosjekter med åpen kildekode i ulike forvaltningsenheter og kommuner.

I arbeidet med rapporten (spesielt 0 og Kapittel 5)⁹⁵ oppsto det en dissens i ekspertgruppen. Mindretallet (Lars Sjørgard) er uenig i flertallets hovedsynspunkter, og tar forbehold om de deler av innholdet som leder fram til flertallets konklusjoner.

Sjørgard mener at sterke føringer fra det offentlige i forhold til bruk av åpne standarder kan ha uheldige effekter. For det første kan det være problematisk på kort sikt for dagens brukere, rett og slett fordi en svært stor andel av dagens programvare i offentlig sektor ikke følger åpne standarder. For det andre kan det være uheldig på lang sikt, fordi det kan hindre en fremtidig bruk av programvare som kan vise seg ha mange gode egenskaper men som ikke er basert på åpne standarder. Det advares derfor mot å føre en streng preferansepolitikk hva angår åpne standarder.

⁹⁵For en helhetlig og samlet fremstilling av mindretallets syn, se Sjørgard, Lars (2004): *Markedsdynamikk i dataindustrien*, SNF arbeidsnotat 38/04 tilgjengelig på ftp://ftp.snf.no/Arbeidsnotat/2004/A38_04.pdf



Kapittel 6 Modernisering – en åpnere forvaltning med IKT

6.1 IT i offentlig forvaltning

I januar 2002 la regjeringen frem sitt program for modernisering i offentlig sektor. Visjonen er en offentlig sektor ”med aktiv deltakelse på demokratiske arenaer, med likeverdige og individrettede tjenester av høy kvalitet, med en effektiv ressursbruk og hvor effektive offentlige tjenester skal bli et konkurransefortrinn for næringslivet”.⁹⁶ IKT er et viktig virkemiddel for å oppnå dette.

Ifølge regjeringens IKT handlingsplan, den såkalte eNorge-planen, skal det satses på bedre organisering og effektiv oppgaveløsning, brukerorienterte elektroniske tjenester, forenklet innrapportering og offentlig sektor som krevende kunde. Tre satsningsområder er fremhevet:

- innrapportering, gjenbruk og tilgjengelighet av data;
- kunnskapsforvaltning og
- elektronisk signatur/PKI (“Public Key Infrastructure”).

Det er viktig ikke å la programvarepolitikk kun være et spørsmål om teknologisk åpenhet. Programvarepolitikk må handle om reell åpenhet. Et viktig aspekt i så måte er hvordan digital saksbehandling stiller nye krav til personvern og innsyn rundt rettigheter, lovverk og saksbehandlingsrutiner.⁹⁷ Dette er viktige hensyn som må ligge til grunn for en demokratisk programvarepolitikk, og som vi gjerne refererer til som *gjennomsiktighet*. Dette er et område vi skal sette fokus på i dette kapitlet.

6.2 Digital lovanvendelse

Offentlig forvaltning styres blant annet ved hjelp av lover og forskrifter. En viktig gruppe regler regulerer hvordan økonomiske ressurser skal beregnes og fordeles i samfunnet, for eksempel rettigheter knyttet til trygdeytelser, sosiale støtteordninger og plikter til å betale skatter og avgifter m.v.

Lovspråket er imidlertid ikke særlig hensiktsmessig til å beskrive regnestykker. Slike beregningsoppgaver har derfor lenge vært uttrykt ved hjelp av programmeringsspråk. Lov- og forskriftsregler som er oversatt til datamaskinprogrammer, kan sies å være fortolket ”en

⁹⁶http://www.enorge.org/modules/module_109/publisher_view_product.asp?iEntityId=1061&iC=5

⁹⁷Schartum, Dag Wiese (2002): *Fra rettsstat til rettsautomat. I Digital makt*, Gyldendal Akademisk, Schartum, Dag Wiese (1993): *Rettsikkerhet og systemutvikling i offentlig forvaltning*, Universitetsforlaget

gang for alle”, fordi fortolkningene i programmet anvendes på alle saker. Vi får med andre ord automatisert rettsanvendelse av regler som beskriver hvordan skatter og trygder m.v. skal beregnes.

I de følgende seksjonene skal vi se på hvordan datasystemer blir brukt til både automatisering av saksbehandling og innhenting av opplysninger som skal brukes i saksbehandlingen. I kapittel 6.2.3 illustrerer vi hvordan dette kan være problematisk i tilfeller som ikke følger tradisjonelle mønstre, og de påfølgende kapitlene tar for seg klagerett og problemer knyttet til at de rettslige beslutningene som tas ikke er forankret i forvaltningsloven. I kapittel 6.2.6 ser vi hvordan endringer i forutsetningene for reglene systemene bygger på, kan gi store konsekvenser.

6.2.1 Automatisering av tolkning og vilkårsprøving

Datamaskinen kan ikke bare foreta beregninger. Den kan også programmeres til å behandle logiske operasjoner, for eksempel prøve om alternative og kumulative vilkår er oppfylt eller ikke. Enkelte lover vil ha særlige bestemmelser for enkelte samlivsformer, som ekteskap, partnerskap eller samboerskap. Har loven for eksempel ”samboer” som et kriterium, kan dette sees som et spørsmål om felles bosted, varigheten av forholdet, felles barn osv., og spørsmålet om noen er ”samboer” kan forstås som et spørsmål om slike delvilkår er oppfylt.

Datamaskinen kan programmeres til å foreta vilkårsprøvingen. Forutsetningen er at det blir skrevet et program som uttrykker de samme vilkårsstrukturene som lovteksten inneholder. Ved hjelp av automatiserte beregninger og prøving av om vilkår er oppfylt, kan en viktig del av arbeidet som er knyttet til anvendelsen av rettsregler utføres av datamaskinsystemer.

Automatisering kan gi flere gevinster: Saksbehandlingen går raskere, og det er mulig å vedta lovregler som er langt mer omfattende og komplekse enn de det er mulig å tenke seg dersom mange funksjonærer skulle sitte og gjøre de samme operasjonene i hver enkelt sak. En viss grad av automatisert anvendelse av lovgivningen er en forutsetning for grunnpilarer i velferdsstaten, som for eksempel skatte- og trygdelovgivningen. En annen automatiseringsgevinst er at lovanvendelsen kan bli riktigere enn før. Dataprogrammet kan riktignok inneholde feil, men har koden først fått et korrekt rettslig innhold og systemet inneholder korrekte data, blir alle avgjørelser korrekte.

6.2.2 Innhenting og utveksling av saksopplysninger

Automatisering av beregninger og vilkårsprøvinger som beskrevet ovenfor, hører langt på vei ”gamle dager” til. Ønsket om å effektivisere offentlig forvaltning leder også i retning av å automatisere resten av lovanvendelsen. For selv om vi har programmer som uttrykker på hvilke vilkår en ytelse skal gis og hvordan den skal beregnes, må det fremdeles være noen som samler inn opplysningene forvaltningen trenger for å treffe vedtak i hver enkelt sak. Beregningsmåten er klar, men tallene i hver enkelt sak og alle andre opplysninger må skaffes og registreres i systemet.

Tradisjonelt skaffer en nødvendig beslutningsgrunnlag ved å la den enkelte som saken gjelder fylle ut skjemaer og legge ved dokumentasjon på at opplysningene er til å stole på. Forvaltningen har deretter overført opplysningene fra papirskjemaer til ”skjerskjemaer”. Deretter har datamaskinsystemet overtatt og behandlet opplysningene i tråd med den aritmetikk og logikk som er lagt inn i systemene.

Altinn-prosjektet⁹⁸ er spennende i så måte. Altinn er et samarbeid mellom Brønnøysund-registrene, Lånekassen, Statistisk sentralbyrå (SSB) og Skattedirektoratet om digital innrapportering. Systemet er ambisiøst og relativt brukervennlig, men er likevel kun for personer med god erfaring i bruk av IKT. Selv om det finnes informasjonssider, kontaktpunkter og brukerstøtte, er dette en meget avansert tjeneste som ikke alle vil kunne benytte. Altinns omtale av begrepet "utfyllerrolle" er relevant her: "Fyller ut informasjonen i skjema og sender inn dersom signering ikke er påkrevd." I Altinn presiseres det også at datasystemer kan ha utfyllerrolle.

Du kan altså delegere deler av arbeidet over til datasystemet. Dette kan enten bety at systemet du skal legge data inn i husker informasjonen om deg, eller at ditt eget datasystem kommuniserer direkte med Altinn og overfører data for deg. Man kan tenke seg tilfeller der tidligere feilinformasjon eller beklagelige historiske fakta blir lagret i systemet. Dette vil kunne få konsekvenser ved for eksempel skatteutmåling, uten at den vanlige bruker av Altinn kanskje er klar over det.

En lovregel inneholder beskrivelser av de saksopplysninger vi må ha tilgang til for å anvende bestemmelsen. Det går for eksempel frem at for å fastsette en ytelse må en kjenne personens "inntekt". Men "inntekt" har ikke samme betydning i enhver sammenheng. Dersom vi leter etter definisjoner, vil vi finne flere som i større eller mindre grad overlapper hverandre. Skal IKT besørge utveksling av saksopplysninger over nettet, er det derfor ikke nok med en teknologisk infrastruktur. I tillegg må det utvikles en "informasjonsinfrastruktur", dvs. en struktur av felles og klare definisjoner av de ulike opplysningstypene som kreves for å la datamaskinene overta hele lovanvendelsen. Her finnes noen ansatser som f.eks. Legal XML.⁹⁹

6.2.3 Behov for klare begrepsdefinisjoner

Lovverket inneholder et mylder av vage begreper som betegner saksopplysninger forvaltningen trenger for å treffe avgjørelser. Dette er opplysninger som skal beskrive de aspektene ved våre liv som etter loven er relevante når forvaltningen skal ta stilling til hvilke plikter og rettigheter vi har. For at maskinelle og effektive rutiner for kommunikasjon av saksopplysninger skal kunne utformes, må slik vaghet reduseres ved å fastsette mest mulig bindende og generelle definisjoner av hver opplysningstype. "Samboer", "inntekt", "forsørger" m.v. må altså bety det samme enten det er Rikstrygdeverket, Skattedirektoratet eller andre deler av forvaltningen som skal bruke dem for å treffe avgjørelser.

Eksempel: Forsørger Fredrik

Dersom Fredrik ønsker å oppnå forsørgertillegg fra Lånekassen, er det avgjørende at de opplysningene om ham som Lånekassen benytter for å avgjøre saken gir "riktig" resultat. La oss si at de lar spørsmålet om "forsørging" bli avgjort uten at det er lagt bestemte føringer med hensyn til hvilke momenter som det kan være relevant å legge vekt på. En slik situasjon gir rom for å trekke frem forhold som er helt spesielle for Fredrik, og som kan hevdes å være relevante for spørsmålet om hans forsørgerrolle: Barnet bor ikke hos ham, men hos hams foreldre som bor i samme hus. Barnet spiser imidlertid alle måltider hos Fredrik, han kjøper alle klær og drar alltid på ferie med det.

⁹⁸Altinns hjemmesider: <https://www.altinn.no/cms/1044/altinn/>

⁹⁹<http://xml.coverpages.org/legalXMLWG.html>

Dersom man skulle ha laget strømlinjeformede og effektive beslutningsrutiner, ville ikke Lånekassen ha spurt Fredrik, men i stedet "oversatt" kriteriet "forsørging" til dels et spørsmål om 1) farskap, 2) felles bolig og 3) barnets alder. Alle disse opplysningene kunne Lånekassen innhente fra maskinlesbare kilder, og på et nanosekund eller tre ville spørsmålet om forsørging være avgjort uten diskusjon. Fordi Fredrik ikke har felles bolig med barnet, ville han ikke bli tildelt forsørgertillegget, noe han ville synes det var skrekkelig urettferdig fordi han *faktisk* oppfattet seg som forsørger.

6.2.4 Klagerett ved digital saksbehandling

Kan offentlig forvaltning frata brukeren sine forhandlingsmuligheter, som i eksempelet med Forsørger Fredrik, ved å etablere en maskinell rutine som henter inn den informasjonen som forvaltningen anser som riktig og tilstrekkelig? Forvaltningen kan her langt på vei kan legge til rette for standardiserte, maskinelle rutiner som tilsynelatende setter den involverte parten til side. Slik loven er i dag kan man imidlertid ikke fratas muligheten til å påstå at den maskinelle rutinen gir galt resultat. Fredrik vil derfor kunne klage, og på den måten få gjennomslag for at han/hun likevel må regnes som "forsørger".

6.2.5 Rettslige beslutninger i det skjulte

I beslutningssystemene fastlegges det "en gang for alle" hvordan lov- og forskriftstekster skal forstås. Det betyr at den forståelsen som nedfelles i datamaskinprogrammer normalt vil bli lagt til grunn i alle fremtidige saker. Rettsreglene fortolkes med andre ord ikke ut i fra hver enkelt sak, men på basis av en rekke tenkte situasjoner som deltakerne i systemutviklingsarbeidet legger til grunn for sine analyser. Det som skjer er en forvandling av regeltekster som inneholder det norske språks vagheter og unøyaktigheter, til programkoder som har et utvetydig regelinhold. For hver tolkningsløsning som velges, er det en eller flere alternative muligheter som velges bort, dvs. det "forhandlingsrommet" som ligger i reglene fra lovgivers og departementers hånd fjernes.

Transformeringen fra flertydige til entydige regler kan ses som en "beslutning" i den forstand at de representerer en lang rekke valg som faktisk får innvirkning på alle saker som blir behandlet av det maskinelle systemet. Siden det er tale om avgjørelser av rettslige spørsmål, må utvikling av slike systemer delvis sies å inneholde *rettslige beslutningsprosesser*. Vi kan kalle disse beslutningene for "rettslige systemavgjørelser".

For andre, tradisjonelle rettslige beslutningsprosesser gjelder det visse saksbehandlingsregler som skal sikre at den enkeltes interesser blir ivaretatt. Forvaltningsloven inneholder således regler om hvilke framgangsmåter som må følges når det skal treffes enkeltvedtak og forskrifter. Rettslige systemavgjørelser *likner* enkeltvedtak, forskrifter og instruksjoner om rettsanvendelsen, men kommer ikke inn under saksbehandlingsregler i forvaltningsloven og ellers. De som treffer avgjørelser om hvilke regler som skal anvendes på saker som beslutningssystemene behandler, er med andre ord ikke bundet av noen spesielle framgangsmåter som kan sikre at brukernes interesser blir ivaretatt. Den tolkningen av en lovtekst som et enkeltindivid mener er riktig, kan med andre ord formfritt velges bort av det forvaltningsorganet som treffer de rettslige systemavgjørelsene.

Historisk riss – utvikling av rettslige IKT-systemer

Hvem treffer beslutningene om hvilket rettslig innhold beslutningssystemene skal ha? Tidligere var det vanlig at det var teknologer og brukerrepresentanter som tok de nødvendige avgjørelsene. De avdelinger hos forvaltningsmyndighetene som formelt var

ansvarlige for den rettslige kvaliteten av saksbehandlingen, var ofte koblet så løst til systemutviklingsarbeidet, at de i realiteten ikke hadde oversikt og direkte innflytelse på de mange rettslige avgjørelser som ble truffet. En viktig årsak har trolig vært at juristene i direktorater og departementer tidligere ikke var brukere av "EDB". De systemløsningene som ble introdusert fra 1980-årene var til bruk for "førstelinjen", dvs. for lokalkontorer i møte med publikum. Representantene for brukerne av slike systemer lærte seg teknologien og satt med betydelig erfaring, noe som gjorde dem egnet som samtalepartnere for teknologene som visste hvordan beslutningssystemer kunne utvikles. De ansvarlige juristene derimot, var uten kunnskap og erfaring med teknologien.

I løpet av de siste 10 årene har det skjedd en tydeliggjøring av ansvarsforholdene ved at fagavdelingene med juristene har blitt "systemeiere", og de har fått en klarere definert rolle i utviklingsarbeidet. Fremdeles er det imidlertid grunnlag for å hevde at ansvarlige fagavdelinger i mindre grad har oversikt over og innflytelse på rettslige systemavgjørelser enn på tradisjonelle instruksjoner og forskrifter.

Usynlige beslutningstakere

Når forvaltningen treffer enkeltvedtak på tradisjonelt vis, har vi som kunde eller part i saken en saksbehandler å holde oss til. Er vi uenige i den lovforståelsen han eller hun har lagt til grunn for sitt vedtak, kan vi ta kontakt og argumentere for et annet syn. Beslutningstakeren er med andre ord synlig for oss, og derfor mulig å forholde seg til. Hver rettslige systemavgjørelse bestemmer hvordan alle enkeltsaker av en viss type skal behandles. "Noen" avgjør altså alle enkeltsaker på forhånd, uten å kjenne dem. "Noen" er gjerne medlemmer i en prosjektgruppe eller lignende, de kan være ansatte eller innleide konsulenter, og slike grupper er sjelden synlige og tilgjengelige for folk utenfor vedkommende forvaltningsorgan. Når de er ferdige med å forhåndsavgjøre enkeltsaker, dvs når systemet er ferdig utviklet, blir ikke dette resultatet sendt på høring eller gjort kjent for allmennheten slik som ved lover og forskrifter. Allmennheten har derfor små muligheter til å påvirke den rettsanvendelse som vedkommende forvaltningsorgan vil følge.

6.2.6 Omfattende konsekvenser ved medhold i klagesaker

Ordinær klageadgang i hver enkelt sak er selvsagt til stede, og klageavgjørelsen kan videre bringes inn for domstolene. En overordnet instans kan for eksempel komme til at en enkeltsak er uriktig avgjort av beslutningssystemet. I så fall inneholder systemet uriktige eller mangelfulle regler, og det kan tenkes at systemet må endres for å unngå at fremtidige saker også blir uriktig avgjort. Dersom vi fastslår at det har foreligget en feil, innebærer det dessuten at en rekke tidligere avgjørelser er uriktige og derfor må/bør omgjøres. En klageavgjørelse som konkluderer med at feil i systemet har gitt uriktig vedtak i en enkelt sak, har derfor ikke bare effekt for den saken det konkret klages på, men kan ha store effekter både forover og bakover i tid. Av hensyn til fremtidig saksbehandling må systemet korrigeres, men dette er ofte kostbart og vanskelig, fordi beslutningssystemene er komplekse, rigide og vanskelige å endre. Av hensyn til fortiden, kan/må forvaltningsorganet omgjøre tidligere enkeltvedtak som på grunn av den aktuelle feilen er uriktige. Også dette kan innebære et ikke ubetydelig administrativt arbeid.

De ofte inngripende konsekvensene av at reglene i beslutningssystemet settes til side, kan muligens bidra til at klageinstansen blir mer tilbakeholdne med å gi klagere medhold. Viktigere er det imidlertid at folk meget sjelden klager over den maskinelle saksbehandlingen. En årsak til dette kan åpenbart være at de aller fleste avgjørelser er korrekte. Selv om

avgjørelser er diskutabile, dvs. selv om det kan hevdes at også en annen løsning er riktig, kan folk derfor velge å godta resultatet. En annen mulig årsak er at de regelverk som styrer avgjørelsene er omfattende og kompliserte, og at den enkelte mangler tilstrekkelig innforståelse om innholdet i den rettslige reguleringen. Riktignok er de aktuelle reglene på detaljert måte uttrykt i datamaskinprogrammer, men disse er skrevet for datamaskiner, ikke for mennesker, og er derfor ikke til hjelp for folk flest.

I tillegg til at rettspraksis har påvirkning på systemene, vil selvsagt også endringer i lover og forskrifter kunne kreve til dels omfattende systemendringer – av og til på kort varsel. Eksempler på dette kan være innføring av kontantstøtten, differensiert MVA og utvidet vrakpant på personbiler over en viss alder.

6.3 Offentlighetsprinsippet og IKT-politikken

Offentlighetsprinsippet er en bærebjelke i vår forvaltning, som blant annet demokrati- og rettssikkerhetsidealene hviler på. Prinsippet kommer for eksempel til uttrykk i forvaltningsloven og offentlighetsloven, og innebærer at parter har krav på innsyn i dokumentene i "sin" sak. Enhver har med noen unntak rett til innsyn i offentlige saksdokumenter, journaler m.v. Et vesentlig poeng med innsynsrettighetene er muligheten for å kunne kontrollere hvordan offentlig myndighet blir utøvet. Her er legalitetskontroll sentralt, dvs. en kontroll av at myndigheter utøver sin kompetanse på en måte som er i samsvar med gjeldende rett.

Hvilken mulighet har vi til å kunne kontrollere den myndighet som utøves ved hjelp av beslutningssystemer? Én mulighet er kontroll av enkeltvedtak som treffes ved hjelp av systemene, ved å anvende de relevante rettsreglene til å undersøke om en oppnår samme resultat som forvaltningen. Dette krever at en for hver sak går grundig inn i det foreliggende rettskildematerialet og foretar en kvalifisert juridisk vurdering av hvert foreliggende vedtak. Slik vil domstoler og klageinstanser arbeide. Metoden er imidlertid arbeidskrevende, og det sier seg selv at kun et lite antall saker kan undergis denne type legalitetskontroll.

6.3.1 Ny offentlighetslov og innsyn i digitale saksopplysninger

Offentlighetslovutvalget fremmet i desember 2003 forslag til en ny offentlighetslov, NOU 2003: 30 *Ny offentlighetslov*. Lovforslaget har vært på høring våren 2004¹⁰⁰. Forslaget tar inn viktige endringer. Teknologirådet finner imidlertid grunn til å støtte mindretallet i Offentlighetslovutvalget i at den nye offentlighetsloven må suppleres med ytterligere to bestemmelser.¹⁰¹

Forslag til endringer i § 9

Et mindretall i offentlighetsutvalget, lederen Harald Hove og Nils E. Øy, fremmet forslag til to nye bestemmelser, § 9 og § 10, som etter Teknologirådets syn peker i retning av et vesentlig fremskritt for åpenheten i informasjonssamfunnet, dvs. et samfunn der en stadig større del av reguleringer og handlingsnormer er nedfelt i IKT-baserte informasjonssystemer og uttrykt som programkode. Rådet har imidlertid funnet det hensiktsmessig å foreslå endringer av ordlyden i mindretallets forslag til §9 i ny offentlighetslov:

¹⁰⁰ <http://odin.dep.no/jd/norsk/publ/hoeringsnotater/012041-080079/ram001-bn.html#ram1>

¹⁰¹ Teknologirådets høringsinnspill til Justisdepartementet:

http://www.teknologiradet.no/files/hring_offentlighetslov_teknologiradet.pdf

Teknologirådets forslag til endret § 9 *Rett til innsyn i opplysninger om regelverk, modeller og andre forutsetninger som dataprogrammer er basert på:*

”Enhver kan kreve å få opplyst hva som er det rettslige grunnlaget (regelverk mv.) for de dataprogrammer forvaltningen benytter til å treffe enkeltvedtak. Enhver kan få opplyst hvilke modeller og forutsetninger ellers som ligger til grunn for analyser og beregninger forvaltningen har utført ved hjelp av et dataprogram.”

Endringsforslaget innebærer for det første en understrekning av at det kan være aktuelt å redegjøre for annet enn regelverk. Rett til innsyn i opplysninger om det rettslige grunnlaget bør ikke begrenses til regelverk, men bør for eksempel også omfatte dommer. For det andre er retten til innsyn avgrenset til ”enkvttvedtak”, jf. forvaltningsloven § 2 bokstav b. Innsynsretten avgrenses altså til de tilfeller der de aktuelle dataprogrammene anvendes for å avgjøre den enkeltes plikter og rettigheter.

I annet ledd er det ikke gjort nevneverdige endringer ut over å sette ”forutsetninger” sammen med ”modeller”. Modeller vil typisk være komplekse, og forutsetninger typisk være enkle.

Forslag til endringer i § 10

Mindretallet i Offentlighetslovutvalget har også foreslått en ny § 10, som gjelder bruken av dataprogrammer forvaltningen benytter. Bestemmelsen innebærer at den enkelte – med visse begrensninger – skal ha rett til å kreve at forvaltningen anvender dataprogrammer for formål som den enkelte selv bestemmer. Dette gjelder med andre ord mulighet til å simulere virkninger av rettsreglene for den enkelte. For eksempel vil den enkelte kunne be om beregning av fremtidig skatt under forutsetning av endret inntekt på grunn av forventet arbeidsløshet.

Forslaget må bygge på den forutsetning at det faktisk finnes et dataprogram som er egnet for formålet; typisk opplæringsmoduler av fagsystemene. Utviklingen av en digital, internetbasert forvaltning innebærer dessuten at flere etater gjør tilgjengelig slike simuleringsrutiner fra sine nettsider. Statens lånekasse for utdanning og Skatteetaten har slike tilbud i dag.

Teknologirådet vil understreke betydningen av at den enkelte gis tilgang til slike rettslig baserte dataprogrammer som er avgjørende for analyse av den enkeltes plikter og rettigheter, jf. forrige avsnitt. På bakgrunn av dette forslår Rådet en ny § 10 som gjelder dette rettslige aspektet:

”Dersom et forvaltningsorgan gjør bruk av dataprogram for å treffe enkeltvedtak, har enhver rett til, i rimelig grad, å kreve at organet benytter dataprogrammet for å vise hva resultatet av mulige, fremtidige enkeltvedtak kan bli. Retten gjelder ikke dersom slike dataprogram er gjort allment og vederlagsfritt tilgjengelige.”

Forslaget til ny § 10 knytter bestemmelsen til enkeltvedtak, dvs. til undersøkelsen av folks rettigheter og plikter. Forslaget vil i stor grad øke forutberegneligheten og begripeligheten av rettslige reguleringer. Bestemmelsen vil trolig også stimulere til økt tilbud av simuleringsrutiner på forvaltningens nettsider, noe som etter Teknologirådets syn vil være en udelte positiv virkning.

6.3.2 Effektivisering av innsynsrettighetene

I tillegg til nye innsynsrettigheter er det grunn til å vurdere i hvilken grad eksisterende innsynsrettigheter kan effektiviseres ved hjelp av nye IKT-baserte verktøy. Særlig bør det vurderes om det kan etableres et standard informasjonssystem som kan gjøre det lettere for brukere å finne ut hvilke innsynsrettigheter de har. Et slikt verktøy bør også kunne gi støtte til brukere som ønsker å formulere en begjæring om innsyn. Slike verktøy kan for eksempel gjøres tilgjengelig fra alle offentlige nettsteder, ved hjelp av et standard "innsynsikon".

6.3.3 Dokumentasjon av rettslige systemavgjørelser

Kildekode er skrevet for maskiner, ikke mennesker. Kontroll med lovligheten av de regler som programmene uttrykker kan derfor ikke skje ved at slike koder leses av mennesker direkte. Når beslutningssystemer skal utvikles, blir de reglene som skal oversettes til programkode først spesifisert av regelkyndige personer. Regelinholdet i beslutningssystemene uttrykkes i en såkalt "kravspesifikasjon". Slike spesifikasjoner er et mulig grunnlag for legalitetskontroller av beslutningssystemene og de enkeltvedtak som disse produserer.

Kravspesifikasjoner er imidlertid ikke umiddelbart tilgjengelige. Det kan nektes offentlig innsyn i slike dokumenter fordi de kan ses som "interne saksdokumenter", jf offentlighetsloven § 5. Den eneste kilden til kunnskap om hvordan forvaltningen utøver sin myndighet, vil således i mange tilfeller ikke være tilgjengelig.

På enkelte områder finnes dermed ingen tilgjengelig dokumentasjon som viser hvilke regler som anvendes når beslutningssystemer treffer enkeltvedtak. Vi kan tenke oss eksempler på situasjoner hvor det kommer en klagesak på en rutine, og den som har skrevet programkoden har sluttet i sin stilling. Det vil da ikke være noen som har detaljert kunnskap om innholdet i rutinen.¹⁰² En slik situasjon samsvarer dårlig med et tradisjonelt rettssikkerhetsideal, og bør ikke aksepteres som en "kostnad" ved automatisert forvaltning. Tvert i mot bør det klare utgangspunktet være at ethvert forvaltningsorgan som automatiserer sine beslutninger, må utarbeide dokumentasjon der det går klart fram hvilke regler beslutningssystemet anvender når enkeltsaker blir avgjort. Denne dokumentasjonen bør inneholde henvisninger til de lovbestemmelser mv. som reglene i datamaskinprogrammene er basert på. Slik dokumentasjon av de rettslige systemavgjørelsene, kan sette tilsynsmyndigheter og enkeltpersoner i stand til å forvandle innholdet i den "den sorte boksen" til et gjennomskiktig system av regler. Dette vil være en forutsetning for å kunne etterprøve hvordan automatisk saksbehandling blir gjennomført.

6.3.4 Oversikt og rettigheter i en digital forvaltning

Som ledd i omstillingsarbeidet i offentlig forvaltning, blir det utviklet nye og nyttige internettbaserte rutiner, som gjør det mulig for folk å delta aktivt i behandlingen av sin egen sak. Toll- og avgiftsdirektoratet, Skatteetaten og Lånekassen er eksempler på etater som har kommet langt i å utvikle slike systemløsninger. Imidlertid ivaretar selv de mest avanserte løsningene brukernes behov på en utilstrekkelig måte. AltInn gir for eksempel utilstrekkelig oversikt over saksbehandlingsrutiner og manglende informasjon om den enkeltes rettigheter.

¹⁰²Eksempelet er beskrevet i Dag Wiese Schartum: *En rettslig undersøkelse av tre edb-systemer i offentlig forvaltning*, Complex 1/89, Tano, s 88-89.

De internetbaserte rutinene som den enkelte bruker gis tilgang til, vil ofte være en begrenset del av den samlede saksbehandlingsrutinen. Øvrige opplysninger i saken kan for eksempel bli hentet inn fra andre kilder enn brukeren, og de opplysningene som brukeren inngir kan være gjenstand for kontroller av forvaltningsorganet. Det kan også være at forvaltningsorganet gjør helhetsvurderinger i tillegg til de beslutningsmekanismer som er nedfelt i systemet. Saksbehandlingsrutinene er dessuten ofte for enkle til at alle spesielle sakstyper kan bli behandlet, noe som innebærer at saken kan bli avvist fra rutinen og må behandles manuelt.

Forvaltningen bør gi full oversikt over de samlede saksbehandlingsrutiner som internetbaserte rutiner er del av, slik at brukerne hele tiden vet hvordan deres sak blir behandlet.

I lovgivningen er den enkelte gitt en rekke rettigheter, og forvaltningen er gitt plikter som skal oppfylles i saksbehandlingen. Slike plikter gjelder selvsagt også når saksbehandlingsrutinen er basert på bruk av internet. Ikke desto mindre er det meget sjelden at lovens krav blir etterlevd i forbindelse med slike rutiner. Dette gjelder særlig i forhold til den informasjonsplikten forvaltningen har som "behandlingsansvarlig" i henhold til personopplysningslovens § 19. Etter denne bestemmelsen skal forvaltningen informere de som registrerer opplysninger om seg selv, herunder om formålet med behandlingen av personopplysninger, hvem opplysningene blir utlevert til og om det er frivillig å gi fra seg opplysningene. Slik informasjon mangler ofte på offentlige nettsteder. Det er også gjennomgående svak informasjon om hvilke rettigheter den enkelte har, for eksempel etter forvaltningsloven og i henhold til særlovgivning. Forvaltningen må ta den enkeltes rettigheter på alvor – også i den digitale forvaltningen. Brukerperspektivet og rettighetsperspektivet må gå hånd i hånd og utfylle hverandre til glede for brukerne.

6.4 Oppsummering – IKT i forvaltningen: lov, formater og konkurranse

Modernisering av forvaltningen ved hjelp av IKT byr på en rekke utfordringer. I de tidligere kapitlene i denne rapporten har vi fokusert spesielt på problemstillinger knyttet til standardisering og åpenhet.

I dette kapitlet har fokus vært på et annet viktig hensyn som må ligge til grunn for en demokratisk programvarepolitikk, nemlig *gjennomsiktighet*. Utfordringene knyttet til dette kan oppsummeres som følger:

- Å sikre fortsatt åpenhet når programmer tar over lovanvendelsen.
- Å revidere offentlighetslov og forvaltningslov og forskrifter med IKT for øye.
- Å sikre at saksbehandlingsrutiner i den digitale forvaltningen blir oversiktlige og tilstrekkelig beskrevet til at brukerne kan forstå sin rolle i den samlede rutinen.
- Å legge betydelig større vekt på å lage digitale saksbehandlingsrutiner som sikrer etterlevelse av lovbestemte krav til utøvelse av forvaltningsmyndighet når det gjelder brukers rettigheter og forvaltningens plikter.



Kapittel 7 Kunnskapsutvikling i Norge og internasjonalt

Undervisningssektoren er en kjernearena som på sikt har innvirkning på flere politikk-områder. Programvarepolitikk i skole og utdanning er således av stor betydning. Skolens programvare bør være et hensiktsmessig og kosteffektivt redskap i læringsprosessen. Den bør også bidra til å gi elevene innsikt i informasjonsteknologi slik at de kan bli aktive deltakere i informasjonssamfunnet, både som brukere og utviklere. I tillegg bør programvare i undervisning være så nøytral som mulig, både når det gjelder den type læringspraksis den legger til rette for, og når det gjelder de leverandører den binder den enkelte utdanningsinstitusjon og elev opp til. Ideelt sett bør programvaren være leverandør-uavhengig, slik skolen generelt skal være fri for produktreklame. Dette kan også ha effekter for konkurransen i programvaremarkedet.

OECD-studien *Learning to change: ICT in schools* (2001)¹⁰³ peker på at IKT over tid vil forandre undervisningssituasjonen for elever verden over, gitt visse forutsetninger:

“As a first step, schools have an urgent need for plentiful software and digital materials of quality. Both teachers and students must become discerning and knowledgeable ICT users. The school environment must be fully equipped and the management must be supportive of ICT, making available expert assistance and advice to the teachers in their rapidly changing and extended professional role. New patterns of curriculum and student assessment will be needed, that are compatible with ICT-enriched learning. Aspects of school organisation such as the structure and length of the working day, must be revised if the full educational potential of ICT is to be delivered.”

Det er viktig å understreke at IKT ikke er et mantra for endring. Egeninnsats vil alltid kreves, både av elever, lærere, foreldre og myndigheter.

Dette kapittelet vil ta for seg spørsmål om valg av programvare i skole og utdanning, store offentlige prosjekter samt bistand. Vi trekker inn erfaringer fra tidligere skolesatsinger og ser spesielt på prosjektet Skolelinux.

7.1 Teknologisk og samfunnsmessig endring – trender i skolen

Skole- og utdanningssektoren står i dag foran en rekke utfordringer, hvorav mange ble berørt i Kvalitetsutvalgets innstilling, NOU 2003: 16 *I første rekke*,¹⁰⁴ og i stortingsmelding nr. 30 (2003-2004) *Kultur for læring*¹⁰⁵ fra Utdannings- og forskningsdepartementet.

¹⁰³http://www.oecd.org/document/0,2340,en_2649_34519_2466626_1_1_1_1,00.html

¹⁰⁴<http://www.odin.dep.no/ufd/norsk/satsingsomraade/kvalitetsutvalget/045071-990317/dok-bn.html>

¹⁰⁵<http://www.odin.dep.no/ufd/norsk/satsingsomraade/kvalitetsutvalget/bn.html>

Globalisering av samfunnet, kunnskaper, teknologier og elever gir skolen mulighet til å åpne seg mot verden, ta imot innspill samt dele erfaringer med andre som er i en læresituasjon. Innvandring og en flerkulturell hverdag, kombinert med mål om toleranse og ytringsfrihet, gjør at skolen må møte realitetene på en annen måte enn før.

Avdemokratiseringen og forvitring av folkestyret gir, slik Maktutredningen¹⁰⁶ peker på, behov for at det politiske menneske kan uttrykke seg i nye former. Skole- og utdanningssektoren må speile disse uttrykksformene, for eksempel ved å legge til rette for internett og den varierte aktivitet som nettet kan formidle.

Teknologisering, det vil si at større og større deler av vår hverdag sammenveves med teknologiske løsninger, idéer og samværsformer, gjør at vi kan si at samfunnet krever at den enkelte behersker teknologi. Alt det nye gir valgfrihet for de som kan navigere i dette nye landskapet, mens det hemmer de som ikke kan det.

Kunnskap om samspillet mellom teknologi og samfunn er sentralt for fremtidig verdiskaping så vel som utdanning til "skikkede samfunnsborgere". Skole- og utdanningssektoren tar derfor i bruk teknologi stadig sterkere grad. Et spørsmål som det er viktig å stille, er på hvilken måte teknologien velges, introduseres og brukes i skole- og utdanningssektoren.

7.2 IKT i skolen

Hos utdanningsmyndighetene har IKT vært gjenstand for en viss satsing helt siden 1980-tallet. Optimismen var stor i den første stortingsmeldingen om IT i norsk utdanning, *Datateknologi i skolen* (1984).¹⁰⁷ Ti år senere kom stortingsmelding nr. 24 (1993) *Om IT i utdanningen*, som ble fulgt opp med den første fireårs handlingsplanen om IT i skole og utdanning, for perioden 1996-1999. Siden år 2000 har IKT-satsingen økt fra ca. 38 til ca. 200 millioner kroner per år, noe blant annet handlingsplanen for 2000-2003 nøt godt av.

Mye har altså skjedd før Utdannings- og forskningsdepartementet våren 2004 la frem sitt *Program for digital kompetanse* (2004-2008).¹⁰⁸ Samfunnet har endret seg fra 1993 til 2004, noe det nye programmet gjenspeiler. Vi snakker om en digital hverdag, der teknologien er vedt inn i alle sider av samfunnslivet og teknologiens plass i samfunnsutvikling og verdiskaping er økende:

"Utvikling av digital kompetanse er viktig for at barn, unge og voksne skal kunne delta i et internasjonalt og mobilt utdanningssamfunn/arbeidsliv [...] og forhindre digitale skiller" (s.11).

Program for digital kompetanse skal bidra til å bygge bro mellom ferdigheter som å lese, skrive og regne og den kompetansen som kreves for å ta i bruk nye digitale verktøy på en kreativ og kritisk måte. Programmet vil nytenke pedagogikken med fokus på kunnskapsdannelse, læring og formidling og styrke infrastruktur, kompetanseutvikling, digitale læressurser samt forskning og utvikling. Programmet nevner følgende hovedutfordringer:

- Å avklare spørsmål med hensyn til åpen kildekode og åpne standarder (s.17).

¹⁰⁶<http://www.sv.uio.no/mutr/>

¹⁰⁷St.meld. nr. 39 (1983-1984) *Datateknologi i skolen*.

¹⁰⁸*Program for digital kompetanse* (2004) <http://odin.dep.no/ufd/norsk/satsingsomraade/ikt/index-b-n-a.html>

- Å utvikle læreres og skoleleders digitale kompetanse (s.20).
- Å styrke skoleeiers bestiller- og innkjøperkompetanse (s.20).
- Å fortsette arbeidet med standardisering av digitale læringsressurser (s.25).

Innen 2006 skal systemer for produksjon, lagring og gjenfinning av digitale læringsressurser så langt som mulig være basert på internasjonale standarder (s.33). Utdannings- og forskningsdepartementet har i den forbindelse etablert det såkalte eStandard-prosjektet for å diskutere problemstillinger knyttet til utviklingsmiljø, læring og programvare.¹⁰⁹

7.2.1 Digital kompetanse

Grunnleggende ferdigheter i bruk av digitale verktøy er en forutsetning for å fungere i dagens samfunn. Utdanningsminister Kristin Clemet la våren 2004 frem en stortingsmelding om kvalitet i grunnopplæringen.¹¹⁰ Å kunne bruke digitale verktøy fremholdes her som én av fem grunnleggende ferdigheter som er forutsetninger for læring og utvikling både i skole, arbeid og samfunnsliv. I dette ligger blant annet å hente frem, lagre, skape, presentere og utveksle informasjon.

Kvalitetsmeldingen fremholder lærerne som skolens viktigste ressurs. Lærer-IKT¹¹¹ er del av den nasjonale satsingen på økt datakunnskap i norske skoler. Den nettbaserte etterutdanningen for lærere er utviklet på oppdrag fra Utdannings- og forskningsdepartementet av en rekke norske universiteter og høyskoler. Om lag 18 000 lærere har fått opplæring gjennom Lærer-IKT siden 2002.

7.2.2 En inkluderende skole

Bruk av IKT skal også bidra til en mer inkluderende skole, blant annet gjennom å bryte ned barrierer for funksjonshemmede.¹¹² Individuelt tilpasset og likeverdig opplæring er et overordnet prinsipp i all virksomhet innenfor skoler og andre pedagogiske institusjoner. Elever skal oppleve en inkluderende og tilpasset skole der enhver får sjanse til å lykkes i forhold til sine forutsetninger, anlegg og interesser. Elever med nedsatt funksjonsevne skal oppleve god planlegging, hjelp og veiledning i overgangen mellom grunnskolen og videregående opplæring og i overgangen til høyere utdanning/arbeid.

Likeledes jobber det europeiske nettverket European Agency for Special Needs in Education for tilrettelegging av undervisning for funksjonshemmede på europeisk nivå.¹¹³ I USA er offentlig innkjøpspolitikk for å bidra til bedre løsninger for mennesker med funksjonshemninger lovregulert gjennom *American Disability Act*, ADA, 1990 og *American Rehabilitation Act*, Section 508. Amerikanske IT-selskaper som Microsoft, Oracle og IBM arbeider med å tilrettelegge sine produkter for funksjonshemmede, og dette er en forutsetning for innkjøp til føderale myndigheter.

¹⁰⁹http://www.estandard.no/om_prosjektet/

¹¹⁰St.meld. nr.30 (2003–2004):

<http://odin.dep.no/ufd/norsk/satsingsomraade/kvalitetsutvalget/bn.html>

¹¹¹<http://www.larerikt.no>

¹¹²St.meld. nr.40 (2002–2003): <http://odin.dep.no/sos/norsk/publ/stmeld/044001-040005/dok-bn.html> og NOU 2001: 22: <http://odin.dep.no/sos/norsk/publ/utredninger/NOU/030001-020008/dok-bn.html>

¹¹³http://www.european-agency.org/publications/agency_publications/ereports/erep4.html

Utviklere av programvare basert på åpen kildekode har også fokus på tilrettelegging for funksjonshemmede. De to største skrivebordsapplikasjonene GNOME og KDE har egne grupper som arbeider med dette, og det er mulig for alle å komme med innspill og nye idéer til arbeidet. GNOME holder blant annet årlige samlinger for brukere og utviklere verden over for å stake ut kursen for den videre utviklingen.¹¹⁴

7.2.3 Programvare i grunnskolen

I skolen brukes programvare for e-post, nettleser, tekstbehandling, regneark m.m.. I grunnopplæringen dominerer i dag Microsoft-baserte løsninger, etter eksperimentering med Norsk Data, Winix (1988-1992)¹¹⁵ og Tiki.

Det finnes også andre alternativer. For eksempel brukes tekstbehandleren Lotus Word Pro og nettleseren Netscape i Kongsberg kommune, og en rekke skoler bruker kontorstøttepakken OpenOffice.org. OpenOffice.org¹¹⁶ på norsk er et samarbeidsprosjekt mellom Skolelinux, Akershus fylkeskommune, Møre og Romsdal fylkeskommune og Sør-Trøndelag fylkeskommune.¹¹⁷ Akershus Fylkeskommune tilbyr kurs i bruk av dette programmet, som kan brukes både på operativsystemene Windows, Linux og Mac OS X, i likhet med et bredt spektrum av annen programvare.

Andre eksempler på programvare som er basert på åpen kildekode, og fungerer på flere plattformer er webserveren Apache, nettleseren Mozilla, databasen MySQL og kontorstøttepakken OpenOffice.org. På tilsvarende måte finnes proprietær programvare som også kan brukes på flere plattformer, deriblant databasene til Oracle, nettleseren Opera, video-redigeringsprogrammet Maya og kontorstøttepakken StarOffice.

Forskjellig programvare i hjemmet og på skolen byr på problemer når en hjemme-PC skal benyttes til å gjøre leksene. Svenske myndigheter har sammenlignet ulike programvare for tekstbehandling¹¹⁸ og funnet ut at det beste formatet for interoperabilitet er Microsoft Word. Dette formatet er imidlertid ikke åpent, og brukere kan derfor ikke gå ut fra at det ved en oppgradering til nyere versjoner av programvaren fortsatt vil egne seg til utveksling mellom ulike tekstbehandlere.

7.3 Tidligere erfaringer med IKT i skolen

Aftenposten formulerte 5. mars 2004 en skarp kritikk av offentlig IKT-politikk.¹¹⁹

"På 1990-tallet var trygdeetatens Tress 90 og skoleetatens Winix skrekkeeksempler på at det offentlige måtte tåle milliardtap på mislykkede IT-satsinger. En fersk undersøkelse fra forskningsinstitusjonen Simula viser at situasjonen ikke er blitt bedre på denne siden av

¹¹⁴<http://guadec.org/> GUADEC er den årlige bruker- og utviklerkonferansen arrangert av GNOME. Konferansen, som ble arrangert i ved Høgskolen i Agder i 2004, trakk over 400 internasjonale deltakere både fra offentlig forvaltning og utviklermiljøer. Program og linker finnes på <http://2004.guadec.org/>. Utover teknisk utvikling i GNOME-prosjektet, belyste konferansen spesielt initiativer for mer tilgjengelige IT-løsninger for mennesker med funksjonshemninger og bruk av IT i fattige land. Konferansen resulterte blant annet i en samarbeidsavtale mellom norske Skolelinux, PSL-Brasil og gnu/LinEx i Spania.

¹¹⁵Den såkalte "Winix-skandalen" (1991-1992) <http://www.uninett.no/uninytt/1991-2/win.html>

¹¹⁶<http://www.openoffice.org.no/nyhetsarkiv.shtml>

¹¹⁷<http://www.openoffice.org.no/nyhetsarkiv.shtml>

¹¹⁸<http://www.statskontoret.se/pdf/2003116.pdf>

¹¹⁹Aftenpostens leder 5. mars 2004

(http://www.aftenposten.no/meninger/leder_morgen/article745805.ece).

tusenårsskiftet. Hvert år sprekker det offentliges budsjetter for dataprojekter med seks milliarder kroner. Det er tre ganger mer enn de samlede budsjettoverskridelsene ved norske sykehus i fjor. Nesten halvparten av de 60 milliarder kroner som totalt blir brukt på IT-utvikling her i landet hvert år, er budsjettoverskridelser eller rene tap. Hverken næringslivet eller staten ville godta slike røde tall på andre sektorer. Det synes åpenbart at for dårlig planlegging og for lavt kunnskapsnivå hos dem som bestiller IT-tjenester, er viktige årsaker til den manglende kostnadsstyringen. Dataselskapene, resten av næringslivet og det offentlige bør ha felles interesse av å samarbeide langt bedre enn tilfellet er i dag, for å fjerne den altfor ukritiske kulturen som ofte hersker omkring slike satsinger. Data- og kommunikasjonsteknologi er blitt en del av vår hverdag. Da bør det kreves like stor kontroll med IT-kostnader som med andre typer investeringer.”

Simula-rapporten Aftenposten refererer til så imidlertid ikke noe om at eksisterende budsjetter er overskredet, men presenterte en simulering som mer enn noe annet viste hvor vanskelig det er å beregne IKT-utgifter. Likevel sier det noe om hvor utfordrende IKT-satsing i det offentlige kan være. Det handler om å forstå både IKT og skolen samtidig, ikke bare IKT eller skolen.

Bruk av IKT i det norske skoleverket har en veldig spesiell historie, og deler av historien er omstridt. Stortingsmelding nr. 39 (1983-84) om *Datateknologi i skolen* initierte handlingsprogrammet for datateknologi skolen 1984-88 og førte til opprettelsen av det såkalte Datasekretariatet i Kirke-, utdannings- og forskningsdepartementet (KUF). Resultatet av dette ble som kjent en satsing på det mange i ettertid mener var ”utdatert og sær maskinvare”, spesifisert av KUF. Erfaringene fra denne satsingen på Winix og den senere satsingen på TIKI-maskiner er verdifulle erfaringer ved utformingen av en fremtidsrettet programvarepolitikk for den norske skole- og utdanningssektoren.

”Winix-skandalen”

Den såkalte Winix-skandalen dreier seg om intet mindre enn et norskutviklet IT-system som skulle kjøre programvare basert på det som på åttitallet var et forskerbasert internett tilgjengelig på PC i norske skoler. Det gjaldt altså å gjøre operativsystemet Unix tilgjengelig på PC, som da betød Windows-plattformen. Prosjektet, som opprinnelig så optimistisk het Pegasus, etter den flyvende hesten i gresk mytologi, startet med støtte fra det daværende Kirke- og Undervisningsdepartementet (KU) i 1988.

Winix hadde i utviklingsfasen kontakt med både Microsoft i Seattle, USA og IBMs utviklingsavdeling i Atlanta, USA. International Data Corporation (IDC) som er et ledende firma innen markedsanalyse, utførte en spesialanalyse for Winix i 1991 og konkluderte med at: ”De voldsomme vekstpotensialene som fremkommer for Winix kommer fra en meget heldig kombinasjon. Man har de rette softwareløsningene tilpasset de rette hardwareløsningene for 90-årenes databehandling.” Videre heter det: ”Og man er kommet langt nok i produktutviklingen til å være med på veksten. Produksiden er med andre ord i orden, det vil bli markedsføringen som er avgjørende for om Winix kan ta del i denne veksten.”¹²⁰

¹²⁰Intern rapport frembrakt av Arild Staupe for Teknologirådet 1. april 2004.

Teknisk sett var Winix, ifølge flere IT-eksperter, meget nyskapende. Dette kommer også til uttrykk i en nyere rapport fra Nærings- og handelsdepartementet (2003).¹²¹ Det var imidlertid

ikke så mange i departementet som "forstod data" så tidlig som på 1980-tallet. De fleste i styringsgruppen var byråkrater uten særlig erfaring med IT-systemer.

I 1992 ble det klart at Winix i skoleverket ble en fiasko. Programvaren ble ikke ferdig i tide, og en amerikansk samarbeidspartner, Winix America, brøt distribusjonsavtalen. Det ble også avslørt at KUF-ansatte hadde inngått muntlige millionavtaler og overskredet budsjetter. Statsråd Gudmund Hernes stoppet alle utbetalinger. En rekke små og store selskaper som hadde avtaler med KUF satt igjen med svarteper. Det kom flere søksmål i ettertid. Riksrevisjonen mener det kostet den norske stat 88 millioner kroner¹²² mens Simula-rapporten anslår beløpet til 100 millioner.¹²³

Problemet med Winix var blant annet at det allerede fantes mange tilsvarende produkter, at utviklingen tok lang tid og at informasjonen til potensielle brukere var mangelfull. Det kan imidlertid ikke på basis av dette ikke trekkes noen generell konklusjon om at det offentlige ikke bør engasjere seg i IKT-utvikling i skoleverket.

TIKI 100 – historien om en norsk datamaskin i skoleverket

TIKI-100 var en norskprodusert hjemmedatamaskin som ble innkjøpt av mange norske skoler på 1980-tallet. Initiativet sammenfalt med opprettelsen av Datapolitisk råd i 1984 og tanken om "den store IT-planen" i flere norske departementer – en plan som senere strandet fordi aktørene ikke klarte å bli enige om retningen.¹²⁴

På 1980-tallet var nasjonale satsninger på "egne" maskiner i tiden: BBC i England, Scandis i Sverige og altså Tiki i Norge. Tiki ble produsert fra 1984 av (Kon)-Tiki Data,¹²⁵ men ble raskt utkonkurrert da PC-markedet tok av for fullt. Tiki var imidlertid i bruk i skoleverket i 10-15 år, som vel er enestående for et proprietært design utenom IBMs serier.

Det var i begynnelsen ikke klart hva data skulle brukes til, men i Kirke- og undervisningsdepartementet ble det etter hvert en forståelse av at det ikke var programmeringsspråket Basic skolebarna skulle lære, men snarere å bruke datautstyr som hverdagsverktøy. Mange som gikk i grunnskolen på 1980-tallet vil imidlertid huske at datamaskinen mest ble brukt til slangespillet Python, selv om Tiki også hadde tekstbehandleren BRUM, regnearket Tiki-Kalk og databasen TIKI-BAS, så vel som tekstbehandleren WordStar og regnearket Super Calc.

¹²¹Winix baserte seg på OSI-standarder (Open Systems Interconnection), så vel som den såkalte TCP/IP-protokollen. Det ble laget en tilpasset norsk OSI-profil, NOSIP. Positiv omtale av det tekniske grunnlaget for Winix finnes i eNorge Tilstandsrapport juni 2003, s. 18.

<http://www.enorge.org/FileArchive/81/eNorge%20Tilstandsrapport%20-%20juni%202003.pdf>

¹²²<http://www.oslo.net/historie/CW/utg/9427/cw942741.html>

¹²³Simula-rapporten (http://www.simula.no/news_one.php?news_id=67). Offentlig IT-sprekk for 6 mrd. hvert år (<http://www.aftenposten.no/nyheter/okonomi/article744978.ece>)

¹²⁴Trond Buland, 1996, *Den store planen: Norges satsing på informasjonsteknologi 1987-1990*. STS-rapport nr. 27 1996. NTNU.

¹²⁵Tiki Data ble gründet av Lars Monrad-Krohn (f. 1933), norsk dataingeniør og gründer. Han var den første diplomstudent på data ved NTH, Institutt for Radioteknikk i 1959) Monrad-Krohn startet Norsk Data i 1967 (CEO 1967-1972), Mycron i 1957 (CEO 1975-1982) og Tiki-Data i 1984 (CEO 1984-1996). Arbeider siden 1999 hovedsakelig med oppfølging av norske IT-gründere i forskningsparken i Oslo.

Da striden om å vinne Kirke- og undervisningsdepartementets innkjøpsavtale for datamaskiner i skolen var hard på 1980-tallet, hadde Tiki én konkurrent – Scandis. Både Tiki og Scandis ble valgt. Tiki havnet hovedsakelig i grunnskolen, Scandis i videregående.

Med 4/6 Mhz prosessor og 64KB RAM var Tiki bare hakket "bedre" enn Commodore64. Tiki kjørte på et eget operativsystem kalt Tiko som var kompatibelt med IBMs operativsystem CP/M 2.2. Det viktigste markedet for Tiki-Data var offentlig sektor og spesielt skolemarkedet.

Modellen Tikulf kom på markedet i 1985, men ble en flopp. Markedets krav om 100 prosent PC-kompatibilitet var mer eller mindre absolutt. Da var det liten vits i å utvikle egen maskinvare, så Tiki begynte som alle andre å importere ymse generasjoner av PC-kompatible hovedkort fra Taiwan som ble puttet inn og videresolgt under varemerket "Tiki". Mot slutten av 1994 fikk Tiki-Data en stor ordre fra Skolesjefen i Oslo. Ordren utgjorde nesten en fjerdedel av omsetningen. Merkantildata overtok alle rettigheter til både varemerket og teknologien i 1998.

Nå var Scandis heller ingen stor suksess. Det var minst tre ting som tok livet av Scandis: Maskinen var proprietær, dvs. den var verken kompatibel med de 8 bits CP/M-maskinene som da var i ferd med å bli historie eller den 16-bits IBM-PC-standarden som var i ferd med å etablere seg. Det fantes praktisk talt ikke programvare til maskinen, og det var vanskelig å få maskinvaren stabil, særlig diskettstasjonen var problematisk.

Kirke- og undervisningsdepartementets kravspesifikasjon nevnte at skoledatamaskinen som ble valgt skulle ha slike ting som CP/M, Z80 samt at maskinen skulle ha "grafikk" og "lyd". Tiki-100 ble utviklet med departementets kravspesifikasjon som mal og mye tyder på at maskinen tilfredsstilte den på alle punkter.

7.3.1 Lærdommer av offentlige "IT-skandaler"

Lærdommen det offentlige kan trekke av erfaringene med Winix og TIKI er:

- Å jobbe med større offentlige IKT-satsinger krever både byråkratisk, forretningsmessig og teknologisk kompetanse. Det er dessuten viktig å ha god kontakt med brukerne av teknologien – allerede fra starten.
- Teknologiskiftene i IKT-bransjen er ofte drastiske og uventede. Det offentlige bør derfor ikke ha ambisjoner om å være først ute utenom der dette eventuelt kan gi store politiske gevinster. Det offentlige bør jobbe med industristandardiserte løsninger av en viss modenhet, og ikke eksperimentere for mye. Å satse på norsk forskning og utvikling (FoU) innebærer imidlertid en viss risiko som man må løpe med tanke på innovasjon. Det kan gå galt. Man bygger likevel opp kompetanse, til tross for tap. Flere elementer fra Winix-satsingen har overlevd, for eksempel teknologien til det norske selskapet Powersim¹²⁶. Dette er kompetanse som har kommet godt med i videre norsk IKT-satsing.
- En årsak til Norsk Datas fall var sen overgang til åpne standarder. Dagens leverandør-avhengighet kan vise seg å bli like problematisk. Erfaringene med Winix og Tiki demonstrerer at det kan være uheldig for det offentlige og skolesektoren å binde seg

¹²⁶Powersim (<http://www.powersim.no/>)

for sterkt opp til en enkelt leverandør eller teknologi. Situasjonen for åpne standarder og Linux er en helt annen i 2004 enn på 1980-tallet som var preget av IBMs dominans.

- Det er på høy tid at entusiasmen for IKT i skolen gjenopptas, samtidig som man holder et blikk på langsiktige effekter og kostnader, så vel som forretningsmessig styring. Alle IKT-kontrakter bør ut på anbud, og IKT-kompetansen blant offentlige innkjøpere må bedres i alle ledd.

7.4 Er Skolelinux et reelt alternativ?

Skolelinux¹²⁷ er et prosjekt med norsk opprinnelse basert på åpen programvare med utgangspunkt i Linux-distribusjonen Debian¹²⁸ som allerede har eksistert i 11 år. Skolelinux-pakken inneholder alle grunnleggende funksjoner for infrastrukturen i en skole integrert med kontorstøttepakken OpenOffice.org. Mer enn 100 skoler i Norge har tatt systemet i bruk. Installasjonsprogrammet til Skolelinux er oversatt til 20 språk.

I Statskonsult-rapporten *Erferinger fra bruk av Skolelinux* (2003:24)¹²⁹ framgår det at skolene som har valgt Skolelinux er fornøyde og ville gjort det samme om igjen. Konseptet synes å være billigere, selv om rapporten ikke angir langtidskostnader og risikoer. Imidlertid er det noen læringsplattformer som foreløpig ikke finnes for Skolelinux, og dette er en utfordring for skolene. Statskonsults konklusjon er like fullt at Skolelinux er velegnet til bruk i skolen.

7.4.1 Mulige hindre for utbredelse av Skolelinux

Selv om Skolelinux så langt har hatt en rask vekst er det ikke gitt at veksten vil fortsette. Fortsatt vekst krever økt tilgang til brukerstøtte ("support") på denne plattformen. Selv om det nå finnes selskap som tilbyr driftstjenester, er det mulig at det er skapt for store forventninger blant kommuner og skoler.

Det ser ikke ut til at tekniske hindre vil bremse veksten til Skolelinux. Derimot er det mulig at rettslige rammebetingelser som f.eks. lovverket for å beskytte intellektuelle rettigheter endres slik at det kan bli vanskelig å utvikle og bruke programvare basert på åpen kildekode. Hvis hver utvikler og bruker må betale et patentgebyr som kanskje er på høyde med en programvarelisens, er det fare for at utviklingen stoppes.

Hvis EU innfører tilsvarende beskyttelse av forretningsmetoder som USA, kan det bli gebyr på bruk av programvare, noe som vil kunne bremse eller stoppe Skolelinux. En annen hindring kan være myndigheter som krever detaljert testing og integrasjon. På grunn av slike krav får for eksempel ikke skoler i Oslo midler til å bytte til Skolelinux.¹³⁰

Hittil har man brukt forholdsvis lite statlige ressurser på Skolelinux, og det er avsatt lite midler til support og oppfølging av skolene som velger denne løsningen. Skolelinux har følgelig ikke har nådd et særlig stort volum, utover skoler med entusiastiske og dyktige IKT-lærere. Skolelinux er imidlertid, på tross av små ressurser, i dag installert på mer enn 100 norske skoler.

¹²⁷<http://www.skolelinux.org/no>

¹²⁸<http://www.debian.org/>. Debian er en Linux-distribusjon med bred støtte både blant utviklere og bedrifter. Over 1000 utviklere deltar i utviklingen av Debian, og selskaper som Hewlett Packard leverer produkter tilpasset Debian.

¹²⁹<http://www.statskonsult.no/publik/rapporter/2003/2003-24.pdf>

¹³⁰<http://www.aftenposten.no/nyheter/nett/article759459.ece>

Den årlige kostnaden for utvikling og vedlikehold av Skolelinux er estimert til ca 2,2 millioner kroner. Skolelinux er så langt finansiert av Norwegian Unix Users Group Foundation (NUUGF) med 4,5 millioner kroner over to år i form av et rentefritt lån. I tillegg har Utdannings- og forskningsdepartementet gitt 250.000 kroner i støtte til en forprosjekt-rapport. Utover dette har prosjektet mottatt støtte i form av driftstjenester og lån av utstyr og lokaler fra bedrifter som er engasjert i åpen kildekode. Det er med andre ord foreløpig en beskjeden offentlig satsing. Dersom denne situasjonen vedvarer, vil det på sikt måtte avklares hvorvidt prosjektet kan videreføres i Norge. Skolelinux er i dag en offisiell del av Debian, og dette gjør at prosjektet er såpass internasjonalt anerkjent at det i prinsippet kan overtas av utviklere i et annet land dersom Norge ikke ser seg tjent med å bygge videre på denne kompetansen.

Statskonsult melder om at det er helt avgjørende for valg av Skolelinux at skolen har dyktige IT-ansvarlige. Dessverre er IT-ressurser i skolen knappe, og lærere med spesiell kompetanse kan flytte eller bytte arbeidssted. Dersom den dyktige IT-ansvarlige forsvinner, kan skolen pådra seg betydelige kostnader ved å måtte migrere tilbake til Microsoft. Flere skoler har valgt å beholde noen maskiner med programvare fra Microsoft for å kunne bruke programvare som ikke er tilgjengelig på andre plattformer. Foreløpig er det ingen av skolene som har begynt å bruke Skolelinux som har valgt å migrere tilbake.

Teleplans rapport om åpen programvare¹³¹ peker på manglende kompetanse som største hindring til økt bruk. Hvor finner man kompetanse? Hvem har ansvaret for å utdanne relevant arbeidskraft? Dette er i hovedsak en utfordring for den norske universitets- og høgskolesektoren.

7.4.2 Pedagogisk programvare

Pedagogisk programvare brukes til å formidle faglig innhold i fag som matematikk, språk og geografi. En rekke bedrifter både i Norge og andre land tilbyr pedagogisk programvare. Dessuten finnes en rekke nettsteder¹³² som tilbyr gratis pedagogisk programvare. Noen av disse er basert på åpen kildekode.

Læringssenteret¹³³ har igangsatt tiltak og bevilget midler til utvikling og utprøving av nye elektroniske læremidler for grunnskole og videregående skole. Skolenettet.no¹³⁴ er en internettportal fra Læringssenteret og ett av tiltakene under Utdannings- og forskningsdepartementets handlingsplan IKT i norsk utdanning. Skolenettet tilbyr et bredt spekter av kvalitetssikrede lenker, nyheter, fagstoff, læreplaner og ikke minst tre innloggingstjenester: Skolepost, som er en gratis, reklamefri, nettbasert e-posttjeneste; Skolesekk, hvor elever og lærere kan lagre sine egne dokumenter; og Skolestue der de kan samarbeide med andre over nettet.

Selskaper som lager pedagogisk programvare for Windows-plattformen kan også gjøre det for Skolelinux-plattformen. Det er ingen forutsetning at programvare som brukes sammen med Skolelinux må være basert på åpen kildekode. Derimot er det ofte en forutsetning for å kunne utnytte programvaren effektivt i en undervisningssituasjon at man som lærer og elev

¹³¹Teleplan (2003): *Åpen programvare i Norge – status, effekter, hindringer og drivere*.

¹³²<http://www.schoolforge.net/software.php>, <http://www.ofset.org/freeduc-cd/>
<http://opensource.schools.org/>

¹³³Læringssenteret ble fra 15. juni 2004 innlemmet i det nye Utdanningsdirektoratet, direktoratet for utvikling av grunnskole og videregående opplæring, <http://www.utdanningsdirektoratet.no/>

¹³⁴<http://skolenettet.is.no/imaker?id=3&omr=1671&mal=forside>

bruker programvare fra én leverandør. For eksempel tilbyr Skolenett.no foreløpig kun pedagogisk programvare for Microsoft Windows-plattformen.¹³⁵

Det er ofte behov for å tilpasse pedagogisk programvare til ulike lokale behov slik som oversettelse til nynorsk eller samisk. En annen form for behov kan være knyttet til bruk av leseleser eller skjermleser. Microsoft Office foreligger nå også i en nynorsk utgave. Den kom like etter at OpenOffice.org hadde oversatt sin programvare for tekstbehandling.

IKT kan også brukes som utredningsredskap for vurdering av ferdigheter i forhold til å betjene utstyr for personer med bevegelsesproblemer og hjerneskader. Programvare kan brukes til å identifisere eventuelle funksjonshemninger på et tidlig tidspunkt slik at en elev kan få bedre tilpasset undervisning om nødvendig. En slik programvare som tilbys gratis fra Skolenettet.no er Ariadne¹³⁶. Denne programvaren er "spesifisert til å kjøres i MS Internet Explorer."

For elever og studenter med funksjonshemninger kan IKT gi økte muligheter for aktiv deltakelse både i undervisningen og i informasjonssamfunnet. Dette forutsetter imidlertid at løsningene blir laget og etterspurt i henhold til retningslinjer for tilgjengelighet.

7.4.3 Gjenbruk av eldre datamaskiner ved hjelp av Skolelinux

Maskinvaren som brukes i skolen er som regel av eldre dato. Handlingsplanen *IKT i norsk utdanning (2000-2003)* omtaler en ordning for brukte datamaskiner for skoleverket. Dette er en ordning hvor Microsoft samarbeider med Alternativ Data for å tilby maskinvare og programvare til skolene. Maskiner med ferdig installert programvare fra andre leverandører enn Microsoft har vært vanskelig å oppdrive i Norge. Dette har vært en alvorlig hindring for utvikling av et mangfold i markedet. Læringssenteret har nylig tegnet en avtale med In/out data¹³⁷ som kan levere brukte maskiner med Skolelinux ferdig installert.

7.4.4 Utfordringene for Linux i skolen

Hva er konsekvensene ved økt bruk av Linux i norsk skole? Dette avhenger av om satsingen er godt forberedt, om det finnes økonomiske og bemanningsmessige ressurser til å gjennomføre dette teknologiskiftet. Uansett vil det handle om:

- mer ressurser til IKT-kompetanse hos lærere
- betalt brukerstøtte lokalt
- at man over tid ikke utelukkende baserer seg på ildsjeler og enkeltpersoner

Erfaring med feilslåtte IKT-satsinger har gjort både politikere og embetsverk mindre eksperimentelle. Det er derfor viktig å støtte Skolelinux i dagens situasjon. Gevinsten vil kunne vise seg på lang sikt ved at elever, lærere og studenter får innsikt i grunnlaget for teknologien. Et viktig moment som burde tale for en slik satsing, er at skolene ikke bør være så sterkt knyttet til enkeltleverandører som de er i dag. Dette gjelder selvsagt også i forhold til Skolelinux-prosjektet.

¹³⁵<http://skolenettet.no/imaker?id=166908&malgruppe=0&trinn=0&omr=100341&mal=sporsmal>

¹³⁶http://www.isaac.no/RTV_HRA/ariadne/index.html

¹³⁷<http://www.inout.no/>

7.5 Programvare i forskning og høyere utdanning

På norske universiteter og høyskoler er det en viktig utfordring å identifisere bidrag til norsk verdiskaping. Grunnlaget for verdiskaping er at kunnskap er tilgjengelig. Programvare med åpen kildekode sprer kunnskap på tilsvarende måte som klassisk vitenskap. Proprietær kildekode gjør at kunnskap ikke kan spres på samme måte.

En rekke universiteter finansierer deler av sin forskning gjennom tett samarbeid med store programvareleverandører. Dette gjelder bedrifter som leverer både proprietær programvare og programvare basert på åpen kildekode. Microsoft er blant selskapene som samarbeider med flere norske universiteter.¹³⁸

En type programvare som brukes i utdanningen er såkalte Learning Management Systems (LMS). Alternativer basert på åpen programvare er f.eks. Moodle¹³⁹, Segue¹⁴⁰ og SAGU¹⁴¹. Det amerikanske arbeidsdepartementet har utviklet en variant kalt Workforce Connections¹⁴² som viser hvordan en myndighet kan ta ansvar for å utvikle en programvare basert på åpen kildekode.

I Norge er det Classfronter fra det norske selskapet Fronter¹⁴³ som dominerer blant universitet og høyskoler. Når slik programvare blir brukt etter sin hensikt, vil en stor del av studentenes interaksjon med lærestoffet og med medstudenter og lærere foregå gjennom systemet. Det kan øke fleksibiliteten for eksempel gjennom tilrettelegging for fjernstudium. Classfronter er imidlertid ikke godt tilrettelagt for blinde studenter eller studenter med synshemninger. Det er betenkelig at et så sentralt produkt kan få en dominerende stilling i Norge uten å oppfylle grunnleggende krav for tilgjengelighet. Dette taler for at innkjøpsrutinene bør skjerpes.

IKT-studenter bør få innsikt i prinsipper og grunnleggende funksjonalitet bak ulike produkter heller enn opplæring i bestemte produkter. De bør få kunnskap om mangfoldet og grunnlag til å ta bevisste valg for å styrke norsk IKT-næring.

På norske universiteter undervises og forskes det på alle plattformer og systemer, og det er ikke tatt noen overordnede valg om hva det skal fokuseres mest på. Likevel er de fleste av back-end systemene Unix/Linux-drevet og kontorprogramvaren mest Microsoft-basert. Flere aktører etterlyser nå en mer strategisk satsing. Ved NTNU har IME-fakultetet, i

¹³⁸ Academic Days er en informasjonsutvekslingsarena der akademikere fra Norden og Benelux møter forskere fra Microsoft Research og representanter fra Microsofts utviklingsgrupper. Siste academic days ble holdt i Oslo og hadde ca. 110 deltagere. Agenda og innhold finnes på <http://research.microsoft.com/collaboration/university/europe/events/AcademicDays/NE/0404%20Oslo/> Microsoft Developer Network Academic Alliance: www.msdn.net – Microsofts program for software tilgang til universiteter og forskningsinstitusjoner. På området finnes også et kursområde (Curriculum repository) som er en slags åpen markeds plass for kurs som kan benyttes i undervisningen. I Norge er totalt 25 institusjoner fordelt på informatikkstudiene ved alle universitetene, samt flere høyskoler og andre utdanningsinstitusjoner medlemmer av programmet. Gratis teknologistøtte til universiteter finnes på <http://www.uscemea.org/>. I tillegg finnes Shared Source-initiativet på .NET, samt informasjon om kriterier og prosess for tilgang til kildekode på andre produkter som f.eks. Windows. I Norge har UiO og UiT slik lisens: <http://www.microsoft.com/resources/sharedsource/default.aspx>

¹³⁹ <http://moodle.org/>

¹⁴⁰ <http://segue.middlebury.edu/index.php?action=site&site=segue>

¹⁴¹ <http://sagu.codigolivres.org.br/>

¹⁴² <http://www.workforceconnections.dol.gov/>

¹⁴³ <http://www.fronter.no/no/>

samarbeid med Teknologirådet, i 2004 satt i gang en fremsynsstudie for å fokusere sin forsknings- og undervisningsstrategi i et 20-års perspektiv¹⁴⁴. Programvaresatsing vil være en sentral del av denne strategien.

Det er alltid stor debatt om behovet for kompetanse på IKT – hvor mange programmerere trenger vi? Hvor mange må forstå ”litt mer” enn alle oss andre? Sikkert er det at universitetene er på banen og vil ha mer penger, og at industrien sier de vil ha flere kandidater å velge mellom.¹⁴⁵

7.6 Bistandspolitikk, programvare og åpenhet

Programvare som utvikles for å effektivisere offentlig forvaltning, vil i stor grad ha samme funksjoner i ulike land. Lokale tilpasninger vil ta hensyn til språk, kultur og lovverk. Åpen kildekode eller fri kildekode i fattige land er viktig fordi det gir landet en mulighet til selv å utvikle programvaren, istedenfor å belaste handelsbalansen med import.¹⁴⁶

NORAD har retningslinjer¹⁴⁷ for bruk av IKT i bistandsarbeidet hvor verken åpne standarder eller åpen programvare omtales. Derimot har NORAD og Utenriksdepartementet støttet arbeidet til den norske organisasjonen FAIR¹⁴⁸, som arbeider med ”åpen programvare” i utviklingsland.

Den svenske *Styrelsen för Internationellt Utvecklingssamarbete* (SIDA), på sin side, anbefaler en strategi for verdiskapning basert på åpen programvare, ifølge en fersk rapport om åpen programvare i utviklingsland.¹⁴⁹ Rapporten tar spesielt utgangspunkt i Sri Lanka. Nylig har IT-Universitetet i Kista¹⁵⁰ sammen med SIDA etablert et kompetansesenter for IT i utviklingsland

Også det finske utenriksdepartementet har gitt ut en rapport om IKT i et bistandsperspektiv, *Free as in Education - Significance of the Free/Libre and Open Source Software for Developing Countries*.¹⁵¹ Den finske rapporten peker på lokale muligheter hva gjelder språk og etablering av arbeidsplasser. Verdensbanken har utgitt rapporten *Open Source Software: Perspectives for Development*¹⁵² som viser til en rekke eksempler hvor åpen programvare med gode resultater brukes til oppbygging av lokal kunnskap og som utgangspunkt til en uavhengig bærekraftig utvikling.

¹⁴⁴<http://www.ime.ntnu.no/infosam2020/>

¹⁴⁵ Allerede en STEP-rapport fra 1999 problematiserte dette med IT-kompetanse, og påpekte dessuten at staten må være flinkere med å komme på banen i standardisering:
<http://www.statskonsult.no/it/itskatt/aekland.doc>

¹⁴⁶http://linuxtoday.com/news_story.php3?ltsn=2004-01-05-013-26-OS-SW-PB

¹⁴⁷<http://www.norad.no/items/1739/108/5436118580/Information%20and%20Communication%20Technology.pdf>

¹⁴⁸<http://www.fair.no/>

¹⁴⁹<http://www.sida.se/Sida/jsp/polopoly.jsp?d=1265&a=23955>

¹⁵⁰<http://www.it-universitetet.se/artikel/754/003002010/se>

¹⁵¹<http://www.maaailma.kaapeli.fi/FLOSSReport1.0.html>

¹⁵²<http://www.infodev.org/symp2003/publications/OpenSourceSoftware.pdf>

7.6.1 Lokal tilpasning av programvare

Enkelte språk har begrenset eller ingen støtte i proprietær programvare. Det gjelder for eksempel tajik i Tajikistan¹⁵³ og inntil ganske nylig også nynorsk. Leverandører av proprietær programvare har valgt å prioritere andre språk.

Programvare basert på åpen kildekode gir mulighet for lokale tilpasninger som eksempelvis oversettelse. Programvare som er tilpasset det lokale språket er en avgjørende forutsetning for effektiv nedbygging av det digitale skillet. På grunn av programvare basert på åpen kildekode kan snart også de ca. 6.8 millioner innbyggerne i Tajikistan få programvare på sitt eget språk.

Skrivebordsapplikasjonen GNOME er oversatt til 28 språk.¹⁵⁴ Til sammen 87 ulike grupper arbeider med oversettelse til hvert sitt språk. Skrivebordsapplikasjonen KDE blir også oversatt til 87 forskjellige språk.¹⁵⁵ Nettleseren Mozilla har 104¹⁵⁶ registrerte prosjekter for oversetting. Dette gir en bedre mulighet til å bevare språk og kultur enn om prioritering skjer kun på grunnlag av estimert inntjening. Tilgang til programvare og informasjon på eget språk er viktig for å sikre demokratiet i informasjonssamfunnet.

I mars 2004 ble den første stavekontroll for Kiswahili, som er et utbredt språk i Afrika, utviklet på AfricaSource.¹⁵⁷ Nyheten ble lagt ut på *slashdot*.¹⁵⁸ I april annonserte Microsoft¹⁵⁹ at Office 2003 og Windows XP vil utgis med støtte for Kiswahili. Noe lignende skjedde i Norge da kontorpakken OpenOffice.org ble utgitt på nynorsk. Dette er enda et eksempel på at åpen kildekode skaper konkurranse og kan stimulere til utvikling av bedre produkter også fra utviklere av proprietær programvare.

Microsoft har etablert et "Local Language Program"¹⁶⁰ som inviterer frivillige til å bidra til oversettelse på en måte som ligner den som brukes for åpen programvare. Før initiering av programmet hadde Microsoft oversatt skrivebordsapplikasjonene til omlag 40 språk.¹⁶¹

7.6.2 Samarbeid på tvers av landegrenser

IT muliggjør samarbeid mellom ulike skoler på tvers av landegrenser. Her har blant annet FN-sambandet gjennom INTIS-prosjektet¹⁶² og Kidlink¹⁶³ gått foran med gode eksempler. GRID (Global Resource Information Database) i Arendal er pådriver for FN-tiltaket Global Virtual University (GVU).¹⁶⁴ Tiltaket er i ferd med å utvikle et masterstudium i ledelse av miljø og naturresurser. GVU er basert på produkter fra Microsoft.

¹⁵³<http://www.infodev.org/symp2003/publications/OpenSourceSoftware.pdf>

¹⁵⁴<http://developer.gnome.org/projects/gtp/teams.html>

¹⁵⁵<http://i18n.kde.org/teams/>

¹⁵⁶http://www.mozilla.org/projects/l10n/mlp_status.html

¹⁵⁷<http://www.tacticaltech.org/node/view/88>

¹⁵⁸<http://developers.slashdot.org/article.pl?sid=04/03/30/0130224&mode=thread&tid=117&tid=126&tid=156&tid=99>

¹⁵⁹<http://www.hananews.org/WholeArticle.asp?artId=1578>

¹⁶⁰<http://www.microsoft.com/Resources/Government/LocalLanguage.aspx>

¹⁶¹<http://www.microsoft.com/presspass/press/2004/mar04/03-16LLPPR.aspx>

¹⁶²<http://fn-sambandet.no/article/articleview/805/1/320/>

¹⁶³<http://www.kidlink.org/>

¹⁶⁴<http://www.gvu.unu.edu/>

Et annet FN-tiltak er kalt GESCI (Global e-schools and Communities Initiative). Det er et samarbeid mellom UN ICT Task Force, GESCI og McKinsey&Company.¹⁶⁵ Tiltaket ble lansert på FN-toppmøtet WSIS¹⁶⁶ i desember 2003 og tar sikte på å koble sammen tusenvis av skoler og byer i utviklingsland. GESCI tar ikke stilling til programvare.

Eksempelet Namibia

Schoolnet Namibia¹⁶⁷ har på grunn av høye kostnader ved bruk av Microsoft programvare valgt å skifte til programvare basert på åpen kildekode på alle skoler i Namibia. De fleste skolene i det afrikanske landet har allerede installert en Linux-løsning som kalles DireqOpenLab¹⁶⁸ som har flere felles trekk med Skolelinux, men som er bedre tilpasset behov i Afrika. Kvadraturen videregående skole¹⁶⁹ i Kristiansand har et utvekslingsprogram med Okahandja Secondary School i Namibia hvor elevene blant annet blir kjent med hverandre ved bruk av IKT.

Norge bør med IKT- og bistandspolitikken bidra til å sikre muligheten for å kommunisere med andre kulturer og land, uansett hvilke tekniske forutsetninger disse har. Dette krever helhetstenkning og valg av teknologi som følger åpne og fritt tilgjengelige standarder. En rekke land deriblant Kina, Japan og Sør-Korea undertegnet nylig en avtale¹⁷⁰ om å sette sammen sin egen Linux-distribusjon for å gjøre seg mindre avhengig av Microsoft. Ikke minst med tanke på veksten i den kinesiske økonomien vil det også være kommersielt interessant for norske bedrifter å lage løsninger som kan brukes i denne sammenheng. Både GNOME¹⁷¹ og norske Trolltech¹⁷² har allerede etablert et samarbeid.

Teknologirådets ekspertgruppe foreslår:

- Det bør utvikles en strategi for IKT i utviklingsarbeid som vurderer potensial for lokale arbeidsplasser og oppbygging av kompetanse.
- Potensialet for gjenbruk av programvare både nasjonalt og i utviklingsland bør vurderes.
- Resultater fra vellykkede norske prosjekter basert på åpen programvare bør videreformidles, f.eks. gjennom et samarbeid mellom FAIR og Skolelinux.
- Det bør gjennomføres sammenlignende studier med proprietær og åpen programvare i bistandsarbeidet.
- Prosjekter som kan etablere samarbeid for utvikling av felles læremateriell og programvare, f.eks. et åpent supplement til Global Virtual University(GVU), bør stimuleres.
- Norge kan profilere seg som initiativtaker til POSS (Pooling Open Source Software) for utviklingsland.

¹⁶⁵<http://www.gesci.org/>

¹⁶⁶<http://www.un.org/News/Press/docs/2003/pi1548.doc.htm>

¹⁶⁷<http://www.schoolnet.na/>

¹⁶⁸<http://www.direqlearn.org/1024/openlab.html>

¹⁶⁹<http://www.kvadraturen.vgs.no/informasjon/internasjonalisering/internasjonalisering.html#>

¹⁷⁰<http://itavisen.no/art/1303248.html>

¹⁷¹<http://www.businesswire.com/webbox/bw.012401/210240273.htm>

¹⁷²<http://www.trolltech.com/newsroom/announcements/00000098.html>

7.7 Oppsummering: IKT i skolen – en samfunnsmessig utfordring

Grunnleggende ferdigheter i bruk av digitale verktøy er en forutsetning for å fungere i dagens samfunn. Men IKT er i seg selv ikke tilstrekkelig for å styrke elevenes motivasjon og læring. Det er det samlede spekteret av arbeidsmetoder som læreren gjør bruk av i undervisningen som er avgjørende. Det viktigste er at IKT i skolen gjør lærestoffet tilgjengelig for alle og følger åpne standarder, at det som læres er direkte relevant i samfunnslivet og at det er tid til refleksjon og forståelse av IKT som samfunnsfenomen.

Kunnskapskulturen knyttet til åpen kildekode, som bygger på åpenhet, er en verdi i seg selv. Støtten til Skolelinux-prosjektet bør derfor utvides. Universitet og høyskoler bør også tilstrebe større grad av uavhengighet fra enkeltleverandører.

Utviklingsland med beskjedent utbygd IKT-industri og IKT-infrastruktur er spesielt sårbare for å bli låst til bestemte leverandører. Derfor har aktører som FN og den nasjonale svenske bistandsorganisasjonen SIDA prosjekter med åpen kildekode. Norge bør også engasjere seg i bistands- og samarbeidsprosjekter med åpen kildekode. Basert på Skolelinux og erfaringer fra FAIR og FN-organet GRID i Arendal, kan Norge ta initiativ til å samarbeide med fattige land basert på felles utvikling av åpen kildekode.

Ordforklaringer

Kildekode

Et programs kildekode er en instruks som forteller hva programmet skal gjøre. Kildekoden kan leses og endres av en programmerer.

Linux

Egentlig en "kjerne", dvs. den mest sentrale biten av et operativsystem. Linux ble opprinnelig utviklet av finnen Linus Torvalds. Begrepet brukes oftest som samlebetegnelse på en hel familie av nært beslektede operativsystemer basert på den samme kjernen.

De forskjellige operativsystemene dekker alt fra spesialiserte verktøy som brannvegger, via hjemme-PC-er til servere og tungregneanlegg. Systemene og kjernen har blitt videreutviklet på Open Source-måten av et felleskap av flere hundre programmerere. Linux tok opprinnelig utgangspunkt i en nedskalert versjon av operativsystemet Unix, kalt Minix, men er nå utviklet til et selvstendig operativsystem. Linux har en beskjeden markedsandel for vanlige PC-brukere i næringsliv og i privatmarkedet, men har hatt stort gjennomslag i enkelte nisjer, blant annet som operativsystem for webservere og hyllevarebrannvegger. Over en million webservere kjører nå Linux.

Mange ser Linux som et mulig alternativ til Windows i næringsliv og organisasjoner ettersom tilgangen på forretningsapplikasjoner samt verktøy for konfigurasjon og administrasjon av operativsystemet er blitt bedre enn tidligere. Linux-basert programvare er ofte gratis. Det finnes også kommersielle Linux-distribusjoner, det vil si komplette operativsystempakker som består av kjernen Linux og ulike verktøy og brukerapplikasjoner. Noen eksempler på forskjellige typer Linux-distribusjoner er Red Hat, Debian, SuSe, Skolelinux og Mandrake.

Lisenser

Kjøp, salg og rettigheter knyttet til programvare utover lovverket for intellektuelle rettigheter (som åndsverk og patentlovgivning) reguleres ofte av lisenser mellom bruker og selger/opphavsmann til programvaren. Lisenser kan være kommersielle eller ikke-kommersielle, og gi forskjellige typer rettigheter og beskyttelse til begge parter, for eksempel knyttet til innsyn, videreformidling, kopiering og endring av programmet.

Lisenser inngår i et samspill med nasjonale åndsverks- og patentregler. Ulovlige brudd på lisens- og åndsverksregulering kalles piratkopiering eller åndsverksnasking ("softlifting"). Piratkopiering kan medføre bøter, saksomkostninger og skadet omdømme både for bedrifter og privatpersoner. Programvarebransjen anser piratkopiering som et alvorlig problem.

Lisenskostnader

Lisenskostnad er den prisen en bruker må betale for å få lov til å installere og bruke et program på sin datamaskin. I tillegg til lisenskostnaden, må man gjerne også betale en årlig avgift for brukerstøtte ("support") til programvareleverandøren.

Dersom brukeren trenger hjelp til installasjon, teknisk drifting, opplæring, oppfølging, tilpassing og vedlikehold i den perioden programvaren benyttes, vil også dette ha en kostnad. Slike kostnader er derfor sentrale i diskusjonen om hvilken type programvare som

er billigst og best, for eksempel for offentlige etater – det blir ikke riktig å sammenligne lisenskostnader isolert.

Åpne lisenser

Det aller meste av programvaredistribusjon er basert på lisenser, men ikke alle må kjøpes. En omtalt lisens er GPL (GNU/Linux General Public License). GPL beskytter brukerens rettigheter til å se, endre og spre kildekoden. Faktisk oppmuntrer den til å kopiere, endre og videreformidle. Samtidig skal GPL sikre at endringer som blir gjort offentliggjøres og kommer alle brukerne til gode. GPL ble laget i 1989 og senere revidert i 1991 av Free Software Foundation. Andre lignende lisenser er LGPL, BSD, MIT og MPL.

Operativsystem

Operativsystemet er den grunnleggende programvaren i enhver datamaskin eller nettverk, fordi den fungerer som en kommunikasjonsentral mellom maskinvaren (prosessoren, hard-disker, nettverkskort etc.) og de applikasjoner som går på datamaskinen. Typiske applikasjoner er e-postprogrammer, regneark og databaser.

Det mest vanlige operativsystemet er Windows. Ellers finnes for eksempel MacOS (brukt på Apples datamaskiner), Unix, Symbian (mye brukt på mobile enheter) og Linux.

Programvare

Programvare er programmer som kjører på datamaskiner og tilhørende utstyr. Maskinvare, derimot, betegner den fysiske siden av datautstyr. Man kan betrakte programvare som den variable dimensjonen av en datamaskin, mens maskinvaren er den konstante. Vi skiller gjerne mellom applikasjonsprogramvare og systemprogramvare. Applikasjoner er programmer som for eksempel tekstbehandling, regneark, database og nettleser. Systemprogramvare inkluderer operativsystemer samt programmer som støtter applikasjoner.

Proprietær programvare

"Eiet programvare" - Programvare som er beskyttet teknisk eller juridisk av personene eller selskapet som har utviklet den, eller som ikke følger åpne standarder. Begrepet forstås oftest slik at kildekoden ikke er åpent og allment tilgjengelig.

Proprietær programvare er grunnlaget for en stor industri av programvareselskaper over hele verden. En sentral aktør på dette området er Microsoft, mest kjent for sin Windows-plattform og kontorapplikasjonspakken Office. Andre viktige aktører på området er eksempelvis Oracle, IBM/Lotus, CA, SAP, Sun og Apple. Store deler av programvarebransjen utvikler applikasjoner og løsninger basert på programvare fra en slik leverandør.

Forkortelser

AAD	Arbeids- og administrasjonsdepartementet
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
CERN	European Organization for Nuclear Research, verdens største senter for elementærpartikkelfysikk
COTS	Commercial Off The Shelf
DRM	Digital Rights Management
EBCDIC	Extended Binary Coded Decimal Interchange Code
EFN	Elektronisk Forpost Norge
GESCI	Global e-schools and Communities Initiate
GIF	Graphics Interchange Format
GNU	GNU's Not Unix. Gratis, Unix-lignende operativsystem
GPL	General Public Licence
GSM	Global System for Mobile Communications
GVU	Global Virtual University
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
IKT	Informasjons- og kommunikasjonsteknologi
IME	Informatikk, matematikk og elektronikk
IT	Informasjonsteknologi
JPG	Joint Photographic Experts Group
LMS	Learning Management Systems
MIT	Massachusetts Institute of Technology
MP3	Mpeg 1 Audio Layer 3 – Kompresjonsalgoritme for lyd

MPEG	Moving Picture Experts Group – Kompresjonsalgoritme for lyd
MPL	Mozilla Public Licence
NCSA	The National Center for Supercomputing Applications
NHD	Nærings- og handelsdepartementet
OSS	Open Source Solutions
PDF	Portable Document Format
PKI	Public Key Infrastructure – opplegg for elektronisk legitimasjon og signature
PNG	Portable Network Graphics
POSS	Pooling Open Source Software
SGML	Standard Generalized Markup Language
SIDA	Styrelsen för Internationellt Utvecklingssamarbete
SMTP	Simple Mail Transfer Protocol
SOA	Service-oriented Architecture
SSB	Statistisk sentralbyrå
TCO	Total Cost of Ownership
TCP/IP	Transmission Control Protocol/Internet Protocol
URL	Universal Resource Locator (web-adresse)
WAI	Web Accessibility Initiative
WSIS	World Summit on the Information Society
XML	Extensible Markup Language

Vedlegg 1 – Om Prosjektet

Mandat

Formålet med prosjektet har vært å vurdere situasjonen for programvarepolitikk i Norge. Det skal særlig fokuseres på demokrati og konkurranse, og på hvordan det teknologiske underlaget påvirker disse. Av spesiell interesse er forholdet mellom åpen og proprietær kildekode, og åpne og lukkede standarder/formater.

Prosjektet skal bidra inn i debatten om det offentliges innkjøp og bruk av programvare, så vel som debattere IKT-bransjens samfunnsansvar for god, brukervennlig programvare. Prosjektet skal jobbe med en bredt sammensatt ekspertgruppe og skal involvere både stakeholders (myndigheter, næringsliv, organisasjoner) og brukere (skoleelever, allmennheten) i arbeidet. Det viktigste vil være å sette programvare på kartet som en viktig politisk problemstilling, og peke på områder hvor ikke markedet alene kan få råde.

Prosjektet ble igangsatt høsten 2002 med full oppstart i februar 2003.

Ekspertgruppe

Prosjektet har benyttet en bredt sammensatt ekspertgruppe og har involvert næringsliv, offentlige institusjoner og organisasjoner i arbeidet.

Eksterne: Professor Dag Wiese Schartum, Institutt for rettsinformatikk ved Universitetet i Oslo, Førsteamanuensis Michael Snaprud ved Høgskolen i Agder, Sjøføkonom Lars Sørgard ved Konkurransetilsynet, Teknologidirektør Håkon Wium Lie fra Opera Software, Postdoktor Tanja Storsul ved Universitetet i Oslo og Daglig leder Grethe Helene Viksaas fra Basefarm as.

Fra Teknologirådet: Professor Einar Johan Aas, fakultet for elektronikk og telekommunikasjon ved Norges Teknisk Naturvitenskapelige Universitet og Professor Magne Espedal, Matematisk institutt ved Universitetet i Bergen.

Dr. polit. Trond Arne Undheim, Teknologirådets sekretariat. Fra 1. september 2004 har sivilingeniør Christine Hafskjold overtatt som prosjektleder.

Ekspertgruppens arbeid

Ekspertgruppens arbeid har bestått i å diskutere de politiske utfordringene knyttet til programvare og sammenholde dette med teknologiske og samfunnsmessige utviklingstrekk. En forutsetning for arbeidet har vært at det skulle presenteres som policyinnspill til Stortinget og øvrige myndigheter, så vel som bidra til en allmenn teknologidebatt.

Gruppen hadde sitt første av i alt 8 ekspertgruppemøter onsdag 23. april 2003, i tillegg har de deltatt på konferanse og høring. Alle medlemmene av gruppen har bidratt til rapporten. Flertallet står bak hele rapporten, med unntak av tekst hvor det eksplisitt er angitt at det er mindretallets (Lars Sørgards) meninger som kommer til uttrykk. Mindretallet (Sørgard) tar forbehold om de deler av innholdet som leder frem til flertallets konklusjoner, samt flertallets anbefalinger, jfr. hans dissenser gjengitt i rapporten.

Høring

3. september 2003 arrangerte Teknologirådet høring om demokrati, standardisering og konkurranse i programvaremarkedet. Høringen var åpen for alle interesserte. Det ble holdt 17 høringsinnlegg fra et bredt spekter av aktører i offentlig og privat sektor, universiteter

samt bransje- og interesseorganisasjoner. Innleggene er tilgjengelig på nettadressen: <http://www.teknologiradet.no/html/505.htm> En kort oppsummering av innleggene er tilgjengelig på nettadressen: <http://www.teknologiradet.no/html/508.htm>

Konferanse

Konferansen *Informasjonssamfunnet for alle*, i samarbeid med Nærings- og handelsdepartementet. Ca. 150 deltakere fra akademia, det offentlige og organisasjonslivet samt skoleelever fra Østlandet, Vestlandet, Sørlandet og Trøndelag deltok. Teknologirådets Eivind Osnes og Statssekretær Oluf Ulseth åpnet konferansen. Øvrige innledere var fra Telenor, DNV Software og Utenriksdepartementet. Skoleelevene var samlet hos Teknologirådet før konferansen for å forberede spørsmål til innlederne. Innleggene fra konferansen er tilgjengelig på nettadressen: <http://www.teknologiradet.no/html/509.htm>

Foredrag for gruppen

Underveis i arbeidet har flere aktører blitt invitert til å holde foredrag for gruppen. Dette gjelder: Konsulent Kjetil Haugland i Teleplan som skrev en rapport om Åpen Kildekode for Nærings- og handelsdepartementet, og Ole Tom Seierstad, Microsoft Norge.

Møter med representanter for IKT-bransjen

I forbindelse med arbeidet har gruppen eller prosjektleder møtt flere IKT-ledere. I et eget arrangement i forbindelse med Jon Halls besøk til Norge ble Linux-miljøet kontaktet. Vi har også hatt samtaler med Microsoft på flere tidspunkt, og fått omfattende skriftlig svar på spørsmål vi har stilt dem.

Database over programvarepolitikk

En omfattende oversikt over andre lands programvarepolitikk (på engelsk), *Software policy Country watch*, tilgjengelig på nettadressen: <http://www.teknologiradet.no/html/429.htm> Forespørsler om programvarepolitikk ble sendt ut på nettet og lagt åpent ut på et nettsted som en ekstra informasjonskilde. Dette genererte et imponerende antall svar. Vi har brukt andre lands erfaringer aktivt i våre forslag til tiltak i Norge.

Formidling

- 1. februar 2004: Fra Rådet til Tinget: *Programvarepolitikk – åpne standarder fremmer demokrati og konkurranse*, tilgjengelig på nettadressen: <http://www.teknologiradet.no/html/556.htm>
- November 2004: *Programvarepolitikk for fremtiden: teknologiske strategier for et åpnere samfunn* – sluttrapport fra prosjektet.
- Prosjektleder Trond Arne Undheim holdt følgende foredrag:

Er programvare politikk?, FrPs Stortingsgruppe, Stortinget, 24. februar 2004: <http://www.teknologiradet.no/html/551.htm>

Er programvare et politisk spørsmål?, Statens dataforums høstkonferanse, Losby gods, 13-14. november 2003: <http://www.teknologiradet.no/html/551.htm>

Er programvare politikk? Elektronisk Forpost Norge, Universitetet i Oslo, 27. april 2004

- Sekretariatsleder Tore Tennøe har holdt følgende foredrag:

Åpne standarder fremmer demokrati og konkurranse, IKT 2004 – Fra visjon til handling – omstilling med IKT, Tønsberg, 3. juni 2004

Programvarepolitikk – anbefalinger fra Teknologirådets ekspertgruppe, Statssekretærutvalget, AAD, 8. juni 2004

Vedlegg 2 – Standardiseringsorganisasjoner

ANSI	American National Standards Institute (ANSI) er USAs medlem i ISO. ANSI har en lang tradisjon for å utvikle standarder for programspråk, f. eks. det eldre ANSI Fortran, og det nyere ANSI C.
IEEE	Institute of Electrical and Electronics Engineers (IEEE) er en medlemsorganisasjon som arrangerer konferanser, utgir publikasjoner og utarbeider standarder. Organisasjonen har noen aktiviteter på området programvare, spesielt vedrørende testing og verifikasjon, men fokuserer ellers mest på maskinvare og transmisjon.
IETF	Internet Engineering Task Force (IETF) har utarbeidet en rekke sentrale standarder for internett, blant dem TCP/IP, HTTP, og SMTP. IETF fokuserer på nettverksprotokoller for internett. Ved å gjøre utviklingsprosessen transparent og standarder fritt tilgjengelige, revolusjonerte IETF måten standarder utvikles på.
ISO	International Organization for Standardization er en sammenslutning av nasjonale standardiseringsorganisasjoner, og Norge er representert ved Standard Norge. ISO utvikler standarder på mange områder, blant de mest kjente er ISO 216 som definerer papirstørrelser (A4 etc.) og ISO 9000 som beskriver prosedyrer for bedrifter. ISO tar ofte over standarder andre organisasjoner har utviklet. Eksempelvis ble EDIFACT utviklet innenfor FN-systemet for så å bli standardisert av ISO.
ITU-T	ITU-T er en avdeling innenfor International Telecommunication Union (ITU) som jobber med standardisering av telekommunikasjon. ITU-T het tidligere CCITT og har tradisjonelt arbeidet med standardisering av telefonsystemer.
OASIS	Organization for the Advancement of Structured Information Standards (OASIS) er et industrikonsortium som særlig arbeider med standardisering av nye XML-vokabular. Et interessant eksempel er Open Office XML som standardiserer et lagringsformat for kontorstøttesystemer.
Standard Norge	I Norge er det Standard Norge som har det offisielle ansvaret for å delta i ISO. Standard Norge har til nå engasjert seg lite på området programvare.
W3C	World Wide Web Consortium ledes av Tim Berners-Lee fra hovedkontoret på MIT. W3C har også avdelinger i Frankrike og Japan. W3C arbeider primært med standarder for weben, f.eks. HTML, CSS og XML.

Vedlegg 3 – Ulike typer standarder

Basisstandarder

Denne gruppen standarder inkluderer fundamentale standarder. Støtte for en av disse standardene vil, i seg selv, ikke garantere at meningsfull informasjon kan lagres eller utveksles. Men støtte for basisstandarder kan være nødvendig for å kunne oppnå meningsfull lagring og utveksling.

Navn	Åpen standard?
ASCII	Ja, ANSI-standard
Unicode	Ja, ISO-standard
SGML	Ja, ISO-standard
XML	Ja, W3C-standard
XSD	Ja, W3C-standard
URL	Ja, W3C-standard

Nettverksstandarder

Standardene under beskriver lavnivå kommunikasjonsprotokoller for internett.

Navn	Åpen standard?
TCP/IP	Ja, IETF-standard
http	Ja, IETF-standard
FTP	Ja, IETF-standard
SSL	Ja, IETF-standard

E-poststandarder

Standardene under brukes for å lagre og utveksle e-post.

Navn	Åpen standard?
POP	Ja, IETF-standard
IMAP	Ja, IETF-standard
SMTP	Ja, IETF-standard
RFC-822	Ja, IETF-standard

Bildestandarder

Standardene under brukes for å lagre og utveksle bilder.

Navn	Åpen standard?
GIF	Nei
TIFF	Nei
JPG	ISO-standard, men med patentkrav
PNG	Ja, W3C-standard

Semantiske standarder

Semantiske standarder beskriver informasjonens mening og roller på et abstrakt nivå. Dette gjør det lettere for datamaskiner å behandle informasjonen, men også tilsvarende vanskelig for mennesker å tilrettelegge informasjonen. Flere av spesifikasjonene under er ennå ikke i vanlig bruk, men er viktige for forskning på området.

Navn	Åpen standard?
RDF	Ja, W3C-standard
Topic maps	Ja, ISO-standard
WSDL	Ja, under utarbeidelse av W3C
UDDI	Ja, OASIS-standard
ebXML	Ja, OASIS-standard
MathML	Ja, W3C-standard
MP3	ISO-standard, men patentert

Dokumentstandarder

Standardene under brukes for å lagre og utveksle elektroniske dokumenter.

Navn	Åpen standard?
HTML	Ja, W3C-standard
Doc	Nei
WordML	Nei
OpenOffice Spec.	Ja, under utarbeidelse av OASIS
PDF	Ja, standardiseres i ISO som PDF/X og PDF/A
Postscript	Nei