

Datavetenskap

Stefan Nilsson, Thomas Ångman

**Utvärdering av kontraktsprogrammering i ett
industriprojekt**

Examensarbete, C-nivå

2002:10

Utvärdering av kontraktsprogrammering i ett industriprojekt

Stefan Nilsson, Thomas Ångman

Denna rapport är skriven som en del av det arbete som krävs för att erhålla en kandidatexamen i datavetenskap. Allt material i denna rapport, vilket inte är mitt eget, har blivit tydligt identifierat och inget material är inkluderat som tidigare använts för erhållande av annan examen.

Stefan Nilsson, Thomas Ångman

Godkänd, 2002-06-04

Handledare: Eivind J. Nordby

Examinator: Martin Blom

Sammanfattning

Denna rapport bygger på en utvärdering av att tillämpa en metod för kontraktsprogrammering i ett industriprojekt. Utvärderingens fokus var att undersöka om metoden påverkar antalet fel respektive tidsåtgång vid programutveckling. Det undersökta projektet genomfördes hos Ericsson Infotech AB och var redan slutfört när vi blev inblandade. I projektet skapades tjänster för virtuella privata telenät. Materialet som fanns att tillgå bestod av fel- och tidsrapporter rutinmässigt registrerade under projektets gång. Inga ytterligare tillägg hade gjorts till dessa normala registreringar. Undersökningen mynnar inte ut i några statistiskt verifierbara slutsatser då antalet mätvärden från projektet var få och mätresultaten inte kunde ge någon acceptabel signifikansnivå. För att kunna dra några relevanta och statistiskt verifierbara slutsatser utifrån det undersökta materialet behöver ytterligare efterforskningar göras där djupare dykningar i materialet måste göras. Detta ger uppslag för vidare forskning.

Evaluation of programming by contract in an industrial project

Abstract

This report is based on an evaluation of a method for programming by contract in an industrial project. The focus of the evaluation was to investigate whether the method affects the number of faults respective the time needed for software development. The project that was investigated was carried out on Ericsson Infotech AB and was already brought to a conclusion when we were involved. In the project, features for virtual private telephone networks were created. The available collection of material contained the ordinary trouble and time reports that were registered during the project. No further supplements had been done to these normal registrations. The investigation doesn't end in any statistical significant conclusions, due to the low number of test data points and the low significance in the statistical tests. To be able to draw any relevant and statistical significant conclusions from the investigated collection of material, further research has to be done. This gives ideas for further research.

Tack

Vi vill tacka vår handledare och uppdragsgivare Eivind J. Nordby för all hjälp och nedlagd tid, utan hans hjälp hade vårt arbete aldrig kunnat slutföras. Vi vill också tacka Martin Blom som hjälpt oss när Eivind inte varit tillgänglig. Ingemar Adolfsson på institutionen för informationsteknologi/statistik tackar vi för hans hjälp med statistiska frågor. Ett tack riktas även till våra kontaktpersoner Mats Larsaeus och Jan Mölder på Ericsson Infotech för deras hjälp med fel och tidsrapporter samt för att de gett oss en inblick i Ericsson arbete. Slutligen vill vi även tacka Susanne Glifberg på Ericsson Infotech som tillsammans med Eivind lade grunden inför våra kontakter och vårt samarbete med personal på Ericsson Infotech.

Innehållsförteckning

1	Inledning	1
1.1	Problemformulering.....	1
1.2	Mål.....	1
1.3	Rapportens struktur.....	1
2	Begrepp	2
2.1	Rational Unified Process	2
2.2	Semla	3
2.3	Konstruktion och tillverkning.....	4
3	Strukturer i dataunderlag	5
3.1	Felrapport.....	5
3.2	Tidsrapport.....	5
4	Analysplanering	6
4.1	Förutsättningar.....	6
4.2	Identifiering av variabler	7
4.3	Hypotesformulering.....	8
4.3.1	Hypotes 1 – Antal fel	
4.3.2	Hypotes 2 – Tidsåtgång	
5	Bearbetning av mätdata	9
5.1	Förberedelser	9
5.2	Bearbetning av mätdata från felrapporter	10
5.3	Bearbetning av mätdata från tidsrapporter	10
6	Redovisning av resultat från bearbetning av mätdata	11
6.1	Undersökning av fel.....	11
6.2	Undersökning av nedlagd tid.....	12

7	Analys	14
7.1	Hypotesprövning	14
7.2	Indikationer.....	15
7.3	Validitet	16
8	Slutsatser	17
9	Erfarenheter	17
9.1	Problem.....	18
9.2	Egna reflektioner	18
	Referenser	20
A	Felrapportsmall för Skywalker VPN	21
B	De olika aktiviteterna i programutvecklingen enligt RUP	24
C	Redovisad tid inklusive personer utan grupptillhörighet	25
D	Förkortningar och begrepp	26

Figurförteckning

Figur 2.1.1: Tidsfördelning mellan faser och arbetsflöden enligt RUP	3
Figur 2.3.1: Arbetsmoment som ingår i KoT	4
Figur 3.2.1: Projektets aktivitetsredovisning jämfört med RUP.....	6
Figur 4.2.1: Omfattning av Total tid och KoT tid i relation till RUP	8
Figur 6.2.1: Fördelning av redovisad tid per grupp	13
Figur 6.2.2: Fördelning av redovisad tid för Semla- respektive icke-Semlagrupper	13
Figur 7.1.1: Rangordningen som ger tydligast utslag	15
Figur 7.1.2: Rangordningen för aktuella mätvärden	15
Figur B.1: Redovisad tid per grupp inklusive personer utan grupptillhörighet	25

Tabellförteckning

Tabell 6.1.1: Fördelning av fel mellan grupperna och inom respektive grupp.....	11
Tabell 6.1.2: Felfördelning uppdelad på klassificeringsstatus.....	12
Tabell 6.2.1: Fördelning av tid mellan grupperna och inom respektive grupp.....	14

1 Inledning

Denna rapport är en redovisning av ett examensarbete på c-nivå i datavetenskap vid Karlstads universitet. I detta examensarbete utfördes en undersökning för "Software Engineering Research Group" (SERG) som är en forskningsgrupp på institutionen för datavetenskap vid Karlstads universitet. Deras projekt "Improved software quality through semantic descriptions" (Skutt), som nyss har avslutats, hade som mål att utveckla en "metod som genom en förbättrad semantisk beskrivning bidrar till förhöjd programvarukvalitet i komplexa system". Skutt stöddes av "Närings- och teknikutvecklingsverket" (NUTEK) och gjordes i samarbete med Ericsson Infotech AB (EIN). Vid första fasen i Skutt framtog programutvecklingsmetoden Semla [1] som fokuserar på semantik. Metoden innebär att "kontrakt" skrivs mellan mjukvarumodulerna i ett system med förvillkor, eftervillkor och invarianter. I vår undersökning avspeglas resultatet av att Skutt tillämpades i ett skarpt industriprojekt, Skywalker VPN hos EIN under år 2000.

I projektet Skywalker VPN skapades funktioner och tjänster för virtuella telefonnät där abonnenter såväl inom fasta som mobila nät sammankopplas så att de alla upplevs ingå i ett internt telefonnät trots att de kan vara spridda runt hela världen.

1.1 Problemformulering

Vi skulle utvärdera resultatet av att tillämpa Semla vid programvaruutveckling i delar av industriprojektet Skywalker VPN hos EIN.

1.2 Mål

Målet med utvärderingen var att undersöka om Semla reducerade antal konstruktion och tillverkningsfel samt om Semla minskade tidsåtgången för konstruktion och tillverkning i det aktuella projektet. Resultaten från de grupper och individer som använde Semla i projektet skulle jämföras med motsvarande resultat från de som inte gjorde det.

1.3 Rapportens struktur

Kapitel 2 lägger en grund för ökad förståelse genom att presentera viktiga begrepp. Här presenteras programutvecklingsprocessen som användes i projektet Skywalker,

programutvecklingsmetoden som skulle utvärderas i denna undersökning samt kopplingen mellan dessa. Kapitel 3 ger läsaren en inblick i strukturen och innehållet i dataunderlaget som erhöles inför undersökningen. Här presenteras fel- och tidsrapporter som fanns att tillgå från projektet.

Kapitel 4 beskriver vår plan, att utifrån de förutsättningar vi hade inför vår undersökning, nå det formulerade målet. Här sker identifiering av variabler och hypoteser formuleras kring dessa variabler. I kapitel 5 beskrivs arbetet med att få fram de variabler som deklarerades i vår plan.

I kapitel 6 redovisas resultatet som vi erhölet från identifierade variabler i dataunderlaget. Detta redovisas i form av tabeller och diagram. I kapitel 7 undersöks möjligheten att nå tidigare beskrivet mål. Här analyseras erhölet resultat, möjligheten att förkasta formulerade nollhypoteser samt validiteten i undersökningen.

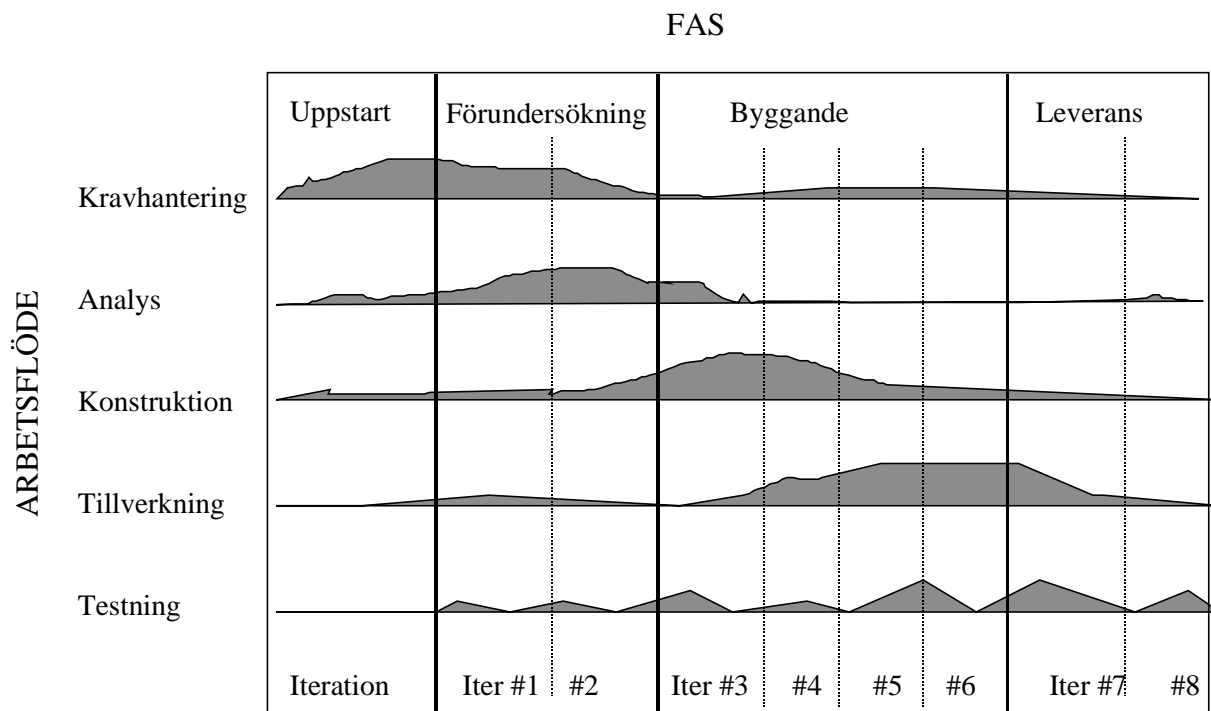
Kapitel 8 presenterar slutsatser utifrån analysen och undersökningen. Kapitel 9 beskriver slutligen de problem vi stött på under arbetets gång samt reflektioner kring vår undersökning.

2 Begrepp

För ökad förståelse kring tankegångar och resonemang i denna undersökning presenteras de grundläggande begreppen inför kommande kapitel. Vi presenterar processen för programutveckling som användes i projektet samt Semla som utvärderas i denna undersökning. Då vi utifrån målet i avsnitt 1.2 formulerar att Semla påverkar konstruktion och tillverkning identifierar vi dessa aktiviteter i processen.

2.1 Rational Unified Process

Rational Unified Process (RUP) [3] specificerar en process för programutveckling och består av en mängd aktiviteter, metoder, procedurer och verktyg sammansatta till en produkt. RUP definierar modeller för att beskriva kravspecifikation, arbetsflöden samt olika faser i utvecklingen. Som visas i Figur 2.1.1 delas processen upp i fyra olika faser, *Uppstart*, *Förundersökning*, *Byggande* samt *Leverans*. Varje fas består av flera arbetsflöden med iterationer av aktiviteter där systemet konstrueras gradvis.



Figur 2.1.1: Tidsfördelning mellan faser och arbetsflöden enligt RUP

En kort översikt över de olika faserna i RUP är:

Uppstart – en fas där idé övergår till att bli ett projekt. Omfattning hos projektet bestäms och grundfunktionaliteten identifieras.

Förundersökning – projektets tekniska omfattning undersöks och bestäms. En mer detaljerad beskrivning på kraven samlas in. Specifikationer, utvecklingsplan och grovdesign skapas.

Byggande – konstruktion och tillverkning, integration och test.

Leverans – leverans av system till pilotkund. Testning och verifikation om kundens önskemål på systemet uppfyllts.

2.2 Semla

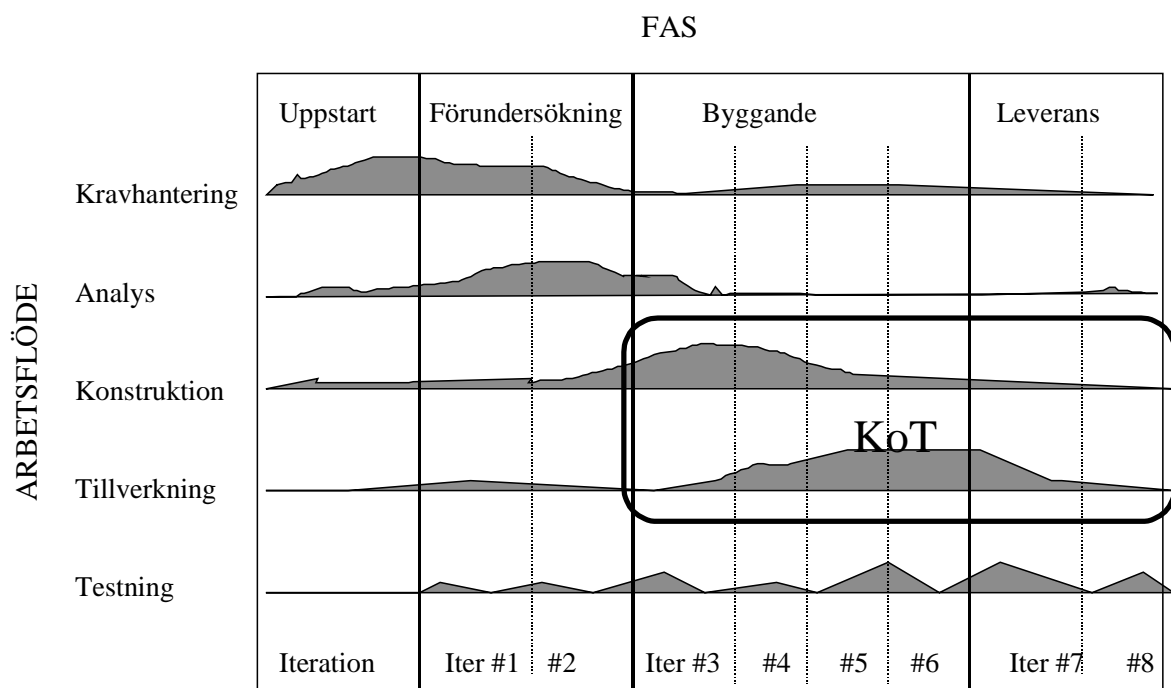
Metoden vi skulle utvärdera är en kontraktbaserad metod som följer ett antal riktlinjer. Metoden fokuserar på regler i form av för- och eftervilkor för operationer och invarianter för dataelement. Dessa regler kan formuleras på svenska, i pseudokod eller i programkod. Förutom att definiera och skriva regler omfattar metoden även normala riktlinjer för programutveckling som förespråkar inkapsling, läsbarhet och liknande kvalitetsaspekter. Målet med Semla är att inkludera välkända akademiska regler för semantisk integritet i det

dagliga arbetet inom mjukvaruindustrin. För mer detaljer se senaste versionen av metodbeskrivningen [1].

2.3 Konstruktion och tillverkning

Enligt vårt mål är vi intresserade av aktiviteterna konstruktion och tillverkning då dessa antas påverkas av Semla. Nedan i Figur 2.3.1 visas RUP-diagram över de faser och arbetsflöden som ingår i processen vid programutveckling. Aktiviteterna konstruktion och tillverkning som vi i fortsättningen kallar **KoT** omfattar arbetsflöde *Konstruktion* och *Tillverkning* i faserna *Byggande* och *Leverans*.

Fasen *Uppstart* omfattar aktiviteter där projektets omfattning bestäms och grundfunktionaliteten identifieras. Fasen *Förundersökning* omfattar aktiviteter där projektets tekniska omfattning undersöks och bestäms. Arbetsflöde *Konstruktion* och *Tillverkning* i dessa två faser innehåller aktiviteter av undersökande karaktär som t.ex. utprovning av lämpligt programspråk för att lösa givet problem. Detta är aktiviteter som inte påverkas av Semla och därför ingår dessa inte i KoT.



Figur 2.3.1: Arbetsmoment som ingår i KoT

3 Strukturer i dataunderlag

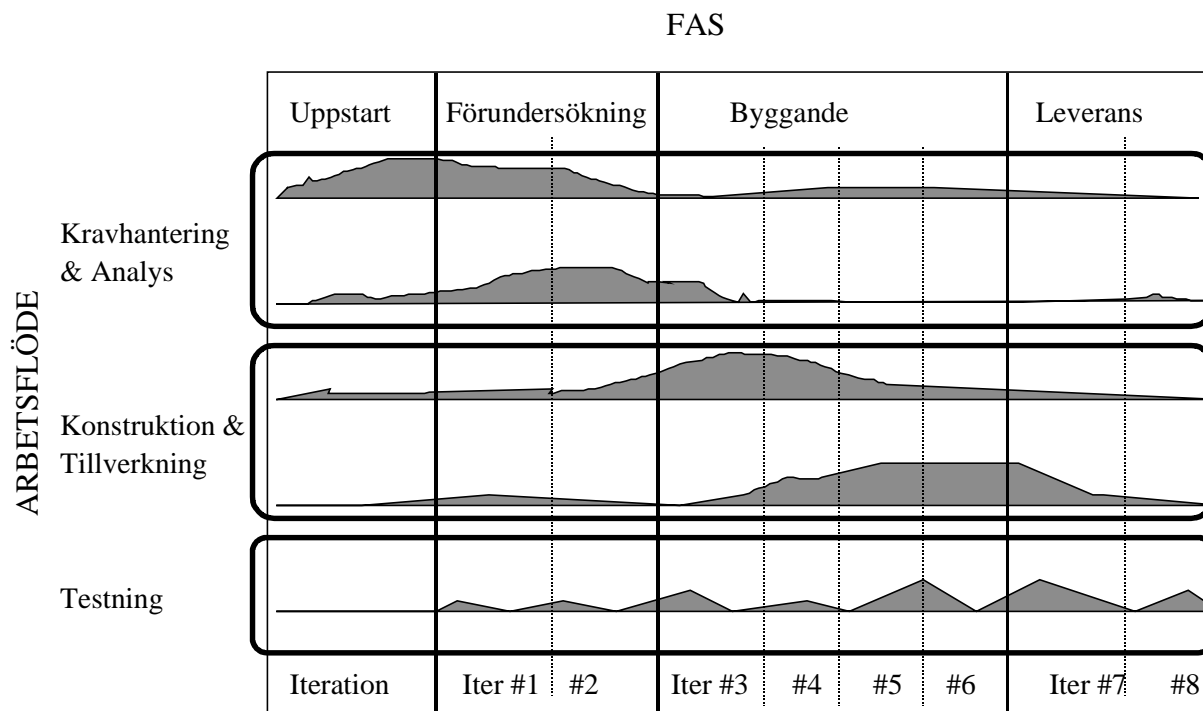
För att ge en inblick i dataunderlaget som fanns att tillgå i undersökningen presenteras strukturen på de rapporter som skapats i projektet Skywalker VPN. De rapporteringar som gjorts var i form av fel- och tidsrapporter.

3.1 Felrapport

I varje felrapport fanns en beskrivning av felet samt dess effekter på systemet. Där fanns även ett förslag på lösning samt eventuella kommentarer på felet och dess lösning dokumenterade. I rapporten fanns även en historik registrerad över processgången med namn på de personer som ingått i hanteringen av felrapporten. Personen som gjort lösningsförslag till felet fanns registrerad under ”TR Answer” ,se avsnitt 2.1 i bilaga A.

3.2 Tidsrapport

Tidredovisningen var tvådimensionell där hela projektets tid redovisats i fyra olika delprojekt. I varje delprojekt redovisades tid på olika aktivitetskoder. Tidsredovisningen kunde mappas mot RUP, se Figur 3.2.1, delprojekten mot RUP-faserna *Uppstart*, *Förundersökning*, *Byggande* och *Leverans*. De olika aktivitetskoderna mot RUP:s arbetsflöden, en mot *Kravhantering* och *Analys*, en mot *Konstruktion* och *Tillverkning* och en tredje mot *Testning*. All tid redovisades på individnivå.



Figur 3.2.1: Projektets aktivitetsredovisning jämfört med RUP

4 Analysplanering

När vi kom in i bilden var projektet Skywalker VPN hos EIN redan avslutat. Vi kunde därför inte påverka vilka data som skulle ingå i råmaterialet utan fick analysera de rapporter som fanns att tillgå i form av fel- och tidsrapportering enligt EIN:s valda rapporteringssätt. För att kunna ställa upp formella hypoteser kring målet, som presenterades i avsnitt 1.2, måste jämförbara variabler identifieras utifrån de förutsättningar som hänsyn måste tas till.

4.1 Förutsättningar

Av de fem utvecklingsgrupper som omfattades av projektet hade två av utvecklingsgrupperna i huvudsak tillämpat Semla och de andra tre hade i huvudsak inte tillämpat Semla. Grupperna som tillämpade Semla har vi för denna redovisning namngivit S1 respektive S2 och de som inte tillämpade Semla har vi namngivit X1, X2 samt X3.

Arbetsbelastningen, i form av komplexitet och omfattning inom ansvarsområdet, hade varierat mellan grupperna. Grupperna S2, X1 och X2 hade likartade uppgifter medan S1 och X3 hade uppgifter av annan karaktär.

I tillägg till att tillämpa Semla var dessutom mycket annat nytt för projektets deltagare. Att vid programutveckling använda ett objektorienterat programspråk som Java var nytt för de flesta projektdeltagarna hos EIN. Både utvecklingsmiljön som utvecklingen skedde på och målmiljön för systemet var nya. Processen för projektet lades upp enligt RUP som var en ny process för projektdeltagarna.

4.2 Identifiering av variabler

För att kunna mäta Semlas påverkan på programkvaliten och utvecklingstid ville vi jämföra felen och tiden som härleds till KoT mellan de grupper som tillämpat Semla och de som inte tillämpat Semla. Då utvecklingsgruppernas uppgifter varierade i komplexitet samt omfattning, och var svåra att mäta, kunde inte felen och tiden som härleds till KoT jämföras mellan utvecklingsgrupperna direkt. För att kunna bortse från komplexiteten och omfattningen behövdes en variabel där enbart Semla påverkade.

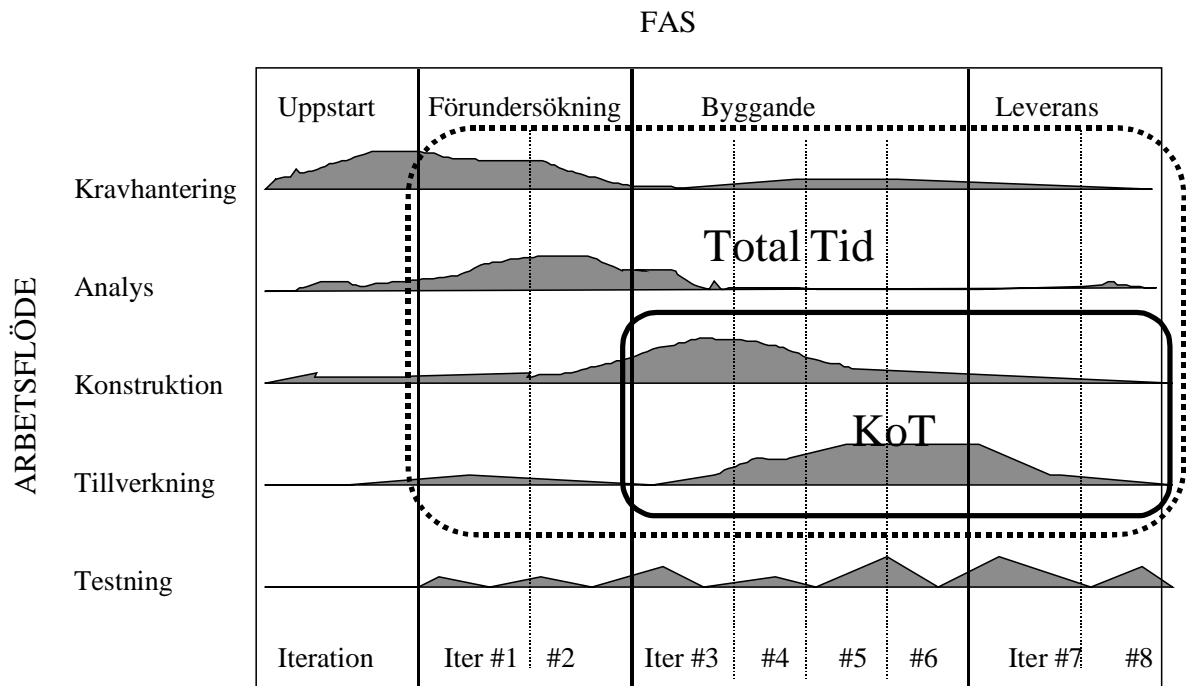
Komplexiteten och omfattningen hos en grupps uppgift antas avspeglas på alla aktiviteter och således spridas ut jämnt fördelat över hela programutvecklingen. Sätts felen som för gruppen kan härledas till KoT i förhållande till det totala antalet fel som gruppen har gjort, antas komplexiteten och omfattningen ha maskats bort.

$$\text{Felförhållande} = \frac{\text{antal fel per grupp härledda till KoT}}{\text{totalt antal fel per grupp}}$$

Samma sak gäller om tiden som gruppen lagt på aktiviteter i KoT sätts i förhållande till den totala tiden för gruppen i projektet.

$$\text{Tidsförhållande} = \frac{\text{nedlagd tid per grupp i KoT}}{\text{total nedlagd tid per grupp}}$$

I Figur 4.2.1 räknades inte tid i fasen *Uppstart* och arbetsflödet *Testning* med. Anledningen till att vi bortsåg från fasen *Uppstart* var att projektet var i "idé-stadiet" och gruppindelningen ännu inte hade genomförts. Då testningens komplexitet och omfattning inte antogs följa komplexiteten och omfattningen i gruppernas uppgifter skulle tiden för detta bortses ifrån i totala tiden.



Figur 4.2.1: Omfattning av Total tid och KoT tid i relation till RUP

4.3 Hypotesformulering

Utifrån våra mål som nämndes i avsnitt 1.2 ställer vi upp hypoteserna att Semla minskar antalet fel som härleds till KoT i programutvecklingen, samt att tidsåtgång för KoT vid programutveckling minskar. Nedan deklarerar våra hypoteser mer formellt.

4.3.1 Hypotes 1 – Antal fel

Nollhypotes 1 – Förhållandet mellan fel som kan härledas till KoT per grupp och det totala antal fel per grupp kommer att vara lika hos de utvecklingsgrupper som använt Semla (S) som hos de som inte använt sig av Semla (X).

$$H_0: \text{felförhållande}_S = \text{felförhållande}_X$$

Alternativ hypotes 1 – Förhållandet mellan fel som kan härledas till KoT per grupp och det totala antal fel per grupp kommer att vara mindre hos de utvecklingsgrupper som använt Semla (S) jämfört med de som inte använt sig av metoden (X).

$$H_{11} : \text{felförhållande}_S < \text{felförhållande}_X$$

4.3.2 Hypotes 2 – Tidsåtgång

Nollhypotes 2 – Andelen KoT-tid per grupp av den totala tiden per grupp vid programutveckling kommer att vara lika för de utvecklingsgrupper som använt Semla (S) jämfört med de som inte använt sig av metoden (X).

$$H_{02} : \text{tidsförhållande}_S = \text{tidsförhållande}_X$$

Alternativ hypotes 2 – Andelen KoT-tid per grupp av den totala tiden per grupp vid programutveckling kommer att vara mindre för de utvecklingsgrupper som använt Semla (S) jämfört med de som inte använt sig av metoden (X).

$$H_{12} : \text{tidsförhållande}_S < \text{tidsförhållande}_X$$

5 Bearbetning av mätdata

I detta kapitel beskrivs arbetet bakom rapporten. Vilka förberedelser gjordes samt hur bearbetning av mätdata gjordes.

5.1 Förberedelser

I inledningsskedet av vårt examensarbete presenterade, vår handledare och tillika vår uppdragsgivare, Eivind J. Nordby EIN-projektet Skywalker VPN och bakgrunden till vår undersökning på detta projekt. Vi fick också en översiktlig genomgång av programutvecklingsmetoden Semla samt en presentation av processen RUP.

För att kunna påbörja arbetet med vår undersökning fick vi namn och telefonnummer till en kontaktperson, Mats Larsaeus, som var ”team leader, function test” för Skywalker VPN på EIN. Vi fick även diverse inläsningsmaterial som grund inför kommande möte med honom.

Vid vårt första möte med vår kontaktperson, fick vi en mer detaljerad presentation av projektet. Vid mötet fick vi även en genomgång av strukturen i felrapporter och hanteringen av dessa inom projektet. Vi introducerades för ytterligare en kontaktperson på EIN, Jan Mölder ”Ass. Project manager”.

5.2 Bearbetning av mätdata från felrapporter

För att spåra antal fel som de olika grupperna gjort gick vi igenom samtliga felrapporter och kopplade dem till respektive grupp via personen som var ansvarig för lösningsförslag, se avsnitt 3.1, på respektive felrapport. Genom tidigare diskussion med vår kontaktperson Mats Larsaeus på EIN kunde vi anta att personen som gett förslag på lösning tillhörde gruppen som skapat modulen som felet härleddes till. Personers gruppstillhörighet bestämdes utifrån dokument från projektet samt genom diskussioner med Mats.

För att identifiera vilka fel som kan härledas till KoT gjordes en grundligare genomgång av felrapporterna där felbeskrivning och eventuella kommentarer analyserades. Vid denna genomgång klassificerades felen genom att de härleddes till KoT respektive inte KoT beroende på vilken aktivitet, enligt bilaga B, som föranlett felet. För att erhålla en pålitligare tolkning av felrapporterna gjorde vi varsin oberoende klassificering som vi därefter jämförde. De felrapporter som tolkats lika i de båda klassificeringarna gavs Status A då de ansågs ha hög säkerhet. Kring de felrapporter som tolkats olika diskuterades en enhällig tolkning fram, dessa ansågs ha lägre säkerhet och gavs Status B. Resultatet redovisas i tabeller i kapitel 6.

5.3 Bearbetning av mätdata från tidsrapporter

För att studera tidshypotesen behövdes den registrerade tiden härledas till respektive grupp. Tiden var registrerad på individnivå och personers gruppstillhörighet bestämdes utifrån dokument från projektet samt genom diskussioner med Jan Mölder.

Tiden som kunde härledas till KoT identifierades direkt då tiden var registrerad enligt RUP, se avsnitt 3.2. Resultatet redovisas i figurer och tabeller i kap 6.

6 Redovisning av resultat från bearbetning av mätdata

I detta kapitel redovisas resultaten från våra undersökningar på fel och tid. Resultaten redovisas i form av tabeller och diagram. Eftersom samtliga rapporter var sekretesskyddade redovisas fel i form av andel av totala antalet fel och tid i form av andel av total tid.

6.1 Undersökning av fel

I Tabell 6.1.1 redovisas resultatet från klassificeringen av felrapporterna. Första kolumnen innehåller de olika grupperna i projektet samt sammanslagning av Semla- respektive icke-Semla grupperna. Andra kolumnen innehåller andelen fel av projektets totala antal fel som kan härledas till KoT för respektive grupp. Tredje kolumnen innehåller respektive grupps totala andel av fel i projektet. Fjärde kolumnen innehåller förhållandet mellan de fel som för respektive grupp är härledda till KoT och för respektive grupps totala fel.

Grupp	Fel från KoT (% av alla fel)	Total (% av alla fel)	Felförhållande $\frac{\text{KoT}}{\text{Total}} (\%)$
<i>S1</i>	0,3	0,3	100
<i>S2</i>	12,7	16,6	77
Total S	13,0	16,9	77
<i>X1</i>	3,0	4,7	65
<i>X2</i>	10,8	13,8	78
<i>X3</i>	53,9	64,6	83
Total X	67,7	83,1	81
Total	80,7	100,0	81

Tabell 6.1.1: Fördelning av fel mellan grupperna och inom respektive grupp

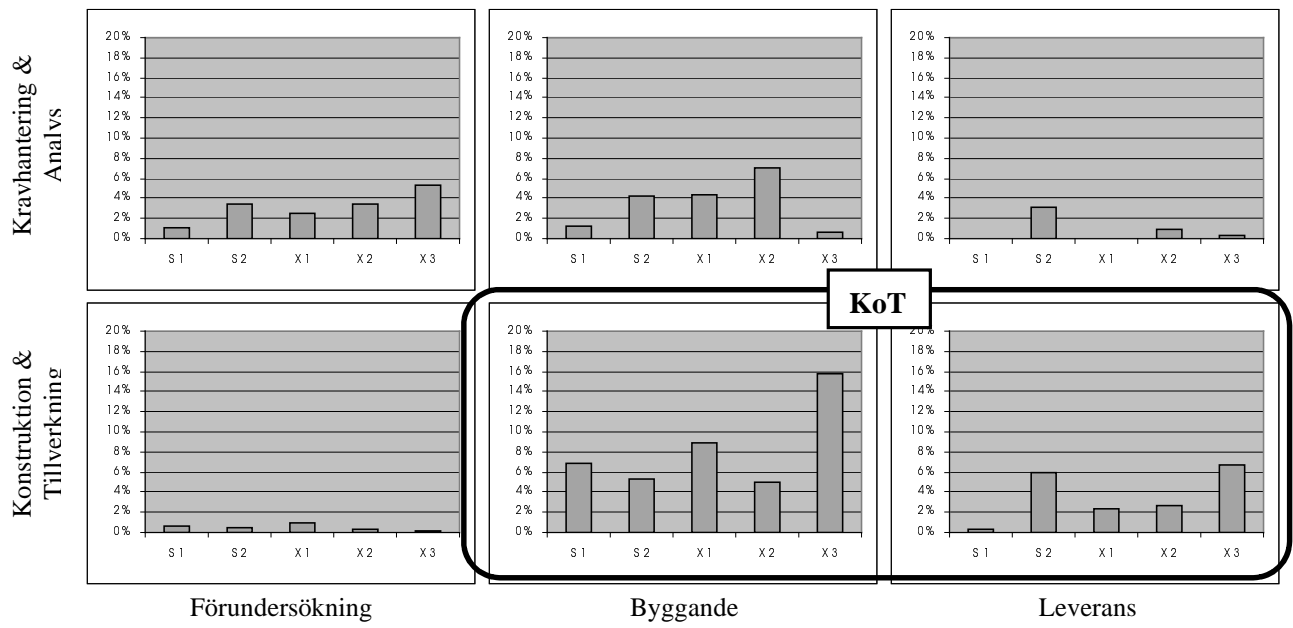
Tabell 6.1.2 visar materialet från Tabell 6.1.1 med status som felrapporterna erhöll vid klassificering, se kap 5.2.

Grupp	Status A			Status B		
	Fel från KoT (% av alla fel)	Total (% av alla fel)	Felförhållande $\frac{\text{KoT}}{\text{Total}}$ (%)	Fel från KoT (% av alla fel)	Total (% av alla fel)	Felförhållande $\frac{\text{KoT}}{\text{Total}}$ (%)
<i>S1</i>	0,3	0,3	100	0,0	0,0	0
<i>S2</i>	8,8	10,8	82	3,9	5,8	67
Total S	9,1	11,0	83	3,9	5,8	67
<i>X1</i>	1,9	2,8	70	1,1	1,9	57
<i>X2</i>	7,2	8,6	84	3,6	5,2	68
<i>X3</i>	42,0	47,5	88	11,9	17,1	69
Total X	51,1	58,8	87	16,6	24,3	68
Total	60,2	70,0	86	20,5	30,0	68

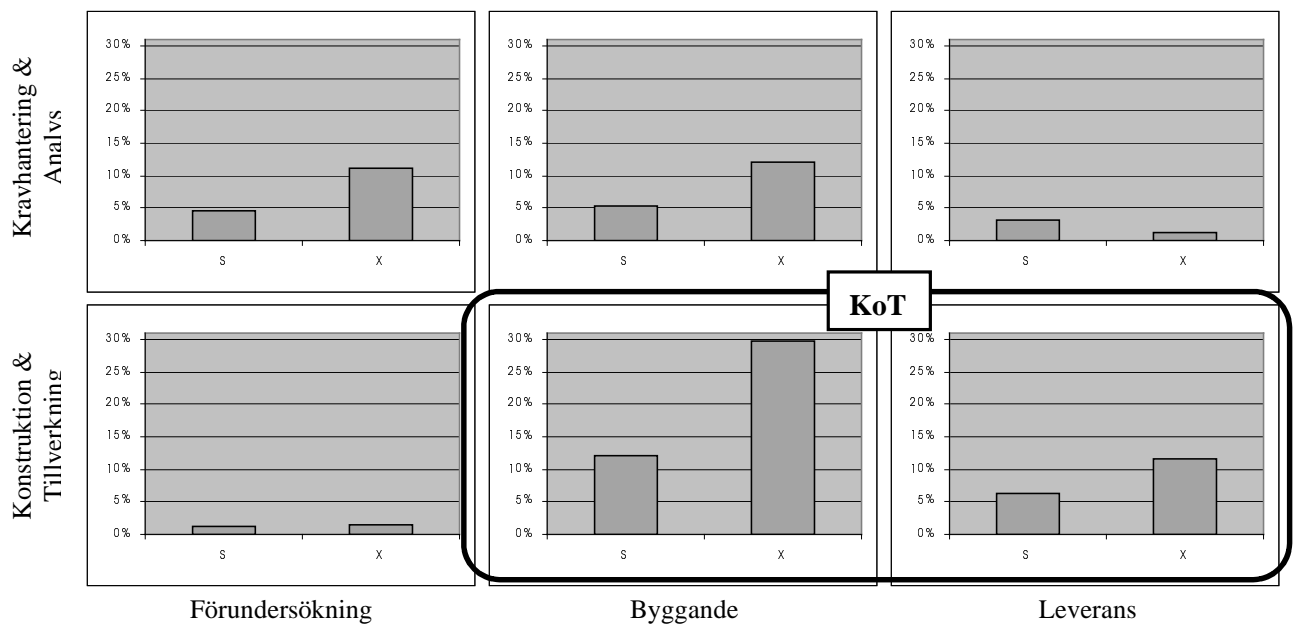
Tabell 6.1.2: Felfördelning uppdelad på klassificeringsstatus

6.2 Undersökning av nedlagd tid

I Figur 6.2.1 presenteras diagram över redovisad tid för grupp S1, S2, X1, X2 respektive X3. Tiden visas för arbetsflöde *Kravhantering & Analys* och *Konstruktion & Tillverkning* i fas *Förundersökning*, *Byggande* respektive *Leverans* i projektet. Dessa kan kopplas till RUP, se avsnitt 2.1. Diagrammen för Figur 6.2.2 visar samma tid med skillnaden att Semlagrupperna, S1 och S2, grupperats till S, och icke-Semlagrupperna X1, X2 och X3 grupperats till X. Tiden för fasen Uppstart och arbetsflöde Test redovisas inte då den inte är intressant för undersökningen, se avsnitt 4.2. I figurerna är tiden för KoT markerad.



Figur 6.2.1: Fördelning av redovisad tid per grupp



Figur 6.2.2: Fördelning av redovisad tid för Semla- respektive icke-Semlagrupper

I Tabell 6.2.1 redovisas tiden från föregående figurer i tabellform. Första kolumnen innehåller de olika grupperna i projektet samt sammanslagning av Semla- respektive icke-Semla grupperna. Andra kolumnen innehåller andelen tid som respektive grupp lagt i KoT av total för alla grupper i projektet nedlagd tid. Tredje kolumnen innehåller respektive grupps totala andel av nedlagd tid i projektet. Fjärde kolumnen innehåller förhållandet mellan den tid som för respektive grupp lagts i KoT och för respektive grupps totala tid.

Grupp	Tid för KoT (%)	Total (%)	Tids- förhållande (%)
<i>S1</i>	7,2	10,2	71
<i>S2</i>	11,2	22,4	50
Total S	18,4	32,6	56
<i>X1</i>	11,3	19,2	59
<i>X2</i>	7,7	19,3	40
<i>X3</i>	22,4	28,9	78
Total X	41,4	67,4	61
Total	59,8	100,0	60

Tabell 6.2.1: Fördelning av tid mellan grupperna och inom respektive grupp

7 Analys

I detta kapitel undersöks de statistiska möjligheterna att testa hypoteserna för att om möjligt förkasta nollhypoteserna. Eventuella indikationer som kan ses utifrån mätdata presenteras och validiteten i undersökningens mätdata diskuteras.

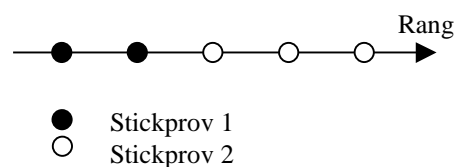
7.1 Hypotesprövning

För att kunna förkasta nollhypoteser av typen $H_0: A = B$, som användes i denna undersökning och anta den alternativa hypotesen $H_1: A < B$, behövdes en statistiskt testfunktion som

applicerades på framtagna värden. Våra värden bestod av två stickprov, med storleken två respektive tre observationer. Det ena stickprovet representerar Semlagrupperna och det andra icke-Semlagrupperna i projektet. Då vi inte kunde säga om stickproven var normalfördelade kom vi fram till att rangsummatest [2] var den enda testfunktionen som var användbar. Vid rangsummatest är det tillräckligt att observationerna går att rangordna efter storlek.

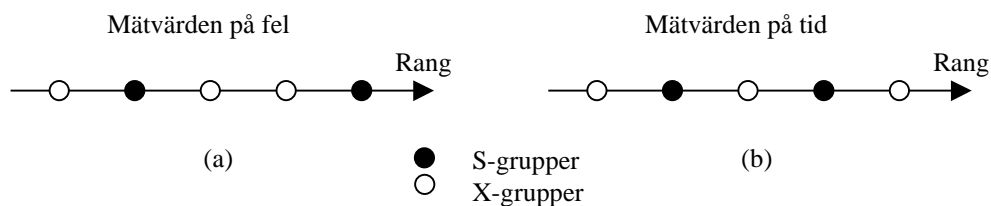
När en nollhypotes ska förkastas och den alternativa hypotesen antas bör signifikansnivån hållas på en låg nivå t.ex. 1 eller 5%. Med signifikansnivå menas risken att förkasta nollhypotesen då den i själva verket är riktig. Studerar man tabell över signifikansnivåer för ett rangsummatest, ser man att en signifikansnivå på 10% är det bästa som kan erhållas med stickprov av storleken som förekom i vår undersökning. Denna signifikansnivå ger en för stor risk att nollhypotesen förkastas på felaktig grund och någon hypotesprövning är inte relevant.

Signifikansnivån på 10% erhålls då spridningen utfaller som Figur 7.1.1 visar, där de olika stickproven hamnar på en varsin kant av spridningen.



Figur 7.1.1: Rangordningen som ger tydligast utslag

I Figur 7.1.2 ses spridningen på de aktuella mätvärden som redovisades i kap 6. Denna spridning avviker grovt från den spridningen som gav en signifikansnivå på 10%.



Figur 7.1.2: Rangordningen för aktuella mätvärden

7.2 Indikationer

När vi studerade tabellen för felförhållandet mellan de olika grupperna, se Tabell 6.1.1, kunde vi se att spridningen mellan Semla och icke-Semlagrupper var stor. Detta visas i Figur

7.1.2(a). Om vi bortsåg från grupp S1 som hade ett litet antal fel och därmed gav ett missvisande felförhållande kunde vi se att icke-Semlagruppen X3 hade största felförhållande. Även om denna grupp hade en annorlunda typ av uppgift kunde vi se att denna grupp hade gjort större andel fel i KoT. Om detta endast berodde på typ av programutvecklingsuppgift var svårt att avgöra. Enligt samtal med Jan Mölder fick vi vissa indikationer på att det inte bara berodde på uppgiftstypen utan kunde också ha påverkats av programutvecklingsmetoden.

När vi studerade de grupper med liknande programutvecklingsuppgifter kunde vi se att bland de tre hade en icke-Semlagrupp, X2, största felförhållandet men den grupp som hade minst felförhållande var också en icke-Semlagrupp, X1. Denna grupp hade totalt mindre mängd fel än de andra två. I denna icke-Semlagrupp fanns en medlem som jobbat enligt Semlaliknande metod tidigare.

Vid studier av Tabell 6.2.1 kunde vi även här se att spridningen var stor mellan grupperna, detta åskådliggörs i Figur 7.1.2(b). I avseende på tidsförhållandet hade X3 störst värde. När vi studerade tidsförhållande på de grupper med liknade programutvecklingsuppgifter kunde vi se att Semlagruppen, S2, hamnade mitt emellan de båda icke-Semlagrupperna. För icke-Semlagrupperna var placering på spridningen det omvända från studien på felförhållande. Här hade grupp X2 minst värde på felförhållande.

När vi jämförde värden i Tabell 6.1.1 och Tabell 6.2.1 kunde vi se att grupp X3 har 65 % av totalt antal fel och 29 % av totala tiden. För övriga grupper var förhållandet mellan fel och tid omvänt och inte lika påtagligt. Vad detta kunde bero på är svårt att säga.

7.3 Validitet

För att vi skulle kunna dra relevanta slutsatser utifrån resultatet och analysen var vi tvungna att kontrollera validiteten för undersökningen.

I och med att projektet Skywalker hos EIN redan var avslutat när vi kom in i bilden kunde vi inte påverka vilka data som skulle ingå i rapporteringen. Detta ledde till att antaganden fick göras för att få fram det vi sökte.

I kopplingen av felrapport till grupp som nämndes i kap 5.2 fanns en liten risk att rapporten kopplades till fel grupp p.g.a. att personen som lämnade lösningsförslaget inte tillhörde gruppen som var källan till felet. Vi fick en del antydningar till att detta kunde ske p.g.a. att personen var mer lämpad till att åtgärda felet.

Som nämnts tidigare hade projektet två grupper som tillämpat Semla och tre grupper som inte gjort det, men i icke-Semlagrupperna fanns avvikelser i form av personer som använt

liknande metod tidigare och var vana vid detta arbetssätt. Det fanns även personer i icke-Semlagrupperna som fastnade för metoden och ville eller försökte jobba enligt den.

Enligt indikationer från personer på EIN hade grupperna S1 och X3 programutvecklingsuppgifter av avvikande karaktär jämfört med S2, X1 och X2, som hade mer jämförbara uppgifter. Detta kunde innebära att vårt antagande att erhålla en variabel som enbart påverkades av Semla genom att komplexiteten och svårighetsgrad i gruppernas uppgifter skulle maskas bort, se kap 4.2, inte kunde appliceras på alla grupper.

Valet av vilka personer som skulle ingå i respektive programutvecklingsgrupper har, med undantag av gruppen X3, utförts slumpmässigt. Större vikt lades på erfarenhet vid valet av personer till denna grupp då dess uppgift antogs vara mer komplex.

Under projektets gång har det funnits personer som inte har tillhört någon programutvecklingsgrupp utan gjort arbete åt olika grupper. Då tiden redovisats på personnivå har det varit omöjligt att koppla redovisad tid till de grupper där arbetet utförts. Vi valde att ta bort tiden för de personer som inte haft någon gruppstillhörighet. Tiden som registrerats på dessa personer låg till stor del i fasen *konstruktion* under aktiviteten *design/implementation*, se i bilaga C. Hur detta påverkade var svårt att avgöra.

8 Slutsatser

Vi har presenterat en undersökning som skulle utvärdera om Semla reducerade antalet konstruktion och tillverkningsfel samt om Semla minskade tidsåtgången för konstruktion och tillverkning i ett industriprojekt.

Det fanns ingen möjlighet att dra några starka slutsatser kring de formulerade hypoteserna då vi inte kunde förkasta nollhypotesen eftersom acceptabel signifikansnivå ej var möjlig att nå med test, se analysen kap 6. Anledningen till detta var att vi hade för få mätpunkter.

9 Erfarenheter

I detta kapitel beskriver vi problem vi stött på under arbetets gång och egna reflektioner kring undersökningen och dess förutsättningar.

9.1 Problem

Största problemet under arbetets gång har varit att identifiera mätbara data då vi inte hade varit med och planerat inför projektet vilka registreringar som skulle genomföras under projektets gång. Många timmar har spenderats på att diskutera och analysera erhållna data för att få fram jämförbara variabler. Dessa diskussioner och analyser har till stor del gjorts i samråd med vår handledare och uppdragsgivare Eivind J. Nordby.

Vår första väg att försöka koppla felrapport till ansvarig grupp var att via ett dokument på relation mellan produktnummer och ansvarig person. Då samma produktnummer förekom på flera personer förkastades detta dokument. Tillsammans med Mats Larsaeus diskuterades en alternativ väg fram där vi genom person som gett lösningsförslag, i varje felrapport, identifierade rapporten till ansvarig grupp.

I felrapportens historik dokumenterades tidpunkten då designern fick felrapporten samt tidpunkten då designern rättat felet och lämnat rapporten vidare till ny testning. Genom dessa tidpunkter hoppades vi att felupprättningstiden kunde bestämmas men då en liggtid kunde förekomma kunde inte denna tid bestämmas.

Utifrån de felkoder som fanns för att beskriva vilken typ av fel som felrapporten handlade om skulle man kunna härleda till var i programutvecklingen felet härstammade ifrån. Då det inte verkade som om stor vikt hade lagts på att välja rätt kod att beskriva felet med, utan ett fåtal felkoder användes mer flitigt, kunde inte dessa användas för härledning. Detta styrktes via samtal med Mats Larsaeus. Därför gjordes en egen härledning genom klassificering av varje fel.

I slutfasen av vår undersökning uppdagades att vi använt en felaktig statistisk testmetod för vår hypotesprövning. Detta gav en hel del merarbete samt att delar av redan genomfört arbete fick förkastas.

9.2 Egna reflektioner

Inför en undersökning av denna karaktär bör en noggrann planering göras där man analyserar vad som är möjligt att mäta under projektets gång så att man direkt får fram variabler som är enbart beroende av det som ska undersökas. För att underlätta detta bör belastningen spridas jämnt på alla grupper samt att bibehålla strukturen på gruppindelning. En annan aspekt att ta hänsyn till är att ha tillräckligt många mätpunkter för att nå acceptabla signifikansnivåer på de statistiska testmetoderna som är möjliga att använda. I det undersökta projektet var det svårt att få mätpunkter på individnivå då det inte gick att se om en person som lade tid på att rätta

ett fel även var den som skapat koden som felet härrörde från. En annan indelning med fler mätpunkter vore att hitta eventuella delgrupper inom grupperna. Om gruppmedlemmarna har jobbat t.ex. parvis. Detta har vi tyvärr inte haft tid att undersöka.

Det optimala vore att genomföra två exakt likadana projekt med likvärdig kompetens i båda. Detta är knappast realistiskt ur ekonomisk synvinkel i ett industriprojekt av det undersökta projektets storlek. En mer realistisk tanke vore att undersöka två liknande projekt med likartade förutsättningar. Detta var tanken då SERG presenterade Semla inför Skywalkerprojektet. Då detta projekt var av en helt ny karaktär inom EIN fanns inget annat liknande projekt att jämföra med.

För att kunna forska vidare och nå målet vi formulerat behövs fler mätpunkter för att kunna erhålla högre signifikansnivå. En annan aspekt inför vidare forskning är att försöka mäta komplexitet och omfattning, t.ex. genom att mäta kodrader eller antalet funktioner, för att erhålla högre validitet. Att analysera förhållandet mellan fel och tid för att söka ett samband kunde eventuellt ge intressant information om komplexiteten hos gruppernas uppgifter.

Referenser

- [1] Blom Martin, Nordby J. Eivind, Brunström Anna. *Metod Description for Semla: A Software Design Method with a Focus on Semantics*, Department of Computer Science at Karlstad University, Ericsson Infotech AB, Karlstad 2000.
- [2] Körner Svante. *Statistisk dataanalys*, Studentlitteratur, Lund 1987.
- [3] Eivind J. Nordby. *A quick introduction to RUP*, Department of Computer Science at Karlstad University, Karlstad 2000

A Felrapportsmall för Skywalker VPN

Date: 2002-02-07

Page last modified: Xxx XX, XXXX

TRSearch: Read XXXXXXXX

Heading: Rubrik, sammanfattning
Priority: Prioritet på felet, A, B eller C
Status: Felrapportens status
Secondary TR(s):
MSS Id:

Customer:
Market ref:
Country Code:
Site id:
Site Status:

Product name: Produktens namn
Product: Produkt ID
Design MHO:
Design Resp.:
Sub System:

Current MHO:
Arrived to MHO:
Current User:

Register Date: Datum då rapporten registrerades
Registered at MHO:
Registered by: ID på person som registrerat rapporten
Prepared by: Namn på person som registrerat
Prepares phone: Telefon till person som registrerat

Additional info:
Comment:

Comp.pform:
Appl.pform:

Technical Answer Date: Datum för svar på felrapporten
Answer code: Vad som ska ske med felet
Fault code: Vad för typ av fel

1. TR Observation

1.1 Trouble Effect

Beskrivning på effekterna av felet

1.2 Trouble Description

Beskrivning av felet

1.3 Measures

Beskrivning av vad som kan vara orsak till felet

2. TR Answer

2.1 TR Answer, Rev. A

Svar, oftast ett lösningsförslag
Answered: Datum och ID på person som svarat

4. TR Notebook

4.1 Current Notebook

Eventuella kommentarer

4.2 Previous Notebook

Eventuella tidigare kommentarer

6. TR History of status changes

Date	Status	Phase	Action	Userid	Mhoid
	FO	Design	Sent to finish by	<u>XXXXXXX</u>	
	FO	Design	Sent to handler by	<u>XXXXXXX</u>	
	TA	Design	Accept and follow up by	<u>XXXXXXX</u>	
	TA	Design	Sent to handler by	<u>XXXXXXX</u>	
	AP	Design	Answered by	<u>XXXXXXX</u>	
	PP	Design	Approved by	<u>XXXXXXX</u>	
	AN	Design	Proposed Tech. Answer by	<u>XXXXXXX</u>	
	RE	Register	Sent to handler by	<u>XXXXXXX</u>	
	PR	Register	Registered by	<u>XXXXXXX</u>	
	PR	Register	Created by	<u>XXXXXXX</u>	

B De olika aktiviteterna i programutvecklingen enligt RUP

Omfattas inte av KoT

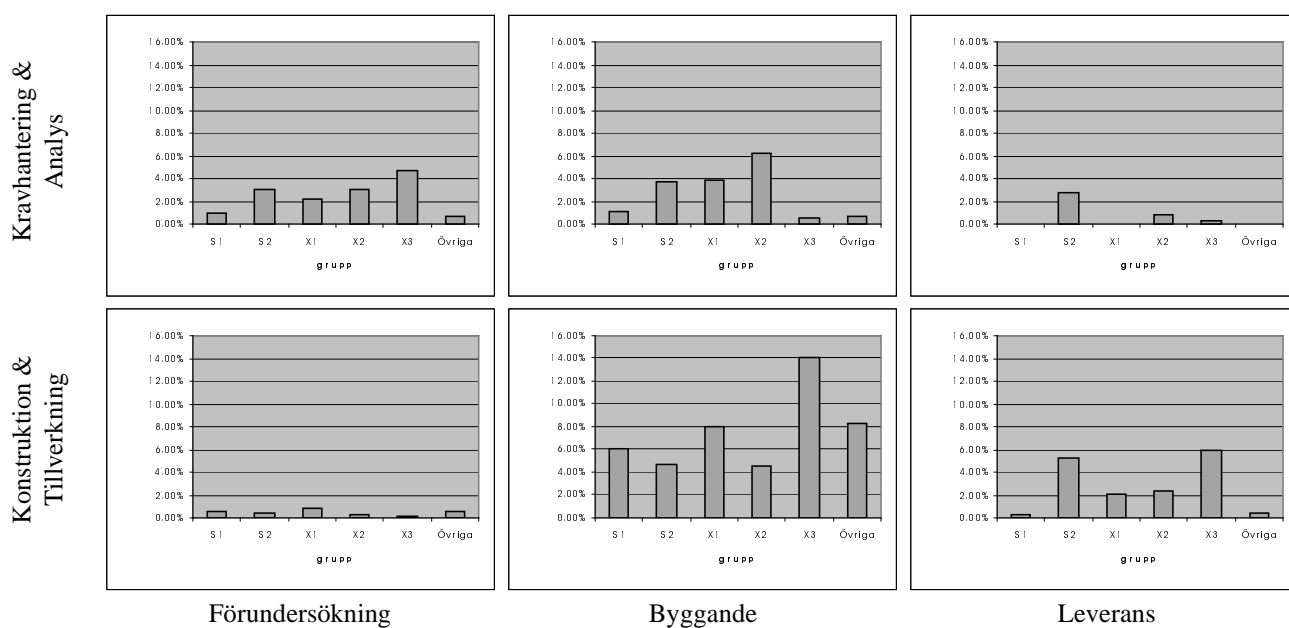
- Tolkning av kundens önskemål.
- Specifikation över vad programmet/systemet ska "kunna".
- Indelning av programmet/systemet i moduler där varje modul har fastställda och skilda uppgifter.
- Grovdesign av modulinterface. Vad ska modulerna kunna erbjuda utåt.

Omfattas av KoT

- Specifikation av hur modulens uppgift ska lösas.
- Fastställande av vilka funktioner som behövs inuti modulen, vad de gör och hur de kommunicerar med varandra inom modulen.
- Beskrivning av gränssnittet utåt för kommunikationen med andra moduler i programmet/systemet.
- Fastställande av vilka algoritmer som ska användas i modulen för att modulens uppgift ska lösas.
- Implementering av funktionerna.

C Redovisad tid inklusive personer utan grupptillhörighet

Nedan presenteras diagram över redovisad tid för varje grupp, inklusive personer utan grupptillhörighet som redovisats som övriga. På grund av sekretesskäl redovisas all tid som procent av den totala tiden.



Figur B.1: Redovisad tid per grupp inklusive personer utan grupptillhörighet

D Förkortningar och begrepp

EIN	Ericsson Infotech AB
KoT	Konstruktion och tillverkning, arbetsflöde Konstruktion och Tillverkning i RUP-faserna Byggande och Leverans
NUTEK	Närings- och teknikutvecklingsverket
RUP	Rational Unified Process, en process för programutveckling
S1, S2	Utvecklingsgrupper som tillämpat Semla
Semla	En programutvecklingsmetod som bygger på kontraktprogrammering
SERG	Software Engineering Research Group, forskningsgrupp vid institutionen för datavetenskap vid Karlstads universitet
SKUTT	”Improved software quality through semantic descriptions”, forskningsprojekt vid institutionen för datavetenskap vid Karlstads universitet
Skywalker VPN	Industriprojekt hos EIN
X1, X2, X3	Utvecklingsgrupper som inte tillämpat Semla