

En systemutvecklingsmetods möjlighet att fånga och överföra kunskap

- En studie av RUP

Magisteruppsats, 10 poäng, i Informatik
Framlagd 17 juni 2005

Författare: Carin Olsson
Handledare: Bo Andersson

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

LUNDS UNIVERSITET
Institutionen för informatik

**En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
- En fallstudie av RUP**

© Carin Olsson

Magisteruppsats framlagd juni, 2005
Omfång: 60 sidor
Handledare: Bo Andersson

Resumé

När man utvecklar eller bygger något så bygger de som är med i utvecklingsprojektet upp en kunskap kring produkten, en kunskap som bygger på erfarenhet och som kan vara svår att förmedla till andra individer som inte varit delaktiga. Denna studies huvudsyfte är att undersöka hur systemutvecklingsmetoden RUP (Rational Unified Processes) tar tillvara på kunskap, som bildas i projekt. Kunskapsbegreppet i uppsatsen syftar till tyst kunskap som baseras på erfarenheter hos individen. Och hur denna kunskap sedan ska kunna överföras till de som ska förvalta produkten. Detta har gjorts genom att undersöka hur metoden används i praktiken. Till grund för de frågor som ställdes vid intervjuerna låg litteraturstudier. Undersökningen har genomförts genom intervjuer på företag, med personer som arbetar med RUP och personer som utbildar inom RUP.

Intervjuerna gav en insikt om att RUP inte fungerar i sig själv. Det i huvudsak undersökta företaget har fått lägga till egna rutiner för att stödja kunskapsöverföring. Företaget använder sig även av PPS (Praktisk ProjektStyrning) i sina projekt, för styrning. Det är väl värt att veta att respondenterna anser att man på Alfa har blivit mycket bättre på kunskapsöverföring sen de införde RUP, detta beroende på att användandet av RUP förespråkar en bättre struktur i dokumentation.

Nyckelord

Systemutveckling, systemutvecklingsmetod, Rational Unified Processes, kunskap, kunskapsöverföring.

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Förord

Jag vill tacka min handledare Bo Andersson för stöd och hjälp, samt respondenterna. Jag vill även tacka ledarna på Mörtsjölägrete för ert stöd och all er hjälp i alla upptänklige frågor och ett stort tack till Maria Johansson på Siemens, Finspång, för stöd, idéer och hjälp.

Innehållsförteckning

1	Inledning	1
1.1	Bakgrund	1
1.2	Frågeställning, problem, syfte och avgränsningar	1
1.3	Metoddiskussion	2
1.3.1	<i>Datainsamlingsmetoder och tekniker</i>	2
1.3.2	<i>Dataanalysmetoder och tekniker</i>	3
1.3.3	<i>Rapporteringsform och källkritik</i>	4
1.3.4	<i>Om respondenterna och företagen</i>	4
1.3.5	<i>Val av intervjufrågor</i>	5
1.4	Sammanfattning av inledning	6
2	Litteraturförankring	7
2.1	Teoretisk bakgrund	7
2.2	Rational Unified Processes	10
2.2.1	<i>Bakgrund</i>	10
2.2.2	<i>Faser inom RUP</i>	10
2.2.3	<i>Roller och artefakter inom RUP</i>	13
2.2.4	<i>RUPs dokument och artefakter i förhållande till kunskap</i>	13
2.2.5	<i>Kritik mot RUP</i>	14
2.3	Kort om PPS	14
2.4	Sammanfattning av litteratur förankringen	16
3	Empirisk studie	17
3.1	Bakgrund	17
3.1.1	<i>Allmänt om RUP och Alfa</i>	17
3.1.2	<i>Alfa och artefakterna</i>	18
3.2	Mognad och RUP	18
3.3	Kommunikation	19
3.4	Kompetens i förvaltningen	19
3.5	Kunskapsöverföring i RUP	20
3.5.1	<i>Checklistor</i>	22
3.6	Sammanfattning av den empiriska studien	23
4	Diskussion	24
4.1	Användning av metoder	24
4.2	Tillvaratagande och överförande av kunskap	25
4.3	Sammanfattning av diskussionen	25
5	Avslutning	27
5.1	Slutsatser	27
5.2	Självkritik	27
5.3	Diskussion kring vetenskaplig och etisk kvalitet	28
5.3.1	<i>Vetenskaplig kvalitet</i>	28
5.3.2	<i>Etisk kvalitet</i>	28

5.4	Förslag till framtida forskning	29
6	Bilagor.....	30
	Bilaga 1: Intervjuformulär	30
	Bilaga 2: Intervju med Arne på Alfa	31
	Bilaga 3: Intervju med Adam på Alfa.....	35
	Bilaga 4: Intervju med Anders på Alfa.....	39
	Bilaga 5: Intervju med Bengt på Beta.....	44
	Bilaga 6: Intervju med Göran på Gamma	46
	Bilaga 7: Roller inom RUP	49
	Bilaga 8: Artefakter inom RUP	50
7	Referenser.....	53

Figuröversikt

Figur 1.1	Denna bild ska visa på skillnaden mellan transkriberat material och det material som sedan redovisas i uppsatsen.....	3
Figur 2.1:	Sambandet mellan aktiviteter, faser och iterationer i RUP (Wiktorin, 2003, s.95) .	11
Figur 2.2	Visar sambanden mellan de olika modellerna som används i projekt (TietoEnator, 2005-05-12).....	14
Figur 2.3	Figuren ska visa på hur beslutspunkterna i PPS kopplas till de olika faserna i projektet (TietoEnator, 2005-05-12).....	15

Tabellöversikt

Tabell 1.1.	Tabellen ska visa sambandet mellan de intervjuade och de företag på vilka de arbetar.....	5
-------------	---	---

1 Inledning

Denna magisteruppsats behandlar överförande av kunskap i samband med att ett systemutvecklingsprojekt blir ett systemförvaltningsobjekt. Inledningskapitlet ska ge läsaren en bakgrund och motivering till val av ämnesområde samt tydliggöra den problemställning på vilken undersökningen bygger. Under denna rubrik behandlas även syfte, avgränsning och de metoder som används för att göra undersökningen.

1.1 Bakgrund

I systemutvecklingsprojekt skapas en hel del kunskap, en kunskap om systemet som senare ska användas av dem som ska förvalta det. Med kunskap menas erfarenheter som individerna tillgodoser sig genom att de medverkar i projektet (Rolf, 1981). Denna uppsats söker svar på hur kunskapen som bildas under systemutvecklingsfasen tas till vara för att senare kunna överbrygga kunskapsglapp som kan uppstå i samband med att systemet tas i drift. Andersen (1994) menar att begreppet kunskap bör skiljas från begreppet information, kunskap är något individuellt som innehåses av en individ. Kunskap skapas utifrån en persons förutsättningar och bakgrund. Därför är det svårare att överföra kunskap än information, eftersom var persons kunskap är baserad på dennes uppfattningar om sin verklighet.

Bakgrunden till mitt ämnesval kommer ifrån tidigare studier i samband med kandidatuppsatsen. Jag och min medarbetare upptäckte då att företaget vars förvaltningsorganisation vi undersökte uttryckte en hel del oro över detta kunskapsbortfall som uppkom i överförandet från utvecklingsfas till förvaltningsfas. Backlund (2004) menar att då kunskap är kopplat till dess kontext så kan den inte överföras mellan exempelvis organisationer utan att man tar till sig att kunskapen är individuell för de personer som innehar den. Alltså är detta en form av tyst kunskap (Meggerle & Steen, 2002). Jag menar att då förvaltningsorganisationen kan ses som skild från utvecklingsorganisationen kan här uppstå ett glapp i överförandet av kunskap.

I tidigare studier har det även visats att just överlämnandet av produkter, i form av exempelvis system eller program, från utvecklingsfasen till förvaltning är något som behandlas som ett svart hål och det är svårt att få en klar bild av hur en sådan överlämning bör gå till. Andersen (1994, s.47) skriver att "I och med implementeringen är utvecklingsarbetet avslutat. Systemet tas i dagligt bruk." och han lämnar i och med detta ämnet outforskat. Därför ska denna uppsats försöka svara på vad det är som ska ske efter att systemet tas i bruk, med fokus på kunskapen som ska överföras och tillvaratas tillsammans med produkt och som därmed ska med in i nästa fas, förvaltning och drift.

1.2 Frågeställning, problem, syfte och avgränsningar

Hur kan användandet av en systemutvecklingsmetod hjälpa till med överföringen av kunskap vid övergången från systemutvecklingsprojekt till förvaltningsfas?

Syftet med denna uppsats är att undersöka hur en av de mest frekvent använda systemutvecklingsmetoderna stödjer överförandet av kunskap från systemutvecklingsfasen till förvaltningsfasen. Målet är att ge läsaren en klar bild vad som i systemutvecklingsmetoden kan tillämpas för att stödja kunskapsöverföring.

Avgränsningen för uppsatsen baseras på litteraturstudien som entydigt säger att Rational Unified Processes (RUP) idag är den systemutvecklingsmetod som är förhärskande samt den metod som täcker de största delarna av systemutvecklingens livscykel (Lunell, 2003 & Wiktorin, 2003). Det företag där jag har gjort mina intervjuer använder RUP i sin systemutveckling.

Undersökningen utreder om systemutvecklingsmetoden RUP stödjer kunskapsöverföring och hur detta sedan realiserar i praktiken på ett företag. Jag har studerat material, artiklar och beskrivningar för systemutvecklingsmetoden för att sedan använda mig av detta material i utformningen av intervjuunderlag. Val av företag utgick från att de använder RUP i sina systemutvecklingsprojekt. Jag tänker emellertid beskriva RUP som helhet på ett generellt plan där jag endast fokuserar på behandling och metoder som styrker tillvaratagande av den, i systemutvecklingsfasen, bildade kunskapen.

Min förfinade frågeställning är;

Hur kan användandet av systemutvecklingsmetoden RUP hjälpa till med överföring av kunskap vid övergången från systemutvecklingsprojekt till förvaltningsfas?

Det förväntade resultatet ligger i att peka på vilka delar, om några, av systemutvecklingsmetoden RUP (Rational Unified Processes) som tar upp ämnet kunskapsöverbryggande eller syftar till att underlätta för kunskapsöverbryggande mellan de olika faserna. Samt att utreda vad som faktiskt görs hos de undersökta företaget för att adressera dessa eventuella problem.

1.3 Metoddiskussion

1.3.1 Datainsamlingsmetoder och tekniker

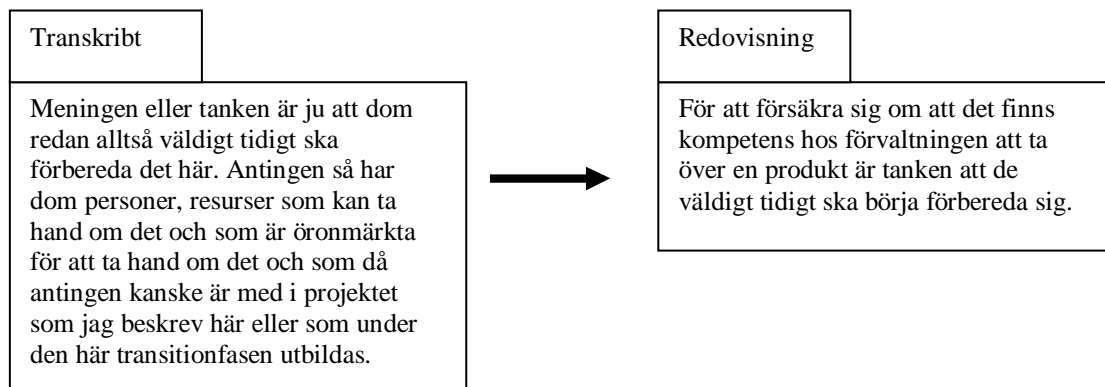
Jag valde att använda mig av studier av dokument samt intervjuer för att få en så korrekt bild som möjligt av mitt undersökningsområde. Målet var att skapa en helhetsbild av företagens användning av systemutvecklingsmetoden RUP. Ett alternativ till intervjuer, som eventuellt kunde ha varit bättre och gett mig ännu mer information hade varit att följa upp en produkt från utveckling till förvaltning, det vill säga med deltagande observation (Bryman, 1997), dock fanns det inte möjlighet till detta då perioden för magisteruppsatsen är tidsbegränsad till 10 veckors heltidsarbete. Efter att ha begrundat detta anser jag att intervjuer med personer som jobbar med RUP är den bästa metoden för att få den information som krävs för att göra undersökningen. Intervjuerna är huvudmetoden i min undersökning och det är från dem jag kommer att utgå i första hand.

Min studie är av kvalitativ karaktär, och merparten av materialet kommer ifrån ostrukturerade intervjuer. Intervjuer är ofta som guidade konversationer snarare än strukturerade frågor. Detta lämnar utrymme för intervjupersonerna att formulera sina svar själva. Jag har valt att använda mig av frågescheman i min undersökning, och inte klara teman från början (Bryman,

1997). I min undersökning har jag gjort intervjuer av just denna typ för att skapa en bild av hur de tänker i denna fråga. Jag ansåg att det trots att denna metod kan medföra att respondenterna kan komma ifrån ämnet var bättre än att styra respondenterna till svar. En fördel med denna metod kan vara att man kan hitta ytterligare sidoämnen som också är av intresse. En risk med den här formen av halvstrukturerade intervjuer som jag har använt mig av är att det kan bli för personligt, då jag som forskare uppmanar den intervjuade att vara så öppen som möjligt. Vilket, enligt Kvale (1997), kan leda till avslöjanden som kan leda till exempelvis etiska problem längre fram. Jag anser att valet att göra en kvalitativ undersökning med ostrukturerade intervjuer var det bästa alternativet för min undersökning eftersom det finns mycket som kan ligga runtomkring ämnet som är av intresse. Möjligheterna för respondenterna att komma med nya infallsvinklar och kommentarer kring mina frågor var också av intresse.

1.3.2 Dataanalysmetoder och tekniker

Intervjuerna spelades in på en bandspelare och har i efterhand transkriberats. Transkriberingen har skett medan intervjun i fråga varit i färskt minne. Transkribten har sedan ”städats”, det vill säga, att de har transkriberats ordagrant så långt det varit möjligt. Men utan ljud som är vanliga i talspråk, dessutom är avsnitt och vissa ordval som skulle kunna leda till identifikation av respondent och/eller företag borttagna. Det finns två skilda anledningar till att jag valt att avpersonifiera respondenterna, den första är att jag ansåg det irrelevant att skriva ut deras namn och företag, det bidrar inte i egentlig mening till studien, förutom för att ge den trovärdighet, vilket i och för sig är viktigt. Här kommer emellertid det andra skälet in och det är att min erfarenhet från kandidatuppsatsen var att det inte var bra, det blev i samband med presentationen av den en del repressalier för de intervjuade. I Bilaga 1 återfinns huvudfrågorna för intervjuerna.



Figur 1.1 Denna bild ska visa på skillnaden mellan transkriberat material och det material som sedan redovisas i uppsatsen

I min undersökning har jag valt att använda mig av kategorisering då materialet från intervjuerna var likartat mellan de olika respondenterna samt så pass överskådligt att detta var möjligt. Den kategoriseringen har gjort att jag utifrån intervjumaterialet kan skapa en sammanhängande text av det material jag får från intervjuerna. Denna form är den som jag anser är bäst för att presentera materialet på ett sätt som är överskådligt och enkelt för läsaren att ta till sig. Kategoriseringen innebär dock att det kan bli svårt att presentera allt det material

som uppkommer i intervjuerna då dessa är ostrukturerade och respondenterna får komma med fria svar. Kategorierna bygger på de frågor jag kommer att ta upp under samtliga intervjuer, se kapitel 1.3.5 för kategorierna.

1.3.3 Rapporteringsform och källkritik

Målgruppen för denna rapport är främst verksamma inom systemutveckling och systemförvaltning samt de som har ansvaret på företag för att välja metoder för systemutveckling.

Jag har valt att använda linjär-analytisk struktur vilket också är den vanligaste rapporteringsformen. Detta innebär att rapporten inleds med ett bakgrunds och metodkapitel vartefter relevant litteratur presenteras. På detta följer den empiriska undersökningen och det hela slutar i en sammanfattning och slutsatser av den gjorda undersökningen. Jag valde denna struktur då jag anser att den passar målgruppen eftersom den är allmänt välkänd och majoriteten av forskare och akademiska kollegor känner till den. Jag anser att detta medför att rapporten blir lättläst i den mån att läsaren vet vad som väntar och även kan relatera rapporten till andra rapporter.

Undersökningsfrågorna som används i intervjuerna baseras på litteraturstudien, den kritik som jag vill rikta mot den litteratur som jag har använt är att den inte tar upp kritik mot RUP som utvecklingsmodell. En anledning till detta tror jag kan vara att det material som jag utgått ifrån till stor del inte går in på de delar av RUP som jag undersöker och delvis endast beskriver den övergripande strukturen för modellen. En ytterligare anledning kan vara att materialet kommer ifrån utvecklarna av RUP eller IBM, eller har granskats av dessa.

Vid val av företag såg jag till om de använde den metod som jag undersökte, jag sökte dels efter företag som själva använde RUP, dels efter företag som utbildade inom RUP. Vid val av respondenter utgick jag ifrån deras befattningar inom företaget, de skulle ha kunskap om RUP och ha arbetat med det en tid, utifrån detta fick sedan företagen själva rekommendera de personer de ansåg vara lämpliga. I samband med intervjun på Gamma specificerade jag urvalet av respondent till att gälla en person som även utbildar inom RUP. Jag har valt att göra företag och respondenter i min undersökning konfidentiella, jag har avpersonifierat transkribten och de citat som återfinns i empirikapitlet så pass att de inte kan kopplas till en viss person, endast till transkribten. Detta val har jag gjort med förhoppningen att om någon av respondenterna sagt något som kan visa sig skadligt för dem själva ska individen lida så lite skada som möjligt (Kvale, 1997).

1.3.4 Om respondenterna och företagen

I tabellen nedan förklaras respondenternas namn samt deras koppling till de olika företag jag har varit på. De namn som används är fiktiva för att hålla företag och respondenter konfidentiella.

Intervjuerna skedde på respondenternas respektive arbetsplats och jag hade ringt eller haft kontakt via e-post med dem tidigare. Alla intervjuer spelades in på band för att senare kunna transkriberas. Under intervjuerna gjorde jag även minnesanteckningar om jag ansåg att de sa något som jag skulle fråga om senare.

Tabell 1.1. Tabellen ska visa sambandet mellan de intervjuade och de företag på vilka de arbetar.

Namn	Företag	Befattning
Arne	Alfa	Projektledare
Adam	Alfa	Projektledare
Anders	Alfa	Processansvarig
Bengt	Beta	Projektledare
Göran	Gamma	Projektledare

Rational Unified Processes (RUP) används numera av företaget Alfa i alla systemutvecklingsprojekt. Respondent Arne och Adam har båda varit projektledare för projekt som använt sig av RUP som utvecklingsmetodik. För att stödja RUP när det kommer till projektledning använder sig Alfa även av Praktisk Projektstyrning (PPS). Alfa har integrerat RUP och PPS för att få fram ett ramverk som fungerar att arbeta efter. Anders är processansvarig för Alfors interna systemutvecklingsprocesser, vilket innebär att han har ansvar för utvecklingsprocesserna och utvecklingsmiljön samt de verktyg och processer som utvecklarna använder, dessa utvecklare arbetar inom IT-delen av Alfa. Tidigare arbetade Anders som projektledare innan han övergick till processarbete. Han arbetar även i en referensgrupp som projekten kan anlita om de behöver hjälp med processerna, denna referensgrupp erbjuder bland annat utbildningar och supportaktiviteter. Beta och Gamma är båda konsultföretag och båda företagen har Alfa som kund. Bengt har tidigare arbetat på Alfa och Göran är just nu projektledare hos Alfa. Göran är dessutom certifierad utbildare inom RUP och har varit med och utbildat personal hos Alfa, den utbildning Göran håller behandlar RUP på ett generellt plan.

1.3.5 Val av intervjufrågor

Val av intervjufrågor baseras i första hand på uppsatsens huvudfråga, i andra hand till att knyta samman denna med den litteraturstudie som gjorts. Intervjuerna har utgångspunkt i hur arbetet med RUP sker i verkligheten, hur används metoden på det aktuella företaget och vad detta får för konsekvenser, både positiva och negativa. Då respondenterna även kan ses sitta på en viss kunskap och erfarenhet inom området systemutveckling handlar frågorna även om vad som saknas, vad de använder som metoden saknar. Frågorna är likartade intervjuerna emellan men ska mer ses som tematiseringar eftersom de frångåtts om andra funderingar kommit upp under intervjuens gång. Fokus i intervjuerna ligger hela tiden på kunskapsöverföring inom RUP. De teman som jag utgått ifrån är:

- Allmänt om RUP och Alfa
- Alfa och artefakterna
- Mognad och RUP
- Kommunikation
- Kompetens i förvaltningen
- Kunskapsöverföring i RUP
- Checklistor

Kategorierna utgår ifrån det som svarar till mitt syfte i första hand och i andra hand att skapa en förståelse och bakgrund till området och användningen av RUP hos Alfa.

1.4 Sammanfattning av inledning

Denna studie har som syfte att undersöka vad systemutvecklingsmetoder har för inbyggd förmåga att fånga och överföra kunskap. Utgångspunkten ligger i Backlunds (2004) mening att då kunskap är kopplat till dess kontext är det svårt att överföra den mellan organisationer, tyst kunskap är även något som innehåses av individer (Meggerle & Steen, 2002). Avgränsningen i uppsatsen är att endast undersöka systemutvecklingsmetoden RUP (Rational Unified Processes), valet föll på att det är den idag största och mest omfattande systemutvecklingsmetoden. Huvudfrågan för studien är;

Hur kan användandet av systemutvecklingsmetoden RUP hjälpa till med överföring av kunskap vid övergången från systemutvecklingsprojekt till förvaltningsfas?

Uppsatsen baseras främst på intervjuer och studier av dokumentation. Frågorna som tas upp i intervjuerna är baserade på de litteraturstudier som gjorts. Intervjuerna är open-ended, det vill säga en form av guidade konversationer. Dessa intervjuer har sedan transkriberats och utifrån transkribten har jag gjort en narrativ struktur baserat på olika teman.

Företag och respondenter har valts med utgångspunkt i deras kunskap och erfarenheter av den valda systemutvecklingsmetoden. Respondenterna och de företag där de arbetar är i uppsatsen konfidentiella. Intervjufrågorna baseras främst på huvudfrågan samt hur arbetet med RUP sker i verkligheten och vad användandet av RUP ger för konsekvenser.

2 Litteraturförankring

I följande kapitel presenteras studiens teoretiska referensram. Kapitlet syftar till att ge läsaren en bakgrund till problemområdet samt en inblick i tidigare forskning och relevanta teorier kring det problemområde som den empiriska studien grundas på. Detta material kommer även att ställas mot undersökningens resultat i den avslutande diskussionen. Avsnitt 2.1 syftar endast till att ge läsaren en bakgrund till de begrepp som studien baseras på. Resterande avsnitt ska ge djupare förståelse för RUP och PPS.

2.1 Teoretisk bakgrund

Systemutveckling syftar enligt Wiktorin (2003) på att utveckla datorstöd för informationshantering i en verksamhet. Detta sker genom tre huvuduppgifter som är; Verksamhetsanalys, Informationsbehovsanalys samt Anskaffning och Utformning. Metoder för att göra detta tillkommer sedan för att de som arbetar med utvecklingen ska ha ett gemensamt arbetssätt. Wiktorin (2003) skiljer mellan metoder och modeller enligt följande; en metod täcker sällan alla faser i utvecklingsmodellen. Vanligtvis är metoden specialiserad, eller fokuserad på ett eller två steg i modellen. Wiktorin nämner även metodkedjor som är sammansatta metoder som bildar en större del av helheten, således utvecklingsmodellen. Wiktorin (2003) påpekar att gränserna mellan vad som är metoder, processer, utvecklingsmodeller och projektmodeller håller på att bli än mer diffusa. Detta beror på att man har gått ifrån att beskriva hur man ska nå ett resultat till att beskriva hur de olika faserna i utvecklingsmodellen ska samverka.

Systemutveckling sker oftast och med fördel i projektform, enligt Wiktorin (2003), större system delas dessutom ofta upp i mindre projekt med delleranser. Projektformen lämpar sig bra eftersom det är en väl definierad uppgift som ska levereras inom en viss tid och har väl definierade resurser. Andersen (1994) menar att projektarbetsformen allmänt används i nya uppgifter som dels är engångsuppgifter, dels kräver medverkan av personer med olika kompetenser och olika ställningar inom verksamheten. Tanken är att då det inte inom den närmaste framtiden kommer att utvecklas något liknande system så kan det inte upprättas en särskild funktion för denna typ arbetsuppgift.

En mycket vanlig utvecklingsmodell är livscykelmodellen som med sina sju faser beskriver ett systems livslängd i sin helhet. Dessa faser är enligt Andersen (1994); Förändringsanalys, Analys, Utformning, Realisering, Implementering, Förvaltning och Drift samt Avveckling. Av dessa menar han att Analys, Utformning, Realisering och Implementering tillhör systemutvecklingen. Analysen i sig består enligt Andersen (1994) av två olika delar; Verksamhetsanalys och Informationssystemanalys. Dessa två analyser ska tillsammans ge en kravspecifikation, och det är denna som sedan binder samman analysfasen med nästa fas; Utformningen. Wiktorin (2003) menar att objektorientering numera är det dominerande arbetssättet vid nyutveckling. Objektorienterad analys omfattar att ta fram alla olika externa aktörer och för dessa identifiera de olika dialoger dessa har med systemet. I senare steg i

analysen omformas denna modell av användningsfall till en mer abstrakt analysmodell. I Andersens (1994) livscykelmodell består även utformningsfasen av två olika delar. I denna fas väljs de tekniska lösningarna som ska användas för att bygga upp det nya informationssystemet. Detta är enligt Andersen (1994) allt ifrån hårdvara till vad man ska använda för filstruktur och utvecklingsverktyg. Realiseringen innebär i systemutveckling oftast programmering bland annat, men även utformning av rutiner för användarna. Den sista fasen som enligt Andersen (1994) är en del av systemutvecklingen är Implementering vilket innebär införande och driftsättning av det nya systemet.

Wiktorin (2003, s.20) definierar en metod enligt följande; ”En metod är en beskrivning av hur man steg för steg löser en uppgift”. För varje arbetssteg anger metoden hur man ska gå till väga. Enligt Wiktorin (2003) är den numera dominerande metoden RUP (Rational Unified Processes), andra metoder han nämner är Daily Build, RAD (Rapid Application Development) och eXtreme Programming. Wiktorin (2003) menar att vid utveckling av system med krav på kort ledtid så ger inte storskaliga metoder som RUP ett bra utbyte. Det har därför uppstått en motreaktion till vad som kan uppfattas som överdriven tilltro till komplexa metoder och processmognad. Denna motrörelse har samlat sig under rubriken Agile Modeling och exempel på metoder inom denna genre är bland annat Daily Build och eXtreme Programming.

Ovan nämnda livscykel är exempel på den klassiska så kallade vattenfallsstrukturen där utvecklingen sker linjärt. Vattenfallsmodeller går från kravanalys via design till testning, vilket fungerar utmärkt förutsatt att kraven är klara från början och inga buggar uppkommer i samband med testningen. Hon menar att denna form av angreppssätt innebär att man skjuter riskerna framför sig och inte upptäcker dem förrän det är för sent. Alternativet kallas för iterativ utveckling och skillnaden är att momenten utförs i spiral, det vill säga en definierad sekvens av aktiviteter som inte avslutas förrän de uppsatta målen blivit uppnådda (Strand, 2001). Detta innebär alltså utveckling av mindre delar av helheten som testas och kan godkännas allt eftersom, så missförstånd kan bli tydliga tidigt i projektet, då det finns möjligheter att rätta till dem (Strand, 2001).

Efter det att systemutvecklingsprojektet är avslutat övergår produkten till att vara ett förvaltningsobjekt med vilket menas själva systemet samt den verksamhet som finns runtomkring. Det är viktigt att se till att det inte är endast de tekniska delarna, såsom applikationer och plattformar som ska förvaltas utan även processer, help-desk-funktioner med mera. Förvaltningsorganisationen ska hantera förbättring, felrättning, teknisk anpassning samt användarstöd (Nordström och Welander, 2002). Eftersom så mycket som två tredjedelar av ett företags kostnader för IT-system går till underhåll och vidareutveckling krävs det att denna fas är effektiv. Även i förvaltningsfasen måste arbetet ske affärsmässigt, med målstyrning och tydliga roller (Wiktorin, 2003). Mycket närliggande till förvaltningen ligger ju självklart driften (Nordström och Welander, 2002), de är ju även begrepp under samma rubrik i Andersens (1994) livscykel. Förvaltning och drift är funktioner som ständigt påverkar varandra, ibland är funktionerna till och med så närliggande att det kan vara svårt att urskilja om det är en driftaktivitet eller förvaltningsaktivitet. Då aktiviteterna är så närliggande kan vissa delar hamna mellan stolarna när, som på senare tid företag valt att dela på förvaltning och driftsfunktionerna. Numera är det dock många som ser sig om efter en helhetslösning som tar hand om både förvaltning och drift (Nordström och Welander, 2002).

Att avsluta ett projekt och övergå till förvaltning innebär en överlämning, detta är egen fas och det beskrivs i RUP som en egen fas. Överlämning betyder en typisk gradvis övergång av produkten från utvecklarna till kunden. Fasens komplexitet är helt beroende av hur kritisk

produkten är i sitt användningsområde (Lunell, 2003). Huvuddelen av denna fas handlar om att utvärdera och anpassa den nya produkten samt att förbättra prestanda och icke-funktionella egenskaper. Vid sidan om detta sker även utbildning av användare, manualer ska fullbordas och det nya systemet ska integreras med det gamla (om det finns ett sådant) (Lunell, 2003). I överlämningskedet anser jag det även viktigt att de som ska ta emot produkten, det vill säga i detta fall, förvaltningsorganisationen, har kunskap om den produkt de ska förvalta. Kunskap, såsom jag använder det, syftar till kunskap som bygger på erfarenheter hos individen.

”Slutsatsen vi drar är att kunskapen och kunnandet (...) behöver medvetandegöras och synliggöras för att kunna utvecklas. Eftersom den praktiska kunskapen (...) visar sig i skickligt utförda handlingar som i sin tur vilar på egna erfarenheter i kombination med andras förmedlade erfarenheter, är erfarenhetsutbytet centralt.” (Meggerle & Steen, 2002, s. 88)

Här talar författarna om erfarenhetsutbyte, vilket jag benämner som kunskapsöverföring. Jag menar att för att förvaltningsorganisationen ska kunna göra ett bra jobb efter överlämning krävs att de har kunskap om den produkt de ska förvalta. Begreppet kunskap är vidlyftigt men Rolf (1981) menar att det väsentliga med begreppet kunskap är att känneteckna personer, organisationer eller professioner som vanligen har kunskap inom ett visst fack. Med detta menar han att kunskapen som kommer från tidigare erfarenheter kan användas i framtida situationer. Kunskap är enligt Rolf (1981) en stabil egenskap som är förankrad i stabila kunskapsobjekt. Han menar att en så kallad expert inom ett fack på grundval av sina tidigare erfarenheter eller kunskaper tolkar den nya informationen han får och kategoriserar och inordnar denna i sitt eget system för att utifrån detta avgöra om den nya kunskapen är korrekt eller inte. Även Meggerle och Steen (2002) håller delvis med om detta och menar att det finns olika sorters kunskap, de tar upp tyst kunskap som de definierar som bakgrundskunskap till något vi fokuserar på. De menar att denna kunskap är personlig eftersom den integreras med vårt arv, alltså våra personliga traditioner, kultur, värderingar med mera. Meggerle och Steen (2002) menar att praktisk kunskap bygger på tyst och personlig kunskap vilken används till att utföra olika handlingar. Att kunskap är tyst, eller något som individen bär på innebär att det kan vara svårt att förklara den, att lära ut den till andra individer. Utifrån detta tolkar jag att det är svårt att överföra kunskap mellan individer då den är bunden till erfarenheter (Rolf, 1981) och till tyst kunskap (Meggerle & Steen, 2002), vilket ger att för att individerna i förvaltningsorganisationen ska kunna ”få” kunskap om en produkt kan den inte bara lämnas över, de måste på något sätt tillgodose sig kunskapen, eventuellt redan innan överlämnandet av produkten. I motsats till tyst kunskap beskriver Meggerle & Steen (2002) något som de benämner som påståendekunskap, denna kunskap är enklare att beskriva än den tysta kunskapen och exempel på den är; regler, metoder, normer och teorier.

2.2 Rational Unified Processes

Detta kapitel syftar till att ge bakgrunden till val av undersökningsobjektet, Rational Unified Processes (RUP) samt att ge läsaren insikt i RUP, dess olika faser och artefakter vars användningsområde är det som senare undersöks i den empiriska studien. Samt även att göra läsaren införstådd med RUPs möjligheter att ta tillvara och överföra kunskap.

2.2.1 Bakgrund

RUP är enligt Wiktorin (2003) den vanligaste nu förekommande systemutvecklingsmetoden för objektorienterad systemutveckling. RUP är enligt Wiktorin mycket mer än bara en metod då den beskriver hur man fördelar arbetet över de olika faserna genom användning av inkrement och iterationer. Fördelarna med denna iterativa utveckling är enligt Strand (2001) flera, bland annat så kan allvarliga missförstånd bli tydliga redan tidigt i livscykeln och man ges tidigt en möjlighet att rätta till dem. Det uppmuntrar även utbyte med slutanvändarna och faktiska krav kan identifieras allt eftersom, dessutom får projektets kravställare löpande och konkreta bevis på projektets status.

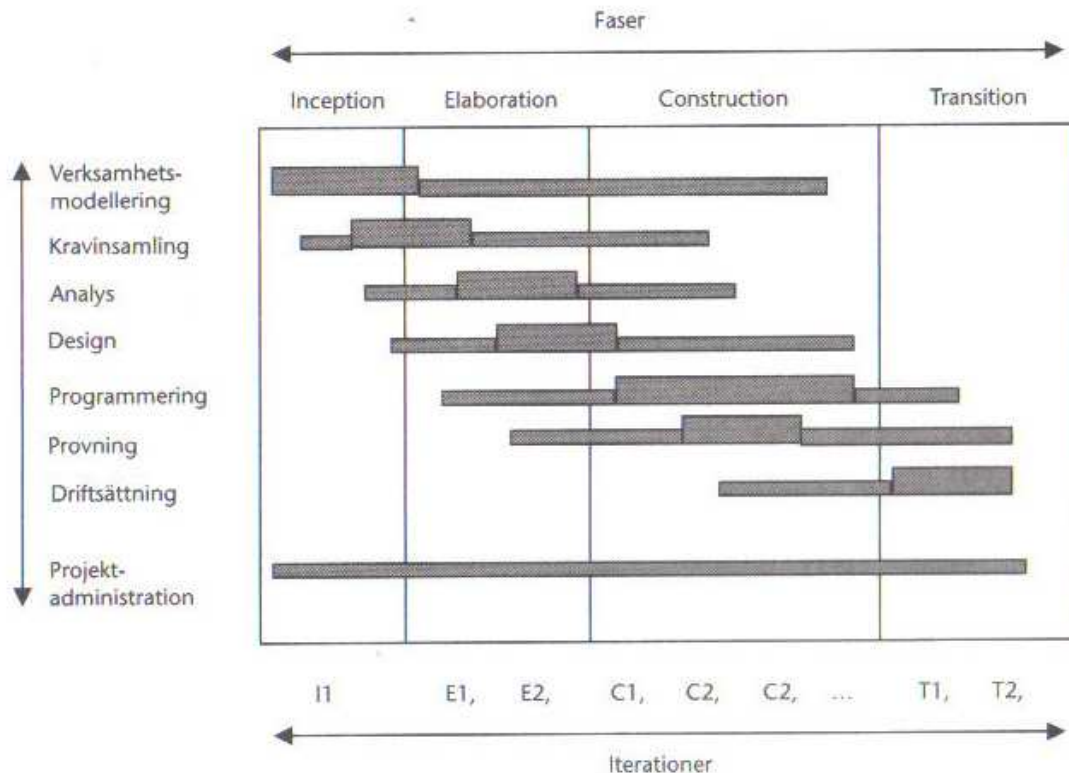
Då RUP är väldigt omfattande är det dock sällan man använder den i sin helhet utan istället använder man tekniken med användningsfall med lämpliga tillägg. Strand (2001, Förord) skriver att; ”RUP är stort, omfattande och en aning tungfotat men innehåller en rik samling av kunskap och erfarenheter som alla som arbetar med mjukvaruutveckling bör bekanta sig med”. Strand (2001) och Lunell (2003) menar att RUP är ett ramverk för mjukvaruutveckling som man kan anpassa i det oändliga för att passa ett visst projekt. Strand (2001) menar att det är upp till de ansvariga för processerna inom RUP att sätta ihop en process som passar i ett visst sammanhang. Hon påpekar att det kanske inte ens är möjligt att använda RUP likadant i alla projekt inom ett och samma företag utan att det även här kan behövas förändringar och anpassningar från projekt till projekt. En av kärnpunkterna i RUP, enligt Wiktorin (2003), är att beskriva ett system utifrån hur det används. Beskrivningen utgår från vilka användare som finns, så kallade aktörer och sedan beskrivs hur dessa agerar, så kallade användningsfall, på engelska Use Case. Wiktorin (2003) och Strand (2001) menar att dessa användningsfall fått så stor genomslagskraft i objektorienterad systemutveckling därför att de är ett mycket bra hjälpmedel för att kommunicera funktionskrav mellan användare och systemutvecklare. Användningsfall är även till stor nytta i testprocessen, de ligger till grund för identifiering av testfall och testprocedurer (Strand, 2001). Dessutom fungerar användningsfallen som en slags bro mellan krav och designaktiviteter.

2.2.2 Faser inom RUP

RUP innehåller enligt Wiktorin (2003) dels metदानvisningar för enskilda aktiviteter samt rekommendationer för hur dessa ska arrangeras i livscykelmodellens olika faser. Arbetet inom RUP löper genom ett antal olika faser som var och en betonar de olika stegen i systemutformningen, allt från tidig analys till driftsättning. RUP skiljer sig enligt Wiktorin (2003) från tidigare objektorienterade metoder på det sättet att den omfattar hela kedjan från den inledande idéfasen, via analys och design till konstruktion, provning och driftsättning. Strand (2001) menar att en fas är en tidsperiod mellan två milstolpar där formulerade och

definierade mål är uppsatta, utifrån detta kan man sedan när milstolpen kommer avgöra om man kan gå vidare till nästa fas.

Precis som i Andersens (1994) livscykelmodell finns det i RUP dels aktiviteter, i RUP kallat discipliner (Lunell, 2003) som ska gå igenom samtidigt som det finns faser. Wiktorin (2003) beskriver aktiviteterna som Verksamhetsmodellering, Kravinsamling, Analys, Design, Programmering, Provning, Driftsättning och Projektadministration. Samtidigt som dessa finns det fyra faser i RUP, dessa är; Inception, Elaboration, Construction och Transition.



Figur 2.1: Sambandet mellan aktiviteter, faser och iterationer i RUP (Wiktorin, 2003, s.95)

De två första faserna, Inception och Elaboration kan enligt Strand (2001) ses som de skapande faserna, det är då arkitekturen fastställs och alla krav samlas in. De senare två faserna, Construction och Transition är de byggande faserna, det är här programmerandet, testandet och överlämnandet till slutanvändarna sker. Faserna utförs i varierande grad inom samtliga av aktiviteterna som en del av dessa. Faserna genomförs i iterationer, det vill säga att var fas genomförs en eller flera gånger och varje fas producerar ett inkrement som förfinar och utvidgar resultatet i föregående iteration (Wiktorin, 2003). En iteration består enligt Strand (2001) av en fullständig utvecklingscykel som ska resultera i en körbar release, om än aldrig så liten. Principen handlar om bredden först och därefter kommer djupet allt eftersom kraven kommer fram i processen. Varje iteration i sig själv är ett vattenfall där aktiviteterna genomförs i logisk ordning och allteftersom får man en bättre uppfattning om vad systemet skall innehålla och kan anpassa sitt arbetssätt därefter. Metoden kan innebära att man måste gå tillbaka och göra om analyser och ändra i tidigare delar av analysen (Wiktorin, 2003).

Inception betyder start (Strand, 2001) eller förberedelse (Lunell, 2003). Det är i denna fas som man lägger grunden till sitt projekt, här skapas och formuleras projektets syfte och mål, man gör riskbedömningar, uppskattar vilka resurser som kommer krävas, skapar en projektplan

och sätter upp milstolpar (Strand, 2001). Huvudsyftet med denna fas är enligt Lunell (2003) att alla parter som är involverade i projektet ska enas om en gemensam uppfattning om vad slutprodukten ska innefatta. Strand (2001) menar att ungefär hälften av alla användningsfall bör identifieras i Inception-fasen men de behöver inte formuleras eller beskrivas så noga. Hon menar även att man bör bygga en körbar prototyp redan i denna fas så att beställare och kravställare tidigt kan avgöra om projektet är på rätt spår.

Elaboration är nästa fas, i denna ska man arbeta vidare på detaljerna Strand (2001) och Lunell (2003) menar att det är i denna fas som problemområdet, domänen, systemets design och arkitektur etableras samt projektplanen vidareutvecklas. Fasen innefattar att skapa en fastare grund för projektet (Lunell, 2003). Efter denna fas där man gått på djupet i att fånga upp alla krav och formulerat dessa avgörs huruvida man ska börja bygga systemet på allvar (Strand, 2001). Lunell (2003) menar att detta är den mest kritiska av de fyra faserna och att felaktiga beslut i denna fas kan få långtgående konsekvenser för projektet, detta eftersom det är i denna fas som projektet övergår från ett småskaligt till ett fullskaligt projekt.

Construction betyder helt enkelt tillverkning, det är i denna fas som produkten konstrueras, byggs. Strand (2001) menar att det är under denna fas som produkten iterativt utvecklas till att bli ett färdigt system. Det är en utvecklingscykel av testande och kodande och det är även här acceptanskraven från mottagarsidan formuleras. Lunell (2003) menar att denna fas går ut på att infoga nya komponenter eller komplettera gamla i en redan existerande och utprovad struktur. I slutet av denna fas fattas det formella beslutet om mjukvara och användare är redo att möta verkligheten (Strand, 2001).

Transition är den sista fasen i RUP och den syftar till överlämning eller övergång. Det är i denna fas som produkten överlämnas till slutanvändarna. Strand (2001) menar att denna fas oftast innebär att slutanvändarna upptäcker än fler krav och att ytterligare utveckling sker. Lunell (2003) skriver att fasen går ut på att "täppa till alla hål", med vilket han menar att det efter fasens slut ska vara klart för användning av systemet. I denna fas sker även utvärderingar av utvecklingsprocessen för att man ska kunna ta med sig erfarenheterna från projektet in i framtida projekt (Strand, 2001). I denna fas löper utvecklingen parallellt, samtidigt som man gör de sista finjusteringarna sker även utbildning av användarna, tester och överföring av data från t.ex. gamla databaser (Lunell, 2003).

Eftersom Transition är den fas där produkten överlämnas och med den även artefakterna behöver denna fas beskrivas ytterligare. Redan när projektet går in i denna fas anses systemet ha en operativ förmåga och det ska vara godkänt för överlämning och installation av projektledare och utvärderingsgrupp (Strand, 2001). I fasen rättas de sista problemen till och mottagarsidan gör acceptanstester. Strand (2001) menar att de aktiviteter som pågår i Transition-fasen är;

- Förbereda beta- och acceptanstester
- Installera (eller förbereda installation av) systemet i användarmiljön, inklusive aktiviteter som att konvertera och migrera data från föregående system
- Hantera testresultaten
- Färdigställa samtliga projektartefakter
- Avsluta projektet

Fasen, och i med den projektet, kan inte avslutas förrän alla artefakter är kompletta. Detta innebär att samtliga modeller, beskrivningen av arkitekturen och så vidare. Detta är viktigt för vidareutveckling och förvaltning av produkten ska gå så smidigt som möjligt men även för att underlätta för framtida projekt (Strand, 2001).

2.2.3 Roller och artefakter inom RUP

I RUP finns det en hel mängd roller som var och en har ansvar för ett visst område, flera av dessa roller kan innehas av samma individ eller vara uppdelade på flera individer, beroende på projektets storlek och andra faktorer (Strand, 2001). I bilaga 6 visas de roller som finns i RUP enligt Lunell (2003). Wiktorin (2003) menar att det finns ett par viktiga roller som bör identifieras och/eller bemannas inom projekt, dessa är; sponsor/beställare, projektledare, utvecklare, användare, modelleringsledare och mentor. Rollen utvecklare består av bland annat programmerare, arkitekt och domänanalytiker. Även Wiktorin (2003) skriver att inom små projekt kan samma individ inneha flera roller.

”En artefakt är något som skapats, ändrats eller använts i processen.” (Strand, 2001, s. 180). Artefakter syftar till att alla de delar som i slutändan leder fram till den färdiga produkten. Många invändningar mot RUP handlar om att det just läggs för mycket tid och kraft på att ta fram alla dessa artefakter. Men Strand (2001) menar att metoden inte är till för att uppmuntra skapandet av en massa pappersdokument utan att uppmuntra deltagarna i projektet till att fokusera på att ta fram UML-modeller, kod mm. Bilaga 7 visar sambandet mellan de olika artefakterna och aktiviteterna i projektet såsom de är i RUP (Lunell, 2003). Jag tolkar detta som att artefakterna är till för att överföra kunskap i viss mån, dock är det endast rekommendationer för att överföra kunskap som inte är tyst, det vill säga explicit kunskap. RUP bygger enligt Strand (2001) på något som kallas för ”Best Practices”. Detta syftar till att det är samlad erfarenhet inom mjukvaruindustrin som står bakom de rekommendationer som RUP står för.

2.2.4 RUPs dokument och artefakter i förhållande till kunskap

Artefakterna är enligt Lunell (2003) till för att visa hur produkten växer fram. Inom RUP omfattar artefakter flera olika saker, från pappersdokument, modeller, modellement, källkodsfiler, testskript till byggspecifikationer och komponenter (Lunell, 2003). De riktlinjer som finns för artefakter är av tre slag;

- Riktlinjer för dokument med råd och anvisningar om hur dokumentet skapas, aspekter att tänka på, metoder som rekommenderas, och dylikt.
- Modelleringsriktlinjer med anvisningar om hur modeller och modellement ska organiseras, namnges etc.
- Programmeringsriktlinjer som ger regler för hur källkod ska ställas upp, variabler namnges etc.

Utifrån ett kunskapsperspektiv anser jag att vissa av artefakterna ses som grund för påståendekunskap, de kan användas för framtida metoder och annat (Meggerle & Steen, 2002). Artefakternas eller RUP som metod och dess förmåga att förmedla erfarenheter är det som är uppsatsens syfte att utreda. Är detta möjligt? Att överföra kunskap och erfarenheter genom dokument är inte helt självklart. Men RUP som metod med de

”Best Practices” som Strand (2001) beskriver skulle kunna innehålla rekommendationer för hur även detta skulle kunna gå till.

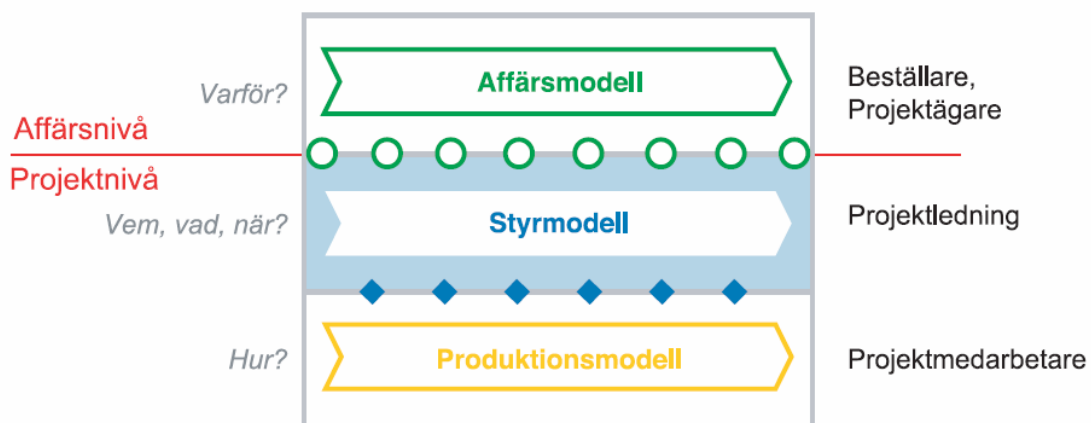
2.2.5 Kritik mot RUP

Henderson-Sellers et al (2000) framför kritik mot RUP på vissa punkter, för det första menar de att RUP endast stödjer projektledning i en separat del av modellen och att detta görs ofullständigt, se nederst i Figur 2.1. Författarna menar att processen är bristfällig när det gäller områdena; kunskap, anskaffningshantering, kommunikationshantering och även personalhantering. De menar att för att modellen ska kunna fungera i sig själv, eller självständigt utan stödjande modeller krävs att man utvidgar just dessa områden av modellen. Wiktorin (2003) tar också upp något som jag uppfattar som kritik när han nämner att vid mindre projekt med krav på kort ledtid så ger inte RUP heller ett bra utbyte. Detta kan bero på det som Strand (2001) skriver, att RUP är stort, omfattande och en aning tungfotat.

2.3 Kort om PPS

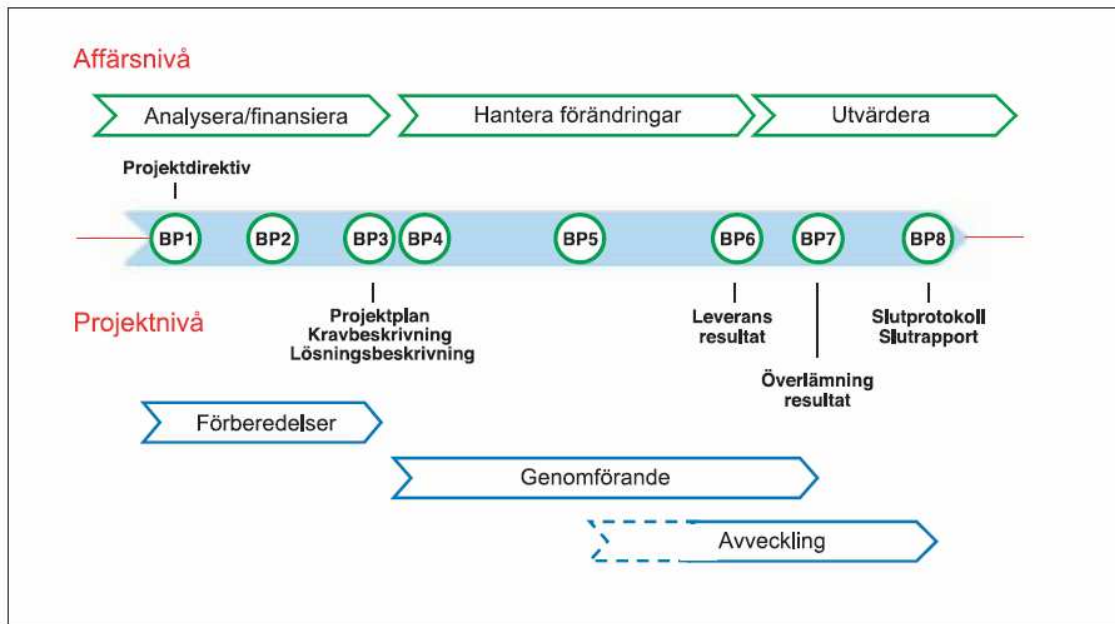
Praktisk Projektstyrning (PPS) är en modell framtagen av TietoEnator för styrning av projekt, för att man aktivt ska kunna leda, styra och följa upp projekt. Syftet med modellen är helt enkelt att skapa fler lyckade projekt. Modellen grundar sig på praktiska erfarenheter från tidigare lyckade projekt samt standarder som finns inom området (TietoEnator, 2005-05-12).

Modellen nedan ska visa sambanden mellan de modeller ett visst företag använder där PPS är delen i mitten (TietoEnator, 2005-05-12). I stora organisationer och koncerner är det vanligt att man definierat en överordnad process som är gemensam för projekt. För att modellerna ska fungera tillsammans är att underordnade modeller får arbeta inom ramarna för den överordnade processen. En del av problemet ligger i hur man ska sammanfoga de milstolpar som finns inom de olika modellerna, som i detta fall, PPS milstolpar och RUPs milstolpar (Lunell, 2003).



Figur 2.2 Visar sambanden mellan de olika modellerna som används i projekt (TietoEnator, 2005-05-12)

PPS skiljer på styrning inom projekt och produktion, detta gör att modellen kan appliceras på alla olika typer av projekt oavsett vilken bransch den används inom. Modellen kan även anpassas efter projektens storlek på ett enkelt sätt. Modellen nedan ska visa på de milstolpar som finns i PPS (TietoEnator, 2005-05-12).



Figur 2.3 Figuren ska visa på hur beslutspunkterna i PPS kopplas till de olika faserna i projektet (TietoEnator, 2005-05-12).

TietoEnator (2005-05-12) menar att bilden ovan inte ska tolkas som att projekten alltid arbetar sekventiellt utan den ska visa på den logiska indelningen av beslutspunkterna i de olika faserna av projektet. Man menar även att punkterna behöver kopplas samman med utvecklingsmodellen på ett lämpligt sätt för varje projekt.

Inom PPS finns tre faser; Förberedelse, Genomförande och Avveckling. Fasen förberedelse innebär att man definierar vad projektet ska åstadkomma för resultat, samt vilka avgränsningar som ska finnas, i denna fas bestäms även hur projektet ska genomföras rent praktiskt. Här beslutas om organisationsstruktur, produktionssätt och kvalitetssäkring bland annat. Fasen genomförande innebär uppfyllande av projektets mål, inom denna fas har man även återkoppling för att se att projektet håller sig inom ramarna för vad som var tänkt. Genomförande fasen handlar om att nå de mål som satts upp för projektet. Avveckling handlar om att lämna tillbaka de resurser som använts i projektet. I denna fas summerar man även projektet i en slutrapport och de erfarenheter man gjort beskrivs för att kunna användas i framtida projekt (TietoEnator, 2005-05-12).

Inom PPS finns det en rad beslutspunkter som ska passeras för att projektet ska få fortsätta. För var och en av dessa punkter finns uppsatta beskrivningar och checklistor. Vid varje beslutspunkt träffas styrgruppen och beslut tas av dem för projektets fortsatta inriktning. Mellan beslutspunkterna pågår arbete för att styra projektet mot mål och för detta finns tre grundprocesser, Åtagande, Återkoppling och Ledning. Åtagande innebär att parterna inom projektet är överens om de förutsättningar som finns samt att man säkerställer att parterna har samma syn på vad resultatet ska bli. Återkoppling syftar till att man under projektets gång säkerställer att åtagandet stäms av och för att efter projektets slut kunna avgöra om arbetet

utförts korrekt. Ledning handlar om att ta fram alternativ och ta beslut som leder fram till resultat. För PPS har man även tagit fram dokumentmallar för att olika projekt ska få samma struktur när det kommer till dokumentation. Dessutom finns mallarna anpassade efter stora och små projekt (TietoEnator, 2005-05-12).

Även inom PPS finns ett antal definierade roller och upphovsmakarna anser att det är viktigt med struktur och tydlighet för rollerna för att underlätta samarbetet inuti och utanför projektorganisationen (TietoEnator, 2005-05-12).

2.4 Sammanfattning av litteratur förankringen

Första delen av den teoretiska förankringen syftar till att ge läsaren en bakgrund till området, det vill säga en bakgrund till de termer som används i undersökningen. Detta syftar främst till termerna systemutveckling, Andersens livscykelmodell, systemutvecklingsmetoder, förvaltning och kunskap. Kunskap är det i sammanhanget viktigaste begreppet eftersom det utgör en stor del av undersökningen. Kunskap i denna uppsats syftar till något som den enskilda individen bär på, något som utvecklas genom erfarenheter, detta gör att det är svårt att överföra kunskap till andra individer. Kunskapsöverföring är fokus för uppsatsen.

Delkapitel två ger en bakgrund till uppbyggnaden av faser, aktiviteter, artefakter och roller inom RUP. Aktiviteterna är de samma som i Andersens (1994) livscykelmodell, det vill säga; Förändringsanalys, Analys, Utformning, Realisering, Implementering, Förvaltning och Drift samt Avveckling. Faserna är Inception, Elaboration, Construction och Transition. Kapitlet tar upp samband mellan de olika faserna och disciplinerna samt även kopplingen till den traditionella livscykelmodellen. Det tar även i korthet upp kritik som finns mot RUP. Delkapitel tre ger en bakgrund till PPS som är den andra metoden som företaget, där intervjuerna gjorts, använder. RUP är en utvecklingsmodell och PPS är en modell för projektstyrning. Det fjärde delkapitlet syftar till att beskriva hur intervjufrågorna satts upp i förhållande till litteraturstudien.

3 Empirisk studie

I detta kapitel återges den data som framkommit vid intervjuerna.

3.1 Bakgrund

3.1.1 Allmänt om RUP och Alfa

Idag är RUP den utvecklingsmodell som alla projekt på Alfa ska använda, men den används inte fullt ut ännu, Adam menar att en del projektledare fortfarande jobbar på sitt eget sätt. Emellertid är RUP, enligt Adam, likt mycket annat, bara det att man använder andra benämningar och kanske fler artefakter.

Tidigare använde Alfa, enligt Arne, en traditionell vattenfallsmodell. Projektledarna anpassade den utifrån sig själva och sitt projekt, valet att använda sig av RUP istället kom ifrån att man kände att arbetet behövdes stramas upp och bli mer likformigt projekten emellan. Adam menar att valet föll på att RUP just nu är den största utvecklingsmodell som finns. Anders säger att den modell de använder nu till stor del består av RUP men att Alfa gjort anpassningar för att den ska passa företaget. För att stödja denna modell använder Alfa en rad verktyg, såsom Clear Case, Clear Quest med flera och för dessa verktyg har Anders ansvaret. Anders berättar även att processen som kommer att hänvisas till som Alfa IT Processes (Alfas egen version av RUP) inte passar i alla utvecklingsprojekt, exempelvis stödjer den inte datawarehousing. På Alfa har man integrerat de milstolpar som finns i RUP med den styrningsmodell som man använder, det vill säga PPS (Praktisk Projektstyrning). Justeringar har gjorts för att de båda modellerna ska passa ihop och passa Alfas ändamål, säger Adam. Adam menar även att RUP inte behandlar hur man som projektledare styr projektet utan att Alfa använder PPS till det, modellen är ett hjälpmedel för att styra projekt och för att projektledaren ska ha ryggen fri.

Arne menar att RUP ska användas som ett ramverk bestående av olika discipliner såsom business modellering, requirement, analyze, design och implementation, test med flera. Arne menar att man orienterar arbetet runt omkring dessa discipliner men kan göra det på olika sätt beroende på ett visst projekts förutsättningar. Arne säger:

”Jag försöker disciplinorientera det hela och försöker få så få personer som möjligt med i projektet, vilket är lite tufft idag eftersom vi går mot en ökad specialisering.”

För Arne är det självklart också en stor fördel om medarbetarna har en grundträning i sina respektive roller, men eftersom RUP för Alfa är relativt nytt är det svårt och han säger:

”Det innebär ju också att vi får en hel del diskussioner och en del störningar för vi vet inte riktigt hur vi ska arbeta. Alla arbetsprocesser sitter inte som dom ska. Men det känner jag att vi tar steg för steg”.

3.1.2 Alfa och artefakterna

Alfa har tagit fram en baslista med innehållande artefakter, utifrån denna avgör sedan det enskilda projektet vilka artefakter man ska ta fram och vem som är ansvarig för dem. Bland dessa artefakter finns även sådana som är obligatoriska att ta fram i varje projekt. För att projektledarna ska veta vad de ska ta fram så finns det på Alfa checklistor som man kan gå efter om man vill. Dessa checklistor utgår dock inte från RUP, menar Anders som har varit med och tagit fram dem. Adam menar att projekten dokumenterar utifrån sina minimibehov, en lista med krav på vad varje projekt måste ta fram. Dessa listor specificerar, enligt Anders, vad projektet ska ha tagit fram till ett visst tillfälle, eller möte. Denna lista finns till för andra intressenter utanför projektet och för att det finns delar i den som måste följa med produkten i dess livscykel, detta är produktartefakter. Dessutom finns projektartefakter, vilka kan väljas friare vilka man ska göra dock finns det ju sådana som måste finnas. Anders säger att vad och hur mycket man dokumenterar beror på projektets eller produktens nytta i dess omgivning, mer dokumentation krävs för system som, om de skulle sluta fungera gör att det kostar mycket. Med detta menar han att om ett system exempelvis är en central del i deras affärssystem skulle gå ner och det kostar dem flera miljoner i timmen att inte kunna sköta transaktionerna, då krävs det mer dokumentation för detta system för att snabbt kunna åtgärda felet än vad det gör för ett system som inte är lika livsviktigt, det vill säga, inte kostar så mycket om det slutar fungera ett tag.

Artefakterna menar Arne är det som blir styrande och vägledande i projektet, och de skapar en trygghet för man vet vad man ska arbeta emot. Göran menar att ett problem inom RUP är att användarna lätt fokuserar för mycket på dokumenten, eller artefakterna. Ändamålet med RUP är att genomföra bra aktiviteter och sedan dokumentera dem men det är lätt att man fastnar i dokumentationen och dess strukturella sidor, som exempelvis hur man ska skriva, vilka mallar som finns och rubriker med mera. Inom varje disciplin finns sedan en eller ett par roller, personer som ansvarar för disciplinen och artefakterna i den. RUP beskriver hur man ska arbeta inom varje disciplin men Arne menar att som projektledare är denna beskrivning inte något Arne lägger så mycket kraft att titta på, Arne förutsätter att innehavaren av en viss roll vet vad som ingår i just den rollen. Göran anser att RUP innehåller många bra idéer om hur man hanterar sina krav och låter kraven styra arbetet och att det även är bra att RUP fokuserar på att man ska hantera de problematiska delarna av ett projekt tidigt istället för att, som man gärna gör, skjuta på dem till man måste göra dem. Bengt menar att RUP kan fungera som en mentor för oerfarna projektledare, att modellen kan användas när projektledaren är ny för att denne ska veta vad som behövs göras för att det ska bli ett bra projekt. RUP är uppbyggt på en massa Best Practices, det vill säga att när någon enligt utvecklat ett projekt som går bra skriver man ner erfarenheterna för att andra ska kunna ta del av dem och applicera dem på sina projekt. Detta har dock inneburit att modellen blivit väldigt stor, så det kan bli svårt att hitta det som stämmer bäst för ett visst projekt att ta efter. Bengt menar att då RUP är så pass omfattande är det viktigt att man noga väljer ut de delar som passar det enskilda projektet och anpassar metoden.

3.2 Mognad och RUP

Adam anser att det är en mognadsprocess som pågår just nu inom Alfa. Då RUP har så många artefakter och dokument att gå efter så blir det tungrott i början innan alla lärt sig hur det ska fungera och vad man kan begränsa sig till. Adam säger: "Det blir ju lite så att man tar det säkra före det osäkra och använder några dokument för mycket när man väljer ut vad man

behöver.” Arne berättar att Alfa just nu håller på att utveckla sina processer och att det blir projekten blir tyngre då man inom företaget inte har de överordnade processerna riktigt klara för sig. Anders menar att misslyckande i projekt inte bara beror på projektledaren eller processerna utan även på styrgruppen, han menar att styrgruppen inte alltid vet vad som förväntas av dem och detta ger projektet i sin tur problem.

Arne menar att eftersom RUP fortfarande är ganska nytt på Alfa så vållar övergången mellan olika moment problem men där är artefakterna något slags hjälpmedel. Dock menar Arne att man inte helt kan förlita sig på dokument och artefakter utan att kommunikationen i projektet också är viktig. Eftersom RUP fortfarande är relativt nytt på Alfa får många av projekten lite av en pilotkaraktär över sig och det blir en hel del diskussioner och störningar för att man inte riktigt vet hur man ska arbeta. Men steg för steg blir detta bättre och allt fler får utbildningar inom sina roller och även hur RUP fungerar som helhet. Arne menar att det är viktigt för medarbetarna i projektet att se hur deras egna discipliner påverkar de kringliggande och få en helhetsbild av vad deras del i ett projekt gör för helheten, detta för att göra överlämnandena av artefakter mindre neutrala.

Göran menar att det ofta är svårt att lära om folks tankesätt till hur RUP fungerar, RUP är ju iterativt till skillnad från de flesta andra systemutvecklingsmetoder. Och det är svårt för projektmedlemmarna att ta till sig detta, det är svårt att förstå att RUP är till för att man stegvis ska komma till insikt om hur/var projektet kommer att sluta, att det exempelvis från början är omöjligt att göra en trovärdig budget innan projektets riktlinjer ens är uppdragna.

3.3 Kommunikation

För att kommunicera inom projekten finns det dels de artefakter som man tillverkar som fungerar som bärare för information men Arne menar att dessa i sig själva inte är tillräckliga. Man har på Alfa strukturerade tillfällen då man inom projekt kommunicerar med varandra. Exempel på dessa är arbetsmöten inom arbetsgruppen och workshops. Till de mer formella mötesformerna hör möten mellan affärsdelen och IT-delen i ett projekt. Och möten inom dessa två delar, detta främst för att orientera respektive projektledare om vad som händer inom projektet. Utanför detta finns även möten med styrgruppen och om man jobbar i större projekt där vårt projekt är en liten del av en större helhet så är det ju även möten i det stora hela projektet för att kolla av att de små delarna är på rätt spår. Möten med styrgruppen är formella och de mötena dokumenteras. Bengt nämner att ett utav problemen med RUP handlar om just styrgrupper eller snarare avsaknad av styrgrupp i metoden, det finns inget i RUP som behandlar styrgrupper, samtidigt som det är väldigt viktigt för projekt att ha en styrgrupp. Bengt går till och med så långt att han säger att: ”har du ingen styrgrupp kan du lägga ner projektet”.

3.4 Kompetens i förvaltningen

För att försäkra sig om att det finns kompetens hos förvaltningen att ta över en produkt är tanken att de väldigt tidigt ska börja förbereda sig. Göran berättar att i projektet som han leder hos Alfa har man fört in ett antal personer från den framtida förvaltningsorganisationen i projektet, dessa ska vara deltagare i projektet i ett antal månader och arbeta med det konkret för att på så sätt lära sig produkten som de sedan ska förvalta. Han påpekar även att det är

viktigt i samband med att man fasar in individer på detta vis att de har en grundkompetens för det arbete de ska göra i projektet och att det är viktigt att de har rätt profil, för detta har man inom projektet tagit fram planer för de personer/kompetenser man söker. Hos Alfa jobbar de för att allt tidigare och tydligare se till att det finns en mottagande organisation redan tidigt i projektets livscykel. De jobbar för att det ska finnas en mottagare som har resurser och som är beredd på att ta produkten. Bengt menar att nackdelen med RUP är att modellen inte tar upp detta med styrgrupp och ägande av produkten. Han menar att om man ingen som äger produkten kan man likaväl lägga ner projektet, i RUP finns heller inget som beskriver hur styrgruppen ska arbeta. Anders säger att det är ett av kraven på Alfa för att starta ett projekt att det ska finnas en produktansvarig, det är den personen som sedan är ansvarig för produktens hela livslängd, inklusive förvaltning. Dock är det ju beroende av hur projektet ser ut hur mycket de behöver förbereda och vad de behöver göra för förvaltningsorganisationen för att överlämningen ska fungera. Arne menar att varje projekt har unika förutsättningar och att det är viktigt att ta reda på vilka förutsättningarna är och agera utifrån dem.

Anders berättar att det finns mallar för hur överlämnande ska gå till för att förvaltningsorganisationen ska kunna vidareutveckla produkten. Det finns även dokument som innehåller de krav som produkten måste uppfylla för att få överlämnas och dessa krav ska godkännas av den person som är produktansvarig innan ett överlämnande. Anders menar att man på Alfa börjar bli allt bättre på att ta reda och dokumentera de icke-funktionella kraven som är viktiga för att kunna sköta förvaltningen. Checklistor för vad som behövs innan överlämning till förvaltning finns hos Alfa, dock anser Anders att dessa inte fungerar som bärare av kunskap utan kunskapen om produkten bärs av individen. Han menar att det är individernas kunskap som avgör hur projektet går, och att det är därför det gäller att ha kompetent personal i både projekt och förvaltning, samt att personal från den blivande förvaltningen får lov att ingå i projektet. Göran påpekar att det inte är artefakterna som är de viktiga utan de aktiviteter man genomför och har inte de som ska förvalta objektet kunskap om dessa aktiviteter så räcker det inte med att gå igenom dokumentation för hur de gick till att göra aktiviteterna, det krävs att man faktiskt varit medverkande i någon mån.

Anders menar även att man inte gör någon skillnad på personal i förvaltningen och i projekten när det gäller utbildningar för att lära sig Alfa IT Processes eller RUP. Adam säger: ”många delar som man använder i RUP använder vi också i förvaltningen. Alltså arbetsprocesserna.” Skillnaden ligger mer i hur man arbetar, menar Anders, i projekt handlar det om att få klart allt och blir det inte klart till det satta datumet så flyttar man på datumet för releasen. I förvaltningen däremot jobbar man i kortare perioder, detta kallas timeboxing, vilket innebär att man under exempelvis tre veckor bygger en ny del och tar endast med det allra viktigaste och låter det som man inte hinner med vänta till en annan gång. Release är alltid på detta satta datumet, oavsett om man fått med alla tänkta funktioner eller ej. Adam menar att Alfa inte har någon egen modell för förvaltningen och man använder inte heller RUP inom förvaltningen, dock är det många processer som används som kommer ifrån RUP. Adam menar att processerna knappast skiljer sig alls för hur man arbetar i projekt hur man sedan arbetar i förvaltningsorganisationen.

3.5 Kunskapsöverföring i RUP

Arne anser att det finns ett visst stöd för kunskapsöverföring inom RUP, detta utifrån alla de artefakter som skapas och att man den vägen är överens om vad projektet ska leverera. Arne säger; ”det finns väldigt mycket inbyggt i det (RUP) som ger bäring på ett överlämnande”.

Bengt menar att artefakterna i RUP är bra för att överföra kunskap, förutsatt att man inte drunknar i det. Emellertid tror han inte att det endast är RUP som ska överföra kunskap utan att det hänger på individerna i projektet om det blir en lyckad överföring. Bengt menar att för att utnyttja RUP som bäst i ett projekt gäller det att ha väldigt god insikt i hur man ska använda det och vad inom RUP man ska använda för att uppnå bästa möjliga resultat för respektive projekt. Huruvida man i organisationen kan utnyttja de skapade dokumenten beror på individerna menar Bengt och han säger även att RUP slutar lite för tidigt när det kommer till just överlämningen från projekt till förvaltning.

För att överföra kunskap som utvecklas i utvecklingsfasen av ett projekt menar Arne att det dokumentmässigt finns de olika artefakterna som man tar fram inom ramarna för RUP. Adam säger;

”Om vi håller oss till själva produktkunskapen så finns ju de här produktartefakterna men i verkligheten så kan man ju inte bara skapa ett antal dokument och lämna över till helt andra personer att förvalta utan det bästa sättet att föra över kunskap, det är ju att vi fasar in antingen från början dom som ska ta hand om applikationen i projektet och sen tar man med sig det då. Men det har ju inte med RUP eller nånting att göra utan det är ju sunt förnuft där.”

För att försäkra sig om att få med kunskapen in i förvaltningen utser Alfa tidigt i projektet en mottagare av sin produkt. Med detta menas att man utser en fysisk person som får möjlighet att följa projektet under dess livstid och med denna persons hjälp försöker man sedan bemanna projektet med personer som senare kommer vara en del av förvaltningsorganisationen. På detta sätt får dessa personer med sig kunskap från projektet in i förvaltningen och Arne menar att detta underlättar hanteringen i senare skeden. Arne säger;

”Och så känner jag att man liksom alltid tänkt och alltid haft personer med i projektet som ska ta hand om det sen. Så då blir det ju en kombination av (...) artefakter men också då kunskap som individer samlar på sig.”

Adam påpekar dock att detta med att fasa in personer som ska förvalta produkten inte kommer ifrån RUP.

Göran menar att man på Alfa blivit bättre på kunskapsöverföring sedan man skaffade RUP. Detta för att man till skillnad från tidigare dokumenterar kraven på ett strukturerat sätt och fångar designen så att det är möjligt att lämna över den till förvaltningen sedan, tidigare var man tvungen att läsa koden för att förstå systemet. Han säger:

”RUP innehåller ju inte så mycket konkreta aktiviteter som man att genomföra för att föra över kunskapen, alltså det är en sak att man dokumenterar och har en hög papper i princip att lämna över och det är en annan sak att man verkligen också lämnar över kunskapen från person till person. Inte bara en hög med papper, utan försöker genomföra aktiviteter på något strukturerat sätt så att man tar till sig den information man får det innehåller inte RUP något specifikt kring. Så på Alfa har man lagt till lite egna rutiner för hur man genomför en handover från utvecklingsprojektet till underhållsverksamheten (...)” och fortsätter ”*Man* ska se till att försöka få in resurser från underhållsverksamheten till att jobba i projektet under en period för att på det sättet lära sig och ta med sig kunskapen mer handfast in i underhållsvärlden sen.”

Adam påpekar dock att kunskapsöverföringen, eller fasen Transition är den svaga delen av RUP, det är där man saknar organisatoriska riktlinjer. Adam påpekar dock en osäkerhet gällande huruvida det är ett ansvar som ligger inom RUP att se till de organisatoriska aspekterna eller ej.

Adam menar att det viktigaste vid överföring till förvaltning är hanteringen av resurser, att man säkerställer att en del av de resurser som finns i projektet även medverkar i förvaltningen. Han säger att han inte tror att man kan följa RUP och ha en massa dokumentation att lämna över och det sedan ska vara bra, denna del, den mjuka delen som han kallar den, delen där man kompetensöverför är det som saknas i RUP.

På Alfa har man i och med införandet av RUP även blivit bättre på att dokumentera de ickefunktionella kraven, enligt Anders. Han säger:

”Det har vi varit dåliga på tidigare men de här ickefunktionella kraven börjar komma in tydligare och tydligare nu så att vi blir bättre på det här. Att hantera sånt som är viktigt för maintenance.”

Dock tror han inte att det är möjligt att dokumentera för kompetens, utan checklistor och RUP handlar mer om att lämna över dokument som kan hjälpa förvaltningsorganisationen i deras arbete. Göran menar även att RUP inte har några konkreta aktiviteter för att föra över kunskap, men att man på Alfa har gjort upp rutiner för denna överföring. Exempel på detta är att de tar in individer från förvaltningen i projektet för att på så sätt skapa en kunskapsbas hos dem. Bengt säger: ”Att fånga kunskap i projekt, RUP hjälper ju egentligen inte till med det.” Han menar att det istället handlar om hur man kan ta tillvara på den dokumentation som framställs inom ramarna för RUP. Bengt säger även att RUP slutar lite grann för tidigt ur just förvaltningssynpunkt. RUP löser helt enkelt inte problemet med kunskapsöverföring.

Arne menar att det inte går att slaviskt följa RUP för det skulle ge stora problem, för att det helt enkelt blir för tungrott, istället ska man se RUP som ett ramverk att tillämpa på ett vettigt sätt. Bengt säger att RUP ”är gigantiskt, alltså åt helvete för stort egentligen.” Anders menar att projekten är beroende av individernas erfarenheter, ju mer erfarenhet projektledaren har desto mer kan han begränsa projektet till de artefakter som behövs. Oerfarna projektledare använder mer artefakter för att inte misslyckas och detta gör att det tar mer tid och blir mer tungrott, däremot behöver det inte bli sämre resultat för det.

3.5.1 Checklistor

Inom Alfa har man tagit fram ett antal checklistor för att kunna stämma av så att man i projektet fått med de viktiga delarna. Arne menar att varje företag individuellt måste bestämma sig för hur de ska förhålla sig till RUP och de har gjort det genom att utveckla dessa checklistor som Arne menar i stort följer RUP. I sin tur måste sedan varje projekt inom Alfa bestämma sig för hur de ska förhålla sig till dessa checklistor, det finns en del som man måste göra men även delar som är frivilliga.

De checklistor som används på Alfa är från början framtagna av Anders men har efter den första versionen blivit reviderade i flera omgångar av andra inom organisationen. Checklistorna som han tog fram grundar sig på hans erfarenhet från tidigare projekt. Checklistorna, menar Anders är dock mer generella, det vill säga, de utgår inte från RUP och de kan användas i vilket utvecklingsprojekt som helst, alltså inte bara för utveckling av

mjukvara. Samma sak gäller för PPS, den modellen kan också användas i alla sorters utvecklingsprojekt, inte bara för mjukvara, menar Anders.

Anders menar han hade två skäl till att ta fram checklistorna, dels att många projektledare ibland finner det svårt att veta hur de ska lägga upp sina planer för projektet. Den andra delen syftar till att ge styrgruppen något att gå efter. Checklistorna ska alltså ge stöd till projektledaren och styrgruppen för att de ska veta vad som väntas av dem i form av dokument och beslut.

3.6 Sammanfattning av den empiriska studien

I detta kapitel sammanställs intervjuerna under olika teman som syftar till att beskriva och sammanfatta respondenternas svar. Kapitel ger en kort beskrivning till företagen, respondenterna och deras arbetsuppgifter, det ger även en kort sammanfattning i hur Alfa använder RUP. Efter detta kommer de fyra temana, Mognad och RUP, Kommunikation, Kompetens i förvaltningen och Kunskapsöverföring i RUP.

Mognad och RUP behandlar främst de problem respondenterna upplevde att Alfa har gällande mognad inom företaget i samband med användning av RUP. Avsaknad av mognad kan sägas främst bero på att RUP för företaget är en relativt ny modell och att den kräver stora organisationsanpassningar.

Kommunikation tar upp de olika former av kommunikation som sker i projektet. Det behandlar även styrgrupper och ägare till projekt, något som enligt respondenterna fattas i RUP och som är något som ett projekt måste ha för att fortskrida.

Kompetens i förvaltningen behandlar på vilket sätt man på Alfa förbereder förvaltningsorganisationen för deras arbete. Hur man under projektets gång förbereder efter den förvaltning som sedan skall komma. Här behandlas även likheter och skillnader mellan projekt och förvaltning.

Kunskapsöverföring i RUP tar upp hur man på Alfa faktiskt gör för att överföra kunskap, här tas även upp hur RUP hjälper till med detta samt de brister som finns. Respondenterna anser att RUPs artefakter till viss del hjälper till med överföring av kunskap, men detta är en svår del i arbetet. På Alfa försöker man för att försäkra sig om kunskapsöverföring på ett tidigt stadium i projektet se till att det finns resurser från förvaltningen som kan medverka i projektet.

4 Diskussion

I detta kapitel analyseras de svar som jag fått fram från mina intervjuer och ställs mot teorin. Kapitlet börjar med att beskriva hur man på Alfa lagt upp sin systemutveckling med de modeller de valt att använda. Efter detta tar jag upp hur man anser att modellen stödjer kunskapsöverföring från systemutvecklingsfas till förvaltningsfas.

4.1 Användning av metoder

Alfa använder sig av två modeller i sina systemutvecklingsprojekt, PPS (Praktisk Projektstyrning) och RUP (Rational Unified Processes). Det ska dock beaktas att användandet av modellerna är en relativt ny företeelse, de har endast använts i två-tre år. Dessutom är genomförandet av processerna inte tvingande utan används mer som riktlinjer. Det har i intervjuerna påpekats att det är en stor omställning som kräver mycket tid och arbete att göra omställningen inom hela organisationen för att övergå till den nya modellen som man kallar Alfa IT Processes. De intervjuade påpekade även att alla inte använder modellen även om det är tänkt så och att man ofta anpassar den utifrån sitt eget sätt att tänka. Trots detta tycker man att det är en bra modell, framförallt som vägledning eller ramverk i arbetet med systemutveckling. Anledningen till att man på Alfa valde att använda just RUP som systemutvecklingsmetod var främst att man kände att man behövde en gemensam metod för att göra arbetet i projekten mer likformigt och sedan föll valet på RUP för att detta är den största och mest heltäckande modellen som finns i dagsläget, vilket även Wiktorin (2003) bekräftar.

På Alfa har man som sagt integrerat RUP med PPS. PPS är i detta fall en överordnad process som används för alla former av projekt inom företaget (Lunell, 2003). PPS är en modell för styrning av projekt som tagits fram av TietoEnator. Modellen innehåller delar som RUP saknar när det gäller just styrning och den är främst till för att projektledaren. I PPS finns ett antal milstolpar som passar mer eller mindre in på RUP och man har på Alfa tagit fasta på dessa och använder dem för att kunna göra överlämningar till styrgruppen. På så vis har man löst det som Lunell (2003) beskriver som en knäckfråga när det gäller att integrera milstolpar i överordnade processer med milstolparna i RUP. Det är på överlämningarna av olika artefakter, som är i dokumentform, som styrgruppen grundar sina beslut om projektet ska fortskrida. Henderson-Sellers et al (2000) menar att RUP i dagsläget inte fungerar självständigt utan att det krävs utvidgning inom områdena; kunskap, anskaffningshantering, kommunikationshantering och personalhantering samt att den är bristfällig när det gäller projektledning.

En av de punkter som respondenterna tog upp som RUP saknar är just de här styrgrupperna, de menade att hade man ingen styrgrupp, så kunde man likaväl lägga ner projektet. De talade även om vikten av att ha en ägare till projektet, någon som ser till att det finns en organisation som man kan lämna över till. Denna ägare utses hos Alfa i den initiala fasen av ett projekt och är sedan ansvarig för att se till att det finns en klar organisation att lämna över till i projektets sista fas.

4.2 Tillvaratagande och överförande av kunskap

RUP innehåller enligt Strand (2001) en stor samling av kunskap och erfarenheter som hon anser att alla som arbetar med systemutveckling bör ta del av. Inom varje fas i RUP finns ett antal artefakter som ska skapas, med artefakter menas något som skapats, ändrats eller använts i processen (Strand, 2001). På Alfa har projekten en baslista för de artefakter de ska ta fram och vem som är ansvarig för framtagandet, det finns bland dessa artefakter även ett antal som är obligatoriska. Strand (2001) menar att metoden egentligen inte är till för att man ska skapa en massa pappersdokument utan för att uppmuntra projektmedlemmarna till att fokusera på att ta fram UML-modeller, kod, med mera. Många invändningar mot RUP handlar om just detta att det tar mycket tid och kraft att skapa en massa pappersdokument, menar Strand (2001). Adam håller med om detta och menar att projekten blir tungrodda i början innan man lärt sig vad man kan begränsa sig till. Fördelen, menar Göran, blir att dokumentationen över projekten hos Alfa numera är strukturerad och man har blivit bättre på att fånga designen och i och med detta kan man lämna över något till förvaltningen som är möjligt att ta till sig.

Respondenterna menar att även om det finns mycket som ”ger bäring på ett överlämnande” som Arne säger så finns det i RUP inga konkreta aktiviteter för att överföra kunskap men rutiner på för överlämnande har Alfa skapat själva, enligt Göran. Exempel på dessa rutiner är att ta in individer som ska finnas i den framtida förvaltningsorganisationen i projektet. Ansvar för att det finns en förvaltningsorganisation ligger hos ägaren och det är ägaren som ska se till att den förvaltningsorganisationen som tar över produkten har den kunskap som krävs för att kunna förvalta produkten. Detta sker som sagt genom att man utser resurser från den framtida förvaltningsorganisationen som får delta i projektet, detta är inte heller något som RUP tar upp utan det kommer sig enligt respondenterna av sunt förnuft. Respondenterna anser inte att det är möjligt för förvaltningen att klara av att göra ett bra jobb om de endast får en massa dokumentation för produkten. Strand (2001) nämner att det är viktigt att samtliga modeller, projektartefakter, beskrivning för arkitekturen med mera finns med vid överlämningen för att underlätta för vidareutveckling och förvaltning. Respondenterna påpekar dock att det krävs att individerna har kunskap om produkten, och att det är svårt att sätta sig in i den om det endast finns en massa dokument. Ändamålet med RUP är att genomföra bra aktiviteter som sedan dokumenteras.

Adam menar att kunskapsöverföringen, eller fasen Transition är den svaga delen i RUP, det är där man saknar organisatoriska riktlinjer, dock påpekar han en osäkerhet kring huruvida det är ett ansvar som ligger hos RUP att se till de organisatoriska aspekterna.

4.3 Sammanfattning av diskussionen

Respondenterna anser att RUP till viss del stödjer kunskapsöverföring genom de artefakter som metoden förespråkar att man inom projektet ska ta fram. De påpekar dock att detta ensamt inte fullt ut gör att överlämningen blir bra och att det blir möjligt för förvaltningsorganisationen att göra ett bra jobb. Det har även påpekats att det är viktigt att inte lägga allt för mycket kraft på att göra dokumenten utan metoden handlar enligt Strand (2001) om att genomföra bra aktiviteter och sedan dokumentera dessa.

På Alfa har man tagit fram riktlinjer för att främja kunskapsöverföring menar Göran, detta genom att utse en ägare till produkten som har ansvar för att det finns en

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap

-En fallstudie av RUP

Carin Olsson

förvaltningsorganisation som under projektets gång får möjlighet att ta del av projektet och som på så sätt kan tillgodose sig med kunskap om produkten. Detta sammantaget menar Göran har inneburit att Alfa blivit bättre på att överföra kunskap från projekt till förvaltning.

Respondenterna menar även att RUP saknar vissa delar när det kommer till styrning, vilket Alfa hanterat genom att även använda modellen PPS som behandlar just styrning. PPS är hos Alfa en process som används vid alla projekt, inte bara vid mjukvaruutveckling.

5 Avslutning

I detta kapitel återfinns en kritisk reflektion över undersökningens resultat.

5.1 Slutsatser

Syftet med denna uppsats var att undersöka på vilka sätt systemutvecklingsmetoden RUP stöder kunskapsöverföring från systemutvecklingsprojekt till förvaltningsfas och hur detta görs i praktiken på företag. Efter gjord litteraturstudie var det fortfarande oklart hur RUP i sig självt stöder kunskapsöverföring och det framkom tidigt i den empiriska undersökningen att RUP i sig själv inte stödjer detta. Respondenterna menar dock att kunskapsöverföringen blivit bättre hos Alfa sedan de införde RUP därför att när man började jobba efter metoden förde den med sig att de dokumenterade noggrannare. Min frågeställning var således; Hur kan användandet av systemutvecklingsmetoden RUP hjälpa till med överföring av kunskap vid övergången från systemutvecklingsprojekt till förvaltningsfas? Och svaret på frågan är att metoden i sig själv inte hjälper till med detta, det finns enligt respondenterna inga Best Practices för kunskapsöverföring men metoden har medfört att de blivit bättre på att dokumentera och på det sättet överföra kunskap. Sammanfattningsvis kan det sägas att vilken metod som helst hade kunnat fungera på samma sätt eftersom det är bieffekterna av metoden som gjort att man strukturerat upp arbetet och blivit bättre på kunskapsöverföring.

I övrigt så stämde användandet av metoden i mycket med det som kommit fram under litteraturstudien i fråga om rekommendationer gällande att hela metoden inte används samt att det större företag även har överliggande modeller.

5.2 Självkritik

Allteftersom min undersökning fortskred blev det allt mer uppenbart att min tilltro till RUP som en metod som fungerar för allt inte stämde. Det har varit svårt att få detta att gå ihop och att ändra min egen inställning till den metoden. Det var dock uppenbart ganska tidigt för mig hur det i slutändan skulle bli men trots det kände jag en motvilja mot resultatet. Det var svårt att få ett ordentligt grepp om begreppet kunskap, även om det var klart för mig var det innebar, var det svårt att beskriva vad som syftades till.

De metoder jag valt för genomförandet av undersökningen kändes helt naturliga och att göra ett fåtal djupintervjuer med användare och utbildare av metoden gav en bra grund för min empiri. Trots detta har jag svårt att tala för att resultatet skulle vara generaliserbart, jag anser att det finns vissa grunder för att RUP inte skulle fullt ut ge underlag för kunskapsöverföring men det går inte säga att det inte finns material eller rekommendationer inom RUP som inte ger en bättre utgångspunkt.

Detta är även den första större uppsats som jag skriver själv och det har inneburit att det varit svårt att tänka om när jag har fastnat och även svårt att hitta någon att bolla idéer med.

5.3 Diskussion kring vetenskaplig och etisk kvalitet

5.3.1 Vetenskaplig kvalitet

För att avgöra validiteten i denna uppsats måste jag fundera och ha i åtanke den bild som jag som forskare ger till mina respondenter vid intervjuer och andra undersökningar. Då jag är positivt inställd till metoder som ser till kunskapsöverbryggande och tror att detta är ett problem kan detta påverka mina respondenter så att även de ser ett problem som kanske inte finns. Gilovich (1991) diskuterar kring hur forskaren genom sina frågor påverkar svaren till att bli det som han själv förespråkar, eller tolkar dem positivt i den riktningen. Därför är det speciellt viktigt att vara medveten om vår påverkan vid exempelvis utformning av intervjufrågor. Jag kommer att försöka neutralisera mina frågor genom att granska dem kritisk och fundera på vad jag egentligen frågar om.

Enligt Gilovich (1991) är det enklare att komma ihåg det som talar för den teori man själv förespråkar än de argument som talar emot den. Detta kan låta självklart eller som något man selektivt gör för att man då vinner på det men som forskare måste jag ha detta i åtanke när jag sätter mig ner för att bearbeta det insamlade materialet.

5.3.2 Etisk kvalitet

Miles och Huberman (1994) påpekar att forskaren måste tänka på etiska frågor innan, under och efter studien. Wallén (1996) anger att kvalitativa studier ger etiska problem som inte uppstår vid kvantitativa studier. Detta beror på att tidigare vetenskapstraditioner starkt betonat forskarens objektivitet. I kvalitativa studier spelar forskarens syn på problemet, hans världsbild och värderingar in.

Enligt Wallén (1996) så är det viktigt med informerat samtycke vid undersökningar och jag kommer därför vara noga med att förklara för deltagarna i studien vad den går ut och därefter låta dem ta ställning till huruvida de vill vara med eller inte. Deltagandet måste även vara frivilligt annars kan undersöks personerna försöka skydda sig själva i relation som saknar tillit (Miles, et al, 1994).

Jag kommer att erbjuda de deltagande i intervjuerna konfidentialitet och jag kommer i så stor utsträckning som möjligt försöka göra det omöjligt att identifiera enskilda personer i rapporten.

En annan viktig aspekt gällande etisk kvalitet är att som forskare fråga sig, har jag den kompetens som krävs för att genomföra studien på ett vettigt sätt (Miles et al, 1994). Jag anser att jag har denna kompetens eftersom jag studerat de aktuella metoderna tidigare. En annan fråga som är berättigad att ställa är huruvida undersökningen är av nyttovärde eller inte (Miles et al, 1994). Jag anser att den har detta nyttovärde då jag tidigare kommit i kontakt med organisationer som upplever brister i just den fråga jag ämnar undersöka.

5.4 Förslag till framtida forskning

En tänkbar fortsättning vore att undersöka hur andra företag gör för att förmedla kunskap mellan projekt och förvaltning och se om mitt resultat gäller generellt. Utifrån detta skulle det sedan kanske vara möjligt att utvidga materialet i RUP till att även innefatta dessa delar.

6 Bilagor

I denna bilaga kommer transkripten från intervjuerna presenteras. Dessa transkript är städade, det vill säga genomgångna för att fungera som löpande text, talspråket är modifierat. Texten är även städad på avsnitt och ordval som skulle kunna göra det möjligt att identifiera respondenterna och företagen. Kursiv text innebär att det är intervjuaren som talar och annars är det respondenten.

Bilaga 1: Intervjuformulär

Detta är endast huvudfrågorna och då intervjuerna var ostrukturerade gavs intervjupersonerna chansen att svara som de ville och kommentera och det utvecklades även andra frågor från intervjuarens sida under intervjutillfällena. För specifikationer om respektive intervju se Bilaga 2-6.

Allmänt om RUP och Alfa/Mognad och RUP

Vad är din befattning?
Hur länge har du jobbat med detta?
Vad för slags projekt arbetar du med nu?
Hur länge har du arbetat med RUP?
Hur länge har Alfa använt RUP?
Varför valde Alfa att använda just RUP?
Används RUP i alla Alfors utvecklingsprojekt?
Använder ni någon annan systemutvecklingsmetod?
Används RUP på samma sätt i alla projekt?

Alfa och artefakterna/ Checklistor

Vilken organisationsstruktur används inom projekten?
Vilka ansvarsområden finns?
Vilka är de ansvariga för de olika processerna ur ett RUP-perspektiv?
Vilka arbetsprocesser används?
Vad finns det för olika kompetenser inom arbetsgruppen?
Vad/vilka processer dokumenteras vid ett projekt?

Kommunikation

Vilka former av kommunikation används i projekt? Formellt/informellt?

Kunskapsöverföring i RUP

Finns det processer för att "ta med" kunskap som bildas i utvecklingsprojekt in i förvaltningsfasen? T.ex. utbildning?
Vad anser du om RUPs förmåga att fånga kunskap i projekt?
Kan artefakterna i RUP användas för att överföra kunskap? Är de tillräckliga?
Stödjer RUP förvaltning?

Kompetens i förvaltningen

Vad finns det för kompetenser inom förvaltningen?

Bilaga 2: Intervju med Arne på Alfa

Nu på Alfa så är jag på mitt andra projekt när det gäller då RUP. Där vi har jobbat med RUP och PPS. Ett projekt som avslutades förra året vid den här tiden och ett projekt som jag håller på med.

Är det stora projekt?

Ja, allting är ju relativt men om vi pratar om antal personer är det lite olika under olika faser här men det är tio till fjorton personer som är involverade i det här projektet. Så det är et medelstort projekt.

Du har jobbat med RUP i två år?

Jag har jobbat med RUP i två år och jag har också en bakgrund såtillvida att jag har jobbat med... när jag började på Alfa (i slutet av 80-talet) jobbade jag med modeller, metoder och verktyg för systemutveckling och för projektstyrning så jag har en viss bakgrund, en viss metodbakgrund också som sagt.

Vet du varför Företaget valde just RUP?

Nej det kan jag väl inte säga, det var under den tiden jag var borta från Alfa egentligen. Men jag vet ju då vad som fanns innan RUP, en ganska traditionell vattenfallsmetod. Den kanske inte var så stringent använd, man använde den på lite olika sätt här. Att man fastnade för RUP, alltså man hade behov av att strama upp arbetet, arbeta mer likformigt och så vidare. Men varför det då blev just RUP i förhållande till något annat, det vet jag inte.

Använder ni någon annan systemutvecklingsmetod nu?

Nej, jag kör enligt RUP. Kopplat till det finns PPS.

Används RUP på samma sätt i alla era projekt?

Alltså, jag vill nog se RUP som något slags ramverk. Sen så får man då göra vissa anpassningar beroende på hur projektet ser ut. Jag vet första gången jag kom i kontakt med RUP, det är ganska precis två år sen, jag visste egentligen inte så mycket om RUP innan. Jag hade i och för sig läst artiklar i tidningar men jag hade liksom inte kommit i clinch med den. Jag vet då för vi hade på den här enheten som jag jobbar på så hade man utsett ett team som skulle då stötta RUP-införandet. Det teamet hade jag nytta av i mitt första projekt men jag minns då när vi träffade dom första gången, då skulle dom berätta lite grann om det här och då pratade som om det här med iterationer och discipliner och roller. Dom pratade mycket om det här med roller. Till slut var jag tvungen att fråga om de här rollerna; hur många är dom? Ja, det är trettiofyra roller sa dom, men vi använder inte alla. Sen så gick det en stund och dom pratade mer och mer om dom här rollerna. Så tänkte jag; hur många använder vi då? Och då var det någon som räknade efter i någon lista och så sa dom; vi använder tjugoåtta. Och där kände jag; det är ett område där jag kanske inte arbetar så tydligt med rollerna som man t.ex. kan läsa då i RUP. Jag ser mera då ett antal discipliner. Vi har business modellering, requirement, Analyze, Design, Implementation, Test och så vidare. Och jag ser då att jag orienterar arbetet runt omkring dom disciplinerna och där kan jag väl tänka mig att man hanterar det lite olika i olika projekt. Dels det som för mig nog blir någon slags utgångspunkt i alla fall det är att det finns någonting som heter Development Case. Vet du vad det är? Det är ett konstigt ord men i alla fall så är det en sammanfattning av hur man ska jobba i det enskilda projektet, framförallt vilka artefakter som man ska använda. Där det egentligen finns en baslista av artefakter där man bestämmer sig i det enskilda projektet vilka artefakter man ska ta fram och vem som är ansvarig för dem och så vidare. Och dom artefakterna man inte ska ta fram av något skäl, dom talar man om varför man inte ska ta fram dom. Och där finns också en del musts så att säga. Project definition och så vidare och andra dokument, dom ska ju finnas. Medan det kan finnas andra saker som inte ska finnas. Det blir ju också något som är styrande och vägledande i arbetet, någonting att arbeta mot. Som tycker jag skapar en ganska stor trygghet.

Vilken organisationsstruktur använder ni? Vilka roller använder du?

Som jag sa försöker jag orientera arbetet runtomkring de här disciplinerna. Pratar vi då business modellering har vi ju då en business analyzer som jobbar med användarna när det gäller modellering och så vidare. Pratar vi requirements då finns en system analyzer som har att jobba med att skriva use case och delta i utformningen av supplementary specification. Pratar vi analys och design så pratar vi om solution architects. Och det finns i och för sig lite olika arkitekter, någon är inriktad mot IS alltså det som så småningom blir databas och andra är mera inriktade på de funktionella delarna. Pratar vi implementation, där har vi väl möjligtvis ett problem i övergångarna mellan de här disciplinerna är inte alltid glasklara. Alltså bara ta där man kan säga att en

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

implementatör kanske också gör en del utav designarbetet. Där möter ju då arkitekten implementatören. Implementatör i de miljöer vi använder så har vi liksom olika skiktningar. NITF har du kanske hört talas om? Den arkitekturen innebär att på implementationssidan så behövs klientutvecklare, serverutvecklare och databasutvecklare. Sen så kommer vi till test, där har vi flera olika roller. Du har testledare, som jämt leder testarbetet, du har den som skriver test-case, den som gör testerna, du har den som har ansvar för testmiljön. Där kan man då hitta en fyra, fem roller. I de flesta projekt så är det liksom en individ som hanterar dom rollerna. Och jag kan nog se att för mig finns det ett ganska starkt behov att klumpa ihop en del av de här sakerna för att inte få för många personer i projektet. Ett projekt skulle väldigt lätt kunna svälla väldigt mycket och det skulle kunna generera en overhead, därför måste man koppla ihop vissa roller. Inom test blir det väldigt tydligt. Även om vi har en fyra-fem roller inom test så är det en person. Sen har vi då någon som jobbar på CM, Change Management. Jag försöker disciplinorientera det hela och försöker få så få personer som möjligt med i projektet, vilket är lite tufft idag eftersom vi går mot en ökad specialisering.

Vilka arbetsprocesser används?

Hur tänker du då?

Jobbar ni enligt det som är specificerat inom RUP?

Man kan ju säga då att inom varje sån här disciplin som jag pratade om så finns det en beskrivning av hur man ska arbeta och där kan jag säga att det kanske man inte tittar så jättemycket på. Åtminstone så gör inte jag det som projektledare utan jag är mer fokuserad på artefakter och resultatet som ska tas fram. Sen måste jag då som projektledare förutsätta att man i respektive roll vet vad man ska göra. Där är vi mitt uppe i nånting just nu, allting sitter ju inte hos oss än, utan vi har mycket att ta till oss. Man kan säga; det som vållar problem är övergången mellan olika moment där man kan säga att artefakterna är något slags medel. Men man får inte förlita sig för mycket på artefakterna heller utan kommunikationen i projektet är viktig också.

Vad finns det för kompetens i arbetsgruppen för att använda RUP?

Man kan ju säga då att när man som projektledare som jag är så vill man ju liksom egentligen få in personer som är tränade i sina roller. Som har en grundträning i sina roller. Men nu är vi i en uppbyggnad och då får man också som projektledare acceptera att alla är inte tränade i sina roller. Utan många av de här projekten får lite grann av en pilotkaraktär över sig. Det innebär ju också att vi får en hel del diskussioner och en del störningar för vi vet inte riktigt hur vi ska arbeta. Alla arbetsprocesser sitter inte så som dom ska. Men där känner jag att vi tar steg för steg och i det projekt som jag leder nu så har jag tre medarbetare iväg denna vecka på en så kallad bootcamp. Där de inte bara ska titta på sin egen disciplin utan de ska försöka förstå helheten. Det jag skickat med dom då är att de ska vara väldigt uppmärksamma på sina, inte bara sin egen uppgift utan det jag gör, vem ska jag lämna det till och det jag får vem får jag det av och vad har de för förutsättningar? Så att man sätter in sin egen uppgift i en helhet. Och det är bland annat då för att komma ifrån allt för mycket det här neutrala överlämnandet i form av artefakter och så.

Hur dokumenteras processerna i ett projekt? Fungerar det bra?

Menar du det som projektet ska leverera? Eller menar du RUPs processer?

Det som projektet ska leverera.

Det är ju dom sakerna som tas om hand om i de inledande disciplinerna, i business modellering och så vidare. Det beror ju lite grann på var projektet har sin utgångspunkt. Finns dom sakerna modellerade sedan tidigare, alltså i en affärsmodul som liksom är gjord innan projektet startar, det är ju en väldigt bra utgångspunkt om det är på det sättet. Och att man kanske då inom ramen för projektet detaljerar saker och ting. Där befinner sig då Alfa i ett läge där vi jobbar med att utveckla våra processer. Nu har ju de olika projekten olika förutsättningar när man går in med sitt arbete. Men det är klart att det blir ju tyngre projekt om de överordnade processerna inte är klara. Där är väl vi i ett läge där inte alla överordnade processer är klara.

Vilka former av kommunikation används i projekten?

Som i det här projektet, som jag sa så är artefakterna i sig också någon form av bärare utav kommunikation. Men det räcker inte liksom. Sen finns det arbetsmöten och arbetsmöten de uppstår ju när det finns ett behov och mellan de människor som har behov av att prata med varandra. Allt sånt här arbete består av individuellt arbete och gemensamt arbete i form utav kortare eller längre workshops och då är det ju alltid någon som sitter i någon slags driving seat då för att genomföra de här workshops. Så det är ju en sida utav det. Så t.ex. någon som jobbar med use case utformning har ju då resurser till sitt förfogande för att genomföra de arbetena och då är det ju arbete i grupp och så vidare. Men sen så behöver man ju även ha lite mer formella möten. Så t.ex. så i mitt projekt så har vi både då business resurser och itresurser. Och min roll är ju då att vara itprojektledare och sen så finns det en businessprojektledare. Jag samlar ju då it resurserna en gång i veckan till ett möte. Varken mer eller mindre än en timme. Där alla får chansen att berätta kort vad jag gjort, vad håller jag på med, vad ser jag för problem. Och det finns två syften med de mötena. Det ena syftet är att man ska vara allmänt orienterad om vad

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

som händer i projektet inte bara där jag håller på med men vad håller de på med på övriga håll. Det är det ena. Det andra är att man den vägen kan lyfta upp problemställningar och få hjälp med att hantera dessa problem. Alltså göra projektet som helhet medvetna om de problemen. Så det är ju då lite mer såna möten som vi har varje vecka med egentligen då två huvudsyften. Sen så har vi också formella möten med businessrepresentanterna där vi också sitter en gång i veckan och går igenom vad som har hänt och vad vi ska göra och också tittar på finns det några stötestenar, finns det några diskussioner som börjar dra ut på tiden och så vidare, som börjar växa. Vi fattar några beslut för att komma vidare. Och sen är det ju också då styrgruppsmöten. Sen har man ju, vi ingår ju då i ett program, vi ingår i nåt som heter (namnet på programmet). Vi har ju också kommunikation mot resurser som finns på programnivå. Våra arkitekter jobbar tillsammans med arkitekterna på programnivå för att säkerställa att det vi gör passar in i nåt slags helhet. Det är mycket kommunikation med lite olika syften. Både då i mera formella sammanhang och mer informella sammanhang. Jag vet inte om det täckte in din fråga om kommunikation.

Jo jag tror det. Businesssidan – är det dom som räknas som kunder?

Ja det kan man säga. I just i det projektet jag jobbar nu så hanterar vi det på det sättet att vi har businessresurser inne i projektet, alltså business developers. Som ju inte sitter ute i verksamheten utan de har då fått avsatt tid och sitter på heltid inne i projektet som representanter för de slutliga användarna. Det som är bra med det är att man verkligen har avsatt resurser, den risk man tar är att vi kommer en bit från de riktiga användarna. Så det är liksom en liten balansgång.

Finns det processer för att "ta med" kunskap som utvecklas i utvecklingsprojektet in i förvaltningsfasen?

Ja, för det första så dokumentationsmässigt; alla de här artefakterna som jag pratade om finns. Man kan säga då att det finns projektartefakter och att det finns produktartefakter. Projektartefakter är såna som typiskt bara lever under projektet. En iterationsplan till exempel. En produktartefakt, en typisk är en use case beskrivning, alltså såna saker som ska leva så i det här development cases som jag pratade om med förteckningen över alla artefakter, där finns det också angivet; vilka är projektartefakter och vilka är produktartefakter. Så det är alltså väldigt tidigt definierat vilka artefakter som vi ska leverera. Sen så har vi även tidigt utsett en mottagare. Alltså i den blivande förvaltningsorganisationen. En fysisk person som då har möjlighet att följa projektet. Men den personen, när vi bemannar projektet så försöker vi också bemanna projektet på ett sånt sätt att kanske en del från den blivande förvaltningsorganisationen är med i projektet. Och därmed också tar med sig kunskap in i förvaltningen. Det kan ju vara bra t.ex. om någon programmerare som sitter i förvaltningsorganisationen är med i projektet, det underlättar ju hanteringen i senare skeden. Där får man ju helt enkelt när man bemannar projektet titta på vilka resurser vill man ta med och kan man ta med från den blivande förvaltningsorganisationen. Och så känner jag att man liksom alltid tänkt och alltid haft personer med i projektet som ska ta hand om det sen. Så då blir det ju en kombination av leverabler av artefakter men också då kunskap som individer samlar på sig. Sen gör man då formella överlämnanden och då gör man oftast det, som jag ser det i alla fall när projektet är avslutat. Även om man gör delleveranser. Det här kan man i och för sig hantera på lite olika sätt om det är olika förutsättningar men den här iterativa planen som finns i RUP här det finns ju tänkt då, eller man ska ju kunna göra delleveranser efter de olika iterationerna. Då kan man ju också fundera, ska man lämna över till förvaltningen efter delleveransen och därmed också ansvaret för support. I dom projekt jag deltagit i så är det nog vanligast så och att man behåller supportansvaret i projektet till en slutöverlämning. Men skulle det vara ett stort antal användare, alltså tusentals användare då är det nog klokt att lämna över till alltså supporten direkt. Där får man alltså fundera från projekt till projekt hur man ska göra.

Vad finns det för kompetenser hos förvaltningen för att ta över ett projekt?

Meningen eller tanken är ju att dom redan alltså väldigt tidigt ska förbereda det här. Antingen så har dom personer, resurser som kan ta hand om det och som är örönmärkta för att ta hand om det och som då antingen kanske är med i projektet som jag beskrev här eller som under den här transitionfasen utbildas. Har man inte det läget så kanske man får rekrytera folk och säkerställa då att de finns på plats när överlämningen kommer. Där är vi väl på väg att bli mer och mer tydliga och liksom kräva att det finns en mottagare verkligen. Och inte bara på pappret utan att det finns en mottagare som har avsatt resurser och som är beredd. Nu när vi utvecklar, det händer ju lite då och då att man utvecklar saker i en ny teknisk miljö och det kan ju innebära att den mottagande organisationen också behöver utbilda sig i den här tekniska miljön. Och det kan ju vara så pass enkelt som att vi byter programmeringsspråk. Då måste det i den mottagande organisationen finnas någon som kan de nya programmeringsspråken. Däremot om vi utvecklar sjunde projektet i samma tekniska miljö då finns ju de tekniska kunskaperna redan i förvaltningsorganisationen. Man kan konstatera att varje projekt i såna här avseenden har sina unika förutsättningar och det gäller bara att ta reda på vilka är förutsättningarna och så agera efter det.

Tycker du att RUP ger stöd för kunskapsöverföring?

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Ja såtillvida alltså jag tycker att en sak som hjälper mig att skapa tydlighet i RUP det trots allt alla de här artefakterna. Alltså att vi den vägen är överens om vad det är projektet ska leverera. Där ser jag RUP är viktig. Sen finns det en iteration, i RUP som heter transition som just är det avslutande överlämnandet. Där man ska avsluta och verkligen lämna över. Så jag tycker att på så sätt att det finns ett stöd i RUP. Men när du kommer sen till förvaltningen då är det ju inte RUP så att säga utan då är det ju nåt annat, då är det väl. Jag tycker att det finns ett visst stöd på det sättet i RUP.

Använder ni någon ISO standard för IT-utveckling?

Nej inte så. Nej man kan ju säga att vi har ju på sätt och vis skapat vår egen ISO standard genom att vi har bestämt oss för att vi ska använda RUP. Genom att vi har bestämt oss för, vi har ju t.ex. mallar för alla dom här artefakterna. Så att på det här sättet så har vi ju skapat en intern ISO standard. Men sen vågar jag inte riktigt säga om den standarden stämmer överens med, det är jag inte riktigt uppdaterad på. Hur våran standard stämmer överens med någon form av industristandard, det vet jag inte.

Det var dom frågorna jag hade. Om det inte var något mer du ville berätta.

Nej. Jag har mycket att berätta. Det är ju frågan om det finns ju mycket att säga om RUP. Vad kommer rubriken på din uppsats att vara? Det kan ju vara möjligtvis att det triggas mig till att säga nåt.

Rubriken är: Systemutvecklingsmetoders innebyggda kapacitet att fånga och överföra kunskap?

Okej. Då tycker jag att använda RUP, det finns väldigt mycket inbyggt i det som ger bäring på ett överlämnande. Inte minst då de här produktartefakterna. Sen hur man ska förbereda för förvaltningsorganisationen på att ta emot någonting, det finns inte så många, det finns inte så mycket RUP i det. Utan det är mer att man tittar på vilka resurser man har och på hur behöver man förbereda dom. Hur komplexa är problemställningen och så vidare. Har dom jobbat med den här tekniska miljön tidigare och så vidare. Så det är väldigt mycket såna frågor som kommer in.

Men det finns ingenting i RUP som hjälper till med att förbereda det?

Ja alltså vi jobbar ju, om det är RUP eller inte, det vet jag inte men vi jobbar ju med en massa checklistor när det gäller såna saker så i det här programmet som jag pratade om att mitt projekt deltar i, (namnet på projektet) heter det här programmet där har vi verkligen jobbat, där har vi alltså på programnivå jobbat med det här med den här överföringen som kan man säga då ett komplement till RUP. Så det finns checklistor som man kan se; har vi gjort det, har vi gjort det, har vi gjort det. Är det RUP eller inte, det är någonting som ligger väldigt nära RUP egentligen som vi har tagit fram. Det blir ju så att varje företag måste på något sätt bestämma sig för hur man ska förhålla sig till RUP i grunden och där har vi gjort då genom att ta fram checklistor, exempel och så vidare. Sen måste varje projekt i sin tur bestämma sig hur man ska förhålla sig till det här. Där det finns vissa måsten och där det finns en viss frihet. Så att man måste nog se RUP som ett ramverk som det gäller att tillämpa på ett vettigt sätt och där det gäller att inte bli fångad utav ramverket i sig vilket skulle kunna vara väldigt lätt om jag knyter an till det här som jag sa från början om de här trettiofå rollerna. Skulle man liksom slaviskt följa det och liksom känna att det var jätteviktigt då skulle man få stora problem. Åtminstone skulle jag få stora problem.

Bilaga 3: Intervju med Adam på Alfa

Adam heter jag då, jobbar här på Alfa numera. Jag har jobbat här sen åttionio. Har varit projektledare sen nittiofem. Så jag har varit här länge. Drivit olika projekt och just nu jobbar jag med lite allt möjligt. Projektledare är jag till huvudsak men jag har inte haft något projekt här på ett halvår. Så jag har jobbat mer med lite annat, teamledare och konsultinköp och så. Senast jag jobbade inom projekt så var det inom (namnet på projektet). Så den här undersökningen då har, den är om RUP? För det har vi på IKEA, vi har ju inte jobbat så jättelänge med RUP men det har du nog förstått. Vi har jobbat med det i två-tre år tror jag.

Har du jobbat med RUP hela den tiden?

Ja, jag har jobbat i, nej jag har jobbat i ett projekt med renodlat RUP. Som var ett större projekt, det var väl, det beror på om man mäter det i pengar eller tid men det var stort i både pengar och tid överhuvudtaget. Ett och ett halvt år men sen slog i ihop oss med ett annat projekt, men det var stort det projektet och vi var väl en 15 i projektgruppen så det var väl medelstort. Det beror lite på vad man jämför med.

Vet du varför Alfa valde att använda just RUP?

Nej det vet jag inte men jag utgår ifrån att, jag hade en liten tid utanför Alfa. På (ett annat företags namn) och när jag kom tillbaka hade de valt RUP. Så jag var inte med i det beslutet men om jag förstår rätt så är det den största utvecklingsmodellen som finns just nu. Och jag tror att valet föll på det.

Används RUP i alla Alfa utvecklingsprojekt?

RUP är sagt att det ska användas, att det är Alfas utvecklingsmodell. Sen vet jag att den används ju inte riktigt överallt än. För en del projektledare jobbar ju på sitt sätt där det är möjligt kanske man inte använder RUP. Men RUP egentligen är ingen nytt, RUP är ju ganska likt mycket annat. Det är ju mer andra benämningar och kanske fler artefakter. Den kritik som är i tidningar och så är den kritik som är här internt på Alfa med, att det är en fruktansvärd massa artefakter och dokumentation. Och då blir det ju lite så att man om har osäkerhet om hur man använder RUP så finns det många som tar det säkra före det osäkra och använder några dokument för mycket när man väljer ut vad man behöver och då blir det ju rätt så tungrott i alla fall till att börja med. Men det är nog en mognadsprocess som vi tar oss igenom.

Använder ni någon annan metod?

Ja, sen har vi ju PPS för projektstyrningen då. Så där har vi mappat ihop PPSen och RUP. För det finns vissa delar som hör ihop och dom har vi använt från PPS som projektstyrningsmetodik.

Vilken organisationsstruktur använder ni skulle du säga? Alltså utgå från det som finns dokumenterat RUP att man ska ha?

Du tänker på roller och så vidare, ja, det gör vi ju också då för att vi har gjort en organisationsförändring, eller håller på, vi är mitt inne i den. Innan var det mycket så att vi hade ju en programmerare som tog hand om GUIt och systemutvecklare och programmerare och testare och allting samtidigt men nu går vi mer mot. Vi har identifierat ett antal processer och dom finns RUP också. CM och testprocesserna har utvecklats och jag vet inte hur många roller det finns i RUP men vi har ju infört mer tydliga processer, hur vi ska göra saker och ting. Och hur man ska kontrollera och hålla kvalitet i våra projekt och även under förvaltningen sen. Och då blir det också mycket att man specialiserar sig mer, man är en testmanager och ett antal testare och använder de verktygen. För vi har ju mycket, vi använder ju Rationals produkter, det är ju testmanager och clear case och clear test som passar in i dom processerna. Vi använder kan man säga den organisationen som RUP egentligen speglar i stora delar. Det anser jag i alla fall.

Vilka arbetsprocesser använder ni? Använder ni de som finns specificerade i RUP?

Ja, jag har ju inte mappat exakt RUPs arbetsprocesser med dom processer vi har men det är ju dom. Testprocessen och CMprocessen använder vi ju, den tror jag är ganska lik men nu är det vad jag tror jag har inte mappat det ordentligt. Men i och med att vi använder RUP och dom processerna, vi använder hela den verktygslådan som tillhör Rational. Och sen har vi ju en utvecklingsprocess också och den tror jag följer RUP givetvis för den använder vi när vi analyserar från krav ner till källkod och det följer RUPs arbetsprocesser.

Vad finns det för kompetenser inom arbetsgruppen?

Du tänker på rollerna, kompetensrollerna. Vi har ju då, de flesta projekt följer de normer och minimirollerna. Jag tror att minimirollerna i våra projekt nu för tiden om vi kopplar det till frågan är ju vi har dom här som kallas, vi

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

kallar dom business analysts, jag vet inte om de kallar, jag tror i RUP kallas dom system analysts. Det är dom som samlar in krav från användare och skriver ner dem use cases då. Sen har vi softvaruarkitekterna också. Och sen har vi utvecklare och sen har vi CM och testmanager, testmanager har ofta flera roller alltså dom både koordinerar testerna, skriver testplaner och i vissa fall också utför testerna. Och i mindre projekt så kanske de håller i hela. Och så projektledare. Och i vissa fall använder vi usabilityarkitekter också. Och sen är det ofta utvecklare som tar lite roller inom CM.

Vilka arbetsprocesser?

Ja och sen har vi också, och det ligger ju inte i RUP så, utan det tangerar ju lite... Vi har ju ofta vi har ju en organisation som heter IS nu eller utvecklingsdelen i Maintenance och sen har vi ju även driften då. Så vi har ju nästan i alla projekt så har vi ju en åtminstone en kontaktperson och sedan beroende på hur stort projektet är så är det mer eller mindre aktiva från driften så de är där och bevakar driftfrågorna.

Det är inte den systemägaren då?

Nej, det är inte systemägaren. Sen har vi ju andra människor som är skriver ihop SLAT och det är systemägarna och sen i projektet ingår det ju också givetvis en representant för beställarsidan eller affärssidan som skriver krav. Och de är också med i projektet på olika tid, det beror på om man ska ersätta något gammalt eller göra något helt nytt. Det styr också hur mycket tid (de ska lägga).

Hur dokumenterar ni era projekt? Gör ni allt det som står i RUP?

Nej det gör vi inte. Projektet dokumenterar ju sina minimibehov då, det gör ju alla, de använder de här use cases för att få ner kraven och även de här ickefunktionella kraven lägger vi i det här supplementary specification och sen då det kan man väl säga är kravdokumentet, sen finns det ofta ibland, vi kallar det visual, jag vet inte om vi tagit det från PPS eller RUP. Där beställaren beskriver på sitt språk vad dom vill ha, sen konverterar vi det då till use case och ickefunktionella krav. Sen vet jag att arkitekterna, använder det här när de bryter ner use case till programkod, bryter ner det i objekt. Och på den vägen har vi, det fungerar så att när vi ska lämna över nånting till driften. Vi har ju en lista, en template, den skulle jag kunna visa dig. Det finns ju en minimilista vad varje projekt måste ta fram, det är ett krav. Nu vet jag inte riktigt hur den ser ut i detalj, det är alltid nånting man glömmer, men det ju ett minimum för att, dels finns det ju andra intressenter som behöver dom då och dels så behövs för de tillhör själva produkten så produktartefakt då så dom måste ju följa med själva produkten sen. Så dom måste ju projektet ha. Sen är det ju lite mer fritt för projektet vad som är projektartefakter. Men vi har ju andra intressenter som t.ex. driften då. Och arkitekter på central nivå som, t.ex. arkitekter på central nivå gör ju sina revideringar, reviews t.ex. och då har ju dom specat ett minimum av artefakter dom vill gå igenom. Och det styr också projektet att vi måste ju ta fram dom för att kunna kommunicera den informationen. Även om vi kanske inte alltid tycker att vi behöver det, om det skulle vara så. Det finns ju andra intressenter som kräver det och vi har ju såna kvalitetsmöten med driften innan vi får lämna över det till produktion och de har ju också angett sina minimum artefakter. Och dom kan man frånga om man kan motivera varför eller om man har nåt annat ersätta dom med. Men det finns ju, det är inte bara projektet och själva produkten som avgör det.

Vilka former av kommunikation används i projekt? Är det formellt eller informellt?

Ja, det är ju till styrgruppen givetvis, det är ju internt i projektet. Eller du tänker mer utanför projektet? Eller internt?

Både internt och externt.

Både internt... ja, vi börjar internt i projektet, där är det ju en massa överlämningar från kravsidan till utvecklare sidan så där är ju ett antal interna processer i dom flesta projekten då. När man ska godkänna t.ex. ett use case, att man går igenom det och verifierar att man förstått kraven när man lämnar över till de olika personerna i projektet. Och det är ju i projektet, där jobbar vi rätt likartat i dom projekten med RUP. Det är ju oftast att man skriver klart use casen från kravsidan. Och sen reviderar man och sen så låter man väl arkitekterna gå igenom det och ställa frågor. Och sen gör man ... och sen är det godkänt. Och sen om man vill ändra det så har vi ju CCBer så det är också en intern process, hela det tillhör ju CM-processen. Sen externt har vi ju styrgrupper som vi kommunicerar med och vi har ju driftsidan som vi kommunicerar med, vi har ju arkitekterna som vi kommunicerar med, referensgrupper. Och sen ibland är det ju också dom här processägarna. Och i lite större program, vi har ju vi driver ju nu ett program inom ramen för, där vi ska processanpassa mycket, så att, det är ju ett program som driver det då. Och under sig har de ett antal projekt så i den programgruppen sitter det ju en konceptgrupp som ska hålla ihop allt det som projekten gör, se så att det stämmer i flödet. Så det är ju en viktig spelare att stämna av med. Det är väl dom stora spelarna.

Dokumenterar ni den kommunikationen?

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Ja, det gör vi ju. Det är ju den formella kommunikationen, styrgrupperna dokumenterar de formella kommunikationerna med arkitekterna och konceptgruppen, det är ju formella möten så dokumenteras det. Sen sker det ju mycket mellan mötena som inte dokumenteras. Men dom formella mötena dokumenteras.

Finns det processer för att "ta med" kunskap som bildas i projekt in i förvaltningsfasen? Finns det utbildning?

Både ja och nej, det beror på hur man ska se det. Om vi håller oss till själva produktkunskapen så finns ju de här produktartefakterna men i verkligheten så kan man ju inte bara skapa ett antal dokument och lämna över till helt andra personer att förvalta utan det bästa sättet att föra över kunskap, det är ju att vi fasar in antingen från början dom som ska ta hand om applikationen i projektet och sen tar man med sig det då. Men det har ju inte med RUP eller nånting att göra utan det är ju sunt förnuft där. Eller så kommer dom ju in nånstans i mitten av projektet eller fasas in efter hand i projektet, de som ska förvalta det. Allra senast kanske när man testar i olika faser. Och det gäller ju både på affärssidan och dom som förvaltar det på IT-sidan. För det kommer aldrig att funka om man tror att man kan lite på RUP och produktartefakterna bara. Och artefakterna, dokumentationen måste man ha i vilket fall som helst. Men ska man göra en bra överlämning så ska man nog ha folk från början med i projektet som är tänkt att förvalta det också. Så det är väl den bästa processen som vi har då som vi försöker följa då. Sen utbildning, ja, projektet ansvarar ju för att utbilda och rulla ut systemen då och kör våra utbildningskampanjer och sen på Alfa har vi ofta byggt upp det med superusers då va som är ett antal människor som får lite djupare utbildning och som utbildar så att man fokuserar på att utbilda dom då, i alla fall den erfarenhet jag har. Och sen utbildar dom i sin hand vidare människor. Sen har vi också mycket tydliga beställarorganisationer idag då. Det finns dom som äger det här systemet sitter ofta i beställarorganisationen. Bland annat då ansvarar för utbildning under applikationens livstid på användarsidan. Utbildning internt, alltså när folk byter jobb och någon ny ska ta över förvaltningen, det sker mer med att vi överlappar.

Dom som är med i projektet och jobbar med RUP, har dom utbildning för det?

Ja, dom flesta har ju gått dom här RUP-utbildningarna och dom som jobbar som CM har ju gått dom utbildningarna. Men det är ju ett jättesteg att gå utbildningar och börja arbeta med det, det krävs mycket erfarenhet.

Vad finns det för kompetenser inom förvaltningen för att ta över?

Där har vi ju ett litet glapp just nu för att dom projekt vi kör och där vi börjar använda RUP rätt mycket nu är ju då dom första RUP med alla dom processerna. Och dom skiljer sig rätt mycket från det vi gjort tidigare så där håller vi på att diskutera för där handlar det om att bygga upp kompetensen. Men så som vi löser det är ju att vi försöker köra in mer och mer eget folk i projekten som ska följa med till förvaltningen för processerna skiljer sig knappt nånting från hur man jobbar i projektet och hur man ska förvalta. Man har ju CCB man följer CM man följer testprocesserna och gör releaser och så vidare. Sen har man ju andra processer för hur man hanterar buggar, men själva arbetssättet är ju RUP oavsett projektet eller dom processerna, arbetsprocesserna. Så det håller vi ju på att fundera på. Det kommer ju rätt stora leveranser rätt snart.

Det stödjer alltså inte RUP alls i sig själv, att gå över till förvaltning?

Nej nu tänker jag ju på processerna i RUP. Om vi tittar på förvaltningen kommer ju inte jobba med RUP i projektform med Inception, Elaboration, Construction och Transition. På det viset, den synen har man ju inte i förvaltningen. Där är det mer löpande verksamhet, sen kan man väl säga dom processerna som man jobbar efter är ju inte RUP-specifika. Men dom finns ju med i RUP, hur vi testar och hur vi ja, CM och utvecklingsprocessen och allt sånt finns med. Det finns i RUP men det är egentligen inget RUP-specifikt. Det RUPspecifika är ju projektdrivningen eller utvecklingsdelen i projektet så där är det ju inte alls RUP och det tror jag man tittade lite på. Vi har ingen egen modell. Vi håller på med en organisationsförändring nu då, och den är ju mycket att se till förvaltningen och hur vi hanterar krav och hur vi ner till arbetsprocesserna, hur vi hanterar dagliga störningar. Men det har inte så mycket med RUP att göra. Så RUP använder vi inte till förvaltningen. Men många delar som man använder i RUP använder vi också i förvaltningen. Alltså arbetsprocesserna.

Har ni någon organisationskarta?

För RUP? Nej, det tror jag inte.

Har ni något annat slags schema?

Du menar på arbetsstegen då? Jo, vi har ju RUP på vårt intranät. Som vi följer eller som man kan titta på. Och det vi har gjort där egentligen, om vi tittar på de här faserna i RUP, Inception, Construction, Elaboration som kommer innan, och transition så har vi ju knutit ihop PPSen till det här så vi har ju våra milstolpar knutna till dom så vi har ju DP2an ligger ju efter Inception då och DP3an ligger ju mellan Elaboration och Construction då. Så att dom slutliga besluten att köra vidare tas ju inte förrän DP3an då, så vi har ju PPS efter Elaboration då vi ska utrett eliminerat dom risker vi identifierat och liksom ha en så säker projektplan som möjligt. Så på det viset med tanke på att det är organisationen.

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Är det eget RUP som ni har gjort kartor för?

Nej, det är Rationals men om har gjort vissa justeringar och nu vet jag inte var precis för jag har ju inte studerat rationals RUP så mycket egentligen men jag tror att dom justeringar vi gjort är just dom delar som krockar med PPSen. Så har man antingen tagit bort PPSen eller RUP. Så vi har inte gjort något annat med formen. Men vi kan titta på det sen.

Kan du berätta lite mer om PPS?

Praktisk Projektstyrning, har du hört talas om det? PPSen är ju vårt projektstyrningsverktyg, det är ju jag som projektledare det är ju mitt styrmedel då. Och modellen är ju mer ett hjälpmedel att styra projektet, så att liksom jag har ryggen fri när jag driver mitt projekt. RUP behandlar ju inte hur jag styr projektet, utan det behandlar ju bara hur jag genomför min utvecklingsprocess, med lite risker. Så PPSen är en metodik som är framtagen av TietoEnator. Och den börjar egentligen vid, den beskriver olika milstolpar, DP1, där startar man projektet, DP8 är det sista där avslutas projektet. Och sen är väl DP3 också mandatory, och DP2 sen kan man använda andra delar. Så det är ju milstolpar där man ska ta beslut, stoppa eller fortsätta. Och vissa minimikriterier för att gå vidare. Och dom passar ganska bra ihop med, DP3an är en stor milstolpe, där har man gjort all analys, man har gjort tidplanerna ska vara framme och det passar ju ganska bra med den fasen som är efter Elaboration då, DP3an ligger ju efter Elaboration. Så det är ju där man tar beslut om man ska gå in i Construction för då bör man ju dra på med resurser och det börjar kosta pengar. Så PPSen är ett projektstyrningsverktyg medan RUP är mer utvecklingsmetodik.

Tycker du att det finns något som saknas när ni överför till förvaltning?

Ja, du menar i metodikerna? Vi kom ju aldrig så långt i det projektet vi körde där, men generellt är väl Transition den svagaste delen i hela RUP så jag tror säkert att det är mycket som saknas där men kanske inte i form av artefakter och så vidare utan det är mer organisatoriska saker. Och det vet jag inte ens om det är RUP som ska ta hand om. Jag tror det är viktigaste vid överföring är hur man hanterar sina resurser. Att man har resurser som är med i projektet så att man säkerställer att dom och att man använder folk som ska förvalta det i vissa faser av projektet. Det beskriver inte RUP. Jag tror inte att man kan följa RUP och sen har man två ton papper att lämna. Så man kan väl säga att RUP saknar lite den mjuka sidan. Hur man kompetensöverför. Nu kan jag inte Transition men som jag förstätt så behandlar inte den det så mycket. Så det stämmer nog att det är den svaga sidan.

Som ni då behandlar genom att föra in förvaltningsfolk tidigare?

Ja det anser jag.

Bilaga 4: Intervju med Anders på Alfa

Vad är din befattning, vad jobbar du med?

Jag är ansvarig för utvecklingsprocessen och utvecklingsmiljön, det vill säga alla verktyg som utvecklar användare och alla processer som utvecklar användare.

Användare som slutanvändare?

Ja du har ju inte, alltså på IT då, på IT internt, jag jobbar inte ut mot övriga Alfa. Mina användare är sådana som sitter och jobbar i projekt. IT-projekt då.

Vad gör du då? Utbildning?

Nej, alltså, jag har, det är väldigt stort egentligen, jag har, var ska jag börja någonstans. Utvecklingsprocesserna är ju liksom inte bara RUP, vi använder ju inte RUP egentligen. Utan vi har någonting som vi kallar för Alfa IT (namnet på deras process). Som är till stor del RUP sen har vi gjort anpassningar. Det är en del. Sen har vi då verktyg som stöder vårt arbete. Det är rena utvecklingsverktyg som dotnet, VSAD men vi har också ClearCase, ClearQuest såna mer processtödande mer som utvecklarna och andra använder för att arbeta. Dom producerar inte kod i ClearCase men dom lagrar den i ClearCase. Så vi har massa, massa verktyg då. Och allt det ansvarar jag för, att det hänger ihop, att vi releasar det till utvecklarna, att vi har support på verktygen. Det är en helt ny funktion, den har funnits se första januari.

I år?

Ja. Preliminärt, men det är egentligen nu första maj som den blir officiell och då blir min roll också officiell. Så utvecklingsprocessen är ju inte bara RUP, vi har ju för datawarehouseutveckling passar inte RUP riktigt. Så då kan man säga att vi byter ut någon disciplin mot något annat istället. Medan t.ex. projektledning, CM och Deployment är mer lika. Men exempelvis kanske kravhanteringen funkar på ett annat sätt. Use Case exempelvis funkar inte riktigt på Datawarehouse. Så att allt det här har jag som ansvar men sen är det så stor och så nytt allt det här så det har hänt så mycket under så många år så vi måste försöka konsolidera. För att få det att funka ihop. Och en del tycker att det är jobbigt och en del tycker att det är bra.

Hur länge har du jobbat på Alfa?

Tre och ett halvt år. Jag är egentligen projektledare från början och sen har jag halkat in på processarbete och fastnat där.

Har du varit med och gjort projekt efter det att ni införde RUP?

Nej, det har jag inte varit.

Vet du varför Alfa valde RUP?

Ja men då får du nog stänga av bandspelaren. Jag tror att det var nog ingen vetenskaplig undersökning, det bestämdes innan jag började. Så det var nog att man behövde en process och sen så vid den tiden var vi mindre mogna än vad vi är idag. Däremot inte sagt att vi är mogna idag men vi var mindre mogna då. Vilket gjorde att man visste inte vad man köpte för något riktigt. Och ja, så är det med det. Och vi använde inte verktygen på flera år egentligen det var först nu för knappt två år sedan vi började använda vissa verktyg. Men Alfa är ett (företag inom en bransch) och inte ett mjukvaruföretag och det syns.

De här checklistorna för projekten, är det du som har gjort dom?

Vilka checklistor menar du då?

För DP (visar exempel)?

Ja just det, nu ska vi se vilken version du har, ja, den här versionen har, jag har gjort dom äldsta versionerna faktiskt, dom här har ändrat sig ganska mycket på slutet. Förut var det mycket mer handson. Den här versionen som jag inte gjort har mer övergripande funktioner.

Hur gick det till att göra dom?

När jag gjorde dom från första början då, jag har ju jobbat med sånt här på andra företag så det var min erfarenhet av vad jag tyckte behövdes för jag såg att många projekt startades på för lösa grunder och man stämde inte av vissa saker i vissa lägen vilket gjorde att man fick bekymmer på slutet. Det var därför vi gjorde checklistorna. Checklistorna har inget direkt med RUP att göra. Vi har ett litet bekymmer här, får jag rita lite grann. Du har, RUP är en produktionsmodell, du är med på vad jag menar med produktionsmodell? Jag ser det som att sen har vi något som heter PPS, det är styrningen. Den modellen är egentligen lika för alla projekt, du kan bygga en bro och använda PPS, du kan bygga en stol med PPS, du kan bygga vad som helst. RUP är vår

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

produktionsmodell, den är anpassad för mjukvaruutveckling. Och i vissa lägen också en viss typ av mjukvaruutveckling. I här så finns det ju då, det finns projektledning här i också. Och här är också projektledning och det är inte alla projektledare som förstår att det finns på två ställen. Men DP-checklistan kommer fram här uppe. Så den ska man kunna använda på vilken typ av projekt som helst inte bara mjukvaruutvecklingsprojektet. Likadant om man liksom använder sig av, om man har ett projekt på driftsidan, för att t.ex. byta ut serverpark så kan man använda PPS men man använder inte RUP för det. DP-checklistan ska funka även för dom projekten. Så den kopplingen, tanken bakom dom är att dom här är ju inte direkt ihop med RUP utan ihop med PPS.

Hur ofta gör man om dom?

Det står i, faktiskt. Vi har tittat på att uppdatera dom igen för jag vill gå tillbaks och vara lite mer jordnära. Men anledningen till det här, alltså RUP, produktionsmodellen är jag ansvarig för, övre delen har någon annan varit ansvarig för och då tycker jag att det har dragit iväg i någon annan riktning i vissa lägen. Inte sämre, men på ett annat sätt. Han var väldigt duktig som har suttit där men nu får jag ta hand om alltihop igen.

Så du skulle vilja att det gick mer åt mjukvara?

Nej, inte mer mjukvara men däremot mer detaljerat, det finns exempelvis frågor här som kanske ska vara kvar men, exempelvis frågor som "Har Have differences in culture and geography been addressed?". Hur vet man att man har adresserat dom frågorna? Nja kanske va, men det ska vara ganska jordnära frågor som "Has the support of executive management been maintained?" ska man ha det i alla projekt? Behövs det alltid? När vi har så många projekt som vi har här ska "executive management" vara inblandade i alla projekt? Alltså det finns såna saker och så finns det såna saker som är kvar sen jag gjorde dom. "Are relations and dependencies to other projects defined and coordinated?" det tycker jag är rätt viktigt. Jag har ju äldre versioner som jag har gjort. Men vissa saker kanske vi skulle ta bort men vi lär oss ju, vi mognar ju. Dom uppdateras med jämna mellanrum.

Så det var bara efter erfarenheter, man hade inget sådant innan?

Nej.

Vad tycker du om checklistornas förmåga att fånga kunskap i projektet?

Syftet är ju inte att fånga kunskap i projekten med checklistorna utan syftet, det finns två syften med dom här. Mina två syften från början var två, tudelat. Det första var att många projektledare behöver lite hjälp med att det här ska jag tänka på när jag gör min plan och sen det andra syftet är att vi att misslyckas med en del projekt är ju inte bara projektledarens fel. Eller processens fel. En viktig spelare är styrgruppen och styrgruppen har inte alltid kunnat veta vad som förväntas av dom. Så tanken var då att styrgruppen ska kunna följa det här då vid sina styrgruppsmöten och kunna följa upp, finns det här med nu, alltså har man säkrat det här i projektet, om inte varför inte? Är det not applicable är det liksom inte gjort. Det var ett stöd för styrgruppen också att kunna arbeta med så att man skulle kunna säga att projektet kom till sin styrgrupp och sa att nu har vi gjort DP3. Här är de saker vi gjort. Här är informationen och här är den rapport som finns på det. Det var det som var tanken, stöd till projektledaren och stöd till styrgruppen.

Vad har dom som använder checklistorna, alltså svarar på frågorna för att göra detta vad har dom för kunskap eller kompetens?

Jag vet inte om jag kan svara på det riktigt. Det är ju, det har ingenting med RUP att göra egentligen utan det är ju kompetens på projektledare. Men det svåra här är, frågan du har i din uppsats är ju lite klurig för att en process bygger ju in kompetens i sig själv, man bygger på Best Practices och såna saker. Förmågan att hantera det här sen det är ju ofta individer. Och vilket gör att har du en process och du har en grupp nybörjare och du har en grupp experter som jobbar med samma process så kanske du får samma resultat men det kommer att ta mycket längre tid för dom som är oerfarna. Därför att man saknar erfarenheten, man gör allting By the Book. Dom som är erfarna kanske kan kasta bort, det här behövs inte, behöver inte det här för den här produkten. Men dom som är oerfarna gör allt. Det tar mycket längre tid. Likadant med handover, det gäller ju att ha med människor, processen hanterar ju inte hur man fasar in och ut människor och såna saker. Vad var din fråga igen?

Vad dom har för kompetens för att använda checklistorna?

Går vi tillbaks till det igen så är det egentligen mer hur mogna projektledare har vi. Så det är nog svårt att svara på det, om de har kompetens att fylla i det här, ja i vissa lägen ja, i andra lägen nej. Jag menar, har vi en projektledare som inte vet vad Configuration Management är så är det svårt att svara på en sån fråga. Sen kan man ju säga att i processen kan man läsa på om vad Configuration Management är för någonting men kan man utifrån att läsa det hantera och utan utbildning egentligen använda CM? Nej det kan man inte. Och det tror jag inte man kan göra med någon process utan det är ju erfarenhet.

Det finns ett minimum av uppgifter som man måste svara på?

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

I det här menar du? Det här används inte så mycket tror jag. Det finns en del projektledare som använder dom och Adam tror jag gör det. Han har du träffat va? Och han är en av våra mest erfarna projektledare och han tror jag tycker att det här är bra. Jag har fått den bilden av honom.

Ja, han sa att han tyckte att det var bra för att man inte skulle glömma någonting.

Ja, många använder ju inte det här.

Men det är fritt att göra som man vill?

Det på checklistorna har ju, det är inte fritt att göra hur man vill, det är det ju inte. Vi har ju liksom listor som säger att dom här dokumentet ska finnas framme vid vissa tillfällen. Det finns på vårt intranät. Så vi har liksom listor; det här ska finnas men problemet är att det är ett annat bekymmer vi har att processer är ju menar en process ser olika ut för varenda produkt egentligen. Om man bygger en central del i vårt affärssystem så kanske, som många system beror på om den går ner så kan vi inte sälja, då kostar det oss flera miljoner i timmen. Den kanske vi måste ha väldigt mycket kontroll på. Väldigt mycket dokumentation. En väldigt rigid hantering av helt enkelt, däremot har vi projekt för återlämning, returnerna i varuhuset, om det går ner kan man ju jobba med papper och penna istället, det kostar ju liksom ingenting. Kanske lite mer tid bara och där har vi liksom inga distinktioner hur alltså processen kan inte tala om i vilket fall som helst hur mycket man ska använda av den. Och det ska den inte göra heller. Utan det är en del av arbetet att jobba igenom det här Development Case och vad är det egentligen vi behöver för den här produkten hur viktigt är det. Och det kan inte jag sitta och bestämma. Och den kompetensen, dom sakerna är tvingade, development case måste finnas. Måste man fylla i dom här, idag så måste man inte det, dom här har försvunnit lite grann i hanteringen. Vi har så mycket checklistor som är en del av vi är omogna och så kommer alla på att vi måste kolla upp det och det och det och så slår det över åt andra hållet. Därför har vi inte slagit så hårt för dom här. Men jag tror att och i Adams fall så har vi uppfyllt ena syftet, projektledaren har använt det här för att ha någonting att arbeta utifrån men däremot kanske inte styrgruppen håller på att använda det här som följer upp det. Det här ett stöd, det är ingenting tvingande på så vis. Jag menar det här med "Have differences in geography, culture and organisation been addressed?" jag menar där, tvingande och tvingande, nej det är inte tvingande. Det var ett långt svar på... Men det är inte alldeles enkelt, det finns så mycket historik och så många dimensioner i det här.

Ger checklistorna något stöd för förvaltning?

Dom här checklistorna?

Ja, eller överhuvudtaget dom dokument som dom tar fram?

Det finns ju dokument för, du har egentligen tre överlämningar kan man säga. Överlämning till slutanvändare, det vill säga utbildning, överlämning till driften, det är också utbildning, installationsinstruktioner, övervakningsinstruktioner och så har vi överlämning till maintenance. För att kunna vidareutveckla det här och det finns det ju checklistor för och mallar för vad man ska ta med. Sen är det så att den som godkänner projektet, det är ju produktansvariga, ska vara, det funkar inte alltid så men den som har produktlivscykeln, en produkt lever så här länge, ett projekt lever så här länge (visar i luften) då måste man ta hänsyn till produkten. Det har vi varit dåliga på tidigare men de här ickefunktionella kraven börjar komma in tydligare och tydligare nu så att vi blir bättre på det här. Att hantera sånt som är viktigt för maintenance.

Dom dokument man tar fram, är det tillräckligt för att förvalta systemet?

Det kan inte jag svara på, det är nog olika från system till system.

Gör ni något mer för att stödja förvaltningsorganisationens kunskapsintag?

Inte från processidan, nej utan det är rätt mycket att det är uttalat att projekt ska ha, eller uttalat, jag har gjort såna saker faktiskt att vi har sagt att man får inte starta projekt om det inte finns en produktansvarig utpekad. Det vill säga att den produktansvarige är ansvarig för maintenance sen. Vi säger att man får inte starta projekt om man inte vet vem som ska ta emot det då. Det har hänt att man har startat projekt och sen när man är färdig så pekar man; nu får du ta hand om det här. Och då blir det inte bra. Så vi har från processidan gjort sådana saker. Ett kriterium för att starta projekt är att det ska finnas en produktansvarig.

Och det är tvingande?

Ja, det är tvingande, sen om det funkar i alla lägen det vet man inte men det är tvingande.

Du sa någonting om checklistor för maintenance...

Inte checklistor för maintenance på så vis utan

Eller för överlämning...

Jag tror jag har i alla fall en plan för det här med maintenance, jag tror den heter Product Handler men vi håller på att byta namn på den just nu. Men jag är lite osäker exakt vad status är.

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Och den innehåller allt det som dom behöver kunna?

Nej, jag har faktiskt inte jobbat med det där dokumentet själv så mycket men jag tror inte att man kan få ner kompetens i dokument utan det är mer att man ser till att lämna över saker, checklistor kanske på vad man ska lämna över för någonting. Det är svårt att få ner kompetenser om hur man gör saker och ting. Om man tänker sig att man driver projekt ett tag så har man lärt sig hur lång tid att göra vissa saker, att estimerar uppgiften. Hur får man över sånt i ett dokument till nästa pro... till dom som gör maintenance, det är rätt svårt. För att produkten är ju inte bara själva koden, att man kan systemet utan det är ju sånt som finns runtomkring också. Och sånt är nog svårt att dokumentera utan det är från person till person. Och det gör man ofta genom att se till att man har med folk ifrån maintenance i projekten.

Men det är funderat över i alla fall att det ska ge ett stöd för...

Ja, processen ger ju inte på så vis stöd, annat än den här checklistan då. Jag är lite osäker på dess status men intentionen är den i alla fall.

Hur många checklistor finns det egentligen?

Många. Det är så att det finns många som har utanför processarbete har tyckt att mitt område är viktigt, vi måste ha en checklista. Och så tvingar man på det på projektet och vi har ingen central uppskrivning av det. Därför har vi väldigt få checklistor på processsidan, vi försöker låta bli det. Det blir för mycket för projekten att göra.

När du säger process, vad syftar du på då?

Utvecklingsprocessen vi kallar inte RUP för RUP men jag använder det för att du ska veta.

Konsulteras du i projekt?

Ja det gör jag väl. Jag sitter med i vi har ju en Internet, de har väl identifierat en sjuttiosju projekt och jag sitter med i deras referens grupp för processarbete. Vi har ju en grupp på sju-åtta personer som utvecklar processen då och dom har ju identifierat supportaktiviteter som projekt kan beställa utav oss. Vi har väl en 25-30 olika aktiviteter som vi kan göra. Iterations assessments, fas assessments, use case workshops, vi har en lång lista på vad man kan beställa av oss. Vi supportar projekten, så att vi är inblandade ja. Sen har vi också ett utbildningspaket nu som vi kör för att folk ska få en inblick i vad en process är för något. Som är en tredagars övning där man jobbar med en faktisk produkt en delrobot där man utvecklar i java. Där man gör allting med, alltså man jobbar igenom ett helt projekt från initieringen till avslut. Med artefakter, uppdaterar tester, verktyg, clear case, clear quest. Och då testarna fyller i sakerna där. Så vi går igenom ett helt projekt från ax till limpa med verktyg och allting, dokumentation och sånt.

Är det det som ni kallar för bootcamp?

Ja.

Har ni alla olika rollerna med då eller?

Dom roller, vi kan väl säga då att vi har tagit vår process så har vi sagt att nu ska vi anpassa den till den här produkten, delroboten. Så att vi har ju inte med alla roller. Vi har med dom som behövs för den här storleken av projekt.

Vad brukar dom vanligaste problemen vara när du konsulteras?

Ja, vanligaste problemen, det beror på... projektledaren ser en typ av problem, jag ser en annan typ av problem. Kan jag tänka mig att... vet inte hur jag ska beskriva men kan väl säga att projektledaren ser att dels kan det vara svårt att se vägen igenom, hur ska jag göra här. Annat kan vara; måste jag göra det här? Den typen av problem. Och möjligtvis kan det vara att jag förstår inte men det är inte så ofta. Oftast är det; måste jag göra det här och hur ska jag ta mig runt det. Och för mig så är det att ja, det är svårt egentligen första gången helt enkelt, det är det som är problemet. Det är inget som slår erfarenhet. Har du gjort det en gång så är det mycket mycket enklare, så är det alltid. Sen om det är mjukvara eller någonting annat... så det är liksom vi har inte så många som är tillräckligt erfarna och så har vi inte det här interna sättet att lära oss av varandra. Så är det ju när man börjar med en ny process, skulle jag tro. Det kommer man ju åt genom att ta in mentorer då som hjälper personerna.

Väljer ni konsulter utifrån kunskap om RUP?

Både ja och nej, en del är här som RUP mentorer och sånt en del kanske är här för att de är duktiga på att hacka java. Så det beror på vilket syftet är.

Det var de frågor jag hade. Arbetet heter; Systemutvecklingsmetodens inbyggda förmåga att fånga och överföra kunskap från projekt till förvaltning. Har du någon fundering kring det?

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Nej som jag sa innan så tror jag inte det spelar någon roll vilken process du tittar på jag tror att det är få som bygga in sånt i processen tror jag är svårt, det har mer att göra med även om du försöker bygga in det så har det med människor att göra och så fort det har med människor att göra så är det deras förmåga att organisera sig och lösa uppgifter än processens inbyggda förmåga att göra sånt. Sen tror jag att det kan finnas olika avsikter, för mig har det hjälpt rätt mycket att vi gör dokumentation av produkter, vi har ju två typer av dokument, projektdokument och produktdokument. Produktdokument ska leva under hela produkten, det hör ju till det som ska lämnas över. Sen har kanske kompetensen att tyda det här och veta hur man ska agera i vissa lägen och sånt tror jag inte är processer, jag vet inte, jag har säkert fel, jag har säkert fel i något fall. Men det som det har mer att göra med individens förmåga att samarbeta exempelvis. Och är det en drivande produktansvarig som ser till att träffa olika och en teamchef som baktill ser till att personerna har rätt kompetens för jobbet så drar de individerna till rätt plats tror jag. Så att men som sagt jag är ingen expert på det här området, jag är mer en projektledare jag kan inte så mycket om use case modellering, jag inte alls inne på det. Mitt jobb är mer att se till att processen finns och fungerar.

Tycker du att den gör det?

Det är återigen det här med att det har mer att göra med den individ som använder den, kommer det hit en projektledare som är erfaren så tycker han säkert att det är omfattande. Vi har ju tagit fram det egentligen med tanke på att vi har ganska många oerfarna personer och vi har haft stora projekt så det blir stora grejer. En erfaren projektledare kan titta på materialet och skala bort det som inte behövs. Det är mycket svårare åt andra hållet. Så att ja, ur det perspektivet ja. Sen är vi långsamma just nu vi har rätt så mycket presentation att göra och det har att göra med att ... och det kommer att ta ett tag att göra.

Utbildar ni folk som sitter i förvaltningen också?

Ja det är ingen skillnad som jag ser det utan jag ser ingen större skillnad egentligen på processen i utveckling och maintenance. Man kan säga att maintenance är som korta projekt. Skillnaden är att man timeboxar mycket mer i maintenance. Att man liksom timeboxar, säg att vi har liksom tre veckor på oss här nu och man gör det man måste hinna med på tre veckor. Sen släpper man det. Nu säger jag inte att det är så jämt heller men det är mer jämna släpp och man lägger in det som måste vara med. Projekten är mer att man ska leverera en viss funktionalitet och kanske istället flytta på releasedatumet för att man ska få med allt det som måste vara med. Så att egentligen är det samma sak man gör här i utvecklingen och maintenance. Så att det finns saker som skiljer sig åt, vi har inom test har vi anpassad maintenance test process men i övrigt så har vi inga speciella anpassningar för maintenance. Nej jag vet inte om det vad finns det mer att säga. Det här har egentligen inte direkt med RUP egentligen att göra utan mer med vår styrmodell och sen har vi ju RUP-inbyggda milestone checklistor som du säkert känner till som vi också använder oss lite grann av och sen har vi även vissa QM-checklistor och QM är quality meeting som egentligen är till för att säkerställa att saker och ting är redo att sättas i drift. Och där sitter jag också med och styr det mötet. Så där ser vi också huruvida projekten förstår hur de ska göra.

Är det samma grupp för alla projekt?

Ja.

Mycket att göra då?

Ja. De har ett möte i veckan med varje projekt.

Visar hur långt de har kommit på de mötena?

Ja, de ska egentligen i dagsläget är det egentligen en massa dokument de ska komma in med men vi ska göra om det i nästa vecka. Den informationen behöver vi för att fatta beslut om det ska gå till drift eller inte. Så det ska bli informationsorienterat istället för dokumentorienterat. Och utifrån det ska vi sen göra riskbedömningar, vad är riskerna med det här så att säga.

Bilaga 5: Intervju med Bengt på Beta

Mitt exjobb handlar om RUP och kunskapsbortfall när man övergår från projekt till förvaltning. Så jag har några frågor om det. Vad är din befattning?

Jag är projektledare.

Hur länge har du jobbat som projektledare?

Det är ungefär tjugo år.

Hur länge har du jobbat med RUP?

Det har jag bara jobbat med sen i höstas. Jag är rätt så ny på RUP. Vi har jobbat lite grann med det tänket men inte just med RUP. Jag började med det när jag började här.

Har Beta jobbat med RUP länge innan?

Ja, men har ju, nu jobbar jag på en branding som heter Beta-B som består av projektledare och vi då har jobbat vi är rätt så nya har inte funnits mer än ett år men på den tiden så har vi jobbat med RUP. Men det var före min tid, jag började i november så jag är grön.

Vad gör Beta-B?

Alla dom här bolagen då jobbar som konsulter. Och det är ute på stora företag i Skåne i första hand, även i Danmark men första hand i Skåne.

Tar ni med er RUP då när ni går in i företagen?

Ja, eller rättare sagt om det företag som vi ska till använder RUP och det är rätt så många som gör det idag, då har vi den kunskapen. Så att vi har ju gjort det, är vi inne i några mindre uppdrag så har vi liksom fört in RUP eller presenterat RUP för dom. Liksom; det här är RUP och så här funkar det och det har de här möjligheterna och så vidare och så vidare.

Vad anser du om RUPs förmåga att fånga kunskap i projekt?

Fånga kunskap i projekt? Det låter svårt att ta på. Att fånga kunskap i projekt, alltså RUP hjälper ju egentligen inte till med det. RUP är en samling av vad man kallar för Best Practices. Och det är en enormt stor samling av Best Practices. Och alla de här Best Practices passar ju inte riktigt i alla projekt. Så därför gäller det ju att välja dom Best Practise när man startar ett projekt, att det är det vi ska jobba efter så här ska vi göra, vi ska använda om här dokumenten och de här arbetssätten för att nå våra mål i det här projektet. Så nackdelen med RUP är ju att man måste ha en väldigt god insikt i vad RUP är för någonting när man startar ett projekt. Och huruvida RUP är bra på att ta den här kunskapen som man tar till sig i projektet och föra det vidare det handlar om hur man utnyttjar den här dokumentationen och RUP slutar ju egentligen lite grann för tidigt. Du har ju bara överlämningen och sen slutar det upp tack och hej. Så har du ju en fas efteråt där som inte riktigt RUP tar hand om.

Tycker du artefakterna i RUP är bra för att överföra kunskap? Eller för att ta med sig information?

Ja, om man inte drunknar i det. Det är ju liksom, det var någon som sa att det var 11000 artefakter i RUP. Och då om man ser det på det sättet så finns det ju ingen möjlighet att få någon överblick över allt detta. Jag tror inte att RUP har ett dugg med detta att göra. Utan det har med individerna i projektet.

Hur tänker ni om det då, när ni går in och gör ett projekt, görs det något från er sida för att stödja förvaltningen?

Oftast går vi in i en väloljad organisation så visst vi försöker att men oftast finns de här sakerna redan på plats, vi har inte möjlighet att styra och förändra någonting. Man kan ju inte bara förändra på ett ställe utan det skulle få väldigt stora effekter i hela organisationen. Så vi har inte dom möjligheterna på det sättet, hade man suttit och jobbat i ett sånt företag som exempelvis Alfa, jag har jobbat på Alfa tidigare, då har man ju större möjlighet att påverka organisationen inifrån. Men som konsult så har vi inte den möjligheten riktigt.

I utvecklingsprojekt, finns det resurser med från förvaltningsorganisationen? Brukar det finnas det? I projekt, förbereds det för förvaltning?

Det ska ju finnas det. Men det är ju inte alltid det gör det, det är ofta man missar det och utvecklar någonting som man inte kan förvalta sen.

Har ni intern systemutveckling här?

Jo, mindre projekt.

Hur fungerar det då med överlämning?

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Oftast dom projekt som jag varit med på är att vi tar fram en liten produkt som ska presenteras för att sedan vidareutvecklas så det är inga stora produkter utan det är bara en produkt som inte ska förvaltas. Utan den slängs och så nästa så typ. Men alltså det är ju mer dokumentation, att beskriva hur programmet ser ut hur det är tänkt att det ska fungera naturligtvis och kanske lite systemdokumentation till det och funktionsbeskrivningar.

Det är det som används sen? Är det nog?

Nej inte om vi pratar om att vi skulle göra något som är större intern, då får vi nog se över rutiner så att säga. Det är det här med skomakarens skor så är det här.

Vad är de största problemen när företag ska använda RUP?

Att man inte gör en... ser på varje projekt som att det är unikt, att man inte gör en Development Case som är det första man gör så går man igenom projektet. Det här är projektet, vad ska vi göra i det här projektet, vad är det då vi behöver i RUP, vi kan ju inte ta allt. Det finns ju inte, det är alldeles för mycket. Vi får plocka ut dom bitarna som vi behöver i RUP och använda det i det här projektet. Det är ju liksom ett unikt projekt. Annars så tar man har man gjort ett paket, har man gjort Development Case redan för hela företaget kan säga såhär jobbar vi med RUP, det funkar för alla projekt, smack. Ja då blir det liksom, då använder man inte RUP då använder man någon egen lösning. Det kan vara bra det också men det är inte RUP. Just den där fasen i början på projektet när man går igenom Development Case, det är väldigt viktigt att man gör det. Att man väljer ut rätt delar av RUP, det är gigantiskt, alltså åt helvete för stort egentligen. Det är rätt vansinnigt men det är ju så. Någon har gjort någonting som sagt; jag gjorde så här och det blev skitbra resultat och då dokumenterar vi det och så kommer nästa jag gjorde så här och det blev också jättebra och då dokumenteras det också. Och sen blir det massor med människor som har dom här bra erfarenheterna som är jättebra. Det blir ju samma sak som visst glass är gott, senap är gott på korven men är senap gott på glass? Nej, utan det är så att man måste se till varje enskilt och anpassa det. Det tror jag är den största missen och det som är svårast med RUP.

Har ni konsulter med i förvaltning?

Det gör vi väl, vi har ju folk som sitter i någon form av linjeorganisation och dom jobbar inte som projektledare utan det är ju utvecklare och sen riktigt hur det funkar... RUP är ju egentligen bara till för att ta fram någonting från ett vitt papper. Det är ju också en nackdel med RUP om man redan har någonting som är färdigt och ska vidareutveckla det vad ska man då använda.

Varför tycker du att företag ska välja RUP som sin systemutvecklingsmodell?

Det finns en del som är bra, det är ju det här iterativa det finns ju saker som är dåligt, det står exempelvis ingenting om vad en styrgrupp är i RUP. Det är jätteviktigt med styrgruppen, har du ingen styrgrupp kan du lägga ner projektet. Det är ingen som äger själva arbetet så det räcker inte bara med RUP. Och har du bra kunskap i RUP så funkar det. Har du inte det så är det inte lika bra längre. Och andra sidan, det är fullt med Best Practise. Jag vet inte riktigt om jag svara varför jag tycker företag ska välja RUP, det är bra att ha en metod att falla tillbaka på. Om man jobbat ett tag som projektledare, man måste gå igenom vissa faser när du vidareutvecklar en produkt, det är rätt naturligt, om man aldrig gjort det förut så kan man titta på hur har dom gjort det. Det är precis som att ha en mentor, som man frågar, hur gjorde du när du kom i den här situationen och istället för att ha en mentor så kan man titta i en bok och säga, ja, han gjorde så där. Så det är ju bra att man har, har man en metod så underlättar det för de här begreppen som man använder, när man jobbat in de här begreppen så vet man vad i pratar om, säger man Development Case eller artefakt så förstår man och så vidare. Det är ju det som är bra. Och det behöver ju inte, det kan ju likaväl vara en annan metod men RUP är ju liksom det är ju väldigt utbrett, i alla fall i Sverige så har det fått bra genomslagskraft, så det är väldigt många som kan det, vilket innebär att det är lätt att få tag på, att anställa nytt folk. Att lära upp folk det får positiva följder.

Har du någon mer fundering kring kunskapsöverföring?

Nej, det är bara det att RUP löser inte det problemet.

Bilaga 6: Intervju med Göran på Gamma

Du kan gärna berätta om dig själv.

Just nu, vi är ett konsultföretag så uppdragen varierar ju, men just nu så är jag projektledare i ett projekt med tjugofem man på Alfa. Alfas IT-bolag i (en stad) och där används ju RUP då som utvecklingsmodell, generellt på Alfa så att säga. Så jag har varit med i projekt tidigare för att anpassa RUP till Alfas verksamhet och försöka implementera det i verksamheten. Så nu praktiserar jag det nu i ett projekt där. Så att innan jag var på Alfa så har jag varit bland annat på (andra stora svenska företag) för att införa RUP och delar av RUP i deras verksamhet.

Tycker du att det fungerar bra för Alfa med RUP?

Ja, det gör det väl. Ja, nej, det fungerar väl bättre än vad det fungerade innan RUP men det finns fortfarande mycket som kan förbättras. Runtomkring egentligen och som kanske inte relaterar till RUP utan mer med hur man hanterar omvärlden runtomkring.

Alfa är mina huvudundersökningsobjekt. Hur länge har du jobbat med RUP?

Jag har jobbat med RUP sen 1998-99.

Så Gamma är konsulter, ni håller på med systemutveckling?

Ja, vi har två huvudaffärsområden, det ena kallar vi Mobil & Multimedia, de utvecklar mobila lösningar, mobila tjänster av olika slag. Jobbar med att utveckla delar, som multimedia delar av telefoner och sånt, hos Ericsson till exempel. Sen jobbar vi med det vi kallar för Software Engineering som vi håller på att vidga till Software Business Management. Och det är där vi jobbar med metoder och för att jobba effektivare i utvecklingsprojekten. Så det är där RUP kommer in.

Använder ni RUP i alla era utvecklingsprojekt?

Nej, det gör vi inte. I kundprojekten är det ju styrt av vilka processer kunden använder, när vi har möjlighet att välja så använder vi det.

Har du utbildat inom RUP?

Ja jag har alltså vi har ju ett partnerskap, vi är partner med IBM. Vi har varit partner med Rational då från början som IBM har köpt så att vi har ett antal personer, bland annat jag som är certifierade användare av delar eller hela RUP. Vi har en kille som är specialiserad på kravhantering som har utbildning specifikt om det och sen har vi en kille som är specialiserad på Configuration and Change Management och jag är lite mer övergripande, jag håller i generalistutbildningarna.

Har du gjort det på Alfa med?

Ja.

Är det svårt att få grepp om RUP?

Det är, ja det är det ju. RUP är en väldig kunskapskälla egentligen, som väldigt få använder alla delar av. Det är enorm mängd dokumentation som finns tillgänglig då det finns väl inte någon som kan få grepp om allt. Inte på detaljnivå i alla fall, övergripande är det väl inte så svårt att få grepp om tycker jag. Om grundtankarna och grundidéerna om hur man jobbar för att det ska fungera bra.

Mitt arbete handlar om Systemutvecklingsmetodens inbyggda förmåga att fånga kunskap och överföra den till förvaltningsfasen? Vad anser du om RUPs förmåga att fånga kunskap i projekt?

Ja alltså förmågan att fånga kunskapen är väl ganska bra, det vill säga RUP har ju om man jämför med, om man tittar på Alfa till exempel då så fångar den ju kunskap på ett mycket bättre sätt än vad man gjorde tidigare i form av att man verkligen dokumenterar kraven på systemet på ett strukturerat sätt som man kan lämna över den kunskapen till de som ska underhålla systemet och man fångar designen på systemet så att man kan lämna över det. Istället för att man bara lämnar koden och måste läsa koden för att förstå vad systemet gör som det har sett ut tidigare. Så det är klart att det fångar kunskapen mycket bättre med RUP innehåller ju inte så mycket konkreta aktiviteter som man att genomföra för att föra över kunskapen, alltså det är en sak att man dokumenterar och har en hög papper i princip att lämna över och det är en annan sak att man verkligen också lämnar över kunskapen från person till person. Inte bara en hög med papper utan försöker genomföra aktiviteter på något strukturerat sätt så att man tar till sig den information man får det innehåller inte RUP något specifikt kring. Så på Alfa har man lagt till lite egna rutiner för hur man genomför en handover från utvecklingsprojektet till underhållsverksamheten

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

egentligen som då ska se till att försöka få in resurser från underhållsverksamheten till att jobba i projektet under en period för att på det sättet lära sig och ta med sig kunskapen mer handfast in i underhållsvärlden sen.

Vad är dom största problemen för företagen när de ska använda RUP?

Ett problem som jag upplever är att man, att det är svårt att lära om folks tankesätt, alltså RUP fungerar ju lite annorlunda eftersom det är iterativt. Det är lite annat tankesätt mot vad folk är vana vid. Så det är svårt att lära om folk i de nya tankarna, till exempel att man ska skapa insikt om att man kan inte redan efter ett par veckor in i ett projekt leverera en trovärdig tidpunkt för när man ska vara färdig. Om ett projekt har en tänkbar löptid på ett år eller ett och ett halvt år eller en trovärdig budget utan man delar upp påsen pengar så att säga som man lämnar till projektet i olika mindre pottor. Man får en viss pott för att göra ett visst arbete och efter det så kan man göra bättre uppskattningar om vad man kommer att hamna och då får man en ny påse pengar för den uppskattningen. Det är väldigt vanligt att folk tar alltid eller har man i projekt när man börjar då med det inledande analysarbetet tar reda på vad man ska göra i det här projektet och såhär ser den övergripande kravbilden ut så måste man ju också naturligtvis fundera på hur lång tid kommer det här att ta utveckla. Men det estimat man ger då är väldigt grovt och inte särskilt tillförlitligt men så fort man har sagt ett datum då till exempel, vi tror att vi är färdiga i december 2005 så tar alla det som en sanning. Och kommer man sen och presenterar ett annat datum lite senare för att då har man jobbat vidare och kommit till bättre insikt så är man liksom ett misslyckande. Det är inte så, RUP är inte tänkt, RUP är liksom tänkt att skapa den insikten att vi måste jobba med vissa saker först och då ger det underlag för att skapa bättre precision i vår estimat. Vi är medvetna att först lägger vi en miljon på att ta reda på mer så att vi kan säga sen hur mycket resten kostar då. Istället för att från början säga att vi behöver tio miljoner. Mer problem när man använder RUP är väl att man ibland fokuserar för mycket på dokument, vad är det för dokument vi ska skriva vad är det för mallar, hur ser mallarna ut, vad är det för rubriker. Det är ju mindre intressant, det intressanta med RUP är ju, det är ju intressant att man dokumenterar saker på ett strukturerat sätt men det mest intressanta är ju vad man genomför vad man gör för aktiviteter. Och resultatet av dom det, visst, dokumenterar man på ett visst sätt. Det är allt för lätt att dyka in i dokument och vad är det för dokument vi ska producera. Det är inte det som är ändamålet i sig utan ändamålet är att genomföra en aktivitet på ett bra sätt och sen dokumentera den.

Varför tycker du att företag ska välja RUP som systemutvecklingsmetod?

Därför att RUP innehåller ju väldigt mycket bra idéer om hur man exempelvis hanterar sina krav och hur man låter kraven styra resten av arbetet. Och det är ofta ett av de stora problem eller ett av de problem som alla som arbetar med mjukvara har att de kanske inte tillräckligt väl hanterar sina krav och hanterar ändringar av krav på ett strukturerat sätt. Plus att RUP också fokuserar mycket på att beta av risker så tidigt som möjligt istället för som man är van vid eller som människor ofta gör är att man väntar med det svåra till sist. Man gör det enkla först och väntar med det svåra till sist, RUP försöker vända på det. Kanske göra det absolut svåraste först då vet vi att vi har klarat det då kan vi också veta hur väl vi klarar resten och ge då trovärdighet i estimaten istället för att göra något lite enkelt först och testa och det klarar vi och så sparar vi det svåra till sist.

Artefakterna i sig, de kan inte användas för kunskap?

Jo det kan de naturligtvis. Det problemet är att det räcker inte bara med artefakterna i någon mening det är som jag sa aktiviteterna som man genomför som är det viktiga om man sitter med artefakterna och så har man ingen aning om hur, vad är det som man gör för att komma till de här artefakterna, det måste man ju också känna till när man underhåller. Det är det som är mer man till man överföringen, antingen måste man få den kunskapen genom kurser eller nånting, hur man konkret jobbar med olika saker som resulterar i en viss artefakt eller så kan man få den då genom att jobba med i projektet.

Men när ni gör projekt i andra företag, går ni in med rekommendationen att de ska ha med som Alfa resurser från projektet i förvaltningen?

Ja, precis.

Har ni interna systemutvecklingsprojekt här?

Ja det har vi faktiskt lite grann också.

Hur fungerar det då?

Där är det så, vi har inte haft så jättemånga, men där är det så att det är samma resurser som fortsätter som när projektet avslutas så blir det samma resurser som fortsätter in i och förvaltar det sen. Där blir det ingen överlämning, att man ska lämna över till några andra. Det är klart det kan ju vara ett jättebra alternativ naturligtvis men så är det inte överallt utan där har man ju, på Alfa har man ju en viss underhållsorganisation som ska ta emot och dom som jobbar i projektet, när projektet är slut så går de in i något annat projekt förhoppningsvis. Så då har man överlämning.

Borde det inte vara så ganska ofta eftersom det är mycket konsultfolk som tas in?

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Jo jo det är ett vanligt problem men själva har vi inte haft det så.

Har du några mer funderingar över det här med kunskapsöverföring från projekt till förvaltning?

Ja det är klart att det är ju också, om du tittar på en sådan organisation som Alfa där man är ganska nya så, nya med at använda RUP då, så är det viktigt naturligtvis, det här med kunskapsöverföring på något sätt det blir ju viktigt för om det nu är projektet som ansvarar för det eller på någon högre ort men att man inte bara tar in resurser i projektet under en viss tid för att lära upp dom utan att man också då kan presentera; vad är det för profil som behövs av resurserna för att de ska klara av det. Vad behöver man för kunskaper eller för erfarenheter för att kunna skriva bra use cases till exempel. Och det var väl lite det jag menade, att det räcker inte bara med att få dokumentationen i handen utan man måste kanske då planera för att de som idag sitter och gör maintenance med gamla system att dom kanske ska gå lite kurser och annat för att lära upp sig.

Börjar det fungera med den delen på Alfa?

Ja... jag tycker att det fungerar om man tittar på det projektet som jag är projektledare för så har vi ganska bra dialog med underhållsorganisationen, vi för in ett par tre stycken därifrån som kommer att jobba i projektet under ett antal månader, konkret med olika delar. På det sättet lära upp sig, lära sig produkten som de sen ska förvalta. Och vi har tagit fram en plan från projektet där vi talar om, där vi då bedömer att vi behöver den här och den här typen av resurser med den här och den här kompetensen för att kunna underhålla systemet. Så försöker dom, håller dom på att rekrytera in mer personal och försöker använda det underlaget för att hitta rätt profil.

Bilaga 7: Roller inom RUP

Tabell i Bilaga 6: Ska visa på samband mellan de olika aktiviteterna och de olika rollerna (Lunell, 2003)

Projektroller	Sponsor/beställare Projektledare Utvecklare Användare Modelleringsledare Mentor
Konfigurations- och ändringshantering	Konfigurationsansvarig Ändringsansvarig Integratör
Utvecklingsmiljö	Processingenjör Verktygsspecialist Systemadministratör Systemanalytiker Användargränssnittsdesigner Arkitekt Testdesigner Teknisk skribent Verksamhetsanalytiker
Verksamhetsmodellering	Verksamhetsanalytiker Verksamhetsutvecklare Verksamhetsmodellgranskare <i>Kund</i> <i>Användare</i>
Krav	Systemanalytiker Kravspecificerare Användargränssnittsdesigner Arkitekt Kravgranskare <i>Kund</i> <i>Intressent</i> <i>Användare</i>
Analys och design	Arkitekt Arkitekturgranskare Designer Kapseldesigner Databasdesigner Designgranskare
Implementation	Arkitekt Implementatör Kodgranskare Integratör
Test	Testledare Testanalytiker Testdesigner Testare
Driftsättning	Driftsättningsansvarig Illustratör Teknisk skribent Implementatör Kursutvecklare <i>Kund</i>

Bilaga 8: Artefakter inom RUP

Tabell i Bilaga 7: Samband mellan aktiviteterna och artefakterna i RUP (Lunell, 2003)

Projektledning	Affärsnytta Granskningsprotokoll Iterationsplan Iterationsutvärdering Risklista Statusutvärdering Utvecklingsplan Åtgärdslista Arbetsorder Kvalitetssäkringsplan Mättningsplan Problemlösningsplan Produktgodkännandeplan Projektmätningar Riskhanteringsplan
Konfigurations- och ändringshantering	Arbetsarea Projektdatabas Ändringsbegäran Konfigurationshanteringsplan Konfigurationsrevisionsresultat
Utvecklingsmiljö	Infrastruktur till utvecklingens artefakter <ul style="list-style-type: none"> • Infrastruktur till utvecklingen • Verktyg Utvecklingsfall Projektspecifika mallar Riktlinjer för användargränssnitt Riktlinjer för användningsfallsmodellering Riktlinjer för design Riktlinjer för programmering Riktlinjer för test Riktlinjer för verksamhetsmodellering Riktlinjer för verktyg Stilregler för manualer Utvärdering av utvecklingsorganisationen
Verksamhetsmodellering	Användningsfallsmodell för verksamhetens artefakter <ul style="list-style-type: none"> • Användningsfall för verksamheten • Användningsfallsmodell för verksamheten • Verksamhetsaktör Kompletterande specifikation för verksamheten Målorganisationsutvärdering Objektmodell för verksamhetens artefakter <ul style="list-style-type: none"> • Användningsfallsrealisering för verksamheten • Objektmodell för verksamheten • Organisationsenhet • Verksamhetsobjekt • Verksamhetsuppgift Ordlista för verksamheten Verksamhetsarkitekturdokument Verksamhetsregler Verksamhetsvision
Krav	Användargränssnittsprototyp

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
 -En fallstudie av RUP
 Carin Olsson

	<p>Användningsfallsmodellens artefakter</p> <ul style="list-style-type: none"> • Aktör (mänsklig, system) • Användningsfall • Användningsfallsmodell • Användningsfallspaket <p>Intressentönskemål Kompletterande specifikation Ordlista Vision <i>Användningsfallsskiss</i> <i>Gränssnittsklass</i> <i>Kravattribut</i> <i>Kravhanteringsplan</i> <i>Kravspecifikation för programvara</i></p>
<p>Analys och design</p>	<p>Användningsfallsrealisering Arkitekturbeskrivning Datamodell Designmodellens artefakter</p> <ul style="list-style-type: none"> • Användningsfallsrealisering • Designsystem • Designklass • Designmodell • Designpaket • Gränssnitt • Händelse • Kapsel • Protokoll • Signal <p><i>Analysmodellens artefakter</i></p> <ul style="list-style-type: none"> • <i>Analysklass</i> • <i>Analysmodell</i> <p><i>Arkitektoniskt provskott</i> <i>Driftsättningsmodell</i> <i>Referensarkitektur</i></p>
<p>Implementation</p>	<p>Bygge Implementationsmodellens artefakter</p> <ul style="list-style-type: none"> • Implementations delsystem • Implementationsmodell • Komponent <p><i>Integrationsplan</i></p>
<p>Test</p>	<p>Testidélista Testplan Testresultat Testutvärderingssammanfattning <i>Testdefinition</i></p> <ul style="list-style-type: none"> • <i>Belastningsmodell</i> • <i>Testdata</i> • <i>Testfall</i> <p><i>Testimplementation</i></p> <ul style="list-style-type: none"> • <i>Testlogg</i> • <i>Testmiljökonfiguration</i> • <i>Testskript</i> • <i>Testsvit</i> <p><i>Testprogramdesign</i></p> <ul style="list-style-type: none"> • <i>Arkitektur för automatiserad test</i> • <i>Gränssnitts specifikation för test</i> • <i>Testklass</i> • <i>Testkomponent</i>

En systemutvecklingsmetods inbyggda förmåga att fånga och överföra kunskap
-En fallstudie av RUP
Carin Olsson

Driftsättning	Driftsättningsplan Produktartefakter <ul style="list-style-type: none">• Driftsättningsenhet• Installationsartefakter• Materiallista• Produkt• Produktgrafik Stödmaterialartefakter för användare <ul style="list-style-type: none">• Stödmaterial för användare• Utbildningsmaterial• Utgåveinformation
---------------	---

7 Referenser

- Andersen, Erling S. (1994) *Systemutveckling - principer, metoder och tekniker*. Studentlitteratur, Lund
- Backlund, Peter, (2004), *Development Process Knowledge Transfer through Method Adaptation, Implementation and Use*, Stockholms Universitet, Akademityck AB, Edsbruk
- Gilovich T, (1991), *How We Know What Isn't So – The Fallibility of Human Reason in Everyday Life*, The Free Press
- Henderson-Sellers B.; Due R.; Graham I.; Collins G., (2000), *Third generation OO processes: a critique of RUP and OPEN from a project management perspective*, pages: 428-435, Software Engineering Conference, APSEC 2000, IEEE
- Huberman, Michael A & Miles, Matthew B, (1994), *An Expanded Sourcebook- Qualitative Data Analysis*, SAGE Publications, London
- Kvale, Steinar,(1997), *Den kvalitativa forskningsintervjun*, Studentlitteratur, Lund
- Lunell, Hans, (2003), *Fyra rundor med RUP*, Studentlitteratur, Lund
- Meggerle, Theis; Steen, Odd, (2002), *IT-kvalitet i praxis – systemutvecklarens kunskap om och syn på kvalitet*, Institutionen för Informatik, Lunds Universitet
- Nordström, Malin; Welander, Tommy, (2002), *Affärsmässig förvaltningsstyrning – en referensmodell för (system-)förvaltning*, Studentlitteratur, Lund
- Rolf, Bertil, (1981), *Kunskapskritik och Kunskapsgenetik – Kunskapsöverföringens grunder*, Filosofiska Institutionen, Lunds Universitet
- Strand, Lotta, (2001), *UML & RUP – Att lyckas med OO-projekt*, Docendo Sverige AB, Stockholm
- TietoEnator, *Kort om PPS – PraktiskProjektstyrning, Modellbeskrivning och PPS online*.
www.tietoenator.se/default.asp?path=485;493;16112;13813;13814;13929;14623 , 2005-05-12
- Wallén, Göran, (1996), *Vetenskapsteori och forskningsmetodik*, Studentlitteratur, Lund
- Wiktorin, Lars, (2003), *Systemutveckling på 2000-talet*, Studentlitteratur, Lund
- Yin, Robert, (2003), *Case Study Research, Design and Methods*, SAGE