



17.11.2016
Høgskolen i Østfold

Akademisk artikkel

I denne artikkelen skal vi ta for oss, hvilke rettigheter en programmerer har i forhold til opphavsrett og eiendomsrett til kode som er produsert.



Håvard Furø Pettersen

Hvilke rettigheter har en programmerer i forhold til opphavsrett og eiendomsrett til kode som er produsert?

Liban Jama, Håvard Pettersen, Azad Shazawari

Sammendrag

Den som skaper et åndsverk har opphavsrett til verket. Domstolen avgjør om noe kan anses som åndsverk. Den som skaper et åndsverk har opphavsrett til verket i henhold til åndsverkloven. I utgangspunktet er den som skaper eieren av verket, men hva skjer hvis et dataprogram er utviklet i vertikalt samarbeid og hva er det? Kan en person slutte hos arbeidsgiveren sin, ta med seg ideer og kunnskap for å starte sin egen bedrift? I utgangspunktet vil dette være tillatt, men det har også en annen side. I følge markedsføringslov skal konkurransen være redelig og i samsvar med god forretningsskikk mellom næringsdrivende. Dersom en konkurrent tar med seg kundelister, priser, markedsanalyser, strategier og så videre fra nåværende arbeidsgiver blir det ikke sett på som redelig konkurranse. Vi har valgt følgende problemstilling. Hvilke rettigheter har en programmerer i forhold til opphavsrett og eiendomsrett til kode som er produsert?

Innledning

Vi bruker Wikipedia (Wikipedia,2016) for å gå en historisk oversikt. Den første loven om opphavsrett kom til Storbritannia i 1710. Det britiske parlamentet vedtok Statute of Anne som ble utformet av Edvard Wortley. Denne loven ga en forfatter enerrett på 14 år til å trykke bøkene sine og kunne fornyes en gang i løpet av livene sine. På 1700 tallet ble det arbeidet med å forlenge vernetiden. I 1774 ble forslaget om eneretten forkastet av det britiske parlamentet(Wikipedia,2016A). USA fikk sin første federal lov om opphavsrett i 1790 og omfatter lik lover som i Storbritannia (Wikipedia,2016A). Vernetiden ble utvidet totalt 11 ganger og i 1998 med (the 105th United States Congress, 1998) ble vernetiden til opphavsman sin levetid tillegg til 70år (Wikipedia,2016A). Under 1800 tallet vokste opphavsretten frem i Frankrike (Wikipedia,2016A). Mens den amerikanske opphavsretten omhandlet økonomiske retter fokuserte det franske systemet seg på forfatteren sine rettigheter. Det franske opphavsretts systemet var besto av økonomiske rettigheter samt moralske og ideelle rettigheter. I 1761ble det fastslått at opphavsrett gikk i arv. I gamle dager ble bare bøker gitt opphavsrett (Wikipedia,2016A). I dag blir opphavsrett gitt til, skrifter av alle slag, muntlige foredrag, sceneverk, musikkverk, filmverk, fotografiske verk, malerier, tegninger, grafikk eller bildekunst, skulptur, bygningskunst, billedvev, kunsthåndverk, kunstindustrim kart, datamaskinprogrammer og oversettelser. Vi bruker Åndsverkloven (Åndsverkloven,2016)for å gi en historisk oversikt. Med så mange elementer som det ble nevnt ovenfor kan det være vanskelig å få klarhet i opphavsrett den enkelte programmerer har til det som er utviklet. Vi har derfor valgt følgende problemstilling. Hvilke rettigheter har en programmerer i forhold til opphavsrett og eiendomsrett til kode som er produsert?

Historien om koding

Koding går helt tilbake til 1800tallet og er en måte fortelle datamaskiner eller andre avanserte maskiner hva de skal gjøre gjennom å lage programmer som skal kjøres (Universitet i Stavanger, 2014). I dagens høyteknologiske samfunn er dette svært essensielt. Dagens nettsider, maskiner, enkelte biler og fly er avhengig av program koder. Dette fordi de utfører

en form for jobb som kodes og programmeres før det ferdigstilles og er satt i verk. Hvis vi går tilbake i tid til 1800 tallet så ser vi den første kodingen som ble satt i verk, og dette var en vev maskin. Ved å produsere kort kunne man produsere forskjellige mønstre og denne kunne programmeres som medfører at du programmerte den ved hjelp av koder(Computer History Museum, 2016). Verdens første programmerer var en kvinne ved navn Ada Lovelace, hun var en engelsk grevinne og datapioner. Hun var mest kjent for å ha skrevet en beskrivelse av Charles Babbages analytiske maskin. Dette var en maskin som skulle utføre alle typer beregninger, ikke bare polynomfunksjoner. Konstruksjonen av denne maskinen bygde på prinsipper fra vev maskinen vi nevnte tidligere. (Wikipedia,2016B).

Koding og programmering er to begreper som brukes hyppig om hverandre, ifølge Den amerikanske Bureau of Labor Statistics anerkjenner de ingen forskjell mellom begrepene programmering og koder for personer som arbeider i programmering(Bureau of Labor Statistics, 2015) . Dette fordi når du programmerer så bruker du det språket datamaskinen vil forstå. Datamaskiner bruker et språk som heter maskin kode og dette er et binært språk. Dette vil si at hver instruks datamaskinen forstår består av ulike kombinasjoner av tallene 1 og 0. Programmere bruker en rekke språk som Java, Python, C#, Ruby, PHP og C++ er bare noen språk innenfor programmering. Noen av disse språkene for eksempel Java, C++ blant annet må kjøres gjennom et program som kalles for en Compiler. Koding er i helheten så essensielt at det er i et med skaperen av koden. Det vil si at kun den som skaper koden har opphavsrett til den. Andre programmere eller investorer som er med på å legge inn innsats ved det sluttferdige produktet kvalifiserer seg ikke til opphavsrettslig vern.

Investorer har ofte en misforståelse for å tro at de får opphavsrett for det arbeidet de betaler for, dette ved at de har latt utviklingen skje på deres maskiner eller som følge av andre bidrag som ikke er skapende art(Frilans, 2009). Opphavsretten kan overdras, man kan for eksempel inngå avtaler som gjør at opphavsretten overdras til de som betaler for arbeidet. Men dette kan i utgangspunktet aldri blandes sammen med spørsmålet om hvem som har den opprinnelige opphavsretten. I loven er det bestemte regler som må oppfylles for at koderen skal anses som opphavsmann, i utgangspunktet er «Som opphavsmann ansees, når ikke annet godtgjøres, den hvis navn eller alment kjente dekknavn eller merke på sedvanlig måte er påført eksemplar av verket eller blir oppgitt når det gjøres tilgjengelig for allmenheten (Åndsverkloven 1961A). I helheten er den som anses som opphavsmann i alminneligheten, den som vil bli behandlet som opphavsmann i rettslig sammenheng.

Det finnes en annen type opphavsrett og det kalles medopphavsrett. Dette vil si at i utgangspunktet har den andre personen vært en essensiell faktor for at sluttproduktet har blitt fullført, men dette vil ikke automatisk føre til medopphavsrett dette må igjen avgjøres konkret. At en har del i opphavsretten til verket vil si at to eller flere har hatt et horisontalt samarbeid. Hvis man har et samarbeidsverk som følger (Åndsverkloven, 1961B) kreves det samtykke fra samtlige opphavsmenn som har vært med på å fullføre sluttproduktet. Det vil si at hvis en mener produktet ikke holder mål, vil det ikke være mulig for de øvrige å ta en flertallsbeslutning slik at det for eksempel kan offentliggjøres eller lanseres. Hvis man på forhånd har hatt en skriftlig eller muntlig avtale hvor alle har blitt enige, kan det utgangspunktet ikke stanses lansering eller offentliggjøring. Det må da være klart at de ikke har stilltende har samtykket på forhånd.

Det finnes klare linjer på hvorvidt den som har opphavsrett kan kreve det eller ikke, for eksempel hvis den enkelte har oversatt en kode fra et annet kodespråk vil du kunne ha opphavsrett på den koden han eller hun har oversatt, men ikke på den originale koden. Det vil da si at du har opphavsrett kun i den skikkelse eller formen du har oversatt kodingen på. For eksempel hvis en person har kodet et spill på Ios form, og en annen gruppe koder det samme

spillet på Android vil de kun ha opphavsrett i kodespråket til Android. Det vil ikke si at de kan råde over det på en måte som gjør inngrep i opphavsretten til originalverket.

Opphavsrett til dataprogrammer utviklet i vertikalt samarbeid

Dataprogrammer i noen størrelser blir utviklet i en arbeidsprosess. Deltagerne bidrar på ulike stadier gjennom arbeidsprosessen. Et eksempel på det kan være at en har et program ide, den andre konsulterer, drøfter, omhandler eller spesialisere ideen for å gjøre ideen gjennomførlig. En tredje spesifiserer hvordan et dataprogram utformes. En fjerde utformer eller koder det av systembeskrivelsen som ble oppgitt. I utgangspunktet vil den som skaper verket få opphavsretten men i et vertikalt samarbeid blir det vanskeligere å avgjøre hvem som kvalifiseres seg til opphavsretten. Hvis alle som har deltatt er ansatt av samme arbeidsgiver går rettighetene til arbeidsgiver i medhold av åvl 39g (Åndsverkloven, 1961E).

Det kan ha vært konsulenter fra andre programmeringsselskaper som har bidratt og da får vi ingen hjelp av åvl 39g. Vi tar i utgangspunktet det eksempel som ble gitt ovenfor, personen som hadde ideen får ikke opphavsrett. Det er fordi ideer ikke er et åndsverk, ingenting ble skapt. For å være støttet av åndsverket må ting bli skapt. De som har deltatt sent i arbeidsprosessen på å skape verket får ikke opphavsrett. Hvis du mener selv at arbeidsinnsatsen din krever opphavsrett tar du feil.

Opphavsretten ligger hos den som spesifiserer hvordan programmet skal utformes for å løse oppgaven, og eller hos den som lager det ferdige programmet i henhold til en slik spesifisering. Hvis du har vært med i et vertikalt arbeidsprosess og ikke får opphavsrett kan du skrive en rapport, rapporten i seg selv får du opphavsrett på og dette kan være en kvalifikasjon for et annet prosjekt.

Kildekoden

Vi har i flere århundre hatt enkelte tvister hvor opphavsretten er problemet, det kan være mellom alt fra arbeidsgiver til samarbeidspartnere. Det var litteraturmetaforen som var den mest nærliggende, dette fordi koding i seg selv kunne mine om en tekst(Torvund, 2010). Kildekodens språk er stivt og lite formulerende. I dagens samfunn er dataprogrammer beskyttet som litterære verk, men det kan heller ikke blandes fordi når det er snakk om litteratur i utgangspunkt for bedømmelse av opphavsrett fanges man lett av den litterære metafor og dette er feiledende. Når en programmerer koder vil han eller hun ikke kode for å ha som mål for at det skal bli lest av mennesker, men heller gi en styringsinstruks til maskinen.

Konklusjon

Det er vanskelig å tenke seg at en programmerer kan utforme kodingen og strukturen, og deretter overlate den endelige utformingen til noen andre. I sånne tilfeller vil da den som utformet kodingen og strukturen få enerettslig opphavsrett. Dette gjør det så vanskelig å blandet med litterære verk som folk flest gjør. Det avgjørende for hvem som har opphavsrett vil være den som bidratt med det skapende arbeid. Dette arbeidet må likevel tilfredstille kravet om verkshøyde. Hvis i den grad programmeren ikke klarer eller kan foreta selvstenidige valg, vil det i noen tilfeller utelukke opphavsrett, Men dette vil ikke si at det

foretas valg er avgjørende for opphavsrett. Det er variasjonsbredden som er en nødvendig men ikke ikke tilstrekkelig betegnelse for å oppnå opphavsrett.

I norsk lov som vi har spesifisert tidligere er det slik at programmeren beholder opphavsretten selvom han er arbeidstaker hos et firma. Dersom dette er tilfellet kan hun eller han inngå avtale med arbeidsgiver. Vi har også tilfeller når et program blir lisensiert som en åpen kildkode, dette betyr at opphavsmannen har gitt fra seg opphavsretten, slik som det er beskrevet i lov om opphavsrett til åndsverk (Åndsverkloven,1961C). Det vil da være slik at brukerne av programmet får de samme rettighetene som den opprinnelige opphavsmannen etter paragraf 2(Åndsverkloven,1961D). Vi kan likevel ikke se bort ifra at den opprinnelige opphavsmann har rett til å bli navngitt på alle fremtidige utdrag eller kopier av programmet. Konklusjon er at den som utarbeider kravspesifikasjonen ikke har opphavsrett til det ferdige programmet. Opphavsretten ligger hos den som spesifiserer hvordan programmet skal utformes for å løse oppgavene, og hos den som lager det ferdige programmet i henhold til en slik spesifisering.

Referanseliste

Bromley, Allan G. "The Evolution of Babbage's Calculating Engines." *Annals of the History of Computing* 9.2 (1987): 113-36.

Computer History Museum. Ada Lovelace biografi. URL: <http://www.computerhistory.org/>. Hentet 07.09.2016.

Kielland, Torger (2007). Rettslige problemstillinger ved åpen programvare i norsk rett. I: *Lov og rett* vol 46, 7, 2007, s406-427

Torvund, Olav, Opphavsrett til dataprogrammer utviklet i vertikalt samarbeid, URL: http://www.torvund.net/index.php?page=opph_vertikal. Hentet 13.09.2016

Tormund, Olav, Hvem har opphavsrett. URL: <http://www.torvund.net/index.php?page=hvem-har-opphavsrett>.

Wikipedia, (2016A), Opphavsrett, Hentet 17.11.2016, fra URL: <https://no.wikipedia.org/wiki/Opphavsrett>.

Wikipedia, (2016B), Ada Lovlace, Hentet 17.11.2016, fra URL: https://nn.wikipedia.org/wiki/Ada_Lovelace.

Åndsverkloven, (1961A), Lov 7, URL: <https://lovdata.no/lov/1961-05-12-2/§7>.

Åndsverkloven, (1961B), Love 6, URL: <https://lovdata.no/lov/1961-05-12-2/§6>

Åndsverkloven, (1961C), Lov 40, URL: <https://lovdata.no/lov/1961-05-12-2/§40>

Åndsverkloven, (1961D), Lov 2, URL: <http://www.lovdata.no/all/hl-19610512-002.html>

Åndsverkloven, (1961E), Lov 39, URL: <http://www.lovdata.no/all/hl-19610512-039.html>.