

# **Återkoppling till ett system**

– Vilka konsekvenser kan det bidra till?

Magisteruppsats, 10 p, Informatik  
*Framlagd:* juni 2006  
*Författare:* Danijela Lazic  
*Handledare:* Linda Nordström

## **Förord**

Jag vill först och främst tacka min handledare Linda Nordström som lyckats handleda mig på distans. Vidare vill jag uttrycka tacksamhet åt Reuters Financial Software som gett mig chansen till en praktisk erfarenhet i deras organisation. Jag vill även tacka Grace K Musiitwa för stödet och viljan att bolla idéer med mig. Även Agneta Olerup får ett stort tack för sina tips till litteratur.

# Återkoppling till ett system

– Vilka konsekvenser kan det bidra till?

© Danijela Lazic

Magisteruppsats framlagd juni, 2006.

Omfång: 50 sidor.

Handledare: Linda Nordström

## Resumé

Behoven i världen kan ständigt förändras. Dessa behov kan även styra systemutvecklingen. Studiens syfte är att genom litteraturstudier samt en fallstudie komma fram till metoder genom vilka behoven blir faktorer för framtida systemutveckling. Ett färdigt system kan bli implementerat trots att det innehåller fel. Användare samt utvecklare blir därefter inblandade i underhållet samt vidareutvecklingen. Utvecklare kan heller inte helt och hållet ägna sig åt användares behov då det trots allt är utvecklarna som besitter kunskapen samt erfarenheten inom sitt yrke samt ämnet systemutveckling. Därför blir det för studien även intressant att veta hur man som utvecklare får veta vad som ska vidareutvecklas. Inför den empiriska fallstudien låg litteraturstudier som grund för en djupare insikt i ämnet. Litteraturstudien bidrog även till större kunskaper i ämnet vilka förenklade och riktade in undersökningsområdet i fallstudien.

Den empiriska studien visar att man i just detta fall använde sig av metoder som även beskrivs av litteraturen. Empirin visade på kunskaper om den faktiska vidareutvecklingen mer konkret. Metoderna som beskrevs i teorin specificerade genom sina mål vad som avses vidareutvecklas. I den empiriska studien visades det konkret vad som förändrades. Den empiriska studien belyste även vilka begränsningar man som utvecklare kan stå inför då man avgör vilka faktorer som kommer att gå vidare och bli vidareutvecklade.

Sammanfattningsvis visade studien på att användbarhet är främsta fokus då användare impliceras i vidareutvecklingen. Samtidigt som utvecklare, med syftet att förutse framtida behov, oavbrutet följer omvärldens förändringar vilka kan medföra nya krav eller önskemål i systemets funktionalitet.

## Nyckelord

Återkoppling, underhåll, vidareutveckling, systemutveckling, användbarhet, test.

# Innehållsförteckning

<b>1</b>	<b>INLEDNING .....</b>	<b>7</b>
1.1	BAKGRUND .....	7
1.2	PROBLEMOMRÅDE .....	7
1.3	SYFTE .....	8
1.4	AVGRÄNSNING.....	8
<b>2</b>	<b>METOD .....</b>	<b>9</b>
2.1	FOKUS.....	9
2.2	TILLVÄGAGÅNGSSÄTT .....	10
2.2.1	<i>Litteratur och teorier .....</i>	<i>10</i>
2.2.2	<i>Empiri .....</i>	<i>10</i>
2.2.3	<i>Induktion .....</i>	<i>11</i>
2.2.4	<i>Reliabilitet och validitet.....</i>	<i>11</i>
2.3	DATA .....	12
2.3.1	<i>Frågor mot insamlad data .....</i>	<i>12</i>
2.3.2	<i>Insamling.....</i>	<i>12</i>
2.3.3	<i>Analys av data.....</i>	<i>13</i>
2.4	ETISKA ASPEKTER .....	14
2.5	KÄLLKRITIK.....	14
2.6	FORSKARKRITIK.....	15
<b>3</b>	<b>VIDAREUTVECKLING AV ETT SYSTEM .....</b>	<b>16</b>
3.1	INLEDNING .....	16
3.1.1	<i>Systemutvecklingens livscykel.....</i>	<i>16</i>
3.2	FAKTISK VIDAREUTVECKLING.....	17
3.2.1	<i>Underhållsstadiet.....</i>	<i>17</i>
3.2.2	<i>Systemets livscykel .....</i>	<i>18</i>
3.2.3	<i>Förväntningar inför utveckling.....</i>	<i>21</i>
3.3	METODER FÖR IDENTIFIERING AV SYSTEMETS BRISTER.....	21
3.3.1	<i>Betatest.....</i>	<i>21</i>
3.3.2	<i>Usabilitytest .....</i>	<i>21</i>
3.3.3	<i>Usability test vs Betatest .....</i>	<i>22</i>
3.3.4	<i>Enkäter och intervjuer .....</i>	<i>23</i>
	<i>Att utforma och planera en enkät.....</i>	<i>23</i>
	<i>Typer av frågor .....</i>	<i>23</i>
	<i>Typer av svar.....</i>	<i>24</i>
	<i>Språkbruk.....</i>	<i>24</i>
3.3.5	<i>FAQ.....</i>	<i>24</i>
3.4	BRISTER PÅ IDENTIFIERINGSMETODER .....	25
3.4.1	<i>Problem och brister med betatestning .....</i>	<i>25</i>
3.4.2	<i>Problem vid insamling av stor mängd återkoppling.....</i>	<i>25</i>
3.5	SAMMANFATTNING .....	26
<b>4</b>	<b>EMPIRI.....</b>	<b>27</b>
4.1	BAKGRUND .....	27
4.2	TYP AV SYSTEM .....	27

4.3	AKTÖRER KRING SYSTEMET .....	28
4.4	SYSTEMETS BEHOV .....	29
4.5	INDIKATIONER TILL VIDAREUTVECKLING.....	29
4.5.1	<i>Buggar</i> .....	29
4.5.2	<i>Användbarhet</i> .....	30
4.5.3	<i>Sökbarhet och data</i> .....	30
4.5.4	<i>Aktörerna och systemet</i> .....	30
4.6	HUR MAN IDENTIFIERAR SYSTEMETS BRISTER .....	31
4.6.1	<i>Betatest</i> .....	31
4.6.2	<i>Användbarhetstest</i> .....	31
4.6.3	<i>Enkäter och intervjuer</i> .....	32
4.6.4	<i>Besök</i> .....	33
4.6.5	<i>Konkurrens</i> .....	33
4.7	VAD FÖRÄNDRAS OCH VIDAREUTVECKLAS .....	33
4.8	BEGRÄNSNINGAR.....	35
4.8.1	<i>Vad mer tillförs till följd av indikationerna</i> .....	36
4.9	SAMMANFATTNING .....	36
<b>5</b>	<b>RESULTAT .....</b>	<b>38</b>
5.1	FAKTORER MOT FRAMTIDA UTVECKLING .....	38
5.2	METODER FÖR IDENTIFIERING AV SYSTEMETS BRISTER.....	39
5.2.1	<i>Betatest</i> .....	39
5.2.2	<i>Usabilitytest</i> .....	39
5.2.3	<i>Enkäter och intervjuer</i> .....	39
5.3	BRISTER PÅ IDENTIFIERINGSMETODERNA.....	40
5.4	VAD FÖRÄNDRAS OCH VIDAREUTVECKLAS .....	40
5.4.1	<i>Beta- och användbarhetstest</i> .....	40
5.4.2	<i>Enkäter och Intervjuer</i> .....	41
5.4.3	<i>Studier och kundkontakt</i> .....	41
5.5	BEGRÄNSNINGAR.....	41
<b>6</b>	<b>DISKUSSION.....</b>	<b>43</b>
6.1	FÖRSLAG TILL VIDARE FORSKNING .....	44
6.2	REFLEKTIONER .....	45
<b>7</b>	<b>SLUTSATS.....</b>	<b>46</b>
<b>KÄLLOR</b> .....	<b>47</b>	
<i>Bilaga 1</i> .....	<i>49</i>	
<i>Bilaga 2</i> .....	<i>51</i>	

## FIGURFÖRTECKNING

FIGUR 1: ORSAK EFFEKT MATRIS .....	14
FIGUR 2 :VATTENFALLSMODELLEN, .....	17
FIGUR 3: SYSTEMETS LIVSLÄNGD. ....	19
FIGUR 4: MED ELLER UTAN ANVÄNDBARHETSTEST .....	20
FIGUR 5: ORSAK-EFFEKT MATRIS FÖR BETATEST .....	31
FIGUR 6: ORSAK-EFFEKT MATRIS FÖR ANVÄNDARTEST .....	32
FIGUR 7: FÖRSLAGETS VÄG TILL KRAV.....	36

# 1 Inledning

## 1.1 *Bakgrund*

Enligt systemutvecklingens livscykel, vattenfallsmodellen, befinner sig implementerade system i cykelns sista fas som efter implementering innebär underhåll (Reynolds, 1995). I den sista fasen ingår vidareutveckling som en del (Reynolds, 1995, Davis & Yen, 1999). Det är logiskt att system vidareutvecklas eftersom cykeln, beroende på olika teorier, består av sex till sju steg (Reynolds, 1995, Davis & Yen, 1999) och istället för att gå igenom dessa varje gång försöker man underhålla och utöka det sista steget. En utvecklare bör därför ha en grund till systemets fortsatta framträdande. Någonstans ifrån får man kännedom om hur systemets nästa steg i vidareutvecklingen ska se ut.

Vattenfallsmodellen beskrivs av många författare. Eftersom den sista etappen, underhåll, enligt Reynolds (1995) till största del innehåller förbättring och vidareutveckling av funktionalitet kan man undra vad det är som utvecklas. Forskning visar på en hel del metoder för insamling av återkoppling från användare. Som ett exempel brukar kommersiell mjukvara komma ut i betaversion och bli tidigt distribuerad till användarna för att möjliggöra testning och tillåta användare att ge sin respons innan det fullständiga systemet utges (Dustin, Rashka & Paul, 1999). Nielsen (1993) tar upp ämnet usability där usabilitytest eller användbarhetstest är en annan metod för insamling av återkoppling. Vidare talar Davis och Yen (1999) om enkät och intervjumetoder för ännu djupare men samtidigt bredare kontakt med användare. De ovanstående författarna poängterar även metodernas mål såväl som vad de försöker bidra med i just vidareutvecklingen. Målen skildrar metodens avsikt till vidareutveckling. Ett användbarhetstest har exempelvis som avsikt att testa systemets användbarhet för att slutligen eventuellt identifiera brister (Nielsen, 1999).

Det som oftast saknas i teorierna är de faktiska förändringarna i systemets vidareutveckling vilket kanske inte är så konstigt då litteraturen ofta talar generellt. Visserligen tas det ibland upp en del exempel men inget som konkretiserar teorin för att knyta den till verkligheten. Metodernas teoretiska mål symboliserar istället vad de ämnar utveckla och redogör därmed för vilka eventuella förändringar de avser medföra i verkligheten.

## 1.2 *Problemområde*

När ett system utvecklas och implementeras händer det, som även nämnt ovan, att fel och brister finns kvar. Ett system kan därför hamna i ett vidareutvecklings stadium. Utvecklare ställs då inför momentet att tillföra "rätt" funktioner samt reparera "rätta" brister. Bristerna samt de nya funktionerna blir till nya motiv för vidare utveckling och kan exempelvis fås via olika kontaktmetoder gentemot användare. Alla tänkbara motiv för vidareutveckling är naturligtvis inte möjliga att

genomföra, intressant är trots allt att veta vad som förändras samt hur man arbetar för att identifiera dessa behov. Intressant är också hur man etablerar kontakt med användare för att få respons angående systemet samt var nackdelarna finns. Motiven kan förslagsvis även "förutspås" då en händelse i världen kan leda till nya behov vilket även faller inom ramen för metoder genom vilka man får kännedom om nya motiv. Till exempel en ny lag.

Studien är tänkt att svara på frågorna:

Vilka faktorer bidrar till motiv för att ett system vidareutvecklas?

- Genom vilka evalueringsmetoder får man kännedom om dessa faktorer?
- Vilka faktiska förändringar i systemet bidrar faktorerna till?

### **1.3 Syfte**

Studien ämnar i att visa vilka evalueringsmetoder som används för att "pröva" ett system samt hur den återkoppling som evalueringsmetoderna genererar kan komma att påverka vidareutvecklingen av ett system. Undersökningen vill även visa vad som vidareutvecklas utifrån återkopplingen.

### **1.4 Avgränsning**

Undersökningen kommer att studera de evalueringsmetoder som används för vidareutveckling av ett system. Empiriskt studeras utvecklingen av ett system på ett företag. Studien kommer därför att ta hänsyn till problemet kring detta systems omgivning. Uppsatsen kommer även endast att ta i anspråk sådana evalueringsmetoder där användare är inblandade och som används av företaget. Tekniska tester som inte tar hänsyn till användare kommer ej att studeras, varken empiriskt eller teoretiskt.



## 2 Metod

Inledningsvis var mina intentioner väldigt specifika då jag endast ville fokusera på återkoppling via ett FAQ<sup>1</sup>-system. Efter en tid insåg jag att jag teoretiskt var väldigt begränsad och bestämde mig för att öppna upp problemformuleringen och även ta andra typer av påverkan i beaktning. Anledningen till detta var att jag i min inledande litteraturstudie iakttog en större bredd till mitt ämne. Jag observerade samtidigt flera olika typer av intryck mot systemet vilket jag arbetade kring under min praktik.

I anslutning till min referensram hittade jag då fler teorier kring min problemformulering vilket bidrog till en aning om vilken metod jag skulle kunna använda. Jag fick även lite tips från Agneta Olerup, professor i informatik vid Lunds Universitet, där jag till exempel fick förslag om generella systemutvecklingsböcker. Detta hade jag inte funderat på tidigare då jag främst sökte efter litteratur efter min problemformulering. I denna litteratur fanns det en hel del omskrivet vilket kom till hjälp och ledde mig in på ett konkret spår. Jag stötte på en hel del olika omskrivna insamlingsmetoder i litteraturen för att testa och utvärdera system inför utveckling. Parallellt med empirin började klarna och mitt ämne blev plötsligt ännu intressantare. Min förstudie väckte funderingar kring det som egentligen vidareutvecklas i ett system samt kring tillvägagångssättet för att nå informationen om det som ska vidareutvecklas. Förstudien fick mig även att begripa innebörden i de omskrivna metoderna då jag i en verklig systemutvecklingsprocess upplevde just samma tekniker.

### 2.1 Fokus

Miles & Huberman (1994) talar, i sitt kapitel om förberedelse inför en studie, om definiering av område och fall där "hjärtat" är fokus men omringas av ett område som man inte kan hoppa över. Samtidigt som en gräns mot vad som inte kommer att studeras knyter ihop själva territoriet. För att välja en bra metod har jag mitt utgångsläge på praktikplatsen var jag väljer att bedriva min empiriska undersökning på plats. Möjligheten ger mig chansen att få en djup inblick och själv erfara det som man annars strävar efter att få veta i en intervju. Runt omkring studien blir litteratur som behandlar generell systemutveckling och metoder för betydelsefull interaktion och testning med användare för att en helhetsbild ska skapas. Jag är undersökningspersonen själv och medverkar i ett projektteam på ett IT-företag, i Paris. Projektteamets uppgift är att förbättra och vidareutveckla en befintlig mjukvara. Min fokus kommer att ligga kring återkoppling och andra påverkande faktorer i just denna utvecklingsprocess.

---

<sup>1</sup> Frequently Asked Questions

## **2.2 Tillvägagångssätt**

### **2.2.1 Litteratur och teorier**

För att hitta relevant litteratur och teorier som inte bara finns i böcker utan likaså i artiklar använde jag mig av bland annat universitetets artikeldatabas ELIN för att från början ringa in mitt ämne. Jag sökte då efter allt som har med användarmedverkan i systemutveckling att göra för att få fokus. När frågeställningen och avsmalnandet av ämnet var definitivt hade jag mina nyckelord såsom: "återkoppling", "user"/"användare", "utveckling", "förfining" och "beta" att söka efter. Biblioteket på den skola var jag bedrev min föregående termins utlandsstudier, i Frankrike, kom väl till förfogande då min geografiska placering måttligt försvårade litteratursökningen. Förutom bibliotek har jag använt mig av elektroniska bokhandlar varifrån jag beställt litteratur med förkunskap om att dess användbarhet onekligen blir outhärlig i studien. Eftersom undersökningen är tänkt att knyta ihop mina fyra års studier har jag även dragit nytta av kurslitteraturen från tidigare studieår. Principiellt har jag förfogat över generell systemutvecklingslitteratur vilken rubricerar systemutvecklingens faser och där hamnade min koncentration företrädesvis på den sista fasen. Litteratursökandet var i huvudsak ganska svårt, jag fann inte tillräckligt med litteratur. Då min frågeställning specifikt vill upplysa om vad som vidareutvecklas och hur återkoppling fås begränsades antalet referenser vid sökning efter dessa ämnen. En annan oerhört lämplig sökmotor har Google Book varit där utdrag från böcker finns publicerade på Internet. Vid sökandet efter litteratur, som varit svår att få tag i fysiskt på bibliotek, har jag via Google Book kunnat hitta litteratur och söka igenom de sidor som intresserade mig. Mina huvudkällor inbegreps av några allmänna systemutvecklingsböcker och artiklar med lite särskilda teman användes för förstärkning och specialisering.

### **2.2.2 Empiri**

Enligt Bryman (1989) består kvalitativa undersökningar huvudsakligen utav tre olika informationskällor: medverkande observationer, ostrukturerade och delvis strukturerade intervjuer samt undersökning av dokument. Källor som tillgåtts i studien är dokumentering av systemet, kundservice som finns på Internet samt samtal med medarbetare.

Min empiriska bit kommer mycket att likna en fallstudie med en medverkande roll. Jag identifierar mig med en av Brymans (1989) omskrivna roller. Eftersom jag har en verklig arbetsuppgift känner jag igen mig i den fullständiga rollen vari man har en riktig och konkret arbetsuppgift men å andra sidan är offentlig forskare (Bryman, 1989). Jag utförde min undersökning efter min problemformulering vari företaget inte har varit inblandat. En inblandning skulle eventuellt ha kunnat resultera i dubbelt arbete för mig då företaget önskar en undersökning av något jag inte har planerat från början. Naturligtvis publicerar denna studie inte några konfidentiella dokument från företaget då tanken med studien blir att endast jämföra de verkliga händelserna mot teorierna.

Den kvalitativa forskningsmetodiken använder oftast ostrukturerade och löst strukturerade intervjuer, till skillnad från enkätliknande undersökningars stängda och helt strukturerade frågor (Bryman, 1989). Ostrukturerade intervjuer som informationskälla blir i mitt fall spontana samtal, möten och alla andra tillfällen var samtal kring ämnet blir huvudfrågan. Inga konkreta och fasta frågor ställdes utan det som för tillfället är aktuellt och essentiellt resulterar i samtalsämnet, även om jag stundtals förmådde att styra in på önskat ämne. Anledningen till valet av intervjumetod är att nästan all information går att tillgå via dokumenteringen varför strukturerade intervjuer inte var nödvändiga. Eftersom jag själv dessutom integrerats i företaget fick jag efter ganska kort tid tillräcklig kännedom för att skapa en förståelse om hur utvecklingsarbetet går till.

### **2.2.3 Induktion**

För slutledning har jag arbetat med en induktiv metod. Det induktiva det vill säga att man har data som man vill behandla med teorier, begrepp och modeller (Rienecker & Jörgensen, 2002), använder jag då jag inte endast genom empiri undersökt mitt ämne utan även teoretiskt. Jag kan heller inte helt och hållet förlita mig strikt på det induktiva då man inte kan komma fram till hundra procentig visshet genom metoden (Thurén, 2003). Med det menar jag att min undersökning endast innefattat ett fall vilket inte kan ge en generell slutsats. Teorierna speglar oftast generalitet medan praktiken är unik för varje fall. Thuren (2003) tar upp några fall för att bekräfta faktumet att trots enormt empiriskt material kan en sanning visa sig vara falsk. Därför har jag tagit avstånd från att med hjälp av min empiri uttrycka sanning för alla.

### **2.2.4 Reliabilitet och validitet**

Då jag använder mig av en induktiv metod, se ovan, bör man se upp med reliabiliteten (Thurén, 2003).

*Reliabilitet* eller tillförlitlighet innebär att undersökningen är korrekt gjord (Thurén, 2003, Svenning, 1999) och skulle ge samma resultat om någon annan gjorde likadant. Hur som helst borde en likadan studie, med samma problemformulering, utförd på samma sätt med stor sannolikhet sluta med samma resultat. En motsägelse skulle vara att en annan forskare skulle kunna uppfatta en likadan miljö på ett annat sätt. Samtidigt som personerna i den undersökta miljön inte skulle bete sig och reagera på exakt likadant vid ett annat tillfälle. Värt att påpeka är även att en likadan studie utförd för ett tiotal år sedan, eller ett tiotal år framåt, däremot skulle kunna resultera i liknande utgång då många systemutvecklingshandböcker har teorier med några år på nacken. Däremot kan en likadan studie utförd för många år sedan ändå ha motfaktorer. Dessa motfaktorer kan te sig i nya teknologier vilka ständigt utvecklas och idag exempelvis möjliggör större exponering mot Internet än förr. Internettekniken kan möjliggöra för fler metoder inför vidareutveckling. Min empiri koncentrerar sig dessutom endast på det system som jag avser att undersöka, närmare beskrivet i första samt fjärde kapitlet. En annan likadan studie utförd mot ett annat system skulle kunna resultera i annat resultat då utvecklingen kan skilja sig från företag till företag. Jag anser trots allt att jag uppnår en tillfredsställande

reliabilitet då min undersökning stärks med teorier som jag i min undersökning funnit relevanta.

Validiteten kan definieras som att man undersöker det man avser att undersöka och inget annat (Thuren, 2003). Svenning (1999) går djupare in på definitionen och menar att man använder rätt instrument för att komma fram till sitt resultat. I mitt fall strävar jag efter att få fram metoder för utvärdering av ett system och vad dess återkoppling innebär för vidareutvecklingen till systemet. Jag undersöker då empiriskt samt teoretiskt de evalueringsmetoder som används samt vilken återkoppling de genererar, för att slutligen identifiera förändringar i systemet. Jag vill inte utesluta det faktum att andra metoder inte skulle kunna undersöka samma problem.

## **2.3 Data**

### **2.3.1 Frågor mot insamlad data**

Jämförelse av teorin samt empirin görs med hjälp av undersökningsfrågor. Frågor som slutligen ställs mot det empiriska samt teoretiska resultatet för att leda till i svar åt min forskningsfråga och möjliggöra en slutdiskussion.

- ❖ Vilken typ av system utvärderas?
- ❖ Vilka motiv påverkar framtida utveckling?
- ❖ Metoder för identifiering av systemets brister?
- ❖ Brister på metoderna?
- ❖ Vad förändras och vidareutvecklas?

### **2.3.2 Insamling**

Empirin genererar data utifrån en fallstudie (Svenning, 1999) vari utveckling av ett system studerats under fyra månader. För att få en viss struktur på datan ställer jag några undersökningsfrågor, enligt ovan, gentemot den insamlade informationen. Enligt Yin (1994) ligger sex olika källor som grund för en empirisk studie; dokument, arkivregister, intervjuer, direkta observationer, deltagande observationer och fysiska produkter. Min studie hämtar sina data i de flesta av dessa källor. Jag kommer att studera befintlig dokumentation såsom broschyrer för de nya och de äldre versionerna samt färdiga arkiverade enkäter redan utförda av företaget. I min praktikperiod tillåts jag studera kundcentrals support för användare, utföra samtal med medarbetare samt praktisera den själva fysiska mjukvaran. Anledningen till att strukturerade intervjuer inte utförs är att min position tillåter mig att komma fram till svaren på undersökningsfrågorna. För att på ett effektivt sätt klarlägga och strukturera mina data väljer jag att placera den i olika modeller som presenteras av Miles och Huberman (1994).

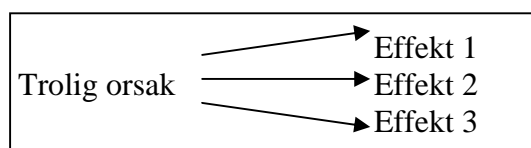
### 2.3.3 Analys av data

Miles och Huberman (1994) skriver om "memoing" som ett avsnitt i "föranalys" kapitlet arbetet vilket i stort sett överensstämmer med det som jag gör. Eftersom min studie handlar om att jag, på plats, måste absorbera och reflektera för jag en dagbok som ingår i en del av min praktikrapportering men utav vilken jag tänker dra nytta inför min undersökning. Miles och Huberman (1994) talar även om kodning. Koderna blir en typ av förkortade ord, etiketter (Svenning, 1999), i en jämförande matris (se avsnitt Kontrollmatris nedan) vilka möjliggör en konkret och enklare sammankoppling mellan empirin och teorin. Kodningen, eller etiketterna, blir som enkla "ordsymboler" för undersökningsfrågornas svar. Dagboksförandet, en parallell till memoing, hjälper mig att återvända till noterade tillfällen och situationer. Denna arbetsmetod i forskningen kan ses som förebyggande och kräver inte mycket tid (Miles & Huberman, 1994). Dagboken blir på något sätt en säkerhet på att vissa detaljer inte faller i glömska. En annan föranalytisk metod som utgör en del av min studie är som Miles & Huberman (1994) kallar det "Case Analysis Meeting" eller "analysmöte" för det studerade fallet. Här finner jag mig själv, en annan student och handledaren som mötespersoner då metoden kräver att man vid några tillfällen träffar någon kollega, kritisk vän eller liknande för att reflektera och diskutera studiens förlopp. Studenten i fråga är inte med i undersökningen utan fungerar endast som en idégivare och kritiker. Problemet man vill förebygga är att information inte faller bort eller måste bearbetas om för att den uppstod i början av studien och man efter ett tag låtit sig "glömma bort" den. Med handledaren och den andra studenten uppstår frågor och återblickar på det som blivit utfört och var i studien jag befinner mig. Vid dessa tillfällen tar jag anteckningar om kritik eller idéer som tagits upp och diskuterats mellan mig och den andra personen under dessa möten. Diskussionerna bidrog till en bättre förstådd uppsats. Som ensam forskare kan problemet att framföra ens information rätt uppstå och om ingen annan läser vad man vill framföra finns risken att uppsatsen inte blir förstådd.

#### **Orsak-effekt matris**

Problemet som undersökningen strävar efter att lösa är vilken effekt olika motiv till ett systems vidareutveckling resulterar i. Med hjälp av en Case ordered effects matrix (Miles & Huberman, 1994) eller "orsak-effekt matris" redogörs det för bland annat återkopplingens samt andra metoders påverkan. Min ambition blir att ta fram faktorer som leder till en viss effekt i systemet varför denna typ av datastruktur passar min frågeställning. Jag ämnar presentera mer än en effektmatrix eftersom systemet onekligen förändras och berörs annorlunda beroende på typen av återkoppling. För att sätta mina data i ett sammanhang så att den blir till information placerade jag den i en orsak-effekt matris enligt modellerna nedan.

En orsak-effektmatris kan se ut på flera sätt. Det sättet jag använder mig utav är presentation utav *en* anledning som kan bidra till flera effekter. Tanken är då att ta fram en orsak till varför ett antal förändringar blir vidareutvecklade i systemet. Matrisen underlättar och visar konkret på något som bidrar till vidareutveckling.



FIGUR 1: ORSAK EFFEKT MATRIS

## 2.4 *Etiska aspekter*

Den första frågan jag ställde mig inför studien var ifall min studie var värd att genomföra och vad den skulle bidra till? Svaret jag gav mig själv var att studien kan komma att hjälpa andra systemutvecklare i arbetet gentemot användare. Några frågor som Miles och Huberman (1994) ställer sig i sitt etiska avsnitt är ifall studien kommer att bidra till min karriär? Då vidareutveckling av system är en så pass stor och viktig fas i systemutvecklingens livscykel såsom i själva systemets livscykel är arbetet där av väldigt stor signifikans. I min studie kommer jag redan att få en lite djupare inblick i hur underhåll av system går till. Att utveckla ett system slutar oftast inte när man implementerat systemet utan lika mycket arbete kan förväntas därefter vad gäller underhåll och utveckling (Reynolds, 1995).

Då min undersökning kommer att utföras vid ett företag fullt med konfidentiella uppgifter är min etiska moral ständigt viktig då jag inte på något sätt vill skada företaget genom att gå ut med hemliga uppgifter. Min roll som medverkande forskare begränsas utav etiska grunder i egenskap av risken att skada och göra intrång hos forskningsobjektet (Bryman, 1989). Jag kommer ständigt att tänka på tre etiska val, diskuterade av Miles och Huberman (1994), vid min vistelse på företaget: beneficiering – att maximalt få ut det jag behöver till min studie utan att skada, riskera och göra fel. Konfidentiell information som jag har tillgång till kommer inte på något sätt att publiceras. Informationen kommer endast att användas till för att försöka identifiera paralleller i riktning mot teorierna.

## 2.5 *Källkritik*

För att uppnå en tillfredsställande sanningsnivå i mina litteraturkällor har jag försökt finna mer än en källa inom samma område. Det syns tydligt då jag har använt några olika generella systemutvecklingsböcker. De källor som var lite specifika är enskilda artiklar som talar om något specifikt. Litteraturkällorna är författade av akademiker och uppnår, enligt mig, en tillfredsställande sanningsgrad till min studie som även den har ett akademiskt ändamål. De empiriska källorna kan

vara otillräckliga eftersom de onekligen blivit dokumenterade och återberättade av någon. Denna dokumentation är trots allt kontrollerad inför sin publicering men kan kanske ändå inte undvika en viss partisk färgning. Detta är ofrånkomligt och medveten om det har jag försökt tolka all information opartiskt.

## **2.6            *Forskarkritik***

Enligt Yin (1994) bör en fallstudie utföras av en erfaren forskare. Efter att ha genomfört en tidigare mer omfattande studie inför kandidatexamen vill jag inte betrakta mig som en erfaren forskare. Jag känner dock igen några av hans påståenden som karaktäriserar en fallstudie. Särskilt den om att studien oftast förändras och inte blir som man har tänkt från början. Eftersom jag från början inte visste hur fallet skulle se ut blev även min studie lite förändrad under förloppet. Konkret innebär det att jag efterhand lärde mig mer och insåg att systemet var mer komplext och att själva fallet hade mer relevant fakta till mitt problemområde än vad jag förmodat från början. Komplexiteten var naturligtvis motiverande och drev mig till att studera ännu mer samt söka fler paralleller eller motsägelser inom litteraturen. Yin (1994) talar om en forskarkvalité som innebär att personen som utför studien måste vara flexibel och adaptiv. Detta är en erfarenhet som jag fick uppleva då just min studie blev lite annorlunda än min förväntan. Trots att studien inte blev precis som jag tänkt från början har jag aldrig helt och hållet behövt ändra på mitt initiala problemområde. Som oerfaren forskare kan detta även ha bidragit till brister i min studie då min adaptationsnivå samt flexibilitet inte är tillräckligt god. Brister såsom att jag eventuellt kan ha bortsett från något som en erfaren forskare kunnat dra nytta av. Rent konkret har jag kanske inte insett värdet i en process eller någon dokumentation i den empiriska studien vilken därmed har hamnat utanför min studie. En annan kvalité påpekad av Yin (1994) är att forskaren måste vara opartisk i sina bedömningar. Jag anser mig själv som opartisk men kan naturligtvis aldrig komma bort från det helt och hållet. Framförallt eftersom jag genomförde denna studie ensam. Detta har kunnat påverka mitt empiriska resultat då jag som forskare omedvetet bortsett från sådant som enligt mig själv bestrider min studie. Även min interna position i en organisation kan påverka slutsatserna såsom att jag själv blivit påverkad av partisk information från personerna runtomkring. Jag har dessutom studerat brochyren samt dokument skapade av själva företaget. Partisk information kan ha belyst empirin positivt, då exempelvis externa brochyren ofta vill förgylla information, detta kan i sin tur ha påverkat mina slutsatser. Jag har trots allt varit medveten om detta och som tidigare nämnt använt mig av en metod där jag bollat idéer med en annan student samt min handledare.

## **3 Vidareutveckling av ett system**

### **3.1 Inledning**

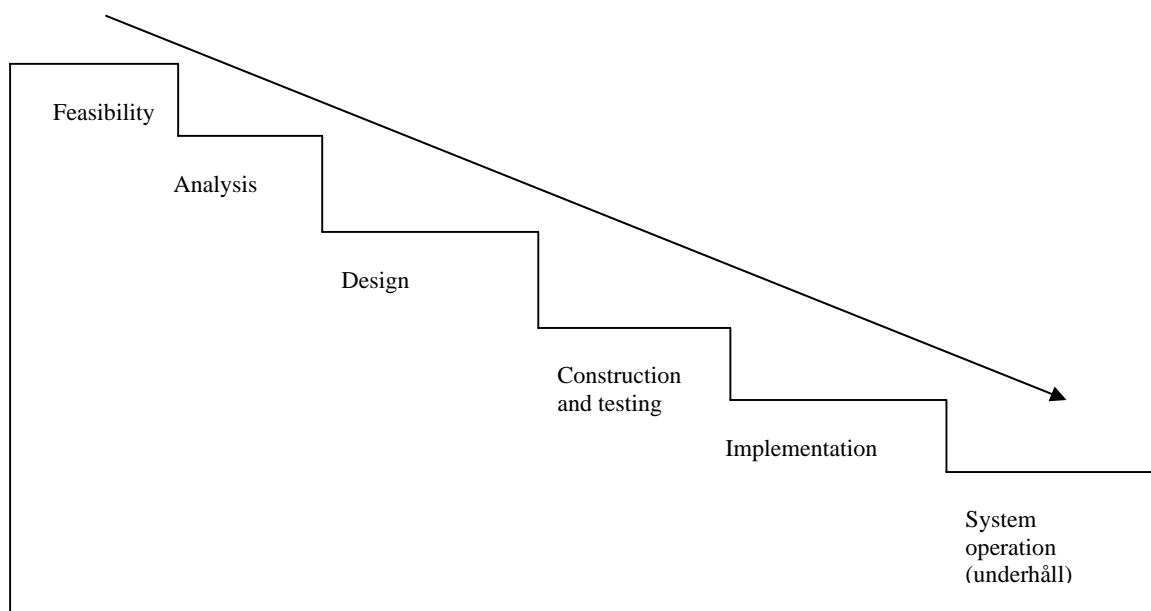
Att involvera användare i utvecklingen av system ökar sannerligen användbarheten av dessa system poängterar Hilbert och Redmiles (2001). Samtidigt kan man antyda på att ett system med god användbarhet förmodligen ökar sin konkurrenskraft mot andra system på marknaden. Författarna menar även att Internet öppnar för billiga metoder för att tillgå användares synpunkter inför utvecklingen. De talar då dels om möjligheten att snabbt och omfattande kunna distribuera mjukvara för evaluering samt möjliggöra metoder för insamling av återkoppling. Via Internet har man som option att exempelvis öppna ett kundcenter vari användare har möjligheten att lösa sina eventuella problem med hjälp av en FAQ-databas.

Återkoppling är heller inte alltid representativt för alla användare då den oftast kommer från klagande användare och kan klassas som negativ (Nielsen, 1993). Hilbert och Redmiles (2001) beskriver även användare som är upptagna och eller har svårigheter för datorsystem och därmed kanske låter bli att lämna några synpunkter. Inför den slutliga versionen korrigeras bristerna med hjälp av responsen på beta- och även användbarhetstester eftersom utvecklarens förväntningar inte alltid motsvarar den verkliga användningen (Hilbert & Redmiles, 2001).

#### **3.1.1 Systemutvecklingens livscykel**

Systemutvecklingen går igenom en del faser. Reynolds (1995) talar om fem olika stadier. Planering, analys, design, konstruktion och underhåll utgör dessa fem. Även den kända vattenfallsmetoden innehåller liknande stadier (Reynolds, 1995, Davis & Yen, 1999). Figur 4 som visar vattenfallsmodellen illustrerar hur dessa faser är relaterade till varandra. Den sista fasen preciserar hur viktig innebörden är i att underhålla och vidareutveckla system. Det finns olika anledningar till varför ett system behöver underhållas och vidareutvecklas. Den huvudsakliga är oftast att affärsvärlden och dess behov förändras genom tiden (Reynolds, 1995). Målet med underhållet i det sista stadiet blir alltså att upprätthålla systemet så att det uppnår en godtagbar nivå (Davis & Yen, 1999).





FIGUR 2 :VATTENFALLSMODELLEN,  
ritad efter Reynolds (1995) sid. 304.

## 3.2 *Faktisk vidareutveckling*

### 3.2.1 Underhållsstadiet

Underhållsstadiet har sin period från och med systemets offentliggörande och varar under all tid som systemet existerar därifrån. Själva underhållsarbetet och systemutvecklingen har en liknande livscykel då analysarbete med mera måste utföras inför själva vidareutvecklingen (Davis & Yen, 1999). Frågor som man som utvecklare bör ställa sig när det kommer till underhåll är ifall systemet ifråga fortfarande möter användarnas krav. Vidare bör man även tänka på ifall systemet har en värdig anledning för att fortfarande köras samt ifall ändringar i affärsvärlden och/eller i lagstiftningen kan utgöra tillräckligt stora anledningar för att systemet behöver förändras (Reynolds, 1995). Samme författare påpekar även att det heller inte är ovanligt att organisationer förändras. Dessa förändringar beror delvis på att omvärlden runtomkring förändras. I sin tur leder detta till ett krav på ändringar i systemen som stödjer organisationen. Davis och Yen går in på ett analysområde som kallas "industry analysis paradigm". De påpekar hur viktigt det är med att tävla mot andra konkurrenter. Därför är även analys av konkurrenters produkter viktig. De nämner metoder som nyheter, tidningar och undersökningar av försäljning för att informeras om händelser hos konkurrenter.

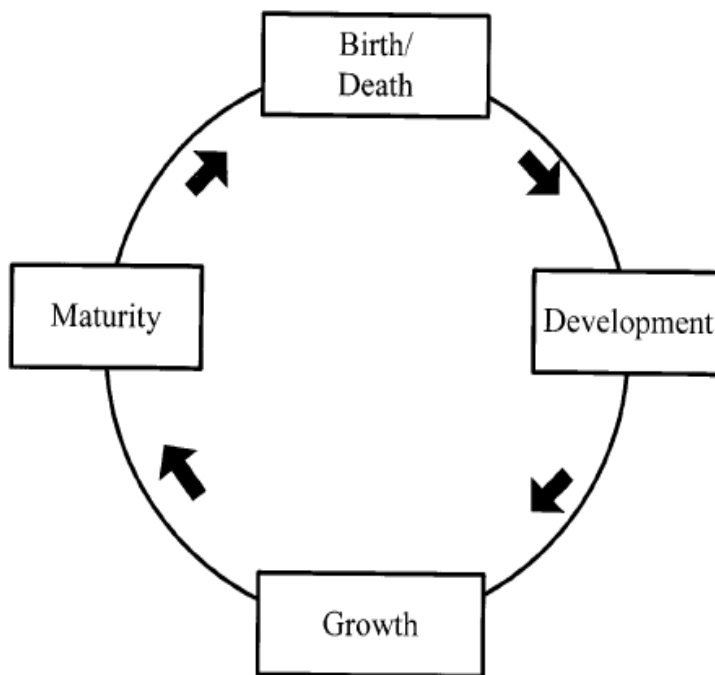
Adaptivt underhåll är ett av partierna i underhållsarbetet som Davis och Yen (1999) tar upp. Underhållsarbetet består av tillägg och utveckling av nya funktioner efter diverse krav såsom nämnt ovan. Denna typ av underhåll utgör sextio procent av underhållsarbetet enligt Reynolds (1995). Perfektivt underhåll utgör förbättringar i

systemets effektivitet, funktionalitet men även uppgraderingar av operativsystem (Davis & Yen, 1999). Reynolds (1995) menar att underhållsarbete där system flyttas, konverteras eller på något sätt förändras för att fungera i andra miljöer innebär tjugo procent av underhållsarbetet. Resterande tjugo procent i Reynolds (1995) uppdelning av typiskt underhållsarbete påstår han vara underhåll av korrekationer. Något som Davis och Yen (1999) kallar för korrektionsunderhåll. Det är mer eller mindre vad det låter, korrigerig av buggar i systemet. Detta kan alltså ses som motiv för att system behöver vidareutvecklas.

Förbättringar och vidareutveckling är tillsynes majoriteten i underhållet av system. Medan själva utvecklingen fokuserar på funktionalitet, flexibilitet och användbarhet blir uppgiften i underhållet att förfina och förbättra operationerna (Davis & Yen, 1999). Reynolds (1995) menar att detta görs för att systemet ska fortsätta att bli användbart även framöver. Man vill alltså inte vänta på att systemet ska "dö ut" (se systemets livscykel i kap 3.2.2) innan man börjar vidareutveckla utan håller livscykeln uppe.

### **3.2.2 Systemets livscykel**

Även själva systemet har en livslängd. Olika förändringar i omvärlden leder till systemets "död". När det är nytt och färdigskapat är intresset högt. Systemet stagnerar däremot efter "mognadsstadiet" och dör sedan ut (Davis & Yen, 1999). Systemets livscykel skulle kunna kopplas ihop med systemutvecklingens utvecklingsstadier då systemet vid sin "mognad" och fram till sin "död" behöver underhållas och vidareutvecklas för att kunna bevaras "vid liv". Att bevara ett system "vid liv" kan betraktas som en annan anledning till varför det behöver vidareutvecklas. Medan dess "födelse" skulle kunna kopplas ihop med implementeringen. Även analys och design kommer in i bilden här. Då man vid underhållet och vidareutvecklingen återigen bör påbörja en analys för att kunna möta användarens behov och erbjuda denne ett system som möter de nya kraven.



FIGUR 3: SYSTEMETS LIVSLÄNGD.  
(Davis & Yen, 1999, kapitel 1;1.4.2 The system life cycle)

”Varför låta en människa göra fel när man kan låta en dator göra rätt?” Detta är ett ofta förekommande slagord från IT-chefen påstår Per Flensburg, Professor i Informatik vid Växjö Universitet, i sin artikel *”Arbetet i den digitala tidsåldern”*. Han spinner vidare på detta påstående som alldeles för optimistiskt påpekar en dators perfekta arbete. Detta arbete är givetvis inte perfekt och han tar upp ett exempel då olika artikelsystem som utvecklades ofta byggde på artikelnummer istället för artikelnamn. Man påstod att inmatning av artikelns nummer, till exempel ”7654-2”, i systemet gick fortare än artikelns namn, ”Mutter, M8, förkromad”. Samme författare omnämner i sin bok *”Mänskligare datasystem”*, 1999, även författad av Siv Friis, att utvecklare ständigt strävar efter numrering av olika slag. De tar, i ett samlande fall, upp ett exempel om att man på samma sätt önskar att numrera människor som är en del av olika system. Identifieringen av dessa personer i det givna systemet sker alltså med hjälp av nummer istället för med deras namn. Anledningen var att namnen i systemet måste vara unika för att kunna särskiljas och därför blev numreringen ett måste. Ingen tänkte på arbetstiden i sökning bland nummerregister som detta skulle medföra. Slutligen blir arbetstiden inte kortare genom inmatning av ett nummer och systemet tillför inte någon effektivare arbetsinsats. Allt detta berodde i grund och botten på utvecklarnas misstro mot användarnas förmåga i att kunna skapa unika namn och undvika dubletter. En dator, å andra sidan, särskiljer på exakt samma sätt bokstäver från siffror varför även bokstäver smidigare skulle ha kunnat användas. Ett annat exempel på ett systems misstag tas upp av Min-Yang Wang. Som i sin doktorsavhandling *”Usability Evaluation for Human-Computer Interaction (HCI)”* (1992), beskriver i ett avsnitt om kommunikationsmetoder mellan en dator och en människa ett systems meddelande som endast kan identifieras som vilseledande.

*"You are about to save your design and return to the ENTER menu.*

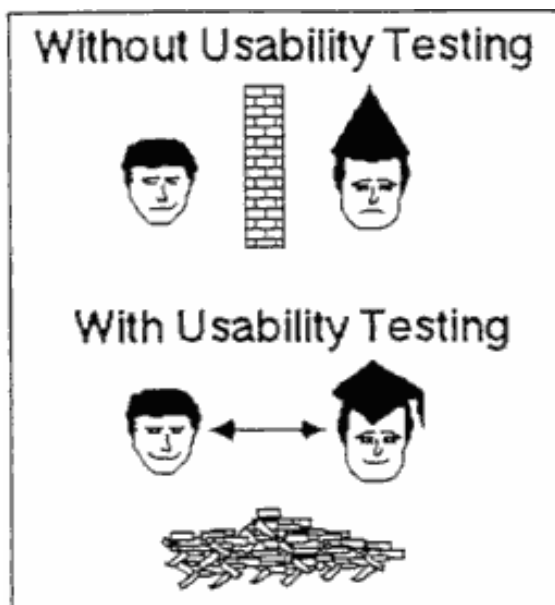
*Do you want to save your design before closing it?"*

*Yes                      No*

(Min-Yang Wang, 1992, Sid. 8)

Hur skulle användare veta ifall svaren Yes eller No syftar på första eller andra meningen?

Jag tar upp dessa exempel för att betona systemens behov av vidareutveckling, förbättring och korrigerigering även efter implementeringen. Klart och tydligt förstås det att system sedan länge vidareutvecklas då de från sin första lansering med största sannolikhet innehåller skavanker, producerar missförstånd eller förvirrar användare på ett eller annat sätt. Man kan fråga sig varför utvecklare inte gör rätt för sig från början då problem finns sedan länge.



FIGUR 4: MED ELLER UTAN ANVÄNDBARHETSTEST  
(Dumas & Redish, 1999, s. 33)

Efter att ha tagit del av exemplen ovan kan man kanske dra slutsatsen att kommunikationen mellan utvecklare och användare i utvecklingssammanhang inte alltid stämmer överens. Fortsättningsvis kanske man kan komma fram till att detta resulterar i "missförstånd" och/eller "misstag" i den slutliga produkten. Mitt påstående kan förstärkas med Reynolds (1995) olika anledningar till varför stora insatser görs åt problemåtgärder. Han påstår att klen kommunikation mellan utvecklare och användare leder till att kraven inte blir konkret specificerade, ibland inte alls. Eventuellt kan bristerna bero på utvecklarens förväntningar inför själva arbetet (se nästa avsnitt). För att komma över detta finns testning av system, var användarna spelar en central roll. Där kontrollerar och verifierar man att systemet möter användarnas behov samtidigt som fel och buggar kan upptäckas. Med hjälp av olika testmetoder lokaliserar man även upphovet till felen för att i sin tur vara kapabel till att korrigera dem (Davis & Yen, 1999).

### **3.2.3 Förväntningar inför utveckling**

Utvecklare har inför utvecklingen av ett system oftast förväntningar till hur uppbyggnaden av systemet kommer att se ut. Dessa förväntningar grundar sig på olika moment; kravspecifikation och tidigare erfarenheter hos utvecklare både vid utvecklande och vid användande av system. Förväntningarna styr sedan en del av utvecklingen. Exempelvis genom att verktyg som man antar kommer att användas mer än andra görs mer lättåtkomliga i systemets meny medan verktyg som inte förväntas användas lika frekvent riskerar att hamna någonstans i en undermeny. Med hjälp utav användares återkoppling av betaversioner (se Betatest nedan) förebyggs felaktigheter i ens förväntningar i den slutliga versionen (Hilbert & Redmiles, 2001). Flensburg och Friis (1999) talar om prototyper som en teknik för att skapa kravspecifikationer. En prototyp kan testa användbarheten (se 3.3.2 Usabilitytest) vilket enligt denna teori i verkligheten kan utveckla en grund för en kravspecifikation åt en framtida version. En fördel med metoden är att användarna som testar kan ge sin respons vilken eventuellt bidrar till en korrigerig i systemet (Flensburg & Friis, 1999) och i tid inför den riktiga lanseringen.

## **3.3 Metoder för identifiering av systemets brister**

### **3.3.1 Betatest**

Dumas och Redish (1999) poängterar användbarheten och hur den ofta inte tänks på förrän i slutet av utvecklingen, då kommer betatestningen in i sammanhanget. Betatestning används av många företag inför lanseringen av en ny produkt. Detta för att utvalda användare ska kunna ge sina synpunkter på det kommande systemet och identifiera buggar. Återkopplingen som fås kommer lagom fram inför lanseringen av den fullständiga produkten (Nielsen, 1993). Ofta utvecklas en ofärdig version för att användare ska testa den.

Beta testning borde dock inte vara den enda evalueringsmetoden då man förmodligen redan är ganska långt kommen i utvecklingen och vissa moment kan leda till ganska omfattande eller omöjliga korrigeringar. När insamling av återkoppling utförs är det viktigt att låta användaren veta att informationen tagits på allvar. Någon form av svar bör lämnas som bekräftelse. Detta kan i sin tur ses som en nackdel med användaråterkopplingmetoden men å andra sidan kommer användare, som ansträngt sig för att ge sina kommentarer utan att få bekräftelse, sluta att ge återkoppling (Nielsen, 1993).

### **3.3.2 Usabilitytest**

Alla usability eller användbarhetstest har några gemensamma huvuddrag. De har bland annat som uppgift att utföra test gentemot riktiga användare och med riktiga operationer. Sedan har man som utvecklare möjlighet att observera och analysera problem samt det som användare gör och säger. Målet med testningen är att slutligen förbättra användbarheten i produkten. Olika test har olika mål. Målen

avgör för hur testet ser ut och hur det ska utföras. För att exempelvis testa användningen av den tänkta produktens menyer finns möjligheten att utveckla någon form av prototyp eller till och med rita upp på papper. Testets resultat kan hjälpa till i själva utvecklingen inför lanseringen. Representationen utav testpersonerna måste vara verkliga användare till produkten, eller framtida användare. (Dumas & Redish, 1999). Samt måste återspegla alla nivåer av datorvana (Nielsen, 1999). Utvecklare kan inte på något sätt utföra ett användbarhetstest trots att de är i stånd att kunna ge synpunkter och återkoppling. Användare måste alltså vara de verkliga och slutliga användarna för att resultatet i slutändan ska återspegla det verkliga behovet. Som utvecklare och testledare är det viktigt med förberedelse av testuppgifter. De måste motsvara verkliga operationer och därför är förståelsen utav klientens arbetsuppgifter och därmed syftet till deras behov av just denna produkt essentiell. Observationer, inspelning samt dokumentation av testningen utgör resultatet som ska analyseras. Utifrån materialet identifierar man verkliga problem som fullgör grund för vidare förslag på förändringar inför vidareutvecklingen. Resultatet har inte endast som avsikt att förändra produkten utan kan även ha en inverkan mot systemutvecklingsprocessen och ha en förebyggande effekt gentemot framtida utveckling. Ett användbarhetstest kan utföras när som helst under utvecklingen av ett system (Dumas & Redish, 1999).

En prototyp kan beskriva utseendet och funktionaliteten för applikationen. Den kan därför exempelvis fungera som ett material till ett användbarhetstest. En prototyp som endast visar på utseende behöver inte ha någon funktionalitet, detta kan däremot ge negativa effekter såsom att användaren endast koncentrerar sig på eventuella fel i prototypen och får en generell negativ bild över systemet. Därför är även korrekthet viktigt. (Flensburg & Friis, 1999). Prototypen som endast visar funktionerna behöver nödvändigtvis inte ha något utseende, det är dock enligt Flensburg & Friis, 1999 klokt att även implementera utseendet med tanke på effekterna vid en inkorrekt prototyp.

### **3.3.3 Usability test vs Betatest**

Dumas och Redish (1999) tar upp några väsentliga skillnader mellan de olika testmetoderna. Det rör sig framförallt om responsen från användarna. Då ett betatest är okontrollerat och mer fritt kan man inte med största säkerhet få tillbaka någon rapportering med man vid ett användbarhetstest är med och observerar. Betatestare kan vara för upptagna av sitt arbete och låta rapporteringen falla åt sidan. Eftersom ett användbarhetstest är kontrollerat och lett utav utvecklarna bestäms de operationer som ska testas. Därmed kan man koncentrera sig på det testet ämnar återspegla och de områden som oroar utvecklarna. Ett betatest är fritt och användarna utför okontrollerade operationer således kan ett betatest helt och hållet gå förbi utvecklarens förväntningar. Men på samma sätt kan det utfalla i återkoppling på områden som utvecklaren inte har förväntat sig.

### 3.3.4 Enkäter och intervjuer

Metoden enkät för insamling av användares intryck utifrån ett befintligt system kan ses som effektivt då man tillåts samla in användares subjektiva synpunkter (Gulliksen, 1996). Metoden är strukturerad och betraktas som en formell variant av en intervju. Enkäter är tidsbesparande, kostnadseffektiva och når ut till många.

En klar nackdel är enkätens bristfälliga förmåga i att bli klart förstådd. Frågorna som ställs kan bli svårtolkade och medför därmed ett svar grundat på ett missförstånd. En annan svaghet är att motivationsnivån är lägre än om man exempelvis skulle intervjuas personligen (Min-Yang Wang, 1992, Davis & Yen, 1999).

#### Att utforma och planera en enkät

För en optimal design av en enkät nämner Davis och Yen (1999) ett antal faktorer. Först och främst poängterar de innebörden i frågornas ordning. Svaret kan påverkas beroende på hur frågorna ordnats. En första fråga kan mer eller mindre påverka besvararen i sina fortsatta svar. Författarna tar upp några exempel och beskriver däribland hur svaren konkret kan påverkas till följd av frågornas placering. De menar, ur ett allmänt perspektiv, att om en enkäts fråga som exempelvis skulle vara att få ut synpunkter beträffande en regerings förfogande utav ekonomiska resurser kan svaren påverkas inte bara genom frågornas placering utan även genom dess karaktär. Mer specifikt poängterar de att föregående frågeställningar med positiv natur gentemot regeringens alla goda tjänster influerar reaktionerna på så sätt att den utfrågade blir positivt påverkad och yttrar sig därefter. Föreläsningsvis belyser de hur svarsalternativens struktur, även de, påverkar frågans svar. Förutom att en utfrågad har tendensen att i de flesta fall välja något av de första alternativen borde svarsalternativens komplexitet och längd noga beaktas. Ett svar som nämligen verkar långt och svårt att förstå leder användaren till att välja att svara med ett annat alternativ som han eller hon känner sig förstått bättre och lämnar därmed en inkorrekt synpunkt.

Genom ordningen och frågornas samt svarsalternativens natur kan man alltså påverka sin tänkta målgrupp i att svara på ett eller annat sätt. Detta kanske man kan dra till sin fördel om man så önskar att få en viss typ av reflektioner tillbaka. Samtidigt är det viktigt att ha i åtanke att enkätens resultat slutligen kan få en felaktig konsekvens ifall man ej i förhand har planerat sin tänkta studie.

#### Typer av frågor

Två typer av frågor diskuteras av Davis och Yen (1999); öppna eller stängda frågor. De talar sitt enkla språk. Antingen vill man ha svaren efter olika alternativ eller mer detaljerat till en öppen fråga. Skillnaderna tenderas inte endast i svaren och det man kan få ut av dem utan även i hur efterarbetet kan komma att utspelas. Till slutna frågor har man de olika alternativen att gå på och för en viss statistik därefter. De öppna frågornas mer detaljerade och subjektiva svar kan resultera i exempelvis tolkningsproblem. Det är med största sannolikhet djupare och utförligare återkoppling via öppna frågor men samtidigt kan olika tolkningar förorsaka "fel" återkoppling.

### **Typer av svar**

Det resoneras även om relevansen i valet utav frågekaraktär då denna i sin tur medför en typ av svar. En sluten fråga med svarsalternativ kan tvinga användaren att lämna ett svar som inte helt och hållet motsvarar dennes uppfattning. Svarsalternativen måste innehålla en så stor bredd som möjligt för att fånga in den utfrågades åsikt med största närhet. En metod som föreslås av samma författare är kontrollfrågor. En fråga ställs för att undersöka användarens kännedom om temat för att sedan låta denne gå vidare eller hoppa över några steg.

Vad ska man då kräva av användarens kunskap? Man kan oundvikligen komma ifrån faktumet att denne riskerar att inte ha någon vetskap i frågan som ställs och enkätens natur måste därför tillåta negativa svar. Dessa svar medför tyvärr ingen återkoppling men måste beaktas och bevilja att den utfrågade målgruppen i förekommande fall inte har någon kännedom om ämnet.

### **Språkbruk**

Frågorna och svaren bör utformas precist och med klarhet. Om man vill undvika alldeles för öppna svar som i slutändan kanske inte motsvarar den informationen man är ute efter är kravet i frågans precishet väldigt högt (Davis & Yen, 1999). På så sätt kan ju även en öppen fråga mynna ut i användarens verkliga svar utan att det för den sakens skull blir långt och svårtolkat. Viktigt är även användandet av naturligt språk. Användaren bör förstå den enkät som han eller hon ska svara på och därmed eventuellt välja ett svarsalternativ som överensstämmer med ens uppfattning.

I en enkät kan man alltså styra det man vill ha svar på eftersom den formas av det egna företaget. Med hjälp av svarsalternativ till frågeställningarna dirigeras användaren in på den önskade linjen från företagets perspektiv. För att kringgå det korta svaret i en fråga som i sin tur lämnar bidrag till mindre ingående synpunkter kan man ställa öppna frågor och till följd därav tillåta användaren att med egna subjektiva ord formulera sina åsikter gentemot systemet.

### **3.3.5 FAQ**

För att besvara användares frågor och avlasta supportavdelningen på ett företag är en FAQ-databas effektiv (Lester Warby III, 2005). Samme författare menar även att dessa baser oftast är webbaserade, sökbara och för fyllda med många frågor och svar som är lättillgängliga för målgruppen. Baserna brukar vara dynamiska så att man kan ställa frågor ifall man inte hittar det man söker och kan på så sätt komma i kontakt med utvecklaren. De bygger oftast på dokumentation som finns till mjukvaran. En användare som alltså får problem med något i ett system han eller hon använder sig utav kan eventuellt söka sina lösningar i en FAQ-databas och snabbt komma till rätta. Detta kanske även kan fungera som en dynamisk guide och handbok som är sökbar. Författarens påstående om lättillgänglighet kan ifrågasättas då målgruppen kan innebära användare i olika nivåer vad gäller datoranvändning. För en ovan användare är sökning tvunget inte något enkelt därför kan denne ställas inför en komplicerad bas som av utvecklaren anses lättillgänglig.



### **3.4 Brister på identifieringsmetoder**

#### **3.4.1 Problem och brister med betatestning**

Dumas och Redish (1999), påpekar att betatestningen inte alltid ger återkoppling för att skapa motiv för utveckling åt användbarheten. Några faktorer som återger denna brist är; användare som är upptagna och glömmer att rapportera när saker och ting går fel samtidigt som de kan glömma bort sina operationer inför ett fel. Andra problem vad beträffar användare och dess rapportering tas upp av Hilbert och Redmiles (2001), de menar att användarna oftast är mer fokuserade på att få sitt arbete gjort i stället för att utvärdera och rapportera problem. De kan då tänkas använda programmet som vanligt och eventuella småfel ignoreras. Fortsättningsvis diskuteras användarnas vana vid vissa gränssnitt av Hilbert och Redmiles (2001) som ett problem. De anser att återkopplingen riskerar att bli ensidig. Då användare som har enklare för särskilda gränssnitt ger sina synpunkter medan de som har svårare för att förstå den gör det enklare för sig och svarar att det är väl designat eller låter bli att lämna synpunkter.

Vidare påpekas faktumet att det i ett betatest endast testas sådant som användare kan komma att utföra och inte sådant som man själv som utvecklare kanske hade önskat lägga mer tyngd på. Hela betatestningen beror på vad användaren kan få för sig att göra. Användare kan med en betaprodukt ämna att utföra riktiga arbetsuppgifter, risken finns att den ofärdiga produkten kan innehålla fel och skapa allvarliga konsekvenser i användarens arbete. Skulle produkten innehålla mycket fel kan det leda till besvär hos utvecklaren som utgivit ett system med stora felaktigheter till vilket användarna kan tappa förtroendet. De riskerar att betrakta produkten som frustrerande och inte vilja införskaffa produkten. Utvecklande företaget kan generellt tappa det förtroende som kunder har även för andra produkter (Dumas & Redish, 1999).

#### **3.4.2 Problem vid insamling av stor mängd återkoppling**

Hilbert och Redmiles (2001) tar upp några gemensamma problem som kan gälla alla typer av återkoppling insamling. Ett av problemen identifieras som "selection problem" eller urvals svårighet det vill säga att data som samlats in oftast endast innehåller en liten del relevant information i förhållande till sin storlek. Det leder till att rätt urval måste utföras för att samla in det som är av betydelse och som man kan utnyttja. Ett annat problem som identifieras av författarna är "reduction problem". De menar att reducering kan utföras redan i analysfasen av den insamlade informationen så att man inte ska behöva gå igenom det återigen efteråt. Företaget kan då med andra ord redan i sin insamling rikta vad man vill samla in, genom att exempelvis rätt frågor i en enkät ställs redan från början, och vid insamling och analys direkt ta bort vad som är irrelevant.

### **3.5 Sammanfattning**

Teorierna talar om generella system och poängterar då att underhåll börjar med en gång från och med lansering. Det har som mål att upprätthålla systemets som tillfredsställande gentemot användarnas behov och att det befinner sig på en godtagbar nivå.

Eventuella fel i ett system samt de nya behoven kommuniceras fram med olika metoder mellan utvecklare och användare. Via betatestning lanseras en ”ofärdig” version av systemet en tid innan lanseringen. Genom att testa sina arbetsuppgifter är det meningen att de ska upptäcka eventuella fel samt skapa egna synpunkter för att slutligen kommunicera dessa till utvecklarna. Ett annat test som används är användbarhetstest där användarna till skillnad från betatestet kontrolleras och ges uppgifter. Man testar antingen en prototyp eller använder sig utav papper och penna.

Förutom betatestning kan man få in återkoppling via enkäter. I anslutning till en betaversion eller till en utgiven version av systemet kan man via en enkät få in användares synpunkter och därmed ställa frågor och rikta in användaren till att svara på det man önskar att undersöka. En djupare variant av direktkontakt med användare är intervjuteknik.

Överlag brister de olika metoderna vad gäller representation av respondenter. Det är svårt att få kännedom om huruvida en godtagbar bredd beträffande respondenternas kunskap samt erfarenhet uppnåtts. En annan brist är ignorans från tänkta respondenter då de prioriterar sina arbetsuppgifter och kanske inte ägnar någon tid åt att ge återkoppling.

Det som slutligen vidareutvecklas är alltså till största del förfining av funktionaliteten. De befintliga operationerna förbättras och nya verktyg läggs till utifrån användarnas behov. För att systemet ska fungera på olika platser ingår även uppgradering som tillåter flyttning och drift på olika plattformar. Slutligen, och som även tidigare indikerat, utgörs vidareutvecklingen till en viss del även av korrigeringar utav fel.

## **4 Empiri**

### **4.1 Bakgrund**

Min empiriska studie har genomförts av mig själv på plats under en systemutvecklingsprocess vid ett ganska så stort företag i Paris med cirka 450 medarbetare. Företaget, Reuters Financial Software, utvecklar en produkt för finansmarknaden. Användare av produkten är till största del banker och dess handlare, Traders, Dealers, Brokers. För att lära mig produkten på en djup nivå var min roll på företaget först och främst att arbeta med support kring produkten, 3000Xtra som min empiri kommer att koncentrera sig på. Supporten var underhåll av en FAQ-databas samt besvarande av nya frågor från användare. Vidare undersökte jag olika möjligheter för sammanslagning utav två liknande produkter inom samma "produktfamilj". Jag har arbetat som assistent till en "product manager" som bland annat har som arbetsuppgift att just undersöka vad som ska vidareutvecklas och i sin tur sätta upp kravspecifikationer. Därför har jag haft en bra position från vilken jag har kunnat följa hur arbetet går till. Förutom att jag själv har sett hur allt går till har jag även vid flera tillfällen fått möjlighet att höra och se hur de andra runtomkring mig arbetar för att få en bredare och djupare förståelse. Som "product manager", eller produktansvarig, arbetar man nära utvecklare, användbarhetsspecialister, systemarkitekter med flera. Jag har då vid olika möten med dessa personer fått en inblick i vad de arbetar med i just vidareutvecklingen av produkten. För att få färdigt mina arbetsuppgifter har jag i stort sett varit beroende av att träffa medarbetare och ställa frågor.

### **4.2 Typ av system**

I den empiriska undersökningen betraktas ett system som är en produkt vilken utges av företaget. Produkten är alltså inget system som utvecklas på beställning utan finns sedan länge och kommer ut i nya versioner vart annat år ungefär och beror av ständig vidareutveckling. Mellan lanseringen av nya versioner kommer underversioner som är förbättringar och åtgärddar till exempel buggar. Produkten utges mer eller mindre i hela världen. Min praktikperiod på företaget började precis innan en ny version kom ut och planeringen inför nästa var redan igång sedan innan. Systemet är en stödtjänst åt finansvärlden. Det hämtar historisk och referensdata samt data i realtid om företag, organisationer, valuta m.m. Det kan handla om växelkurs eller till exempel aktiekurs var man direkt kan följa utvecklingen. Andra funktioner är exempelvis grafer till vilka man kan bestämma tidsintervall för att följa utvecklingen i form av ett diagram. Även nyheter kan fås i realtid, där nya artiklar fås i takt med publicering. Allt detta används slutligen, som av exempelvis banker eller investmentbolag, för att exempelvis beräkna ett optimalt pris. Systemet kommer i ett paket bestående av två delar. Den ena är en tilläggstjänst till Excel medan den andra är fristående. Tilläggstjänsten fungerar som så att en flik läggs till i Excel menyn från vilken man kan få fram data i realtid till sitt Excel blad, tjänsten utökar även funktionerna i Excel med en mängd analytiska beräkningsfunktioner. Den andra fristående delen är grafisk och tar fram data precis

som Excel-tjänsten förutom att det sker i ett grafiskt format. I den fristående delen finns det alltså ett antal olika objekt med funktioner och verktyg genom vilka man kan visa upp informationen man har hämtat. Objekten är graf, matris, nyheter, aktiekurs och en "ticker" som visar kurs i önskat intervall och tar fram kurs för alla önskade och börsnoterade organisationer. Systemet erbjuder en mängd funktioner och verktyg varför det är beroende av användbarhet då användarna kan vara allt ifrån systemets nybörjare till experter. I och med sin allsidighet används det av användare som helt självklart inte har samma arbetsuppgifter. Oftast utnyttjar de endast det som de har behov av till sina arbetsuppgifter och inget annat. Vad gäller tekniken är det självklart ganska komplext dessutom hämtas ju informationen från olika källor vilka ständigt behöver förnyas, uppdateras och underhållas. Varför även den biten är beroende av stor uppmärksamhet för att möta användarnas krav på tillförlitlighet. Systemet är också uppbyggt så att man söker data med hjälp av en kod som identifierar för vad man söker information. Det är koder för företag, organisationer, valuta m.m. I anslutning till systemet är användarna kan söka upp vilken kod som motsvarar det företag, den valuta, organisation etc. han eller hon vill söka efter.

### **4.3 Aktörer kring systemet**

För att utveckla systemet finns det olika grupper som arbetar runtomkring. Först och främst en ansvarig för olika delar i produkten men även uppdelade enligt region, då olika världsdelar har olika behov. En ansvarig för delar i produkten kan ansvara för realtidshantering, funktionalitet, analys och så vidare. Företaget har filialer i alla världsdelar samt i de flesta länder, alla dessa filialer har dock inte systemutveckling som funktion. Vidare finns det utvecklare som arbetar med själva utvecklingen, programmeringen. Det är de ansvariga som undersöker behov och tar fram krav inför nästa version. Eftersom företaget är stort har man även tillgång till användbarhetsspecialister, ekonomer och så vidare. Runtomkring produkten finns sedan en supportgrupp samt försäljningsspecialister och utbildare. Eftersom denna produkt är en utav många som utges finns det även avdelningar som arbetar med analyser och marknadsföring för att kunna få fram behov utifrån användning, statistik med mera. Resterande och de mest betydande aktörerna är alltså användare. Då systemet riktar sig mot "alla", med det vill jag mena att det går att köpa och att det inte är riktat mot någon särskild, används oftast inte alla funktioner utav varenda användare. Systemet köps in utav ett företag som ett verktyg åt dess arbetsuppgifter var de anställda använder de delar som stödjer deras område. När jag säger att systemet riktar sig mot alla menar jag att systemet inte utvecklas åt någon särskild, det köps givetvis av företag som har nytta av dess tjänster och verktyg, exempelvis banker.

## **4.4 Systemets behov**

Eftersom systemet kommer ut på de flesta kontinenterna till en mängd olika typer av företag måste det fungera på olika plattformar. Förutom plattformar ska det fungera med olika operativsystem och regioner. Programmet använder dock engelska som språk i alla sina verktyg.

Omkring systemet finns en dokumentering, support samt kundservice tillgänglig. Det är inte självklart att användare vet hur alla verktyg fungerar. Därför finns det användningsguider tillgängliga samt ett kundcenter. I kundcentrat kan de söka bland tidigare ställda frågor men även bidra med egna vilka eventuellt blir en FAQ. Genom kundcentrat möjliggörs interaktion med företagets support. Oftast är det dock så att en användare inte har tid eller tålamod för att gå igenom en guide och tar det enkla före det svåra när de kontaktar kundsupport. För att trots allt undvika en kundstorm mot supporten måste systemet naturligtvis vara användbart. Tanken med kundcentrat är just att förmå användarna att söka upp lösningar till problem för att inte behöva kontakta någon direkt.

För pålitligheten måste informationen som genereras via programmet vara korrekt och aktuell. Varför källorna även de måste vara korrekta och uppdaterade. Då de är grundstenen till produktens funktionalitet är dess kontinuerliga och oavbrutna tillgänglighet ytterst viktig. Användarna ser endast det de får fram och litar därmed på att de kan ta del av den data som systemet anger går att hämta. För användarna är källorna osynliga såsom antalet källor, vilka de är och så vidare. För att hålla förtroendet uppe är det de interna aktörerna, det vill säga de aktörer som arbetar kring produkten, som onekligen är i position att upprätthålla korrekthet, reliabilitet samt stabilitet för att användarna inte ska tappa förtroendet. De är medvetna om det mesta då de är informerade om användarnas frågor och utifrån frågorna den funktionalitet som skulle kunna fungera bättre, systemets brister, källornas brister, verktyg som används mer än andra och så vidare.

## **4.5 Indikationer till vidareutveckling**

### **4.5.1 Buggar**

Brister finner man alltid i ett system. Enligt dokumentationen är det oftast från början buggar man söker igenom för att korrigera. Detta sker genom testning som görs inför lanseringen. Dessa tester finner inte alla buggar utan många dyker även upp efter versionens lansering, vilket inte är konstigt. Buggarna korrigeras med en gång eller prioriteras efter viktighetsgrad. Man vill helst inte att buggar ska synas för användare, vilket är normalt, därför utför man dessa tester inför lanseringen. Subversioner som kommer ut kort efter huvudlanseringen korrigerar dock oftast dessa fel.

## 4.5.2 Användbarhet

När användarna kommunicerar med kundcentrat handlar det om användbarhet. Kundcentrat är låter användare lösa sina problem genom att söka svar bland tidigare ställda frågor (FAQ). Problem som kan uppstå vid användningen, funktionella problem. Hur man utför en viss kalkyl med programmet? Hur man räknar fram ett visst värde med hjälp av de analytiska funktionerna? Deras problem kan även vara väldigt grundläggande och kan då signalera om att användbarheten i systemet är begränsat. FAQ-databasen är organiserad och sorterad enligt version och produkt. Där kan man alltså se vilka funktioner som gäller genom flera versioner och även att en del funktionalitet har förändrats från version till version. Det är oftast att användbarheten förbättrats för att underlätta arbetsmomenten för användarna, det är således oftast det som syns för dem. De behöver alltså inte återkomma till samma problem om de använder en nyare version. Det är inte bara de enkla och grundläggande problemen som kan vara ett rop på bristande arbetsflöde även de mer komplicerade frågorna ger en effekt av att programmet inte är tillräckligt tydligt. Naturligtvis krävs det att många stöter på samma svårigheter för att man ska bedöma det som ett problem och eventuellt förbättra arbetsflödet. Det är dock alla typer av operationer som man strävar att förbättra efter återkommande problem hos användare. Reuters har avdelningar som har hand om statistik för just sökningarna i kundcentrat, både externt av kunder samt internt från Reuters support för att kvalitetssäkra samt identifiera frågor som genererar lyckade lösningar. En av mina huvuduppgifter var uppdatering av frågorna relaterade till det angivna systemet. Frågorna gällde den föregående versionen men krävde en uppdatering för den senaste. Genom arbetet fick jag ta del av vad som förändrats till den nya versionen vilket jag tidigare läst om i dokumenteringen.

## 4.5.3 Sökbarhet och data

Förutom användbarheten förväntar sig klienten att kunna få fram den information han eller hon önskar. Detta enligt de frågor som ställts mot kundcentrat. När man utvecklar ett system kan man omöjligt föreställa sig alla typer av scenario och därmed kan man oavsiktligt låta utebli något som en användare förväntar sig. Det kan då handla om data och information som denne söker efter. Slutligen leder det till att man utökar sökbarheten för att klienten på ett enklare sätt ska kunna komma åt det han eller hon söker efter.

## 4.5.4 Aktörerna och systemet

Det som just påverkar systemet och dess förändringar är dess aktörer. Förutom aktörerna finns en omvärld kring systemet och dess aktörer som ständigt förändras och förnyas. Därmed förnyas och förändras även kraven på hur systemet ska tillfredsställa användarnas behov eftersom finansvärlden, i detta fall, förändras ständigt. Systemets förmåga att hämta information i realtid är just en duglighet som möter världens föränderlighet.

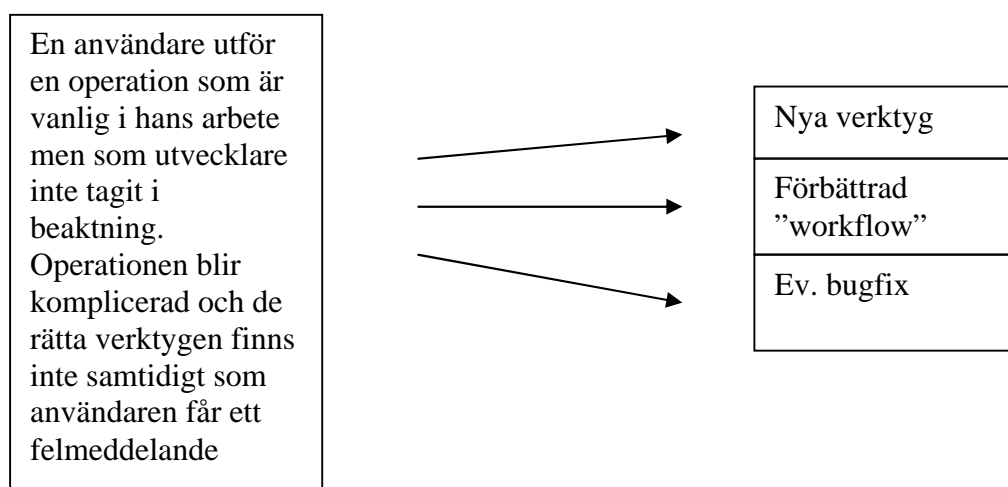
Systemets viktigaste aktörer visar ett resultat av omvärldens påverkan då det slutligen är dem som kommunicerar fram sina förändrade behov. Efter exempelvis en enkät kan man bland annat få fram önskemål om information som för tillfället inte kan tillgås av systemet. Användarna kan likaså genom kundcentrat kommunicera fram sina svårigheter och signalera åt utvecklarna var arbetsprocesserna är invecklade. De produktansvariga kommunicerar även direkt med kunder med hjälp av exempelvis enkäter och besök.

## 4.6 *Hur man identifierar systemets brister*

### 4.6.1 **Betatest**

Enligt företagets dokumentation fanns det inför lanseringen av förra versionen en betaversion under några månader. Denna fungerade precis som tidigare beskrivits i teorin. Den låter alltså användarna prova på den nya versionen i väntan på den fullständiga. De kunde på så sätt interagera och ge sina synpunkter mot det fullständiga systemet. Interaktionen skedde som så att användare fick vända sig till supporten med sina problem.

Betatest



FIGUR 5: ORSAK-EFFEKT MATRIS FÖR BETATEST

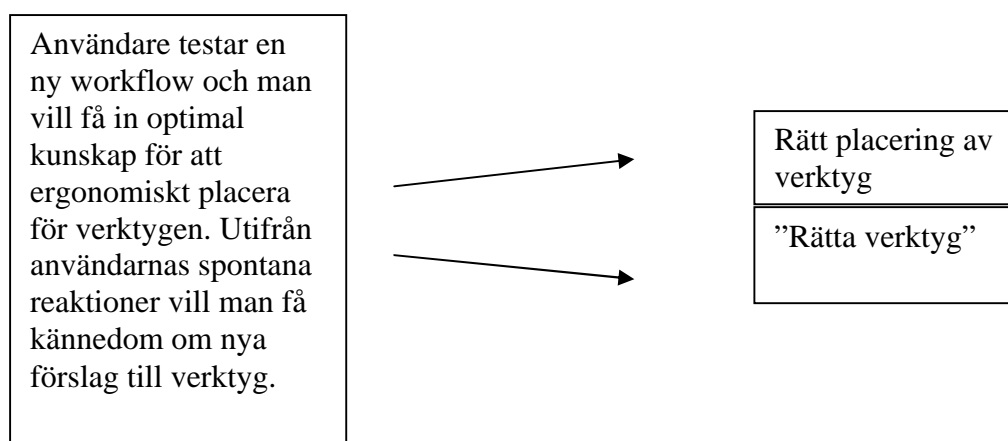
Figuren som använder en orsak-effektmatris vill visa på hur ett betatest kan bidra till vissa effekter. Alltså en användare som testar betaversionen och som genom sina synpunkter bidrar till effekter i systemets vidareutveckling.

### 4.6.2 **Användbarhetstest**

Enligt användbarhetsspecialisterna utför man användbarhetstest för att konkret testa systemets användbarhet gentemot dess användare. Tester utförs i grupper av användare på 30-40 personer. Antalet beror på intresse samt test. Användarna får

testa funktionalitet på exempelvis en prototyp som inte alls behöver vara särskilt utvecklad. En typ är någon form av PowerPoint-variant endast för att visa den tänkta designen och testa den inför utveckling. Idén är främst att sätta användaren inför en planerad design och studera hur denne orienterar sig. Även papper och penna är ett verktyg för att låta användarna rita hur de tycker menyn ska vara uppbyggd. Man försöker så mycket som möjligt blanda nivån på användarna beträffande datorvana, kännedom om programmet och så vidare. Resultatet på observationerna, inspelningarna samt noteringarna fungerar som en bas för vidareutveckling av eventuellt identifierade svårigheter som kräver en ändring. Testet indikerar även på om något i designen inte alls fungerar och kräver ny planering.

#### Användbarhetstest *inför* utveckling



FIGUR 6: ORSAK-EFFEKT MATRIS FÖR ANVÄNDARTEST

Figuren använder en orsak-effekt matris för att visa effekter utifrån ett användartest.

### 4.6.3 Enkäter och intervjuer

Företagets dokumentation visar att man använder sig av enkäter som en kommunikationsmetod gentemot användare. Enkäten skickas ut till ett antal användare som får svara på frågor. Fokuset för enkätens tema är användbarhet samt funktionalitet men även det som betraktar dokumentation, guider, modeller, handledning och liknande kring produkten. I sin tur delar man upp denna fokus över produkten och går igenom systemets delar var för sig. Förutom enkätmetoden insamlas information via intervjuer genomförda vid verkliga besök. Ett verkligt besök är då en produktansvarig besöker en användare personligen. Under intervjun får användaren tillfälle att uttrycka sig och mer utförligt dela med sig av sina synpunkter. Användarna får delge de områden som mest frekvent används, vad som



kan saknas, synpunkter på förändring och så vidare. För att på något sätt klargöra bilden som illustreras med hjälp av enkäterna och intervjuerna tar man reda på användarens nivå. Nivån kan förklara en hel del om användarens svar. Exempelvis kan en ovan användare indikera på vissa svårigheter i användningen som en van användare redan känner till och därmed inte uppfattar som lika svårt. Resultatet medför hur användbarheten kan förbättras. Om responsen skiljer beroende på användarnas nivå där exempelvis ovan användare ofta finner något besvärligt men "medelanvändarna" inte ser det som ett problem kan det innebära att systemet trots allt inte är komplicerat att använda. Dessa enkäter och intervjuer utförs eller skickas inte bara för betaversioner utan även för riktiga versioner då det ligger i de ansvarigas intresse att få in användarnas åsikter. Man vill fortsättningsvis avancera i sin strävan för förbättring och utveckling till nästkommande version av produkten.

#### **4.6.4 Besök**

Företagets produktansvarige och utvecklare utför besök hos användare. Med sig för de ett protokoll med bland annat användbarhet som en punkt. Protokollet är framtaget inför besöket och är förberett med frågor samt ämnen som är tänkta att tas upp. Protokollet fylls i av produktansvarige eller utvecklare utifrån de intryck och svar som lämnas av användarna. Information som vana, hur användaren söker hjälp, vilken hjälp han eller hon redan har fått, vilket ändamål de har för produkten samt vad användaren utnyttjar i sitt dagliga arbete helt konkret är en del av protokollets signifikans. Slutligen samlar de in kommentarer på förbättring som en del av sin studie. Svaren resulterar i vidare undersökningar och, om möjligt, förbättring samt vidareutveckling. Ett konkret exempel är en användare som tyckte fält från databasen var svåra att hitta. Motresponsen i en senare version var bättre exponering i sökassistenten av just detta för att underlätta arbetsflödet i sökandet. Allt detta dokumenteras sedan i anslutning till produkten.

#### **4.6.5 Konkurrens**

Efter samarbete och samtal med medarbetare fick jag det uppenbart för mig att Reuters jämför sig ofta med sin största konkurrent Bloomberg. Användare av Bloomborgs motsvarighet har många gånger även Reuters system och kan komma med kommentarer som att hur man kan göra den och den operationen från Bloomberg med Reuters system. Detta är naturligtvis en fråga som direkt berör och man försöker komma ut med nya verktyg för att möta konkurrensen.

### **4.7 Vad förändras och vidareutvecklas**

Enligt broschyrer samt dokumentation är de förändringar som först och främst syns för användare funktionalitet och användbarhet. Utifrån mina arbetsuppgifter har jag uppfattat att det från version till version alltid finns förändringar vad gäller dessa två områden. För att underlätta användbarheten integrerar man de två produkterna

så mycket som möjligt. Eftersom de ingår i ett paket är tanken att de även ska kunna fungera ihop. Då har det till den senaste versionen lagts till verktyg och symboler som ser likadana ut för att undvika förvirring. Många användares frågor till kundcentrat handlar just om hur man använder produkterna "kors och tvärs". Förut gick inte det helt och hållet medan det nu går. Man kan i det fristående programmet öppna upp data som man sedan kan kopiera och klistra in i Excel och få realtidshämtningen att fungera precis som vanligt. Det är en av de stora förbättringarna som har efterfrågats utav användare då det just finns en hel del ämnen kring det i kundcentrats FAQ-databas. En efterfrågan på att kunna spara sina tidigare objekt, såsom exempelvis en aktiebörsmatris, har tidigare inte kunnat göras men har utifrån kundbehov utvecklats för att ytterligare förbättra användandet. Filerna sparas i en mapp särskild för filer skapade genom företagets produkter och tillgås därmed genom båda delarna i produktpaketet. Kundernas önskemål leder oftast till förbättringar inom användbarheten och funktionaliteten det finns dock önskemål som leder till teknisk förbättring. Man har utvecklat systemet som så att företagen kan använda sina egna applikationer för att kommunicera med källorna i sitt arbete med systemet. Detta har alltså lett till tekniska förändringar för att förverkligas.

Min erfarenhet pekar också på att de största förändringarna kommer utifrån krav som väcks genom de ansvariga för olika delar i produkten. Deras kunskaper och arbetsuppgifter utgör studier för att realisera vilka krav som kommer ifrån just omvärlden. Det behöver därmed inte vara direkta krav från användare utan baseras helt och hållet på studier av omgivningen som konkretiserar vilka behov och förbättringar som skulle kunna sättas upp som krav inför nästa version. Deras arbete utgörs bland annat just av samtal med de övriga aktörerna runt systemet; försäljare, utbildare med flera.

En viktig faktor som måste anmärkas är att denna produkt, som tidigare nämnt, är beroende av sina källor. Dessa källor är inte fria utan tillgås med hjälp av avtal och licenser med leverantörer. Därför är det i allas intresse viktigt att de kostnader detta medför hålls låga och förmånliga för att inte resultera i en alldeles för dyr produkt. Enligt dokumentationen har det till den senaste versionen lagts till och ändrats vad gäller tillgången till källor. Vissa källor köps dessutom upp utav det själva företaget. Den senaste versionen har alltså utökats och tillgång framförallt bredare och djupare fundamental och referensdata, data som även möjliggjorts genom ny serverteknologi vilken kan tillgå fler källor och leverera data smidigare till användaren.

En annan signifikant förbättring är öppenheten som låter användarna utveckla egna applikationer och hämta data från källorna. Detta är fördelaktigt för de avancerade användarna som kanske önskar utveckla ett eget gränssnitt för att visa data som relaterar till deras behov.

Vidare förbättringar baserade på de produktansvarigas begär är utökade verktyg samt förenkling vad gäller installationsprocessen. Det komplexa systemet kräver användarlicens för att koppla upp sig via olika tekniker och till olika baser finns det ett verktyg för inställningar. Detta verktyg har förbättrats så att man kan gå in och ändra inställningar som gjorts automatiskt under installation. I verktyget kan man se

sina användaruppgifter samt alla installerade komponenter som krävs för applikationen. Förenklingen har förbättrat verktyget både vad beträffar användningen samt funktionaliteten. Den erbjuder fler möjligheter till förenklad och utökad hantering samt underhåll av applikationen åt användarna.

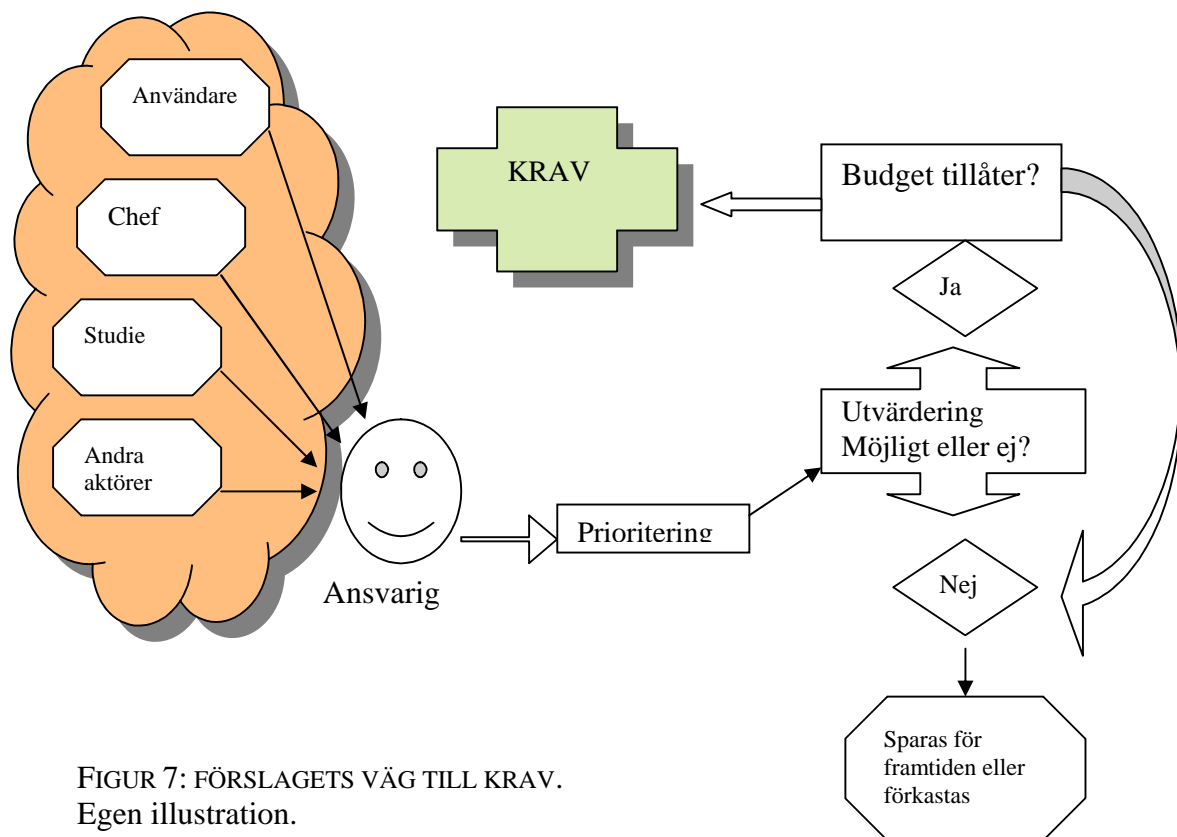
Ett ökat behov hos professionella investerare som är beroende av diagram för sina tekniska analyser har försatts med ett bättre verktyg som helt och hållet går att skraddarsy efter behov. Olika design har lagts till för att förenkla avläsningen av diagrammet och medför ett mer precist resultat i användarens arbete. För att användarna vid uppgradering inte ska behöva förlora någon information har man möjliggjort migrering utav filer från föregående version. Detta är då självklart ett resultat av produktansvarigas krav och ökar produktens konkurrenskraft på marknaden.

Eftersom systemet söker på olika identifieringskoder, kända för affärsvärlden, kan man inte förvänta sig att användarna känner till alla dessa. Som tidigare nämnt. Till den senaste versionen har det då utvecklats en ny och fullständig sökfunktion med ett webbgränssnitt. Ur användarperspektiv är det en stor förbättring som tillåter kunderna att söka helt fritt, och inte bara efter koder.

All kunskap om förändringar grundar sig på dokumentering samt den erfarenhet jag fått genom mina arbetsuppgifter. Erfarenheten grundar sig till största del på arbetet gentemot det webbaserade kundcentrat.

## **4.8 Begränsningar**

Det är de produktansvariga som har ansvar för att sätta upp nya krav för sin del av systemet. Naturligtvis kan inte alla krav accepteras och endast de viktigaste faller inom budget. Budget och en del krav på förändringar kommer ”uppifrån” i företagets hierarki. När de produktansvariga sätter upp de krav som de enligt sina studier, kundkrav och eventuellt från beslut utav den högre chefen kommit fram till måste de studeras vidare för att slutligen godkännas. De som godkänner är de högre ansvariga; försäljningsexperter, budgetansvariga, utvecklingsansvariga osv. Först och främst är prioriteringsfrågan principiell. Sedan måste man vara överens om att det är tekniskt möjligt, ibland har användbarhetsexperterna visioner och krav vilka koncentrerar sig alldeles för mycket på användbarhet där andra faktorer, såsom systemets kraftfullhet, hamnar åsido. Det kan finnas krav och önskemål om en hel del förbättringar som helt enkelt är omöjliga eller innebär så pass mycket arbete att man i slutändan inte tjänar något på det. För att komma fram till ett tekniskt beslut utvärderas de tänkta kraven med utvecklingsgruppen. Förslagen som sedan faller inom budget specificeras ytterligare och prioriteras därefter. Det går till som så att produktansvarige specificerar förslaget som sedan utvärderas och prioriteras av utvecklingsgrupp/ansvarig innan det realiseras. Dessa krav tar tid att både ta fram och sedan utveckla då detta framförallt signaleras av versionernas intervall på ett till två års mellanrum mellan lanseringarna. Kunskaperna baseras på dokumentering samt mina arbetsuppgifter i samarbete med de produktansvariga.



FIGUR 7: FÖRSLAGETS VÄG TILL KRAV.  
Egen illustration.

#### 4.8.1 Vad mer tillförs till följd av indikationerna

Återkopplingen och indikationerna som kommer in till följd av studier, enkäter, intervjuer, tester, kundcentra och så vidare resulterar inte med en gång i en vidareutveckling. Efter diskussioner och djupare studier resulterar ämnet inte alltid i ett krav till nästa version. Det kan dock finnas kvar och hamna under en lägre prioritering för tillfället. Ett förslag som inte gått igenom till ett krav slutar inte för den sakens skull i bortfall. Det bäddar samtidigt för planering och budgetering med längre sikt och kan resultera i ett krav för en version längre fram. Även här har nära samarbete samt dokumentation påvisat kunskaperna.

#### 4.9 Sammanfattning

Inför en lansering kan systemet ha brister. Därför utför man tester för att lokalisera samt korrigera buggar. Testerna finner inte alltid buggarna och en version lanseras ändå. Därefter rapporteras buggar allteftersom av användare samt utvecklare. Även med hjälp av betatestning kan buggar och användbarhet testas inför en lansering. Betatestningen tillåter systemet att prövas för verkliga och spontana operationer. Dessa korrigeras efter prioritering, tyvärr korrigeras inte alla på en gång. Andra brister kan förekomma inom användbarheten samt funktionaliteten. Dessa brister identifieras av både utvecklare samt användare. Användarna har möjligheten att

kontakta utvecklare via kundcentrat för att kommunicera fram sina synpunkter. Utvecklare gör även enkäter samt utför intervjuer för att få användarnas återkoppling. Alla andra tänkbara områden för vidareutveckling, såsom teknik, identifieras av utvecklare efter diverse studier. Innan en ny funktionalitet, teknik eller bugg ska realiseras eller korrigeras måste den bli ett krav. Ett krav specificerar vad som ska vidareutvecklas och utvärderas gemensamt av berörda parter i utvecklingsgrupperna. Bakom kravet finns även en viss studie och undersökning som ligger till grund och styrker behovet. För att ett krav sedan ska realiseras och slutligen utvecklas studerar man den tekniska möjligheten för att realisera detta samt utvärderar budget. Utifrån användbarhetstester samt enkäter vidareutvecklas sådant som underlättar användningen. Det kan vara grafiska utseendet, placering av verktyg och arbetsflöde. Utifrån att en bugg identifieras förebyggs den efter sin karaktär, oftast är det dock en teknisk förändring då en bugg genererar ett fel i användningen. Efter studier från produktansvariga vidareutvecklas sådant som förbättrar kapacitet såsom nya källor, ny teknologi, ny arkitektur etc.

## 5 Resultat

Matrisen i bilaga 1 utgör grund för resultatet och innehåller en jämförelse mellan litteraturstudien samt empirin.

### 5.1 *Faktorer mot framtida utveckling*

Flensburg och Friis (1999), Reynolds (1995) m.fl. talar om skavanker och missar i utvecklingen som resulterar i fel i en färdig produkt. Detta bekräftas genom dokumentationen till det undersökta systemet samt faktumet att det kommer ut i nya versioner. Systemet i observationen är inte något system som utvecklas på beställning utan kommer ut i versioner och säljs därefter. Versionerna samt dess underversioner åtgärdar buggar och missar samt förbättrar användbarheten. Trots att en del av teorierna kommer från äldre källor bevisas det att man fortfarande begår fel och misstag. Reynolds (1995) teori om att förändringar i omvärlden ställer nya krav och förändrar behoven som tillfredsställs av systemet kan rättfärdigas med empirin för just detta system. Det studerade systemet genererar aktuell och realtidsinformation vilket ständigt kräver uppmärksamhet. Man underhåller detta krav genom uppdatering och utökning av databaser. Davis och Yen (1999) olika typer av underhåll, adaptivt, perfektivt samt korrektionsunderhåll motsvarar empirins verklighet. Varje ny version kommer ut med nya eller förbättrade funktioner efter nya behov. Det adaptiva, det vill säga konverteringar och uppgraderingar förändras även de då installations- och administrationsverktygen förbättrats till den senaste versionen samt då "bakomarbetet" resulterat i nya typer av databaskopplingar för tillgång till fler källor. Korrektionsunderhållet är det som man i stort sett befinner sig i kontinuerligt enligt empirin då buggar och fel hittas och korrigeras. Å andra sidan kommer korrektionerna till buggarna ut i underversionerna och implementeras alltså inte allteftersom som de korrigeras. I teorin nämns generellt endast utvecklare och användare och de utgör därmed kommunikationspelarna för att framhäva alla dessa indikationer. I den empiriska studien visas att utvecklare, i denna undersökning, utgörs av många personer. Det vill säga alla aktörer som tidigare nämnts. De behöver inte ha en direkt "utvecklarroll" utan det räcker med dess efterarbete kring produkten för att deras inverkan kan falla inom ramen för indikationerna mot utveckling. En som utbildar användare är minst lika viktig i vidareutvecklingen som en programmerare då utbildaren kommer i kontakt med användarna och därmed identifierar eventuella svårigheter.

## **5.2 Metoder för identifiering av systemets brister**

### **5.2.1 Betatest**

Ett betatest är enligt litteraturen en beskrivning på hur det fungerar och motsvarar väldigt väl ett betatest i verkligheten. Precis som i teorin utvecklas en betaversion innan lansering för att kunna testa den och därigenom identifiera fel. Givetvis finner man inte alla fel och produkten lanseras trots allt.

### **5.2.2 Usabilitytest**

Även usabilitytestet stämmer i stort sett överens i empiriska undersökningen som i den teoretiska. Även här beskriver teorierna hur ett usabilitytest går till vilket i hög grad tillämpas i verkligheten. På företaget har man samma mål för ett användbarhetstest som i litteraturen, det vill säga att slutligen förbättra produkten. Precis som Nielsen (1999) understryker vänder sig testet mycket riktigt mot riktiga användare och man strävar efter stor spridning för bred representation. Riktiga och förutplanerade operationer undersöks på exempelvis en prototyp eller papper och penna. Användbarhetstestets resultat används sedan som en del av grunderna till framtida planering precis som Dumas och Redish (1999) påpekar.

### **5.2.3 Enkäter och intervjuer**

Litteraturen och empirin stämmer överens när det kommer till enkätens fördelar. Det vill säga att det är en kostnadseffektiv metod som når ut till många. Enligt företagets dokumenterade enkät stämmer det även att svaren man får är väldigt detaljfattiga och ibland svårtolkade. Empirin och teorierna skiljer sig åt beträffande utformningen av enkäten. Davis och Yen (1999) ger riktlinjer på hur en enkät bör utformas vad gäller språk, frågetyper med mera stämmer empirins verklighet inte riktigt överens där. Företagets enkät är inte särskilt djup eller välutformad. De essentiella hos utvecklarna är förmodligen att mest få in information effektivt. Enkäten frågar mest på användningen och exempelvis vilka verktyg användarna väljer att utnyttja då de utför sitt dagliga arbete. Språket är däremot enkelt och precist. Sammanfattningsvis använder företaget enkätens positiva sidor genom att kostnadseffektivt nå många och kunna precisera de frågor man vill ha svar på. De har däremot "kortat" ner sitt arbete i utformningen. Svaren är dock oftast förståeliga och respondenterna är villiga att ge sina synpunkter.

Intervjuerna har enkäternas nackdelar som fördel. Man kan säga att företaget använder sig av intervjuer som ett komplement. Deras intervjuer är, till skillnad från enkäterna, mer välutformade. Man går igenom respondentens vanor, synpunkter och sist och inte minst får respondenten en möjlighet att ge förslag till förändringar.

## **5.3            *Brister på identifieringsmetoderna***

Metoderna som omskrivits i teorin motsvarar de metoder som används av företaget. På något sätt förebyggs den ena metodens nackdelar med hjälp av en annan metod och dess motsatta fördelar.

Först och främst riskerar ett betatest att ”struntas” i av sina respondenter (Dumas & Redish, 1999). Denna brist förebyggs av företaget genom de andra metoderna. Ett användbarhetstest har användare fysiskt närvarande tillsammans med utvecklarna. En enkät är direkt adresserad till en respondent och ännu starkare riktad vid en intervju. Ett betatest är trots allt viktigt då operationer som man tidigare inte förutspått testas av användarna, skulle de då låta bli att lämna sina synpunkter kan man gå miste om viktig återkoppling.

Den ständigt återkommande nackdelen är representationen på respondenterna och dess motsvarighet till den verkliga användningen. Företaget förebygger enkätens detaljfattigdom med hjälp av intervjuer för att täcka över detaljerna. Metoden används dock inte i lika stor utsträckning som enkäten då den, som även nämnt i litteraturen, är mer kostsam. Enkäten kan även ses som ett komplement till användbarhets- och betatestens brister. Då dessa två sistnämnda metoder kan brista i representation och antal har men med en enkät möjlighet att få upp antalet respondenter. Enkäten har dessutom en funktion som ett komplement till betatestet då den har som avsikt att få in synpunkter och åsikter efter att just en betaversion har blivit lanserad och på så sätt få upp antalet medverkande.

## **5.4            *Vad förändras och vidareutvecklas***

### **5.4.1        *Beta- och användbarhetstest***

Eftersom teorierna talar för utveckling av generella system redogörs endast empirins resultat för. Testmetoderna som dock tagits upp teoretiskt talar sitt tydliga språk i metodernas målsättning om vad som ämnas testas och vilka ändringar det kan medföra. Beta och användbarhetstester medförde funktionella och användbarhetsförändringar. Eftersom de funktionellt testar antingen användbarhet på en prototyp i ett användbarhetstest eller en ofullständig version i ett betatest bidrar det till vidareutveckling i just det området. En användare som exempelvis uttryckt sitt önskemål om en viss analytisk funktion i Excelvarianten har fått sin önskan uppfylld då dessa funktioner utökas och utvecklas kontinuerligt efter behov. Ett användbarhetstest som testar en användare och en prototyp kan observera användarens tendens till att söka efter verktyg. Utifrån det utvinns man en idé om optimal och ergonomisk placering för de olika verktygen till produkten. Det kan handla om testning av något så grundligt som kryssknappen som avslutar programmet eller endast stänger dokumentet man arbetar med. För att undvika förväxling mellan de två vilket kan föra med sig stora konsekvenser. Ett betatest kan upptäcka buggar och fel såsom få respons på användbarhet samt funktionalitet. Då man åtgärdar buggar och fel leder det till tekniska förändringar vilket innebär att ett betatest inte bara medför vidareutveckling i funktionalitet. En bugg kan generera



en krasch och tvingar utvecklarna till tekniska förändringar. Vad beträffar funktionaliteten och verktygen får man indikationer och utvecklar områden vilka man kanske inte har betraktat som ett eventuellt problem. Som tidigare nämnt är det en av skillnaderna mellan betatest och användbarhetstest.

#### **5.4.2 Enkäter och Intervjuer**

Enkäter och intervjuer ställer frågor apropå användning, arbetsflöde, vana och funktionalitet. Resultatet medför synpunkter beträffande förändringar som påverkar funktionaliteten. Användarna frågas vad de utnyttjar mest, vad de inte utnyttjar alls, hur de använder verktygen och så vidare. Rent metodiskt uppfyller enkäter och intervjuer sitt teoretiska mål då de får in återkoppling som bidrar till vidare studier och utveckling på funktionalitet. Företaget har även förslag och önskemål från användare som en del av enkäten eller intervjun. Användare önskar då oftast nya verktyg, såsom exempelvis nya formler för finansiell kalkylering eller disposition av verktyg, samt nya verktyg. Andra önskemål innefattar förändringar i arbetsflödet. Detta resulterar i sin tur i effektivisering beträffande produktens arbetsflöde. Det som användare uttrycker i ett önskemål brukar oftast uppfyllas då det ökar produktens konkurrens kraft. Användarna ber många gånger om funktioner som de har sett i en konkurrents produkt vilket är ett starkt motiv för vidareutveckling.

#### **5.4.3 Studier och kundkontakt**

Precis som Davis och Yen (1999) talar om som form av analys metod studeras konkurrenterna väldigt frekvent. Vid sidan om det måste man ju även hålla sig uppdaterad om vad som händer i världen och låta systemet möta förändrade krav enligt Reynolds teori. Detta utförs av produktansvariga. Förutom studier av omvärlden besöker utvecklarna även användarna. Detta kanske ses som individuellt och uppfyller inte någon direkt teori men kan trots allt jämföras med intervjutekniken. Det som förändras utifrån studier är utökade källor vilka utökar den tillgängliga informationen. Och precis som ovan nämnt vad gäller konkurrensen utför de produktansvariga även studier mot konkurrenterna för att vidareutveckla sådant som kan "saknas".

### **5.5 Begränsningar**

Begränsningar finns alltid och man är inte kapabel till att utveckla ett "drömsystem" som kan innehålla allt vad en användare kan önska sig och på samma gång vara enkelt att använda. Begränsningarna som inte låter utvecklare att få allt genomfört är väldigt individuella och de kan inte jämföras med någon teori då teorierna oftast talar om generella system. Just i detta fall är budget en stor begränsning. Då systemet säljs versionsvis får man arbeta mot att hålla kostnaderna nere. För varje version avsätts en budget och tanken är att utvecklingen ska hållas inom den ramen. Nya krav måste vara starka och motsvara stora behov för att kunna placeras under

budgeten och börja vidareutvecklas. Inför budgetplaneringen avgörs även teknisk möjlighet vilken även den är väldigt individuell. För just detta system kanske man behöver se till att välja källor då man inte kan använda alla källor i hela världen, både vad gäller teknik samt kostnad.

.

## 6 Diskussion

Till att börja med kan man påstå att faktorerna som leder till motiv för framtida vidareutveckling stämmer överens generellt i teori samt empiri. När Reynolds (1995) tar upp vilka typer av underhåll som tillämpas i systemutveckling stämmer det väldigt bra överens med det verkliga fallet. Även faktumet om fel och skavanker i system (Flensburg & Friis, 1999) kan stämma väl med verkligheten. Ny- och vidareutveckling av verktyg och funktioner, uppdatering samt konvertering och korrigering är precis vad som händer vid vidareutveckling. Hilbert och Redmiles (2001) påstående om förväntningar inför utvecklingen till slut förändras utifrån användares återkoppling förstärks med empirin.

Metoderna för interaktion med användare är även de helt överensstämda genom teori och empiri. I den empiriska studien iakttogs att betatestning används frekvent och kompletteras med användbarhetstest för att träffa användare. Nästa kundkontaktmetoder är intervju och enkät vilket även dem sågs användas. Man kanske kan tänka sig att dessa teorier som ju kommer från generella "handböcker" för systemutveckling har gått igenom många utbildningar och har därmed resulterat i verklig användning vid systemutveckling. Något som dock inte stämde helt var enkäten. I verkligheten, för denna studie, var den inte alls utvecklad efter teorin. Mycket kan bero på att man vill dra nytta av enkätens fördelar och låter bli att lägga ner alldeles för mycket tid på den. Enkäten används i större utsträckning och tydligen mest för att dra nytta av dess fördelar. Davis och Yens (1999) indikationer på hur en enkät kan utformas, frågornas karaktär och placering som påverkan på svaret verkar inte ha varit lika viktigt. Jag misstänker att det är en tidskrävande process att bara utföra en enkät. Om man sedan djupare ska tänka på alla detaljer blir enkätens fördelar kanske inte tillräckligt starka för att man ska ha nytta av att genomföra en sådan undersökning. Vad sedan gäller studier kan de till en viss del stämma överens med Reynolds (1995) men detta är ju väldigt individuellt för systemet. Exempelvis är valutakursens förändring väldigt essentiell för det studerade systemet vilket kräver studier i det området medan det för andra system krävs studier i helt andra områden. Vidare beror analysområdet även på typen av slutanvändare. Som utvecklare måste man känna till deras arbete och arbetsuppgifter som kan ingå i studieområdet för framtida utveckling.

Dumas och Redishs (1999), Hilbert och Redmiles (2001) samt Davis och Yens (1999) beskrivningar på brister i metoderna återspeglas också i verkligheten. Bara faktumet att man använder flera metoder kanske talar om att man försöker förebygga bristen hos den ena metoden med att tillämpa den andra. Trots att man kanske försöker få reda på respondentens nivå är det fortfarande han eller hon som svarar och bedömer sig själv. Denna brist är svår att gå om oavsett vilken metod man använder. Kanske är användbarhetstestet närmast. Där kan utvecklaren själv se användaren och hur denne fungerar med systemet och då samtidigt bedöma nivån. Metoden är trots detta begränsad då man inte kan testa oändligt många användare. Ett betatest kan kompletteras med ett användbarhetstest ur operationernas synvinkel. Man kanske kan tro att ett användbarhetstest som är mer strukturerat och övervakat har mycket bättre effekt. Samtidigt får man inte glömma att utvecklare

kan ha missbedömt systemets användning och operationer som man inte har räknat med från början bidrar till fel. Särskilt i utveckling av system som kommer ut i versioner kan det bli svårt att med säkerhet definiera alla användares behov. Skulle det varit ett system på beställning har man kontakt med beställaren och kan koncentrera sig på just dennes behov och utföra mer detaljerade tester. Ett betatest får ta emot oväntade operationer och kan dirigera om utvecklaren som blir tvungen att prioritera nya behov.

Studier har ju tydligen inte bara användaren som sitt enda fokus. När man studerar sina konkurrenter är användaren visserligen en anledning men hamnar då i passivt fokus. Det är ju trots allt användarna som jämför konkurrenterna emellan och förväntar sig samma verktyg och funktioner. För Reuters Financial Softwares del är konkurrensen väldigt viktig. Då de är nummer två på marknaden i sin bransch, inom edition av mjukvara för finansmarknaden, är de väldigt måna om att bli bättre än sin största konkurrent. Det är inte ovanligt att man studerar konkurrentens nya produkter, dess dokumentation och lyssnar på deras användare. Som samtidigt också kan vara ens egna användare.

Det som slutligen vidareutvecklas speglas väldigt mycket av teorierna särskilt beskrivningen av vattenfallsmodellens sista steg, underhåll. Att sedan Reynolds (1995) även påstår att anledningen till vidareutvecklingen till stor del grundas på förändringar i omvärlden förstärks av empirin. Metodernas mål enligt teorin poängterar även de vad de ämnar vidareutveckla vilket även detta understryks av empirin. Enligt litteraturen är ju målen att testa användbarhet och funktionalitet vilket sedan i verkligheten vidareutvecklas. Naturligtvis går det inte att trycka på detaljer då de är individuella och beror på själva systemet. I just det undersökta systemet förbättrar och vidareutvecklar man funktionaliteten och verktygen efter användares yttrande. Anledningen till att man även använder sig av andra metoder, såsom studier av omvärld eller tekniska tester (som ej beaktats i denna undersökning) kan vara att man inte vill lägga alldeles för mycket förtroende till användare. De förfogar förmodligen inte över den relevanta kompetensen som krävs för att kunna bedöma framtida faktorer för vidareutveckling.

## **6.1 Förslag till vidare forskning**

Underhåll av system kan ses som ett stort kapitel i utvecklingen. Det finns onekligen redan en hel del forskning inom ämnet. Det som skulle ha kunnat göras utifrån min studie är att kanske göra en annan liknande studie men mer omfattande. Den empiriska studien skulle ha kunnat vara mer komplex och studera mer än ett system. För att utföra en sådan studie behöver man kanske fler forskare.

För att gå över på andra sidan, det vill säga användarnas sida, kan man förslagsvis följa vidareutvecklingen utifrån deras perspektiv. Man kan undersöka hur representationen vid testning påverkas utifrån deras nivå och ambition. Eller varför inte studera deras belåtenhet samt mäta i vilken grad de anser sig vara medräknade i utvecklingen.

## 6.2 Reflektioner

Jag tycker att min studie har visat mig vad vår förberedande kurs inom forskningsmetodik (VUM) egentligen ville uttrycka. Under kursens gång verkade en studie så självklar och att massvis med metoder kunde tillgås och dras nytta av. Med hjälp av alla möjligheter trodde jag att en studie kunde planeras och utföras enklare än det sedan visade sig vara. Nivån mellan en kandidat och magisteruppsats blev allt mer klar för mig efterhand som jag gjorde min undersökning. Kandidatuppsatsen var min enda konkreta "forskningserfarenhet" inför denna studie. Eftersom mitt kandidatarbete var mycket mer praktiskt och skiljde sig väldigt från denna kunde jag inte dra mycket nytta därifrån. Efter all läsning av metodlitteratur som tillsynes kunde ge ett så logiskt och simpelt intryck visade den praktiska upplevelsen något helt annat. En studie kräver så mycket mer planering samt lika mycket, om än mer, kvalitéer hos en forskare för att kunna hantera plötsliga förändringar i planeringen.

I början var det svårt att hitta ett ämne. Detta främst eftersom mina intentioner ville kombinera den erfarenhet jag skulle få på den planerade praktikplatsen med min undersökning. Allt detta för att tjäna in så mycket tid som möjligt. Min praktikplats utspelade sig på heltid vilket inte bidrog till mycket tid över för annat under veckodagarna. När jag väl fann det ämne som då skulle kunna kombinera mina två uppgifter insåg jag efter ett tag att mitt problemområde var alldeles för snävt och begränsat både vad gäller empiri men framförallt teori. Mina erfarenheter som forskare fick sättas på prov då jag stod inför en förändring av problemområde och mitt i en fallstudie vilket jag för övrigt gjorde för första gången. Att jag även skrev ensam medveten om att min tid på företaget var begränsad underlättade inte situationen. Jag öppnade upp min problemformulering vilken i sin tur även den bidrog till bredare tillgång till relevant litteratur.

Allteftersom jag arbetade på företaget insåg jag att jag lärde mig mer och mer vilket även bidrog till att jag snabbt kunde fatta beslutet att inte genomföra intervjuer. Då min studie baserades på en sex månader lång erfarenhet ansåg jag mig vara tillräckligt informerad själv. Intervjuandet skulle även ta lång tid, framförallt med tanke på transkriberingsarbetet, vilket slutligen skulle ha resulterat i en undersökning utöver tio akademiska poäng. Hursomhelst har min erfarenhet under sex månader inte bara berikats praktiskt på en praktikplats utan även som forskare. Att utan erfarenhet dessutom först och främst sättas på verkliga arbetsuppgifter och sedan utföra en akademisk studie i samma veva begärde stora krav. Eftersom jag ändå kunde kombinera mina två uppgifter var arbetet från båda hållen intressantare och på så sätt mer konstruktivt. I takt med att jag lärde mig mer på företaget kunde jag samtidigt ge svar och berika min empiriska undersökning. Jag betraktade dessutom nyttan utifrån båda perspektiv då jag å ena sidan fick pröva på en riktig arbetsomgivning men samtidigt närma mig mitt mål i en magisterexamen efter fyra års studier.

Jag skulle starkt rekommendera praktik som ett komplement till en undersökning eller varför inte under studieåren.

## 7 Slutsats

Denna studie undersöker det som bidrar till att ett system vidareutvecklas samt evalueringsmetoder genom vilka man vill få in återkoppling om vad som ska vidareutvecklas. Först och främst är fel och skavanker faktorer som bidrar till motiv för ett systems vidareutveckling. Detta visas både i empirin samt i teorin. Felen upptäcks genom tester som genomförs inför lansering av ett system men de upptäcks även efteråt. Brister som bidrar till svårigheter i användningen faller även inom ramen för faktorer och bidrar till motiv. Ett system som får synpunkter från användare såsom problem i användningen signalerar om behov till förändring. En annan faktor, som delas av både den teoretiska samt empiriska undersökningen, är omvärlden kring systemet och dess användare. En föränderlig omvärld bidrar till nya behov vilka i sin tur återspeglas i kraven för ett system.

Metoder som genomförs för att få reda på faktorerna delas även de mellan empiri och teori. Beta samt användbarhetstest är väldigt vanliga och interagerar direkt med en användare som låter sig exponeras för systemet. Två andra vanliga metoder som även de interagerar med användarna är enkäter samt intervjuer. En mer specifik metod, men trots det representerad i båda källorna, är den som innebär studier från utvecklarens sida. Särskild då studierna är relevanta till ett unikt system.

Utifrån metoderna har man som mål att vidareutveckla det man vill testa. Teoretiskt bestäms detta genom metodernas mål. I empirin är det dessutom sant att användbarhetstestet är en metod för att låta användare testa användbarheten som i sin tur ska ge sina synpunkter gentemot systemet. Detta leder till förbättrade arbetsflöden verktygen. Nya verktyg kan även fås utifrån användares behov som styrs av omvärlden. Exempelvis nya källor för information. Fel och brister som upptäcks leder till korrigeringar vilket kan utgöra en stor del av vidareutvecklingen och kan vara ett krav på löpande utveckling. Något som utöver allt upptäckts i empirin är att dessa faktorer inte alltid innebär en omedelbar vidareutveckling utan kan även ligga till grund för längre planering. En möjlig anledning till detta kan vara att litteraturen inte kan tala för specifika system utan generaliserar och försöker tala allmänt. Utifrån detta perspektiv kan det ses som naturligt att sådana skillnader dyker upp i en undersökning.

Slutligen kan allt förkortas med att

- Enkäter och intervjuer mest bidrar till att nya verktyg utvecklas eller gamla tas bort.
- Användbarhetstest bidrar till att verktyg och funktioner förbättras och effektiviseras för att förenkla användningen av systemet. Betatest kan även det bidra till bättre användbarhet samtidigt som buggar och fel kan upptäckas.
- Studier från utvecklare leder till ny funktionalitet.

## Källor

### Litteratur

- Backman J, *Rapporter och uppsatser*, Studentlitteratur, Lund, 1998.
- Bryman A., *Research Methods and Organization Studies*, Routledge, London, 1989.
- Davis W. S. & Yen D. C., *The Information Systems Consultants Handbook*, CRC Press, Florida, USA, 1999.
- Dumas J. S. & Redish J. C., *A Practical Guide To Usability Testing*, Intellect Books, Exster, UK, 1999.
- Dustin E., Rashka J. & Paul J., *Automated software testing: introduction, management and performance*, Addison & Wesley, Upper Saddle River NJ, USA, 1999.
- Flensburg P. & Friis S., *Mänskligare Datasystem*, Studentlitteratur, Lund, 1999.
- Föllesdal D. & Wallöe L., *Argumentationsteori, språk och vetenskapsfilosofi*, Thales, Stockholm, 2001.
- Gulliksen J., *Designing for Usability – Domain Specific Human-Computer-Interfaces in Working Life*, Acta Universitatis Upsliensis, Uppsala, 1996.
- Miles M. B. & Huberman A. M., *Qualitative Data Analysis*, Sage Publications, London, 1994.
- Ming-Yang Wang E., *Usability Evaluation for Human-Computer Interaction (HCI)*, Division for Industrial Ergonomics Department of Human Work Sciences, Doctoral thesis, Luleå University, 1992.
- Nielsen J., *Usability Engineering*, Academic Press San Diego, San Diego, USA, 1993.
- Reynolds G. W., *Information Systems for Managers*, Third Edition, West Publishing Company, St Paul, 1995.
- Rienecker L. & Jörgensen P., *Att skriva en bra uppsats*, Liber, Malmö, 2002.
- Svenning C, *Metodboken*, Lorentz Förlag, Lund, 1999.
- Thurén T., *Vetenskapsteori för nybörjare*, Liber, Malmö, 2003.
- Yin R. K., *Case Study Research – Design and Methods, Second Edition*, Sage Publications, Thousand Oaks, California, 1994.

## Artiklar

Hilbert D. M. & Redmiles D. F., *Large-Scale Collection of Usage Data to Inform Design*, <http://www.fxpal.com/people/hilbert/papers/EDEM-INTERACT01.pdf>, 2001. <http://www.fxpal.com/people/hilbert/cv.html#publications>.

## Tidskrifter

Warry L. III, Beyond FAQs, *Credit Union Magazine*, <http://creditunionmagazine.com>, [www.cuna.org](http://www.cuna.org), 2005.

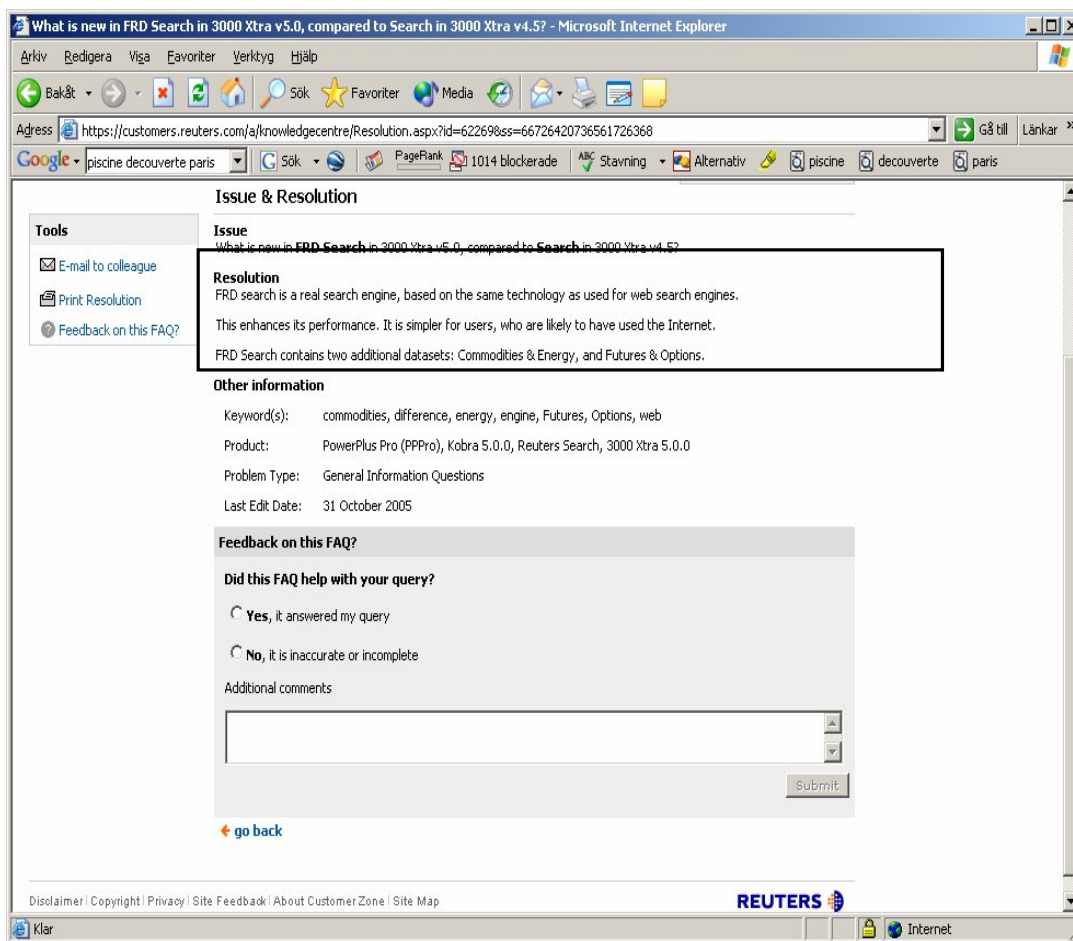
Flensburg P., Arbetet i den digitala tidsåldern, *Digitallife*, 4, [www.digitallife.se](http://www.digitallife.se), 2005.

## Dokument

Hörte S. Å., Hur kan man ge struktur åt rapporter och uppsatser, Luleå Tekniska Högskola, 1999.







En FAQ som visar på nytt verktyg.

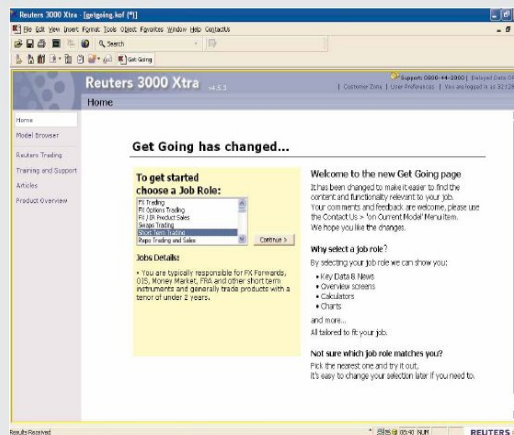
## Bilaga 2

Utdrag ur broschyren för den senaste versionen:

### Simplified data retrieval

The **Get Going Home Page** has changed so it's easier to find the content and functionality relevant to you.

With no codes to remember, just select your job role and Reuters 3000 Xtra v5.0 will tailor Get Going to your requirements. Quickly view relevant data, news, analytics and calculators. Gain easy access to training and support, additional services and new trading capabilities.



#### Trading

An at-a-glance view of Reuters trading capabilities. Find out more about these new services, view online demos and register your interest.

#### Training and Support

One click to access all Reuters training and support details. Find out where to sign up for courses, access online interactive training and all support documentation.

#### News and Events

Quick and easy access to news and events that matter to you. Directly access Reuters international and domestic news services.



#### Key data

Just by clicking on the menus you can get straight to the data relevant to your job – no codes to remember, just click and view.

#### What's new

Quickly view new additions to Reuters 3000 Xtra relevant to your job. When new data, news coverage, analytics and calculators come online this is the place to find out about them.

#### Highlights

Use the Highlights section to get the most out of Reuters 3000 Xtra. It's updated regularly with the most relevant content and functionality applicable to your job role.

Server-based technology provides new **fundamental and reference** datasets automatically. Immediately access mutual funds information from Lipper, institutional and stakeholders data from Factset, future events from Kalends and merger and acquisition content from Reuters.

Name	Last	% Held by Stakeholders	No. of Institutional Holders	% Held by Institutional holders	No. of Shares
ANGLO AMERICAN	1412	7.046	344	51.37	1493849728
BT GROUP	218.1845	0.785	489	60.897	8497272832
HAMMERSON	907.8908	0.529	181	71.835	284652640
NEXT	1.500	3.557	237	84.128	256973408
SCHRODERS NV	807	9.903	75	39.962	70939776
ASSOC BR FOODS	849.666	54.273	124	26.085	791674176
CADBURY SCHWEPPE	559.28	0.443	324	57.271	2081238656
HANSON	582 1/2	0.487	268	73.561	729468864
OLD MUTUAL	144 1/2	0.159	156	67.347	3855783360
SAGE GROUP	225	1.098	281	75.755	1284379392
ALLIANCE & LEICS	864.32	12.314	222	47.652	446680480
CARNIVAL	2843.26	20.146	209	45.997	212311168
HSBC HOLDINGS	891.55	0.416	485	77.016	1.126776368e+010
O2	154 3/4	0.192	240	64.559	8702862336

Reuters 3000 Xtra v5.0 has a new, fast and intuitive browser-based search tool for high-value fundamental and reference data.

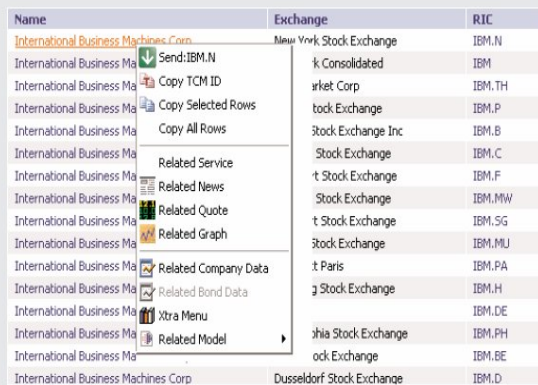
'Search All' Google-style with free-text search terms as well as names and codes. Enter queries like **Microsoft Options** or **Simex Euroyen Futures** to retrieve the instruments you're looking for. No more fiddling with multiple menus that don't support combinations of search terms.



Results are intelligently ranked for easy review.



Clicking on the selected RIC will send it to a display you have previously highlighted – for instance, a quote or graph. Alternatively, you can right-click on a search result to bring up a menu of displays for that instrument and then select the one you want.





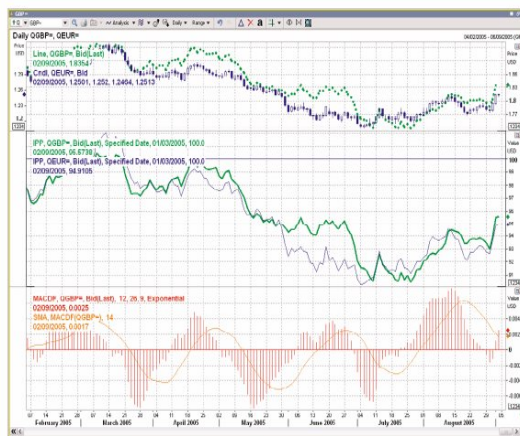
## New graphics

Reuters has listened to your feedback and created MetaStock Premium, a brand new charting application that replaces Reuters Graphics Object (RGO).

MetaStock Premium chart object lets you choose from 101 different analyses, with more trend lines and drawing tools. New features include:

**Session mode** You can easily display data for a specific date range, such as the current session, day or week. There's no need to waste time lining up the chart manually.

**Study-on-a-Study** You can base one analysis on the results of another. For instance, calculate a Relative Strength Index based on the Momentum of an instrument, or apply Bollinger Bands to the Relative Performance line of equity against an index.



**Improved usability.** It's much easier to change or add an analysis, control time in the x-axis, create composite expressions and more. And the Undo/Redo function means you can easily click back again if you don't like the results of what you've done.

**More trendlines and drawing tools.** These include Regression trendlines, Fibonacci Fan Lines as well as various symbols.

**Undo/Redo** This makes it simple to go back if you don't like the change you just made... or if you decide you do like it after all. There is no limit on how much you can undo."

**In-place editing** Change almost anything on the chart in a few simple clicks. To view a different instrument, click on the current name and enter the new one. Job done. Changing other details – interval, date range, time zone, analysis type, analysis, currency/units, and more – is equally simple.



Xtra Charts, updated from Reuters 3000 Xtra v4.5.x to take advantage of the new MetaStock charting object, are now called **Quick Charts**. Technical analysis, economic indicator, IRS, FX forecasts and curves charts now have a web-style look and feel, and can be easily integrated into your flex screen. Quick Charts use a custom template pre-populated with instrument data. There's no need to remember any codes; just choose your instruments from a drop down.

