

2007-04-24

Hur sambruka kommunala e-tjänster? - Arbetsformer och affärsmodeller för semi-öppen programvara

Göran Goldkuhl, Thomas Rosén, Anna Öhrwall Rönnbäck
Forskningsnätverket VITS
Institutionen för Ekonomisk och Industriell utveckling (IEI)
Linköpings universitet

Sammanfattning:

Sambruk är en intresseorganisation för kommuner. Syftet är att bedriva gemensam verksamhetsutveckling och sambruk av elektroniska tjänster, som förväntas ge effektivisering i form av ökad produktivitet med bibehållen eller högre kvalitet för medborgare. Ambitionen är att sänka kommunernas kostnader för utveckling och drift av e-tjänster samt att minska ledtiderna för utveckling och införande av tjänsterna. En möjlighet att uppnå dessa mål är att öka användningen av öppen programvara inom Sambruk. Erfarenheterna från genomförda och pågående upphandlingar visar dock på betydande svårigheter att etablera öppen programvara som ett reellt alternativ för Sambruk och dess medlemskommuner. Utifrån dessa erfarenheter har ett förslag till ”semi-öppen” programvara utarbetats. Detta förslag har formulerats av forskare som deltar i och studerar Sambruks arbete med verksamhetsutveckling och e-tjänsteinförande. Förslaget är inspirerat av principer bakom öppen programvara, men utformat för att kunna fungera inom sammanslutningen Sambruk. Förslag till licensform kallas här för Sambruk Community License (SCL). Arbetsformer och affärsmodeller som understödjer denna typ av licensform beskrivs i rapporten. Jämförelser görs mellan öppen programvara, semi-öppen programvara (SCL) och proprietär (sluten) programvara.

Denna forskning har utförts med finansiellt stöd från VINNOVA (FoU-program om Gränsöverskridande offentliga e-tjänster).

1 Inledning

1.1 Bakgrund

Det finns starka förväntningar på utveckling av kommunala e-tjänster som riktar sig mot medborgare, företag och andra organisationer. Detta kommer bl.a till uttryck i statsmakternas politiska skrivningar om 24-timmarsmyndigheten och offentliga e-tjänster (se t.ex SOU, 2005). Förväntningar riktas mot såväl statliga myndigheter som landsting och kommuner. Inom kommunerna finns det inom många områden potential för e-tjänster. Målsättningar handlar om:

- Förbättra kommunal service och tillgänglighet för medborgarna
- Sänka kommunernas kostnader för den kommunala förvaltningen
- Bidra till en samordnad offentlig förvaltning

Sveriges 290 kommuner står inför stora utmaningar. Hur skall utveckling av kommunerna ske? Ska varje kommun utveckla sina e-tjänster eller ska man förlita sig på leverantörsmarknadens utbud? Utveckling som drivs av varje enskild kommun skulle bli samhälls-ekonomiskt kostsamt. Användning av standardprodukter från IT-leverantörer bör vara en mer ekonomisk väg. Detta kan emellertid leda till att kommuner behöva invänta produktutveckling hos leverantörer och dessutom få små möjligheter till styrning och påverkan på sådan produktutveckling. Som ett tredje alternativ finns möjligheten att en grupp av kommuner tar initiativet och driver utveckling och gör det i samverkan med leverantörer. Ett sådant initiativ finns genom Föreningen Sambruk (www.sambruk.se).

Sambruk initierades som ett informellt nätverk 2002 och ombildades till en formell intresseförening 2005. Sambruk har fn 71 kommuner som medlemmar. På senare tid har även två regionförbund tillkommit. Verket för förvaltningsutveckling (Verva) och Sveriges Kommuner och Landsting (SKL) är också medlemmar i Sambruk (Sambruk 2007). Visionen är gemensam verksamhetsutveckling och sambruk av elektroniska tjänster som förväntas ge effektivisering i form av ”ökad produktivitet med bibehållen eller högre kvalitet och samtidigt skapa ökad tillgänglighet och sänkta kostnader för att producera och leverera framtidens e-tjänster” (Sambruk, 2006:1). Följande viktiga målsättningar finns för Sambruks arbete:

- Arrangera gemensam utveckling av kommunala e-tjänster
- Skapa gemensam teknisk och juridisk grund för sambruk av e-tjänster
- Sänka kommunernas kostnader för utveckling och drift av e-tjänster
- Minska ledtiderna för utveckling och införande av tjänsterna

En av drivkrafterna bakom bildandet av Sambruk var att skapa en väl fungerande marknad för offentliga e-tjänster (Sambruk, 2004). För att åstadkomma detta behövs ett mer balanserat förhållande mellan kund och leverantör. Leverantörssidan domineras av en handfull aktörer och kundsidan av många delvis små aktörer (290 kommuner där mediankommunen har 15 000 invånare och endast 10 kommuner är större än 100 000 invånare). För den enskilda kommunen kan e-tjänstutveckling innebära att man ofta är hänvisad till en leverantör på grund av den brist på öppenhet som interna IT-system har. För att erhålla fungerande e-tjänster krävs ofta att man kopplar samman dem, genom datautbyte, med interna verksamhetssystem. Att uppnå en sådan systemintegration är ofta ett svårt arbete och kräver dessutom normalt en medverkan från aktuell leverantör. En viktig drivkraft bakom Sambruks bildande var därför att skapa samarbetskonstellationer för att kommunsidan ska kunna utgöra en kompetent och kraftfull beställare (Sambruk 2004).

Sambruks arbete med e-tjänsteutveckling drivs bl.a med FoU-stöd¹ från VINNOVA (Sambruk, 2004). I detta FoU-projekt medverkar forskare från Forskningsnätverket VITS (www.vits.org) vid Linköpings universitet och Internationella Handelshögskolan i Jönköping. Forskarna roll är att studera och stödja (genom modeller och metoder) kommunernas arbete med e-tjänsteutveckling. FoU-projektet skall också studera hur affärsmodeller utvecklas mellan kommuner/Sambruk och leverantörer. Ett uttalat intresse i detta sammanhang finns från Sambruk för öppen programvara: ”De grundläggande värderingar som Sambruk vilar på är likartade med de värderingar som är vägledande för ’Open source’ konceptet. ’Open source’ lösningar kommer därför att övervägas för alla här ingående e-tjänster” (Sambruk, 2004). I ansökan ges vidare förslag till möjliga affärsmodeller som: ”1) Sambruk eller annan aktör förvaltar specifikation och källkod enligt ’Open source’-konceptet. 2) Kommunerna genom Sambruk äger och förvaltar såväl specifikationer som källkoden”. Genom denna skrift tar vi forskare ett steg mot operationalisering av dessa mål och affärsmodeller.

Arbete inom Sambruk vad gäller e-tjänsteutveckling bedrivs normalt genom samverkan mellan ett antal kommuner. I en projektgrupp deltar representanter från olika kommuner (från aktuellt verksamhetsområde). Sådana projektgrupper arbetar med att ta fram förstudier och kravspecifikationer. Med sådana gemensamt utvecklade kravspecar kan sedan Sambruk och kommunerna gå vidare i en upphandlingsprocess. På detta sätt skapar man ett mer genomarbetat underlag vid upphandlingen och kan agera mer samlat och professionellt gentemot leverantörer.

1.2 Syfte och målgrupper

Denna rapport är skriven av tre forskare som är knutna till Sambruks-arbetet genom VINNOVA-projektet (se ovan). Vi har utifrån planer och uppdrag i FoU-projektet (Sambruk, 2004) utvecklat ett förslag som bygger på inspiration från området ’öppna programvaror’, Open Source Software (OSS). I denna skrift utvecklar och redovisar vi förslag till arbetsformer, affärsmodeller och överenskommelser (inkl licensformer) för semi-öppen programvara avsedd för Sambruk. Med semi-öppen menar vi att programvaran inte är helt öppen enligt vedertagna OSS-principer. Vi presenterar en modell som innebär öppenhet inom en avgränsad krets, dvs vad som ibland kallas ’community source’. Det handlar alltså om öppen programvara bland Sambruks medlemmar.

Sambruks medlemmar behöver utveckla sätt att anskaffa och förvalta e-tjänsteapplikationer genom gemensamma insatser. För detta behövs lämpliga affärsmodeller. E-tjänster som utvecklas och används av en eller flera kommuner behöver på ett enkelt sätt kunna spridas till fler. De affärsmodeller som tillämpas idag är oftast baserade på traditionella köpar-leverantörskontrakt mellan en kommun och en leverantör. Samtidigt möjliggörs via SKLs Programverket² den andra ytterligheten; öppen källkod och kostnadsfri delning av gemensamma resurser. För mellantinget, när flera kommuner vill dela initiala och förvaltningskostnader, saknas idag regler och rutiner. För att underlätta sambrukande av e-tjänster mellan kommuner (och andra offentliga aktörer) föreslås därför här ”Sambruk Community Licence”. Förslaget baseras på erfarenheter från Sambruks hittills genomförda och pågående upphandlingar av e-tjänster samt en jämförelse mot de vanligt förekommande licensformerna kring Open Source (General Public License, GPL) och proprietära programvarulicenser (leverantörsägd programvara och kod).

¹ FoU-projektet bedrivs inom ramen för VINNOVA:s program ”Gränsöverskridande offentliga e-tjänster” (GRO).

² <http://www.programverket.org>, utvecklad av Sveriges Kommuner och Landsting (SKL)

Denna rapport bygger på en tidigare intern Sambruksrapport som diskuterats i olika sammanhang. Den har nu omarbetats till en extern version för att uppnå en ökad spridning och diskussion av dessa idéer.

Målgrupper för denna rapport är medlemmar i Sambruk och andra praktiker samt forskare. Vi vänder oss till personer med intresse för frågor om

- IT-utveckling inom offentlig sektor
- Affärsmodeller för IT-utveckling
- Öppen programvara

1.3 Tillvägagångssätt

Som framgått ovan vänder vi oss med denna rapport till både forskare och praktiker. Vi tillämpar ett praktikvetenskapligt angreppssätt (Goldkuhl, 2006) med syften 1) att bidra till konkret och specifik kunskap för våra samarbetspartners i Sambruk, 2) att bidra med användbar kunskap för andra praktiker samt 3) att bidra till den vetenskapliga kunskapsbildningen inom detta område. Denna forskningsansats som tillämpas kan karaktäriseras som praktikforskning (ibid) som just har sådana flertaliga syften. Aktuell praktikforskning utförs också som aktionsforskning (se t.ex Susman & Evered, 1978) och designforskning (se t.ex Hevner et al, 2004). Utifrån våra observationer och analyser har vi utvecklat förslag (till affärsmodeller, arbetsformer och licensformer). Dessa olika förslag har successivt förfinats genom diskussioner och värderingar. Vi har utifrån ett aktionsforskningsperspektiv studerat och värderat hittillsvarande upphandling i Sambruk och utifrån denna diagnos har vi skisserat designförslag. Detta innebär att vi har genomfört faserna ”diagnosis” och ”action planning” enligt den vedertagna aktionsforskningsmodellen (Susman & Evered, 1978).

Vi planerar att senare ytterligare precisera dessa förslag samt vår förhoppning är att några av dessa förslag kommer till praktisk användning för att därefter kunna utvärderas vad gäller praktisk tillämpning och effekter.

2 Förutsättningar och observationer

En viktig poäng i Sambruks arbete är gemensam specificering och upphandling av e-tjänsteapplikationer. Detta innebär också att de utvecklade/anskaffade e-tjänsteapplikationerna behöver fungera i olika *IT-systemmiljöer*. De olika kommunerna kan ha olika IT-system internt i verksamheten och den nya e-tjänsteapplikationen skall kunna fungera tillsammans med olika sådana typer av interna system. Kommunerna har olika systemarv. Utifrån dessa förutsättningar har Sambruk formulerat en systemarkitekturpolicy kallad *Öppen Teknisk Plattform – ÖTP* (Sambruk, 2005). Denna arkitekturmodell bygger på en modulär arkitektur samt principen om datasamverkan via *öppna standardiserade gränssnitt*. Detta innebär ett datautbyte enligt väldefinierade protokoll (standardmeddelanden³).

Genom sådana arkitekturprinciper finns möjlighet att en anskaffad e-tjänsteapplikation kan samverka med olika typer av system (dvs en hög grad av interoperabilitet), under förutsättningen att dessa system kan utbyta data enligt specificerade protokoll för standard-

³ Standardmeddelanden kallas inom ÖTP för ’nyttomeddelanden’.

meddelanden. ÖTP skapar också förutsättningar för att ersättning (utbyte) av IT-system kan ske på ett mer kontrollerat sätt.

Det finns en strävan mot ökad användning av *öppna programvaror* inom offentlig sektor. Inom EU har initiativ till detta tagits. Bl.a finns en särskild licensform framtagen: European Union Public Licence - EUPL (EU, 2007). I Sverige har också initiativ tagits. I en pågående (våren 2007) statlig utredning om IT-standardisering studeras detta närmare (SOU, 2006). Även Sveriges Kommuner och Landsting (SKL) önskar öka användningen av öppna programvaror inom offentlig förvaltning. SKL har startat Programverket, en programpool där kommuner och landsting kan deponera källkod med öppna programvarulicenser. Detta sk programarkiv har varit i drift ungefär ett år. Via Programverket tillhandahålls även tjänster kring hur öppna programvaror upphandlas, utvecklas och förvaltas. På sikt ska Programverket bli ”ett kompetenscenter kring öppen programutveckling” och ”en arena för erfarenhetsdelning och kunskapsuppbyggnad” som syftar till att ge förutsättningar för en effektiv IT-utveckling med öppna program (Programverket, 2007).

Programverket och Sambruk skulle kunna bli goda komplement till varandra. I Sambruk görs gemensamma upphandlingar, med utveckling av kravspecifikationer och verksamhetsprocesser, samt delas erfarenheter och gemensamma utvecklingsprocesser. Vid gemensamma möten sprids kunskap om möjligheterna att gemensamt agera beställare till de kommuner och på senare tid även regionförbund som är medlemmar. Genom Programverket kan en kanal erhållas för att föra ut resultat av gemensamma satsningar till fler kommuner och myndigheter i offentlig sektor.

I praktiken genomförs medlemssamverkan inom Sambruk genom att projekt sätts upp för specifikation och upphandling samt eventuellt också drift av gemensamma e-tjänster. Medlemskommunerna delar på kostnaderna i enlighet med invånarantal. Några projekt har *avropat* e-tjänster *enligt ramavtal* från Verva medan andra går till *gemensam upphandling* för specifika e-tjänster, sk förhandlad upphandling. Vid avrop enligt ramavtal sluts kontrakt mellan kund och leverantör på traditionellt sätt, och endast leverantörer inom ramavtalet är aktuella. Vid förhandlad upphandling kan även andra leverantörer komma i fråga.

Det finns i Sambruk också en önskan att *attrahera nya leverantörer*, gärna små eller medelstora företag med lokal förankring: ”Sambruk förenar det verksamhetsnära lokalt förankrade användarperspektivet med en helhetssyn som bygger på en gemensam målbild och en utvecklad horisontell samverkan hos flera lokala aktörer.” (Sambruk, 2006:2). Det kan emellertid vara svårt att attrahera små/medelstora leverantörer i mindre kommuner via centrala stora upphandlingsprocesser. En fråga är hur man kan skapa affärsmodeller som lämpar sig för lokala upphandlingar även i ett sambruksförfarande.

Erfarenheter i Sambruk visar att några kommuner är med och ”drar det tunga lasset” i att genomföra upphandlingar, för att det därefter ska vara möjligt för fler att ta del av den upphandlade e-tjänsten. Att upphandla en kommunal e-tjänst är ofta ett omfattande arbete, innefattande processutveckling, kravspecifikationsarbete, samordning med upphandlande sambruksmedlemmar och leverantörer. Arbetet kan också inrymma utredning av gränssnitt mot andra myndigheter och organisationer och dess IT-system samt ibland utredning av lagar och bestämmelser för aktuell e-tjänst. Det är viktigt att det finns *incitament för kommuner att gå före och driva utvecklings- och upphandlingsarbete*. Sambruk kan inte fungera om alla ”sitter på läktaren” och väntar på att andra ska göra arbetet. Här behövs troligen klarare

principer för att sprida kostnader och låta de sambruksmedlemmar som tagit kostnader för gemensam utveckling få vissa fördelar gentemot övriga tillkommande.

Vi har som en del av vårt forskningsuppdrag kontinuerligt följt några av Sambruks upphandlingsprocesser. Öppen källkod har i dessa processer inte funnits med som ett reellt alternativ trots att det ibland förts upp på agendan. Projektledare, upphandlingskonsulter och upphandlingsgrupper har avfärdat detta som ett alltför oklart alternativ. Man *saknar kunskap om* hur man ska upphandla *öppen programvara* eller programvara för gemensamt ägande. Man vet hur man upphandlar proprietär programvara och valt denna säkra väg istället för en osäker väg med öppen programvara. Det saknas affärsmodeller, arbetsformer och licensformer för ett alternativ till traditionell upphandling av standardprogramvara av proprietär typ.

Det förefaller också bland Sambruks medlemmar finnas en osäkerhet beträffande *gemensamt ägandet av källkod*. Det finns en oro att skapa ett nytt "Kommundata"⁴, vilket också påverkar diskussionen. Man önskar att Sambruk ska vara "slimmad" organisation och inte dra på sig omfattande resurser.

Vi uppfattar utifrån våra observationer av pågående upphandlingsprocesser att det finns ett stort behov av att artikulera vad ett delvis öppet programvarualternativ skulle kunna innebära för Sambruk. Om man i upphandlingssituationer inte har en klar bild av vad inköp av utvecklad programkod för sambruksägande skulle kunna innebära och få för konsekvenser, finns det stor risk att man istället "rullar på i samma hjulspår" som tidigare och köper dyra standardprogram med begränsade möjligheter till inflytande över dess vidareutveckling.

Det är inte bara via gemensam specifikation och upphandling som programvaror kan komma Sambruk och dess medlemmar till del. Ett alternativ är att *befintliga programvaror skänks* till Sambruk från någon av dess medlemmar. Hur detta sker är ännu inte fullständigt juridiskt reglerat. Vi uppfattar att det finns ett stort behov att klargöra hur sambruksägd programvara skall hanteras. Det har redan skänkts vissa programvaror till Sambruk och det finns i dagsläget en osäkerhet hur dessa skall spridas vidare till intresserade kommuner (bl.a juridiska former för detta) och hur kostnader för programvarors vidareutveckling skall fördelas. Vi uppfattar också att dessa oklarheter har lett till att kommuner avvaktar med att skänka programvaror till Sambruk. Genom att *gåvoalternativet* inte har klargjorts bättre finns här hinder för exploatering av en viktig del i sambrukstanken.

Det finns ett antal viktiga frågor att resa avseende specifikation, upphandling/anskaffning, drift och förvaltning av e-tjänster som till exempel: - Hur kan man fördela arbete mellan kommuner vid specifiering och upphandling? - Hur skrivs avtal med leverantörer på bästa sätt? - Vem styr gemensam vidareutveckling av e-tjänster? - Hur styrs gemensam vidareutveckling av e-tjänster? - Hur delar man på kostnader? - Kan även mindre kommuner utan egen kompetens inom öppen programvara ta del av de gemensamma resurserna? - Hur ska förhållanden mellan Sambruksmedlemmar regleras? - Hur kan kommunägd programvara tillföras Sambrukskollektivet på enkelt sätt?

Nedan (särskilt avsnitt 4) presenteras ett förslag till hur Sambruk kan äga, sambruka och dela kostnader för e-tjänster. Genom förslaget vill vi visa hur anskaffning, ägande, underhåll och förvaltning, samt support kan kombineras och göras tillgängligt för Sambruks medlemmar.

⁴ Kommundata var ett kommunägt IT-företag för utveckling och drift av kommunernas datatjänster. Det ansågs av många ineffektivt och tungrott, och såldes senare ut.

3 Licensformer för programvara

Vi ger nedan kort redogörelse för olika typer av licensformer för programvara. I huvudsak finns det två 'extremformer'; öppna programvaror (Free and Open Source Software) samt slutna (proprietära) mjukvaror (Closed Source Software). Vi kommer även att visa fördelar respektive nackdelar med varje typ.

3.1 Öppna programvaror

Öppna programvaror är det svenska samlingsnamnet för Free Software (FS) och Open Source Software (OSS). Utan att gå in i detalj på vad som skiljer de två begreppen, så handlar FS mer om en moralisk, ideologisk ståndpunkt om hur programvaror skall användas och distribueras, medan OSS har en mer pragmatisk inriktning som handlar om att de särskilda processerna kring utvecklingen av öppna programvaror gör dem bättre än slutna.

Öppen programvara som företeelse har blivit uppmärksammat eftersom det har utpekats som konkurrenshot av Microsoft, världens största tillverkare av slutna programvaror (som dock under senare tid också har startat och underhåller flera projekt med öppna programvaror). I praktiken innebär det att en stor mängd utvecklare, både på sin fritid och i många fall även under arbetstid, tillsammans skapar mjukvaror som de sedan delar ut *fritt* för andra att *använda, distribuera* och *vidareutveckla*. Filosofin bakom öppna programvaror handlar om tillgänglighet, interoperabilitet (dvs samfunktionalitet mellan programvaror) och modularitet. Ofta används begrepp som "rapid prototyping" eller "release fast and often" (Raymond, 2001). Vissa pekar på att utvecklingen av mjukvaran liknar livet, vidareutvecklingen tar aldrig slut, det finns ingen sista utgåva förrän ingen längre använder programmet.

Källkoden, själva "receptet" för vad datorn skall utföra, är den viktigaste dokumentationen inom öppen programvara. En öppen arkitektur och lättläst kod med tydliga kommentarer är det som eftersträvas, och grupptricket är starkt! Om en programmerare inte kan skriva " snygg kod" med bra kommentarer är det inte säkert att hans eller hennes förändringar läggs in i mjukvaran – på så sätt reglerar utvecklarna sig själva. För större system, som exempelvis operativsystemet Linux, tillkommer förfarandet med en kärna av programmerare och "löjtnanter" som tillsammans med Linus Torvalds styr vad som får läggas in i en utgåva (release) och när den ska släppas. En annan viktig princip är att undvika att "uppfinna hjulet" om och om igen. Därför är öppna standarder viktiga, dvs gemensamma metoder eller regler, eftersom de gör att utvecklarna kan återanvända det som utvecklats och istället fokusera på det som de anser behöver programmeras. Att uppfinna nya metoder och regler för varje programmeringsprojekt skulle bara kosta tid. Utvecklingen regleras under licensavtal, exempelvis General Public License (GPL). De mest kända exemplen på öppna programvaror är ovan nämnda operativsystemet Linux, webbserverprogrammet Apache och den svensk-finska databasen MySQL. De här exemplen har hundratals, troligen tusentals utvecklare och en mycket stor mängd användare. Till exempel MySQL har över åtta miljoner kända installationer världen över (MySQL, 2006). Det är också till största delen öppna programvaror som driver Internet, något som kanske är okänt för de flesta.

Det finns även företag involverade kring öppen programvara, några exempel är Novell, Red Hat och Sun Microsystems. De två förstnämnda arbetar främst med sina egna Linux-distributioner, medan den sistnämnda ansvarar för flera projekt, bland andra programspråket Java, operativsystemet Solaris (även OpenSolaris) och kontorsprogrammet StarOffice (även OpenOffice.org).

Alla öppna programvaror är dock inte så stora. Programvaruutvecklingen sker i sk öppna projekt. En av de stora projektplatserna, SourceForge.net, har över 130 000 registrerade öppna utvecklingsprojekt och 1,4 miljoner registrerade användare och utvecklare (SourceForge.net, 2006). Nackdelen med så många projekt är att uppmärksamheten inte fördelas jämt och det är en stor del av projekten som dör ”sotdöden” eftersom intresset för dem helt falnat eller övergått i ett annat liknande projekt.

Inom forskningen studerar man främst innovationskraften hos användarna och utvecklarna, men även drivkrafter för utvecklare och användare att arbeta med öppen programvara. Kopplat till Sambruks forskningsprojekt studeras affärsmodeller för öppen programvara⁵.

De främsta **fördelarna** med öppna programvaror handlar om *frihet* (att använda, förändra och vidare distribuera), *demokrati* (offentlighetsprincipen, full insyn i koden), *innovation* (ju fler kockar desto bättre soppa), *billigare underhåll* samt *främjandet av öppna standarder*. Källkoden finns spridd och tillgänglig – försvinner en leverantör, kan en ny anlitas.

Nackdelar är att det *saknas en klar huvudman* som tar ansvar för utvecklingen. Vidare kan det uppstå en *osäker förvaltnings- och supportsituation*, med *bristande framtidsplanering* och ofta *total avsaknad av deadlines*. Saknas intresse för programvaran, försvinner de flesta goda effekter. Ibland kan det även finnas *bristfällig dokumentation* (för icke-tekniska användare).

3.2 Slutna mjukvara

I datorhistorien är slutna programvaror någonting relativt ungt. I början levererades datamaskinerna utan att vara förprogrammerade och kunden fick själv utveckla de program som behövdes. Under hand utvecklades kommersiella program och i och med att företag började ta betalt för mjukvarorna kan man tala om att koden ”läses in” och blir slutna. En av förgrundsfigurerna inom Free Software, Richard Stallman, beskriver hur det beteendet fick honom att sluta sitt arbete på MIT och börja utveckla fri programvara (Stallman 2002).

Anledningarna till att företagen ville ta betalt var för att komplexiteten hos maskinerna och programmen ökade exponentiellt samtidigt som användarna ökade och de var beredda att betala för vissa funktioner. I en början utvecklades operativsystemet av tillverkaren av hårdvaran, men under 1980-talet fanns det små företag som gjorde mer generella system som fungerade på flera maskiner. Ett av dem var företaget Microsoft som utvecklade ett operativsystem som hette MS-DOS. De fick ett avtal av IBM om att leverera till den nya produktserien ”Personal Computers” (persondatorer, PC). Vad som hände därefter är historia.

All tid som företagen lägger ned i programutveckling behöver täcka motsvarande kostnader, annars vore företaget olönsamt. Leverantörer som kan sprida sina kostnader på exempelvis hårdvaror eller eftermarknadstjänster har lättare att släppa sin källkod fri än de som enbart lever på sin mjukvara.

Slutna programvara utvecklas på en mängd olika sätt – men det finns ett stort mått av planering och design. Det finns ofta en vägkarta (roadmap) där nya funktioner presenteras och ett ungefärligt datum när den nya versionen skall vara färdig. I många fall ges versionen ut innan den är helt färdig, det talas om ”good enough”, dvs tillräckligt bra för att köras. Exempelvis levererade Microsoft Windows XP med ett stort antal kända fel – dock finns det

⁵ Öppen källkod som strategisk resurs, forskningsprojekt finansierat av Lundberg forskningsstiftelse 2005-2008, se http://www.eki.liu.se/forskning/forskningsprojekt/open_source.

ofta en plan för hur fel rättas till. Ett sådant förfarande ifrågasätts ofta av programmerare som anser sig veta bättre, inte minst inom open source-sfären, och har bidragit till att vissa företag har dåligt rykte bland programmerare. I dagstidningar kan man läsa om strider som blossar upp mellan företag och företrädare inom programvaruutvecklingen vad gäller utvecklingsmetoder och standardutveckling.

Förespråkare för sluten programvara talar ibland om ”security by obscurity”, vilket innebär att eventuella säkerhetshål är svåra att finna för sådana som vill producera skadlig kod – detta till motsats till öppen programvara, där källkoden finns tillgänglig och öppen för alla att se. Dokumentationen är viktig för slutna mjukvaror. Eftersom användarna inte har tillgång till källkoden måste all information som kunderna behöver ha göras tillgänglig. System för support och underhåll är likaså viktiga.

Den allra viktigaste skillnaden är att om en sluten programvara läggs ner, eller om företaget köps upp eller går i konkurs, finns stor risk att källkoden försvinner. I vissa fall finns det avtal mellan leverantören och kunden om så kallad källkodsdeponering, men det är något som används endast i undantagsfall. Dessutom kan källkoden vara i sådant skick att den är svår att använda sig av. När Netscape skänkte sin källkod som öppen programvara 1998 visade den sig vara så dålig att utvecklarna bestämde sig att börja om från början (Wikipedia, 2006).

Fördelar med sluten programvara är exempelvis *tydlig huvudman med plan för utvecklingen av produkten, etablerad förvaltning och support, samt god dokumentation.*

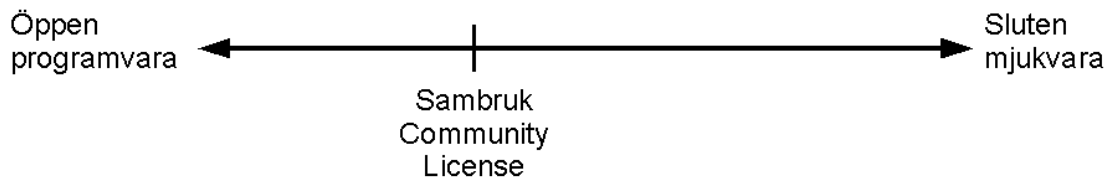
Nackdelar med sluten programvara är exempelvis *inlåsnings effekter, ointresse att införa öppna standarder. Det kan även vara svårt att påverka leverantören för förändringar; som användare är man helt i händerna på leverantören vad gäller beslut om vidareutvecklingar. Kan vara dyrt i längden. Risk för problem om leverantören lägger ner programvaran, köps upp eller läggs ned. Det kan finnas säkerhetsproblem.*

4 Arbetsformer och affärsmodeller för semi-öppen programvara

4.1 Sambruk Community License

Vi föreslår en mellanform av öppen och sluten programvara som kombinerar fördelarna med båda, men även eliminerar de flesta nackdelarna. Vi kallar den *Sambruk Community License* (SCL), för att markera att det som licensen omfattar är öppet för medlemmar i sammanlutningen (”community”). Detta är bakgrunden till att vi kallar den ”semi-öppen” programvara.

Även om flera aktörer förordar öppna programvaror så finns det allvarliga risker med att alltför tidigt släppa programvaror helt fritt. Det är viktigt att få till stånd fungerande arbetsformer för förvaltning och vidareutveckling av programvaror. Detta kan vara svårt vid ett helt öppet programvarualternativ. Även om SCL är en blandform så finns det mer av öppen programvarumodell i upplägget än sluten mjukvara, vilket illustreras i figur 1.



Figur 1: Sambruk Community License som mellanform

Tanken är att behålla en stor del av filosofin bakom öppna programvaror, exempelvis att *en programvara aldrig blir helt färdig, arkitekturen skall vara öppen och modulär, samt att alla anpassningar skall återföras fritt tillgängliga för programvarans användare*. Samtidigt skall det finnas en trygghet i form av en huvudman (föreningen Sambruk) som ansvarar för att det sker förvaltning och underhåll av programvarorna.

En viktig drivkraft är att få medlemmarna intresserade av att delta tidigt i utvecklingsprocessen och bidra med kompetens och tid för att få till stånd kravspecifikationer inför anskaffning. Det skall finnas incitament för ett sådant tidigt deltagande i form av lägre kostnader för licenser, service och support samt vidareutveckling. Modellen bygger på en uppdelning i två skeden, *anskaffning* (initialskedet) och *fortvarighet* (när programvaran används och förvaltas).

4.2 Anskaffning

Anskaffningsfasen kan sägas bestå av tre delar: *specifikation, upphandling* och *programvaruutveckling*. Vi har identifierat ett antal typfall för den här fasen:

1. En grupp av Sambruks medlemmar tar tillsammans fram en gemensam specifikation. Baserat på denna genomför de upphandling och utser leverantör(er) för utveckling/anpassning av programvara. Denna modell förefaller vara den mest tillämplade modellen inom Sambruk för närvarande. Den har klara fördelar genom att flera kommuners kunskaper, ansvar och engagemang utnyttjas för kravspecifiering och upphandling (val av leverantör). Modellen kan dock bli tungrodd (då många kommuner deltar) och ledtiderna kan bli långa.
2. En grupp av Sambruks medlemmar skapar en gemensam specifikation, men en enskild kommun sköter upphandlingen och utser leverantör(er) för programutveckling. Modellen har vissa likheter med ovanstående (1), genom att flera kommuner deltar i specifikationen för att säkerställa bredd i lösningsförslag. I detta fall delegeras därefter upphandlingen till *en* kommun för att därigenom förhoppningsvis effektivisera genomförandet. T.ex. kan en kommun med goda leverantörskontakter och ramavtal välja en lämplig leverantör för programvaruutvecklingen. Sambruk och de i specifikationen deltagande kommunerna kan välja att tillämpa denna modell för effektivisera sitt eget genomförande och då uppdra åt eller överenskomma med en enskild kommun att ansvara för anskaffning.
3. En grupp av medlemmar skapar en gemensam specifikation, men en enskild utvecklar programvaran själv (egenutveckling). Denna modell liknar 2, men här utför en kommun med egna resurser programvaruutvecklingen. Denna modell bygger också på att de övriga kommunerna beslutat att låta en kommun ansvara för anskaffning av programvara. Denna modell passar förmodligen främst för anskaffning av mindre applikationer där god utvecklingskompetens finns enkelt tillgängligt inom en kommun. Liksom för (2) så handlar det om att effektivisera genomförandet.

4. En enskild kommun skapar en specifikation och sköter upphandlingen och utser leverantör(er) för programutveckling. Detta kan ske på eget initiativ eller i samråd med Sambruk som då låter en enskild kommun gå före.
5. En enskild kommun skapar specifikation och utvecklar programvaran själv (egenutveckling). Denna modell liknar 4, men här har kommunen egna resurser för programutveckling.

Semi-öppen programvara enligt SCL skulle fungera på så sätt att när programutvecklingen är färdig (till en fungerande version) överlämnas ägandet av programvaran till Sambruk. Det kallas att programvaran ”hanteras under licensformen SCL”. Om utveckling har skett hos leverantör skall överlämnandet till Sambruk vara specificerat i upphandlingsavtalet. Det kan i upphandlingsavtalet finnas specificerat eventuella rättigheter för leverantören att vidareutveckla och använda programvaran. Om en kommun har utvecklat programvaran själv (typfall 3 och 5), skänks den till Sambruk som *benefik gåva*. De fem olika typfallen finns sammanfattade i tabell 1 nedan.

Tabell 1 Olika fall av anskaffning

	Specifikation	Upphandling	Program- utveckling	Överlåtelse till Sambruk
Fall 1	Gemensam (flera kommuner)	Gemensam	Extern leverantör	Enligt upphandlingsavtal
Fall 2	Gemensam (flera kommuner)	Enskild kommun	Extern leverantör	Enligt upphandlingsavtal
Fall 3	Gemensam (flera kommuner)	(Enskild kommun)	Egenutveckling (enskild kommun)	Benefik gåva (enligt överenskommelse i specifikation)
Fall 4	Enskild kommun	Enskild kommun	Extern leverantör	Enligt upphandlingsavtal (benefik gåva)
Fall 5	Enskild kommun	(Enskild kommun)	Egenutveckling (enskild kommun)	Benefik gåva

Den grundläggande sambruksmodellen för anskaffning av programvara är att det sker en gemensam specificering (i första hand fall 1). Den är lämplig i många situationer, men behöver kompletteras med andra former som beskrivits ovan i text och tabell 1. Det kan många gånger vara värdefullt för Sambruk (och därmed för dess medlemmar) att kunna använda det som görs lokalt av enskilda medlemmar. Det finns en stor outnyttjad potential i att lokala utvecklingar och andra initiativ skulle kunna komma andra kommuner och andra offentliga aktörer till del. Det gäller att finna ett sätt att undvika ett ”Kommundata-syndrom” med en stor resursslukande och tungrodd organisation. Genom att utforma arbetsformer för programvaruhantering enligt öppen källkodsfilosofi kan man undvika att all utveckling måste administreras via Sambruks ledning, utan ansvaret kan fördelas på många parter. På detta sätt kan man ta tillvara den omfattande lokala utveckling som sker bland Sambruks medlemskommuner.

Många applikationer kan vara tämligen enkla till sin karaktär och är knappast lämpliga för en mer omfattande ”byråkratisk” hantering som gemensam specificering och upphandling. För en enkel applikation är det kanske lämpligare att den initialt utvecklas av en enskild kommun för att därefter på ett kontrollerat sätt spridas till övriga kommuner där sedan vidareutveckling kan ske enligt öppna principer; se vidare nedan.

De fem olika typfallen kompletterar varandra och skall ses som fem möjliga varianter av anskaffning av programvara. Det handlar om att utnyttja olika sätt att få in programvaror till Sambruk för att de därefter skall kunna användas och förvaltas. Det handlar om ”*fem vägar in*” till en sambruksstyrd användning och förvaltning av programvaror.

4.3 Fortvarighet

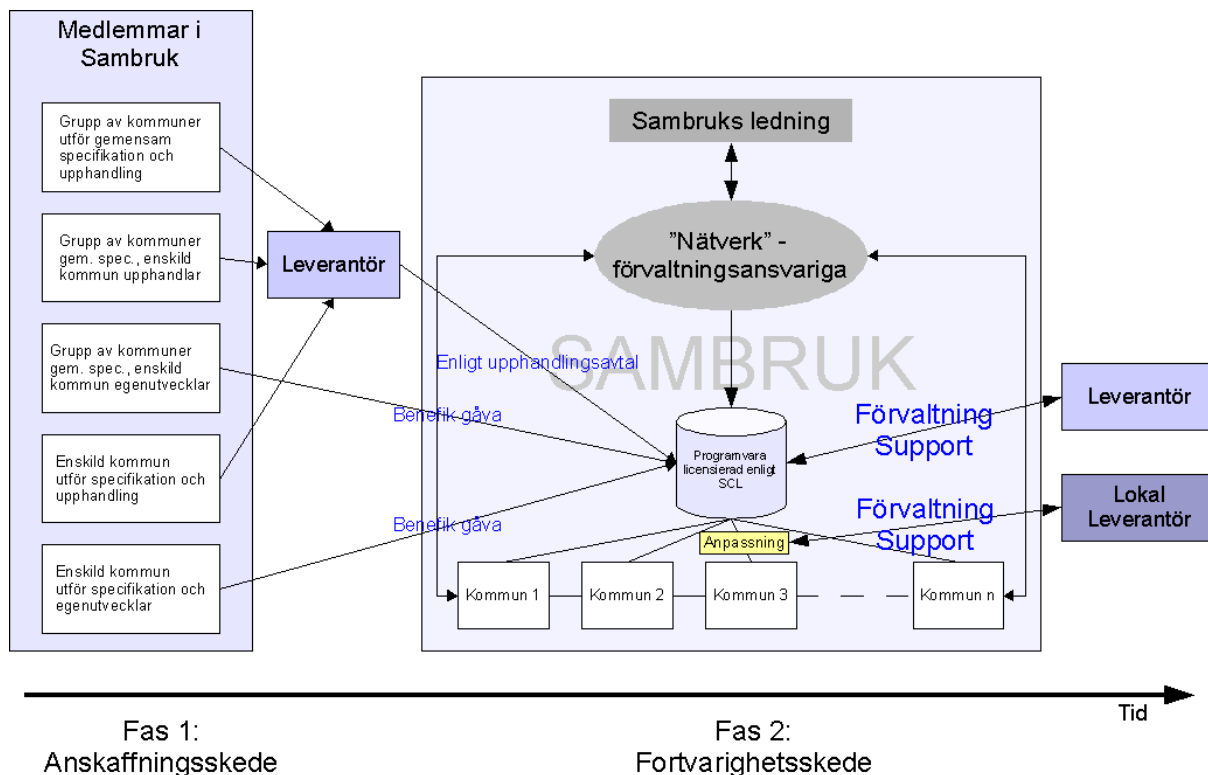
När det gäller användning och förvaltning (inklusive vidareutveckling) som följer efter att programvaran anskaffats på något av de fem sätt som klarlagts ovan föreslår vi att Sambruk som huvudman är formellt ansvarigt för programvaran. Programvaran licensieras under SCL, vilket skulle innebära att medlemmarna betalar en prenumeration till Sambruk för att täcka kostnaderna för upphandlingen och utvecklingen, samt förvaltning av programvaran.

För att en programvara skall kunna upptas som en SCL-programvara krävs förmodligen en del egenskaper hos den. Viktigt är att den är utvecklad baserad på öppna standarder; i första hand de principer som finns specificerade i Sambruks Öppen Teknisk Plattform (ÖTP). Det handlar om att skapa programvaror som har hög grad av interoperabilitet via öppna standardiserade gränssytor, dvs. möjlighet att fungera tillsammans med andra redan existerande och framtida applikationer. Man kan tänka sig att det sker en inledande kvalitetsgranskning och klassificering av inkommande programvaror. Vissa programvaror, första hand de som anskaffats enligt kravspecifikationer baserade på ÖTP, bör uppfylla sådana ÖTP-kriterier. Det kan komma in andra programvaror till Sambruk som dock inte till fullo uppfyller dessa ÖTP-kriterier; förmodligen i sådana fall som benefika gåvor från enskilda kommuner. För sådana programvaror kan eventuellt inte en förvaltningsorganisation etableras på samma sätt som för ÖTP-klassade programvaror.

Prenumerationsavgiften för en SCL-programvara ger medlemmen rätt till användning av denna programvara, löpande generell förvaltning (korrigerig och vidareutveckling), viss support och tillgång till källkoden. De medlemmar som har arbetat med initial upphandling och utveckling i anskaffningsskedet får rabatt i förhållande till nedlagda kostnader. Prenumerationen erbjuder viss förvaltning och support, men viktigt är att varje betalande medlem själv har rätt att vidareutveckla programvaran för egna behov och teckna egen förvaltning, eller kanske komplettera supportavtalet med lokala leverantörer.

Förvaltningen och vidareutvecklingen av programvaran drivs som ett öppet programvaruprojekt inom Sambruk, där projektservern (med programvaror i olika versioner) är tillgänglig för samtliga betalande medlemmar. Sambruk ansvarar för att upphandla förvaltningen, men det är *ett förvaltningsråd* som ansvarar för styrningen av vidareutvecklingen av den generella programvaran. Alla betalande medlemmar erbjuds plats i förvaltningsrådet som kan ses som ett nätverk av de Sambrukskommuner som använder programvaran enligt SCL.

Även om Sambruk ansvarar för en generell version, har medlemmarna rätt att göra egna anpassningar av programvaran som inte omfattas av förvaltningsavtalet eller supportavtalet (se ovan). Enligt SCL skall alla förändringar göras tillgängliga för Sambruk och de kommuner som har sambrukslicens för den aktuella programvaran. En enskild medlem som har anpassat programvaran själv får också ansvara för support och underhåll för den egna varianten. Genom att lokala anpassningar görs fritt tillgängliga för övriga användare utnyttjas fördelarna i ”*öppna principer*” som att olika användare kan bygga vidare på varandras lösningar i ett kreativt samspel. Figur 2 sammanfattar hela modellen.



Figur 2: Sambruksmodellen för Community Licenses

4.4 Affärsmodeller vid anskaffning

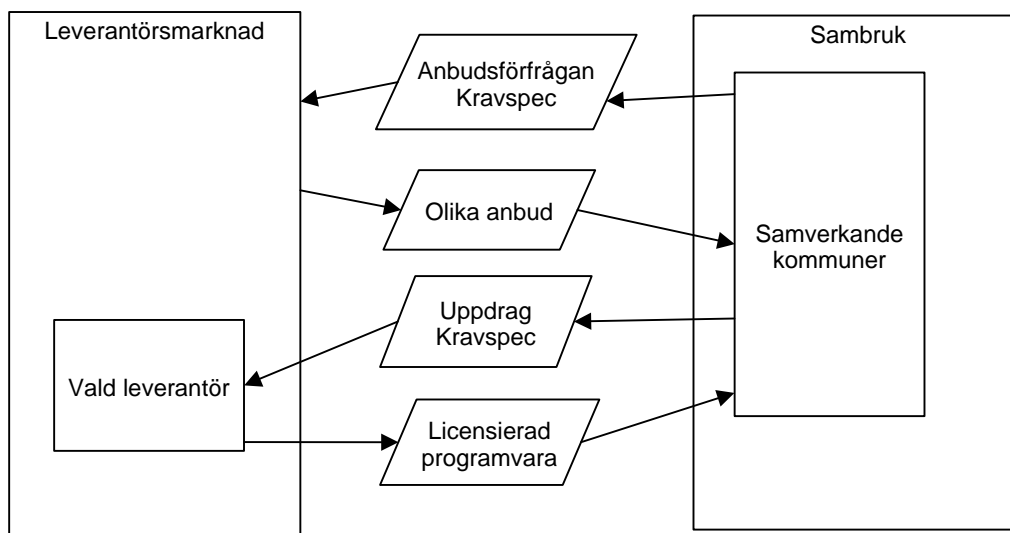
Vi kommer nedan att göra en viss ytterligare precisering av de affärsmodeller som blir aktuella i de fem anskaffningsfallen beskrivna ovan; avsnitt 4.2 (tabell 1). Vi gör detta med hjälp av grafiska illustrationer⁶ (figur 3-8). Vi baserar analysen av de olika affärsmodellerna på en generisk modell för affärssamverkan mellan kunder och leverantörer; den sk BAT⁷-modellen (Goldkuhl & Lind, 2004).

I dagsläget sker anskaffning av programvara (e-tjänsteapplikationer och andra typer av system) i stort sett enligt fall 1. En viktig skillnad finns dock. Det är de upphandlande kommunerna som erhåller programvaran och inte hela sambrukskollektivet. Denna programvara är normalt också licensierad till de upphandlande kommunerna. Vi har illustrerat samspelet mellan aktörerna i en modell (figur 3) nedan. En delmängd av Sambruks medlemmar genomför specifikation och upphandling. Upphandling innebär att utifrån lämnade anbud att ta ställning till vilken leverantör⁸ av programvara som ska väljas. Den av kommunerna framtagna kravspecifikationen spelar flera roller i processen: 1) Den är ett underlag vid anbudsfrågan till leverantörer, 2) den är ett underlag för kommunernas val av leverantör och 3) den är ett underlag för den valda leverantörens utveckling/anpassning av programvara.

⁶ Vi följer i princip notation för sk samverkansgrafer; se Röstlinger & Goldkuhl (2006).

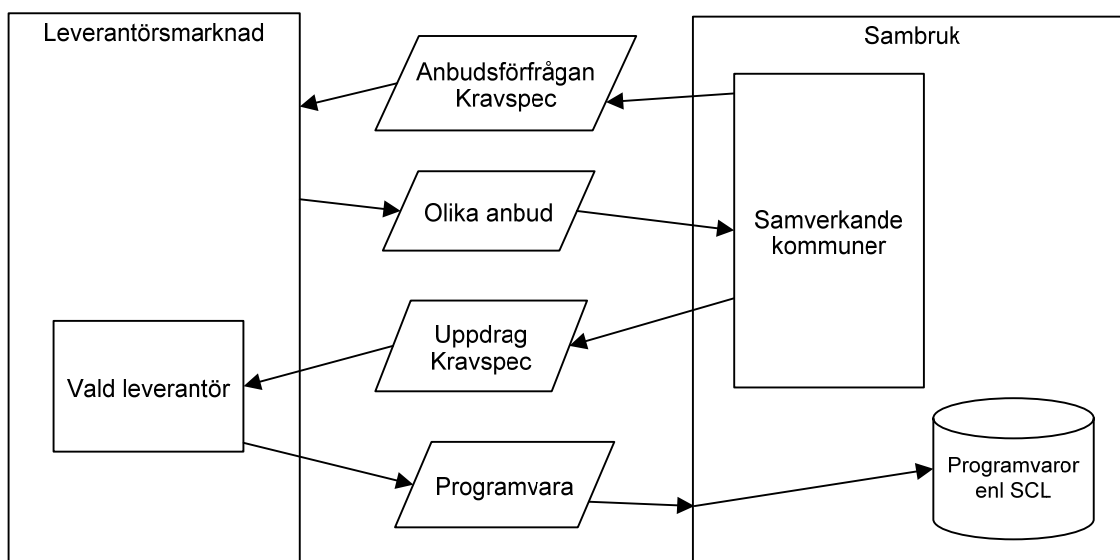
⁷ BAT står för Business Action Theory.

⁸ Ibland kan flera företag vara aktuella som leverantörer.



Figur 3: Anskaffning enligt etablerad sambruksmodell (gemensam specificering och upphandling av proprietär programvara)

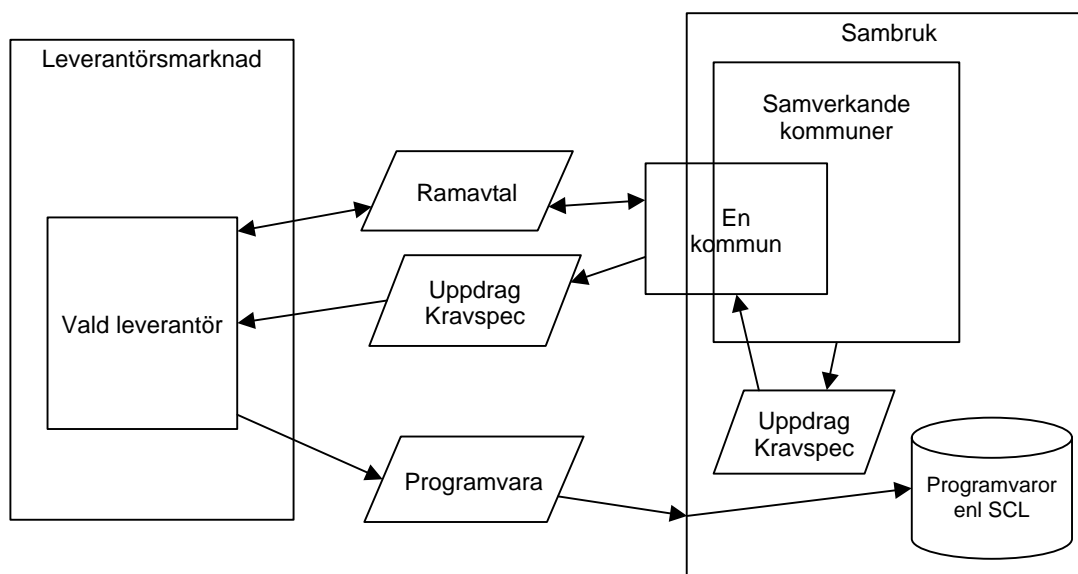
Fall 1 för SCL-anskaffning bygger på den etablerade modellen för sambruksanskaffning beskriven ovan (figur 3). Här finns dock en viktig skillnad. Mottagare av den utvecklade programvaran blir Sambruk som då blir ägare av programvaran och licensierar den öppet (inom Sambruk) till dess medlemmar. Figur 4 beskriver detta fall av anskaffning.



Figur 4: Anskaffning - fall 1 (gemensam specificering och upphandling av programvara)

Fall 2 har likheter med fall 1, men här delegeras upphandlingen till en kommun. Det kan finnas bland de samverkande kommunerna någon kommun som har goda etablerade relationer med en leverantör som bedöms som lämplig att svara för programvaruutvecklingen. Ett ramavtal kan finnas med denna leverantör och den specifika kommunen. Baserat på sådant ramavtal utväljer den specifika kommunen aktuell leverantör. I avtal mellan kommunen och leverantören behöver det framgå att mottagaren av den utvecklade programvaran inte bara är den upphandlande kommunen utan Sambruk som medlemsorganisation. Denna typ av

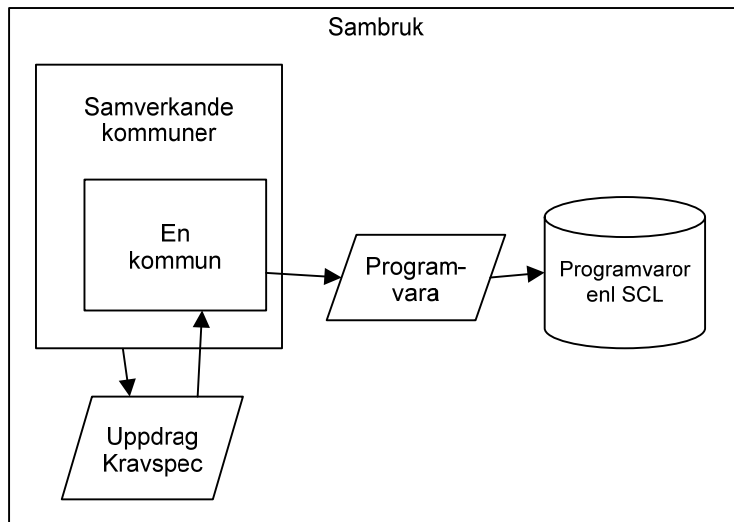
anskaffning är illustrerad i figur 5. Där framgår också att de samverkande kommunerna delegerar (upptrar åt) den specifika kommunen att anskaffa programvaran. Den aktuella kommunen är beskriven både som del av de samverkande kommunerna (inom Sambruk) och som en "fri aktör". Det ramavtal som upphandlingen baseras på är (förmodligen) ingått av kommunen som autonom juridisk person och inte i egenskap av Sambruksmedlem. Denna anskaffningsmodell innebär alltså att ett antal Sambrukskommuner ger uppdrag till en medlemskommun att anskaffa programvara via extern marknad och de avtal som kommunen har med aktörer på denna marknad. Denna anskaffningsmodell skall inte uppfattas som ett sätt för Sambruk att kringgå regelverk för upphandling (LOU). Den specifika kommunen skall givetvis basera sitt agerande på upphandlingsregler⁹. Ett antagande som denna modell baseras på att det kan gå enklare och smidigare (än fall 1) om man utnyttjar enskilda kommuners redan ingångna ramavtal med leverantörer.



Figur 5: Anskaffning - fall 2
(gemensam specificering, en kommun upphandlar programvara)

Fall 3 har likheter med fall 2 beskrivet ovan. I fall 3 agerar en specifik kommun på basis av en gemensam framtagen specification. I detta sker ingen extern upphandling utan kommunen utvecklar själv programvaran med hjälp av egna personella resurser. Inget samspel sker med externa leverantörer i detta fall som framgår av figur 6. Denna anskaffningsmodell är som synes tämligen "enkel" och kan möjliggöra ett snabbt utvecklingsgenomförande under förutsättning att tillräcklig kompetens finns.

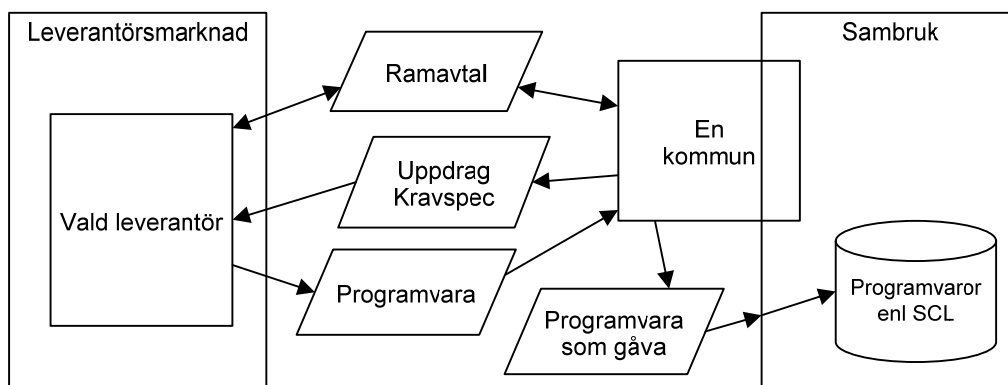
⁹ Vi har inte genomfört någon djup upphandlingsjuridisk konsekvensanalys på dessa anskaffningsmodeller. Detta behöver göras i samband med att modellerna konkretiseras ytterligare.



Figur 6: Anskaffning - fall 3
(gemensam specificering, en kommun utvecklar programvara)

Fall 4 och 5 skiljer sig från de övriga fallen på en avgörande punkt. Här sker ingen gemensam specificering utförd av medlemskommuner. En enskild kommun svarar för specificering av programvara. Detta innebär att denna utveckling förmodligen inte sker direkt i Sambruks regi som fall 1-3. En enskild kommun tar initiativ till utveckling och driver detta själv. Själva upphandlingen kan ha direkta likheter med upphandlingen i fall 2 ovan. Kommunen utnyttjar sina etablerade leverantörskontakter (ingångna ramavtal). När programvaran är utvecklad (med kommunen som ägare av denna programvara) kan sedan kommunen överlåta denna programvara till Sambruk som en benefik gåva. Programvaran kan då hanteras enligt SCL bland Sambruks medlemmar. De får tillgång till programvara som en enskild programvara tagit initiativ till och anskaffat.

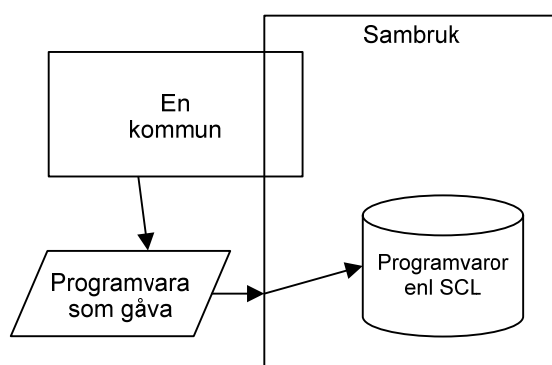
Utveckling och anskaffning av programvaran kan ske helt oberoende av intentioner att skänka programvaran till Sambruk. Utvecklingen kan ske helt med den aktuella kommunens egna behov för ögonen. Först senare kan ett intresse identifieras att skänka programvaran till Sambruk.



Figur 7: Anskaffning - fall 4
(enskild kommun specificerar och upphandlar programvara)

Fall 5 har likheter med fall 4. Här utförs inte heller någon gemensam specificering av medlemskommuner i Sambruk. En enskild kommun tar initiativ till utveckling av programvara. I detta fall sker ingen upphandling av programvara från extern leverantör utan kommunens egna resurser utnyttjas för utvecklingen. Efter utvecklingen – efter en kortare eller längre period – skänkes programvaran till Sambruk för att användas av medlemmarna under licensformen SCL.

I både fall 4 och 5 sker drivs utveckling av en enskild kommun och inte av Sambruk i form av en projektgrupp sammansatt av flera medlemskommuner. Vi har hävdad att utvecklingen i fall 4-5 sker oberoende av Sambruk och på kommunens egna initiativ. Det är tänkbart att något samråd eller åtminstone informationsutbyte kan ske mellan den aktuella kommunen och Sambruk innan utveckling påbörjas. Sambruks ledning kanske ser att man inom Sambruk och bland dess medlemmar inte har tid att sätta på ett speciellt utvecklingsarbete i närtid. Sambruk uppfattar det positivt att någon kommun kan ta på sig enskilt att driva utvecklingen och därefter skänka programvaran som en SCL-resurs.



Figur 8: Anskaffning - fall 5
(enskild kommun specificerar och utvecklar programvara)

5 Jämförande analys

SCL är en typ av grupplicens. Som licensform finns som en mellanform mellan helt öppna och proprietära programvaror. Syftet med licensformen och ackompanjerande arbetsformer är att möjliggöra praktiskt ”sambrukande” av programvara för såväl små som stora medlemmar. Samtidigt eftersträvas en mer balanserad marknadssituation med aktiva beställare/kunder samt underlättande att välja lokala leverantörer. Att kombinera den framgångsrika filosofin bakom öppna programvaror, och samtidigt eliminera många av nackdelarna med densamma kan vara ett steg mot en öppnare marknadssituation och effektivare verksamheter i kommuner och landsting. Tabell 2 summerar skillnaderna mellan licensformerna:

Tabell 2: Sammanfattning, licensformer

	Öppen programvara	Semi-öppen programvara (SCL)	Sluten programvara
Tydlig huvudman	-	X	X
Tydlig utvecklingsplan (roadmap)	-	X	X
Förekomst av dokumentation: Teknisk/Icke teknisk (T/It)	T	T/It	It
Förvaltningsansvarig	-	X	X
Supportansvarig	-	X	X
Påverkan – startskede av utveckling	-	X	- ¹
Påverkan – under utveckling	X	X	X ²
Påverkan – under förvaltning/vidareutveckling	X	X	X ²
Tillgång till källkod	X	X	- ³
Rätt till egna anpassningar	X	X	-
Rätt till valfri leverantör av support/underhåll	X	X	-
Öppna standarder	X	X	- ⁴

1. Möjligheter till påverkan kan finnas vid utveckling av ny, eller upphandlad utveckling av programvara. Gäller ej etablerade mjukvaror.
2. Påverkan beror på leverantören och kontakten med denne. I allmänhet relativt svårt att påverka.
3. Vissa leverantörer kan ge tillgång till källkod, eller del av källkod under specifika villkor.
4. En del av inlåsnings effekterna som ett mjukvaruföretag vill skapa är så kallade *de facto*-standarder, som bygger på slutna, eller egna protokoll, exempelvis formatet ".doc".

Nackdelen med att ha en grupplicens så som föreslås med Sambruk Community License är att programvaran inte blir fri. Fri programvara skulle kunna finna andra, externa intressenter, som vill utveckla koden (som på Programverket), dock är troligtvis det intresset relativt lågt utanför det offentliga rummet¹⁰. Eftersom det finns ett starkt önskemål av effektivisering och kostnadstäckning, kan inte licensen vara helt öppen från början. SCL kan vara ett lämpligt inbrott för kommuner till mer öppna programvaror.

Vissa programvaror under SCL skulle av Sambruk efter viss tidsperiod kunna släppas helt fria (till 'General Public License' - GPL). Programvaran skulle då t.ex kunna överföras till Programverket för en bredare spridning och användning. Underhålls- och supportavtal skulle dock fortfarande kunna upphandlas.

För att Sambruk skall kunna dra största möjliga nytta av en grupplicens, är det viktigt att samtliga kommuner och projekt är medvetna om vilket regelverk som gäller. Det är troligt att företagen till en början kommer att vara skeptiska eller avvaktande till den här typen av avtal,

¹⁰ Forskning visar att det kan vara svårt att attrahera användare/programmerare av öppna programvaror, det kan krävas en del arbete för att få fart på ett sådant projekt (cf. Goldman & Gabriel, 2005).

vilket kan medföra svårigheter i upphandlingsprocessen om inte alla förutsättningar är klarlagda.

Vi har i denna skrift försökt klargöra vad semi-öppen programvara skulle kunna innebära för Sambruk och dess medlemmar. Arbetsformer och affärsmodeller har beskrivits. Olika argument för en SCL-lösning har presenterats. Vi vill samtidigt poängtera att vi inte ser SCL som den enda rätta lösningen för Sambruk. SCL bör kunna samexistera med andra alternativ, både proprietära programvaror och helt öppna programvaror. Semi-öppen programvara är en filosofi som behöver vara öppen gentemot andra alternativ.

Vilka argument ser vi finns mot SCL? Det beror på vilket perspektiv man väljer att betrakta SCL utifrån. Om man ser det ifrån perspektivet proprietära programvaror har vi identifierat följande motargument:

1. Sambruk och kommuner saknar erfarenhet från upphandling av programvaror enligt SCL-modellen
2. Det är endast leverantörer som kan sköta förvaltningsstyrning professionellt (ligger utanför Sambruks och kommuners kompetens)
3. Sambruk bör av principiella skäl inte äga programvaror

Dessa argument behöver skärskådas. Det är helt riktigt att Sambruk i dagsläget saknar erfarenhet av upphandling av semi-öppen programvara (argument 1). Det är just därför SCL-konceptet har utvecklats och denna rapport har framtagits. Man behöver börja arbeta efter SCL-principer för att lära sig hur det kan gå till (1, 2). Övning ger färdighet! Vi har svårt att tro att kommuner inte kan utveckla kompetens att sköta förvaltningsstyrning på ett professionellt sätt (2). Rädslan att Sambruk skall utvecklas till ett nytt tungrott 'Kommundata' kanske kan vara befogad. Denna oro kan föra med sig inställningen att Sambruk av principiella skäl inte skall äga programvaror (3). Upplägget med SCL innebär dock inte att man bygger upp en omfattande organisation/bemanning inom Sambruk, utan istället att förvaltningsarbete sker bland medlemskommuner och externa leverantörer. Vi kan inte se några hållbara principiella skäl mot att Sambruk skall äga programvaror. Om detta möjliggör ökad förvaltningskontroll bland kommunerna och lägre programvarukostnader för kommunerna så bör detta vara ett lämpligt medel. Vi ser sambruksägandet som instrumentellt (och inte 'principiellt', som ett värde i sig); dvs vi betonar att sambruksägandet just kan möjliggöra bättre styrning och lägre kostnader.

Om man ser det ifrån perspektivet öppna programvaror har vi identifierat följande motargument:

1. Sambruk ska inte bara vara till för sina medlemmar utan för alla Sveriges kommuner
2. Det är viktigt att tillämpa öppen källkod fullt ut

Att även bidra till andra kommuner är ett vällovligt syfte (1). Sambruk är en medlemsorganisation av kooperativ karaktär. Ett viktigt syfte är att bidra till medlemmarnas nytta av ekonomiskt och annat slag. Att anlägga ett ekonomiskt tänkande – vad medlemmarna får ut av sina medlemsavgifter – kan inte vara ett felaktigt resonemang. Att skapa nytta och ekonomiska fördelar för sina medlemmar måste vara en grundläggande bevekelsegrund för en medlemsorganisation som Sambruk. Det är ett val för en kommun att delta i Sambruk eller inte. Medlemsavgiften och nedlagt engagemang i Sambruk skall göra meningsfull skillnad för kommunen. Argumentet att tillämpa öppen källkod fullt ut (2) är också ett principargument (i

stil med argument 3 ovan om att Sambruk inte bör äga programvara). Eftersom vi anlägger en instrumentell hållning till ägandeformer så avvisar vi också detta. Det handlar om att välja de ägande- och licensformer som är mest adekvata för Sambruk. Vi delar många av de värderingar som ligger bakom 'öppen källkod', men ser det inte som ett självändamål, utan istället som ett medel för att öka kvalitet, utvecklingskontroll och kostnadskontroll.

Vi sammanfattar argumenten för SCL-modellen på följande sätt:

- Undvika dyra licenskostnader; inte köpa samma sak gång på gång
- Ta kontroll över förvaltning/vidareutveckling
- Förenklad följsamhet till ÖTP-principer
- Viktigt att påbörja SCL-upphandling för att få erfarenheter
 - Kontinuerligt lärande
 - Utvecklad beställarkompetens
- Det ska finnas mening (incitament) att delta i Sambruk
 - Medlemmar ska ha fördelar
- Det finns en stark innovationskraft i öppen källkod
 - Men former kan variera; SCL anpassat för Sambruk
 - SCL ett steg mot helt öppen programvara
- Ökad handlingsfrihet – ökad handlingsrepertoar för Sambruk
 - Olika anskaffnings-/upphandlingsalternativ
 - Kunna mottaga/hantera benefika gåvor från medlemmar

6 Fortsatt arbete

SCL (Sambruk Community License) existerar inte i dagsläget som formulerad licensform. Den behöver formuleras, analyseras och diskuteras. Det finns olika öppna licenser som kan ligga till grund som förebilder för denna licensutformning. Det är viktigt att en granskning sker av juridisk expertis för att undvika potentiella fällor i framtiden.

Vi planerar att formulera förslag till Sambruk Community License baserat på principer i denna skrift. Det krävs emellertid inte bara att denna licensform tas fram. Det finns behov av andra typer avtal/dokument som vi avser att ge förslag till. Vi ser behov av följande dokument:

- 1) Sambruk SCL licensavtal (mellan Sambruk och "prenumeranter")
- 2) Sambruks uppdragsavtal (mellan upphandlande kommuner som uppdragsgivare och extern leverantör som uppdragstagare och Sambruk som produktmottagare)
- 3) Sambruk SCL gåvoavtal (mellan kommun som gåvogivare och Sambruk som gåvomottagare)
- 4) Sambruk samverkansöverenskommelse - SCL förvaltningsråd (mellan Sambruks styrelse och förvaltningsråd med verkställande utskott och "prenumeranter"/medlemmar i förvaltningsråd)

Vi kommer att delta i arbete med affärsmodeller, arbetsformer, licensavtal i Sambruk och då både bidra till arbetet och studera det. Framtida rapporter kommer bl.a att inriktas mot olika erfarenheter från detta förändringsarbete mot ett semi-öppet programvarualternativ. Olika hinder och möjligheter behöver studeras empiriskt.

Referenser

Tryckta källor

- Goldkuhl G (2006) What does it mean to serve the citizen? - Towards a practical theory on public e-services founded in socio-instrumental pragmatism, accepted to *the International Workshop on E-services in Public Administration*, Borås
- Goldkuhl G, Lind M (2004) The generics of business interaction - emphasizing dynamic features through the BAT model, in *Proc of the 9th Intl Conference on the Language Action Perspective (LAP2004)*, Rutgers University
- Goldman R, Gabriel R P (2005) *Innovation Happens Elsewhere*. Morgan Kaufmann. San Francisco.
- Hevner A R, March S T, Park J, Ram S (2004) Design science in information systems research, *MIS Quarterly*, Vol 28 (1), pp 75-15
- Raymond E S (2001) *The Cathedral & The Bazaar*. Sebastopol. O'Reilly.
- Sambruk (2004). E-tjänster för Sambruk, beviljad ansökan hos VINNOVA E-tjänster för offentlig sektor, diarienummer 2004-02447.
- SOU (2005) E-tjänster för ett enklare och öppnare samhälle, Statens Offentliga Utredningar, SOU 2005:119
- Susman G I, Evered R D (1978) An assessment of the scientific merits of action research, *Administrative Science Quarterly*, Vol 23 (4) p 582-603
- Stallman R M (2002) *Free Software Free Society*. Boston. GNU Press.
- Röstlinger A, Goldkuhl G (2006) *Grafnotation för SIMM metodkomponenter*, VITS, Institutionen för ekonomisk och industriell utveckling, Linköpings universitet

Elektroniska källor

- EU (2007) European Union Public Licence V.1.0. (Elektronisk) Tillgänglig: <http://ec.europa.eu/idabc/servlets/Doc?id=27470> (2007-04-10)
- MySQL (2006) *MySQL Awarded GSA Contract Schedule 70*. (Elektronisk) Tillgänglig: http://www.mysql.com/news-and-events/press-release/release_2006_01.html (2006-12-18)
- Programverket (2007) *Inledning och Om Programverket*. (Elektronisk) Tillgänglig: <http://www.programverket.org/> (2006-12-04, 2007-01-29).
- Sambruk (2005) Öppen teknisk plattform (ÖTP) V1.2. (Elektronisk) Tillgänglig: http://www.sambruk.se/download/18.70c5f153108d7c77dae800043/Sambrplform_OTP_v12.pdf (2007-04-10)
- Sambruk (2007). (Senast uppdaterad 2007-01-22) Aktuell medlemsförteckning finns på www.sambruk.se. Tillgänglig: <http://www.sambruk.se/vanstermeny/medlemmar.4.3633a839107753a07be8000123.html> (2007-01-29)
- Sambruk (2006). (Senast uppdaterad 2006-01-27) *Om Sambruk*. (Elektronisk) Tillgänglig: <http://www.sambruk.se/vanstermeny/omsambruk.4.3633a839107753a07be8000115.html> (2006-12-04)
- Sambruk (2006). (Senast uppdaterad 2006-10-30) *Vision och mål* (Elektronisk) Tillgänglig: <http://www.sambruk.se/vanstermeny/omsambruk/visionochmal.4.1e380ebd1078f9ae1cc80003480.html> (2006-12-04)
- SOU (2006) Kommittédirektiv IT-standardiseringsutredningen, 2006:36, (Elektronisk) Tillgänglig: http://www.sou.gov.se/kommittedirektiv/dir2006_36.pdf (2007-04-09)

SourceForge.net (2006). *Welcome to SourceForge.net*. (Elektronisk) Tillgänglig:
<<http://sourceforge.net/>> (2006-12-18)
Wikipedia (2006). Netscape Navigator. (Elektronisk) Tillgänglig:
<http://en.wikipedia.org/wiki/Netscape_Navigator>