



## Rational Unified Process



### En modell för kvalitetsfrämjande systemutveckling?

Sandra Pettersson

---

#### Sammanfattning

I en värld som befinner sig i ständig förändring och samtidigt överflödar av en omfattande och överblickbar mängd modeller, metoder och tekniker för systemutveckling, låter varje försök att skapa överblick och klargöra modellernas och metodernas starka och svaga sidor samt rekommendera strategier och åtgärder för att förbättra deras systemutvecklingsvärde, rimligt.

Systemutvecklingen har blivit en mycket komplicerad och dynamisk process. De varierade och ständigt föränderliga kundkraven har drivit önskemålet att integrera eller samordna verksamhetslivscykeln med systemlivscykeln. Denna integration har bl.a. resulterat i ett omfattande behov av nya kunskaper och färdigheter. Utan lämplig modell och metodstöd som samordnar kundens, användarnas och systemutvecklarnas intuition, erfarenheter, färdigheter etc. finns det ringa förutsättningar att klara av att balansera kunskapsbehovet. Osäkerheten i de dagliga beslut och handlande som avgör systemutvecklingens framgång har blivit ett faktum. Samtidigt finns det inte någon modell eller metod som är bäst för alla systemutvecklingssituationer. Dessutom är inte alla modeller och metoder lika goda i alla situationer. Hur klarar vi att välja lämplig systemutvecklingsmodell för att absorbera systemutvecklarnas osäkerhet?

Denna studie analyserar, resonerar och söker klargöra på ett systematiskt sätt RUP-modellens, (Rational Unified Process) och kvalitetsfrämjande stöd vid systemutveckling. Utredningen har fokuserat på och belyser frågan hur en systemutvecklingsmodell i allmänhet och RUP-modellen i synnerhet uppfyller den kvalitetsbild som berör systemutvecklingens tre grundpelare nämligen process, produkt och miljö.

Studien har visat på RUPs klara och oklara sidor angående kvalitet och har även gett svar på i vilka situationer RUP är bäst lämpat. RUPs klara sidor är att den en relativt heltäckande iterativa utvecklingsprocessen etc. De oklara sidorna är att den i inte har ett klart varumärke, den saknar stöd för Knowledge management och konflikthantering etc. Slutligen passar RUP i situationer där relationerna är klara (ömsesidigt förtroende) och kunskapen hög eller relationerna klara och kunskapen låg (t.ex. vid införandet av ett helt ny systemtyp).

---

#### Handledare:

Thanos Magoulas

Maria Johnson-Kjällström

Annika Nilsson

## Förord

Under hösten 1998 gick jag kursen Informationssystemmiljöer, och inom kursens ramar ingick en 5-poängsuppsats. Uppsatsen titel var *Kvalitet - kan den garanteras i informationssystem?*, och den berörde alltså ämnet kvalitet. Frågorna den försökte belysa var bl.a.

- Vad är kvalitet?
- Hur kan kvalitet används vid utveckling av informationssystem?
- Kan vi försäkra oss om bra kvalitet?
- Är kvalitet mätbart?
- Vilka olika hjälpmedel finns det vid kvalitetsutveckling?

I den ovan nämnda 5-poängsuppsatsen fanns det inte möjlighet att fördjupa sig så mycket som jag hade velat, så när valet av magisteruppsats skulle göras bestämde jag mig för att fortsätta studera detta område.

Under våren kom jag, i och med Institutionen för Informatiks arbetsmarknadsdagar, i kontakt med Enator, och efter ett antal samtal med Annika Nilsson och Maria Johnson-Kjällström på Enator Informationssystem Väst AB stod det klart att vi hade ett gemensamt intresse för kvalitet och systemutveckling. För dem var det intressant att utvärdera RUP, en modell som vill stödja systemutvecklingsprocessen ur en kvalitetssynvinkel, och för mig var det intressant att försöka finna en vägvalsmodell för att utvärdera modeller (såsom RUP) ur ett kvalitetsperspektiv.

För att lösa denna uppgift har jag haft hjälp av mina handledare Annika Nilsson och Maria Johnson-Kjällström på Enator som varit ett stöd vid insamling av material, ett bollplank, en hjälp för att förmedla kontakter till intervjuerna och en intressant informationskälla. Jag har även haft stor hjälp av min handledare Thanos Magoulas på institutionen för Informatik, som alltid har ställt upp, varit en fantastisk källa för information och som inspirerat mig till denna uppsats.

Jag vill tacka:

- Mina handledare Maria Johnson-Kjällström, Annika Nilsson och Thanos Magoulas.
- Enator Informationssystem Väst AB.
- ”Intervjuoffrena” som tagit sig tid till mig.
- Slutligen min sambo som stått ut med mig under denna tid.

# Innehållsförteckning

<b>1. Inledning .....</b>	<b>5</b>
1.1 BAKGRUND.....	5
1.2 SYFTE OCH UTREDNINGSTRATEGI .....	6
1.3 HUVUDFRÅGESTÄLLNING OCH AVGRÄNSNING .....	7
1.4 DISPOSITION .....	8
<b>2. Utredningsmetodik .....</b>	<b>9</b>
2.1 SÖKANDE EFTER RELEVANTA KUNSKAPER OCH ERFARENHETER OM SYSTEMUTVECKLING UTIFRÅN ETT KVALITETSPERSPEKTIV .....	9
2.1.1 Tidigare studier.....	9
2.1.2 Kvalitetsfrågor.....	10
2.2 SKAPANDE AV EN VÄGVALSMODELL FÖR ATT BEDÖMA RUPS LÄMPLIGHET INFÖR VALET .....	10
2.3 GRANSKNING AV RUPS SYSTEMUTVECKLINGSVÄRDE OCH VÄGVALSMODELLENS LÄMPLIGHET.....	11
2.4 UTREDNINGSMETOD .....	12
<b>3. Kvalitetsramverk .....</b>	<b>14</b>
3.1 EN ANALYS AV BEGREPPET PROCESS .....	15
3.1.1 Olika föreställningar av begreppet process.....	15
3.1.2 Processens målbild och kvalitetsmål .....	17
3.1.3 Processens utformning.....	19
3.1.4 Förhållande mellan process och produkt.....	21
3.1.5 Processens bakomliggande filosofi.....	21
3.2 EN ANALYS AV BEGREPPET PRODUKT .....	22
3.2.1 Olika föreställningar av begreppet produkt.....	22
3.2.2 Produktens målbild .....	24
3.3 EN ANALYS AV BEGREPPET MILJÖ.....	26
3.3.1 Mål och strategi.....	26
3.3.2 Humanresurser .....	27
3.3.3 Organisationsform.....	27
3.3.4 Utvecklingsinfrastruktur .....	28
3.4 SAMMANFATTNING .....	29
<b>4. Empirisk undersökning .....</b>	<b>31</b>
4.1 INTERVJUFRÅGORNAS RELEVANS .....	31
4.1.1 Bakgrund.....	31
4.1.2 Miljö.....	32
4.1.3 Arkitektur .....	32
4.1.4 Mål.....	33
4.2 REDOVISNING AV DEN EMPIRISKA UNDERSÖKNINGEN .....	35
4.2.1 Bakgrund.....	35
4.2.2 Miljö.....	36
4.2.3 Process.....	40
4.2.4 Produkt.....	41
4.3 SAMMANFATTNING .....	42
<b>5. Vägvalsmodellen.....</b>	<b>43</b>
5.1 BAKGRUND.....	43
5.2 GENOMGÅNG AV VÄGVALSMODELLEN .....	43
5.2.1 Miljö.....	43
5.2.2 Systemutvecklingsprocess .....	48
5.2.3 Samordningseffekter (Målbilder).....	49
5.2.4 Modellens egenskaper.....	50
5.3 SAMMANFATTNING .....	51
<b>6. Granskning och resultat av RUP .....</b>	<b>52</b>
6.1 SAMMANFATTNING AV RUP.....	52
6.1.1 Process.....	53
6.1.2 Produkt.....	54
6.1.3 Miljö.....	54
6.1.4 Intervjusammanställning om RUP .....	54

6.2 GRANSKNING AV RUP MED HJÄLP AV VÄGVALSMODELLEN .....	58
6.2.1 Miljö.....	58
6.2.2 Systemutvecklingsprocess .....	64
6.2.3 Samordningseffekter (Målbilder).....	66
6.2.4 Modellens egenskaper.....	67
6.3 SAMMANFATTNING .....	68
6.3.1 Kan RUP säkra processkvalitet? .....	68
6.3.2 Kan RUP säkra produktkvalitet? .....	69
6.3.3 Kan RUP säkra miljökvalitet? .....	69
<b>7. Slutsatser och rekommendationer.....</b>	<b>71</b>
7.1 EN MODELL FÖR ATT BEDÖMA RUPS TILLÄMPNINGSOMRÅDE.....	71
7.1.1 RUP som styrande modell.....	72
7.1.2 RUP som stödjande modell.....	72
7.1.3 RUP som lärande modell .....	72
7.1.4 RUP som inspirationskälla och idégenerator .....	73
7.2 SLUTORD .....	73
7.2.1 Vidare studier.....	73
<b>8. Referenser .....</b>	<b>75</b>
8.1 BÖCKER.....	75
8.1.1 Häfte.....	76
8.1.2 Magister uppsats .....	76
8.2 INTERNET .....	76
8.3 ARTIKLAR .....	76
8.4 ÖVRIGT.....	76
<b>Appendix.....</b>	<b>77</b>
A. EVOLUTIONÄR, INKREMENTELL OCH ITERATIV UTVECKLING .....	77
A.1 Evolutionär utveckling.....	77
A.2 Inkrementell utveckling.....	78
A.3 Iterativ utveckling .....	79
A.4 Slutsats.....	80
B. INTERVJU MED PÄR JANSSON.....	81
C. HISTORIA BAKOM KVALITET .....	84
D. INTERVJUERNA .....	86
D1. Via telefon den 23/8 kl.15-16 .....	86
D2. Intervju den 16/8 kl. 13-14 .....	87
D3. Intervju den 26/8 kl. 8-9.....	90
D4. Intervju den 25/8 kl. 9-10.30.....	91
D5. Intervju den 30/8 kl. 14-15 .....	94
D6. Intervju den 25/8 kl. 13-14.30.....	96
D7. Intervju den 24/8 kl.10-11.15.....	99
D8. Intervju den 23/8 kl. 9.30-11.15 .....	102
E. INTERVJUERNA.....	107
E.1 Binomen.....	107
E.2 Frontec.....	108
E.3 Skandia Liv.....	109
F. RUP .....	113
F1. Varför RUP?.....	113
F2. Processöversikt.....	115
F3. Faser.....	118
F4. Huvudarbetsflöden .....	119
F6. Kvalitet och RUP.....	121

# 1. Inledning

I detta kapitel redovisas arbetets bakgrund, syfte, problemställning, avgränsning och hur innehållet av min studie har organiserats.

## 1.1 Bakgrund

Strävande efter kvalitet i systemutveckling har varit ett huvud tema under hela 90-talet<sup>1</sup>.

Intressen för kvalitet reflekteras i begrepp såsom ISO-9000, Total Quality Management (TQM), Product Life Cycle och Process Life Cycle etc. Kvalitet utgör ett komplicerat begrepp eftersom det sällan kan definieras i objektiva och mätbara termer, och i dagens samhälle ses kvalitet som en respons till kundens heterogena och föränderliga förväntningar.

FRISCO rapporten (figur 1.1) ger en relevant bild om denna situation. Enligt denna undersökning är endast 10 % av alla system som utvecklas välanpassade till människans behov, alltså är det 90 % av utvecklingsvisionerna som inte blir som det var tänkt.

<b>40%</b> av alla utvecklingsvisioner ser aldrig verklighetens ljus
<b>50%</b> av alla systemen skapar mer störningar och resursslöseri än stöd och rationalisering
Endast <b>10%</b> av systemen är välanpassade till människans dagliga behov

**Figur 1.1:** Undersökning om systemutvecklingens lyckande och misslyckande (FRISCO, 1998)

Det finns även andra undersökningar som har ungefär samma dystra siffror (Flynn, 1997), se figur 1.2.

<b>20%</b> ger en positiv effekt för företagen
<b>40%</b> ger varken eller
<b>40%</b> är misslyckade vilket innebär ej levererade eller oanvändbar

**Figur 1.2:** Undersökning om systemutvecklingens lyckande och misslyckande (Flynn, 1997)

Vad ligger bakom denna utveckling? Vilka faktorer har förändrats och ligger utanför de traditionella systemutvecklingsmodellernas gränser? Finns det systemutvecklingsmodeller som är lämpliga för att bemöta kundens förväntningar? Hur granskar man dessa modellens lämplighet?

Först och främst har systemutvecklingsmiljöer förändras radikalt. Om man utgår från dagens litteratur kan man härleda nedanstående bild av förändringar.

1. **Ökad komplexitet i systemdesign.** Därmed ökade krav för flexibilitet och variation.
2. **Ökade krav på att halvera systemutvecklingstiden och minimera systemutvecklingskostnaderna.**
3. **Ökad medvetenhet av systemlivscykel (SLC)** dvs. en beskrivning av systemet från koncept till disposition.
4. **Tillgänglighet av bättre teknologi** för systemutveckling som samtidigt kräver utbildning och kompetens.

---

<sup>1</sup> Se appendix D som beskriver historien om kvalitet från dess upptäckt tills idag

5. **Ökade krav för bättre kommunikation mellan IT-leverantörer och IT-användare.**
6. **Ökade krav att integrera systemutveckling med affärsutveckling och kompetensutveckling.**
7. **Utveckling av "outsourcing" filosofi.**
8. **Produktutveckling istället för systemutveckling** genom att standardsystem har ökat i popularitet.

En sak är klar: gårdagens systemutvecklingsmodeller, metoder och tekniker är i princip olämpliga för att bemöta ovanstående förändringar. T.ex. påstår J. Mylopoulos (1998) i sin analys om designmetodernas status följande:

*"Traditional techniques for building information system are not longer adequate."*

Då det är så stor del av alla systemutvecklingsprojekt som misslyckas behöver man en bättre förståelse av själva systemutvecklingsprocessen. Varje företag måste ställa sig ett antal frågor såsom hur lämpliga deras systemutvecklingsmodeller, metoder, tekniker och verktyg är för att bemöta kvalitetsvärden. Denna kunskapsmässiga uppdatering kommer att leda till sökande och val av nya systemutvecklingsmodeller, men hur vet vi att den valda systemutvecklingsmodellen säkrar kvaliteten? Mao, vilka kriterier bör användas vid val av lämpliga modeller, metoder och tekniker för en kvalitetsfrämjande systemutvecklingsverksamhet?

Mitt intresse om en kvalitetsfrämjande systemutvecklingsverksamhet var i harmoni med Enator Informationssystem Väst AB. Företagets intresse låg i behovet att utvärdera **RUP** (Rational Unified Process) för att se hur väl denna modell främjar företagets kvalitetssträvan. Att företaget lägger fokus på kvalitet i systemutvecklingsprocessen kändes relevant, då det förväntas leda till konkurrensmässiga fördelar. Fördelarna skapas genom att företaget i varje ny situation kan garantera rätt process, rätt produkt och i rätt miljö.

RUP är en produkt som har funnits på marknaden i sin nuvarande form sedan november 1998, och den vänder sig till mjukvaruutvecklare. RUPs styrka ligger i att organisera och utrusta själva systemutvecklingsprocessen på ett sätt som tillgodoser målet att producera högkvalitativ mjukvara som:

- **Möter kundens krav...**
- **...inom budget...**
- **...och i tid.**

Denna marknadsmässiga målsättning möter vad de flesta utvecklarna vill, att nå rätt produkt på rätt sätt, men frågan är hur RUP förhåller sig i en systemutvecklingsmiljö?

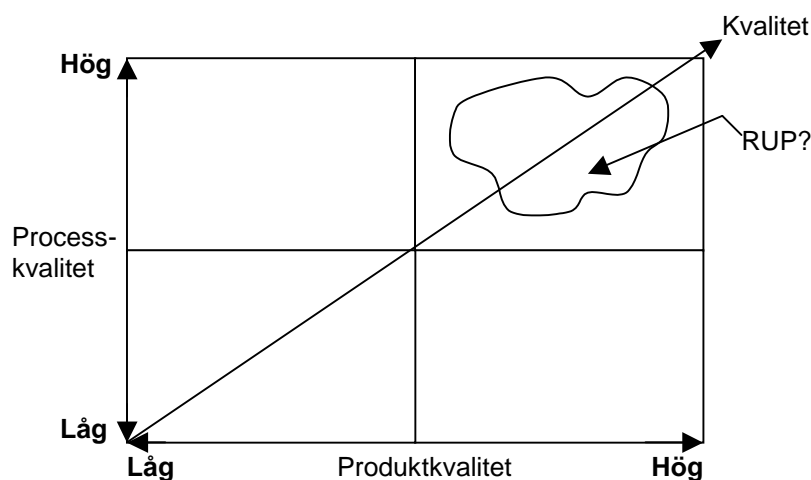
## 1.2 Syfte och utredningsstrategi

Syfte med denna uppsats är att på ett systematiskt sätt skapa ett beslutsunderlag för bedömning av RUPs kvalitetsfrämjande värde. Med systematik menar jag skapande av en s.k. metaarkitektur som sammanfattar systemutvecklingens dimensioner och krav på kvalitet. Därmed, min utredningsstrategi består av tre steg:

1. Sökande efter **relevanta kunskaper och erfarenheter om systemutvecklings kvalitetskrav och önskemål.**
2. Användning av dessa kunskaper för att **skapa en vägvalsmodell för att bedöma RUPs lämplighet inför valet**

3. **Validering av såväl RUPs lämplighet, som vägvalsmodellens relevans för andra liknade situationer.** På detta sätt uppnår min uppsats ett praktiskt syfte och en teoretisk mer akademiskt syfte.

Arbetets intentioner och utgångspunkt syns i nedanstående figur 1.3. Vanligtvis definieras kvaliteten i systemutvecklingen i termer av de inbördes förhållandena mellan produktkvalitet och processkvalitet. Därmed blir frågan om var någonstans RUP ligger intressant. Enligt min mening krävs det, för att kunna få en tydligare bild, både en teoretisk och empirisk studie av såväl systemutvecklingsprocessen som systemutvecklingsprodukten utifrån ett kvalitetsmässigt perspektiv. Vidare blir förhållandet mellan process och produkt systemisk eftersom den representerar två olika intressen, nämligen IT-användarens intressen och IT-utvecklarens intresse. Förhållandet mellan produkt och process utgör tillsammans en systemutvecklingsmiljö, och är intressant då den visar ytterligare ett kvalitetsområde nämligen miljö kvalitet<sup>2</sup>. (Paashuis, 1997)



Figur 1.3: Utgångspunkt

### 1.3 Huvudfrågeställning och avgränsning

Arbetets huvudverksamhet kan sammanfattas i en representativ fråga, nämligen:

**Hur en systemutvecklingsmodell i allmänhet och RUP-modellen i synnerhet uppfyller den kvalitetsbild som berör systemutvecklingens tre grundpelare nämligen process, produkt och miljö?**

Såsom det framgår ovan, har mitt arbete fokuserat på och belyser hur RUP-modellen förhåller sig till behovsbilden som härleds från dagens professionella uppfattningar om processen, produkten och miljön. (grundpelarna beskrivs mer i kapitel 2 och 3)

Därmed aktualiseras följande detaljerade frågor som har avgränsat och styrt utredningens verksamhet.

**Hur avgör vi på ett systematisk sätt RUP-modellens kvalitativa värde?**

- Vilka vitala **kvalitativa egenskaper kännetecknar systemutvecklingsprocessen**, samt vilka av dessa främjas av RUP? Dvs. kan RUP försäkra oss om att vi håller på med rätt systemutvecklingsprocess?

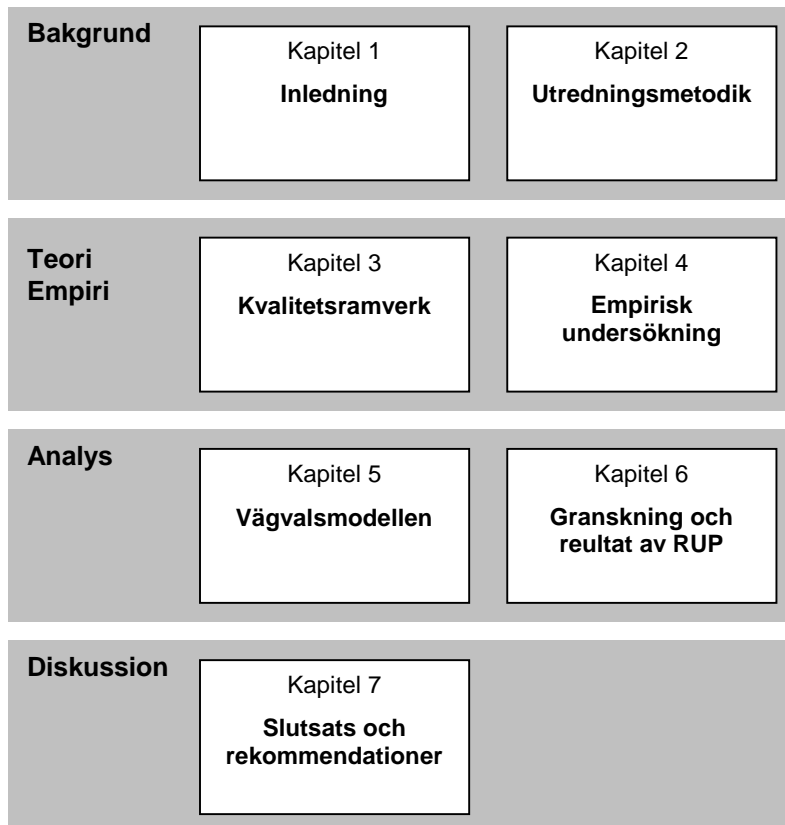
---

<sup>2</sup> Process, produkt och miljö definieras i kapitel 3.

- Vilka **väsentliga kvalitativa egenskaper associeras till systemutvecklingens produkt** samt vilka av dessa främjas av RUP? Dvs. kan RUP försäkra oss om att vi utvecklar rätt informationssystem?
- Vilka **kritiska faktorer karakteriserar systemutvecklingsmiljöns attraktivitet** samt vilka av dessa främjas av RUP? Dvs. kan RUP försäkras oss om att vi har rätt social arbetsmiljö?

## 1.4 Disposition

Frågeställning och syfte ledde fram till att uppsatsen fick följande disposition. (figur 1.4)



**Figur 1.4:** Disposition



## 2. Utredningsmetodik

Min utredningsstrategi består av tre steg:

1. **Sökande efter relevanta kunskaper och erfarenheter om systemutveckling** utifrån ett kvalitetsperspektiv
2. **Skapande av en vägvalsmodell** för att bedöma RUPs lämplighet inför valet, samt
3. **Validering av såväl RUP, som vägvalsmodellens lämplighet respektive relevans.**

Dessa tre delar beskrivs i avsnitt 2.1-2.3, och i avsnitt 2.4 beskrivs den utredningsmetod som skapats för att stämma överens med utredningsstrategin.

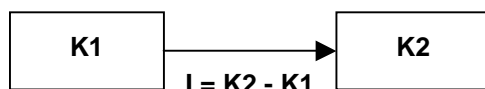
### 2.1 Sökande efter relevanta kunskaper och erfarenheter om systemutveckling utifrån ett kvalitetsperspektiv

Denna aktivitet syftar till att fylla kunskapsbehovet som relateras till utredningens syfte. Syftet har givit en klart inriktning till sökandet nämligen, kunskaper och erfarenheter som belyser följande frågor, som beskrivits i 1.3:

- Vilka **vitala kvalitativa egenskaper (t.ex. effektivitet och synlig) kännetecknar systemutvecklingsprocessen?**
- Vilka **väsentliga kvalitativa egenskaper (t.ex. användbarhet och korrekt) associeras till systemutvecklingens produkt?**
- Vilka **kritiska faktorer karakteriserar systemutvecklingsmiljöns attraktivitet (t.ex. mål och människor)?**

Mina förväntningar är att genom litteraturstudier och genom intervjuer med systemutvecklare kunna uppfylla det behov av kunskaper som behövs för att skapa en vägvalsmodell för val av metoder och modeller, såsom RUP, för att stödja en utvecklingsprocess.

Bilden nedan (figur 2.1) ger en lämplig reflektion av mina intentioner med utredningens syfte, nämligen att ge en uppdaterad och fullständig bild som syftar till att klargöra modellernas (RUPs) lämplighet utifrån ett kvalitetsperspektiv.



**K1** Kunskap innan studie

**K2** Kunskap efter studie

**I** Den utökning av kunskap som studien gav

**Figur 2.1:** Intentioner bakom min utredning

#### 2.1.1 Tidigare studier

Studier som berör metodval, metodutvärdering, metodjämförelser etc. är inte främmande inom informatik. Informatik som vetenskap sysslar med belysning av frågor som berör design, användning och management av informationsteknologi. Metoderna av olika slag härleds från olika teoretiska modeller som kan ses som informatikens vetenskapliga produkter, men även olika slags konsulter skapar egna modeller och metoder som dokumenterar deras erfarenheter från olika branscher hörande till informatik. Därmed utgör studier av modeller, metoder, tekniker, verktyg, etc. och deras lämplighet för olika faser av systemutvecklingen en väsentlig del av informatikens verksamhet.

Vanligtvis handlar systematiska kunskaper om hur man skapar, klassificerar, värderar, jämför, bedömer, väljer och inför metoder och modeller, frågor om metodläran förekommer under benämningen metaarkitektur. Det finns ett antal studier om metaarkitekturer:

- Mylopoulos (1998), har presenterat en metaarkitektur för **klargörande av modellerings teknikens lämplighet** i termer av begrepp, struktureringsmekanismer samt modelleringsaktiviteter såsom analys, design och management.
- Avison och Fitzgerald (1998) har **sammanställt olika kvalitetsfrämjande faktorer som bör karakterisera såväl process som produkt**, men skillnaden mellan Mylopoulos och Avison är att i Mylopoulos studie berörs även systemutvecklingsmiljö.
- Molina, Kusiaki och Sanchez (1998) har **studerat metodernas verksamhet och system utifrån livscykel perspektivet** dvs. fullständig krav.
- Bernus, Nemes och Williams (1996) har **skapat en vägvalsmodell**, en så kallad metaarkitektur, för utvärdering av systemutvecklingens och verksamhetsutvecklingens modeller t ex. Cimoso, Pera och Grai. Denna metaarkitektur baseras på IT-leverantörens perspektiv och berör metoder, modellerings teknik och miljö aspekter mm. Idéen om morfologi<sup>3</sup> har hämtats här ifrån.
- En intressant studie som jag har tagit hänsyn till i min utredning är Enquist (1999) **beskrivning av metarkitekturen som ett verktyg för att samordna IT-leverantörernas och IT-användarnas intressen, perspektiv och kvalitetsbilder** etc.

### 2.1.2 Kvalitetsfrågor

Kvalitetsorientering gör sökningsproblemet ännu mer problematiskt. Kvalitet har en hård objektiv sida som kan mätas i termer av funktionalitet och reliabilitet dvs. de krav som kunden vanligtvis kan specificera. Den har även en mjuk subjektiv sida som kunden grundar på sina upplevelser efter leveransen. Det har i sin tur att göra med det estetiska och symboliska, som inte kan beskrivas innan kunden har fått känna och använda produkten (Dahlbom, Mathiassen, 1993). Jag hoppas att i och med kontakter och intervjuer med systemutvecklare göra denna mjuka kvalitets sida klarare.

## 2.2 Skapande av en vägvalsmodell för att bedöma RUPs lämplighet inför valet

Systemutveckling, oberoende om den är kvalitetsfrämjande eller produktivitetsfrämjande, är en process bestående av många komplicerade, överblickbara och osäkra faktorer. Kraven att halvera utvecklingstiden och minimera kostnaderna utan att påverka kvalitetsmålen gör situationen ännu mer komplicerad.

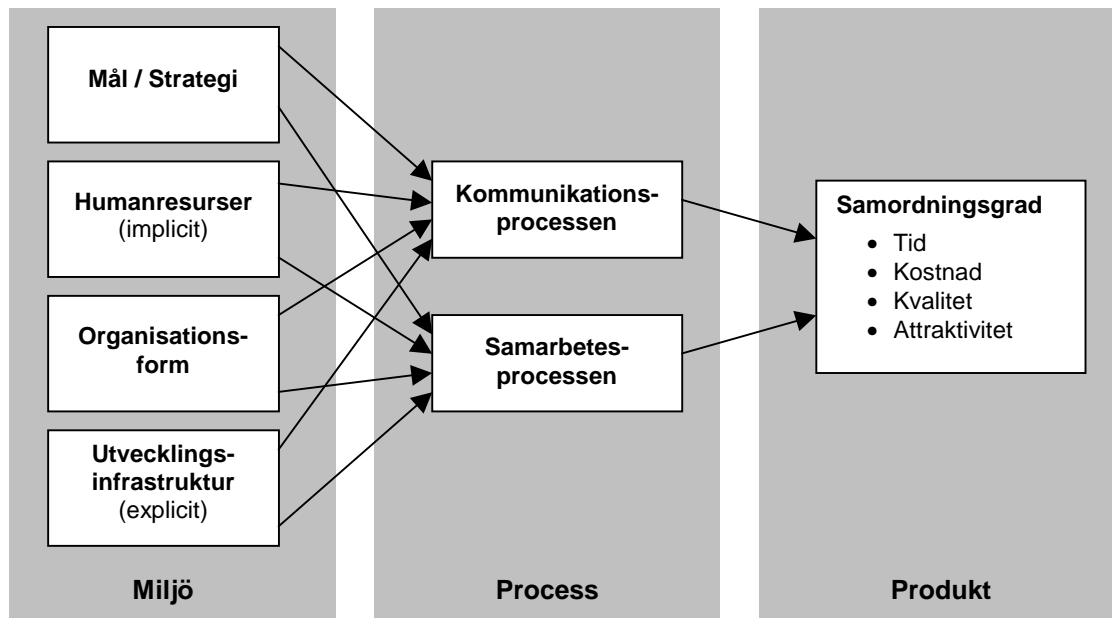
RUP-modellen fokuserar på såväl processen och produkten utifrån ett kvalitetsperspektiv. Det är just denna kvalitetsorientering som motiverar granskningen av RUP-modellen. Men kan vi göra en sådan granskning mer systematisk? Kan vi skapa en representativ beslutsunderlag utifrån teoretiska argument och empiriska upplevelser. M.a.o.:

---

<sup>3</sup> Fokus på att klargöra möjliga alternativ som relateras till ett visst förhållande, dimension eller attribut. (Khosrowpour, 1994)

## Hur avgör vi på ett systematiskt sätt RUP-modellens kvalitativa värde?

Paashuis, (Paashuis, 1997) har i sin studie om integrerad produktutveckling presenterat en modell (figur 2.2) som har påverkat mitt arbete och lett mig fram till de frågor som varit viktiga för såväl litteraturstudie som den empiriska utredningen. Modellen har hjälpt mig att avgränsa arbetet och stöttat mig vid den teoretiska inläsningen.



Figur 2.2: Modell (Paashuis, 1997)

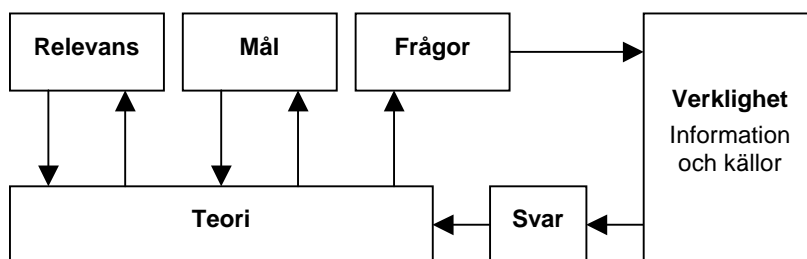
Modellen visar vad som är avgörande för om ett projekt blir ett fiasko eller en succé. En succé kan anges i termer av effektiv samordning mellan olika aktiviteter som utförs av människor som är ömsesidigt beroende av varandra. Samordningseffektivitet kan anges i termer av tid, kostnad, kvalitet och attraktivitet. I kapitlet 3 kommer denna modell att beskrivas utförligare.

### 2.3 Granskning av RUPs lämplighet och vägvalsmodellens relevans för andra liknade situationer

Efter konsultation med min handledare bestämde jag mig för att använda modellen (figur 2.2) för val av de mer detaljerade frågor som skall användas i den teoretiska och empiriska undersökningen. Modellen lämplig och relevans framgår från mina utredningsfrågor nedan:

- **Kan RUP försäkra oss om att vi utvecklar rätt systemutvecklingsprocess?**
- **Kan RUP försäkra oss om att vi utvecklar rätt informationssystem?**
- **Kan RUP försäkrar oss om att vi har rätt social arbetsmiljö?**

Vidare är min förhoppning att den vägvalsmodell som kap 3 och 4 utmynnar i inte skall bli engångsföreteelse. Istället hoppas jag att mitt arbete skall bli underlag för vägvalsmodellens vidare utveckling, så att den kan användas praktisk i liknade situationer. Min strävan är att definiera förutsättningar för modellens generaliserbarhet.

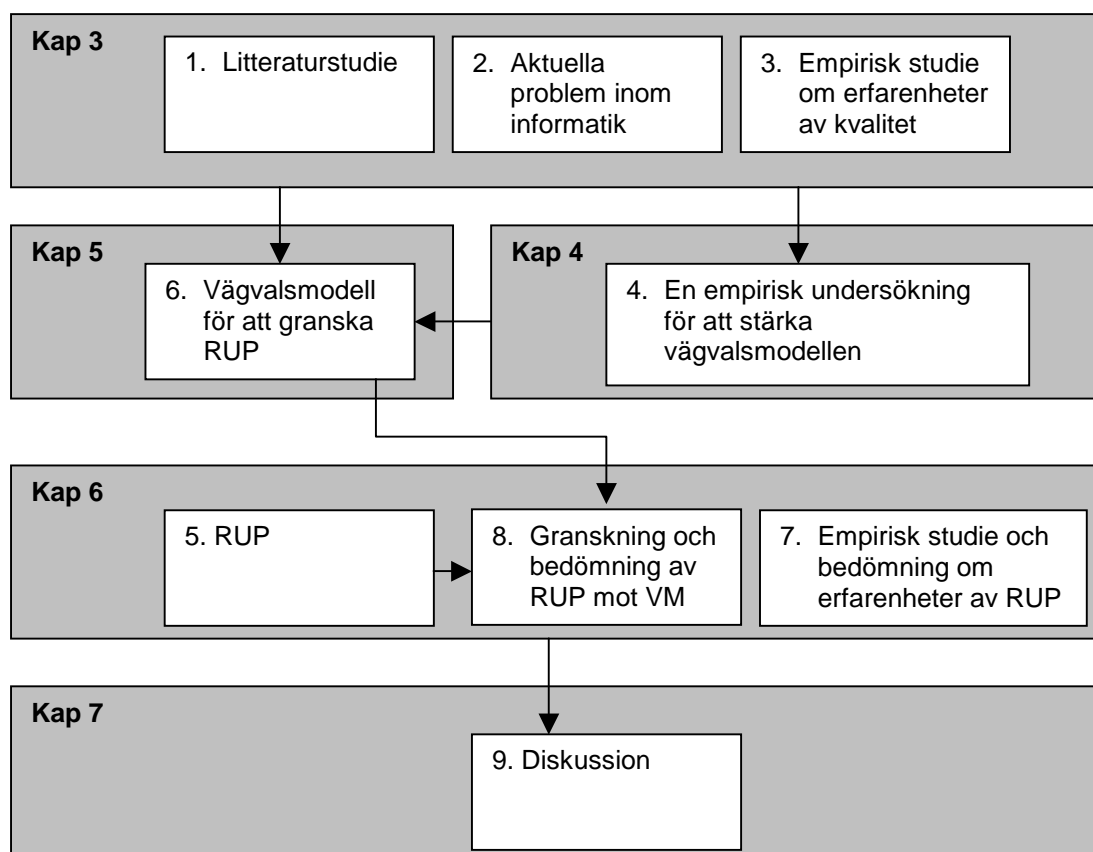


Figur 2.3: Studiesätt

Figur 2.3 ovan visar mitt sätt att bygga upp min vägvalsmodell. Först med utredningsmålets stöd har jag valt en relevant men grov modell för litteraturstudie, sedan med hjälp av modellen och litteraturstudie har jag försökt få fram lämpliga utredningsfrågor. På detta sätt har jag säkrat utredningens validitet. Däremot är utredningens reliabilitet och generaliserbarhet beroende av urvalet av de personer som jag har intervjuat. Till sist utgör intervjumaterialet grunden för den valda modellens kvalitet (figur 2.2).

## 2.4 Utredningsmetod

Utvecklingsstrategin ovan leder fram till figur 2.4 som beskriver hur hela utredningen har gått till.



Figur 2.4: Utredningsmetod

### Min utredningsmetod:

- Jag startade med att försöka skapa mig en bild av begreppet kvalitet. Målet var att dels komma fram till relevanta frågor om kvalitet inför intervjuerna, och dels att skapa en grund för vägvalsmodellen. Kvalitetsramverket växte fram genom inläsning

följt av diskussion med handledare, därefter komplettering av litteratur och ny inläsning och ny diskussion osv. tills alla ansåg att vi uppnått en bas som vi kunde vara överens om. **(1, 2 och 3** i figur 2.4 ovan)

- Utifrån kvalitetsramverket fick jag fram ett antal intervjufrågor för att dels komplettera vägvalsmodellen, och dels validera teorin. På grund av det begränsade antalet veckor som stått till förfogande har en avgränsning och prioritering varit tvungen. De frågor som dock används bör vara tillräckliga för att den empiriska undersökningen skall vara av relevans.

Intervjuerna utfördes både på Enatoranställda (4 personer) och icke Enatoranställda (4 personer) för att få en högre relevans i undersökningen. De icke Enatoranställda är människor som arbetar inom andra företag där det kan finnas behov av stöd för utvecklingsprocessen **(4)**

- Nu ansåg jag att bilden var så pass tydlig att jag kunde skapa en vägvalsmodell med fokus på process-, produkt- och miljö kvalitet. **(6)**
- För att kunna använda vägvalsmodellen krävs stor kunskap om utvecklingsmodellen (i detta fall RUP), som skall utvärderas. Jag försökte genom litteraturstudie, internet och en djupintervju med Pär Jansson på Rational skapa mig det. **(5)** Denna relativt objektiva bild användes vid utvärderingen av RUP mot vägvalsmodellen, och resultatet blev en metaarkitektur. **(8)**

Jag utförde även en empirisk studie av RUP, för att komplettera den objektiva bilden av RUP. Undersökningen utfördes på företag som redan använder sig av RUP, genom att de fick chans att komplettera den teoretiska bilden. **(7)**

Resultatet av utvärderingen blev en bild av RUPs klara och oklara sidor utifrån en objektiv bedömning **(5)**, samt en validering av resultatet genom den subjektiv bedömning (från användare av RUP) **(7)**.

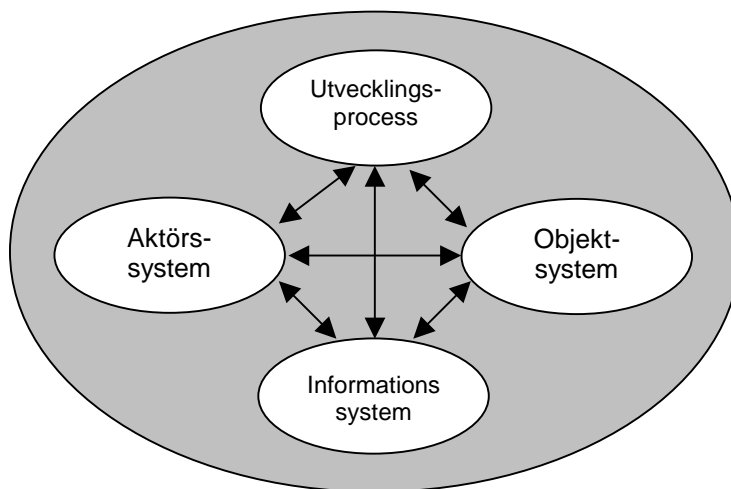
- Slutligen försöket jag föra en diskussion **(9)** kring huvudfrågan:

**Hur en systemutvecklingsmodell i allmänhet och RUP-modellen i synnerhet uppfyller den kvalitetsbild som berör systemutvecklingens tre grundpelare nämligen process, produkt och miljö?**

### 3. Kvalitetsramverk

Den verklighet som systemutvecklaren befinner sig i kan beskrivas i termer av fyra världar och deras inbördes förhållande (se figur 3.1):

- Den värld av **objekt, händelser och processer** som representeras i informationssystemet.
- Den värld av **informationssystem** som redan finns och består av såväl datorbaserade som ickedatorbaserade system. (Produktvärld)
- Den värld av **aktörer och intressenter** som påverkar och påverkas av såväl informationssystemet som systemutvecklingen (användarvärld).
- Den värld av **processer** som relateras till datorer, utveckling, vidareutveckling och avveckling etc. av informationssystemet (utvecklarevärld).



**Figur 3.1:** Fyra världar (Mylopoulos, 1998, Roland, 1998)

Ovanstående föreställning kan ses som "the state of the world", alltså den professionella uppfattningen inom informatik, och därmed refererar kvalitetsfrämjande faktorer till såväl process, produkt (informationssystem) som miljö (objektsystem och aktörssystem). Denna professionella uppfattning har skapats på 90-talet, och innan dess fanns det bara tre världar (processen utesluten). Dessa världar är beroende av varandra och den kvalitet som finns bland de fyra världarna är som följer:

- **Systemisk kvalitet** finns mellan aktör och informationssystem. Denna kvalitet berör mjukt systemtänkande och handlar om begriplighet och bekvämlighet mm.
- **Strukturell kvalitet** finns mellan informationssystem och utvecklingsprocessen. Den berör flexibilitet och säkerhet mm.
- **Funktionell kvalitet** finns mellan informationssystem och objektsystem. Den handlar om funktionalitet.

Om någon värld sidosätts kan detta inte ske utan att skada kvaliteten, vilket ger oss följande ekvation:

$$\text{Arkitekturell kvalitet} = \text{Funktionell kvalitet} * \text{Strukturell kvalitet} * \text{Systemisk kvalitet}$$

I figur 2.2 beskrivs den modell som jag har valt skall leda fram till dels de empiriska frågorna och dels de områden som skall beröras i teorin. Figuren visar på ett tydligt sätt de områden, eller som ovan de världar, som en modell (RUP) bör stödja.

För att kunna handskas med kraven på kvalitet så ser sig verksamheter om efter modeller för att stötta utvecklingsprocessen, men hur vet verksamheten att modellen de väljer passar? Enligt min utredningsstrategi är målet med kapitel 3 och 4 att skapa en vägvalsmodell för att kunna välja rätt modell för att stödja processen, och i detta kapitel försöker jag beskriva de olika delfrågorna (från avsnitt 2.3) för att kunna skapa en relevant sådan.

Avsnittet kommer att börja med att beskriva processen, för att sedan beskriva produkten och slutligen beskriva miljön.

### 3.1 En analys av begreppet process

I detta avsnitt kommer begreppet systemutvecklingsprocess i allmänhet och mjukvaruutvecklingsprocess i synnerhet att redogöras. Min strävan är att genom litteraturstudie klargöra de avgörande faktorerna som påverkar processens kvalitativa egenskaper. Studien styrs därmed av utredningens första delfrågan, nämligen:

#### **Vilka vitala kvalitativa egenskaper kännetecknar systemutvecklingsprocessen?**

Klargörande av begreppet och dess karakteristiska egenskaper presenteras i följande delavsnitt:

1. **Olika föreställningar av begreppet process**
2. **Processens målbild**
3. **Processens utformning**
4. **Förhållande mellan process och produkt**
5. **Processens bakomliggande filosofi**

#### 3.1.1 Olika föreställningar av begreppet process

Begreppet process i allmänhet refererar alltid till något pågående i verksamheten, därmed saknar begreppet betydelse utan referens till antingen det objekt som verksamheten sysslar med, eller till den entitet som bedriver själva verksamheten. Vanligtvis kan varje process anges i termer av tillståndsförändringar associerade till ett och endast ett objekt i system eller entitet. Som exempel på processer, kan omvandlingen av en kreativ idé till en attraktiv fysisk produkt eller framställningen av ett mjukvarusystem från en given systemspecifikation, nämnas. På samma sätt kan man betrakta människans kunskapsutveckling och mognad, och i den bemärkelse är en process en komplicerad händelse som alltid består av fler och mer elementära händelser, eller en komplicerad verksamhet bestående av fler mer elementära handlingar.

Det finns en rad olika attribut som associeras till begreppet process. Följande lista utgör ett enkelt representativt uttryck om hur vi brukar karakterisera en process.

- Naturlig/artificiell
- Rationell/irrationell
- Kreativ/rutinmässig
- Kontinuerlig/diskontinuerlig
- Känsломässig/känslökall
- Samordnad/oorganiserad
- Enkel/komplicerad
- Målmedveten/omedveten
- Konceptuell/fysisk

- Motiverad/omotiverad
- Homogen/heterogen
- Statisk/dynamisk

Men vad menas egentligen med begreppet systemutvecklingsprocess, och mer bestämt en mjukvaruutvecklingsprocess?

Enligt min utredning finns det inte någon enhetlig standarddefinition. Vad som istället finns är olika föreställningar som tillsammans ger en bra reflektion av begreppets innehåll och referens. Här följer tre sådana definitioner av en process:

*"The software process is a set of activities and associated which produce a software product."* (Sommerville, 1997)

*"The process allows the method to be scaled up, so that it can be applied to projects with many interacting activities and parties."* (Jacobson, 1996)

*"Mjukvaruprocess innebär de verktyg, metoder och övningar som vi använder för att producera mjukvaruprodukter. Processen måste se till relationer mellan verktyg, metoder, tekniker under användning och kunskap, övning och motivation av människor involverade."* (Humphrey, 1990)

Enligt definitionerna ovan kan en mjukvaruprocess karakteriseras av följande aspekter:

- Själva **verksamheten**, dvs. ett antal aktiviteter som kan associeras med mjukvaruproduktion.
- Verksamhetens **teknologi**, dvs. ett antal verktyg, metoder och olika slags exempel/övningar som används för att organisera och strukturera verksamheten och därmed producera mjukvaruprodukter.
- **Relationerna mellan processen och teknologi**, dvs. verktyg, tekniker och metoder.
- **Relationerna mellan processen och människor** som besitter de rätta kunskaperna och som är välmotiverade.

Alltså är den företeelse som i litteraturen kallas mjukvaruverksamhet alltid en strukturerad och väldefinierad process som producerar en väldefinierad/specificerad mjukvaruprodukt. Det innebär att:

1. Kreativiteten och intuitionen har lämnat plats till en **rutinmässig och formaliserad verksamhet**.
2. Arbetets natur och kultivering har ersatts av **metoder och tekniker**.
3. Produktens komplexitet och variation har absorberas genom nedbrytning av produkten till ett antal **homogena och enklare komponenter**.

En väsentlig egenskap hos alla strukturerade processer är att de tillåter en fullständig specificering av verksamhetens beståndsdelar, dvs. processen blir definierad i termer av en metod. Denna aspekt kan behöva ett exempel.

Att utveckla en ny medicin i ett laboratorium skiljer sig från att utveckla medicinen i tillverkningsverksamhet (se figur 3.2 nedan). I laboratoriet är kreativitet och intuition en viktig del av arbetet och personalen prövar sig fram explorativt. Målet är att hitta en ny produkt som kan användas. Ibland lyckas forskarna och ibland inte, men när de lyckas så kan de inte lämna ifrån sig produkten i det skicket, utan måste generalisera, standardisera och formalisera en metod för att kan garantera att produkten blir rätt på ett snabbt och effektivt sätt. I tillverkningsverksamheten är det alltså nödvändigt att ha en metod för att undvika att arbetsuppgifter utförs på ett ineffektivt sätt, t.ex. personalen vet inte vilka uppgifter som de



skall göra vilket leder till att vissa gör samma sak (dubbelarbete) och andra gör ingenting. Sist men inte minst förutsätter en fullständig specificering av processens struktur att denna typ av verksamhet är förekommande och därför användbar på flera liknade projekt.

Process	Laboratorieverksamhet	Tillverkningsverksamhet
Formaliseringsgrad	Låg	Hög
Standardiseringsgrad	Låg	Hög
Samordningsgrad	Låg	Hög
Variationsgrad	Hög	Låg
Struktureringsgrad	Låg	Hög
Generaliseringsgrad	Låg	Hög
Pålitlighetsgrad	Låg	Hög
Stödandegrad	Låg	Hög

**Figur 3.2:** Laboratoriet kontra tillverkningsverksamheten

Sammanfattningsvis representerar en strukturerad och välspecificerad process människans erfarenheter och förmåga att göra någonting. I den meningen utgör processen en modell eller en metod och som sådan utgör den alltid en ofullständig avbildning av människans erfarenheter och kompetens. Det finns alltså alltid kunskap som inte kan synliggöras.<sup>4</sup>

### 3.1.2 Processens målbild och kvalitetsmål

#### 3.1.2.1 Processens målbild

Processens målbild kan beskrivas i termer av effektivitet och produktivitet.

Effektivitet kan i sin tur delas upp i

- **Kundens kvalitet**
- **Professionell kvalitet**

För kunden är funktionalitet ett viktigt kvalitetsbegrepp, medan däremot flexibilitet är något som de är ganska ointresserade av. Kunden är intresserad av produktens utseende och inte vägen till det.

Professionell kvalitet har med systemutvecklare att göra och de vill att processen är flexibel, eftersom det finns många risker med att bara tänka på funktioner. Riskerna som finns är bland annat krav som förändras, kunden har ofta inte hela målet klart när processen startar, oväntade problem som kan uppstå, genom att t.ex. någon i under processens gång blir sjuk, och problem med tekniska begränsningar som förhindrar målet att gå i uppfyllelse.

För att komma tillrätta med riskerna och nå flexibilitet används produktlivscykeln. Genom att tänka på den när en produkt skapas så finns hela tiden vidareutveckling och riskerna ovan i tankarna.

<sup>4</sup> Här kan nämnas några intressanta teorier om processens natur såsom t.ex. (1) H. Simons arbete om välstrukturerade vs ostrukturerade beslutsprocesser, (2) N. Macintosh klassificering av processer i rutinmässiga, professionella, intuitiva och forskningsliknande, (3) J. Thompsons teori om standardiserade, seriekopplade respektive ömsesidigt beroende processer, samt den hårda systemvetenskapliga skolan som sammanfattas i begreppen systemanalys, systemkonstruktion och operationsanalys. Strävan har varit att ersätta den intuitiva, kreativa känslomässiga tänkande med en rationell, rutinmässig och individoberoende verksamhet. (Dahlbom, Mathiassen, 1993)

Produktivitet delas in i:

- **Tid**
- **Kostnad**

Dessa delar är viktiga både för kunden och utvecklarna och är ofta en fråga som det måste kompromissas om. Kompromissen gäller att tid är pengar och tid är antagligen bättre kvalitet. Var någonstans skall ribban läggas?

En summering av det ovan nämnda ger en processmålbild, vars intresse är att stödja leverantören till att hjälpa kunderna i en föränderlig miljö.

### 3.1.2.2 Processens kvalitetsmål

Nedan följer en tolkning av mjukvaruprocessens kvalitetsmål:

*”Målet med mjukvaruprocessen är att undvika att fel process används, och istället för att efter intuition utveckla produkter producera dem efter en plan. Processen skall vara ett stöd för hur kvalitet på människor uppnås och hjälpa människor att kommunicera i gemensamma termer och därigenom stödja till att rätt produkt på rätt sätt uppnås.”*  
(Humphrey, 1990)

Kvalitetsmålet innebär alltså att utveckla rätt produkt på rätt sätt och Watts Humphrey säger ovan att för att nå detta krävs det en plan, människor och kommunikation. Ann Hägerfors anser detsamma när hon beskriver att hög kvalitet på process uppnås genom att (Hägerfors, 1995):

- **Välja rätt tekniker och metoder** för att utföra arbetet så att det överensstämmer med situationen: organisationen, personer, och mål o.s.v.
- **Tekniker och metoder används på ett kompetent sätt.**
- **Fördela arbetet** på ett bra sätt med hänsyn till tillgängliga personer.
- **All behövlig kompetens finns tillgänglig.**
- Personerna kan **arbeta och att trivas tillsammans** (socioemotionella aspekter).

Dock är det inte lätt att uppnå kvalitetsmålet då det finns ett antal faktorer som påverkar processen, Håkan Dyrhage från Rational ger här exempel på några:

- **Budget- och tidsramar**
- **Existerande process och eventuella problem**
- **Personalens kompetens, erfarenhet och attityder**
- **Typ av applikationsdomän**
- **Relation mellan kund och IT-leverantör**
- **Intressenter** (eng. ”stakeholders”) internt och externt.
- **Tekniska problem och risker**

För att komma tillrätta med dessa faktorer har ett antal kriterier/attribut definierats för att stödja utvärderingen av processen i strävan mot kvalitet. Vissa av attributen nedan motsäger varandra och det är alltså inte möjligt att maximera alla attribut.

Ett antal kriterier(Hägerfors,1995):

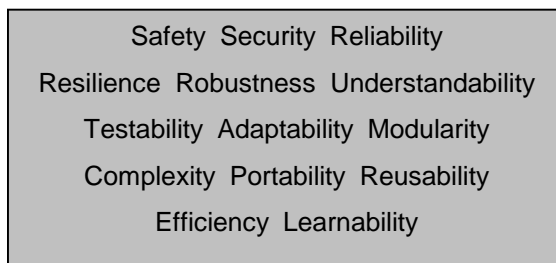
- **Verkningsgrad.** Fungerar medlen, t.ex. valda tekniker? Nås den förändring som önskas med dessa medel? Fungerar processen? Resulterar processen i den önskade produkten?

- **Effektiv.** Används minimala resurser? Finns det andra sätt att åstadkomma förändringar som är billigare, t.ex. andra tekniker? Är produkten effektiv?
- **Effektfull.** Leder medlen till uppnående av långsiktiga mål?
- **Etik.** Är det moraliskt rätt att göra denna förändring? Uppfattas processen som bra eller dålig? Uppfattas produkten som bra eller dålig?
- **Elegans.** Tilltalar denna förändring vårt skönhetsinne? Leder produkten till att verksamheten fungerar bättre och smidigare? Är processen utformad på ett tilltalande sätt? Kan produkten kallas behaglig?

Värdering av processer (Sommerville, 1997):

- **Understandability.** Hur lätt är det att förstå processens definition?
- **Visibility.** Kulminerar processens aktiviteter i klara resultat så att framstegen är externt synliga?
- **Supportability.** Kan process aktiviteten stödjas av Case-verktyg, och till vilken grad?
- **Acceptability.** Är processen accepterad och användbar för ingenjörerna som ansvarar för produktionen?
- **Reliability.** Är processen så konstruerad så att krav tas om hand innan det blir produktfel?
- **Robustness.** Kan processen fortgå fastän nya problem dyker upp?
- **Maintainability.** Kan processen utvecklas för att reflektera förändringar i organisationskrav eller process förbättring?
- **Rapidity.** Hur snabbt kan processen utveckla ett system från en specifikation?

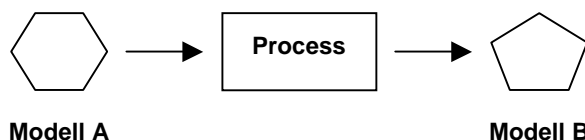
I figuren nedan följer ytterligare ett antal attribut (Jacobson, 1996)



**Figur 3.3:** Processkaraktärer (Jacobson, 1996)

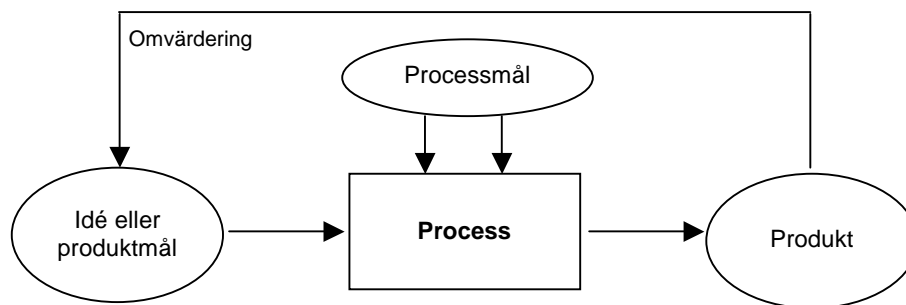
### 3.1.3 Processens utformning

En process utformning handlar, som beskrivits ovan, om en transformation från en idé till en produkt, från ett objekt i systemet till ett annat, eller information in och information ut. Hur denna transformation sker beror på hur en verksamhet bestämmer sig att organisera den. Figur 3.4 visar hur transformationen kan beskrivas.



**Figur 3.4:** Process transformation, en modell till en annan (Jacobson, 1996)

Processen kan också ses i form av en produktlivscykel, och då får den ett annorlunda utseende. I figur 3.5 visas hur utdata ur processen omvärderas och om produktmålet inte är uppfyllt så tar man ett varv till i processen.



Figur 3.5: Mjukvaruprocess

För att lyckas uppnå produktmålet och processmålet kan processen ta hjälp av diverse modeller för att stödja processen.

Processen är inte statisk utan under ständig utveckling och det finns ett antal principer att tänka på vid processförändring (Humphrey, 1990):

- **Större förändringar av mjukvaruprocessen måste börja ovanifrån**, eftersom det är där som regler och prioritet sätts.
- **Alla måste involveras**. Vid förändring kan människor känna sig hotade, därför är det viktigt att de involveras och känner sig delaktiga.
- **Effektiv förändring kräver mål och medvetenhet** om den nuvarande situationen.
- **Förändring är ständigt pågående**.
- **Förändringar av processen kommer inte att bestå**, utan alla måste kämpa för att ha den kvar.
- **Mjukvaruprocesser kräver investeringar**.

I laboratorieexemplet försöker forskaren efter att ha forskat fram en produkt skapa en flexibel modell att stödja processen så att den klarar en föränderlig miljö.

### 3.1.3.1 Kommunikation och samarbetsprocessen

Vid en närmare studie av processens innehåll kan man i figur 2.2 se att processen bör innehålla två delprocesser: en kommunikationsprocess och en samarbetsprocess.

Kommunikationsprocessen är när en eller flera människor kommunicerar, vilket kan ske direkt eller indirekt (via hjälpmedel såsom e-post eller telefon). Att kommunicera är att utbyta information, men att förstå informationen är inte samma sak. T.ex. kan två personer stämma möte via telefon men till mötesplatsen kommer bara den ena personen, vad gick fel? Människorna hade en kommunikationsprocess men de förstod inte varandra.

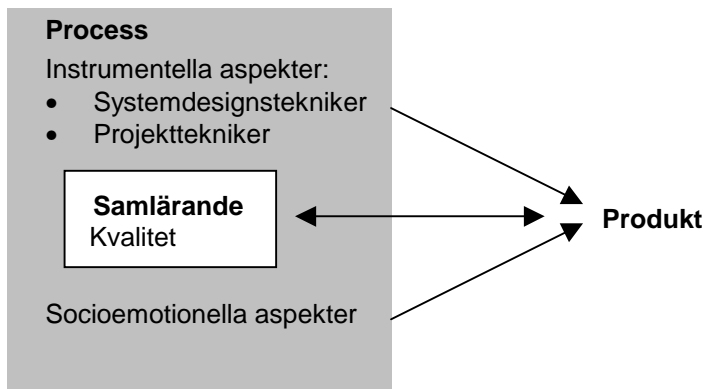
Samarbetsprocessen är att gå ett steg längre, samarbete innebär att man förstår varandra, alltså att se varandras tankar. Samarbete är betydligt svårare att skapa än kommunikation, då det krävs att människorna är intresserade av varandra och vill utbyta och förstå informationen.

Kommunikation och samarbete tar sig olika former i olika företag beroende av utvecklingsmiljön som beskrivs i avsnitt 3.3, då samordningsgraden blir olika beroende på hur väl processen passar i företaget.

### 3.1.4 Förhållande mellan process och produkt

Processen har ett nära samband till produkten, då produktarkitektur måste speglas i processarkitekturen för att kvalitet skall vara möjligt. Med detta vill jag säga att för att processen skall producera rätt produkt så måste processen vara gjord för att producera just sådana produkter. Vid ett val av fel process så kommer det att påverka möjligheterna att nå rätt produkt.

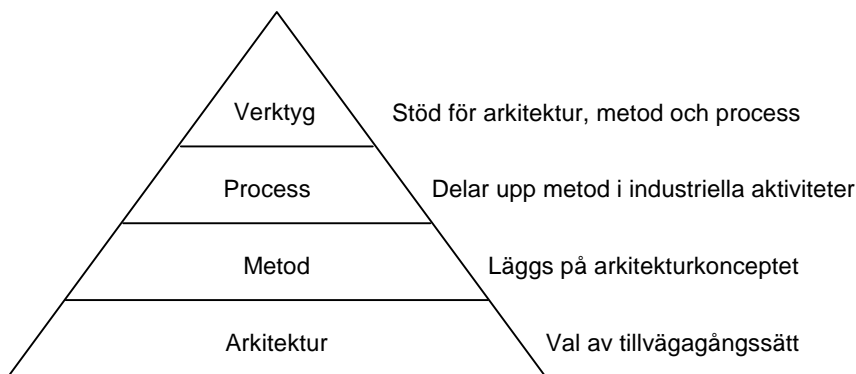
I figur 3.6 beskrivs hur process och produkt måste lära av varandra för att nå fram till målet. Vid lyckade produktutvecklingar sker ibland generaliseringar av processen, så att den kan användas på andra liknade projekt. Problemet vid mjukvarukonstruktion är att projekten varierar i större utsträckning än i produktutveckling, eftersom det innehåller mer socioemotionella aspekter i mjukvarukonstruktion och varje projekt är unikt.



Figur 3.6: Process och produkt relationer (Hägerfors, 1995)

### 3.1.5 Processens bakomliggande filosofi

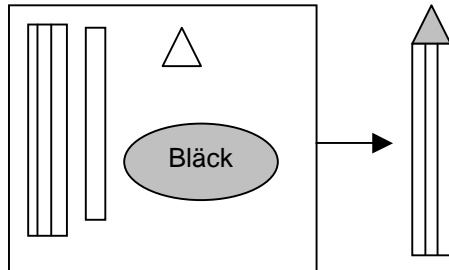
Vid systemutveckling är det viktigt att det finns en filosofi (figur 3.7) som fungerar som guide genom hela utvecklingen. I denna filosofi ingår också process som ett av de fyra nyckelbegreppen. (Jacobson, 1996)



Figur 3.7: Filosofi (Jacobson, 1996)

De fyra bitarna i filosofin är:

- **Arkitekturen** formar hur systemet skall byggas och hur systemet skall behandlas. Ett exempel: En bläckpenna har ett antal grundstenar (se figur 3.8), och utan dessa grundpelare är det ingen bläckpenna. Så länge ingen förändrar grunderna kan tillägg göras t.ex. blå färg på skaftet. Alltså är arkitekturen en ritning eller en modell.



Figur 3.8: Arkitektur

- **Metoden** är en väl planerad procedur med ett visst mål som planeras steg för steg. Den kan liknas vid ett recept. I ett recept finns det både en beskrivning av de resurser som behövs och tillvägagångssättet. En bra metod förenklar utveckling av system och arkitektur.
- **Processen** har redan beskrivits i detta kapitel.
- **Verktyg** stödjer arkitekturen, metoder och processerna.

## 3.2 En analys av begreppet produkt

Detta avsnitt berör den produkten som utvecklingsprocessen skall leverera. Studien styrs av utredningens andra delfrågan, nämligen:

### **Vilka väsentliga kvalitativa egenskaper associeras till systemutvecklingens produkt?**

Klargörande kommer att ske på ungefär samma sätt som klargörandet av begreppet process. Delavsnitten berör:

1. **Olika föreställningar** av begreppet produkt
2. **Produktens målbild**

### 3.2.1 Olika föreställningar av begreppet produkt

Vad är en produkt? Vid fabriksstillverkning kan en produkt vara:

- En **penna** som du köper i en affär har ofta ett ganska litet värde för dig, det är mer en slit och släng produkt. Om pennan inte fungerar så gör man ingen stor sak av det, utan köper en ny.
- En **bil** däremot har ofta ett stort värde för dig pengamässigt, bekvämlighetsmässigt och det är ett köp som ofta tar tid. Bilen är inte bara plåt utan även service, eftersom det i och med köpet av bilen ingår en garanti som ger service om problem uppstår.

En produkt inom systemutvecklingsbranschen är en:

- Icke fysisk produkt som paketeras för att ge sken av att vara fysisk. Detta för att det är lättare för människor att ta till sig fysiska produkter, då de går att känna på och föreställa sig. Produkten är ett system som förhoppningsvis stödjer och underlättar för användarna och har ofta även som bilen service garanti mm.

En bil och ett IT-system kan vara väldigt lika men också väldigt olika. Inom systemutvecklingsbranschen finns det standardsystem och dessa system kan anpassas in i verksamheten, sådana system är inte olik en bil. En bil vänder sig till en viss målgrupp, beroende på vad du är beredd att betala så anpassar sig bilen till viss del till kundens behov (taklucka, bilstereo och läderklädsel).

Olikheten ligger i en systemutvecklingsprocess, där systemet går från idé till produkt. Det är som att en kund skulle komma till Volvo och säga jag vill att ni gör en bil efter bara mina behov. Volvo utvärderar behoven och gör en bil från början till slut som passar dig och det finns servicepersoner som bara tar hand om din bil.

Ut ur en process kommer en produkt som beroende av hur väl kommunikation och samarbetet har fungerat får en viss samordningsgraden vad det gäller:

- **Kostnaden**, hur mycket kostade produkten?
- **Attraktivitet**, hur trivs människor med produkten?
- **Kvalitet**, hur väl tillfredställde produkten kunden?

För kunden är det kvalitet som är det viktiga och för utvecklaren är det tid, kostnad och attraktivitet.

Kvalitet och systemutveckling har inte alltid förknippats med varandra, systemutveckling förknippas av många som något som drar ut på tiden, drar över budget och ofta något som inte blir vad kunden önskar. **Vad innebär kvalitet?** Nedan följer ett antal definitioner i ett försök att sprida ljus om just detta:

- *“The quality of one and the same system is normally judged differently by the actors involved in its production and use.”* (Dahlbom, Mathiassen, 1993)
- *”Kvalitet är det samma som egenskap i filosofiska sammanhang. Demokritos, Newton, Locke och Descartes har traditionellt försökt skilja mellan kvaliteter av olika slag t.ex. primära, sekundära och tertiära, men deras distinktioner mellan dem har varit olika. Primära egenskaper kan vara t.ex. volym, massa, och dessa är oberoende av vad vi varseblir dem. De sekundära är t.ex. röd, grå, och de är beroende av våra sinnesorgans sätt att fungera. Tertiära kvaliteterna kan vara t.ex. harmonisk, balanserad, och är beroende av det primära och sekundära.”* (Nationalencyklopedin, 1993)
- *“Quality has been defined by some to be degree of excellence and by others as a degree of performance to a standard. Modern definition of quality does not require the highest degree of excellence. Instead... meet the determined standard in every respect.”* (Miller, 1995)
- *”Kvalitet betyder i grunden hur något är – varans, tjänstens eller processens egenskaper. Varje organisation och medarbetare måste utifrån sina kunder bestämma vad man skall lägga i begreppet och formulera en definition som passar den egna verksamheten.”* (Blomqvist, Haeger, 1996)
- *”En varas eller tjänsts kvalitet är dess förmåga att tillfredsställa kundernas behov och förväntningar.”* (Nordkvist, 1996, Armstrong, Kotler, Wong, Saunders, 1999)
- *”Alla sammantagna egenskaper hos en produkt som ger dess förmåga att tillfredsställa uttalade eller underförstådda behov.”* (CAMA Konsult AB)

- ”Produkt kvalitet: prestanda, varaktighet, egenskaper, namn, överensstämmelse med specifikationen, nytthet, storlek och utseende. Service kvalitet: gripbarhet, pålitlighet, lätt tillgänglighet, kompetens och empati.” (Aaker, 1995)

Kvalitet är ett mångdimensionellt begrepp som måste betraktas ur en rad perspektiv för att få en heltäckande bild.

Ordet kvalitet bör användas med viss försiktighet eftersom risken för missförstånd är påtaglig.

Det är svårt, och inte alltid ens önskvärt, att hitta en universell kvalitetsdefinition.

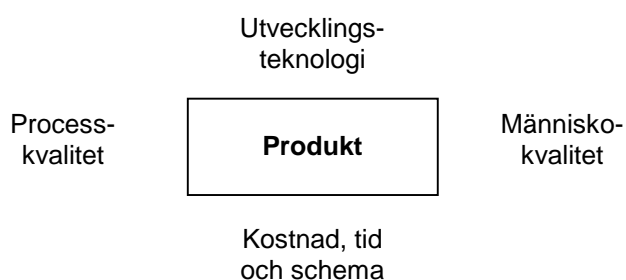
**Figur 3.9:** Ingen universell kvalitetsdefinition (Blomqvist, Haeger, 1996)

Innan vi ser närmare på definitionerna kan det vara bra att ta en närmare titt på figur 3.9. Figuren visar ytterligare en tolkning av kvalitet och drar slutsatsen att det kanske inte är önskvärt att hitta en universell kvalitetsdefinition. Ur definitionerna går det ändå att utläsa följande:

- Det finns **likheter mellan definitionerna** och en viktig sådan är att allt går tillbaka till att kvalitet tolkas olika av olika personer. Kvalitet är alltså någons, det kan vara företagets eller kundens eller er kvalitet.
- **Kvalitet är inte längre att nå högsta nivå** av det utan att nå rätt nivå ut efter vad någon tycker. Kvalitet kan vara att möta en standard.
- Det gäller att försöka **tillgodose kundens behov**.
- Målet för organisationer är att **förutse kvalitet och att sedan möta den**.
- **Produktkvalitet är mer gripbart än tjänstekvalitet**, och alltså är den oftast mer mätbar. Produktkvalitet innehåller alltså mer primära och tjänstekvalitet mer sekundära och tertiära kvaliteter.
- Att **dela upp kvalitet görs för att förenkla** eftersom det är ett komplext begrepp.

### 3.2.2 Produktens målbild

Systemutvecklingsprodukten har två målbildsperspektiv, kundens och leverantörens. Kundens perspektiv bygger på behov och tillfredsställelse, vilket innebär att de vill att produkten skall ha vissa funktioner t.ex. billig eller hjälpa mig med verksamhetens ekonomi. Ur leverantörens perspektiv är deras mål naturligtvis att ge kunden det han vill han, men även att göra kunden uppmärksam på funktioner som han inte ser t.ex. produktens återanvändbarhet.



**Figur 3.10:** Principiella produktkvalitet faktorer (Sommerville, 1997)



Figur 3.10 visar på faktorer som påverkar produkten. Dessa faktorer måste finnas i tankarna när en värdering av en produkt skall göras. Om någon av dessa faktorer är svaga kan det vara skillnaden mellan en bra och en dålig produkt. Värdering av produkt kan ske genom en (Hägerfors, 1995):

- Jämförelse mellan uttalade och outtalade behov och produkten.
- Jämförelse mellan förväntningar på och erfarenheter av produkten.
- Jämförelse mellan kvalitetsbedömning av produkten över tid.
- Jämförelse mellan kvalitetsbedömningar från olika perspektiv.

Det finns ett antal attribut som brukar förknippas med värdering av produkten och i figur 3.11 visas klassiska kriterier för programvara.

Kriterium	Mått på
Användbarhet	Anpassning av systemet till organisatoriska, arbetsmässiga och tekniska omgivningen
Säkerhet	Säkerhet mot oönskade tillgång till systemets data och faciliteter
Effektivitet	Utnyttjande av faciliteter i den tekniska plattformen
Korrekt	Uppfyllande av de uppställda kraven
Pålitlighet	Uppfyllande av den funktionalitet som krävs med precision
Lättunderhållet	Kostnad för att hitta och rätta fel i systemet när det är igång
Lätt testat	Kostnad för att testa systemet gentemot de uppställda kraven
Flexibilitet	Kostnad för att ändra i systemet när det är igång
Begriplighet	Besväret med att skaffa sig överblickbarhet över och förstå systemet
Återanvändbarhet	Användbarheten för att flytta systemet till andra tekniska plattformar
Portabelt	Kostnad för att flytta systemet till andra tekniska plattformar
Integrerbart	Problemen med att sammankoppla systemet med andra system

**Figur 3.11:** Klassiska kriterier för kvalitet hos programvara (Mathiassen, Munk-Madsen, Nielsen, Stage, 1998)

Nedan en lista med produktattribut. I listan motsäger vissa attribut varandra och det är alltså inte meningen att alla dessa attribut skall uppnås (Avison, Fitzgerald, 1998):

- **Acceptability.** Är människorna som kommer i kontakt med systemet tillfredställda med det. Uppfyller det deras behov?
- **Availability.** Är tillgången tillräcklig?
- **Cohesiveness.** Fungerar integration av informationssystemet?
- **Compatibility.** Passar systemet organisationen?
- **Documentation.** Stödjer den kommunikationen?
- **Ease of learning.** Kort lärotid och intuitivt?
- **Economy.** Är det kostnadseffektivt inom resurs gränserna?
- **Effectiveness.** Används systemet på bästa möjliga sätt för att möta överliggande organisationsmål?

- **Efficiency.** Används systemet resurserna på bästa sätt?
- **Fast development rate.** Är utvecklingen snabb relativt till systemets storlek?
- **Flexibility.** Modifieringsgrad?
- **Functionality.** Ombesörjer systemet kraven?
- **Implementability.** Hur lätt skedde implementering?
- **Low coupling.** Kopplingsgrad?
- **Maintainability.** Är det mycket arbete med att hålla systemet igång, och möta det föränderliga kraven?
- **Portability.** Kan annan utrustning användas?
- **Reliability.** Är felen minimerade, och utdata koncistent?
- **Robustness.** Felsäker, Feltolerant?
- **Security.** Säker mot missbruk?
- **Simplicity.** Komplexiteten minimerad?
- **Testability.** Kan system testas under tiden för att minska otillfredsställelse?
- **Timeliness.** Funkar den i alla situationer?
- **Visibility.** Hur är möjlighet att se varför saker händer?

### 3.3 En analys av begreppet miljö

Systemutvecklingsmiljö består av fyra delar:

- **Mål och strategi**
- **Humanresurser** (implicit)
- **Organisationsform**
- **Utvecklingsinfrastruktur** (explicit)

Dessa delar skall skapa förutsättningarna för samverkan och kommunikation. Att tro att det går att uppnå kvalitet genom ett mål, är oftast att missta sig. I ett projekt finns det flera bitar som måste vara en del av helheten för att lyckas.

I detta avsnitt skall delfrågan tre utredas:

#### **Vilka kritiska faktorer karakteriserar systemutvecklingsmiljöns attraktivitet?**

Avsnittet är uppdelat efter de fyra delområdena som finns:

- **Mål och strategi**
- **Humanresurser**
- **Organisationsform**
- **Utvecklingsinfrastruktur**

#### 3.3.1 Mål och strategi

Företag ägnar ofta mycket energi åt att skapa **strategier och mål**. Detta görs för att bestämma långsiktiga mål, anamma ett handlings sätt, och för att kunna bestämma vilka resurser som behövs för att genomföra dem. Strategier kan var olika detaljerade och innehålla ett antal mål. Målen kan vara olika i komplexitet och antal, och det finns ett antal måltyper, allt från outtalade till uttalade, tabulagda till herdvärdiga mål. De mål som finns kan om dem är i

konflikt störa möjligheten till att skapa samordning, eftersom målet jag har som person kanske är i konflikt med det som finns i företaget. (Bolman, Deal, 1997)

För att en strategi och/eller mål skall vara till någon nytta krävs det att den är förankrad. Strategier för IT-organisationer berör kundrelationer, organisationsform, humanresurser, (dvs. systemutvecklarens kunskaper, kompetens, lojalitet och handlingsfrihet etc.) och utvecklingsinfrastruktur (dvs. metoder, tekniker och Case-verktyg etc.).

### 3.3.2 Humanresurser

Humanresurser handlar om implicit kunskap, och är alltså den kompetens någon bär på. Kompetens kan en människa få genom att t.ex. läsa, lyssna eller praktiskt upptäcka något.

I en IT-organisation är kompetens ett viktigt vapen i kampen om kunderna och företag försöker locka till sig rätt kompetens. Om kompetensen saknas måste verksamheten ta ställning till om den skall köpas eller om möjligt vidareutveckla den personal som finns.

IT-organisationer är kända för att ha svårt att behålla kompetens i företaget och detta beror på att det råder brist på personal, vilket leder till höga löner som lockar och möjlighet till att välja nya företag om man vantrivs. De är även kända för att vara dåliga på att hantera kunskap och kompetens, så kallad Knowledge Management.

Eftersom systemutvecklingsprocessen sträcker sig över ett stort område varierar det mycket vad för implicit kunskap som efterfrågas. Vid de tidiga faserna behövs människor som kan lyssna och ställa rätt frågor så att kundens behov utreds. I designfasen behövs människor med stort tekniskt kunnande.

### 3.3.3 Organisationsform

Organisationsform har med utformning att göra. Om t.ex. målet och processen fungerar bra ihop, finns det ändå en stor risk till misslyckande då organisationen kanske inte passar. Mintzbergs fem sektorer är ett känt exempel som berättar om olika strukturer som en organisation kan ha, och de för- och nackdelar som ofta finns med dessa. Det finns (Bolman, Deal, 1997):

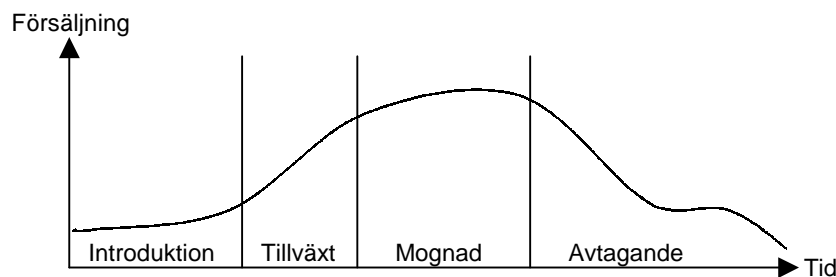
- **Enkelstruktur** som bara består av en operativ kärna och strategisk topp. Fördelen med den är att den är flexibelt och ha en hög anpassningsförmåga, då det är en strategisk topp som styr hela verksamheten. Nackdel är att personen på toppen snabbt kan skapa missnöje genom att inte vara rättvis.
- **Maskinbyråkrati** har till skillnad från enkelstruktur stödfunktion och teknostruktur. Stödfunktioner har som uppgift att underlätta för andra t.ex. sekreterare, medan teknostrukturen består av specialister som t.ex. mäter och standardiserar processer. I en maskinbyråkrati kan nackdelarna vara att människorna tröttnar på de ofta repetitiva uppgifterna, mellanchefer kan känna att deras beslut inte får gehör då de mesta direktiven kommer uppifrån.
- **Professionsbyråkrati** har en förhållandevis stor operativ kärna och detta ger en decentraliserad platt struktur. Nackdelen här är att förändringar ofta tar tid och detta beror på att de ser förändringar i omgivningen som ett hot.
- **Divisionaliserade former** här sker arbetet i skenbart självständiga enheter. Risken i sådana strukturer är att det blir en katt och råtta lek, mellancheferna vill ha mer inflytande medan toppen vill ha mer kontroll, och risken är att kontrollen förloras och den ena människan vet inte vad den andra gör.

- **Adhocrati** är en lös flexibel struktur som hålls samman av laterala medel. Problem här kan vara ledningen som inte vet hur den skall styra med fria tyglar, och anställda som inte vet hur de skall handskas med friheten.

Organisationsform handlar även om nätverk, hierarkier och livscykeln. **Nätverk** är ett sätt att kommunicera, i vilken kommunikationen bara bör ske så länge som det finns ett ömsesidigt utbyte. Svårigheten med denna form är dock att det kan vara svårt att bryta upp banden när det har gått en tid och människorna är vana vid varandra. **Hierarkier** är intressanta, då det berör Mintzbergsstruktur och maktförhållanden.

Livscykeln kommer nu att behandlas lite närmare, eftersom den behöver förklaras då den har fått ett antal betydelser:

1. Inom marknadsföring kallas den **produktlivscykeln**. (figur 3.12)



Figur 3.12: Produktlivscykeln (Kotler m.fl, 1999)

2. **Organismmetaforen** kallas en livscykel som talar om ett objekt/människas födelse, utveckling och död.
3. Inom informatik finns det två avdelningar, dels den **ingenjörsmässiga** och dels den **arkitekturella**. (Molina, Kusiaka, Sanchez, 1998). I den ingenjörsmässiga livscykeln är modellen beroende/styrande, medan den arkitekturella livscykeln är modellstödande. I den arkitekturella spelar sociala relationer, motivation och känsla en viktig roll, och den modell som passar bäst för båda parterna används. I den ingenjörsmässiga är det däremot modellen som styr.

Faran med arkitekturella livscykeln är att utvecklarna aldrig får chansen att lära sig någon modell riktigt bra, vilket i sin tur kan leda till misslyckande. En annan fara kan vara att relationerna blir så starka att det vid ett eventuellt uppbrytande blir mycket smärta. Risken med den ingenjörsmässiga är att modellen inte passar situationen och att köparen inte blir hörd.

### 3.3.4 Utvecklingsinfrastruktur

Utvecklingsinfrastrukturen har att göra med explicit kunskap och är sådan kunskap du kan ta del av genom att den finns dokumenterad på något sätt. Svårigheten när det gäller denna typ av kunskap är att det finns mängder och det är svårt att selektera ut den rätta kunskapen, och det kan också vara svårt att ta till sig kunskapen. Utvecklingsinfrastrukturen utgör stöd för aktiviteter i utvecklingsprocessen, och i den ingår därmed tekniska hjälpmedel, metoder och filosofin (Bennett, Culverhouse, Hughes, 1992).

Metoder kan ses som recept som beskriver i numerisk ordning hur kakan bakas, den beskriver vilka hjälpmedel (t.ex. visp) och vad för ingredienser som behövs för att lyckas. En metod bör vara så exakt beskrivet att två personer oberoende av varandra kan komma fram till samma resultat. (Andersen, 1994)

Teknik och verktyg är kunskap som kan hjälpa till att lösa problem t.ex. kan ett verktyg hjälpa oss att modellera en verksamhet och på så sätt nå en högre förståelse.

Filosofin kan vara en ledstjärna mot vårt mål, finns det finns t.ex. VBS som har en filosofi att information skall lagras lokalt och om det är någon som vill komma åt min information får han be mig om tillgång. En annan filosofi är när man förespråkar central lagring av all information.

### 3.4 Sammanfattning

Målet med en process är att utveckla en produkt:

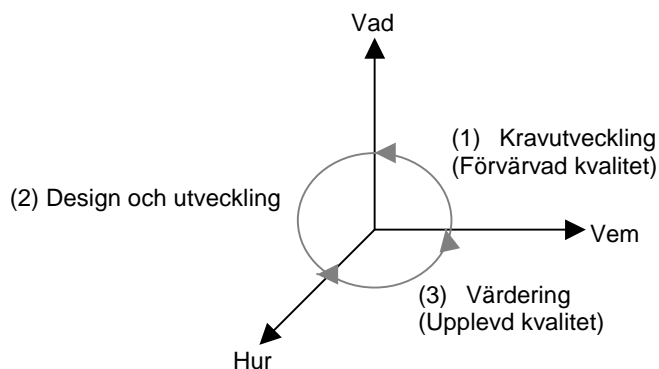
- **I rätt tid**
- **Till rätt kostnad**
- **Till rätt kvalitet**
- **Med attraktivitet**

**Tidskrav** är en fråga om parallell respektive överlappande systemutvecklingsaktiviteter dvs. tidsbesparingar blir en fråga om hur aktiviteter organiseras och samordnas. **Kostnadskrav** är en fråga om såväl processdesign som produktdesign. Produktdesign genom att modularisering skapar förutsättningarna för att komponentutveckling på ett relativt frikopplat sätt, utifrån detta nås:

$$\text{Attraktivitet} = \frac{\text{Upplevd kvalitet}}{\text{Förväntad kvalitet}}$$

**Kvalitetsfrågor** saknar någon bestämd kravbild varje utvecklingssituation är unik.

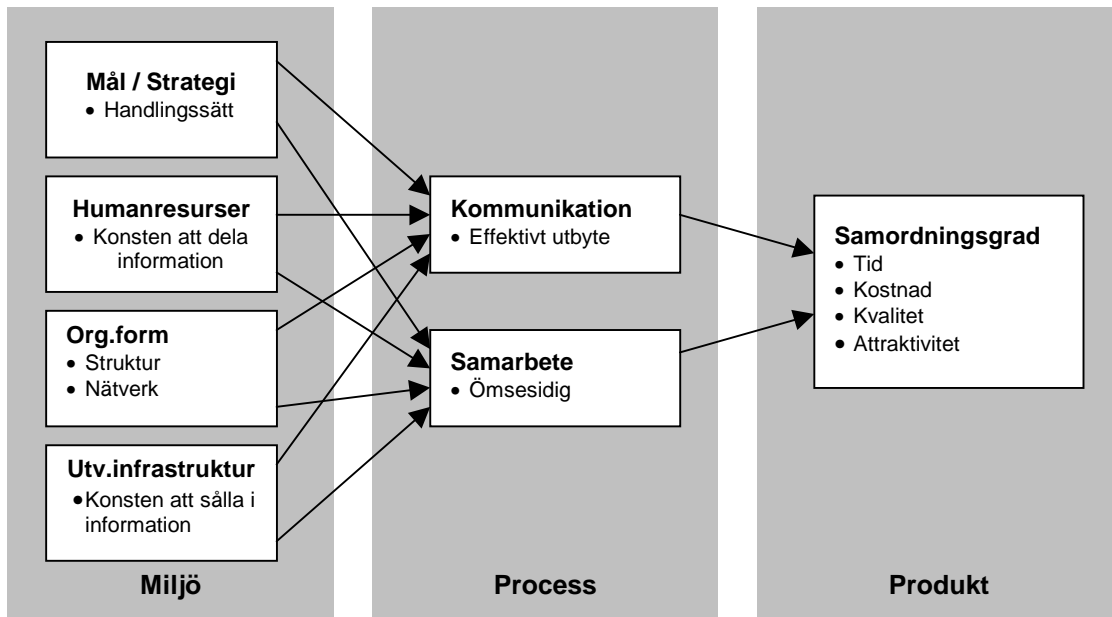
För att nå dessa mål krävs att figur 3.13 finns i tankarna, vem skall produkten vara till, vad skall systemet tillfredsställa för behov, och hur skall det utvecklas.



**Figur 3.13:** Process

Min studie har presenterat olika föreställningar om process, produkt och miljö, och det är viktigt att förstå att det bara är föreställningar. Det är kunden/användaren som avgör kvalitet utifrån sin unika situation, men någon bestämt avgörande blir det inte eftersom kundens kravbild är föränderlig. Denna föreställning ligger bakom processlivscykel perspektiv och flexibilitets krav som är de nödvändiga förutsättningarna för att tillgodose kundens föränderliga behovsbild. Slutligen såväl besparingar som kvalitet blir en faktor av kommunikation och samarbete mellan IT-leverantör och IT-användare.

Figur 3.14 sammanfattar detta kapitel genom den modell som är central för hela mitt arbete.



Figur 3.14: Modell

## 4. Empirisk undersökning

Modellen i figur 3.14 sammanfattar kvalitetsramverket, dock saknas komplettering av modellen utifrån kundernas synvinkel (verkligheten), och i detta kapitel sammanfattas en sådan empirisk undersökning.

När jag talar om kunden så menar jag två typer, den ena är kunden som kommer att använda ett stöd för utvecklingsprocessen och den andra är kunden som kommer att beröras av stödet. Frågorna vänder sig till den första sortens kunder, men har kompletterats med någon intervju med den andra sortens kund.

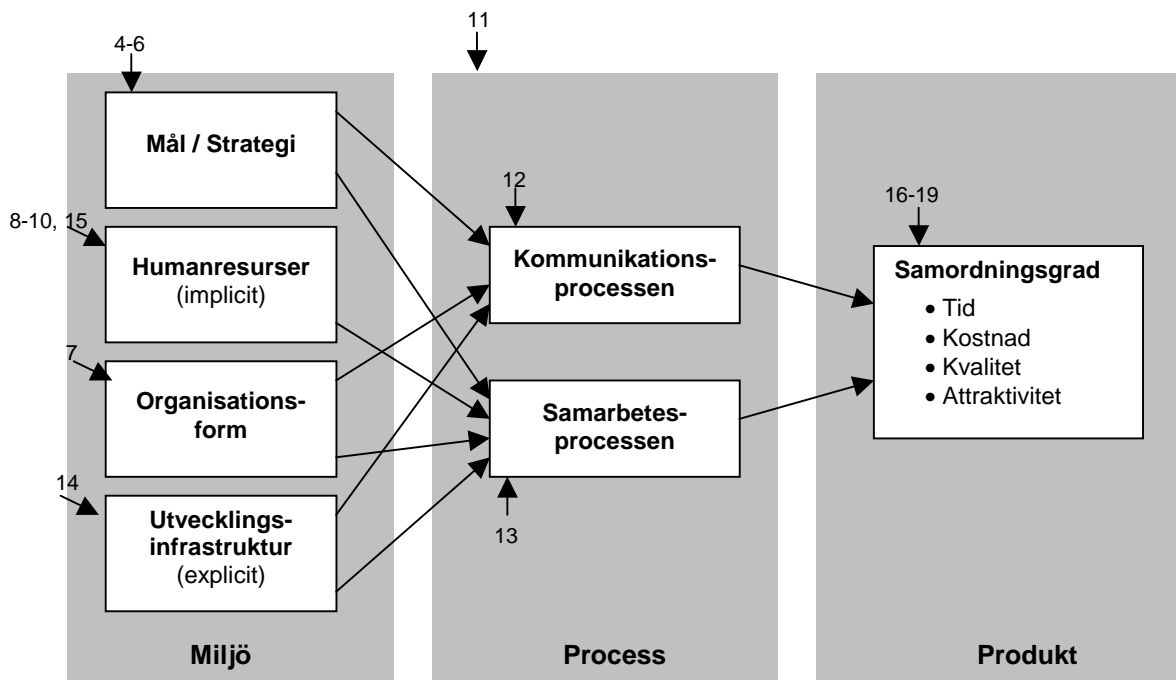
Syftet med intervjuerna är att få fram fler frågor, fler attribut och egenskaper till vägvalsmodellen.

Kapitlet redovisar **intervjufrågornas relevans**. För att underlätta detta försöker jag för varje fråga ställa frågan:

### Varför ställs denna fråga?

Efter frågornas relevans redovisa **resultatet av den empiriska undersökningen** och slutligen sker en **sammanfattning**.

### 4.1 Intervjufrågornas relevans



Figur 4.1: Frågeredovisning

Nedan kommer frågorna att gås igenom och i figur 4.1 visas var frågorna kan härledas ifrån.

#### 4.1.1 Bakgrund

Frågorna nedan är till för att se om de olika åldrarna, könen eller titel skiljer sig i sina åsikter om systemutvecklingsprocesser. Svaren på de 3 första frågorna är ganska självklara och behöver ingen närmare genomgång.

##### 1. Ålder:

Under 25

Under 50

50 +

**2. Kön:**

**3. Titel:**

**4.1.2 Miljö**

Miljö delas in i fyra områden vilka beskrivs mer i kapitel 3, och på dessa områden ställs frågor eftersom det påverkar hur bra kvaliteten blir i projektet. De fyra områdena är:

1. **Mål och strategi** (fråga 4-6)
2. **Organisationsform** (fråga 7)
3. **Humanresurser** (fråga 8-10 och 15)
4. **Utvecklingsinfrastruktur** (fråga 14)

**4. Vilka områden bör en IT-strategi beröra?**

- Kundrelationer
- Organisationsform
- Humanresurser
- Utvecklingsstruktur

**5. Vad har ert företag för IT-strategi?**

**6. På vilket sätt främjar/hindrar strategin kommunikationen och samarbete mellan:**

- Kunden och systemutvecklare?
- Mellan olika systemutvecklare?
- Mellan projektledare och systemutvecklare?

**7. Hur främjar/hämmer den formella strukturen (roller och hierarki kontra nätverk) kommunikation och samarbete?**

**8. Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system?**

- Vad är det som hindrar er?

**9. Vilken slags information kommuniceras vanligtvis mellan de aktörer som är involverade i systemutvecklingen?**

**10. Vilka slags kompetenser använder ni i systemutveckling?**

**15. Hur hanteras informationen från olika projekt?**

- Hur förs informationen vidare?

**14. Beskriv ett typiskt utvecklingsprojekt:**

- Vilka faser genomgår projektet?
- Vilka är de vanligaste misstagen som begås?
- Hur hanteras risker i projektet?
- Vad är projektets svagast punkt?
- Vad är det viktigaste resurserna i ett utvecklingsprojekt?

**4.1.3 Arkitektur**

Frågorna 11-13 försöker ge svar på frågor om kommunikationsprocessen och samarbetsprocessen som är det centrala i processen.

**11. Har organisationen en tydlig modell för att stödja systemutvecklingsprocessen?**



- Är du nöjd med modellen?
- Vad vill du ändra på?

## 12. Hur fungerar kommunikationen i systemutvecklingsprocessen?

- Vad anser ni som störande och som ni gärna vill förbättra?

## 13. Hur fungerar samarbete i systemutvecklingsprocessen?

- Vad anser ni som störande?
- Vad anser ni att det saknas?

### 4.1.4 Mål

Det kritiska i processen och produkten beskrivs i fråga 16-19.

Fråga 18-19 har relevans då den berör ett antal attribut som har att göra med en process/produkts chans att lyckas, vilket i sin tur har att göra med kvalitet.

**16. Vilka egenskaper/delar anser ni som kritiska i en systemutvecklingsprocess?** (t.ex. i rätt tid och rätt kostnad)

**17. Vilka egenskaper anser ni som kritiska i den produkt som systemutvecklingen skapar?** (t.ex. hög systemkvalitet (funktionalitet, korrekthet, konsistens etc.), hög system flexibilitet och bra service)

**18. Vilka attribut anser ni vara viktigast i en process**

- Välj ut de 5 viktigaste attributen
- Komplettera gärna med fler attribut

Kryss:	Attribut:	Förklaring:
	Förståelig (Understand- ability)	Hur lätt är det att förstå processens definition?
	Synlig (Visibility)	Kulminerar processens aktiviteter i klara resultat så att framstegen är externt synliga?
	Stödjande (Supportability)	Kan process aktiviteten stödjas av verktyg och tekniker och till vilken grad?
	Pålitlig (Acceptability)	Är processen så konstruerad att processfel undviks och tas om hand innan det blir produktfel?
	Robust (Robustness)	Kan processen fortgå fastän nya problem dyker upp?
	Varaktig (Maintainability)	Kan processen utvecklas för att reflektera förändringar i organisationskrav eller process förbättring?
	Snabbhet (Rapidity)	Hur snabbt från en specifikation kan processen utveckla ett system? Stödjer processen tidsplanering?
	Acceptabel	Är processen accepterad och användbar för ingenjörerna som ansvarar för framställning av slutresultatet?
	Verkningsgrad	Nås den förändring som önskas med medlen? Används metoder och tekniker på ett kompetent sätt? Fungerar processen?
	Effektiv	Optimeras användningen av resurser i processen?

Elegans	Är processen utformad på ett tilltalande sätt?
Tydlighet i användarroll	Finns det beskrivet hur arbetet skall fördelas mellan personer?
Anpassningsbarhet	Överensstämmer processen med situationerna (organisationen, personerna, mål) som kan uppstå? Kollar processen om nya krav uppstår?
Trivsel	Är processen utvecklad så att den även stöder det socioemotionella?

### 19. Vilka attribut anser ni vara viktigast i den produkt som processen skapar?

- Välj ut de 5 viktigaste attributen
- Komplettera gärna med fler attribut

Kryss:	Attribut:	Förklaring:
	Användbarhet	Anpassning av systemet till organisatoriska, arbetsmässiga och tekniska omgivningen?
	Säkerhet	Är systemet säkert mot oönskade tillgång till systemets data och faciliteter?
	Effektivitet	Använder systemet resurserna på bästa sätt?
	Korrekt	Uppfyllande av de uppställda kraven?
	Pålitlighet	Är felen minimerade, och outputen konsekvent?
	Lättunderhållet	Är kostnad för att hitta och rätta fel i systemet låg när det är igång?
	Lättestat	Är kostnad för att testa systemet gentemot de uppställda kraven låg?
	Begriplighet	Är det besvär att skaffa sig överblickbarhet över och förstå systemet?
	Återanvändbarhet	Hur är användbarheten vid flytt av systemet till andra tekniska plattformar eller andra sammansättningar av resultatet?
	Portabelt	Kan annan utrustning användas?
	Integrerbart	Är det lätt att koppla samman systemet med andra system?
	Accepterbarhet (Acceptability)	Är människorna som kommer i kontakt med systemet tillfredställda med det. Uppfyller det deras behov?
	Tillgänglighet (Availability)	Är tillgången tillräcklig?
	Kompabilitet (Compatibility)	Passar systemet organisationen?
	Dokumentation (Documentation)	Stödjer den kommunikationen?

Lätt att lära (Ease of learning)	Kort lärotid och intuitivt?
Snabb utvecklingsfart (Fast development rate)	Är utvecklingen snabb relativt till systemets storlek?
Funktionalitet (Functionality)	Ombesörjer systemet kraven?
Implementerbart (Implementability)	Hur lätt implementering skedde?
Ekonomisk (Economy)	Är systemet kostnadseffektivt och inom resursgränser och begränsningar?
Låg koppling (Low coupling)	Är interaktionen mellan subsystem sådana att de kan ändras utan att det påverkar resten av systemet?
Underhåll (Maintainability)	Är de mycket arbete med att hålla systemet igång, och möta det föränderliga kraven?
Robust (Robustness)	Felsäker, Feltolerant, driftsäkert?
Enkelhet (Simplicity)	Är komplexiteten minimerad?
Testbar (Testability)	Till vilken grad kan systemet testas med avseende av slutanvändning?
Tidlöshet (Timeliness)	Funkar den i alla situationer?
Synlighet (Visibility)	Är det tydligt hur systemet skall uppträda i vissa situationer?

## 4.2 Redovisning av den empiriska undersökningen

Här kommer resultatet av undersökningen beskrivas utifrån fyra områden:

1. **Bakgrund**
2. **Miljö**
3. **Process**
4. **Produkt**

### 4.2.1 Bakgrund

Mina handledare Maria Johnson-Kjällström och Annikas Nilsson gav mig ett antal namn att intervjua. Jag fick till slut tag i fyra Enatoranställda och fyra icke-Enatoranställda att intervjua. I figur 4.2 visas fördelningen av män och kvinnor, ålder och var de arbetar.

Titel	Kön	Ålder
Produktchef, Banteknik Spårvägen	Man	50+
Line Quality Coach Ericsson ETX	Kvinna	25+
Teknikområdesansvarig	Man	25+

Volvo VAK		
Universitetsadjunkt Högskolan i Skövde	Kvinna	25+
Konsult Enator Informationssystem Väst	Kvinna	25+
Konsult och teamleader för utvecklingsenheten Enator Informationssystem Väst	Man	25+
Verksamhetskonsult Enator Informationssystem Väst	Man	25+
VD Enator Informationssystem Väst	Man	25+

**Figur 4.2:** Bakgrund

Intervjuerna skedde på deras arbetsplatser med ett undantag, universitetsadjunkten, som intervjuades via telefon. Som jag nämnt tidigare har intervjufrågorna anpassats efter hand som intervjuerna ägt rum, allt för att få ut som mycket som möjligt av dem (se appendix D).

#### 4.2.2 Miljö

#### 4. Vilka områden bör en IT-strategi beröra?

Omkring hälften av de intervjuade ansåg att IT-strategins fokus ligger i teknik och hur IT kan användas i företaget för att underlätta för personal. Andra ansåg att IT måste beröra alla bitarna som beskrivs i parentesen ovan.

#### Åsikter:

- Ett bättre namn på strategin är IS/IT strategin, eftersom strategin även bör beröra den miljö som informationssystemen finns i. Det viktiga i en sådan strategi är bl.a.: filosofin om att sprida och dela information (informationsresursen), sambandet mellan verksamhet och IT, metodfrågor som t.ex. standardprodukter, egenutveckling och integration mellan system, tekniska aspekter och slutligen ansvar. (Universitetsadjunkt)
- En IT-strategi bör lägga fokus på frågorna: Hur bör IT användas för att underlätta för verksamheten (t.ex. vad för funktion och roll skall IT ha)? Vart vill man nå med verktyget IT? (Verksamhetskonsult)
- I IT-strategi är relationen med leverantören en viktig bit. Leverantörsantalet är viktigt, då det genom att få ner antalet kan skapas en bra kommunikation och samarbete. Antalet är även viktigt för konkurrens, om det bara finns en leverantör känner de sig säkra och då höjs priset. (Teknikområdesansvarig)

#### 5. Vad har ert företag för IT-strategi?

Tre av de personer som jag intervjuade kunde tala om att det fanns en strategi i deras företag, resten visste inte om det fanns någon.

#### Åsikter:

- ... dock finns det ett problem och det är att IT-strategin ofta snabbt blir omodern. På Spårvägen har de inte tid att ständigt uppdatera den. (Produktchef)
- Problemet när man är ett konsultföretag är att det är svårt att få strategin att genomsyras bland de anställda, eftersom man ofta sitter hos kund och då får rätta sig efter deras strategi. (Konsult och teamleader för utvecklingsenheten)

- Har vi någon? Koncernen kanske håller på att skapa en, men frågan är om den ger något för oss. För att en strategi skall vara lönt måste den vara förankrad. Den måste med andra ord vara nedbruten till en viss nivå inom koncernen.  
(Verksamhetskonsult)

#### **6. På vilket sätt främjar/hindrar strategin kommunikationen och samarbete mellan:**

- kunden och systemutvecklare?
- mellan olika systemutvecklare?
- mellan projektledare och systemutvecklare?

De intervjuade var överens om att om en strategi inte är förankrad så främjar den inte kommunikation.

Ett problem för stora företag är att en delverksamhet inte får så stort utrymme, då deras egen strategi hindras pga. den övergripande strategin. Man kan dock vända på det och säga att strategin hjälper till att få alla i verksamheten att kämpa åt samma håll, vilket i sin tur leder till att kommunikationen ökar.

#### **Åsikt:**

- En strategi får inte vara begränsad så att den hindrar relationer mellan kunder och systemutvecklare. IT-strategin kan specificera val av teknik mm. och detta kan främja kundrelationen, då kunskapen inom den valda tekniken blir stor.  
(Teknikområdesansvarig)

#### **7. Hur främjar/hämmar den formella strukturen (roller, hierarki vs nätverk) kommunikationen och samarbete?**

Det är viktigt att strukturen är tydlig, att ansvarsområdena är noggrant beskrivna, och att rollstrukturen är tydlig, annars kan bl.a. synkroniseringsproblem och barriärer mellan grupper lätt uppstå.

#### **Åsikt:**

- För att den formella strukturen skall kunna främja kommunikation och samarbete är det viktigt att ha en struktur som har klara ansvarsförhållanden, vilket innebär att det finns fasta roller, av rapporteringspunkter och att ansvar och befogenheter går hand i hand. (Universitetsadjunkt)
- Vad som hindrar samarbete och kommunikation är att mätning sker genom beläggning per konsult, alltså på individnivå. Detta leder till att konsulterna hellre lägger timmar på kunden, än på att utbilda sina arbetskamrater, då det senare ej syns i mätningen. Mätningen bör istället ske på teamnivå, eftersom det skulle uppmuntra till samarbete. (Verksamhetskonsult)

#### **8. Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system?**

I undersökningen visade det sig att det var 50 % som ansåg att det ibland går och 50 % ansåg att det går åstadkomma bra system. De som ansåg att det går, var överens om att det är utvecklarens ansvar att ställa rätt frågor, att veta om att kraven aldrig är tydliga från början utan mognar fram och att processen bör vara öppen för förändringar.

#### **Åsikt:**

- Ibland. Det finns en stor risk att fel sak görs och att väntat resultat inte uppnås. För att lyckas skapa ett bra system krävs att rätt frågor ställs och att kraven utreds. Det krävs också att systemutvecklaren har kunskapen att se effekterna av de krav som kunderna ställer. (Konsult och teamleader för utvecklingsenheten)

- Ja, inom Enator Informationssystem Väst AB så finns det goda chanser för detta, då de har både verksamhetsutveckling och systemutveckling. Det är viktigt att verksamhet och systemutveckling arbetar ihop för att nå kundens mål och gemensam förståelse. (VD)

### **9. Vilken slags information kommuniceras vanligtvis mellan de aktörer som är involverade i systemutvecklingen?**

Den information som kommuniceras är:

- Interna strukturen av systemet (teknisk information, om databaser och modeller mm).
- Systemmiljö.
- Identifiera problemen som systemet skall lösa och hur det skall utföras (användarfall).
- Att ha avstämningar som berättar var man ligger tidsmässigt och vad man skall göra.

Kommunikationen sker vanligtvis via:

- Möten
- E-post
- Fax
- Protokoll
- PPS (håller reda på den formella informationen)
- Genom att möblera så att de som arbetar i samma projekt sitter i samma rum
- Filer
- Dokument

### **10. Vilka slags kompetenser använder ni i systemutveckling?**

Nedan följer en sammanfattning av åsikter:

- Bred och allsidig kompetens
- Verksamhetskompetens (de olika miljöerna)
- Teknisk kompetens (verktygen i verksamheten)
- Generalister
- Social kompetens
- Kunna argumentera och uttrycka sig i ord och skrift
- Systemutvecklare bör inte vara krokodiler, stora i käften och med små öron.
- Färdighet
- Erfarenhet
- Värderingar
- Teknisk projektledning (viktigt i vissa fall)
- Diskutera användarmiljön (ha en gemensam bas)
- Bred analytisk kompetens
- Att vara strukturerad

### **15. Hur hanteras informationen från olika projekt?**

### Hur förs informationen vidare?

De flesta ansåg att hanteringen av kunskap inte sköttes på ett bra sätt, ofta samlades informationen in men blev liggande. Detta berodde på att den inte var tillräcklig åtkomlig, vilket fick till följd att misstag ofta begicks flera gånger om.

Informationen förs därför vidare genom människors erfarenheter, med risken att när en person slutar så förloras kunskapen.

#### Åsikt:

- Det skall finnas en kortfattad slutrapport där det framgår klart och tydligt om projektet blev lyckat eller mindre lyckat, en kortfattad beskrivning av hela projektet och där skall även kostnad och tidsramar, som är viktiga och svåra faktorer, framgå. Varje slutrapport skall ha samma upplägg och vara åtkomlig på ett bekvämt sätt. (Konsult)
- Erfarenheterna samlas i en slutrapport som projektledaren gör och den skall sedan alla i verksamheten läsa och lära sig av. I SQA (Software Quality Assurance) ingår det att se till att alla roller tar del av slutrapporten i det efterföljande projektet. (Line Quality Coach)

### 14. Beskriv ett typiskt utvecklingsprojekt:

I figur 4.3 syns exempel på ett antal faser som projektet kan gå igenom. Det bör dock sägas att detta är idealfall då projektet fungerar och alla krav är klara.

VD	Konsult och teamleader för utvecklingsenheten	LQC (Line Quality Coach)	Teknikområdesansvarig	Verksamhetskonsult
1. Processkartläggning 2. Målbildsdefinition 3. Kravdefinition 4. Analys 5. Design 6. Applikationsutveckling 7. Test 8. Implementering 9. Underhåll	1. Förstudie 2. Genomförande 3. Leverans 4. Godkännande	1. Kravhantering 2. Analys och design 3. Implementering 4. Test 5. Leverans	1. Förberedande utveckling 2. Systemutveckling 3. Produktutveckling	1. Idé 2. Beställning 3. Insamling av information 4. Grov inkrementell indelning 5. Kravbild tydlig 6. Utveckling i inkrement

Figur 4.3: Faser i ett utvecklingsprojekt

#### - Vilka är de vanligaste misstagen som begås?

De flesta ansåg att kravhanteringen är en svag punkt som ofta underskattades. Ett annat misstag som ofta begås är att gå för fort fram.

#### - Hur hanteras risker i projektet?

Riskerna hanterades ganska likvärdigt genom riskanalys och genom uppföljning under projektets gång, dock ansåg en del att de pga. tidsbrist inte hade tid med en ordentlig uppföljning.

Exempel på riskanalys:

- När målet för projektet är spikat brainstorma för att få fram alla risker
- Rangordning
- Ta hand om de riskerna som är för farliga

- Uppföljning hur det går med elimineringen

**- Vad är projektets svaga punkt?**

- Okända händelser
- Tidsplaneringen
- Projektledarkompetens
- Beställarkompetensen
- Att inte veta vad kunden har för krav

**- Vad är det viktigaste resurserna i ett utvecklingsprojekt?**

Åsikterna skildes åt då vissa ansåg att alla resurser är lika viktiga, medan andra gav förslag på det viktigaste resurserna.

**Åsikt:**

- Människor
- Projektledningen
- Utvecklingsmiljön
- Nyckelkompetenser
- Pengar

**4.2.3 Process**

**11. Har organisationen en tydlig modell för att stödja systemutvecklingsprocessen?**

Fem personer hade inte någon modell, två personer hade en modell och en person var kund och svarade ej på den frågan. De som hade modeller ansåg att deras modeller inte var bra och behövde förbättras.

**Åsikt:**

- Modellen bör vara tydligt förankrad och känd. Den skall vara beskriven på ett sådant sätt att alla vet vad som skall göras. Det är viktigt att man i modellen vet vad den börjar och när den slutar (in kommer detta och ut kommer detta), och att det på ett tydligt sätt finns beskrivit att det är det här målet som gäller och hur målen skall nås. (konsult)
- Har en modell som heter SDPM, men den skall bytas ut till RUP. Detta för att den förra modellen inte ger stöd för iterativutveckling. (Line Quality Coach)

**12. Hur bör kommunikationen fungera i systemutvecklingsprocessen?**

**Vad anser ni som störande och som ni gärna vill förbättra?**

De flesta ansåg att det bör finnas någon modell/metod som stödjer och säkrar förståelse.

Modellen/metoden måste ha tydlighet i sina steg, så att personer vet vad för uppgifter de skall utföra. Det är viktigt att det lätt går att ge besked om var vi är och vart vi ska.

**Åsikt:**

- Ett sätt att komma tillrätta med problemet med barriärer är att involvera aktörer i varandras faser. (Teknikområdesansvarig)
- Enator Informationssystem Väst AB försöker genom att ha en plattorganisation ha högt i tak. Det leder till en öppen diskussion och att fler säger vad de tycker. (Konsult)

**13. Hur bör samarbete fungera i systemutvecklingsprocessen?**



### Vad anser ni som störande? Vad anser ni att det saknas?

Det bör finnas ett stöd för den kompetens som behövs, vad som krävs av kompetensen och stöd för en gemensam begreppsbas och en tydlig målbild.

#### 4.2.4 Produkt

Upptäckt under intervjuerna att fråga 16 och 17 var svåra att svara på, och att jag bara fick svar från vissa vilket gjorde denna fråga kan bli missvisande.

**16. Vilka egenskaper/delar anser ni som kritiska i en systemutvecklingsprocess?** (t.ex. i rätt tid och rätt kostnad)

- Designfasen (när funktioner och teknik skall fungera ihop)
- Tidsplanen
- Kraven
- Kompetens
- Pengar
- Leverantörerna
- Förmåga att förstå kunden

**17. Vilka egenskaper anser ni som kritiska i den produkt som systemutvecklingen skapar?** (t.ex. hög systemkvalitet (funktionalitet, korrekthet, konsistens etc.), hög system flexibilitet och bra service)

De få som svarade på denna fråga ansåg att det var kundtillfredsställelse som var mest kritiskt.

#### Åsikt:

- Teknik, tid och pengar (Teknikområdesansvarig)

**18. Vilka attribut anser ni vara viktigast i en process?**

Egenskaper	Antal
Effektiv	5
Synlig	4
Pålitlig	4
Förstårlig	4
Snabbhet	4
Varaktig	3
Robust	3
Tydlighet i användarroll	3
Acceptabel	2
Anpassningsbarhet	1
Stödjande	1
Verkningsgrad	1

**Figur 4.4:** Viktigaste attributen vid systemutveckling

I figuren ovan visas det viktigaste attributen, dock finns det en person som inte har svarat och alltså är det sju personer som har gett sina åsikter.

## 19. Vilka attribut anser ni vara viktigast i den produkt som processen skapar

Egenskaper	Antal
Användbarhet	7
Accepterbarhet	4
Begriplighet	4
Funktionalitet	4
Integrerbarhet	3
Ekonomisk	3
Korrekt	2
Effektivitet	2
Pålitlighet	2
Robust	2
Synlighet	1
Låg koppling	1
Enkelhet	1
Tidlöshet	1
Lätt underhållet	1
Tillgänglighet	1
Återanvändbarhet	1

**Figur 4.5:** Viktigaste egenskaperna i en produkt

Figur 4.5 visar de viktigaste attributen i den produkt som processen skapar.

### 4.3 Sammanfattning

Resultatet som redovisats i avsnitt 4.2 och kapitel 3 har nu givit en grund för vägvalsmodellen. Intervjuerna kanske kunde ha blivit fler till antalet men resultatet är tillräckligt för den fortsatta studien.

## 5. Vägvalsmodellen

Idén att skapa en vägvalsmodell kommer från boken *Architectures for Enterprise Integration* (Bernus, Nemes, Williams, 1996) där just en modell för att välja en metod skapas genom kryssfrågor. Kryssfrågor väljs eftersom det är ett enkelt sätt att jämföra olika modeller, det är dock viktigt att göra en heltäckande jämförelse så att bilden som målas upp blir så tydlig som möjlig.

Grundstommen i vägvalsmodellen är modellen som beskrivs i kapitel 3 och empiriska studien i kapitel 4 och de viktigaste bitarna är kvalitet på miljö, process och produkt.

I detta kapitel kommer vägvalsmodellen att gås igenom och varje frågeområde kommer att utredas utifrån frågan:

### Varför ställs denna fråga?

#### 5.1 Bakgrund

Systemutvecklingsprocessens primära verksamhet är att omvandla kundens oklara och föränderliga behovs- och kvalitetsbild till ett mjukvarusystem som uppfyller båda kraven. Såväl indata, som utdata till processen är information i olika former.

- Rätt produkt och rätt process förutsätter rätt systemutvecklingsprocess, dvs. rätt **kommunikationsform** och rätt **samverkansform**.
- Rätt systemutvecklingsprocess förutsätter rätt **systemutvecklingsmiljö**.
- Rätt systemutvecklingsmiljö förutsätter rätt **samarbetsform med kunden**, rätt **kunskap och kompetens**, rätt **utvecklingsinfrastruktur** och rätt **organisationsform**.
- Därmed bör utvecklingsstrategi referera till **kundrelationer**, **humanresurser**, **utvecklingsinfrastruktur** och **organisationsform**.

Denna vägvalsmodell kan delas in i fyra delområden:

1. **Miljö**
2. **Process**
3. **Produkt**
4. **Övergripande generaliseringsgrad**

Det är genom dessa ett kvalitetstänkande skapas.

#### 5.2 Genomgång av vägvalsmodellen

Nedan kommer vägvalsmodellen att beskrivas.

##### 5.2.1 Miljö

Det första området som berörs är strategin och vilka områden som modellen stödjer. Berörs i avsnitt 3.3.1 och fråga 4-6 i kapitel 4.

#### **A0. Hur relateras modellen till IT-organisationens strategiska områden?**

- Vilka strategiska område refererar modellen till?

- Relationer (kundrelationer och leverantörsrelationer)
- Organisationsform
- Humanresurser (dvs. systemutvecklarens kunskaper, kompetens, lojalitet och handlingsfrihet)

- Utvecklingsinfrastruktur (dvs. metoder, tekniker och Case-verktyg)

**A1-A4** går igenom strategin vilket i sin tur innebär att det går igenom miljön.

### **A1. Relationer**

Relationer betyder här den formella relationsformen som ofta finns beskriven i kontraktet.

Vilka samarbetsformer mellan kundorganisation och IT-organisation omfattas av modellen?

- Avtalad samverkan, (dvs. juridiskt kontrakt)
- Informell samverkan (dvs. socialt kontrakt)
- Kooperation, (dvs. man deltar och påverkar varandras beslut)
- Koalition, (dvs. man etablerar en gemensam men annars autonom enhet)

### **A2. Organisationsform**

I avsnitt 3.3.3 och fråga 7 (kapitel 4) så beskrivs det hur organisationen ser ut och nedan görs ett försök att se närmare på det.

#### **A20. IT-organisationens verksamhetsområde**

Hur relateras modellen till IT-organisationens verksamhet? Vilka av de nedanstående kompetensområdena omfattas av modellen?

- Marknadsföring
- Produkt/Systemdesign
- Produkt/Systemutveckling
- Kundservice och systemunderhåll
- Utbildning och träning
- Personalservice
- Ekonomi
- Strategisk planering
- Projekt management
- Resurs management
- Infrastruktur management

#### **A21. IT-organisationens beslutsstruktur**

Vilken organisations struktur främjas av modellen?

- Hierarkisk
- Nätverk
- Lärande
- Virtuellt
- Teamorienterat
- Platt

#### **A22. IT-organisationens sätt att organisera ett utvecklingsprojekt**

Vilken arbetsform främjas av modellen?

- Vattenfall (dvs. sekventiell beroende mellan olika aktiviteter)
- Spiral (dvs. inkrementell lärande process)
- Överlappande (dvs. en aktivitet kan starta innan en annan har avslutat)
- Parallell (dvs. många aktiviteter pågår parallellt)

#### **A23 Organisationens sätt att se på livscykel frågan**

Vilken form av livscykel stödjer modellen?

- Produktlivscykel (marknadsföring och produkt efterfrågan som avgör livscykel)
- Organismmetaforen (födelse, utveckling och död och händelsebaserad anpassning)

- Ingenjörsmässiga (komponenttänkande och snabbanpassning)
- Arkitekturella (lärande)

### **A3. Utvecklingsinfrastruktur**

Utvecklingsinfrastruktur har att göra med explicit kunskap och är t.ex. metoder, tekniker och Case-verktyg. Detta redogörs i avsnitt 3.3.4 och fråga 14.

### **A30. Modeller**

#### **A301. Modelltyper**

Vilka slags verksamhetsmodeller förespråkas av systemutvecklingsmodellen?

- Funktionella verksamhetsmodeller
- Aktivitetsmodeller
- Systemmodeller
- Beslutsmodeller
- Objektmodeller
- Informella modeller

#### **A302. Nödvändiga modelltyper**

Vilka slags verksamhetsmodeller anses som nödvändiga av systemutvecklingsmodellen?

- Funktionella verksamhetsmodeller
- Aktivitetsmodeller
- Systemmodeller
- Beslutsmodeller
- Objektmodeller
- Informella modeller

#### **A303. Modellkategori**

Vilken slags modellkategori förespråkas av systemutvecklingsmodellen?

- Grafiska modeller
- Matematiska modeller
- Tabeller
- Rika bilder

### **A31. Datorstöd**

#### **A310. Verktystyp**

Vilka slags verktyg förespråkas av systemutvecklingsmodellen?

- Case (computer aided software engineering)
- Dokumentationsverktyg
- Rittekniska verktyg

#### **A311. Datorstödda aktiviteter**

Vilka slags aktiviteter bör ha datorstöd enligt systemutvecklingsmodellen?

- Verksamhetskartläggning
- Kravdokumentation
- Funktionell design
- Arkitekturell design
- Detaljerad design
- Systemdokumentation
- Systemverifikation
- "Reverse engineering" (t.ex. man har ett program och vill få fram källkoden)

- Processdokumentation
- Utvärdering

### **A32. Metoder**

Vilka slags metoder förespråkas av modellen?

- Verksamhetsanalytiska
- Informationsanalytiska
- Systemanalytiska
- Prototyping
- Simuleringsmetoder

Vilka "produkter" förespråkar modellen att det framställas med stöd av en metod?

- Riskanalys
- Kravspecifikation
- Analys
- Designspecifikation
- Detaljerad design specifikation
- Körbart system
- Systemens operationella och förvaltnings förutsättningar

Vilken slags metoddokumentation använder modellen?

- Handbok
- Utbildningsmaterial
- Organisatoriska procedurer
- Case-baserat stöd för systemutvecklare
- Case-baserat stöd för teamarbete
- Interaktivt hjälpmedel

### **A33. Information**

#### **A330. Information management**

Vilken slags information management filosofi förespråkar modellen?

- Global (tillgänglig för alla)
- Sluten
- Ingen
  
- Differentierad (delat i ansvarsområden)
- Odifferentierad (odelad)
- Ingen
  
- Central (ett enda ställe)
- Lokal (flera ställen)
- Ingen

#### **A331. Informationshantering i organisationen**

Hur hanteras kunskap från olika projekt i modellen?

- Informellt via människor
- Formellt via dokument eller e-post mm
- Inte alls

#### **A4. Human resurser**

Human resurser har att göra med implicit kunskap och beskrivs i avsnitt 3.3.2. Det berörs även i fråga 8-10, och 15 i kapitel 4.

#### **A40. Utbildning och träning**

##### **A400. Former av lärande**

Vilka former av lärande förespråkas av modellen?

- Learning by teaching
- Learning by doing (learning from experience)
- Learning by design support
- Learning by "guru" support
- Learning by example

#### **A41 Kunskaper och Kompetens**

##### **A410. Teknisk kompetens**

Vilken slags design kompetenser bör systemutvecklare besitta enligt modellen?

- Förmåga att skapa lösningar
- Förmåga att upptäcka felaktigheter innan de blir allvarliga problem
- Förmåga att modularisera / systematisera
- Förmåga att modellera
- Förmåga att skapa arkitekturer (dvs. förmåga att skapa överblickbarhet, säkra flexibilitet och skydda kvalitet etc.)
- Förmåga att integrera och samtidigt hålla komplexiteten under kontroll
- Förmåga att förstå kundens verksamhet
- Förmåga att granska systemens tekniska kvalitet
- Förmåga att förstå de verktyg som är aktuella

##### **A411. Social kompetens**

Vilken slags social kompetens bör systemutvecklare besitta enligt modellen?

- Förmåga att lyssna
- Förmåga att samarbeta
- Förmåga att kommunicera med kunden på ett naturligt sätt
- Förmåga att kommunicera med andra systemutvecklare
- Förmåga att vägleda projektdeltagarna
- Förmåga att kunna argumentera i skrift och tal

##### **A412. Management kompetens**

Vilken slags management kompetens bör systemutvecklare besitta enligt modellen?

- Förmåga att klargöra och förhandla (omförhandla) olika förutsättningar
- Förmåga att organisera och samordna insatserna
- Förmåga att värdera insatsernas resultat
- Förmåga att värdera alternativ
- Förmåga att balansera resp. prioritera
- Förmåga att skapa kognitiv samordning (dvs. förståelse och medvetenhet)
- Förmåga att skapa en social miljö
- Förmåga att vara en "agent of change"
- Förmåga att se förstå vilken kompetens som behövs till utvecklingen

## 5.2.2 Systemutvecklingsprocess

Systemutvecklingsprocess berörs i avsnitt 3.1 och i frågorna 11-13.

### **B0. Processens natur**

#### **B00. Systemutvecklingsomfång**

Vilka systemutvecklingsaktiviteter omfattas av modellen?

- Verksamhetsanalys
- Kravspecifikation
- Arkitekturell design
- Detaljerad design
- Implementering (kodning och test)
- Utbildning
- Drift och underhåll
- Avveckling
- Utvärdering

#### **B01. Val av källor för att fylla kunskapsinnehåll**

Hur hanterar modellen kunskapsbristen / kompetensbristen?

- kortsiktigt samarbete med tredje part
- Långsiktig samarbete med tredje part
- Ny rekrytering
- Vidarutveckling av den kompetens som finns tillgänglig i företaget

### **B02. Kundmedverkan**

#### **B020. Samarbete och samverkan mellan kund and systemutvecklare.**

I vilka av nedanstående systemutvecklingsaktiviteter är - enligt modellen - kundens medverkan nödvändig?

- Verksamhetsanalys
- Kravspecifikation
- Arkitekturell design
- Detaljerad design
- Implementering (kodning och test)
- Utbildning
- Drift och underhåll
- Avveckling
- Utvärdering

### **B1. Kommunikationen**

Kommunikation beskrivs i avsnitt 3.1.3.1 och i fråga 12.

#### **B10: Kommunikationen mellan systemutvecklare och projektledare**

Vilken kommunikationsform mellan systemutvecklare och projektledare främjas av modellen?

*Direkt (face-to-face)*

- Möten
- Workshop
- Prototyping

*Indirekt*

- Brev



- E-post
- Telefon
- Testprototyper
- Fax

### **B12. Kommunikationen mellan kunden och systemutvecklare**

Vilken kommunikationsform mellan kund och systemutvecklare främjas av modellen?

*Direkt face-to-face*

- Möten
- Workshop
- Prototyping

*Indirekt*

- Brev
- E-post
- Telefon
- Testprototyper
- Fax

### **B2. Samarbete**

Samarbete beskrivs i avsnitt 3.1.3.1 och i fråga 12.

### **B21. Beroende förhållande mellan teamgrupper**

Hur hanterar modellen beroende mellan olika teams?

- Standard metod/teknik
- Tidsplaner
- Planerade möte
- Workshop
- Central informationsbasen
- Informella sociala möte
- Genom att låta den sitta i samma rum

### **B22. Management av oförutsatta händelser**

Hur hanterar modellen oförutsatta händelser?

- Hierarkisk ledning
- Projektledning
- Öppen debatt and medbestämmande

### **5.2.3 Samordningseffekter (Målbilder)**

I avsnitt 3.2 och i frågorna 16-19 beskrivs samordningseffekter.

### **C01. Processrelaterade mål**

Vilka processrelaterade mål främjas av modellen?

- Minskning av utvecklingskostnader
- Lämplig samordningsgrad
- Minskning av leveranstid
- Ökad garanti för att hålla och synliggöra tidsplanen
- Ökad förmåga att förstå kundens behov
- Ökad möjlighet till att använda optimala resurser
- Förmåga att ta omhand förändringar i krav ökas

## **C02. Systemrelaterade mål**

Vilka system/produkt relaterade mål främjas av modellen?

- Hög systemkvalitet (funktionalitet, korrekthet och konsistens etc.)
- Hög system flexibilitet
- Lämplig grad av systemintegration
- Hög systemöverblickbarhet
- Snabb systemanpassning till kundverksamhetens föränderliga natur utan att orsaka komplexitet
- Hög kundtillfredsställelse
- Lättbegriplighet (snabb systemöversikt)
- Kostnadseffektiv inom resursernas gränser

## **C03. Service relaterade mål**

Hanterar modellen service relaterade mål såsom:

- Väl informerad kund om eventuella förseningar etc.
- Minimal kund klagomål
- Hög tillgänglighet av systemutvecklare
- Kontinuitet (kunden träffar alltid samma systemutvecklare)
- Snabb respons till kundens klagomål
- Hög vilja att bemöta/uppfylla kundens behov

### **5.2.4. Modellens egenskaper**

Slutligen beskrivs nedan modellens generaliseringsgrad.

## **D0. Modellens generaliseringsgrad**

**Verksamhetssektor.** Vilken slags verksamhetssektor refererar modellen till?

- Privatsektor
- Offentligsektor

**Industri.** Vilken slags industri refererar modellen till?

- Handels
- Transport
- Tillverkning
- Tjänster
- Utbildning

**Systemtyp.** Vilka slags systemtyper refererar modellen till?

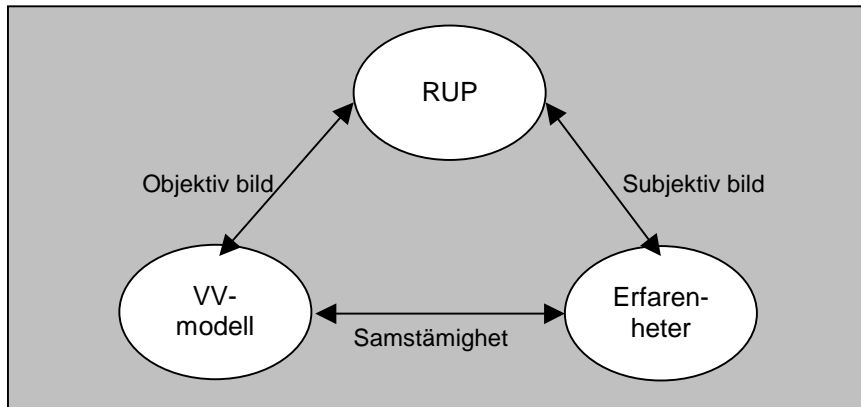
- Transaktionssystem (TPS)
- Beslutsstödsystem (BSS)
- Strategiska informationssystem (SIS)
- Rapporteringssystem (MSS)
- Ledningssystem (ESS)
- Ekonomisystem (MIS)
- Interorganisatoriska informationssystem (IOS)
- Office informationssystem (OIS)
- Distribuerade informationssystem (DIS)
- Federativa informationssystem (FIS)

### **5.3 Sammanfattning**

Denna modell har alltså skapats för att vara ett bra hjälpmedel vid val av en systemutvecklingsmodell. I kapitel 6 kommer den att testas på RUP för att se vägvalsmodellens kvalitet.

## 6. Granskning och resultat av RUP

I detta kapitel kommer en kortfattad introduktion av RUP att göras utifrån miljö, process och produkt (se även appendix E), efter det granskas RUP utifrån ett empiriskt perspektiv, och sist sammanfattas kapitlet med en normativ och en deskriptiv bedömning.

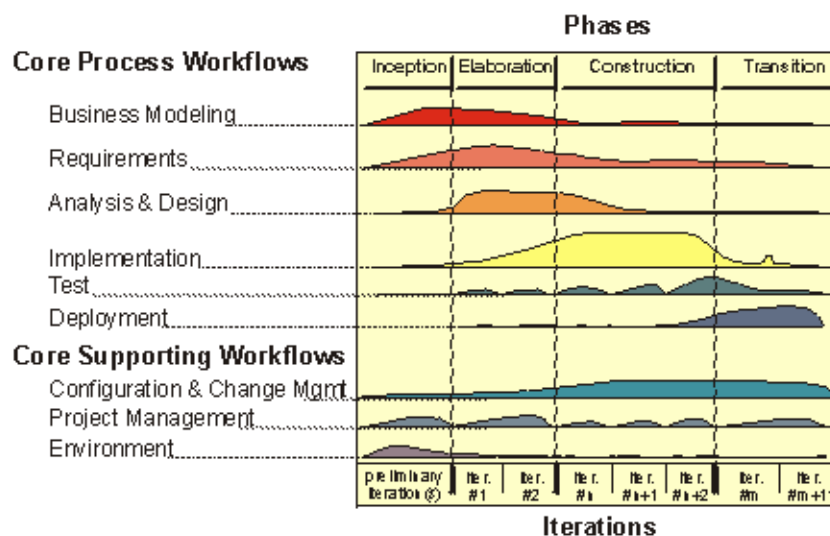


**Figur 6.1:** Tillvägagångssätt

Min empiriska undersökning har fokuserat på att ge en konsistent och fruktbar bild av RUP utifrån de som redan använder sig av RUP och därmed har erfarenheter. Vidare har vägvalsmodellen istället fokuserat på fullständighetskraven och med andra ord vad som förväntas att en utvecklingsmodell som RUP bör innehålla.

### 6.1 Sammanfattning av RUP

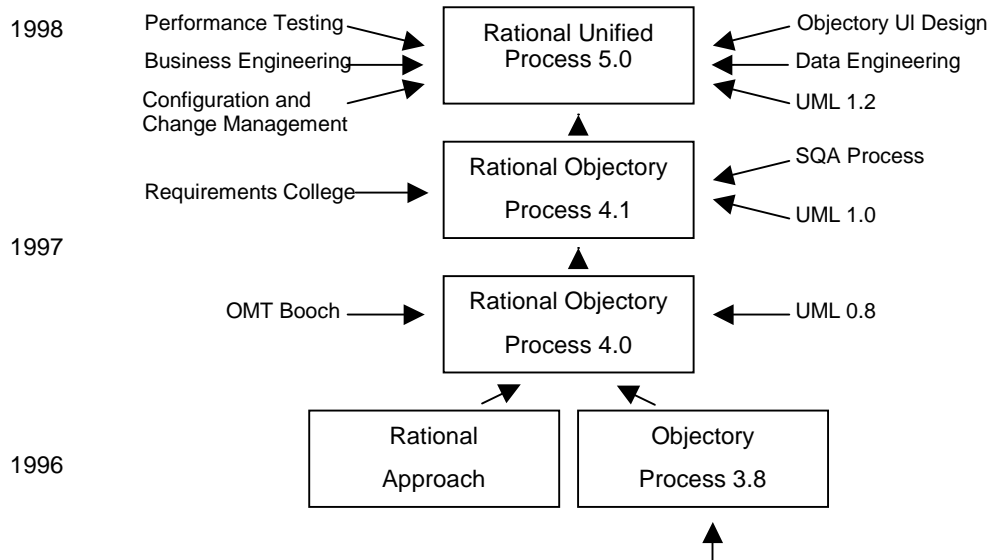
RUP är en av Rationals produkter som levereras på CD eller via Internet, och det är en utvecklingsmodell för att stödja utvecklingen av mjukvaruprodukter (figur 6.2). Till den finns ett stort ramverk som används i olika stor utsträckning beroende av projekt storlek och typ mm. Vid ett beslut om att använda RUP finns valmöjligheten att bara använda sig av vissa bitar, och införandet i verksamheten rekommenderar ett tillvägagångssätt som är iterativt och som sker genom att först implementera de mest kritiska delarna och sedan ta ett nytt varv o.s.v. Det är alltså meningen att RUP skall anpassas till den verksamheten som den skall införas i.



**Figur 6.2:** Utseende av RUP (RUP CD-ROM)

## 6.1.1 Process

Figur 6.3 beskriver utveckling mot RUP.



Figur 6.3: Utveckling mot RUP (Kruchten, 1998)

**ROP** eller Rational Objectory Process är ett resultat av en integration mellan Objectory och Rational Approach.

**Objectory** skapades av Ivar Jacobson och den tar upp principer för processer. Den ger stöd för en iterativ utvecklingsprocess igenom hela livscykeln, består av verktyg, en objektorienterad designmetod som hjälper arbetet och dessutom använder den sig av användarfall (use-case). Objectory består av fem delar: kravanalys, robustanalys, design, implementering och testning.

**Rational approach** innehåller kunskap om iterativutveckling och arkitektur.

Vid skapandet av ROP togs även **UML** (Unified Modelling Language) in. UML är ett standardspråk som tillåter användare att kommunicera krav, arkitektur och design. Målet med UML är att (Lerman, Persson, 1999, Kruchten, 1998):

- modellera system genom objektorienterade koncept.
- skapa en koppling mellan olika artefakter.
- hitta problem i komplexa system.
- skapa ett språk som är användbart för både människa och maskin.

**RUP** har tagit till vara på mycket av det som fanns i ROP, men utökat processen ytterligare inom områden som datautveckling, företagsmodellering, projekthantering, konfigurationshantering och slutligen ökat integrationen med Rational Software Suite (verktyg). Verktygen har gjort att mycket av utvecklingen har kunnat automatiseras.

För processen är det viktigt att veta **när** och **hur**:

- **När** kallas i RUP för arbetsflöden (sex stycken för utveckling och tre för support) och faser.(Se figur 6.2) Varje fas kan ha en eller flera iterationer med specifika mål och varje iteration går igenom alla arbetsflödena. När en iteration är genomkörd frågar man sig om iterationsmålet är uppnått, om så är fallet fortsätter man, annars måste problemen lösas innan man går vidare.
- **Hur** kallas för ”activities” och har klara mål. Varje aktivitet tillhör en specifik ”worker” (se miljö), och storleken på den är mellan några timmar upp till ett antal

dagar. Aktiviteter kanske upprepas flera gånger under utvecklingens gång och måste då ha samma worker men ej samma individ. Ett exempel på en aktivitet är planera en iteration.

### 6.1.2 Produkt

RUP har som mål att utveckla **produkter** som är av rätt kvalitet, rätt kostnad och som utvecklas i rätt tid. Metoden utvecklar en mängd delprodukter längs vägen mot slutmålet och det kallas i RUP för **vad** (artefakt). Detta inkluderar information som produceras, modifieras eller används av processen t.ex. objektmodeller, aktivitetsmodeller och systemmodeller.

### 6.1.3 Miljö

Miljön kan delas in i kompetens, utvecklingsinfrastruktur, organisationsform och mål:

- I RUP beskrivs kompetensen i **vem** vilket innebär ”worker” eller roller. Detta kan ses som en hatt en individ har under projektet. En individ kan ha flera hattar. Ett exempel på en arbetare är designern som definierar ansvar, händelser, attribut, relationer av en eller flera klasser och bestämmer hur dessa bör anpassas till implementeringsmiljön.
- Mål som RUP försöker uppnå är en produkt som möter kundens krav, inom budget och i tid. I strategin beskrivs att den bör variera beroende på antal intressenter, komplexitet, samarbetsform m.m.
- Organisationsformen som finns i företaget försöker RUP anpassa sig efter i den mån det är möjligt. Livscyklifaktorn beskriver hur långt mjukvarans liv bör vara.
- Utvecklingsinfrastrukturen stöds genom diverse förslag på verktyg, metoder och tekniker.

### 6.1.4 Intervjusammanställning om RUP

För att min studie av RUP skulle bli så heltäckande som möjligt gjorde jag en intervju med Pär Jansson som är Produkt Manager på Rational (appendix B). Jag nöjde mig dock inte med det utan kompletterade intervjun genom att låta användare av RUP titta på svaren på frågorna och komplettera dem ytterligare (appendix E). De frågor jag har valt har tagits med för att täcka upp hela (figur 3.14) i kapitel 3 och nedan följer resultatet av den undersökningen (figur 6.4).

Frågor	Svar	Kommentarer
<b>Process</b> Hur stödjer RUP ett företag till att få människor att arbeta bra tillsammans?	Språk, roller och genom att fokusera på att granskning ses som något positivt. Se en granskning som något man kan lära av. Inspektionsfasen, en värdering av vad för resurser som kan tänkas behövas.	Det finns en direkt koppling mellan kvalitetsbegreppet och de många informella granskningarna i RUP, men att gå så långt som att säga att metoden främjar människor att arbeta bra tillsammans vågar jag inte säga något om. (Skandia Liv)

<p><b>Process</b> Vad gör RUP för att kommunikationen i ett företag skall fungera?</p>	<p>RUP använder en objektorienterad designmodell som hjälper till med förståelse, modifiering och ger en gemensam begreppsbas. Språket som används är UML. (Används mest i design och analys)</p> <p>Processen är standardiserad, vilket gör att människor får en gemensam bas.</p> <p>Namnstandard gör att alla vet vad alla pratar om.</p> <p>Worker gör att alla vet vad som skall göras, vad ens ansvar är, och vilket resultat som skall presteras.</p> <p>Nya medarbetare kan lätt hoppa in, eftersom roller (worker) är väldefinierade.</p>	
<p><b>Process</b> Hur gör RUP för att skapa samarbete? (Vilja att dela med sig av sin kunskap)</p>	<p>En process som alla i organisationen varit delaktiga i att skapa.</p> <p>Klar struktur om vem, vad, hur och när.</p> <p>RUP försöker bryta ner barriärer mellan olika utvecklingsgrupper (designer och analytiker), genom att använda use-case och genom att involvera grupperna tidigare.</p>	<p>Förståelig: 2 Synlig: 2 Stödjande: 1 Robust: 2 Snabbhet: 1 Anpassningsbarhet: 2 Tydlighet i användarroll: 2 Pålitlig: 1 Elegans: 1 (Binomen, Frontec)</p>
<p><b>Process / Produkt</b> Kan vi trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? Om nej, vad är det som hindrar oss?</p>	<p>Ja, genom RUP skall utvecklarna kunna få fram vad intressenterna tycker med hjälp av use-case, prototyper och genom att hela tiden driva kraven (The requirement workflow).</p>	<p>OK. Även Business Modeling hjälper till. (Frontec)</p>
<p><b>Produkt</b> Hur stödjer RUP ett företag till att produkten skapas i rätt tid, budget, med rätt utvecklingsteknik och att produkten blir vad kunden vill ha?</p>	<p>Genom att tänka efter de grunderna som visas i stjärnan (appendix F).</p> <p>Genom att tänka på det använda ett tillvägagångssätt som värnar om kvalitet på process och produkt.</p> <p>Genom att hela tiden ha kontroll över projektet och se redan på ett tidigt stadium om förseningar kommer att inträffa.</p> <p>Slutligen riskhantering och målhantering.</p>	<p>Genom att tidigt definiera en iterationsplan och köra en iteration och därefter svara på frågor som hur mycket resurser behöver vi, vilken tid kommer det här att ta och är det överhuvudtaget genomförbart. (Skandia Liv)</p> <p>Ja, om processen är rätt konfigurerad från början, alternativt att man i praktiken tillåter sig att konfigurera om processen under resans gång. Kort sagt; RUP hjälper duktiga och erfarna människor att bli ännu skickligare, men den kan lika gärna hjälpa mindre skickliga och oerfarna personer att gömma sig bakom en formell fasad. (Binomen)</p>

<p><b>Produkt</b></p> <p>Vilka attribut anser ni vara viktigast i den produkt som RUP skapar?</p>	<p>Pålitlig. Lättunderhållet. Lätt testat. Synlighet.</p>	<p>Användbarhet 2 Korrekthet 1 Lättunderhållet 1 Dokumentation 1 Lätt att lära 1 Snabb utvecklingsfart 1 Funktionalitet 1 Begriplighet 1 Integrerbar 1 Accepterbarhet 1 Funktionalitet 1 Underhåll 1 Testbar 1 (Binomen, Frontec) Robust (Skandia Liv)</p>
<p><b>Produkt</b></p> <p>Vad har RUP för Strategi och mål?</p>	<p>Mål 1 är att möta kundens krav, inom budget och i tid.</p> <p>Mål 2 är att processmodellen blir en OMG-standard och att RUP blir en defacto-standard och en instans av OMG-standard.</p> <p>Filosofin är att utbilda mjukvaruindustrin.</p> <p>Strategin är att stödja webb och e-commerce och för att klara även detta kommer bl.a. nya roller att behövas.</p>	
<p><b>Produkt/Miljö</b></p> <p>Hur stödjer RUP ett företag när oväntade problem uppstår?</p>	<p>Projekt Management Workflow.</p> <p>Den iterativa utvecklingen stödjer riskhantering och omprioritering.</p>	<p>Generellt sett så ger RUP bra överblick, men den dag man använder RUP som ett verktyg för att hantera oväntade problem så bottenar det förmodligen i en osäkerhet hos projektledningen. Kort sagt bör projekt som förlitar sig alltför hårt på metoder och processer i krissituationer - kommer ofelbart att misslyckas! (Binomen)</p>
<p><b>Miljö</b></p> <p>Vilken slags organisationer vänder sig RUP till?</p>	<p>Alla system- och mjukvaruutvecklingsföretag. Allt från konsulter till banker till försäkringsbolag.</p> <p>RUP fungerar på projekt i storlek 5-25 personer. Även på mindre projekt, men då kan antagligen bara bitar av RUP användas.</p>	<p>Vi upplever att RUP kan bli för detaljerad och tung i små projekt. (Skandia Liv)</p> <p>Vi tycker att det är viktigt att poängtera att det inte finns en metod som passar vid alla tillfällen. Man kan dock få den uppfattningen om man läser processleverantörernas reklamlad. (Skandia Liv)</p> <p>Det krävs mycket kunskap för att klara av att konfigurera processen optimalt till verksamheten. (Binomen)</p>



<p><b>Miljö</b> Hur bör information hanteras från olika projekt enligt RUP? (Hur används den? Hur förs kunskapen vidare?)</p>	<p>Komponentbaserad utveckling används, eftersom den stödjer återanvändning av komponenter och arkitekturen. Komponenterna återanvänds mest i iterationerna inom projektet, vid en återanvändning i flera projekt tillkommer än större kostnad för test mm.</p> <p>Worker (Projekt Manager) tar hand om information från iterationerna.</p> <p>The Enviroment Workflow och process konfigurerbarhet stöttar en förbättring av processen.</p>	<p>Även om RUP är komponentbaserad, så säger den inget om hur man skriver återanvändbar kod. Återanvändning är ett eget område som kräver stora insatser för att lyckas. På Skandia har vi använt oss av komponentbaserad utveckling i ett antal år nu och vi återanvänder med stor framgång komponenter, men det rör sig ofta om mer tekniska komponenter, t.ex. kommunikation och säkerhet mm. Att återanvända affärslogik på högre nivå är mycket mer komplicerad och vår vision är att nå dit. (Skandia Liv)</p> <p>Finns inte definierat hur artefakter skall hanteras mellan projekt. Under ett projekt lagras alla artefakter enligt Config &amp; Change Management Workflow. Hur reuse skall gå till får man alltså hitta på själv. (Frontec)</p>
<p><b>Miljö</b> Hur stödjer RUP ett företag i val av rätt teknik och metoder?</p>	<p>Genom "hur", toolmentor och guideline.</p> <p>Best Practices som är att utveckla iterativt, ta hand om krav, använda komponentbaserad utveckling, verifiera mjukvarukvalitet, visuell modelleringsteknik och kontrollerade förändringar i mjukvara. (I RUP finns dessa bitar i grunderna som visas i en stjärna (appendix F))</p>	
<p><b>Miljö</b> Hur stödjer RUP ett företag i val av rätt människor?</p>	<p>The Projekt Management Workflow beskriver vad för kompetens som måste finnas för projekt och om dessa inte finns vet man vad som behövs skaffas.</p> <p>Finns än så länge ingen beskrivning hur team skall gå tillväga.</p> <p>Genom "vem".</p>	<p>Är det verkligen processens ansvar, ligger inte det mer på projektstyrningsmodellen, t.ex. PROPS. Vad betyder kompetens, om någon gått en kurs, så behöver ju det inte betyda att den personen behärskar området. (Skandia Liv)</p> <p>Detta låter bra men fungerar sällan i praktiken. Personer som är "bäst lämpade" för alla typer av uppgifter i processen är antingen "doers" som jobbar nära design och implementation, eller visionära "inventers" som föder ramverket till konstruktionerna, eller "managers" som håller i pengar och tidplaner. Några andra alternativ existerar egentligen inte. Men okej, får man åtminstone ut en lista på "hur många man behöver av varje sort" så kan det eventuellt vara en kontroll på om man har anställt /allokerat rätt personal. (Binomen)</p>

<b>Miljö</b> Hur stödjer RUP ett företag i fördelning av arbete efter den kompetens som finns?	Se frågan ovan. Genom aktiviteter.	RUP stödjer i praktiken inte alls detta. Personkemi går inte att formalisera. (Binomen)
<b>Miljö</b> Hur stödjer RUP ett företag till att ta hänsyn till de externa faktorerna?	Riskhantering som tar hand om t.ex. semestrar. Use-case ger förståelse utöver gränserna. The Business Modeling Workflow skapar förståelse.	
<b>Miljö</b> Hur stödjer RUP ett företag till att skapa rätt relation till kunder och andra intressenter?	The Business Modeling Workflow, genom att försäkra sig om att kunder, utvecklare och användare har en gemensam förståelse av dynamiken i en organisation. Workshop (idéinsamling) Use-case gör att intressenterna och utvecklarna förstår varandra Den iterativa utvecklingen gör att resultat kan ses tidigt och få intressenterna intresserade.	

Figur 6.4: Frågor, svar och kommentarer

## 6.2 Granskning av RUP med hjälp av välgvalsmodellen

Nedan kommer en granskning av RUP att äga rum i syfte att skapa en metaarkitektur av RUP, som klargör vad den har för styrkor och svagheter.

### 6.2.1 Miljö

#### A0. Hur relateras utvecklingsmodellen till IT-organisationens strategiska områden

Vilka strategiska område refererar RUP till?

- Relationer (kundrelationer och leverantörsrelationer)
- Organisationsform
- Humanresurser (dvs. systemutvecklarens kunskaper, kompetens, lojalitet och handlingsfrihet)
- Utvecklingsinfrastruktur (dvs. metoder, tekniker och Case-verktyg)

Det beskrivs i RUP att utvecklingsprocessen påverkas av:

- **Utvecklingsinfrastruktur:** Technical Factors som t.ex. programmeringsspråk, utvecklingsverktyg och existerande system. Domain Factors vilket innebär bland annat applikationsdomän och verksamhetsprocesser att stödja.
- **Organisationsform:** Organizational Factors som innebär organisationsstruktur och kulturen i verksamheten. Lifecycle Factors som beskriver hur långt liv mjukvaran skall ha.
- **Humanresurser:** Organizational Factors så som kompetens, kunskap, erfarenhet, attityd och även worker (roller).
- Företagets **strategi** beror bland annat på hur många intressenter som finns och ju fler desto mer formellt, och detta tas upp i t.ex. Business Case.

## A1. Relationer

Vilka samarbetsformer mellan kundorganisation och IT-organisation omfattas av RUP?

- Avtalad samverkan, (dvs. juridiskt kontrakt)
- Informell samverkan (dvs. socialt kontrakt)
- Kooperation (dvs. man deltar och påverkar varandras beslut)
- Koalition (dvs. man etablera en gemensam men annars autonom enhet)

Kontraktprojekt beskrivs i RUP som något som sker mot en given kund, medan Internal Project är när kund och utvecklare arbetar tillsammans.

En slutsats som dras är att ju flera Stakeholders (köpare, kunder och slutanvändare), desto mer formella bevis (dokument, rapporter och prototyper).

## A2. Organisationsform

### A20. IT-organisationens verksamhetsområde

Hur relateras RUP till IT-organisationens verksamhet? Vilka av de nedanstående kompetensområdena omfattas av RUP?

- Marknadsföring
- Produkt/Systemutveckling
- Kundservice och systemunderhåll
- Utbildning och träning
- Personalservice
- Ekonomi
- Strategisk planering
- Projekt management
- Resurs management
- Infrastruktur management

**Produkt/Systemutveckling:** RUP berör systemutvecklingslivscykeln i sin fyra faser Inception, Elaboration, Construction och slutligen Transition.

**Projekt Management:** Ett av huvudflödena i RUP.

### A21. IT-organisationens beslutsstruktur

Vilken organisations struktur främjas av RUP?

- Hierarkisk
- Nätverk
- Lärande
- Virtuella
- Teamorienterad
- Platt
- Projektorganisationer

**Projektorganisationer** är den typ av samarbetsform som passar bäst för RUP. Virtuella strukturer och nätverksstrukturer kan vara otydliga och därför ibland vara svåra att använda på RUP.

## A22. IT-organisationens sätt att organisera ett utvecklingsprojekt

Vilken arbetsform främjas av RUP?

- Vattenfall (dvs. sekventiell beroende mellan olika aktiviteter)
- Spiral (dvs. inkrementell lärande process)
- Överlappande (dvs. en aktivitet kan starta innan en annan har avslutat)
- Parallell (dvs. många aktiviteter pågår parallellt)

RUP är en iterativ inkrementell process och just Boehms **spiralmodell** är ett känt exempel på en sådan.

## A23. Organisationen sätt att se på livscykel frågan

Vilken form av livscykel stödjer RUP?

- Produktlivscykel (marknadsföring och produkt efterfrågan som avgör livscykel)
- Organismmetaforen (födelse, utveckling och död och händelsebaserad anpassning)
- Ingenjörsmässiga (komponenttänkande och snabbanpassning)
- Arkitekturella (lärande)

Det stod mellan den **ingenjörsmässiga** och den arkitekturella, men när man väl har valt RUP så är det inte meningen att det beslutet skall omvärderas vid varje nytt utvecklingsprojekt därmed är RUP ingenjörsmässig.

## A3. Utvecklingsinfrastruktur

### A30. Modellering

#### A301. Modelltyper

Vilka slags verksamhetsmodeller förespråkas av RUP?

- Funktionella verksamhetsmodeller
- Aktivitetsmodeller
- Systemmodeller
- Beslutsmodeller
- Objektmodeller
- Informella modeller

I RUP är det ett genomgående tema att använda **modeller av olika slag** för att öka förståelsen mellan alla inblandade i projektet.

#### A302. Nödvändiga modelltyper

Vilka slags verksamhetsmodeller anses som nödvändiga av RUP?

- Funktionella verksamhetsmodeller
- Aktivitetsmodeller
- Systemmodeller
- Beslutsmodeller
- Objektmodeller
- Informella modeller

En styrka med RUP är att den kan anpassas efter den storlek, kunskap och kompetens som finns i en verksamhet, och om det är så att en verksamhet anser sig ha en utvecklingsdel som fungerar utan **verksamhetsmodeller** så går det att ordna, men om styrkan i RUP skall utnyttjas till fullo bör de modeller som beskrivs användas. Detta är framförallt viktigt ur kvalitetssynpunkt, eftersom genom att använda RUP får man en gemensamt språk och ökad förståelse vilket leder till bättre chanser till kundtillfredsställelse.

### A303. Modellkategori

Vilken slags modellkategori förespråkas av RUP?

- Grafiska modeller
- Matematiska modeller (exempel Excel)
- Tabeller
- Rika bilder

RUP förespråkare **visuell modelleringsteknik**, detta för att skapa förståelse inom projektet och en gemensam begreppsbas och språket som används är UML. Till intressenter som inte är bekanta med UML kan andra hjälpmedel användas som t.ex. rika bilder (**grafiska modeller**).

För att tydliggöra hur processen fungerar kan man använda **matematiska modeller**.

### A31. Datorstöd

#### A310. Verktygstyp

Vilka slags verktyg förespråkas av RUP?

- CASE (Computer Aided Software Engineering)
- Dokumentationsverktyg
- Rittekniska verktyg

**Alla** dessa **verktyg** förespråkas i RUP. Rational anser att RUP är verktygsberoende, men de vill gärna att deras produkter används. Rational ClearCase, Rational ClearQuest, Rational RequisitePro, Rational Rose, Rational TestManager (SQA Suite), Rational Robot (SQA Suite), Rational Purify, Rational Visual Pure Coverage, Rational Visual Quantify och Rational SoDA är produkter från Rational.

#### A311. Datorstödda aktiviteter

Vilka slags aktiviteter bör ha datorstöd enligt RUP?

- Verksamhetskartläggning
- Kravdokumentation
- Funktionell design
- Arkitekturell design
- Detaljerad design
- Systemdokumentation
- Systemverifikation
- "Reverse engineering" (t.ex. man har ett program och vill få fram källkoden)
- Processdokumentation
- Utvärdering

De **olika delarna bör ha datorstöd**, genom allt från Word till prototyper gjorda i t.ex. Microsoft Visual Basic.

### A32. Metoder

Vilka slags metoder förespråkas av RUP?

- Verksamhetsanalytiska
- Informationsanalytiska
- Systemanalytiska
- Prototyping
- Simuleringsmetoder

Det beskrivs **vad som skall göras, vem, när och hur i processöversikten**. Just hur något skall göras är en metod, och i RUP är det en viktigt bit i deras mål att vara tydlig.

Vilka produkter förespråkar utvecklingsmodellen att det framställas med stöd av en metod?

- Riskanalys
- Kravspecifikation
- Analys
- Designspecifikation
- Detaljerad design specifikation
- Körbart system
- Systemens operationella och förvaltnings förutsättningar

RUP vill vara en strukturerad utvecklingsmodell som hjälper människor att skapa system med rätt process, och målet uppnås enligt RUP genom att ha en **noggrann och tydlig process**, alltså skapas alla delar t.ex. olika artefakter med hjälp av metoder.

Vilken slags dokumentation använder RUP?

- Handbok
- Utbildningsmaterial
- Case-baserat stöd för systemutvecklare
- Case-baserat stöd för teamarbete
- Interaktivt hjälpmedel

Det finns **böcker** att använda som stöd för processen, olika **utbildningsmaterial** att ta del av och **elektroniska dokument** via RUPs CD-ROM.

### **A33. Information**

#### **A330. Information management**

Vilken slags information management filosofi förespråkar RUP?

- Global (tillgänglig för alla)
- Sluten
- Ingen
  
- Differentierad (delat i ansvarsområden)
- Odifferentierad (odelad)
- Ingen
  
- Central (ett enda ställe)
- Lokal (flera ställen)
- Ingen

I utvecklingsmodellen definierar ett antal valmöjligheter och ger en något oklar bild av vad för information management förespråkas, men informationen skall vara **global, centraliserad** och **differentierad**.

#### **A331. Informationshantering i organisationen**

Hur hanteras kunskap från olika projekt i RUP?

- Informellt via människor
- Formellt via dokument eller e-post mm
- Inte alls

Detta är något som RUP inte tar upp, utan utvärderingen **faller utanför dess ram**.

## A4. Humanresurser

### A40. Utbildning och träning

#### A400. Former av lärande

Vilka former av lärande förespråkas av RUP?

- Learning by Teaching
- Learning by Doing (Learning from Experience)
- Learning by Design Support
- Learning by Mentor Support
- Learning by Example

- **Learning by Teaching:** det finns kurser att gå i RUP-användning.
- **Learning by Doing:** vid införandet av RUP förespråkas att det görs ett pilotfall, vilket innebär att införandet sker genom att de viktigaste bitarna införs först och att det sedan sker en utvärdering innan ett nytt varva tas tills att man har nåt en fram till rätt process.
- I RUPs CD-ROM kan inläring ske genom att en person testat sig fram ibland de exempel som finns (**Learning by Example**).
- Det beskrivs att kompetensbrist fylls effektivt med **mentorsledda workshops**.

### A41 Kunskaper och kompetens

#### A410. Teknisk kompetens

Vilken slags design kompetenser bör systemutvecklare besitta enligt RUP?

- Förmåga att skapa lösningar
- Förmåga att upptäcka felaktigheter innan de blir allvarliga problem
- Förmåga att modularisera / systematisera
- Förmåga att modellera
- Förmåga att skapa arkitekturer (dvs. förmåga att skapa överblickbarhet, säkra flexibilitet och skydda kvalitet etc.)
- Förmåga att integrera och samtidigt hålla komplexiteten under kontroll
- Förmåga att förstå kundens verksamhet
- Förmåga att granska systemens tekniska kvalitet
- Förmåga att förstå de verktyg som är aktuella

Beroende på den roll som systemutvecklaren har i projektet krävs **olika kompetenser**. En designer bör t.ex. ha kunskap om Use Case Modeling-teknik, krav på systemet, objektorienterad analys och design teknik o.s.v. medan en person som arbetar med analys måste ha god kommunikationsförmåga.

## A411. Social kompetens

Vilken slags social kompetens bör systemutvecklare besitta enligt RUP?

- Förmåga att lyssna
- Förmåga att samarbeta
- Förmåga att kommunicera med kunden på ett naturligt sätt
- Förmåga att kommunicera med andra systemutvecklare
- Förmåga att vägleda projektdeltagarna
- Förmåga att kunna argumentera i skrift och tal

Som nämns i fråga A410, så krävs det **olika kompetens** av olika deltagare. I RUP är det dock viktigt att tänka på att den försöker bryta ner barriärerna mellan olika grupper av systemutvecklare, och därför kan förmågan att kommunicera här vara viktig i fler roller eftersom man kommunicerar över rollgränserna.

## A412. Managementkompetens

Vilken slags managementkompetens bör systemutvecklare besitta enligt RUP?

- Förmåga att klargöra och förhandla (omförhandla) olika förutsättningar
- Förmåga att organisera och samordna insatserna
- Förmåga att värdera insatsernas resultat
- Förmåga att värdera alternativ
- Förmåga att balansera resp. prioritera
- Förmåga att skapa kognitiv samordning (dvs. förståelse och medvetenhet)
- Förmåga att skapa en social miljö
- Förmåga att vara en "agent of change"
- Förmåga att förstå vilken kompetens som behövs för utvecklingen

Som ovan krävs **olika kompetens** för olika roller. En Project Manager behöver t.ex. förstå vilka kompetenser som krävs i projektet och hålla reda på projektet så att det går åt rätt håll. Däremot står det mycket lite beskrivet om svårigheter med lagarbete och dess relationer, utan här förutsätter man att de medlemmar som valts kan arbeta tillsammans.

## 6.2.2 Systemutvecklingsprocess

### B0. Processens natur

#### B00. Systemutvecklingsomfång

Vilka systemutvecklingsaktiviteter omfattas av RUP?

- Verksamhetsanalys
- Kravspecifikation
- Arkitekturell design
- Detaljerad design
- Implementering (kodning och test)
- Utbildning
- Drift och underhåll
- Avveckling
- Utvärdering av hela projektet

Den avslutande fasen heter **Transition Phase** och vid slutet av denna fas sker en utvärdering som berör om kunden anser att de är nöjda eller om de vill att det skall tas ett varv till. I den avslutande fasen berörs även utbildning av användarna.

Modellen omfattar dock inte drift och underhåll eller avveckling och inte heller hur erfarenheter skall föras vidare mellan projekt.



## **B01. Val av källor för att fylla kunskapsinnehåll**

Hur hanterar RUP kunskapsbristen/ kompetensbristen?

- Kortsiktig samarbete med tredje part
- långsiktigt samarbete med tredje part
- Ny rekrytering
- Vidarutveckling av den kompetens som finns tillgänglig i företaget

De tre övre bitarna berörs inte i utvecklingsmodellen utan ligger utanför dess gränser och det är Project Managers uppgift att se till att kompetensen skaffas fram till projektet. Det nämns lite om hur det inför nya projekt ofta är nödvändigt att starta upp projekten med **att utbilda gruppledammarna** i nya roller som det råder brist på.

## **B02. Kundmedverkan**

### **B020. Samarbete och samverkan mellan kund and systemutvecklare.**

I vilka av nedanstående systemutvecklingsaktiviteter är, enligt RUP-kundens medverkan, nödvändig?

- Verksamhetsanalys
- Kravspecifikation
- Arkitekturell design
- Detaljerad design
- Implementering (kodning och test)
- Utbildning
- Drift och underhåll
- Avveckling
- Utvärdering av hela projektet

I RUP har man tagit fasta på att **kundens medverkan** är viktig i alla faserna.

## **B1. Kommunikationen**

### **B10: Kommunikationen mellan systemutvecklare**

Vilken kommunikationsform mellan systemutvecklare främjas av RUP?

*Direkt, face-to-face*

- Möten
- Workshop
- Prototyping

*Indirekt*

- Brev
- E-post
- Telefon
- Testprototyper
- Fax

Kommunikationen mellan systemutvecklare beskriver RUP inte speciellt mycket, dock beskrivs att **möten** skall ordnas och vilka roller som bör delta. RUP lämnar beslut om det indirekta kommunikationen till deltagarna i projektet.

## B12. Kommunikationen mellan kunden och systemutvecklare

Vilken kommunikationsform mellan kund och systemutvecklare främjas av RUP?

*Direkt, face-to-face*

- Möten
- Workshop
- Prototyping

*Indirekt*

- Brev
- E-post,
- Telefon
- Testprototyper
- Fax

**Direkta kommunikationsformer** är viktigt då de ska tydliggöra kundens krav och inte heller här beskrivs det något om den indirekta kommunikationsformen.

## B2. Samarbete

### B21. Beroende förhållande mellan teamgrupper

Hur hanterar RUP beroende mellan olika team?

- Standardmetod eller teknik
- Tidsplaner
- Planerade möte
- Workshop
- Central informationsbasen
- Informella sociala möte
- Genom att låta den sitta i samma rum

Genom att ha en **tydlig utvecklingsmodell** som beskriver vad som skall göras, till vem det är ämnat och genom att involvera de olika teamen i varandras delar.

### B22. Management av oförutsatta händelser

Hur hanterar RUP oförutsatta händelser?

- Hierarkisk ledning
- Projektledning
- Öppen debatt and medbestämmande

Hantering av oförutsatta händelser hanteras genom **riskanalyser**, och att genom hela projektet ha en aktiv uppdatering av dessa risker. När en medlem i projektet upptäcker problem/risker så meddelas det vidare och beslut tas om hur man skall gå vidare med det.

## 6.2.3 Samordningseffekter (Målbilder)

### C01. Processrelaterade mål

Vilka processrelaterade mål främjas av RUP?

- Minskning av utvecklingskostnader
- Lämplig samordningsgrad
- Minskning av leveranstid
- Ökad garanti för att hålla och synliggöra tidsplanen
- Ökad förmåga att förstå kundens behov
- Ökad möjlighet till att använda optimala resurser
- Förmåga att ta omhand förändringar i krav ökas

Det försöker genom stjärnan (appendix F) och tydligheten i RUP uppnå detta.

### **C02. Systemrelaterade mål**

Vilka system-/produktrelaterade mål främjas av RUP?

- Hög systemkvalitet (funktionalitet, korrekthet och konsistens etc.)
- Hög system flexibilitet
- Lämplig grad av systemintegration
- Hög systemöverblickbarhet
- Snabb systemanpassning till kundverksamhetens föränderliga natur utan att orsaka komplexitet
- Hög kundtillfredsställelse
- Lättbegriplighet (snabb systemöversikt)
- Kostnadseffektiv inom resursernas gränser

Genom att i hela projektet ha en **aktiv kravhantering** så ökas chansen till det ovan nämnda.

### **C03. Service relaterade mål**

Hantera RUP service relaterade mål såsom:

- Väl informerad kund om eventuella förseningar etc.
- Minimala kund klagomål
- Hög tillgänglighet av systemutvecklare
- Kontinuitet (kunden träffar alltid samma systemutvecklare)
- Snabb respons till kundens klagomål
- Hög vilja att bemöta/uppfylla kundens behov

**Produktkvalitet** mm. förespråkas genom hela processen.

#### **6.2.4. Modellens egenskaper**

##### **D0. Modellens generaliseringsgrad**

**Verksamhetssektor.** Vilken slags verksamhetssektor refererar RUP till?

- Privatsektor
- Offentligsektor

Spelar ingen roll!

**Industri.** Vilken slags industri refererar RUP till?

- Handels
- Transport
- Tillverkning
- Tjänster
- Utbildning

Den refererar till **alla industrier** som har avdelningar för utveckling av mjukvarusystem.

**Systemtyp.** Vilka slags systemtyper refererar RUP till?

- Transaktionssystem (TPS)
- Beslutsstödsystem, (BSS)
- Strategiska informationssystem (SIS)
- Rapporteringssystem (MSS)
- Ledningssystem (ESS)
- Ekonomisystem (MIS)
- Interorganisatoriska informationssystem (IOS)
- Office informationssystem (OIS)
- Distribuerade informationssystem (DIS)

- Federativa informationssystem (FIS)

RUP ger exempel på **olika typer av system** som kan skapas t.ex. nämns de ikryssade ovan. Det beskrivs att olika typer av system bör hanteras på olika sätt vad det gäller säkerhet, minne och informationshantering mm.

## 6.3 Sammanfattning

Sammanfattningen kommer nu att ske utifrån process, produkt och miljö, och dessutom utifrån två bedömningar:

- Den objektiva som sker med hjälp av avsnitt 6.2
- Den subjektiv bedömning utifrån avsnitt 6.1.4

### 6.3.1 Kan RUP säkra processkvalitet?

#### 6.3.1.1 Objektiv bedömning

RUPs **klara sidor** för att säkra processkvalitet kan anges i termer av:

- **Relativt heltäckande systemutvecklingsomfång** med kundmedverkan genom alla faserna.
- **Relativt välformad möjlighet till vidareutbildning** för att öka kompetensen under projekt.
- **Kommunikationsformerna mellan systemutvecklare** t.ex. möten och workshops.
- **Kommunikationsformerna mellan systemutvecklare och kund** t.ex. möten, workshops och prototyping.
- Det finns **såväl formella som informella samarbetsformer**.
- Modellen erbjuder **riskanalyser** för att klara av oförutsedda händelser.
- RUP försöker genom att ha en bra kontakt med kunden skapa **förutsättningarna för att synliggöra och hålla tidsplanen**.
- Har en **aktiv kravhantering**.
- Försöker nå en lämplig **samordningsgrad**.

RUPs **oklara sidor** för att säkra processkvalitet i termer av:

- **Underhåll och drift, avveckling och utvärdering** av hela projektet ligger utanför RUP. (B00)
- För att **hantera kunskapsbrist** beskrivs bara exemplet med kompetensutveckling. Någon kunskap om nyrekrytering eller samarbete med tredje part hanteras inte. (B01)
- Det är oklart hur de **indirekta formerna av kommunikation**, mellan systemutvecklare och mellan systemutvecklare och kund, används. (B10 och B12)
- **Formativ kunskap** försummas p.g.a. att t.ex. fysiska distanserna inte betraktas.
- Relativt oklart hur **oförutsedda händelser** skall hanteras. (B22)
- Det är oklart om en **minskning av utvecklingskostnader och leveranstid** sker. (C01)

#### 6.3.1.2 Subjektiv bedömning

Den subjektiva bedömningen stämmer här i stort sett överens med ovanstående bild.

RUP verkar ha tagit fasta på att genom modeller, språk och olika granskningsätt så kan vi försäkra samarbete. Detta är inte tillräckligt.

Oförutsedda händelser är en annan del som RUP tycker sig lösa genom riskanalys, men den dagen man bara använder riskanalys för att hantera detta är man ute på hal is.

De viktigaste attributen i RUP är att den är tydlig, förståelig, anpassningsbar, tydlig i användarroll och robust för framtida förändringar. Allt detta bottnar i det självförtroende som RUP vill locka fram.

RUP kan genom att först gå igenom en iteration, och sedan ge svar på frågor angående tid mm. skapa en bra grund för att lyckas hålla tidsplanen. Den anser att det är viktigt att ha nära kontakt med kunderna och att när problem uppstår delge detta till kunden för att kunna uppdatera tidsplanen.

### 6.3.2 Kan RUP säkra produktkvalitet?

#### 6.3.1.1 Objektiv bedömning

RUPs **klara sidor** för att säkra produktkvalitet i termer av:

- **Systemrelaterade krav på systemkvalitet, flexibilitet, systemintegration** beskrivs utifrån systemutvecklaren.
- **Produktkrav som snabbanpassliga och lättbegripliga** program utlovas.
- **Kundrelaterade krav på tillfredsställelse** utlovas.
- Många **systemrelaterade mål** hanteras av modellen.

RUPs **oklara sidor** för att säkra produktkvalitet i termer av:

- Det är dock oklart hur **systemöverblickbarhet och kostnadseffektiviteten** uppnås. (C02)
- I princip **odifferentierad produkt**, alltså den vänder sig till systemutveckling av produkter. (A20)

#### 6.3.1.2 Subjektiv bedömning

Även här stämmer den subjektiva bilden väl överens med den objektiva.

RUP har försöker skapa rätt produkt genom att se kundkrav som något föränderligt och språket som något som måste standardiseras för att alla skall förstå varandra.

Det finns en otydlighet i vad som anses vara de viktigaste med den produkt som RUP skapar. Antagligen grundar sig denna osäkerhet i att RUP inte på ett tydligt sätt definierat vilka system den lämpas för.

I RUP beskrivs att komponenter återanvändas men dock inte hur man bör skriva återanvändbar kod.

### 6.3.3 Kan RUP säkra miljö kvalitet?

#### 6.3.1.1 Objektiv bedömning

RUPs **klara sidor** för att säkra miljö kvalitet i termer av:

- Utvecklingsmodellen refererar till alla de **viktiga strategiska områdena** som finns i en IT-organisation.
- Berör **samverkan och Kooperation**.
- Utvecklingsmodellen vänder sig till **projektorganisationer**.

- Systemutvecklingsprojekt organiseras enligt **spiralmodellen**.
- **Ingenjörsmässiga livscykeln** och detta stämmer överens med spiraltänkandet.
- Relativt omfattande vad det gäller olika **modelleringstyper** (både formella och informella). RUP är även beroende av sina modeller för att säkra kvalitet.
- Alla aktiviteter i modellen bör ske med hjälp av **datorstöd**.
- Den är utrustad med de nödvändiga **metoderna som behövs i systemutvecklingsprocessen**.
- **Differentierad dokumentation för alla kategorier av intressenter**
- Använder sig av **rika bilder**.
- Refererar till såväl **teknisk, social som managementkompetens**.
- Skapar **optimering av resurserna**.

RUPs **oklara sidor** för att säkra miljö kvalitet i termer av:

- Informationshanteringen är global och differentierad, men **hanteras endast under projektets gång**. (A330)
- Oklar bild på hur **managementkompetensen skall samordnas**. (A412)
- Det är oklart hur man skall skapa **hög tillgänglighet av systemutvecklare**. (C03)

#### 6.3.1.2 Subjektiv bedömning

I stort sätt sammanfaller den empiriska bilden med den formella vägvals bilden.

RUP lämpar sig i projekt som är i storleken 5-25 personer, alltså är RUP inget för riktigt stora projekt.

Ett frågetecken är om RUP är tillräckligt vid val av rätt människor, detta ansågs nämligen ligga på någon projektstyrningsmodellens ansvar. Det är sant att man i RUP kan se vad var och en skall göra, men valet av kompetens ligger på projektledaren.

Under processens gång beskrivs hur informationen bör lagras, men efter projektets slut beskrivs det ej hur man skall ta vara på informationen, man bortser alltså från Knowledge Management.

## 7. Slutsatser och rekommendationer

Min studie har haft titeln:

### **RUP – en modell för kvalitetsfrämjande systemutveckling?**

Detta är en mycket intressant fråga som inte kan behandlas i ett entydigt svar, då modeller såsom RUP beror av den situation som den ställs inför. I detta sista kapitel kommer jag att sammanfatta min tolkning av RUP, och utifrån materialet dra några slutsatser om hur RUP bör användas för att ge de förväntade effekterna i systemutveckling, d.v.s. säkring av process, produkt och miljö kvalitet. Jag sammanfattar mina slutsatser genom en modell som försöker klargöra RUP-modellens fyra olika användningsmöjligheter.

### **7.1 En modell för att bedöma RUPs tillämpningsområde**

Det är ingen tvekan om att hela 90-talet har präglats av kvalitetssträvande och utveckling av olika metoder/modeller för att just säkra kvalitet. En del av den osäkerhet som förekommer med kvalitetsfrågor försöker man absorbera med hjälp av modeller/metoder, men bara en del kan absorberas med hjälp av strukturerade, explicita och professionella kunskaper. RUP är ett typiskt exempel av en sådan produkt, intensionen bakom den sammanfattas m.a.o. i termer av kvalitetssäkring genom osäkerhetsminskning.

Modellen<sup>5</sup> kommer att synliggöra de områden där RUP har sina starkaste fästen, och även övriga områden där RUP kan vara ett lämpligt instrument om det kompletteras med ytterligare metoder och modeller. Sammanfattningsvis kan osäkerhet ses som ett resultat av den kunskap som behövs för att bedriva en kvalitetsfrämjande systemutveckling och den kunskap som finns tillgängligt för detta ändamål - detta utgör den första dimensionen i min bedömningsmodell.

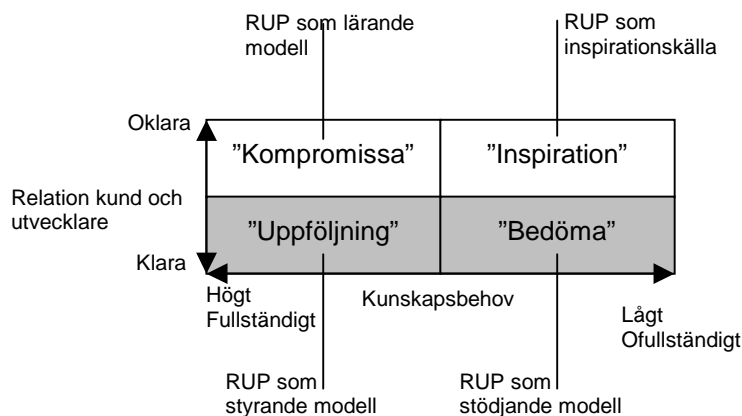
Vidare präglas systemutveckling av de emotionella aspekterna som har en direkt koppling till den sociala miljön som formas av relationer mellan kund och systemutvecklare. Dessa kan vara klara då det finns en fullständig enighet om hur systemutvecklingen skall bedrivas och hur beslut bör fattas o.s.v. - denna relation utgör min andra dimension i modellen (figur 7.1).

---

<sup>5</sup> Modellen härleds från J.Thomsons beslutsteori (Hur organisationer fungerar, 1967) och sammanfattar flera olika teorier: 1) Argyris strävan att balansera rationellt och emotionellt i lärande syfte. 2) Boland strävan att balansera hårt och mjukt tänkande. 3) Hedbergs föreställning om systemutv. såsom en rationell process där rationalitet beskrivs i kunskaps tillgänglighet, medan politiska dimensionen representera de olika intressen som vanligtvis förekommer i systemutvecklingen. 4) Hoffmans modell strävar efter att samordna affärsutv. och systemutv. utifrån strategiska och meningsfullt perspektiv. 5) Checklands strävan att nå överblickbarhet genom formella och informella metoder. (Maguolas, Kalevi, 1998)

De förhållanden mellan dessa två dimensioner definierar fyra olika utvecklingssituationer, där var och en av dessa reflekterar olika grader av osäkerhet:

- Den första situationen förutsätter att **balansen mellan behövlig och tillgänglig kunskap** är relativt hög, samtidigt som relationerna mellan kund och systemutvecklare är tydlig (t.ex. genom mångårigt samarbete och förtroende).
- Den andra situationen förekommer i de fall där vi trots att vi har klara harmoniska relationer har en **omfattande kunskapsbrist**, och därmed stor osäkerhet.
- Den tredje situationen reflekteras i att kunskapen är balanserad men **förhållandena mellan utvecklare och kund är konfliktladdade**.
- Den fjärde situationen kan sammanfattas i termer av **enorm osäkerhet och oklara förhållanden mellan kund och utvecklare**.



Figur 7.1: Modell för fyra olika situationer.

### 7.1.1 RUP som styrande modell

Jag kan tänka mig att RUP är användbar i situationer som karaktäriseras av klara relationer med hög säkerhet (balans mellan kunskap och relationer), om och endast om det iterativa mönstret omvandlas till ett sekventiellt mönster (t.ex. vattenfallsmodellen). I sekventiella utvecklingar kan vi genom planering nå kvalitet p.g.a. att kunden och utvecklare vet vad för produkt som skall skapas. Alla situationer är inte unika, redovisningssystem och lönesystem är t.ex. i princip inte definierade av kunden och i sådana situationer är återanvändning av komponenter mycket användbart. RUP bör alltså beskriva hur anpassningen bör ske efter denna situation.

### 7.1.2 RUP som stödjande modell

RUP kan i situationer som präglas av klara förhållanden, men där kunskapen är obalans vara ett bra instrument. Detta eftersom RUP är en iterativ utvecklingsprocess, och i iterativtänkande ligger just idén att utveckling inte kan planeras för långt i förväg beroende på kunskapsbristen som ofta finns. Kunskapsbristen bör kompletteras, precis som i RUP, med olika råd, modeller och metoder.

### 7.1.3 RUP som lärande modell

När situationen präglas av oklara förhållanden men med ett klart kunskapsbehov, kan man tänka sig att RUP skulle kunna användas som lärande instrument. För att RUP skall klara detta bör den kompletteras med modeller och metoder för att säkra samarbete, främja dialoger och stärka kommunikation. Checklands SSM-modell (Maguolas, Kalevi, 1998) skulle t.ex.



vara ett bra alternativ, då den kombinerar både formella och informella metoder för att hjälpa kunden och systemutvecklarna att skapa en bra relation.

#### 7.1.4 RUP som inspirationskälla och idégenerator

Enligt min tolkning har RUP vissa förutsättningar för att tillämpas även i miljöer där osäkerheten är stor och relationerna oklara. Här kan RUP bara användas som idékälla, då den inte är tillräcklig för att själv handskas med komplexa situationer. För att klara detta måste den kompletteras med andra modeller (modellallians).

## 7.2 Slutord

### **Hur uppfyller en systemutvecklingsmodell i allmänhet och RUP-modellen i synnerhet den kvalitetsbild som berör systemutvecklingens tre grundpelare, process, produkt och miljö?**

Huvudfrågan för denna uppsats (se ovan) har byggts på hur RUP kan klara av att nå kvalitet, och för att besvara det har jag försökt skapa mig en bild av miljö, process och produkt. Dessa tre områden har teoretiskt (kapitel 3) och empiriskt (kapitel 4) undersökts för att få en behovsbild av kvalitet. Denna har sedan syftat som indata för den vägvalsmodell (kapitel 5) som jag skapat.

Ju mer mobilt och heterogent samhället blir, desto viktigare blir modeller och metoder i kampen för att främja kvalitet. Modeller och metoder stödjer oss för att överblicka, synkronisera och samordna utvecklingen för att öka förutsättningarna att nå kvalitetsmål, minskade kostnader och minimering av tid. **Det är omöjligt för en modell, att i detta samhälle tillgodose alla situationer, därför är en vägvalsmodell ett viktigt medel för att kunna välja de rätta modellerna i rätt situation.**

Från granskningen av RUP (avsnitt 6.3) fick jag fram en bild av RUP som en:

- **Relativt heltäckande modell.**
- En modell som **stödjer sig på modeller, rekommendationer och teknik.**
- **Iterativt** för att ta kunna handskas med förändringar mm.
- Använder sig av ett **gemensamt språk** för att minska missförstånd.
- mm.

RUP saknar dock:

- **Metoder för konfliktlösning** mellan intressenter (RUP arbetar dock aktivt för att få upp konflikter till ytan).
- **”Strategier” för Knowledge Management.**
- **Tydligt varumärke.**
- mm.

Slutligen har **RUP förutsättningarna att skapa bra kvalitet i situationer som uppföljningssituationen och bedömningssituationen (figur 7.1), om det finns en bra organisation med kunniga människor och rätt resurser.**

#### 7.2.1 Vidare studier

Under min undersöknings gång har det dykt upp frågor som kan vara intressanta att utreda vidare bl.a. kan jag rekommendera två områden:

1. För den som är intresserad av utvecklingsprocesser rekommenderar jag att **utveckla min vägvalsmodell** vidare, göra den mer omfattande och mer tydlig. Fortsätt sedan gärna med en jämförelse mellan ett antal olika metoder/modeller för att skapa en karta över de mest relevanta. Denna karta skulle fungera som hjälpmedel vid val av en systemutvecklingsmodell.
2. Ju mer mobilt och ju större krav man ställer på kvalitet, desto nödvändigare blir det att kartlägga **vilka instrument som en metod/modell bör vara utrustad med** för att effektivisera kommunikation och samarbete. Vilken slags teknik rekommenderas för olika situationer, och hur kan kvalitet stärkas med hjälp av dessa?

## 8. Referenser

### 8.1 Böcker

- Aaker, David A, *Strategic Market Management*, John Wiley & Sons, 1997, ISBN 0-471-30956-7
- Andersen, Erling S, *Systemutveckling - principer, metoder och tekniker*, Studentlitteratur, 1994, ISBN 91-44-31042-0
- Avison D. E, Fitzgerald G, *Information Systems Development: Methodologies, Techniques and Tools*, McGraw-Hill Book Company Europa, 1998, ISBN 0-07-709-233-3
- Bernus, Peter, Nemes, Laszlo, Williams, Theodore J, *Architectures for Enterprise Integration*, Chapman & Hall, 1996
- Bennett J. P. H, Culverhouse P. F, Hughes D. R, *World Class Performance by Design in International operations crossing borders in manufacturing and Service*, New Elsevier Science Puplichers, 1992
- Blomqvist, Rolf och Haeger, Tomas, *Kvalitetsutveckling*, IHM Förlag AB, 1996
- Bolman, Lee G, Deal, Terrence E, *Nya perspektiv på organisation och ledarskap*, Studentlitteratur, 1997, ISBN 91-44-00610-1
- Dahlbom, bo, Mathiassen, Lars, *Computer in context*, Liber ekonomi, 1993, ISBN 91-47-04023-8
- Flynn, Donal J, *Information systems requirements: Determination and analysis*, McGraw, 1997, ISBN 0077093089
- Humphrey, Watts S, *Managing the software Process*, Addison-Westley Publishing Company, 1990, ISBN 0-201-18095-2
- Hägerfors, Ann, *Att samlära om systemdesign*, studentlitteratur, 1995, ISBN 91-44-60411-4
- Jacobson, Ivar, *Object-Oriented Software Engineering*, Addison-Wesley, 1996, ISBN 0-201-54435-0
- Kotler, Philip, Armstrong, Gary, Saunders, John, Wong, Veronica, *Principles of Marketing*, Prentice Hall Europé, 1999, ISBN 0-13-262254-8
- Kruchten, Philippe, *The Rational Unified Process*, Addison-Wesley, 1998, ISBN 0-201-60459-0
- Magoulas, Thanos, Kalevi, Pessi, *Strategisk IT-management*, Vasastadens Bokbinderi AB, 1998, ISSN-1400-741X
- Mathiassen, Lars, Munk-Madsen, Andreas, Nielsen, Peter A, Stage, Jan, *Objektorienterad analys och design*, Studentlitteratur, 1998, ISBN 91-44-00693-4
- Miller, Irwin, Miller, Marylees, *Statistical methods for Quality*, Prentice hall, 1995, ISBN 0-13-013749-9
- Molina, A, Kusiaka, A, Sanchez, J, *Handbook of lifecycle engineering*, Kluwer academic publishers, 1998
- Nordkvist, Håkan, *Kvalitetssystemet ISO 9000*, Liber Utbildning, 1996

Rolland, Colette, Perniu, Barbara, Thanos, Constantino, , *Advanced Information System Engineering (A comprehensive view of Process Engineering)*, Springer International Conference, 1998, ISBN 3-540-64556-X

Sandholm, Lennart, *Kvalitetsstyrning med total kvalitet*, Studentlitteratur, 1995, ISBN 91-44-17183-8

Paashuis, Victor, *The organisation of Integrated Product Development*, Springer Verlag, 1997, ISBN 3540762256

Sommerville, Ian, *Software Engineering*, Addison-Wesley, 1997, ISBN 0-201-42765-6

Svenska språknämnden, *Svenska Skrivregler*, Almqvist & Wiksell, 1998, ISBN 91-21-11280-0

Bokförlaget Bra Böcker, *Nationalencyklopedin*, Bokförlaget Bra Böcker, 1993, elfte bandet

### 8.1.1 Häfte

*Praktisk kvalitetssäkring PQS*, Enator (Prototyp)

### 8.1.2 Magister uppsats

Lerman, Sara, Persson, Anna-Karin, *Analyzing the Impact of a New Software Development Process at IMIPD*, Institutionen för Matematik, natur-, och datorvetenskap, Universitetet i Gävle, 1999

## 8.2 Internet

Wallström, Martin, *Spara pengar på bättre kvalitet*, Computer Sweden, avdelningen Perspektiv, 1996, nr 27 <http://domino.idg.se/cs/artikel.nsf/>

Dorfman, Merlin, *Requirements Engineering*, 1999, <http://interactive.sei.cmu.edu/1999/March/Background/background.mar99.htm>

Sanders, Bruce W, *Software Lifecycles*, 1998-1999 <http://www.cs.colorado.edu/~sanders/guide/lifecycle/lifecycle.html>

*Ordbok*, [http://www.cs.umu.se/~fredrik/hela\\_ordboken.html](http://www.cs.umu.se/~fredrik/hela_ordboken.html)

## 8.3 Artiklar

Mylopoulos, John, *Information modeling in the time of revolution*, 1998, Information system vol 23 no3/4 sida 127-135

## 8.4 Övrigt

Magoulas, Thanos, *Thanos anteckningar*

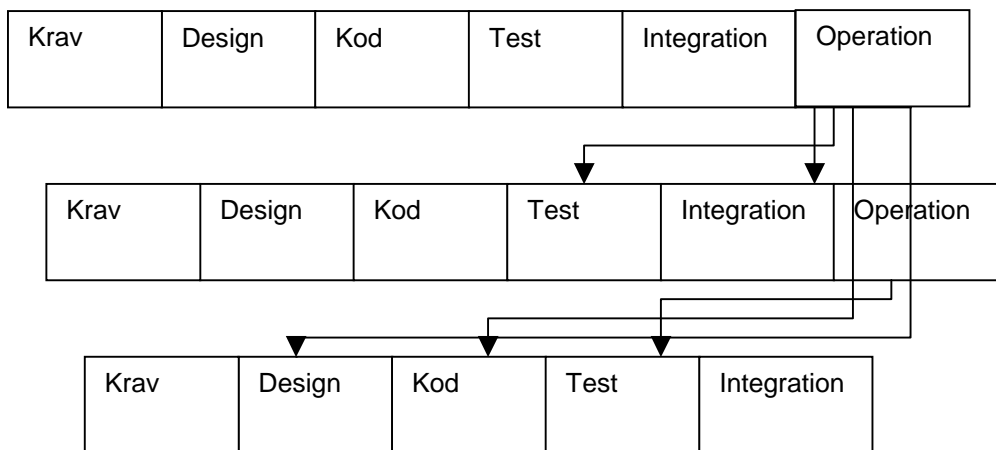
*Rational Unified Process CD-ROM version 5.1*, Rational Software Corporation. 1998

# Appendix

## A. Evolutionär, inkrementell och iterativ utveckling

### A.1 Evolutionär utveckling

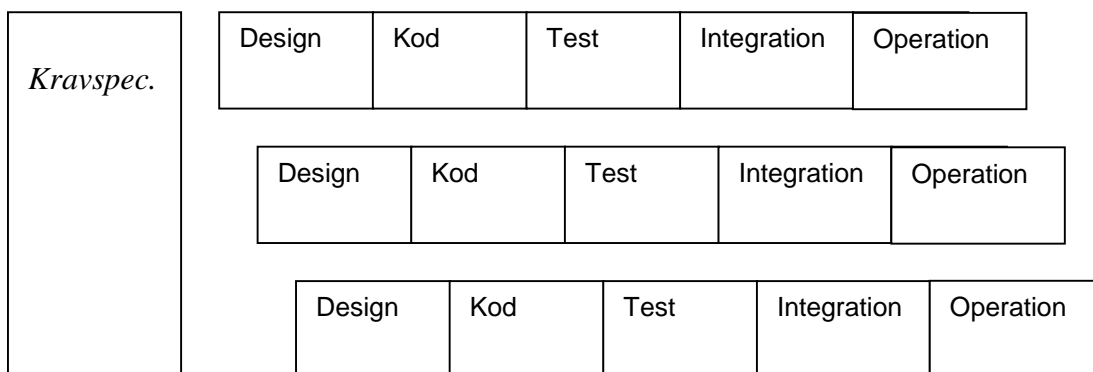
Den traditionella vattenfallsmetoden har även kallats den revolutionära utvecklingsmetoden, då implementeringen sker i ett svep och sedan skall det nya systemet vara igång. Det ha dock funnit en skepsis mot den då den ej tar hänsyn till att kravspecifikationen oftast förändras vid långa utvecklingstider. Ur denna skepsism har den evolutionära utvecklingen utvecklats. (Andersen, 1994)



Figur A.1: Evolutionär utveckling (Dorfman, 1999)

**Evolutionär utveckling innebär att det produceras ett antal delprodukter som efter utveckling skall användas i företaget.** Efter att en delprodukt implementerats ges feedback som kommer att påverka resten av utvecklingen, och varje leverans är alltså en förbättring av förra delprodukten och leder till en klarare och klarare bild av slutprodukten. Utvecklingsprocesserna överlappar varandra (figur A.1). (Dorfman, 1999, Andersen, 1994)

Evolutionär utveckling liknar prototyping med skillnaden att det inte finns en slutversion utan utvecklingen fortsätter. Även när systemet är operationellt fortsätter utvecklingen och systemet behöver aldrig ”dö” utan kan hela tiden utvecklas efter nya krav. (Avison och Fitzgerald, 1998)



Figur A.2 : Klassisk inkrementell utveckling (Dorfman, 1999)

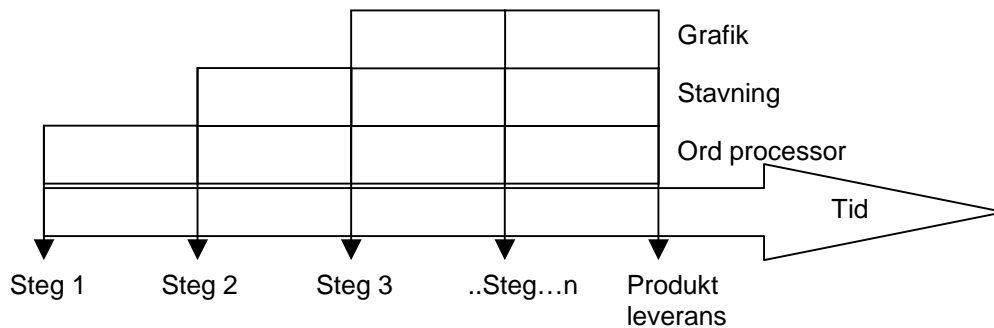
## A.2 Inkrementell utveckling

**Inkrementell utveckling innebär att införandet av kravspecifikation sker i ett antal inkrement (tillägg) som är distinkta men som kan överlappa varandra.** Idén med denna utvecklingsmodell var från början att kravspecifikationen skulle vara statisk (figur A.2). (Dorfman, 1999, Sanders, 1998) Detta sätt att utveckla har av Sanders i artikeln *Software Lifecycles* fått namnet inkrementell implementeringsmodell.

Idag har det ruckats på regeln om statisk kravspecifikation och det är tillåtet att förändra den om så behövs, alltså liknar den mer och mer den evolutionära utvecklingsmetoden.

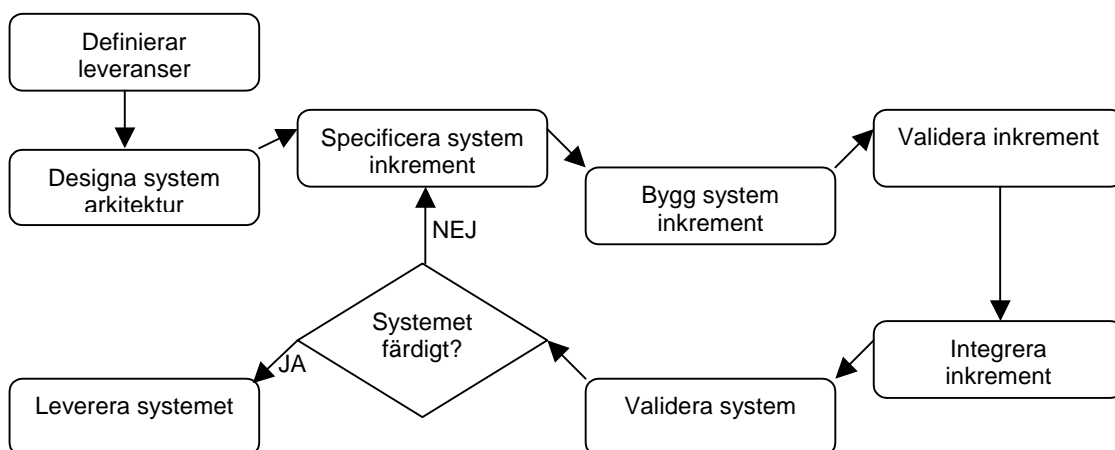
Denna mer moderna tolkningen av inkrement får ett annat utseende (figur A.3), då systemet utvecklas successivt tills att en tillräcklig nivå är uppnådd. Varje utvecklingssteg (inkrement) är en livscykel. (Jacobson, 1996)

Detta utvecklingsätt har fördelen att det är lättare att utveckla inkrement än att gå på helheten direkt (Sanders, 1998). I artikeln *Software lifecycle* kallar Sanders den för *inkrementell utveckling och levererings modell*.



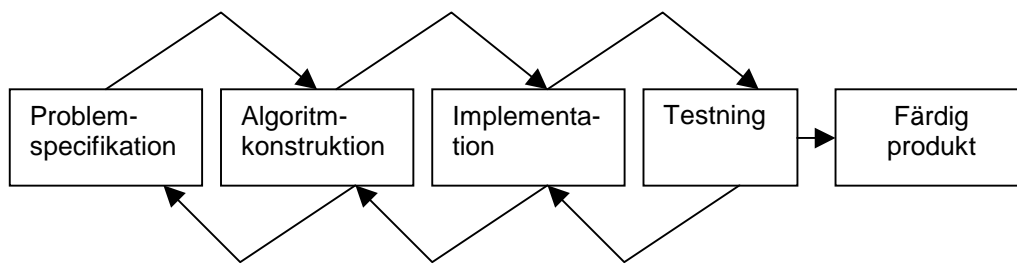
**Figur A.3:** Ett exempel på en modern utvecklingsprocess (Jacobson, 1996)

Enligt *Object-Oriented Software Engineering* av Ivan Jacobson finns det en känd regel som säger att när ett delsystem levererats så uppkommer nya krav på förbättringar. Inkrementell utveckling har detta i tankarna, och istället för att använda en vattenfallsmodell så utgår den från funktioner, och tillvägagångssättet blir följande: en funktion levereras, synpunkter tas till vara och nya funktioner kan läggas till (figur A.4).



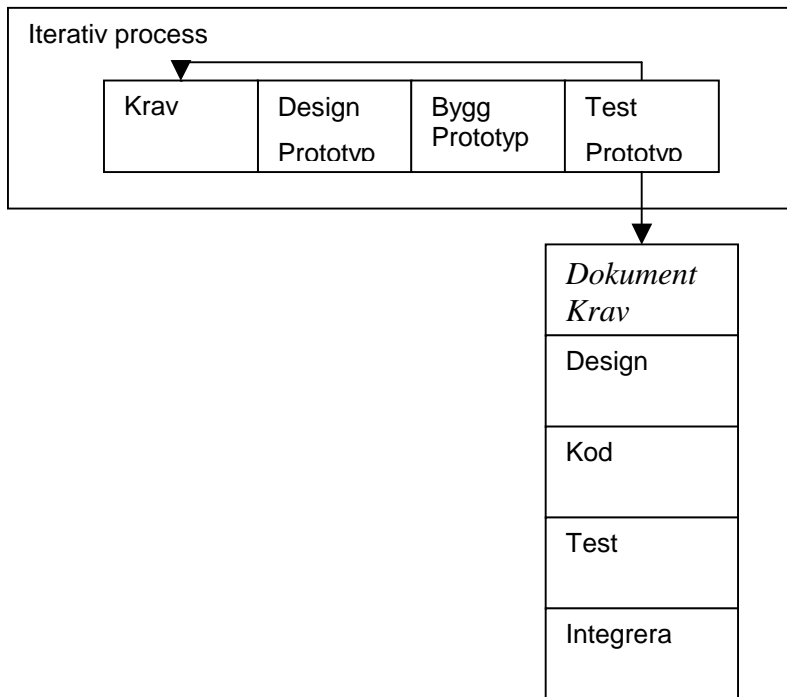
**Figur A.4:** Inkrementell utveckling steg för steg (Sommerville, 1997)

### A.3 Iterativ utveckling



**Figur A.5:** Iterationer som går mot strömmen (Ordbok på internet)

Livcykelmodellen kallas för "strömkarlsmodellen" då den alltid rinner med strömmen. Iterationen har att göra med att se tillbaka, att gå mot strömmen (figur A.5). **Med iterativ process menas enligt ordboken "att man släpper en testversion och ser hur den fungerar och genom erfarenheten går man sedan vidare och förbättrar programmet".**



**Figur A.6:** Prototyping (Dorfman, 1999)

Ett exempel på iterativutveckling är prototyping som visas i figur A.6. När prototyping används byggs dessa med minimala formaliteter och testas sedan på användarna för att se om kravspecifikationen bör uppdateras, om så är fallet görs ofta en ny prototyp o.s.v. Prototyper används idag på andra sätt än förut, då verktyg har gjort det möjligt att använda prototyperna i det slutliga systemet, alltså kan prototypningsverktyg idag användas på ett mer evolutionärt sätt. (Dorfman, 1999)

#### A.4 Slutsats

	Evolutionär	Inkrementell	Iterativ
Definition	Utveckling sker i delprodukter som efter implementering utvärderas och påverkar den fortsatta utveckling.	<b>Klassisk:</b> Införande i ett antal inkrement, där utvecklingens kravspecifikation är statisk, medan resten av livcykeln går igenom i varje inkrement. <b>Modern:</b> Varje inkrement är en livscykel, och efter att en ny funktion har bildas tas åsikter upp som påverkar det fortsatta arbetet.	Att med hjälp av minimala formaliteter bygga en testversion som sedan utvärderas och om svaren fortfarande är oklara byggs en ny testversion... sedan sker utvecklingsarbetet när svaren är klara.  Utvecklingsarbetet sker antingen genom att bygga på testversionen eller genom att utveckla nytt med hjälp av erfarenheten från testversionen.
Likhet	Modern inkrementell utveckling liknar mycket den evolutionära.  Iterativ utveckling kan användas i evolutionär utveckling.	Evolutionära likar mycket den moderna inkrementell utvecklingen.  Iterativ utveckling kan användas i Inkrementell utveckling.	Iterativ utveckling används ofta i andra systemutveckling-tillvägagångssätt.



## B. Intervju med Pär Jansson

**Titel:** Product manager (Rational)

**Namn:** Pär Jansson

**Datum:** 6/8 kl. 18.30-20.00 Anteckningar från intervjun som skedde per telefon till USA.

### Vilken slags organisationer vänder sig RUP till?

- Alla system- och mjukvaruutvecklingsföretag. Allt från konsulter till banker till försäkringsbolag. Alla **företag som sysslar med programvaruutveckling**.
- RUP fungerar på projekt i storlek 5-25 personer, men även för mindre projekt fast då måste en prioritering ske och man använder sig istället av de bitar som anses mest relevant.

### Vad gör RUP för att kommunikationen i ett företag skall fungera?

- Just att **processen är standardiserad** gör att människor får en gemensam bas.
- Kommunikationen fungerar bra för att **alla vet vad som skall göras** genom sina roller= worker. I worker står det precis vad den har för ansvar och vad för resultat som krävs av den.
- Genom att ha en process som alla arbetar efter är det lätt för nya medarbetare att komma in, eftersom det bara är att säga vilken worker den nya medarbetaren skall vara och sedan ser den snabbt vad som skall göra.
- När man väljer RUP så väljer man ett sätt att kommunicera och namnen på saker blir standardiserade. Detta uppskattas av vissa företag medan andra anser att de har egna namn, men efter ett tag upptäcker företagen ofta att genom **namnstandard** så ökar chansen till kommunikation där alla vet vad alla talar om.
- UML hjälper vid analys och design.

### Hur gör RUP för att skapa samverkan? (Vilja att dela med sig av sin kunskap)

Ett vanligt problem är att det finns ett gäng som gör analysen och ett gäng som gör design och ett tredje som sköter implementering. Efter att analysgänget är klart lämnar de över till design, som blir arga för att de tycker att analysgänget inte fattat någonting och att de inte kan använda det som de gjort. Det bildas barriärer mellan de olika grupperna.

- Målet i RUP är att **bryta ner barriärerna** och det görs genom Use-case. I beskrivningen av dessa står det vem som skall få use-case efter att det är klart, alltså görs inte use-case bara med kunden i huvudet utan även den worker som skall använda det senare.
- Ett annat sätt som nedbrytning av barriärerna sker är **genom att grupper blandas och involveras tidigare**. Det finns inte klara gränser emellan dem. Ex. verktyget Rose används i analys men genererar även kod.

### Hur bör information hanteras från olika projekt enligt RUP? (Hur används den? Hur förs kunskapen vidare?)

- I RUP finns det bland annat project manager (worker) som tar hand om information från varje iteration. (En project manager försöker generellt att hålla projekt teamet på rätt spår, koordinera relationen med kund och användare, försäkra sig om kvalitet i projektets artefakter och se till så att förändringar i produkt sker effektivt. (CD RUP))
- Det finns beskrivet att **arkitekturen** antingen kan köpas eller använda sådana som finns eller göra nya.

- **Komponentbaserad utveckling** ger dig möjlighet att återanvända komponenterna, men om så är fallet blir det en extra kostnad då kvaliteten måste vara högre och testas mer. Komponenterna återanvänds mest i iterationerna inom ett projekt.

### Vad har RUP för Strategi och mål?

- Mål, att processmodellen blir en **OMG-standard** och att RUP blir en defacto standard och en instans av OMG-standard. För att lyckas med detta har de samarbete med bland annat Microsoft och IBM. De anser sig ha kommit en bit på vägen då företag använder RUP som ett argument att få olika projekt, genom RUP kan de på ett tydligt sätt beskriva sina mål mm.
- Filosofin är att **utbilda mjukvaruindustrin**, och det sägs behövas då objektorientering inte är så stort som man kan tro.
- Strategin är att **stödja web och e-commerce**, för att klara även detta kommer nya roller mm att behövas.

### Hur stödjer RUP ett företag i val av rätt teknik och metoder?

- **Best practices** som är att utveckla iterativt, ta hand om krav, använda komponentbaserad utveckling, verifiera mjukvarukvalitet, visuell modelleringsteknik och kontrollerade förändringar i mjukvara. (I RUP finns dessa bitar i grunderna som visas i en stjärna). Organisationen bör jämföra sig med dessa och se vad de saknar och börja där.
- **Hur**, genom guideline (beskriver bland annat hur artefakter utvecklas) och toolmentor (ger en detaljerad beskrivning om hur rational Software tools kan ge support till vissa aktiviteter (CD RUP)).
- En styrka med RUP är att det är verktygsberoende om man så vill. Du kan välja ett annat verktyg än det som finns som förslag.

### Hur stödjer RUP ett företag i val av rätt människor?

RUP beskriver än så länge inte hur ett team skulle kunna byggas upp.

- **The Project management workflow** beskriver vad för kompetens som måste finnas för projekt och om dessa inte finns vet man vad som behövs skaffas.

### Hur stödjer RUP ett företag till att få människor att arbetar bra tillsammans?

- Se fråga 7
- Genom **aktiviteter**

### Hur stödjer RUP ett företag i fördelning av arbete efter den kompetens som finns?

- **Språk**, roller och genom att fokusera på att få granskning som något positivt. Se en granskning som något man kan lära av, få helhet och stil.
- Förhoppningen är att RUP-intresserade använder sig av work through mm

### Hur stödjer RUP ett företag till att skapa rätt relation till kunder och andra intressenter?

- **The Business modeling workflow**
- **Workshop** (idéinsamling)
- **Use-case** gör att intressenterna och utvecklarna förstår varandra

- Genom att få **intressenterna intresserade**, då det vid iterativutveckling ges resultat fort. Intressenterna kan få se de olika artefakterna som produceras med projektets gång.

### **Hur stödjer RUP ett företag till att produkten skapas i rätt tid, budget, med rätt utvecklingsteknik och att produkten blir vad kunden vill ha?**

- **6 best practice** eller stjärna som visar grunderna i RUP.
- Genom att tidigt se resultat och se om ett projekt kommer att ta länge tid än tänkt. Kunna säga det till kunden så att inte förtroendet minskas av en massa oväntade förseningar. Här ger det istället förtroende att veta snabbt ungefär vad för tidsplan som gäller.
- **Riskhantering och målhantering**

### **Hur stödjer RUP ett företag till att ta hänsyn till de externa faktorerna?**

- Externa faktorer inte bara tekniska tas omhand av **riskhantering**, ex semester, sjukdom.

### **Hur stödjer RUP ett företag när oväntade problem uppstår?**

- **Den iterativa utvecklingen** stödjer riskhantering och omprioritering.

### **Kan vi trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? Om nej, vad är det som hindrar oss?**

- Ja, genom RUP skall utvecklarna kunna få fram vad intressenterna tycker med hjälp av use-case, prototyper och genom att hela tiden driva kraven (The requirement workflow).

### **Vilka attribut anser ni vara viktigast i RUP?**

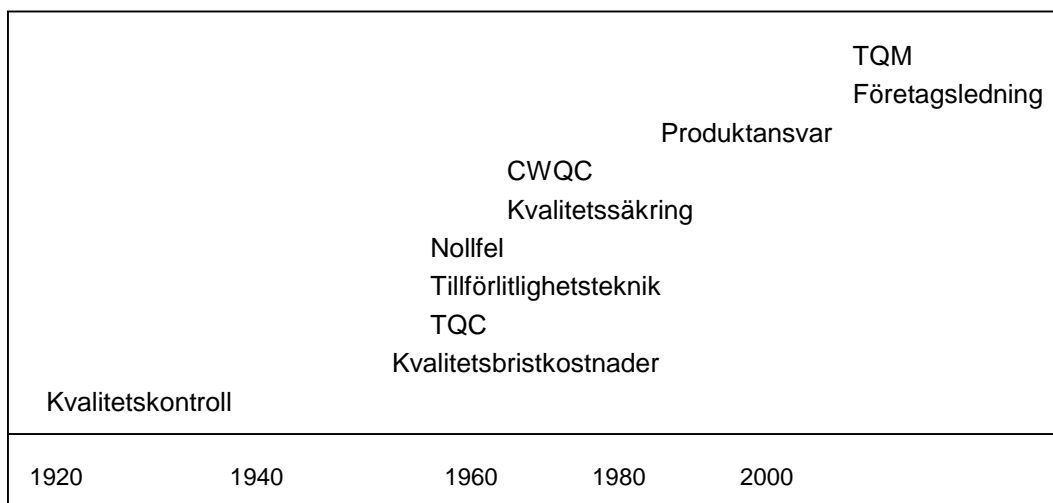
- **Trivsel**, genom att den kämpar för att bryta ner undvika barriärer.
- **Robust**, för framtida förändringar
- **Iterativ**

### **Vilka attribut anser ni vara viktigast i den produkt som RUP skapar?**

- **Pålitlig**
- **Lättunderhållet**
- **Lätttestat**
- **Överblickbart**

## C. Historia bakom kvalitet

I figur C.1 visas utvecklingen inom kvalitetsområdet:



Figur C.1: Utveckling inom kvalitetsområdet

**Innan industrialismen** gjorde sitt intåg bodde människor i små byar där alla kände alla. Så när en kund ville ha något beskrev kunden det för säljaren och de visste för det mesta hur de skulle göra för att leva upp till förväntningar och ge en vara av rätt kvalitet. Om vi ser tillbaka är nog detta den mest utvecklade formen av kundstyrd utveckling.

Men redan på denna tiden fanns det vissa hantverkare kunderna tyckte var bättre än andra. De var bättre på att lyssna och hade ett större kunnande. Dessa kan sägas ha haft högre kvalitet än andra.

**Industrialismen** medförde att företag började specialisera sig, och hantverkarna försvann och istället kom specialister som var ett led emellan kunden och fabriken, detta medförde att överblickbarheten minskade.

Under mitten av 1800-talet utvecklades produkttekniken så att människor lärde sig att utveckla identiska komponenter, detta medförde att intresset för olika mätinstrument ökade. Instrument för att **inspektera** produkterna tog över det som tidigare skett med visuell granskning.

1920 blev Watter A Shewart, anställd vid Bells telelaboratorier i USA, den förste att ge kvalitetsarbete ett vetenskapligt innehåll. Han ansåg att variabilitet var ett naturligt inslag vid produktion, alltså kunde den analyseras med hjälp av statistiska metoder. Alla produkter behövde alltså inte undersökas utan det räckte att göra stickprov. Detta medförde att det på företag bildades avdelningar för **kvalitetskontroll**. (Nordkvist, 1996)

Efter andra världskriget slut vidgades synen på kvalitetsbegreppet (Nordkvist, 1996):

- **Kvalitetsbristkostnader** myntades av Joseph Juran, har haft stor betydelse för kvalitetsutvecklingen, och kostnaderna kunde delas upp i två delar, oundvikliga kostnader som t.ex. kostnad för provtagning av kvalitet och "Guldet i Gruvan" som t.ex. spill.
- Armand Fengbaum introducerade 1956 **TQC**, total Quality Control, där han betonar att kvalitet är allas ansvar, och att kvalitet måste finnas mer än i tillverkningen.
- **Tillförlitlighetsteknik** infann sig när produkt drift och tillförlitlighet blev intressant, och avancerade sannolikhetskalkyler började användas.
- Försvarsindustrierna ville ha produkter t.ex. missiler som inte hade ett enda fel, därför blev **nollfel** en term, och här var det viktigt att se till alla bitar.

I slutet av 1960 blev begreppet "quality assurance" eller **kvalitetssäkring** intressant. Kvalitetsstyrning och kvalitetssäkring brukar det sättas ett likhetstecken mellan, men alla är inte överens om detta då kvalitetsstyrning vill styra kvaliteten till en önska nivå, medan kvalitetssäkring innebär en granskning av arbete med produkter och tjänstekvalitet. De som anser att kvalitetsstyrning och kvalitetssäkring är samma sak gör detta påstående eftersom de anser att de har samma mål. (Andresen, 1994, Sandholm, 1995)

Japan som tidigt intresserat sig av kvalitet blev under 60-talet ledande inom området och skapade **CWQC**, Companywide Quality Control, som sammanfattar japanska synen på kvalitet (Nordkvist, 1996):

- Alla funktioner inom ett företag måste involveras. Liknar mycket Feigenbaums TQC.
- Medarbetare på alla nivåer deltar aktivt i kvalitetsarbetet.
- Kvalitetsarbetet bygger på Kaizen, en filosofi om ständiga förbättringar.
- Kvalitet skall definieras utifrån kundens behov.

Under 1970-talet talades det mycket om vem som var ansvarig när en skada uppstod, och **produktansvar** blev ett hjälpmedel för att förebygga skador.

På 1980-talet ökade intresset för kvalitet hos **företagsledningen**, de hade insett att det kunde utnyttjas som konkurrensfördel.

En nationell satsning utfördes i Sverige och den ledde fram till att SIQ bildades, som har till uppgift att bidra till modernt kvalitetstänkande.

(Sandholm, 1995)

På 80-talet skapades en Amerikansk- Europeisk variant av japanernas CWQC den fick namnet **TQM**, total Quality Management. Den kan beskrivas så här (Nordkvist, 1996):

*"TQM är ett sätt att driva ett företag eller någon annan organisation så att resurserna på ett systematiskt och disciplinerat sätt fokuseras på att ständigt förbättra kvaliteten och allt det som görs. Det är inte ett företagsledningsverktyg som skall läggas till alla andra, utan ett övergripande sätt att leda företag. Det är inte universalmedicin mot alla problem, men de kvalitetsnivåer som åstadkoms vid TQM håller på att bli en förutsättning för överlevnad."*

Kvalitets framväxt har skapats ur industrialismen och massproduktionen, men idag har dock kvalitet en annan innebörd än då. Vid industrialismens intåg var det medelmänniskans kvalitet. Idag har en återgång till hantverket skett och nu vill de åt just er och se vilka behov just ni har, och mätmetoder har gjort det möjligt och globalt och konkurrens har tvingat fram det.

## D. Intervjuerna

D1. Via telefon den 23/8 kl.15-16

**1. Ålder:** Under 50

**3. Titel:** Universitets adjunkt (Rektor Skövde)

**4. Vilka områden bör en IT-strategi beröra? (Kundrelationer, organisationsform, Humanresurser, utvecklingsstruktur)**

Tycker att IT-strategi ofta förknippas med teknik, och att det är farligt då det kan göra att vissa människor inte vill ha med strategin att göra. Ett bättre namn på strategin är IS/IT strategin, eftersom strategin även bör beröra den miljö som informationssystemen finns i. Det viktiga i en sådan strategi är:

- Filosofin om att sprida och dela information (informationsresursen).
- Att sambandet mellan verksamhet och IT förstås
- Metodfrågor som t ex standardprodukter, egenutveckling och integration mellan system.
- Tekniska aspekter
- Ansvar

**5. Vad har ert företag för IT-strategi?**

Har börjat arbeta på en strategi som bygger på frågan *hur informationsteknologin skall användas?*

**6. På vilket sätt främjar/hindrar strategin kommunikationen och samarbete mellan: kunden och systemutvecklare? /mellan olika systemutvecklare? /mellan projektledare och systemutvecklare?**

Om en IT-strategi är tekniskt fokuserad har en kund svårt att känna sig delaktig i den. Det är väldigt viktigt att strategi är förankrad åt båda hållen.

**7. Hur främjar / hämmar den formella struktur (roller, hierarki vs nätverk) kommunikationen och samarbete?**

För att den formella strukturen skall kunna främja kommunikation och samarbete är det viktigt att ha en struktur som har klara ansvarsförhållanden, vilket innebär att det finns fasta roller, av rapporteringspunkter och att ansvar och befogenheter går hand i hand.

**8. Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? (Vad är det som hindrar er?)**

Det går att åstadkomma bra system men det beror av om systemutvecklaren ställer rätt frågor. Det gäller att systemutvecklaren gräver rätt och talar med rätt människor.

Man kan se utvecklingsarbetet som en kunskapsprocess både för kunder och systemutvecklare och därför är det viktigt att låta kraven mogna fram. För att kunna klara detta krävs diverse tekniska hjälpmedel.

**9. Vilken slags information kommuniceras vanligtvis mellan de aktörer som är involverade i systemutvecklingen?**

Informationen som kommuniceras är dels om den interna strukturen av systemet och dels om systemmiljön.

**10. Vilka slags kompetenser använder ni i systemutveckling?**

Vilken typ av kompetens beror av projektet.

Bred och allsidig kompetens

Verksamhetskompetens

Teknisk kompetens

Generalister

Social kompetens

Kunna argumentera och uttrycka sig i ord och skrift

Systemutvecklare bör inte vara krokodiler, stora i käften och med små öron.

**11. Har organisationen en tydlig modell för att stödja systemutvecklingsprocessen? Är du nöjd med modellen? Vad vill du ändra på?**

Har ingen modell.

**15. Hur hanteras informationen från olika projekt? Hur förs informationen vidare?**

Systemutvecklare är dåliga på att föra kunskap vidare, vilket betyder att vi ofta kastar kunskapen över axel. Vi har dålig kunskap om knowledge management och hur vi får kunskap att stanna inom företaget.

**18. Vilka attribut anser ni vara viktigast i en process (Välj ut de 5 viktigaste attributen)? (Komplettera gärna med fler attribut)**

Synlig

Varaktig

Acceptabel

Verkningsgrad

Tydlighet i användarroll

**19. Vilka attribut anser ni vara viktigast i den produkt som processen skapar (5 viktigaste attributen)? (Komplettera gärna med fler attribut)**

Användbarhet

Begriplighet

Integrerbarhet

Funktionalitet

Låg koppling

**D2. Intervju den 16/8 kl. 13-14**

(OBS! första intervjun och frågorna ändrades efter detta test)

**1. Ålder:** Under 50

**2. Kön:** Kvinna

**3. Titel:** Konsult på Enator infosystem Väst AB

**4a. Har ditt företag en tydlig systemutvecklingsprocess?**

Nej, den är varken definierad eller känd (fortsätt till fråga 5b)

**5b. Om nej, vad vill du ändra på? (se punkterna ovan och beskriv de som behöver förbättras, får även beskriva andra idéer)**

Processen bör vara tydlig förankrad och känd. Den skall vara beskriven på ett sådant sätt att alla vet vilket vad som man skall göra. Den skall vara utformad så att det är lätt att hitta i den (tydlig), t ex skall det finnas en klar struktur som är lätt att klicka sig igenom på sitt sökande efter hjälp.

Det är viktigt att man i en process vet vad den börjar och när den slutar (In kommer detta och ut kommer detta) och att det på ett tydligt sätt finns beskrivet att det är det här målet som gäller och hur målen skall nås.

Den skall kunna hjälpa till när oväntade problem uppstår t ex förändring i krav, det skall då finnas tekniker för hur detta tas om hand.

Processen skall hjälpa människorna se hur man ligger till och på så sätt se när man är på väg åt fel håll.

Det är viktigt att processen gör klart vad som krävs av människorna och att vems ansvaret är.

Punkter nedan visar vad Ann-charlotte tycker att en process bör stödja:

- val av rätt teknik och metoder (denna punkt är viktig)
- fördelning av arbete efter den kompetens som finns (kan vara bra i vissa fall)
- till att få människor att arbeta bra tillsammans (Bör finnas tekniker/metoder som beskriver hur man bör ta tillvara på detta)
- till att skapa rätt relation till kunder och andra intressenter (Bör finnas tekniker/metoder som beskriver hur man bör ta tillvara på detta)
- till att produkten skapas i rätt tid, budget, med rätt utvecklingsteknik och att produkten blir vad kunden vill ha (detta stödet bör processen ge)
- till att ta hänsyn till de externa faktorerna (riskhantering)
- när oväntade problem uppstår (bör finnas tekniker för detta)

På Enator infosystem Väst AB tyckte hon att det skulle vara bra att ha en process att luta sig tillbaka på när problem uppstår. PPS skulle vara övergripande och en process skulle stödja den.

## **6. Hur viktig är processen för dig i ditt dagliga arbete?**

Viktig

## **7a. Kan vi trots kundens oförmåga att precisera sina krav åstadkomma ett bra system?**

Ja, detta kan ske genom att hela tiden försöka ha en bra kommunikation med kunderna och att inse hur viktigt det är med kravhantering. Kraven skall inte definieras och sedan lämnas, utan hela tiden omvärderas.

## **8. Vad föredrar du:**

Att modellen som används väljs genom en kompromiss vid varje projekt, med den risken att man aldrig får en chans att lära sig den riktigt bra.

## **9. Vilka attribut anser ni vara viktigast i en process (Välj ut de 5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Synlig

Pålitlig

Varaktig

Effektiv

Anpassningsbarhet

## **10. Vilka attribut anser ni vara viktigast i den produkt som processen skapar (5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Användbarhet



Korrekthet  
Accepterbarhet  
Ekonomisk  
Synlighet

### **11. Hur bör information hanteras från olika projekt? (Hur används den? Hur förs kunskapen vidare?)**

Det skall finnas en kortfattad slutrapport där det framgår klart och tydligt om projektet blev lyckat/ eller mindre lyckat, en kortfattad beskrivning av hela projektet och där skall även framgå kostnad och tidsrammar som är viktiga och svåra faktorer.

Varje slutrapport skall ha samma upplägg och vara åtkomlig på ett bekvämt sätt.

### **12. Vad har Enator för systemutvecklingsutvecklingsstrategi?**

Har vi någon sådan ???

### **13a. Beskriv ett typiskt utvecklingsprojekt? (Vilka kompetenser använder ni?)**

Projektledare, ett antal systemutvecklare, testare, och Användarinformatörer (gör användarmanual. Dessa är beroende på projektets storlek samma eller olika personer.

### **13b. Hur samordnar Enator delarna? (hierarki eller nätverk)**

VD → Gruppchefer → och ”resten”

Det är en platt organisation.

I varje projekt finns det projektledare och sedan ett antal kompetenser som samlas för att de skall vara till nytta i projektet

### **14. Vad bör göras för att kommunikationen i en verksamhet skall fungera?**

Försöker genom att ha en plattorganisation ha högt i tak. Det leder till en öppen diskussion och att fler säger vad de tycker.

Nackdelen med detta är att det är svårt att komma någon vart när det inte finns någon som håller i diskussionen, alla säger var de tycker men ingen leder det framåt.

Kanske skulle behöva lite mer styrning.

### **15. Hur kan en verksamhet göra för att skapa samverkan? (Vilja att dela med sig av sin kunskap)**

Målen som finns måste gälla hela gruppen och inte individen... framförallt när det gäller pengar. Om det finns individuella mål med pengar som belöning blir det ofta en konflikt mot samverkan, eftersom om jag delar med mig till dig kan du dra nytta av det.

Kanske om det fanns en möjlighet till en kompromiss däremellan genom att individerna får belöning om de delar med sig av sin information...t ex utbildar någon.

### **16. Hur länge har ni arbetat i företaget?**

2 ½ år

Tror till viss del att attraktiviteten i ett företag beror av hur länge en person stannar i företaget. Risken med att arbeta i konsultföretag och ute hos kund är att man tillslut får mer samhörighet med kunden än med sitt företag. Det är viktigt att ha aktiviteter och att man känner att det är vi (företaget).

### D3. Intervju den 26/8 kl. 8-9

1. **Ålder:** 50+

3. **Titel:** Produktchef (Banteknik)

4. **Vilka områden bör en IT-strategi beröra? (Kundrelationer, organisationsform, Humanresurser, utvecklingsstruktur)**

IT-strategin bör beröra hårdvara, mjukvara och nätverk. All IT-strategi bör ske med kundnytta eller verksamhetsnytta i fokus.

5. **Vad har ert företag för IT-strategi?**

De har en strategi som berör områdena mjukvara, hårdvara och nätverk, dock finns det ett problem och det är att IT-strategin ofta blir snabbt omodern. På Spårvägen har de inte tid att ständigt uppdatera den.

6. **På vilket sätt främjar/hindrar strategin kommunikationen och samarbete inom verksamheten?**

Problem som ofta uppstår då det är en spretig verksamhet och det finns en övergripande IT-strategi är att delar vill gå olika fort. Detta slår ofta hårdare mot de små delarna som inte har lika mycket att säga till om.

6. **Hur främjar / hämmar den formella strukturer (roller, hierarki vs nätverk) kommunikationen och samarbete?**

Den formella strukturen skall inte behöva vara något problem om det finns hjälpmedel som stöd för kommunikation och samarbete. Dessa hjälpmedel skall hjälpa människor att få samma bakgrundsmaterial och skapa åtkomlighet.

7. **Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? (Vad är det som hindrar er?)**

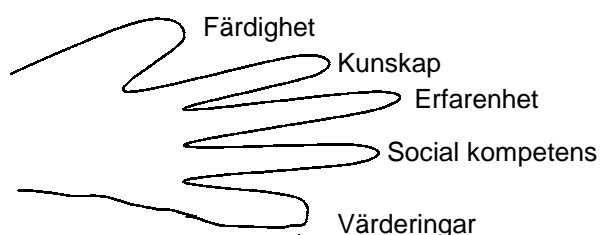
Spårvägen är ju en kund och för dem är det viktigt att nå en gemensam förståelse mellan dem och konsulten... detta sker inte direkt utan mognar fram. Ett bra sätt är att genom prototyping öka förståelsen.

9. **Vilka slags kompetenser är viktiga vid val av leverantör?**

När införandet av Lotus Notes skulle ske så hade Spårvägen kontakt med ett antal leverantörer som fick presentera sina idéer, valet av Enator berodde på deras presentation.

Det fanns företag som presenterade guld och gröna skogar och hur fantastiskt Lotus var och vilka möjligheter det fanns i framtiden... men nuläget diskuterades det lite av och lyssnade gjorde de inte.

Enator valdes p.g.a. att de lyssnade och var lyhörda.



Figur E1: Handen

Ovan syns en hand som visar viktig kompetens hos en person. Om bara tummen och pekfinger fanns så har vi en person som bara har en begränsad kompetens. De kvarvarande tre fingrarna kan ofta vara viktigare, men det beror på vad det är för arbete.

#### **14. Hur hanteras informationen från olika projekt?**

De har ett antal informationsbaser som håller reda på informationen.

#### **Hur förs informationen vidare?**

Erfarenheter förs vidare genom nyckelpersonerna plus att man kan gå in och läsa om projekten.

Efter varje projekt sker ett uppföljningsmöte där projektet går igenom och avvikelser tas om hand och åtgärdas.

#### **15. Vilka egenskaper/delar anser ni som kritiska i en systemutvecklingsprocess?**

I systemutvecklingen är implementeringen det mest kritiska för Spårvägen, då de måste "sälja" sin produkt till andra i verksamheten. Om det finns mycket fel kvar kan det leda till mycket motstånd, som ibland känns onödigt då felen är banala och borde varit åtgärdade.

#### **16. Vilka egenskaper anser ni som kritiska i den produkt som systemutvecklingen skapar?**

Att produkten lever upp till förväntningarna och att det är en professionell lösning.

#### **18. Vilka attribut anser ni vara viktigast i den produkt som processen skapar (5 viktigaste attributen)? (Komplettera gärna med fler attribut)**

Användbarhet  
Effektivitet  
Begriplighet  
Pålitlighet  
Integrerbarhet

#### **D4. Intervju den 25/8 kl. 9-10.30**

**1. Ålder:** Under 50

**3. Titel:** Konsult och team leader för utvecklingsenheten

#### **4. Vilka områden bör en IT-strategi beröra? (Kundrelationer, organisationsform, Humanresurser, utvecklingsstruktur)**

Tycker att de områden som har tagits upp ovan är de viktiga. En IT-strategi bör beröra hela verksamheten, fast den kanske inte kan styra allt.

#### **5. Vad har ert företag för IT-strategi?**

Vi har en IT-strategi, som berör våra kompetenser, kundarbetet men det saknas strategier för hur tekniken ser ut.

Problemet när man är ett konsultföretag är att det är svårt att få strategin att genomsyras bland de anställda, eftersom man ofta sitter hos kund och då får rätta sig efter deras strategi.

#### **6. På vilket sätt främjar/hindrar strategin kommunikationen och samarbete mellan kunden och systemutvecklare? /mellan olika systemutvecklare? /mellan projektledare och systemutvecklare?**

Strategin skall föra människor i företaget närmare varandra, men det är svårt då Enator är ett företag i förändring och omorganisering.

Hur väl en strategi slår igenom beror av det personliga intresset.

**7. Hur främjar / hämmar den formella struktur (roller, hierarki vs nätverk) kommunikationen och samarbete? Är emot hierarkier, men när ett företag blir stor är det svårt att nå ordning utan en sådan.**

Enator Infosystem Väst AB har en platt organisation och detta är möjligt då företaget har ca.30 anställda. Den nya strukturen är uppdelad i två team med en VD ovanför.

En risk som kan finnas är att det bildas barriärer mellan de olika teamen. En fördel med team är att teamen är uppdelade efter människors intresse och att det ökar möjligheten till utbyte av kompetens och intresse inom gruppen.

**8. Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? (Vad är det som hindrar er?)**

Ibland. Det finns en stor risk att fel sak görs och att väntat resultat inte uppnås.

För att lyckas skapa ett bra system krävs att rätt frågor ställs och att kraven utreds. Det krävs också att systemutvecklaren har kunskapen att se effekterna av de krav som kunderna ställer.

**9. Vilken slags information kommuniceras vanligtvis mellan de aktörer som är involverade i systemutvecklingen?**

Informationen brukar handla om:

- Identifiera problemen som systemet skall lösa och hur det skall utföras.
- Att sedan ha avstämningar som berättar var man ligger tidsmässigt och vad man gör. Detta berör både kund och projektet i sig.
- Informationen förmedlas via möten, protokoll, e-post mm.

**10. Vilka slags kompetenser använder ni i systemutveckling?**

Vad för kompetens som behövs beror av vad för slags verksamhet man arbetar i.

Förutom de direkta kunskaperna för uppgiften bör alla kunna PPS, vilket är projektstyrning och det är viktigt att ha en social kompetens för att få en bra relation med kunden.

**11. Har organisationen en tydlig modell för att stödja systemutvecklingsprocessen? Är du nöjd med modellen? Vad vill du ändra på?**

Nej, och det beror på att företaget har arbetat med resurskonsulter, alltså att konsulter är ute hos kund och rättar sig efter deras modeller.

Nu, när företaget skall försöka ta hem mer uppdrag har de börjat titta på diverse modeller och metoder där ibland RUP. Det behövs en modell för att skapa gemensamma begreppsramar och kunna kontrollera hur projektet ligger till.

**12. Hur bör kommunikationen fungera i systemutvecklingsprocessen? Vad anser ni som störande och som ni gärna vill förbättra?**

Metoden bör bestämma kommunikationen och bör vara ett hjälpmedel för att säkra förståelse. Den bör dock inte vara för exakt, utan det bör finnas mallar som man tar till sig. Dock bör metoden vara flexibel och beroende av projektet storlek kunna förändras.

**13. Hur bör samarbete fungera i systemutvecklingsprocessen? Vad anser ni som störande? Vad anser ni att det saknas?**

Genom ett formaliserat sätt för att säkra flödet, men även ett icke formaliserat.

#### **14. Beskriv ett typiskt utvecklingsprojekt:**

**vilka faser genomgår projektet,**

I PPS finns det tre faser:

Förstudie

Genomförande

Avveckling (Leverans och godkännande)

**vilka är de vanligaste misstagen som begås,**

Att inte veta vad som skall göras innan projektet dras igång, alltså att kraven inte är klara.

**hur hanteras risker i projektet,**

Riskerna bedöms, rangordnas och det skapas en åtgärdsplan för de största riskerna. Denna åtgärdsplan går igenom med kund om riskerna kan påverka systemets budget och leveranstid.

**vad är projektets svagast punkt,**

Varierar, men en viktig punkt är okända faktorer (se vanligaste misstagen)

**vad är det viktigaste resurserna i ett utvecklingsprojekt?**

Människor. En väldigt viktig bit är att rätt människor i kundens verksamhet får delta, eftersom de vet hur verksamheten fungerar.

#### **15. Hur hanteras informationen från olika projekt?**

Hanteras inte på något bra sätt och detta beror oftast av att det inte finns tid för detta.

**Hur förs informationen vidare?**

Informationen förs vidare genom erfarenheter som människor har, vilket är väldigt sårbart, eftersom om en person slutar kan det föra med sig att ingen annan kan den personens delar. Inom IT-branschen är det ett rotation på människor, då högre löner lockar. För att hålla kvar människor och kompetensen försöker verksamheten med aktiviteten (vi-känsla) och genom att binda människor till företaget genom ex. optioner.

#### **18. Vilka attribut anser ni vara viktigast i en process (Välj ut de 5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Förstårlig

Synlig

Robust

Acceptabel

Tydlighet i användarroll

#### **19. Vilka attribut anser ni vara viktigast i den produkt som processen skapar (5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Användbarhet

Effektivitet

Accepterbarhet

Funktionalitet

Enkelhet

D5. Intervju den 30/8 kl. 14-15

**1. Ålder:** Under 50

**3. Titel:** VD

**4. Vilka områden bör en IT-strategi beröra? (Kundrelationer, organisationsform, Humanresurser, utvecklingsstruktur)**

De områden som beskrivs ovan tycker jag är av stor vikt för det slutliga resultatet.

- Strategin skall beröra det som sker internt såsom hur informationen skall nås i verksamheten.
- Kundrelationer är en viktig fråga, hur får kunderna information är en fråga som måste besvaras.
- Utvecklingsstrukturen är viktig både för de tekniska förutsättningarna och för hur processen bör stödja oss i vårt arbete.
- Organisationsdesignen är platt på Enator och behöver därför inte så stort fokus i IT-strategin.

**5. Vad har ert företag för IT-strategi?**

All information och underlag som är nödvändigt för en person skall vara tillgänglig var du än befinner dig.

**6. På vilket sätt främjar/hindrar strategin kommunikationen och samarbete mellan: kunden och utvecklare? mellan olika utvecklare? mellan projektledare och utvecklare?**

Strategin bör beröra den metod som används i systemutvecklingsprocessen och om denna brister kan de få konsekvenser för kommunikation och samarbete. Det är viktigt att metoden som valts stödjer en gemensam begreppsbas så att alla vet vad alla pratar om.

**7. Hur främjar / hämmar den formella struktur (roller, hierarki vs nätverk) kommunikationen och samarbete?**

I verksamheten varierar rollerna ganska mycket p g a att de arbetar i olika projekt... ena gången utvecklare nästa gång projektledare och eftersom strukturen varierar så mycket så främjar det samarbete.

Om man ser på koncernnivå så kan den strukturen hindra, då det är svårt att få en sådan strategi att vara så tydlig att den värnar om kompetenser och kommunikation mm. Det är svårt att hitta rätt kompetens för uppdrag inom koncernen, eftersom vi består av så många små bolag och enheter.

**8. Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? (Vad är det som hindrar er?)**

Ja, inom Enator Informationssystem Väst AB så finns det goda chanser för detta, då de har både verksamhetsutveckling och systemutveckling. Det är viktigt att verksamhet och systemutveckling arbetar ihop för att nå kundens mål och gemensam förståelse.

**9. Vilken slags information kommuniceras vanligtvis mellan de aktörer som är involverade i systemutvecklingen?**

Det som diskuteras är:

- Funktion (hur fungerar det?)
- Användarfall (Förstår vi varandra?)

- Teknik

Informationen kommuniceras dels med hjälp av PPS som håller i den formella kommunikationen och dels med hjälp av att de som arbetar i ett projekt får sitta i samma rum/område.

### **10. Vilka slags kompetenser använder ni i systemutveckling?**

En systemutvecklare bör kunna:

- Verktyg (Java, VB, Microsoftprodukter)
- Olika miljöer (Telecom, Fordon, Logistik)
- Teknisk projektledning (Viktigt i vissa fall)
- Diskutera användarmiljön (Ha en gemensam bas)

### **11. Har organisationen en tydlig modell för att stödja systemutvecklingsprocessen? Är du nöjd med modellen?**

Nej, har ingen bra modell men har upptäckt behovet av en och letar nu efter någon variant.

### **12. Hur bör kommunikationen fungera i systemutvecklingsprocessen? Vad anser ni som störande och som ni gärna vill förbättra?**

Det som är viktigt i den kommande modellen är:

- Tydlighet i stegen
- Roller, vad som skall göras och när
- Kunna ge snabba besked om var vi står och vart vi skall

### **13. Hur bör samarbete fungera i systemutvecklingsprocessen?**

Genom att ha en gemensam begreppsbas och entydig målbild.

### **14. Beskriv ett typiskt utvecklingsprojekt:**

#### **vilka faser genomgår projektet,**

Processkartläggning

Målbildsdefinition

Kravdefinition

Analys (I den ingår även processutveckling)

Design (Utveckling genom användarfall och pilotprojekt)

Applikationsutveckling

Test

Implementering

Underhåll (Om kund önskar)

#### **vilka är de vanligaste misstagen som begås,**

Det vanligaste misstaget är att inte definitionsarbetet blir riktigt och att inte alla funktionskrav fångas in, det kan få konsekvensen att man måste gå tillbaka i ett långt gånget projekt.

#### **hur hanteras risker i projektet,**

Risker tas omhand i PPS genom risk analys, riskseminarier och åtgärdsplan.

#### **vad är projektets svagast punkt,**

Tidsplaneringen

#### **vad är det viktigaste resurserna i ett utvecklingsprojekt?**

Projektledningen

**15. Hur hanteras informationen från olika projekt? Hur förs informationen vidare?**

Pilotprojekten som gjorts i projekt sparas och kan visas för kunder som exempel, naturligtvis bara grundstommen. Det finns även en projektplan som beskriver projektet kort, dock är teknik svagt dokumenterad. Tekniken finns istället som kunskap som förs vidare via personer som varit delaktiga i projektet.

**16. Vilka egenskaper/delar anser ni som kritiska i en systemutvecklingsprocess?**

Designfasen (när funktioner och teknik skall fungera ihop) och tidsplanen.

**18. Vilka attribut anser ni vara viktigast i en process (Välj ut de 5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Förstårlig

Pålitlig

Robust

Snabbhet

Effektiv

**19. Vilka attribut anser ni vara viktigast i den produkt som processen skapar (5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Användbarhet

Pålitlig

Accepterbarhet

Ekonomisk

Tidlöshet

**D6. Intervju den 25/8 kl. 13-14.30**

**Ålder:** Under 50

**Titel:** LQC Line Quality Coach. Hon arbetar med kvalitet inom tre områden, linje kvalitet, SQA i projekt och processarbete.

**4. Vilka områden bör en IT-strategi beröra? (Kundrelationer, organisationsform, Humanresurser, utvecklingsstruktur)**

**Utvecklingsmiljön bör ingå.** Ett exempel, just nu har de genomgått en strategiförändring från att ha använt Framemaker till att de använder Word.

**6. På vilket sätt främjar/hindrar strategin kommunikationen och samarbete mellan:**

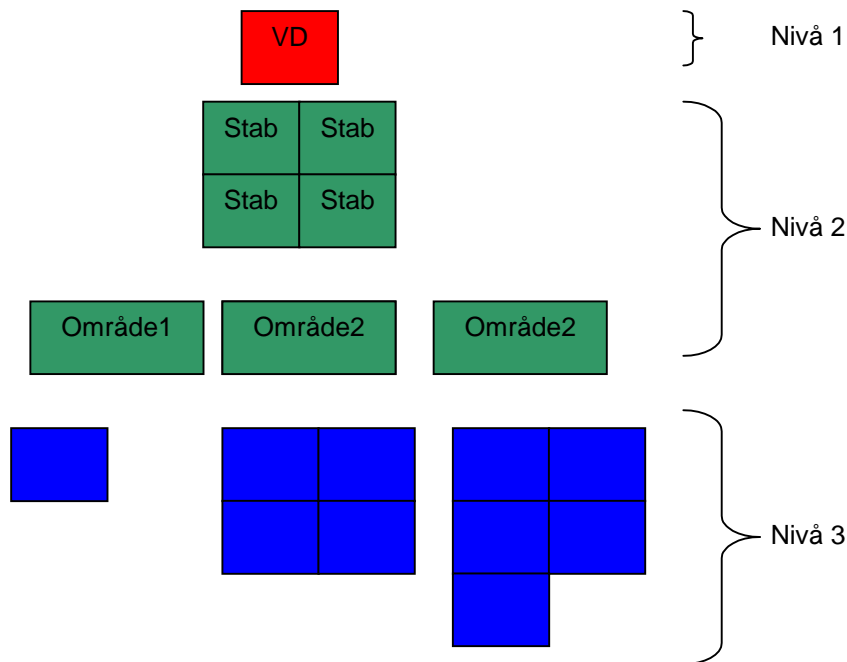
**- kunden och systemutvecklare? /mellan olika systemutvecklare? /mellan projektledare och systemutvecklare?**

Den bör främja kommunikation och samarbete, vilket den gör på Ericsson genom t ex att det finns strategier för vad för program som skall användas och strategier för att alla skall kunna nå sin information vart de än är i världen.

**7. Hur främjar / hämmar den formella struktur (roller, hierarki vs nätverk) kommunikationen och samarbete?**



Strukturen ser ut som i figur E2 och i den arbetar ca. 350 människor varav 300 i Göteborg resten i Vänersborg.



**Figur E2:** Struktur

Avsikten med en struktur är att främja möjligheten till kommunikation och samarbete och det görs genom att det finns tydliga roller, och ansvarsområden i hierarkien, och att det är på ett åtkomligt ställe finns information om detta. Denna hierarki fungera bra hos oss.

**8. Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? (Vad är det som hindrar er?)**

Kunden har ofta luddiga krav, så vad som måste göras är att försöka skapa tydlighet genom att ställa rätt frågor och om det finns tvivel ställa dem om och om igen. Det är otroligt viktigt att kunden har rätt kunskap så att kund och utvecklare förstår varandra.

Dock kan oftast inte kraven slutspecificeras i början av ett projekt för kunden mognar och utvecklar sina krav.

För att lyckas nå bra system krävs det att man är öppen för förändringar och kan hantera dem på ett bra sätt.

**9. Vilken slags information kommuniceras vanligtvis mellan de aktörer som är involverade i systemutvecklingen?**

De pratar sinsemellan genom projektmöten och teammöten.

**10. Vilka slags kompetenser använder ni i systemutveckling?**

**Teknisk kompetens:**

- Kunskap om systemet du arbetar med
- Verktyg
- Social kompetens
- Vid arbete i grupper är det viktigt att detta finns

### När det gäller projektledare:

- Förutom teknisk kompetens och kompetens att driva projekt
- Tillgänglig
- Öppen
- Kunna känna av stämningen i projektet/gruppen.

### Kunnande om företaget sätt att arbeta:

- Processer/verktyg/metoder

### 11. Har organisationen en tydlig modell för att stödja systemutvecklingsprocessen?

Har en modell som heter SDPM, men den skall bytas ut till RUP. Detta för att den förra modellen inte uppfattas ge stöd för iterativ utveckling.

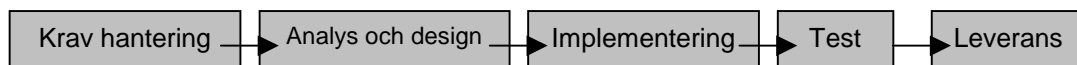
Har en projektstyrningsmodell som heter PROPS.

### 12. Hur bör kommunikationen/Samarbete fungerar i systemutvecklingsprocessen? Vad anser ni som störande och som ni gärna vill förbättra?

Alla skall förstå hur arbetet fungerar och detta görs genom att skapa tydliga roller där det beskrivs vad som förväntas av mig och vad jag har för ansvar. Allt måste vara beskrivet på en ganska detaljerad nivå för att alla skall kunna förstå varandra. Detta är viktigt för att även en nyanställd skall komma in snabbt i arbetet genom att kunna läsa om sin roll och förstå hur arbetet bör ske.

### 14. Beskriv ett typiskt utvecklingsprojekt:

vilka faser genomgår projektet,



Figur E3: Utvecklingsprocess

OBS! Den har möjlighet att gå tillbaka om nya krav uppstår.

vilka är de vanligaste misstagen som begås,

**Kravhantering:** kraven är otydliga och hanteringen av kraven fungerar inte.

hur hanteras risker i projektet,

- Risker hanteras genom en riskanalys som går till som följer:
- När målet för projektet är spikat brainstorma för att få fram alla risker
- Rangordning
- Ta hand om de riskerna som är för farliga
- Uppföljning hur det går med elimineringen
- Det är önskvärt att riskanalys görs flera gånger under ett projekt men detta sker sällan p g a tidsbrist.

vad är projektets svagast punkt,

Beror av projektet t ex kan det vara projektledarens kompetens eller resurser.

vad är det viktigaste resurserna i ett utvecklingsprojekt?

Människor och utvecklingsmiljön

### 15. Hur hanteras informationen från olika projekt? Hur förs informationen vidare?

Erfarenheterna samlas i en slutrapport som projektledaren gör efter en mall, den skall sedan alla i verksamheten läsa och lära sig av. I SQA ingår det att se till att alla roller tar del av slutrapporten i det efterföljande projektet.

#### **16. Vilka egenskaper/delar anser ni som kritiska i en systemutvecklingsprocess?**

Kraven förändras mycket.

#### **17. Vilka egenskaper anser ni som kritiska i den produkt som systemutvecklingen skapar?**

Vår egen kunskap om vad kunden skall ha produkten till för att kunna göra rätt prioriteringar i projektet och därigenom möta upp mot kundens avsikter med produkten? Vart är vi på väg? Insikt och förståelse för att förstå sammanhanget.

Att det är bra kvalitet dvs. motsvara kundens förväntningar, gärna överträffa dem.

#### **18. Vilka attribut anser ni vara viktigast i en process (Välj ut de 5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Pålitlighet  
Robust  
Snabbhet  
Effektiv  
Förstålig

#### **19. Vilka attribut anser ni vara viktigast i den produkt som processen skapar (5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Användbarhet  
Korrekthet  
Lätt underhållet  
Tillgänglighet  
Robust

#### **D7. Intervju den 24/8 kl.10-11.15**

**1. Ålder:** Under 50

**3. Titel:** Verksamhetskonsult (Enator infosystem Väst AB)

#### **4. Vilka områden bör en IT-strategi beröra? (Kundrelationer, organisationsform, Humanresurser, utvecklingsstruktur)**

En IT-strategi bör lägga fokus på frågorna:

Hur bör IT användas för att underlätta för verksamheten (t ex vad för funktion och roll skall IT ha)?

Vart vill man nå med verktyget IT?

#### **5. Vad har ert företag för IT-strategi?**

Har vi någon? Koncernen kanske håller på att skapa en, men frågan är om den ger något för oss, eftersom för att en strategi skall vara lönt måste den vara förankrad. Den måste med andra ord vara nedbruten till en viss nivå inom koncernen.

#### **6. På vilket sätt främjar/hindrar strategin kommunikationen och samarbete mellan: kunden och systemutvecklare?**

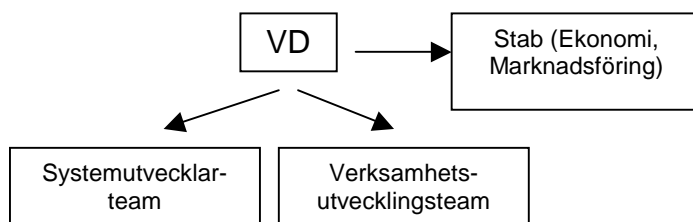
En strategi får inte vara begränsad så att den hindrar relationer mellan kunder och systemutvecklare. IT-strategin kan specificera val av teknik mm. och detta kan främja kundrelationen då kunskapen inom de områdena blir stor.

### **mellan olika systemutvecklare?/ mellan projektledare och systemutvecklare?**

Förknippar strategi med ett högre plan, men tycker att för att kunna förankra den måste den finnas på lägre plan.

### **7. Hur främjar / hämmar den formella struktur (roller, hierarki vs nätverk) kommunikationen och samarbete?**

Inom Enator Infosystem väst AB finns det en väldigt liten formell struktur, vilket möjliggörs genom att antalet anställda är ca. 30 st. Strukturen har gått från ett antal grupper till figur E4.



**Figur E4:** Formell struktur

Systemutvecklarteamet är teknikdrivet, medan verksamhetsutvecklingsteamet arbetar med förbättring av verksamheter och ser teknik som ett hjälpmedel.

Vad som hindrar samarbete och kommunikation är att mätning sker genom beläggning per konsult, alltså på individnivå. Detta leder till att konsulterna hellre lägger timmar på kunden, än på att utbilda sina arbetskamrater, då det senare ej syns i mätningen.

Mätningen bör istället ske på teamnivå, eftersom det skulle uppmuntra till samarbete.

### **8. Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? (Vad är det som hindrar er?)**

Ja, detta kan ske genom en inkrementell utveckling som ger en ständig uppdatering av krav.

### **9. Vilken slags information kommuniceras vanligtvis mellan de aktörer som är involverade i systemutvecklingen?**

Teknisk information, om databaser och modeller mm.

Use-case scenarior (till kund)

### **10. Vilka slags kompetenser bör man använda i systemutveckling?**

- Bred analytisk kompetens
- Förståelse för slutanvändarna (IT = verktyg ej medel)
- Behärska verktygen
- En god lyssnare

### **11. Har organisationen en tydlig modell för att stödja systemutvecklingsprocessen?**

Nej!

### **Är du nöjd med modellen? Vad vill du ändra på?**

Har en projekt styrningsmetod, men den kontrollerar bara projektet. (PPS)

## 12. Hur bör kommunikationen fungera i systemutvecklingsprocessen? Vad anser ni som störande och som ni gärna vill förbättra?

Genom att ha en process som har ett antal väldefinierade avstämningpunkter, dock bör processen vara flexibel, så att det undviks avstämningar som sker bara för att man måste... men som egentligen ej är nödvändiga. Avstämningar bör ske i team, med kunder och i projekt mm.

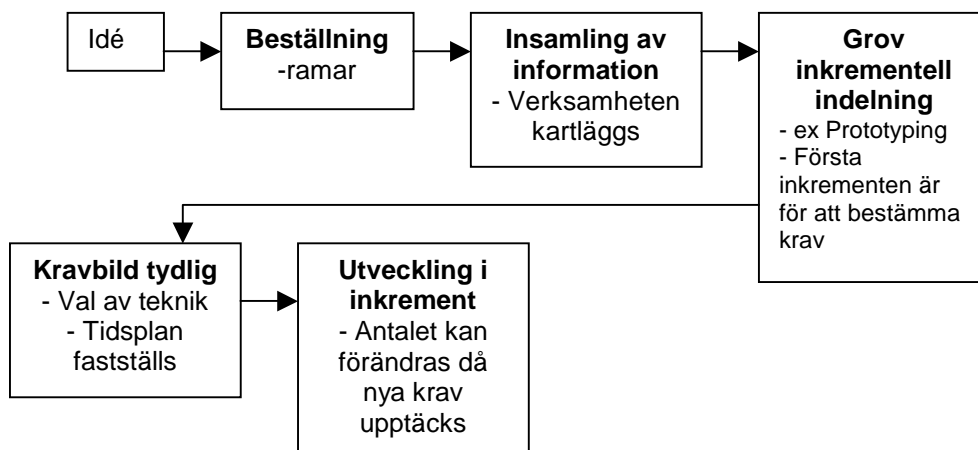
## 13. Hur bör samarbete fungera i systemutvecklingsprocessen? Vad anser ni som störande? Vad anser ni att det saknas?

Det bör finnas ett stöd för vad för slags kompetens som behövs, vad som krävs av kompetensen mm.

## 14. Beskriv ett typiskt utvecklingsprojekt:

**vilka faser genomgår projektet,**

Figur E5 visar ett utvecklingsprojekt. En sak som är viktig är att processen bör stödja förändring i tidsplanen.



Figur E5: Utvecklingsprojekt

**vilka är de vanligaste misstagen som begås,**

Att man tillåter kunden att ha för bråttom

**hur hanteras risker i projektet,**

Genom riskanalys, och i den tas risker fram och det beskrivs hur man bör reagera om de uppstår. Riskerna bör tas upp vid flera tillfällen i projektet, men när beror av projekt storleken.

**vad är projektets svagast punkt,**

Det är projektledarrens kompetens och beställarkompetensen

**vad är det viktigaste resurserna i ett utvecklingsprojekt?**

Alla resurser är lika viktiga, om det är någon resurs som är mindre viktig så är det fel sammansättning.

## 15. Hur hanteras informationen från olika projekt? Hur förs informationen vidare?

Informationen sparas i ett så kallat "projektkontor", där det läggs slutrapporter, dokument mm. Dock finns det en konflikt när man arbetar hos kund, eftersom det kan vara så att informationen från projektet är viktig för kunden och kunden inte vill att den skall föras vidare.

Ett problem är att ”sparkontoret” används alldeles för lite och att misstag begås om och om igen.

**16. Vilka egenskaper/delar anser ni som kritiska i en systemutvecklingsprocess?**

Förmåga att förstå kunden

**17. Vilka egenskaper anser ni som kritiska i den produkt som systemutvecklingen skapar?**

Hur produkten upplevs av slutanvändaren.

**18. Vilka attribut anser ni vara viktigast i en process (Välj ut de 5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Förstårlig  
Stödjande  
Snabbhet  
Effektiv  
Tydlighet i användarroll

**19. Vilka attribut anser ni vara viktigast i den produkt som processen skapar? (5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Användbarhet  
Begrifflighet  
Integrerbarhet  
Accepterbarhet  
Funktionalitet  
Enkelhet  
Synlighet

D8. Intervju den 23/8 kl. 9.30-11.15

**1. Ålder:** Under 50

**2. Kön:** Man

**3. Titel:** Teknikområdesansvarig

**4. Vilka områden bör en IT-strategi beröra? (Kundrelationer, organisationsform, Humanresurser, utvecklingsstruktur)**

IT-strategiskt är relationen med leverantören en viktig bit. Strategiskt har de alltid försökt få ner antalet leverantörer till två, men just nu är de fyra. Leverantörsantalet är viktigt då genom att ha få kunna ha en bra kommunikation och samarbete. Antalet är även viktigt för konkurrens, om det bara finns en leverantör känner de sig säkra och då höjs priset.

Produktstrategi är viktig då den lägger en bas för vad för språk som skall användas (t ex C++), vilken hårdvara mm.

**5. Vad har ert företag för IT-strategi?**

Leverantörer skall det finnas två av, och kraven på leveranserna skall specificeras hårt.

Egenutvecklingen skall öka för att öka säkerhet och hålla informationen inom företaget.

**6. På vilket sätt främjar/hindrar strategin kommunikationen och samarbete mellan: kunden och utvecklare?**

Eftersom leverantören mer är i fokus här bör frågan ställas om till leverantör och utvecklare. Just att ha få leverantörer kan försvåra då priset ofta höjs när de känner sig säkra.

### **mellan olika utvecklare?**

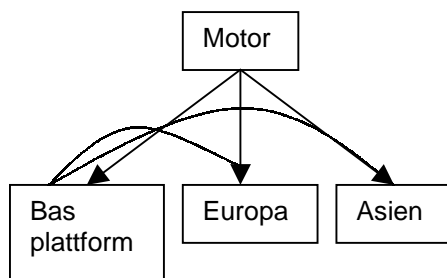
Inom verksamheten finns det flera system (Se fråga 7). Att ha en uppdelning skapar barriärer mellan de olika utvecklingsgrupperna, och problem som då skapas är bland annat att det i grupperna kan gå parallella likadana spår, vilket är kostsamt, och det kan vara svårt att föra över delar emellan systemen.

### **mellan projektledare och utvecklare?**

Eftersom verksamheten har människor som arbetar med olika leverantörer kan det leda till att projektledare bara får kunskap om ena systemet, vilket i sin tur leder till svårigheter om projektledaren skall ha ett projekt med det systemet som de inte har någon erfarenhet av. Om de hade haft ett system som de hade egenutvecklat hade detta problem inte funnits.

## **7. Hur främjar / hämmar den formella struktur (roller, hierarki vs nätverk) kommunikationen och samarbete?**

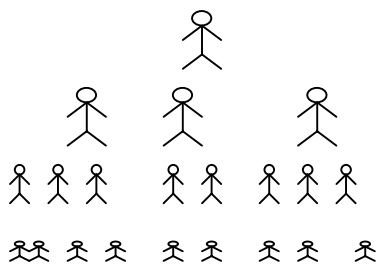
Volvo är ett stort företag med olika delar att se hela Volvo som en verksamhet, skulle inte bli rätt, att däremot se varje delverksamhet ger en bättre bild (figur E6):



**Figur E6:** Motor

Basplattformen stödjer de andra två delarna. Det är en ganska platt struktur.

Vid projekt som rör en produktion så sitter det en projektledare högst upp, under honom breder det beroende på projektets storlek ut ett stort nät av roller (Figur E7).



**Figur E7:** Rollhierarki

Det finns antal problem med att projektet delas upp över ett antal områden med begränsad kommunikation:

Synkroniseringsproblem

Målet blir inte tillräckligt tydligt

Stoppas upp hela utvecklingen när det blir förseningar i någon del, då delarna tillslut skall bli en helhet

## 8. Kan ni trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? (Vad är det som hindrar er?)

I denna verksamheten så är det produktplanering som kommer med en förfrågan på någon produkt/system. Problemet är att kraven ofta är så diffusa, och att när utvecklingen tar en längre tid så hinner kraven förändras. Om t ex kravet är att utveckla en bensinsnål produkt så kanske det gör att priset på produkten som kommer ut är högt... kraven förändras då en produkt med *så* högt pris inte säljer.

Det som hindrar oss från att åstadkomma ett bra system är framförallt att processen innan man kan testa produkten på kund är för lång.

Ett annat problem är att begränsningsfaktorerna oftast inte är specificerade innan projektet är igång, vilket gör att det tar lång tid innan man inser att detta inte är något att satsa på.

## 9. Vilken slags information kommuniceras vanligtvis mellan de aktörer som är involverade i systemutvecklingen?

Volvo har länge använt sig av ett **memo-system** innehåller e-post anslagstavla, databas att söka efter t ex protokoll.

Vid kommunikation med leverantör används ofta **e-post** och av säkerhetsskäl även **fax**, DHL används för att skicka vissa **dokument**, och **filer**.

Varje projekt skapar sin egen informationsstandard, t ex vi använder bara fax och efter den här strukturen.

## 10. Vilka slags kompetenser använder ni i systemutveckling?

En viss grundkunskap efterfrågas t ex objektorientering.

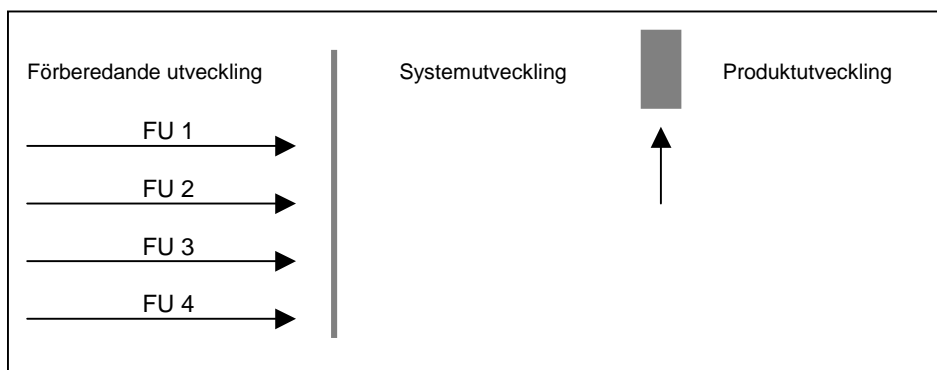
Att kunna arbeta i team och att kunna ge och lyssna och kompromissa.

Att vara strukturerad.

## 11. Har organisationen en tydlig modell för att stödja systemutvecklingsprocessen?

V++ är en modell med riktlinjer för att stödja programvaruutvecklingen, används inte speciellt mycket.

Där emot har de en annan modell som används se figur E8. In i den förberedande fasen kommer en idé som sedan delas upp på diverse delfaser som sker pararellt. När väl alla delfaser är färdiga sätts de ihop i systemutvecklingen, därefter tas det ett beslut om produkten är tillräckligt relevant för att ta i produktion (pilen i figuren), och om så är fallet sker en plan för produktionsutveckling.



Figur E8: Modell



Vad som finns specificerat i t ex förberedelse fasen är vad som skall komma in och vad som skall komma ut, kostnad, tidsplan, och prestanda.

### **Är du nöjd med modellen?**

Modellen fungerar men bara när det gäller modifiering av produkter. När det gäller nyproduktion så kör den fortfarande efter den tidsplan som gäller för modifiering... vilket inte fungerar då nyproduktion kräver längre tid för utveckling.

En annat misstag är att modellen inte tar hanterar erfarenheter för att på så sätt kunna och vidareutvecklar modellen.

### **12. Hur fungerar kommunikationen i systemutvecklingsprocessen?**

Det bildas lätt barriärer mellan de olika faserna i processen, de förberedande faserna arbetar för sig själva och när det lämnas över till systemutvecklingen så anser de att de inte kan använda det som FU har gjort.

Vad anser ni som störande och som ni gärna vill förbättra?

Ett sätt att komma tillrätta med problemet med barriärer är att involvera dem i varandras faser t ex FU i systemutvecklingen och tvärtom.

### **13. Hur fungerar samarbete i systemutvecklingsprocessen?**

Samarbetet fungerar inte så bra, men genom att involvera varandra i varandras faser kan man komma tillrätta med det här.

### **14. Beskriv ett typiskt utvecklingsprojekt:**

**vilka faser genomgår projektet,**

Se modell figur 3 ovan.

**vilka är de vanligaste misstagen som begås,**

Går vidare från en fas till nästa innan en fas är färdig. Detta görs för att hålla tidsplanen.

Tidsplanen är svår att uppskatta, och det kanske borde finnas en utredningsfas innan den förberedande utvecklingsfasen.

Vid utveckling av helt ny teknologi, så bör inte tidplanerna sträcka sig ut ända till produktionsinförande. Det är bättre att istället låta tidplanerna löpa till ett avstämningstillfälle med tydliga delmål. Fokuserar man för tidigt på produktionsinförande så blir det onödigt omständigt .

**hur hanteras risker i projektet,**

Riskerna tas upp på möten när de uppdagas och de tas även upp i inledningsskedet av ett projekt.

**vad är projektets svagast punkt,**

Att inte veta vad kunden har för krav.

**vad är det viktigaste resurserna i ett utvecklingsprojekt?**

Nyckelkompetenser och pengar.

### **16. Vilka egenskaper/delar anser ni som kritiska i en systemutvecklingsprocess?**

- Tid
- Kompetens
- Pengar

- Leverantörerna

**17. Vilka egenskaper anser ni som kritiska i den produkt som systemutvecklingen skapar?**

- Teknik
- Tid
- Pengar

**18. Vilka attribut anser ni vara viktigast i en process (Välj ut de 5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Synlig

Pålitlig

Varaktig

Snabbhet

Effektiv

**19. Vilka attribut anser ni vara viktigast i den produkt som processen skapar (5 viktigaste attributen)? (Kompletera gärna med fler attribut)**

Begriplighet

Återanvändbarhet

Funktionalitet

Ekonomisk

Robust

## E. Intervjuerna

### E.1 Binomen

Frågor:	Svar:	Kommentarer:
Vilken slags organisationer vänder sig RUP till?	Alla system- och mjukvaruutvecklingsföretag. Allt från konsulter till banker till försäkringsbolag.  RUP fungerar på projekt i storlek 5-25 personer. Även på mindre projekt men då kan antagligen bara bitar av RUP användas.	RUP:ens konfigurerbarhet är dess stora förtjänst, men det krävs också kunskap och erfarenhet för att kunna konfigurera RUP:en på ett optimalt sätt. Det svåra är att välja bort större delar av processen. I tveksamma fall tar den oerfarne alltid med mer än vad han/hon egentligen behöver.
Hur bör information hanteras från olika projekt enligt RUP? (Hur används den? Hur förs kunskapen vidare?)	Komponentbaserad utveckling används, eftersom den stödjer återanvändning av komponenter och arkitekturen. Komponenterna återanvänds mest i iterationerna inom projektet, vid en återanvändning i flera projekt tillkommer än större kostnad för test mm.  Worker (projekt manager) tar hand om information från iterationerna.  The enviroment workflow och process konfigurerbarhet stöttar en förbättring av processen.	Korrekt , men lite luddigt uttryckt
Hur stödjer RUP ett företag i val av rätt människor?	The Projekt management workflow beskriver vad för kompetens som måste finnas för projekt och om dessa inte finns vet man vad som behövs skaffas.  Finns än så länge ingen beskrivning hur team skall gå tillväga.  Genom "vem".	Detta låter bra men fungerar sällan i praktiken. Personer som är "bäst lämpade" för alla typer av uppgifter i processen är antingen "doers" som jobbar nära design och implementation, eller visionära "inventers" som föder ramverket till konstruktionerna, eller "managers" som håller i pengar och tidplaner. Några andra alternativ existerar egentligen inte. Men, okay, får man åtminstone ut en lista på "hur många man behöver av varje sort" så kan det eventuellt vara en kontroll på om man har anställt /allokerat rätt personal.
Hur stödjer RUP ett företag i fördelning av arbete efter den kompetens som finns?	Se frågan ovan.  Genom aktiviteter.	RUP stödjer i praktiken inte alls detta. Personkemi går inte att formalisera.
Hur stödjer RUP ett företag till att produkten skapas i rätt tid, budget, med rätt utvecklingsteknik och att produkten blir vad kunden vill ha?	Genom att tänka efter de grunderna som visas i stjärna.  Att tänka på det använda ett tillvägagångssätt som värnar om kvalitet på process och produkt (Test workflow).  Att hela tiden ha kontroll över	Ja, om processen är rätt konfigurerad från början, alternativt att man i praktiken tillåter sig att konfigurera om processen under resans gång. Kort sagt; RUP hjälper duktiga och erfarna människor att bli ännu skickligare, men den kan lika gärna hjälpa mindre skickliga och

	<p>projektet och se redan på ett tidigt stadium om förseningar kommer att inträffa.</p> <p>Riskhantering och målhantering.</p>	<p>oerfarna personer att gömma sig bakom en formell fasad.</p>
<p>Hur stödjer RUP ett företag till att ta hänsyn till de externa faktorerna?</p>	<p>Riskhantering som tar hand om semestrar mm som kan uppstå.</p> <p>Use-case ger förståelse utöver gränserna</p> <p>The business modeling workflow, skapar förståelse.</p>	<p>Stämmer, men riskhanteringen blir aldrig bättre än kunskapen hos dem som gör riskanalysen!</p>
<p>Hur stödjer RUP ett företag när oväntade problem uppstår?</p>	<p>Projekt management workflow.</p> <p>Den iterativa utvecklingen stödjer riskhantering och omprioritering.</p>	<p>Generellt sett så ger RUP bra överblick, vilket är bra i alla lägen, men den dag man använder RUP som ett verktyg för att hantera oväntade problem så bottnar det förmodligen i en osäkerhet hos projektledningen. Kort sagt: Projekt som förlitar sig alltför hårt på metoder och processer i krissituationer - kommer ofelbart att misslyckas!</p>
<p>Viktigaste attributen i RUP?</p>	<p>Trivsel, genom att den kämpar för att bryta ner undvika barriärer.</p> <p>Robust, för framtida förändringar.</p> <p>Iterativt.</p>	<p>Förståelig</p> <p>Synlig</p> <p>Stödjande</p> <p>Robust</p> <p>Snabbhet</p> <p>Anpassningsbarhet</p> <p>Tydlighet i användarroll</p>
<p>Vilka attribut anser ni vara viktigast i den produkt som RUP skapar?</p>	<p>Pålitlig.</p> <p>Lättunderhållet.</p> <p>Lätt testat.</p> <p>Synlighet.</p>	<p>Användbarhet</p> <p>Korrekthet</p> <p>Lättunderhållet</p> <p>Dokumentation</p> <p>Lätt att lära</p> <p>Snabb utvecklingsfart</p> <p>Funktionalitet</p>

## E.2 Frontec

Frågor:	Svar:	Kommentarer:
<p>Hur bör information hanteras från olika projekt enligt RUP? (Hur används den? Hur förs kunskapen vidare?)</p>	<p>Komponentbaserad utveckling används, eftersom den stödjer återanvändning av komponenter och arkitekturen. Komponenterna återanvänds mest i iterationerna inom projektet, vid en återanvändning i flera projekt tillkommer än större kostnad för test mm.</p>	<p>Finns inte definierat hur artefakter skall hanteras mellan projekt. Under ett projekt lagras alla artefakter enligt Config.&amp;Change Management workflow.</p> <p>Hur reuse skall gå till får man alltså hitta på själv.</p>

	<p>Worker (projekt manager) tar hand om information från iterationerna.</p> <p>The enviroment workflow och process konfigurerbarhet stöttar en förbättring av processen.</p>	
Hur stödjer RUP ett företag till att få människor att arbetar bra tillsammans?	<p>Språk, roller och genom att fokusera på att granskning ses som något positivt. Se en granskning som något man kan lära av och få helhet och still med.</p> <p>Inspektionsfasen, en värdering av vad för resurser som kan tänkas behövas.</p>	Inspektionsfasen ???
Kan vi trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? Om nej, vad är det som hindrar oss?	Ja, genom RUP skall utvecklarna kunna få fram vad intressenterna tycker med hjälp av use-case, prototyper och genom att hela tiden driva kraven (The requirement workflow).	OK. Även Business Modeling hjälper till
Viktigaste attributen i RUP?	<p>Trivsel, genom att den kämpar för att bryta ner undvika barriärer.</p> <p>Robust, för framtida förändringar.</p> <p>Iterativt.</p>	<p>Synlig</p> <p>Stödjande</p> <p>Pålitlig</p> <p>Robust</p> <p>Elegans</p> <p>Tydlig i användarroll</p> <p>Anpassningsbarhet</p>
Vilka attribut anser ni vara viktigast i den produkt som RUP skapar?	<p>Pålitlig.</p> <p>Lättunderhållet.</p> <p>Lätt testat.</p> <p>Synlighet.</p>	<p>Användbarhet</p> <p>Begriplighet</p> <p>Integrerbar</p> <p>Accepterbarhet</p> <p>Funktionalitet</p> <p>Underhåll</p> <p>Testbar</p>

### E.3 Skandia Liv

Frågor:	Svar:	Kommentarer:
Vilken slags organisationer vänder sig RUP till?	<p>Alla system- och mjukvaruutvecklingsföretag. Allt från konsulter till banker till försäkringsbolag.</p> <p>RUP fungerar på projekt i storlek 5-25 personer. Även på mindre projekt men då kan antagligen bara bitar av RUP användas.</p>	<p>Vi upplever att RUP kan bli för detaljerad och tung i små projekt. Vi funderar på att skapa en "RUP light" för dessa. I övrigt så är våra erfarenheter mycket goda. Vi har lagt ned mycket arbete på att anpassa processen till våra förhållanden.</p> <p>Vi har konkretiserat processen, anpassat den till vår arkitektur och skapat mallar och exempel. Vi tror</p>

		att detta är den enda vägen till att få alla på företaget att använda metoden. Jag tycker dock att det är viktigt att poängtera att det inte finns en metod som passar vid alla tillfällen. Man kan dock få den uppfattningen om man läser process-leverantörernas reklamblad.
<p>Vad gör RUP för att kommunikationen i ett företag skall fungera?</p> <p>Precisera gärna frågan, kommunikation i företag?</p> <p>Kommunikation mellan aktörer i systemutvecklingsprocessen?</p>	<p>RUP använder en objektorienterad designmodell som hjälper till med förståelse, modifiering och ger en gemensam begreppsbas. Språket som används är UML. (Används mest i design och analys)</p> <p>Processen är standardiserad, vilket gör att människor får en gemensam bas.</p> <p>Namnstandard gör att alla vet vad alla pratar om.</p> <p>Worker gör att alla vet vad som skall göras, vad ens ansvar är, och vilket resultat som skall presteras.</p> <p>Nya medarbetare kan lätt hoppa in, eftersom roller(worker) är väldefinierade.</p>	<p>Skulle just en objektorienterad designmodell främja kommunikation?</p> <p>I så fall varför? Det finns så många andra designmodeller...</p> <p>Den håller väl på att bli defacto, men det finns ju ett leverantörsberoende som man inte kan bortse från.</p>
<p>Hur gör RUP för att skapa samverkan?</p> <p>(Vilja att dela med sig av sin kunskap)</p>	<p>En process som alla i organisationen varit delaktiga i att skapa.</p> <p>Klar struktur om vem, vad, hur och när.</p> <p>RUP försöker bryta ner barriärer mellan olika utvecklingsgrupper (designer, analytiker), genom att använda use-case och genom att involvera grupperna tidigare.</p>	<p>På vilket sätt garanterar processen delaktighet?</p> <p>Klar struktur tycker vi med.</p> <p>På vilket sätt bryter användningsfall ned barriärer?</p>
<p>Hur bör information hanteras från olika projekt enligt RUP? (Hur används den? Hur förs kunskapen vidare?)</p>	<p>Komponentbaserad utveckling används, eftersom den stödjer återanvändning av komponenter och arkitekturen. Komponenterna återanvänds mest i iterationerna inom projektet, vid en återanvändning i flera projekt tillkommer än större kostnad för test mm.</p> <p>Worker (projekt manager) tar hand om information från iterationerna.</p> <p>The enviroment workflow och process konfigurerbarhet stöttar en förbättring av processen.</p>	<p>RUP är definitivt komponentbaserad och arkitekturdriven och detta upplever vi som det centrala. Just det att man genomför en riskanalys och baserar första iterationen på detta material, dvs. försöker hitta ett antal användningsfall som t.ex. är arkitekturdrivande, behandlar tekniska utmaningar och central funktionalitet utifrån verksamhetens perspektiv.</p> <p>Även om RUP är komponentbaserad, så säger den inget om hur man skriver återanvändbar kod. Återanvändning är ett eget område som kräver stora insatser för att lyckas. På Skandia har vi använt oss av komponentbaserad utveckling i ett antal år nu och vi återanvänder med</p>

		stor framgång komponenter, men det rör sig ofta om mer tekniska komponenter, t.ex kommunikation, säkerhet mm. Att återanvända affärslogik på högre nivå är mycket mer komplicerad, vår vision är att nå dit.
Hur stödjer RUP ett företag i val av rätt människor?	The Projekt management workflow beskriver vad för kompetens som måste finnas för projekt och om dessa inte finns vet man vad som behövs skaffas.  Finns än så länge ingen beskrivning hur team skall gå tillväga.  Genom "vem".	Är det verkligen processens ansvar, ligger inte det mer på projektstyrningsmodellen, t.ex. PROPS. Vad betyder kompetens, om någon gått kurs, så behöver ju det inte betyda att den personen behärskar området.
Hur stödjer RUP ett företag i fördelning av arbete efter den kompetens som finns?	Se frågan ovan.  Genom aktiviteter.	Samma som ovan.
Hur stödjer RUP ett företag till att få människor att arbeta bra tillsammans?	Språk, roller och genom att fokusera på att granskning ses som något positivt. Se en granskning som något man kan lära av och få helhet och still med.  Inspektionsfasen, en värdering av vad för resurser som kan tänkas behövas.	Det finns en direkt koppling mellan kvalitetsbegreppet och de många informella granskningarna i RUP, men att gå så långt som att säga att metoden främjar människor att arbeta bra tillsammans vågar jag inte säga något om.
Hur stödjer RUP ett företag till att skapa rätt relation till kunder och andra intressenter?	The business modeling workflow, genom att försäkra sig om att kunder, utvecklare och användare har en gemensam förståelse av dynamiken i en organisation.  Workshop (idéinsamling)  Use-case gör att intressenterna och utvecklarna förstår varandra  Den iterativa utvecklingen gör att resultat kan ses tidigt och få intressenterna intresserade.	Yepp!  Workshop är lysande, enda förbehållet vi har är att det skall ledas av en erfaren modelleringsledare för att bli riktigt bra.
Hur stödjer RUP ett företag till att produkten skapas i rätt tid, budget, med rätt utvecklingsteknik och att produkten blir vad kunden vill ha?	Genom att tänka efter de grunderna som visas i stjärna.  Att tänka på det använda ett tillvägagångssätt som värnar om kvalitet på process och produkt (Test workflow).  Att hela tiden ha kontroll över projektet och se redan på ett tidigt stadium om förseningar kommer att inträffa.  Riskhantering och målhantering.	Genom att tidigt definiera en iterationsplan (ovan) och köra en iteration och därefter svara på frågor som hur mycket resurser behöver vi, vilken tid kommer det här att ta, är det överhuvudtaget genomförbart.
Hur stödjer RUP ett företag till att ta hänsyn till de externa faktorerna?	Riskhantering som tar hand om semestrar mm som kan uppstå.  Use-case ger förståelse utöver	?

	gränserna The business modeling workflow, skapar förståelse.	
Kan vi trots kundens oförmåga att precisera sina krav åstadkomma ett bra system? Om nej, vad är det som hindrar oss?	Ja, genom RUP skall utvecklarna kunna få fram vad intressenterna tycker med hjälp av use-case, prototyper och genom att hela tiden driva kraven (The requirement workflow).	Ja, men det krävs ändå ett engagemang/erfarenhet från användare, finns inte det så räcker användningsfallen inte långt.
Viktigaste attributen i RUP? (Se * nedan för alternativ)	Trivsel, genom att den kämpar för att bryta ner undvika barriärer. Robust, för framtida förändringar. Iterativt.	Trivsel? Synlig Stödande Pålitlig Snabbhet Accepterbarhet Tydlighet i användarroll Anpassningsbarhet
Vilka attribut anser ni vara viktigast i den produkt som RUP skapar? (Se figur ** nedan för alternativ)	Pålitlig. Lättunderhållet. Lätt testat. Synlighet.	Underhåll/förvaltning tack vare komponent tänket och robust designmodell.  Denna fråga kan jag inte svara på då förklaringen till utfallet lika väl kan bero på något annat som på RUP.



## F. RUP

### F1. Varför RUP?

Samhället blir allt mer beroende av system och dessa blir mer och mer komplexa och större, samtidigt kräver utvecklarna att utvecklingen går snabbare och ändå har en hög kvalitet.

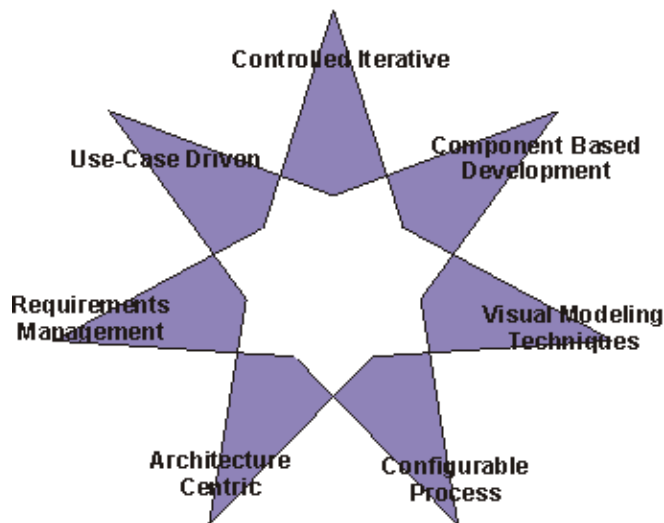
Det finns ett antal symtom som är vanliga vid misslyckade utvecklingsprojekt och dessa är bland annat missförstånd mellan kund och utvecklare, oförmåga att ta hand om förändrade krav, moduler som inte passar, mjukvara som är svår att underhålla, fel upptäcks sent i utveckling, dålig mjukvarukvalitet och team medlemmar som inte vet vad de andra gör. Dessa olika symtom kan härledas till ett antal rotorsaker som t ex:

- Ad hoc kravhantering.
- Dålig kommunikation.
- Hög komplexitet.
- Oppptäckt inkonsistens i krav, design och implementering.
- Otillräcklig testning.

RUP har som mål att producera hög kvalitativ mjukvara som (Krutcher, 1998):

- Möter kundens krav...
- ...inom budget...
- och i tid.

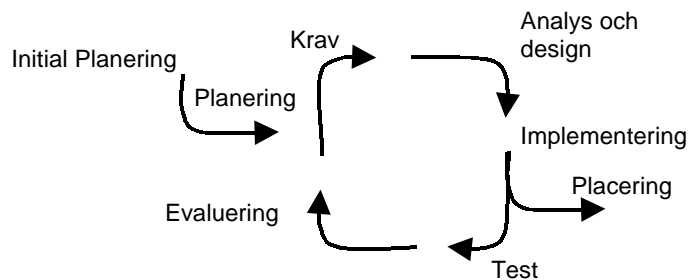
RUP vill nå målet och komma tillrätta med rotorsakerna genom att arbeta efter figur F1.



Figur F1: Grunder i RUP (CD RUP)

Nedan kommer en genomgång av det grundläggande i RUP:

- **Kontrollerad iterativ utveckling**



**Figur F2:** Iterativ och inkrementell process

Många företag använder sig idag av vattenfallsmodellen och stegar sig igenom allt från krav till analys och design, dock har det skett en förändring och fler och fler lämnar detta sätt och går mot ett mer iterativt tillvägagångssätt. Barry Boehm's spiral är ett känt exempel. (figur F2) Orsaken till att folk lämnar vattenfallsmodellen är att de inte anser sig vara säkra på att kraven kommit fram i tid och att genom ett iterativt tillvägagångssätt kan flaskhalsar undvikas.

RUP använder sig av en kontrollerad iterativ utveckling, vilket innebär att varje iteration planeras noggrant i antal, längd och mål.

- **Komponentbaserad utveckling**

Alla intressenter vid ett utvecklingsprojekt vill ge sin bild av hur systemet skall fungera. Systemets arkitektur är en viktig del för att hantera dessa olika bilder, då den kontrollerar de olika bilderna som skapas i den iterativa- och inkrementella utvecklingen. (appendix A) Komponentbaserad utveckling används, eftersom den stödjer återanvändning av komponenter och arkitektur, och den ger en klar uppdelning för de som arbetar med komponenterna.

- **Visuell modelleringsteknik**

Idag använder sig många av objektorienterade språk för att uppnå återanvändning mm, och när så är fallet är det viktigt att även använda sig av objektorienterad teknik i design. RUP använder en objektorienterad designmodell som hjälper till med förståelse, modifiering och ger en gemensam begreppsbas. Språket som används är UML.

- **Process konfigurerbarhet**

Processer måste ständigt förbättras och kompletteras, och i RUP brukar kompletteringarna oftast röra artefakter, arbetare, aktiviteter och arbetsflöden.

- **Arkitektur**

Arkitekturen i ett mjukvarusystem är en struktur av systemets komponenter som interagerar genom ett gränssnitt med komponenter som i sin tur interagerar med mindre komponenter genom ett gränssnitt och så vidare.

I RUP beskrivs arkitekturen i *Software Architecture Document*. I detta dokument visas arkitekturen i vyer, och varje vy visar något av vikt för intressenterna i processen.

Vyerna fångar de största strukturella designbesluten genom att visa komponenterna och hur de sammanfogas till en användbar form, och vyn ställs sedan mot de krav som finns.

- **Kravhantering**

Ett krav innebär ett dynamiskt tillstånd eller kapacitet som ett system behöver. RUPs hantering av krav sker genom att systematiskt hitta, dokumentera och organisera förändrade krav på systemet. För att spåra krav anser det att följande behövs:

- Problemanalys måste finnas för att hitta problemen bakom problemen.
- Förståelse av intressenternas intressen genom t.ex. rollspel.
- Definition av systemet, vilket innebär en meningsfull beskrivning av det tänkta systemet.
- Hantering av omfattningen av projektet.
- Ständig förbättring av systemdefinitionen.
- Hantering av förändring i krav.

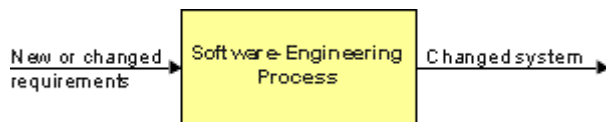
- **Användarfalldriven (Use-Case driven) utveckling**

Användarfall är en sekvens av händelser som ett system utför, vilket ger ett observerbart resultat för någon aktör. Den visar de möjliga sätt på vilket ett system kan användas.

Användarfall finns med i alla utvecklingsstegen i RUP, och används dels som en bas för planering av den iterativa utvecklingen, och formar dels grunden för vad som beskrivs i en användarmanual.

## F2. Processöversikt

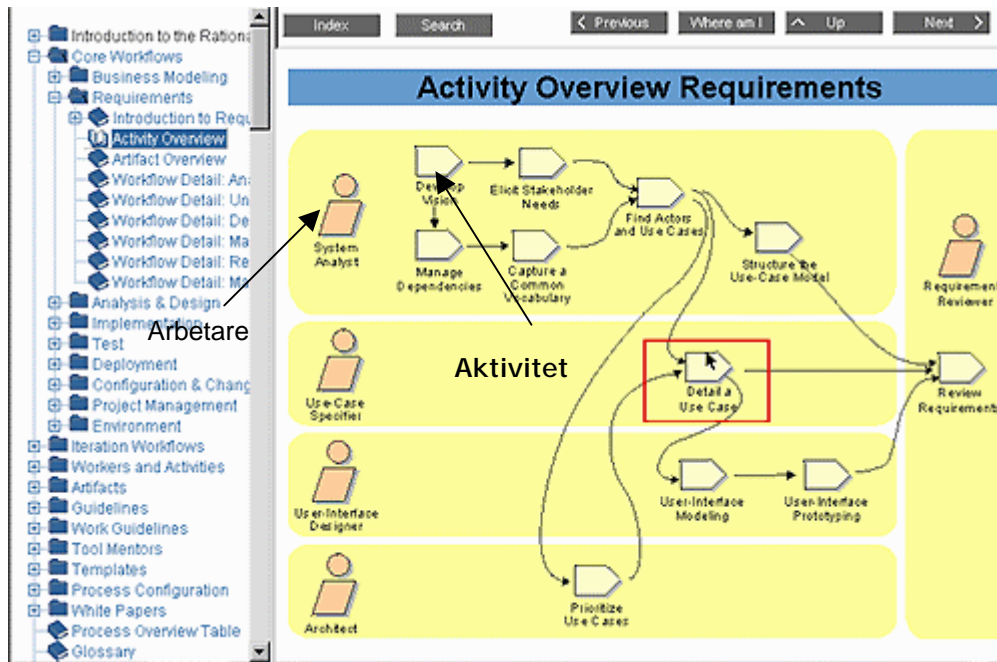
En process är delvis ordnade steg som har som avsikt att nå ett visst mål (figur F3).



**Figur F3:** Mjukvaruprocess (CD RUP)

Det finns två dimensioner, de statiska processer som är arbetsflöden, och de dynamiska som är tid vilket visas i iterationer och milstolpar. (avsnitt F4)

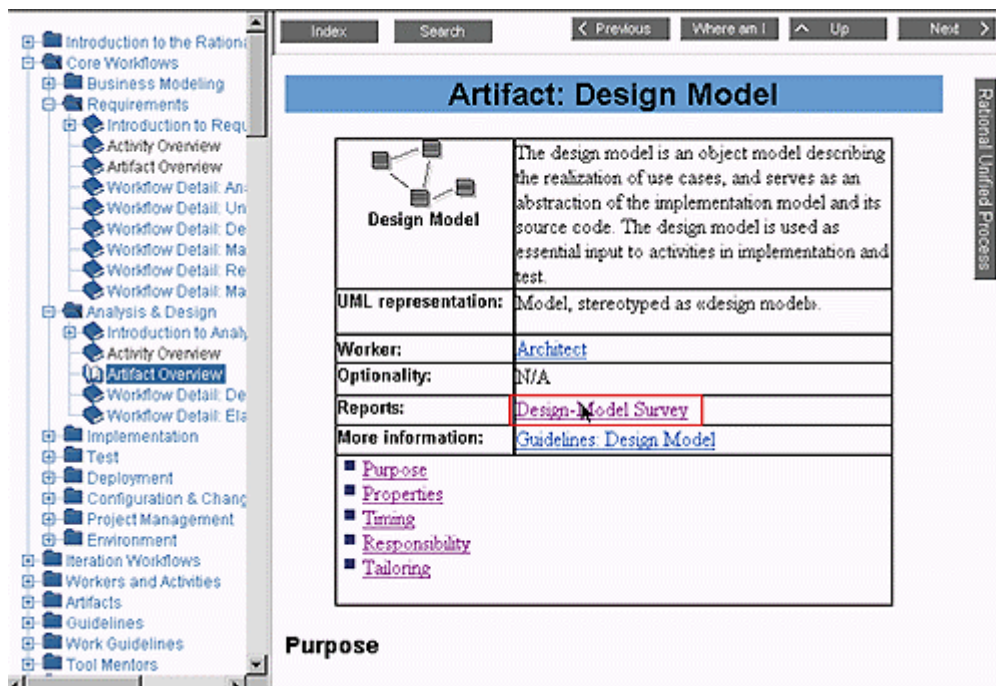
Processen beskrivs i vem som gör vad, hur och när.



**Figur F4:** Visar aktiviteter och arbetare (CD RUP)

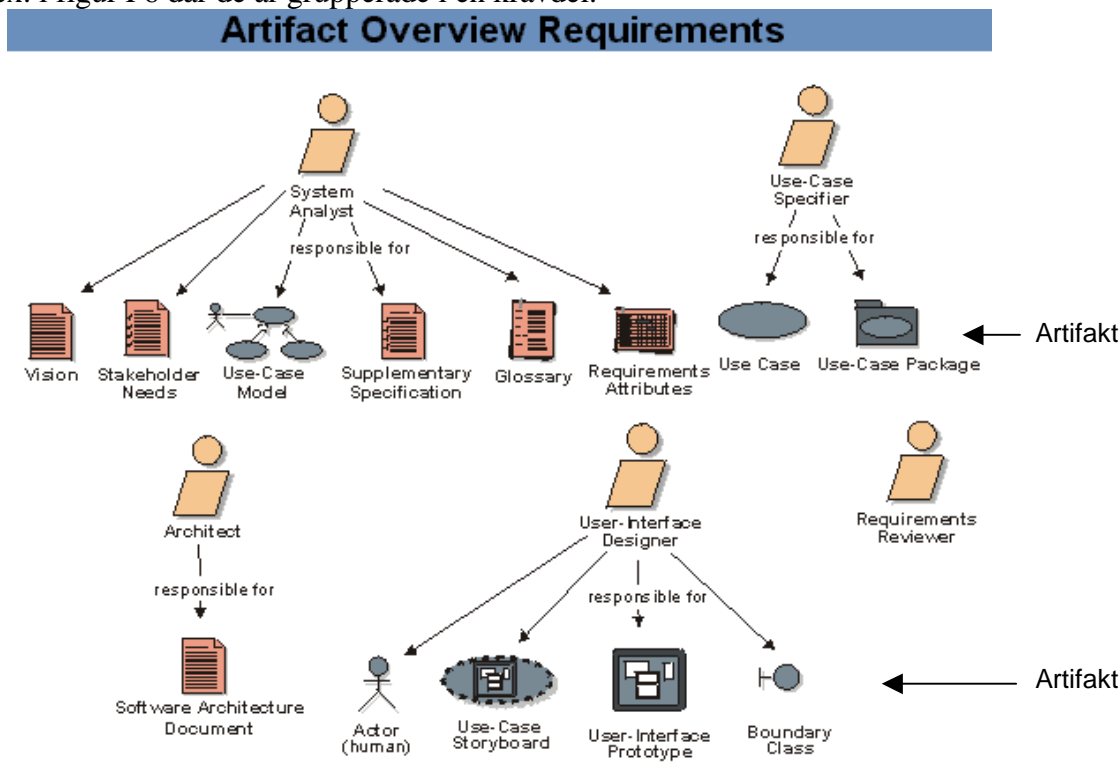
**Vem** kallas för "worker" och det kan ses som en hatt en individ har under projektet och en individ kan ha flera hattar. (Figur F4) Ett exempel på en arbetare är *designer* som definierar ansvar, händelser, attribut, relationer av en eller flera klasser och bestämmer hur dessa bör anpassas till implementeringsmiljön.

**Hur** kallas för "activities" och har klara mål. (Figur F4) Varje aktivitet tillhör en specifik "worker", och storleken på den är mellan några timmar upp till ett antal dagar. Aktiviteter kanske upprepas flera gånger under utvecklingens gång och måste då ha samma worker men ej samma individ. Ett exempel på en aktivitet är planera en iteration. Här är guideline (beskriver bland annat hur artefakter utvecklas) och toolmentor (ger en detaljerad beskrivning om hur Rational software tools kan ge support i vissa aktiviteter (CD RUP)) viktiga ingredienser.



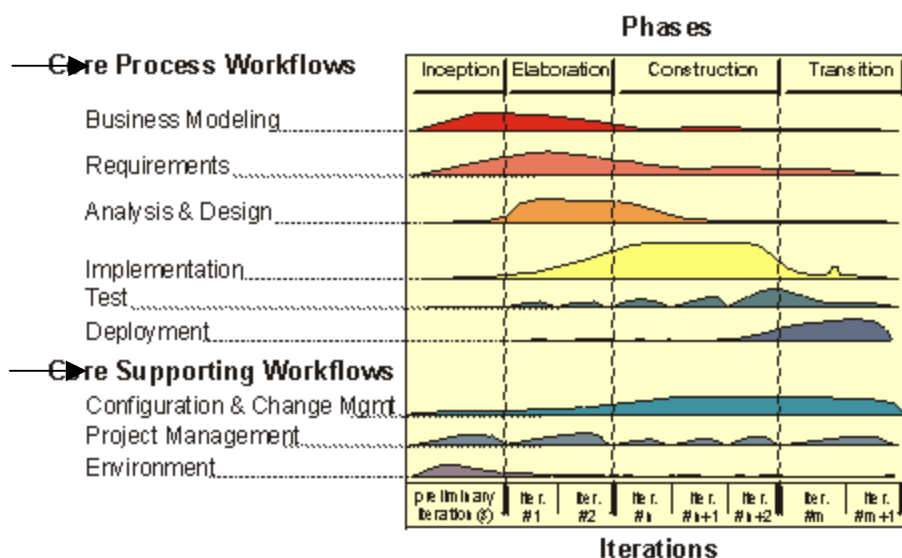
Figur F5: Artfakt (CD RUP)

Vad kallas för "artifact" (figur F5), och är information som produceras, modifieras eller används av processen t.ex. modell, kod eller dokument. Artfakterna kan grupperas som t.ex. i figur F6 där de är grupperade i en kravdel.



Figur F6: Artfakter som skall komma ut ur kravdelen (CD RUP)

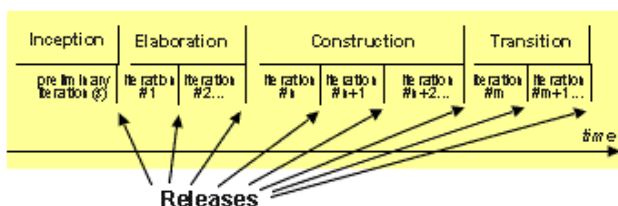
När kallas för "workflows" eller arbetsflöden, och innebär en sekvens av aktiviteter som producerar ett resultat av den observerbara världen t.ex. aktivitetsdiagram. RUP använder sig av statiska "core workflow", som det finns sex stycken för utveckling och tre för support. (Figur F7)



Figur F7: Arbetsflöde (CD RUP)

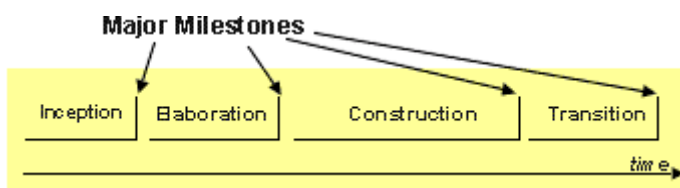
### F3. Faser

Livscykelmodellen som här består av fyra faser (figur F8), har brutits ner i små livscyklar, och efter varje sker en evaluering och en förnyelse av krav. Detta kallas ett iterativt tillvägagångssätt, där varje iteration är en hel utvecklingsprocess som slutar med att en delprodukt introduceras. Iterativ utveckling beskrivs ytterligare i avsnitt F2 och även i appendix A.



Figur F8: Faserna och iterationer (CD RUP)

Ur ett managementperspektiv är det viktigt att ha kontroll och det nås genom milstolpar (figur F9). Vid dessa ställs frågorna skall man fortsätta, sluta, eller byta kurs.



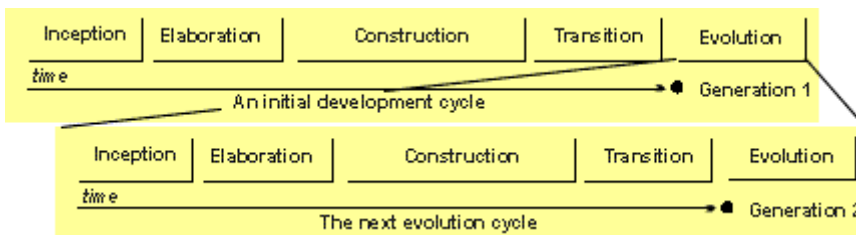
Figur F9: Milstolparna (CD RUP)

De fyra faserna är:

- **Inspektionsfasen**  
Här identifieras alla intressenter, det sker en identifiering av mål, riskanalys, succé krav, användarfallen identifieras och beskrivs, det skapas en plan som visar de viktigaste milstolparna och tiden och slutligen sker en värdering av vad för resurser som kan tänkas behövas. **Fokus ligger på mål och övergripande krav.**
- **Ellaborationsfasen**

Analys av problem, utveckling av en projektplan och eliminera högrisk faktorer och arkitekturella beslut testas. Detta anses vara den mest kritiska av de fyra faserna. **Fokus ligger på krav.**

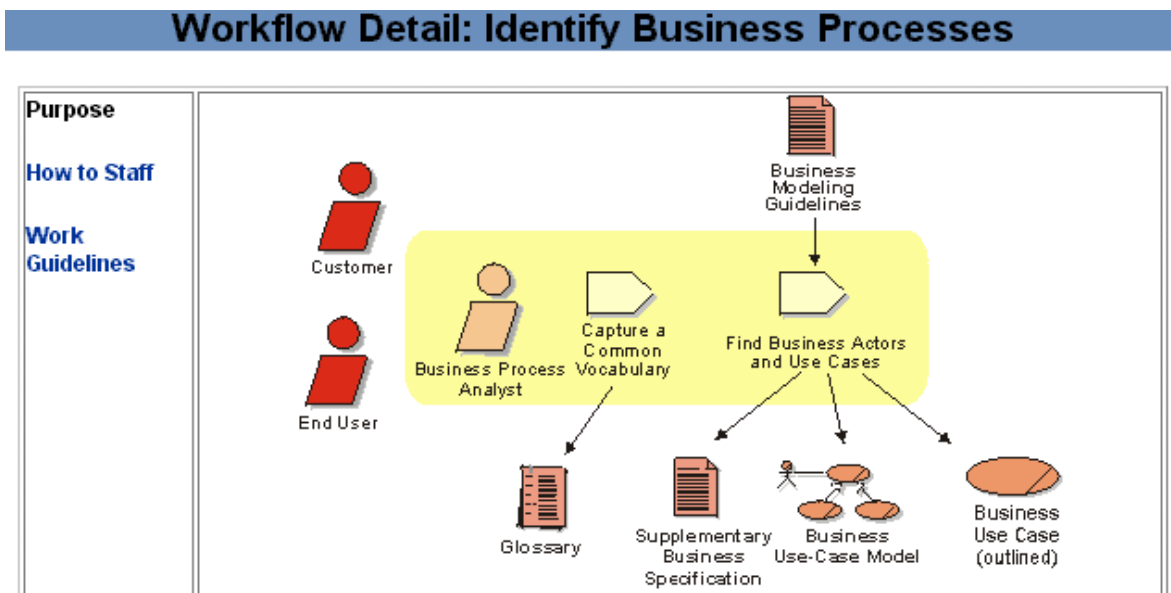
- **Konstruktionsfasen**  
Utveckling av produkt och beslut om produkt är redo för introduktion. **Fokus på design och implementering.**
- **Transformationsfasen**  
Under denna fas införs produkt i företaget, och ofta måste produkten efter implementering vidare utvecklas efter krav från kunden. I slutet tas ett beslut om en ny utvecklingscykel skall startas eller om produkten är tillräckligt bra (figur F10). Detta kallas evolutionär utveckling (appendix A). **Fokus på kvalitetssäkring av produkt.**



**Figur F10:** Utveckling (CD RUP)

#### F4. Huvudarbetsflöden

Nedan kommer en beskrivning av de nio huvudarbetsflödena göras, och de är alltså ”när” i processen och är statiska. (Figur F7) De representerar en uppdelning av arbetare och aktiviteter i logiska grupper. I figur F11 visas ett exempel på ett arbetsflöde för Business modeling workflow.



**Figur F11:** Arbetsflöde (CD RUP)

#### The business modeling workflow

Den finns för att försäkra sig om att kunder, utvecklare och användare har en gemensam förståelse av dynamiken i en organisation. Genom modellering av företaget skapas dels

gemensam förståelse och dels får man fram de systemkrav som behövs för att stödja organisationen.

### **The requirements workflow**

Den är till för att komma överens om vad systemet bör göra, ge systemutvecklaren bättre förståelse av kraven på system, definiera funktionaliteten av systemet och definiera ett användargränssnitt för systemet. Det är en bas för planering av det tekniskt innehållet i iterationer och en bas för utvärdering av kostnad och tid för utveckling av systemet.

### **The analysis and design workflow**

Målet är att transformera kraven till en specifikation som beskriver hur implementeringen skall äga rum. Här används usecase (användarfall) för att identifiera objekt, klasser, subsystem och paket.

### **The Implementation workflow**

Målet är att förfina organisationen av kod till implementeringsbara klasser och objekt i termer av komponenter, sedan försöka testa och utveckla komponenter som enheter och slutligen integrera det i ett exekuterbart system.

### **The Test workflow**

Detta arbetsflöde finns för att verifiera interaktionen mellan objekt och komponenter, verifiera integrationen mellan alla komponenter av mjukvara, verifiera att alla krav har blivit korrekt implementerade och slutligen identifiera och garantera att alla fel som upptäcks adresseras.

Denna del har mycket med kvalitet att göra och här anses det att både process- och produktkvalitet bör mätas för att nå kvalitet. I RUP är kvalitet allas ansvar och testning är ett sätt att uppnå kvalitet. (Se avsnitt F6)

### **The Deployment workflow**

Den finns för att producera en hel extern release av mjukvaran, och det görs genom att det finns planer för paketering, distribuering och installering av mjukvaran. Här finns även hjälp för användarna eller säljstyrkan.

### **Projekt management workflow**

Målet är att vara ett ramverk för att hantera mjukvaruintensiva projekt, att praktiskt guida planering, anställda, utförande och kontrollerande av projekt. Det skall vara ett ramverk för hantering av risker, och det sker genom att aktivt ta ansvar för risker och beslut om vad som skall göras när en risk uppstår.

Här sker bland annat planering för den iterativa processen och detta görs på två plan. Det ena planen är en tidlina med datum för stora milstolpar och de mindre milstolpar med personal profiler och datum, och den andra skapar en noggrannare genomgång av varje iteration.

### **The Configuration and Change Management workflow**

Det finns för att hitta och bevara integriteten av projektets mål under tiden utveckling sker och se till att projektet håller spåret. Den beskriver hur kontroll av artefakter görs av de inblandade.

### **The Environment workflow**

Målet är att ge rätt support för utveckling av organisation med hjälp av verktyg, metoder, och processer. Den ger support för val av verktyg, verktygsanpassning till organisation, processkonfiguration, processförbättring, träning och teknisk service.



## F6. Kvalitet och RUP

*"The characteristic of having demonstrated the achievement of producing a product which meets or exceeds agreed upon requirements, as measured by agreed upon measures and criteria, and is produced by an agreed upon process." (CD-RUP)*

Vad definitionen säger är att kvalitet är produkter som uppnår krav, som mäts med överenskomna kriterier och som är producerade efter en överenskommen process. Fokus i mjukvaruproduktion ligger på produkt- och processkvalitet.

Produktkvalitet innebär den slutprodukt som tillverkas och alla element som finns runt omkring t.ex. komponenter, arkitektur och subsystem. Processkvalitet innebär att försöka nå en produkt på ett lämpligt sätt.

Produktkrav finns bland annat i användarfall och för att uppnå produktkvalitet måste kraven på produkt vara klara, konsistenta och verifierbara (testbara). Processkvalitet är allas ansvar, men t.ex. kan projektmanager påverka processkvaliteten mer än vissa andra "arbetare".

Processkvalitet mäts på två sätt:

- Graden av fasthållandet av den standard, guidelines och implementation av en accepterad process.
- Kvalitet på artefakterna som produceras.

RUP stödjer företag som försöker uppnå CMM nivå 2 och nivå 3 (Capability Maturity Model). CMM har som mål att hjälpa företag att nå mål som har att göra med kostnad, funktionalitet och produktkvalitet.