

# Design av en applikation för stöd av samarbete i projekt

Patrik Löfgren

6 september 2006

Examensarbete i Datavetenskap, 20 poäng  
Handledare på Datavetenskap: Helena Lindgren  
Extern handledare: Mats Johansson  
Examinator: Per Lindström

UMEÅ UNIVERSITET  
INSTITUTIONEN FÖR DATAVETENSKAP  
901 87 UMEÅ  
SVERIGE



## Abstract

The computer has for decades been a tool to support collaboration, the progress of groupware goes forward with new solutions, however because collaboration can be carried out in many different ways it still remains many areas in the field to develop. We have created a proposal for the design for an application that will support the cooperation in smaller projects. It will give the members opportunity to work on common documents, store documents centrally and with an overview reach documents and other resources connected to the project. The program was prepared for the developing platform Lotus Domino<sup>®</sup> and through the integration with SWING Integrator<sup>®</sup> the managing of document includes some features like the possibility to create templates and connect reusable content and transfer the content from Lotus document into Microsoft Word or Excel<sup>®</sup>.

The development work has been carried out with the interaction design model. We have identified needs and established requirements with help of a pilot study where we have studied cooperative work in a literature review. We have also applied some features of the activity theory in the analysis of the cooperative work environment since the theory takes into account the surrounding factors that influence on how an activity best should be performed. Our design proposal covers essentially the requirements, but some parts need further development.



# Innehåll

<b>1 Inledning</b>	<b>3</b>
1.1 Bakgrund . . . . .	3
1.2 Förutsättningar för utvecklingsarbetet . . . . .	4
1.2.1 Lotus Domino . . . . .	4
1.2.2 SWING Integrator4 . . . . .	5
1.3 Rapportens uppbyggnad . . . . .	5
<b>2 Metod</b>	<b>7</b>
<b>3 Förstudie</b>	<b>9</b>
3.1 Erfarenheter från tidigare OSF-projekt . . . . .	9
3.2 Datorstöd för kooperativt arbete ur ett teoretiskt perspektiv . . . . .	10
3.2.1 Verksamhetsteorins ursprung . . . . .	11
3.2.2 Verksamhetsteorins huvudprinciper . . . . .	11
3.2.3 Engeströms modell för verksamhet . . . . .	14
3.2.4 Verksamhetsteorin och programutveckling . . . . .	15
3.2.5 CSCW och verksamhetsteorin . . . . .	17
3.2.6 OSF-projektets motsättningar . . . . .	19
3.3 Relaterade program . . . . .	20
3.4 Sammanfattning . . . . .	22
<b>4 Kravspecifikation</b>	<b>23</b>
4.1 Användbarhetsmål . . . . .	23
4.2 Användbarhetsprinciper . . . . .	23
4.3 Valfrihet för användaren . . . . .	24
4.4 Specifikation . . . . .	24
4.4.1 Krav på ingående element . . . . .	26
<b>5 Utveckling</b>	<b>29</b>
5.1 Skisser . . . . .	30
5.2 Post-it lappar . . . . .	30
5.3 Formulär i domino . . . . .	30

---

5.4	Vidareutveckling i Macromedia Flash . . . . .	30
5.5	Dokumenthantering . . . . .	30
<b>6</b>	<b>Resultat</b>	<b>33</b>
6.1	Uppbyggnad . . . . .	33
6.2	Utformning . . . . .	33
6.2.1	Övergripande utformning . . . . .	33
6.2.2	Ikoner, fonter och färger . . . . .	35
6.2.3	Datum . . . . .	36
6.2.4	Dokumentdatabas . . . . .	36
6.2.5	Navigering . . . . .	36
6.3	Formulär . . . . .	37
6.3.1	Formulär: Personlig översikt . . . . .	37
6.3.2	Formulär: Aktivitet . . . . .	37
6.3.3	Formulär: Person . . . . .	42
6.3.4	Formulär: Plats . . . . .	42
6.3.5	Formulär: Organisation . . . . .	42
6.4	Utvärdering mot kravspecifikationer . . . . .	45
<b>7</b>	<b>Diskussion</b>	<b>47</b>
7.1	Begränsningar . . . . .	49
<b>8</b>	<b>Sammanfattning</b>	<b>51</b>
<b>9</b>	<b>Framtida arbete</b>	<b>53</b>
<b>10</b>	<b>Tack</b>	<b>55</b>
	<b>References</b>	<b>55</b>
<b>A</b>	<b>Scenario</b>	<b>59</b>
A.1	Scenario 1 . . . . .	59
A.2	Scenario 2 . . . . .	59

# Figurer

2.1	Modell av interaktiondesign processen (fritt från [15, s.186]). . . . .	8
3.1	Verksamhethierarkin, verksamhet med dess undernivåer som kan kopplas till respektive motiv, mål och förutsättningar (fritt från [11, s.29]). . . . .	12
3.2	Medierade relationer på individuell nivå (fritt från [11, s.27]). . . . .	14
3.3	Basstrukturen på en verksamhet (fritt från [11, s.27]). . . . .	15
3.4	Fyra nivåer av motsättningar i ett nät av mänsklig verksamhet, varje siffra representerar en nivå (fritt från [6, s.89]). . . . .	16
3.5	Dynamiken i grupparbete (fritt från [1, s.92]). . . . .	18
3.6	Beståndsdelar från centrala aktiviteten i ett OSF-projekt. . . . .	20
5.1	Arbetsgång, visar tre faser i utvecklingsarbetet: skiss, A4 och Lotus Domino formulär. . . . .	29
6.1	Navigeringsträd, med programmets olika projektfaser. . . . .	34
6.2	Layouten på programmet. . . . .	35
6.3	Ikoner i applikationen. . . . .	36
6.4	Navigeringsträd, visat komprimerat och expanderat. . . . .	36
6.5	<i>Personlig översikt</i> , här visad med en översikt över <i>Ärende</i> och <i>Aktiviteter</i> . . . . .	37
6.6	<i>Aktivitetsformulär</i> , här visad vid registrering an <i>Ny Aktivitet</i> . . . . .	38
6.7	<i>Aktivitetsformulär</i> , här visad med <i>Översikt</i> . . . . .	39
6.8	<i>Aktivitetsformulär</i> , här visad med <i>Nytt Inlägg</i> . . . . .	40
6.9	<i>Aktivitetsformulär</i> , här visad med fliken <i>Dokument</i> med sparad dokument. . . . .	41
6.10	<i>Personformulär</i> , här visat i inmatningsläget <i>Ny Person</i> . . . . .	42





# Tabeller

3.1	Identifierade motsättningar i ett OSF-projekt. . . . .	21
4.1	Formulärens <i>Element</i> . . . . .	27
6.1	<i>Översiktsformulärets</i> innehåll. . . . .	38
6.2	<i>Aktivitetsformulärets</i> innehåll. . . . .	43
6.3	<i>Personformulärets</i> innehåll. . . . .	44
6.4	<i>Platsformulärets</i> innehåll. . . . .	44



# Kapitel 1

## Inledning

### 1.1 Bakgrund

Lotsa Utveckling är ett konsultföretag med inriktning mot verksamhets- och affärsutveckling med fokus på informationsförsörjning och kompetensförsörjning. I konsultuppdragen ingår utredningsarbeten och problemlösningsarbete utmynnande i handlingsplaner och nyttoanalyser i organisationers förändrings- och utvecklingsarbete. Företagets arbetshypotes är att företag, kommuner och andra offentliga aktörer, med hjälp av enklare och effektivare informationshantering, både kan få lägre kostnader och högre kundtillfredsställelse.

Lotsa Utveckling har utarbetat ett arbetssätt som de kallar Offentliga ServiceFlöden (OSF). Utifrån denna arbetsmall utförs uppdrag för kunder då främst mot offentliga aktörer. Arbetet består av tre faser: kartläggning; informations och organisationsdiagnos; nyttovärdering och handlingsplan. Dessa består i sin tur av olika metoder och arbetssätt för att nå målet.

Företaget har under hösten färdigställt en systemlösning Luna3 byggd på Lotus Domino<sup>®</sup>. Luna3 är en applikation som stödjer en rikstäckande administrationen av sjukskrivna byggnadsarbetare, som är utplacerade för arbetsträning i olika byggnadsföretag.

Syftet med detta projekt är att utveckla gränssnittet och flöden i en applikation som skall stödja projektarbeten med tillfälligt sammansatta medlemmar. Utvecklingen skall göras med utvecklingsverktyget Lotus Domino<sup>®</sup>. Lösningen skall vara generell, för att kunna implementeras i olika versioner hos skilda uppdragsgivare. En version av detta program kommer att utformas för att arbeta med OSF-projekt här fortsättningsvis benämnt OSF-applikationen, och här stödja arbetet med processen och stödja kunden i genomförandet av föreslagna förbättringsprojekt.

En litteraturstudie är genomförd för att få en teoretisk bakgrund och en djupare förståelse för de problem som kan uppstå i en verksamhet och med användandet av ett grupparbetsprogram. Verksamhetsteorin har studerats syfte att se hur verksamhetsteorin kan användas för design och programutveckling och då speciellt för CSCW (Computer Supported Cooperative Work).

## 1.2 Förutsättningar för utvecklingsarbetet

OSF-applikationen skall byggas upp runt fyra moduler *Systemkonfiguration*, *Resurser*, *Kontakter* och *Ärende*. *Systemkonfiguration* skall innehålla egenskaper anpassningar och integration. *Resurser* skall innehålla bokningsbara objekt, tjänster/produkter och inventarier. *Kontakter* skall innehålla organisation plats och personer. Innehållet i *Ärende* är ärende aktivitet och tid. Denna uppbyggnad liknar den i Luna3 och detta medför att delar av Luna3 skall kunna återanvändas.

Vår uppgift var att utforma ett gränssnitt för inmatandet och uppvisandet av uppgifter gällande organisation, plats, personer, ärende och aktiviteter. Vidare skulle vi utforma möjliga flödeshanteringar av ärende och aktiviteter. För detta skall Lotus Domino användas som utvecklings plattform.

### 1.2.1 Lotus Domino

Lotus Notes<sup>®</sup> är ett program som har utvecklats från kalkylprogrammet Lotus. I Lotus Notes finns en samling program utvecklade för att effektivisera arbete i grupp. Som, innehåller funktioner för e-post, kalender och adressbok. Vidare finns funktioner som serverbaserad lagring och åtkomst till gemensamma dokument och arbetsgrupper, affärsprocess automatisering, webbutveckling. Eftersom dessa applikationer måste vara flexibla och utformas specifikt för dess användningsområde innehåller Notes och Domino kraftfulla utvecklingsmiljöer. Den mest kraftfulla av dessa är Domino Designer<sup>®</sup> <sup>1</sup>.

Lotus Notes<sup>®</sup> är ett klientprogram och Lotus Domino<sup>®</sup> är ett serverprogram som är åtkomligt genom ett LAN eller WAN eller genom en modem uppkoppling. För att kunna få tillgång till uppgifter på servern måste man vara upplagd som användare och logga in. Användare kan ges olika slags access, en del användare kan enbart ha tillgång att läsa från databasen andra kan både läsa och skriva till den. En applikation Domino består av en eller flera databaser, där en databas är en samling dokument med ett index. Databasen består av fyra olika slags dokument:

- *Huvud dokument* (Header documents) Innehåller databasens namn och inställningar.
- *Återkomstlista* (Access Control List) Innehåller rättigheter på servrar användare och grupper.
- *Design dokument* (Design documents) Innehåller design som formulär, vyer och agenter.
- *Data dokument* (Data documents) Innehåller de dokument som produceras i databasen.

Med denna samling dokument kan nya applikationer skapas antingen utifrån givna mallar, som ger ett skelett att utforma från eller från en helt tom databas. Ett givet dokument kan skapas i ett dokument och uppvisas i ett annat. Olika användare kan använda olika formulär för att se samma dokument. Det är genom vyer som användaren får tillgång till dokument presentationen sker i tabellform och även här kan olika profiler få informationen presenterad på olika sätt.

Målsättningen bakom Lotus är att förstärka datorns roll som ett verktyg för kunskaps och affärsbaserat arbete och därigenom underlätta dagens arbetssätt. Men framför allt ge möjligheten att skapa nya mer effektiva sätt att samarbeta och hantera information.

<sup>1</sup><http://www-142.ibm.com/software/sw-lotus/products/product4.nsf/wdocs/dominohomepage>

Utformningen av Lotus dokument är till viss del styrd av att det arbetas med tabeller. Vidare karakteriseras Lotus applikationer av att nya sidor (databaser) öppnas upp i en flik i Lotus programmet, detta ger en bra översikt och gör det lätt att växla mellan sidor. Många av Lotus egna databaser har en knapprad längst uppe på sidan för val av kommando. [16]

### 1.2.2 SWING Integrator4

För att ge möjlighet att utveckla automatiseringen av dokumenthanteringen kommer SWING Integrator4<sup>®</sup> att integreras i vårt program. Med SWING Integrator4<sup>®</sup> ges möjligheter att förena Lotus Notes<sup>®</sup> data med Microsoft Word<sup>®</sup> eller Microsoft Excel<sup>®</sup> mallar. Härigenom kan flera processer automatiseras och användandet av systemet i många avseenden förenklas. Några exempel på vad som kan göras är att: automatiskt sätta ihop Microsoft Word<sup>®</sup> dokument genom att använda villkorliga och valbara avsnitt; spara Microsoft Office<sup>®</sup> dokument tillbaka till Notes; redigera Office dokument på plats, då både i Notes och på Internet; skapa Excel rapporter och kartor från Notes data och skapa pdf dokument och rapporter. <sup>2</sup>

## 1.3 Rapportens uppbyggnad

Vi inleder med en metodbeskrivning innehållande de metoder som vi har använt i vårt arbete. Därefter kommer en förstudie där vi har studerat hur Lotsa har arbetat i tidigare OSF-projekt. Här ingår även en teoridel där vi studerat teorier för datorstöd i kooperativt arbete. Vi har även studerat liknade program och utvecklingsprojekt. I kapitel 4 sätter vi upp en kravspecifikation som vi arbetar efter under utvecklingsarbetet som presenteras i kapitel 5. Slutligen redovisas och diskuteras resultaten.

---

<sup>2</sup><http://www.swingsoftware.com/>



# Kapitel 2

## Metod

Arbetet innefattar ett designuppdrag och en litteraturstudie. Litteraturstudien riktar sig mot att belysa de behov som kan uppstå med grupparbetsprogram samt att ge idéer till nya lösningar. Vi vill även med utgångspunkt med verksamhetsteorin ge en teoretisk grund för uppkomna lösningar och utvecklingsarbete. Litteratur har eftersökts på artiklar i databaser som ACM Digital Library, ACM guide to computing literature och Academic Search Elite (EBSCO).<sup>1</sup> Följande sökord har använts: CSCW, Activity theory, Victor Kaptelinin, Bonnie A Nardie, Yrjö Engeström.

Designuppgiften har genomförts med Interaktion design processen (hädanefter ID) beskriven av Preece [15]. Vi går kort igenom ID processen, den definieras lättförståeligt av Preece [15, s.6] som:

“designing interactive products to support people in their everyday working lives. ... In particular, it is about creating user experiences that enhance and extend the way people work, communicate and interact”

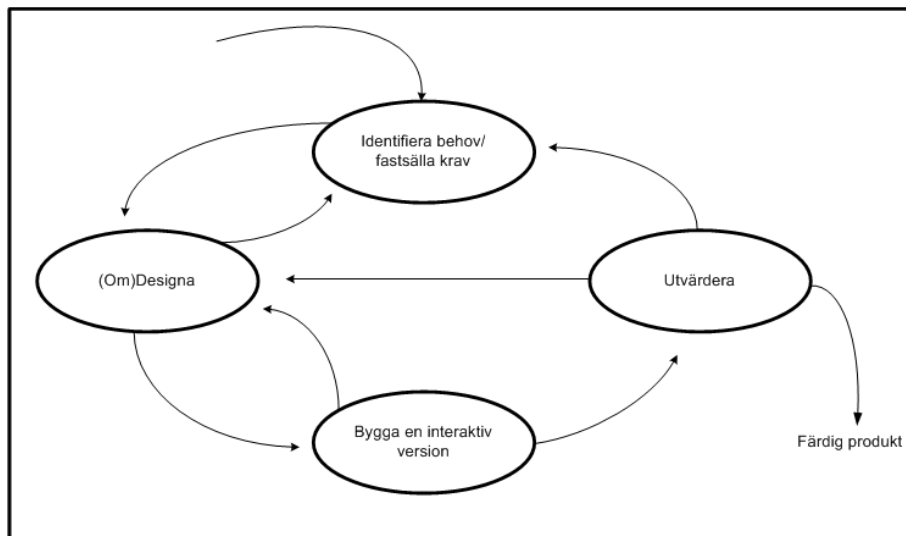
About

ID-processen består i huvudsak av fyra grundaktiviteter:

1. *Identifiera behov och fastställa kravspecifikation.* Denna del har genomförts med en vår förstudie som grund.
2. *Utveckla olika designförslag som möter de fastställda kraven.* Här har design förslag utvecklats i cykler, med början med enkla skisser för att fastställa innehållet i formulär och vyer. Vidare har med post-it lappar för att vidare utveckla designen men även för att testa flödet i programmet.
3. *Bygga interaktiva versioner av designförslagen så de kan kommuniceras med och utvärderas.* Här har det arbetats med A4 sidor med post it lappar för att provköra flödet även färdigställda enskilda formulär som till delar fungerat har testas.
4. *Utvärdera vad som åstadkommit genom processen.* Här har vi genomfört egna tester mot kravspecifikationen och “talk aloud”-tester med personer från Lotsa utveckling.

---

<sup>1</sup><http://portal.acm.org>  
<http://portal.acm.org/portal.cfm>  
<http://web32.epnet.com/>



Figur 2.1: Modell av interaktionsdesign processen (fritt från [15, s.186]).

Dessa aktiviteter bygger på varandra och skall repeteras i cykler. Exempelvis kan nya behov och krav identifieras under processen eller vissa krav kan visa sig ogenomförbara tillsammans med andra krav. Under genomförandet understryks att användarna skall involveras i processen, specifika användbarhets och användarupplevelse mål identifieras och dokumenteras i början av projektet, och att utvecklingen skall ske i cykler genom de fyra aktiviteterna.

Den person på Lotsa som har som uppgift att arbeta med OSF-projekt har deltagit i designprocessen och representerat utvecklingen. Vi studerade även hur det har arbetas från Lotsas sida i tidigare OSF-projekt genom att gå igenom den dokumentation som hade genomförts.



# Kapitel 3

## Förstudie

I vår förstudie har vi arbetat utifrån olika synvinklar. Inledningsvis studerades hur Lotsa har arbetat med OSF-projekt, då specifikt i ett projekt mot Vindelns kommun. Vidare studerades litteratur om grupparbetsverksamhet och hur datorapplikationer kan stödja den verksamheten. I detta sammanhang kommer vi in på *Verksamhetsteorin* som är en teori som kan användas för att analysera både individuella verksamheter och gruppverksamheter. Studien om verksamhetsteorin syftade till att tillämpa teoretiska tankegångar om hur verksamheter bedrivs och utvecklas. Vi har även studerat liknande program och utvecklingsprojekt som Lotus TeamRoom, ContactMap [18], Presto [5] och UMEA [9] detta för att få nya idéer och nya infallsvinklar.

### 3.1 Erfarenheter från tidigare OSF-projekt

Studien av lotsas arbetssätt Operativa Service Flöden (OSF) gav att processen består av tre delmoment: kartläggning; informations och organisationsdiagnos samt nyttovärdering och handlingsplan. I dessa ingår syfte, mål, metod och ett resultat. Vi ser här möjligheten att handlingsplanen skall kunna innehålla färdiga ärenden med exempel: uppgifter, personer och tidsplan. Denna del skall då övertas av kunden som ett arbetsverktyg för att genomföra förändringsprocessen och vidare uppföljningar.

Vi antar också att som i de flesta projekt att ett OSF-projekt hos en kund integreras i linjeorganisation, något som innebär att deltagare delar tiden med andra uppgifter och projekttiden kan bli utsträckt över en längre tidsperiod. Samt att tider för möten kan vara svåra att boka in passande för alla deltagare.

Genomgång av ett tidigare OSF-projekt med Vindelns kommun gav att undersökningar och sammanställningar av undersökningar görs manuellt. Och att tänkta målgrupper är både kommunens egna anställda men även de som på något vis arbetar med eller har kontakt med kommunen.

Spektrumet på de som skal använda programmet är mycket brett. I första hand är det ett verktyg för projektledarna på Lotsa och projektmedlemmar hos kommunen, men programmet kommer både att användas före att samla in information från kunder till kommunen och vid genomförandet av de handlingsplaner som projekten utarbetar. Det medför att den tänkbara användarprofilen sträcker sig från mycket vana datoranvändare till datoranvändare med kunskaper begränsade till specifika program.

## 3.2 Datorstöd för kooperativt arbete ur ett teoretiskt perspektiv

Eftersom vårt program är ett samarbetsprogram med dess speciella problem har vi studerat forskningsområdet CSCW, (Computer Supported Cooperative Work) som innefattar design, införandet och användandet av grupprogram. Detta inrymmer förutom frågeställning runt arbetsrelaterat grupparbete, frågor om tävlande, socialisering och spel. Förutom de traditionella kunskaperna om gränssnittsdesign kräver grupprogramdesign även förståelse för hur grupper fungerar och hur individer uppträder i grupper och en uppfattning om hur nätverksteknologin fungerar och dess inverkan på användarna av system. CSCW har många aspekter som gör utvecklingen av program svåra, några av de som Brinck [3] nämner är:

- Organisering och schemaläggning blir svårare för en grupp än för en individ.
- Sättet på hur en grupp samspelar är svårare att förutse än hur individer uppträder.
- En grupp är dynamisk medlemmarnas roller förändras och gruppen interaktion förändras över tiden.
- Kan vara svårt att testa nya program eftersom det kan innefatta att det måste testas på en hel organisation.

Införandet av CSCW-system kan vara en avgörande fas för hur systemet kommer att fungera i en organisation. Denna fas kräver ofta att en kritisk massa använder systemet för att det skall fungera tillfredställande. En av förutsättningarna för att nå detta mål, är att individen tjänar på att använda systemet. Ofta räcker det inte heller att alla på individnivå tjänar på att använda systemet, för att om några av andra anledningar inte använder systemet som det är tänkt kommer systemet inte att fungera tillfredställande. Något som är vanligt förekommande är att bokningssystem på arbetsplatser inte fungerar i verkligheten, detta på grund av att några inte för in bokningen i systemet. En annan viktig aspekt är att undvika att användare missbrukar systemet genom att sabotera grupparbeten eller inkräktar på någons privata område. [3]

Kommunikationen mellan människor har alltid någon slags struktur. Ställer vi en fråga förväntas alltid ett svar eller en önskan om ett förtydligande. I en organisation kan officiella dokument vandra förutbestämda vägar för godkännande och underskrifter. Om sådana funktioner skall införas i ett samarbetsprogram kan strukturen bli mer eller mindre teknisk eller social. Detta beroende på hur mycket av kommunikationen mellan applikationens användare som med automatik styrs av systemet. Inom CSCW används termerna teknisk förmedlad kommunikationsstruktur och social förmedlad kommunikationsstruktur. Tekniskt förmedlad kommunikationsstruktur är när systemet bestämmer exakt hur kontakten mellan användarna är strukturerad. Emedan socialt förmedlad kommunikationsstruktur bestämmer användarna hur och när kommunikationen skall ske. Exempelvis när e-post skall skickas och om det skall svaras på och när. Socialt förmedlad kommunikation kan ta längre tid och det finns större risk för fel. Emellertid kan undantag i kommunikationsstrukturen många gånger vara mera regel än undantag. Detta gör att teknisk förmedlad kommunikation ofta kan bli ett hinder för att få ett arbete effektivt utfört. [3]

Vi har studerat verksamhetsteorin utifrån den ståndpunkt att den kan ge ett teoretisk grund för hur en verksamhet fungerar, samt att den kan vara ett verktyg för att identifiera problemområden vid program utveckling och då speciellt vid CSCW.

Traditionellt har mänsklig aktivitet varit utgångspunkten för analys inom psykologiska teorier. Denna utgångspunkt gör det relativt enkelt att genomföra experiment i laboratorier, där isolerade aktiviteter kan studeras och utvärderas. Däremot blir det svårare att analysera isolerade aktiviteter i verkliga situationer utanför laboratorier. Med verksamhetsteorin däremot utgör verksamheten grunden för analysen, här måste hänsyn tas till det sammanhang där verksamheten äger rum. Vidare är det även viktigt att ta hänsyn till den historiska utvecklingen av verksamheten, den måste analyseras för att förstå användandet av specifika arbetsmetoder. Denna verksamhet definieras som det minsta meningsfulla sammanhang i vilken de enskilda aktiviteterna i en verksamhet kan förstås.

Kuutti [11] sammanfattar verksamhetsteorin som ett filosofisk och tvärvetenskapligt skelett för att studera olika former av mänskligt deltagande i utvecklingsprocesser, detta både på den individuella och sociala nivån på samma gång.

### 3.2.1 Verksamhetsteorins ursprung

Verksamhetsteorin (Activity Theory) har sin härkomst från flera olika håll, grundtankarna härrör från tysk filosofi runt sekelskiftet till 1900-talet. Kant, Fichte och Hegel introducerade de första tankarna runt verksamhetsbegreppet, de framhävde både tankar om utvecklingens och historiens betydelse för en verksamhet, samt en aktiv och konstruktiv roll från människans sida. Marx och Engels utvecklade tankarna vidare och betonade den praktiska sidan av verksamheter, närmare bestämt omvandlandet av materiella objekt. Med inspiration från dessa föregångare lades grunden till verksamhetsteorin som den ser ut idag av de ryska kulturhistoria psykologernas Vygotsky, Leont'ev och Luria arbete på 1920 och 1930-talet. Vygotskys modell av mänsklig utveckling benämns som sociokulturell, han menade att individens mentala utveckling såsom tankar, språk och logisk tänkande är beroende av omgivningens influenser, och att dessa färdigheter utvecklas genom social interaktion med människor i omgivningen. På 1980-talet tog forskare i väst upp teorierna inom beteendevetenskap och sociologi och verksamhetsteori utvecklades vidare. [2] Medvetandet har en central roll i verksamhetsteorin och definieras av Vygotsky som ett fenomen som sammankopplar uppmärksamhet, avsikt, minne, resonemang och tal [14]. Verksamhetsteorin uttrycker att detta medvetande är socialt betingat, att medvetandet inte kan isoleras till bara representationer i någons hjärna. Medvetandet formas och utvecklas av människor och verktyg i miljön genom att verksamheter utförs [11].

### 3.2.2 Verksamhetsteorins huvudprinciper

En verksamhet består av subjekt, objekt, aktiviteter och operationer, subjektet är en person eller en grupp som är inbegripen i verksamheten. Subjektet har ett objekt som verksamheten styrs mot. Bakom objektet finns det ett behov, motiv, mot vilken verksamheten alltid svarar. Stegen mot objektet delas in i aktiviteter och operationer. Verksamhetsteorin präglas av två grundstenar som enligt Kaptelinin [10, s.28] kan uttryckas på följande sätt:

“(1)the human mind emerges exists, and can only be understood within the context of human interaction with the world; and (2) this interaction, that is, *activity*, is socially and culturally determined.”

Dessa idéer manifesteras i fem nyckelbegrepp: objektorientering, hierarkisk aktivitetsstruktur, internalisering / externalisering, mediering och utveckling.

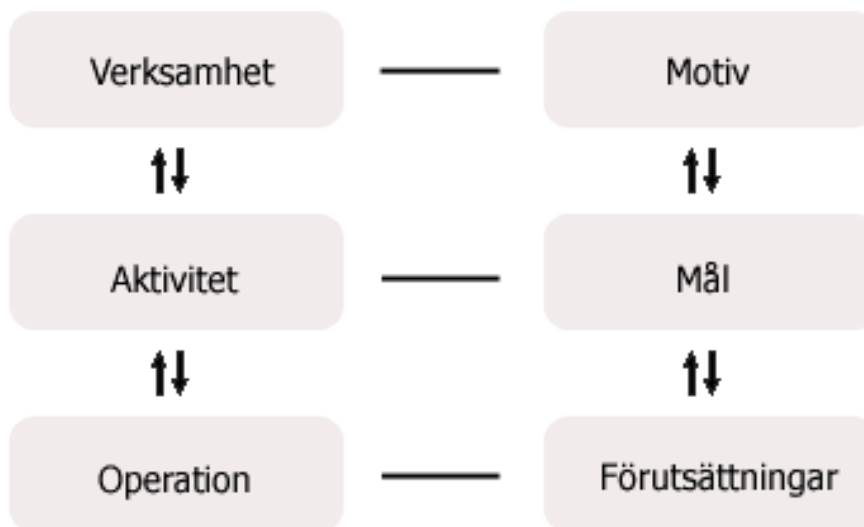
### Objektorientering

Principen för objektorientering är att varje verksamhet har ett motiv som är riktat mot något som objektivt existera, ett objekt. De objekt mänsklig verksamhet kan rikta sig mot kan antingen vara saker eller människor. Objektet kan vara någon framtida lösning som verksamheten är riktad mot och som slutligen kommer att vara verksamhetens resultat när den är genomförd. Denna verksamhet är inte begränsad till det fysiska, biologiska eller kemiska tillståndet hos objektet utan även sociala och kulturella tillstånd. Detta innebär även att ett tanketillstånd att utföra en specifik handling är ett tillstånd. [10]

### Hierarkisk aktivitetsstruktur

En *verksamhet* (Activity) är en handling som utförs för att möta ett motiv, som en deltagare i en verksamhet har. Den består av att i en längre process, som oftast består av att under flera steg och faser, bearbeta ett objekt för att nå det önskade målet. En verksamhet kan utföras individuellt eller som en gruppaktivitet, och kan bilda en kedja eller nätverk av verksamheter som binds samman av ett de riktas mot ett gemensamt objekt och motiv.

Förutom denna huvudstruktur finns även en inre hierarkisk aktivitetsstruktur. Den består tre nivåer: *verksamhet*, *aktiviteter* (Actions) och *operationer* (Operation). Som figur 3.1 visar kan verksamheten kopplas till ett motiv, aktiviteten till ett mål och en enskild operation till förhållandena. En och samma verksamhet kan genomföras genom att utföra flera olika aktiviteter beroende på situationen. En och samma aktivitet kan också tillhöra olika verksamheter.



Figur 3.1: Verksamhetshierarkin, verksamhet med dess undernivåer som kan kopplas till respektive motiv, mål och förutsättningar (fritt från [11, s.29]).

I sin tur består en aktivitet av en eller flera operationer, en operation är handlingar som utförs automatiskt utanför medvetandet. En aktivitet kan övergå i operationer om den blir rutinmässig och omedveten genom övning. Ett exempel är när vi lär oss växla

under bilkörning. I början måste vi tänka på att vi skall växla, att vi skall trycka ned kopplingen och föra växelspaken till rätt läge. Men efterhand är detta något som blir helt automatiserat, aktiviteten övergår till en operation. En operation kan dock återgå till en aktivitet om det uppstår något problem (breakdown), som innebär att aktiviteten inte kan utföras genom de invanda operationerna, vilket leder till att vi blir medvetna om att vi utför aktiviteten. Till exempel om vi kör en bil som vi inte är van att köra så kan växlandet övergå från att vara en operation till en aktivitet, då föraren eventuellt måste titta på växelspaken för att hitta igen rätt växelläge.

### Internalisering vs. Externalisering

Verksamhetsteorin särskiljer på inre och yttre verksamhet. I verksamhetsteorin betonas att den inre mentala verksamheten inte kan förstås om den analyseras separat, isolerat från den yttre verksamheten. Detta för att det är den ständiga transformeringen mellan dessa som är grunden för den mänskliga kognitionen och verksamheten. Internalisering (Internalization) är överföringen av yttre verksamhet till inre verksamhet. Exempel; att lära sig räkna kan inledningsvis bestå i att räkna på fingrarna, när processen är internaliserad kan räkningen utföras i huvudet, utan hjälp av yttre hjälpmedel. Internalisering ger oss vidare möjlighet att simulera den yttre verksamheten mentalt som en inre verksamhet, på detta vis kan alternativa handlingsplaner prövas innan vi utför en handling.

Externalisering (Externalization) transformerar inre verksamhet till yttre, något om ofta är nödvändigt när något går fel med den inre verksamheten exempel; att räkna ett komplicerat tal, uppgiften kan vara svår att lösa med huvudräkning, men kan lätt lösas med hjälp av penna och papper. Externalisering är också nödvändig vid samarbete mellan flera personer, där verksamheten måste utföras externt för att koordinera verksamheten. [10]

### Mediering

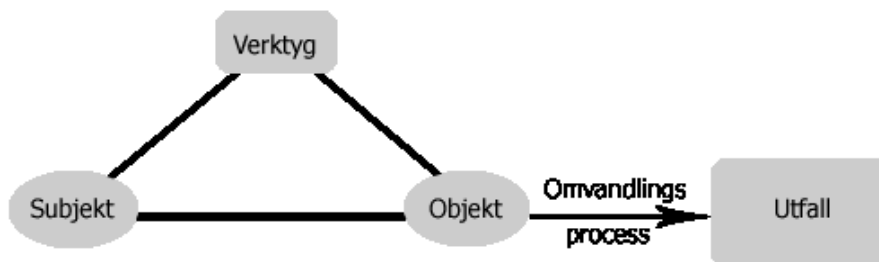
Verktygen formar det sätt som människan samspelar med omgivningen, formas den externa verksamheten resulterar de i att även den interna verksamheten formas. Vidare reflekterar oftast verktygens utformning erfarenheter från tidigare användare, som har försökt lösa liknande problem. Detta ger både egenskaper för verktyget (form, storlek, material) och kunskap om hur verktyget skall användas. Verksamhetsteorin betonar att ett verktyg kommer till sin mening när det används och kunskapen om hur verktyget används är av stor vikt. Verktyget tar därigenom med sig en kunskap som tidigare användare investerat i verktyget genom att utforma det på ett för verksamheten funktionellt sätt. [10]

Begreppet verktyg är omfattande, eftersom en verksamhet både kan riktas för att omforma ett fysiskt objekt eller påverka ett mänskligt objekts inre verksamhet (sig själv eller någon annan). Dessa betecknas som tekniska verktyg (ex hammare) och psykologiska verktyg(ex multiplikationstabellen, reklam och språk). [14]

### Utveckling

Enligt verksamhetsteorin är en verksamhet dynamisk och genomgår hela tiden förändringar, nya färdigheter utvecklas hos deltagarna, nya verktyg utvecklas eller de yttre förhållanden förändras, allt påverkar hur verksamheten utförs. Men utvecklingen utgår alltid från tidigare erfarenheter. Verktygen har en viktig del i denna process och för att

förstå denna utveckling är det viktigt att se hur verktygen används, och detta i dess rätta miljö och det under en längre tid. För det är ofta verktygens utformning som gör det möjligt att utveckla nya arbetssätt hos användaren. Denna process kan ibland leda till att verktygen blir mer användbara än vad som ursprungligt var tänkt. Det kan också vara så att ett verktyg har begränsningar som hindrar utvecklingen och begränsar användarens utvecklingsförmåga. [11] [10]



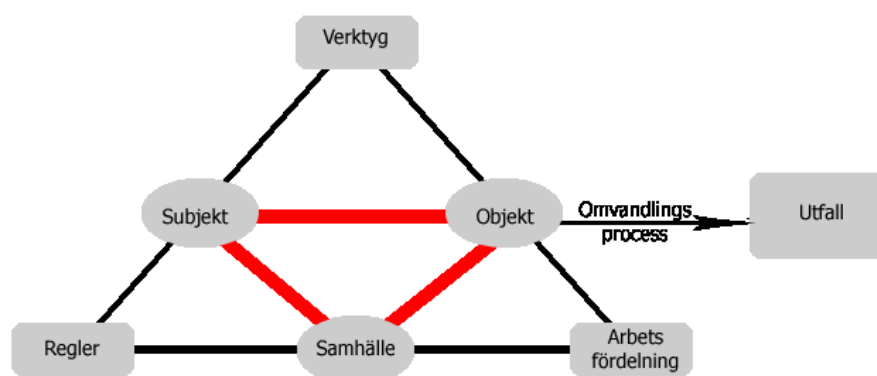
Figur 3.2: Medierade relationer på individuell nivå (fritt från [11, s.27]).

### 3.2.3 Engeströms modell för verksamhet

En forskare som har influerat verksamhetsteorin det senaste decenniet är finländaren Yrjö Engeström [4], han har utvecklat Vygotskys modell för att vara tillämplig inom utvecklande arbetsforskning. Engeström utgick från modellen av enskild förmedlad verksamhet beskriven av Vygotsky [17] (se figur 3.2) innehållande subjekt, objekt och verktyg. Där *Subjektet* syftar på individen eller den speciella grupp vars förehavanden är utgångspunkt för analysen. *Objektet* syftar på det som skall förädlas eller den problemrymd där verksamheten rör sig för att bli transformerad till lösningar. Detta uppnås med hjälp av fysiska och symboliska *verktyg* som antingen se som interna eller externa. Utifrån denna modell och modellen av socialt medierad verksamhet beskriven av Leontiev [12] byggde Engeström sin modell, hans resonemang influerades även från filosofi och stenåldersantropologi. En ny nod blev *Omgivningen* (samhälle/gemenskap) något som innefattar flera individer eller och specifika grupper som delar samma objekt. Omgivningen har en ömsesidig relation med subjektet och objektet. Dennes relation förmedlas med *regler* respektive *arbetsfördelningen*. Regler innefattar normer och överenskommelser som begränsar samspelet inom verksamhetssystemet kan vara både explicita eller implicita. Arbetsfördelningen pekar på uppdelningen av arbetet samt graden av styrka och status.[2] (se figur 3.3)

Engeströms modell kan även innefatta konsumtion, med de tre huvuddominerande synsätten på mänsklig aktivitet som undernivåer: *produktion*, *distribution* och *utbyte*. Med denna uppdelning tillkommer ytterligare möjligheten att analysera en mängd olika relationer inom verksamheten. [4]

Engeström säger vidare att mänsklig verksamhet inte existerar i ett vakuum utan samspekar hela tiden med ett nät av andra verksamhetssystem. Denna yttre påverkan på verksamheten sker inte direkt utan är först anpassade och modifierade till inre faktorer i verksamheten. Det verkliga orsaks sambandet visar sig då denna yttre påverkan internaliseras med verksamheten, något som inträffar vid obalans i systemet. Verksamhetssystemet arbetar hela tiden med att hantera motsättningar inom och emellan sina element. I



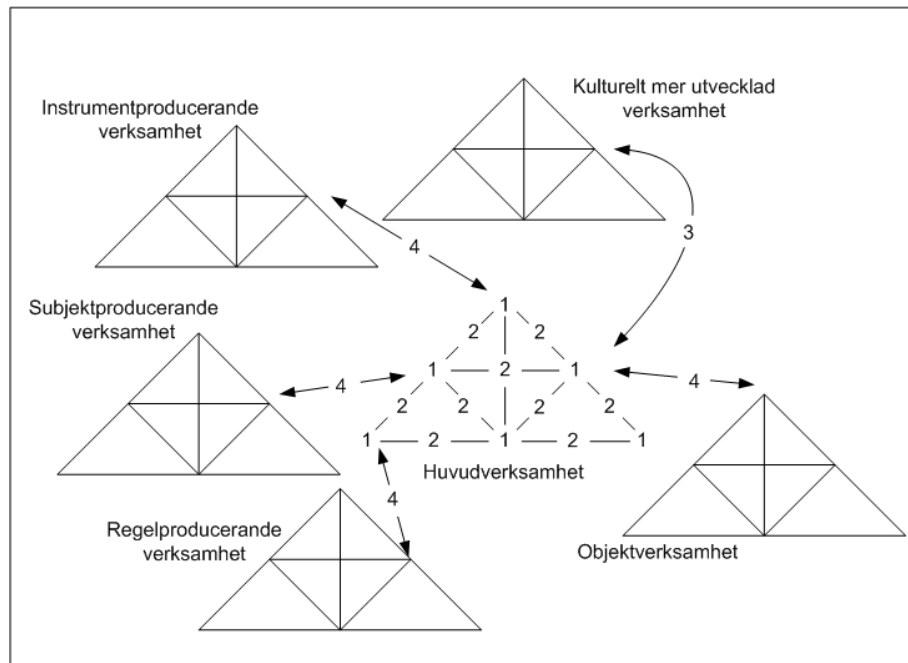
Figur 3.3: Basstrukturen på en verksamhet (fritt från [11, s.27]).

denna process utvecklas verksamheten. Vi kan se ett verksamhetssystem som ett system som genom ständiga störningar blir en uppfinningsproducerande maskin. Med denna yttre påverkan får vi något som Engeström benämner som fyra olika klasser av motsättningar (se figur 3.4). *Primära motsättningar*, är de inre motsättningar inom varje beståndsdel i den centrala verksamheten. Då i första hand värdet av vad som fås ut i utbyte mot vad som sätts in. *Sekundära motsättningar*, innefattar motsättningar mellan beståndsdelar i den centrala verksamheten. Dessa uppstår främst då nya element kommer in i verksamheten. *Tertiära motsättningar*, är motsättningar mellan objekt/motiv i huvudverksamheten mot objekt/motiv i en mer kulturellt avancerad huvudverksamhet. Detta kan uppstå då nya modeller för hur verksamheten skall utföras förs in i systemet. *Kvartära motsättningar*, är motsättningar mellan huvudverksamheten och dess närliggande aktiviteter. Dessa uppstår då huvudverksamheten utvecklas eller att det ses möjligheter att utveckla densamma och de konflikter som uppstår i samspelet med närliggande verksamheter. Här uppstår ofta konflikter och missuppfattningar mellan de olika systemen. [4][6]

### 3.2.4 Verksamhetsteorin och programutveckling

Utifrån verksamhetsteorin är en dator med dess program bara ett nytt verktyg som förmedlar interaktionen mellan människan och omgivningen. [8] För att förstå denna interaktion måste vi titta på den övergripande uppgiften för datoranvändandet, denna verksamhet blir den minsta meningsfulla sammanhang som interaktionen kan förstås. Kaptelinin [8] ställer några frågor som uppkommer i detta sammanhang: Hur ser den hierarkiska aktivitetsstrukturen ut i användarkontexten i ett verksamhetsteoretiskt perspektiv? Vilka verktyg förutom datorns är tillgängliga för användaren? Vad har användaren för fokus och hur ser omgivningens fokus ut? Dessa frågor kan verka övergripande och löst kopplade, vid arbetet med att utveckla användargränssnitt, men är enligt Kaptelinin nödvändiga. Om inte dessa faktorer tas hänsyn till kan exempelvis användarnyttan bli låg eller programmet inte passa för en speciell målgrupp.

En annan viktig aspekt på användarsituationen är i vilken mån utveckling sker av verktyg och användare, det är viktigt att analysera användare situationen utifrån ett utvecklingsperspektiv, då både på individuell nivå och på grupp och organisationsnivå. Detta eftersom införandet av nya program innebär att nya arbetssätt tillkommer och



Figur 3.4: Fyra nivåer av motsättningar i ett nät av mänsklig verksamhet, varje siffra representerar en nivå (fritt från [6, s.89]).

andra kanske försvinner. Ett sätt att stödja denna process är att stödja användaren i att utveckla systemet enligt dennes behov. Denna process kan vara svår beroende på att användarna kan ha svårt att formulera sina behov.

Utvecklingen av den individuella färdigheten är också en viktig faktor som kan identifieras med hjälp av hierarkiska verksamhetsstrukturen.

Nardi [13] ger fyra praktiska tips vid användandet av verksamhetsteorin för design och utveckling av applikationer:

1. Undersökningstiden måste vara tillräcklig lång för att förstå användarnas objekt, och ge möjlighet att observera när det sker även byten av objekt, fokusskifte.
2. Breda mönster av aktiviteter måste studeras.
3. Använd olika tekniker för insamlande av material som, intervjuer, observationer, video, historiskt material.
4. Lägg stor vikt att förstå verksamheten ur användarens perspektiv.

Vidare föreslår hon att med kreativ syntes med verksamhetsteorin och distribuerad kognition och detta med ett situationsaktion perspektiv skulle vara en väg till framgång.

### Verktyg för tillämpning av Verksamhetsteorin

Med syftet på att få ett praktiskt verktyg för att analysera verksamheter utifrån utarbetade Kaptelinin [10] en checklista (Activity Checklist) som är en samling specifika frågor att ställa mot verksamheten. Verksamhetschecklistan är tänkt att kunna användas både



tidigt i utvecklingsarbete och vid utvärdering av färdiga system. Arbetsgången är att först utvärderas hela problemrymden för att vidare fokusera på specifika problemområden. Strukturen på Checklisten utgår från de fem grundstenarna i verksamhetsteorin som vi har gått genom tidigare. Eftersom uppgiften är att analysera hur vi använder datorprogram är tyngdpunkten lagd på förmedling som kombineras med de övriga principerna. Detta ger fyra avdelningar med ett antal frågor. De fyra avdelningarna och dess utgångspunkt beskrivs av Kaptelinin [10] som:

- *Mål och medel* (Means and ends), i vilken grad tekniken underlättar och begränsar användarens strävan att nå målet och det vis tekniken kan lösa eller skapa konflikter mellan olika mål.
- *Sociala och fysiska aspekter på omgivningen*, hur systemet samverkar med krav, verktyg, resurser och sociala regler hos omgivningen.
- *Inläring, kognition och artikulation*, inre delar ställs mot yttre delar i verksamheten och deras gemensamma stöd från systemet.
- *Utveckling*, innefattande omvandling och utveckling av de övriga punkterna.

[10]

När det gäller användandet av checklisten poängterar Kaptelinin [10] att den inte är tänkt att användas som det enda stödet vid programutveckling utan bör kombineras med andra utvecklingsmetoder. Det har bland annat visat sig att den är mycket bra att använda för att identifiera de viktigaste problemområdena, något som gör det lättare att veta vilka områden och frågor som skall behandlas i intervjuer eller observationsstudier.

Även om Activity Checklist har en linjär struktur så har det visats att den med fördel kan användas över de olika sektionerna, för att följa mönster av relaterade aspekter. Vidare behöver bara de frågeställningar som är relevanta i den aktuella utvecklingssituationen användas.

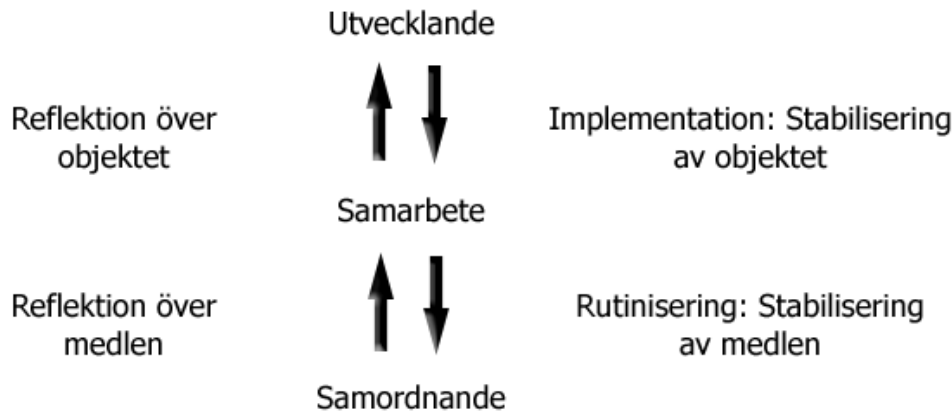
Viktigt är att användaren väl sätter sig in i de olika frågeställningarna. Den kan med fördel läras in utantill så det blir ett tankesätt. Den bör användas upprepade gånger i de olika faserna, inledningsvis för att identifiera större felaktigheter, senare även för att mer grundligt analysera identifierade punkter som visar viktiga för systemet.

Sist och kanske viktigast, användaren måste veta varför den används och följaktligen ha ett klart mål med vad som vill åstadkommas. [10]

### 3.2.5 CSCW och verksamhetsteorin

Verksamhetsteorin kan även användas för att studera och analysera dynamiken i grupparbetsverksamhet. För denna analys indelar verksamhetsteorin grupparbetsverksamhet i en heuristiskstruktur i tre nivåer: samordnande (co-ordinated), samarbetande (co-operativ) och utvecklande (co-constructive) grupparbetsverksamhet. [7]

*Samordnande verksamhet* är den normala och rutinaktiga flödet av interaktion mellan medlemmarna. Aktiviteterna riktas mot samma objekt, men dessa individuella aktiviteter är bara externa. Skriften som följs representeras i skrivna regler, i planer, i scheman och i gängse normer. Detta leder till att deltagarna är passiva medlemmar i aktiviteten. Samordnande verksamhet möjliggör att verksamheten kan genomföras tillsammans med andra verksamheter.



Figur 3.5: Dynamiken i grupparbete (fritt från [1, s.92]).

*Samarbetande verksamhet* riktar sig mot ett gemensamt objekt men den enskilde medlemmen delar även motivet med verksamheten med övriga medlemmar. Med hänsyn till detta gemensamma objekt måste den egna aktiviteten balanseras med de aktiviteter som utförs av andra medlemmar. Det kan även bli aktuellt att påverka andra medlemmars aktiviteter i syfte att nå målet med verksamheten.

I *utvecklande verksamhet* är motivet stabilt och oftast överlag överenskommet om. Däremot förekommer det att medlen för att genomföra verksamheten inte finns eller är bestämda. Som medel räknas mallen (script) som ger fördelningen av verksamheten på flera aktiviteter och medlemmar, och förmedlande verktyg. Som ett led i att utföra utvecklandeverksamhet måste dessa medel fastställas. [1]

Viktigt i förståelsen av den heuristiskstrukturen av en verksamhet är de dynamiska övergångarna mellan de olika nivåerna.(se figur 3.5) Övergångarna är bundna till stabiliteten av medlen av arbetet och objektet som bearbetas. Uppåtgående övergångar orsakas av reflektioner över medlen för att utföra verksamheten eller reflektioner mot objektet. Dessa reflektioner kan uppstå på grund av sammanbrott (breakdown) eller avsiktligt ändrande av fokus.

Neråtgående övergångar orsakas av nya lösningar på motsättningar och problem, och även innefattande införandet av lösningen på den lägre nivån som resultat.

Bardram [1, s.97] sammanfattar med två nödvändiga design val vid utformandet av datorstött grupparbete:

“...support collaborative work activities at all three levels, and the dynamic transition between the levels, and integrate this support for dynamic cooperation an the support for the object of work.”

Vidare menar Bardram att design av datorprogram som stöder något arbete inom en organisation är i huvudsak en utvecklande verksamhet av arbetssättet och behov, likväl som det är ett verktyg för tillämpning och införandet av rutiner för arbetet. Svårigheten i att skapa ett fungerande program som stöder CSCW är många. Olika angreppssätt har tagits utifrån vilka olika program har utvecklats, Kaptelinin [9] nämner fyra olika

synsätt som använts för att förenkla gruppaktiviteter:

- *Personal information management (PIM)*, exempelvis Microsoft Outlook, Lotus system är elektroniska versioner av traditionella organiseringsverktyg såsom kalendrar, adressböcker. Nackdelen är att organisationen av aktiviteter är skilt från organisationen av arbetsobjekten såsom filer och arbetsverktyg. Detta innebär att det blir svårt att koppla aktivitet till resurser.
- *Detected project spaces*, är avskilda delar av det virtuella arbetsutrymmet. Kan inbegripa att skapa en folder för ett specifikt projekt till att innehålla alla resurser för att lösa en uppgift. Här finns program som Task Gallery och Manufactor, nackdelen är att det krävs organisation, användarna måste sätta upp ett projekt, arrangera nödvändiga resurser och regelbundet uppdatera informationen.
- *Communication-based virtual work*, exempelvis ContactMap och TaskMaster de organiserar projektrelaterade resurser runt kontakter eller kommunikationstrådar som gör det möjligt att automatiskt koppla nya meddelanden och filer till existerande projekt. Nackdelen är att kontaktnätet inte alltid överensstämmer med det aktuella projektet, då flera projekt kan vara på gång samtidigt och ha olika medlemmar och då måste användaren definiera dokumenten manuellt.
- *Nonhierarchical architectures of information spaces*, exempelvis Lifestreams och Presto, Lifestreams organiserar dokument efter deras kronologiska ordning och kombinerar dokumentorganisationen med organisationen av individuella och gruppaktiviteter.

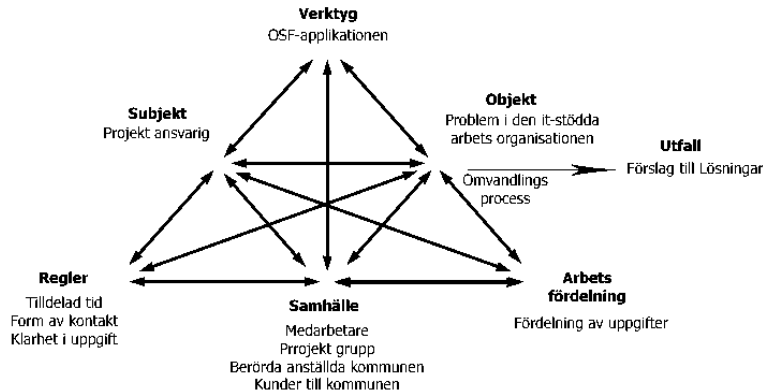
[9].

Sammanfattningsvis menar Kaptelinin att ett CSCW program skall stödja många olika sätt att arbeta på och användaren skall så lite som möjligt behöva lägga ned tid på att administrera systemet. Utifrån detta utformar Kaptelinin [9] tre krav för system som stödjer gruppverksamhet:

- Sammankoppla personlig informationshantering, kommunikation och hantering av verktyg och material.
- Kraftsamla på faktiska arbetsmetoder.
- Minimera administrationen (overhead) och låt användarna skapa sina egna projekt miljöer.

### 3.2.6 OSF-projektets motsättningar

För att visa hur verksamhetsteorin kan kopplas till vår uppgift tillämpas Engeströms modell för att analysera en verksamhet där OSF-applikationen används som verktyg. Vi har utgått från att använda OSF-projektet som verksamhet, då med projektansvarig som subjekt och OSF-applikationen som verktyg. Som objekt ser vi olika problem i den it-stödda arbetsorganisationen där utarbetade förslag till lösningar blir utfall. Som samhälle ses projektmedlemmar från kommunen, kommunarbetare och kunder till kommunen som i slutändan blir alla kommunmedborgare. Som regler kan ses tilldelad tid för projektet, vilken form kontakter tas med olika intressenter och klarhet i uppgifterna, det vill säga vilken information om uppdraget som har gått ut till de olika intressenterna och hur de har uppfattat informationen.



Figur 3.6: Beståndsdelar från centrala aktiviteten i ett OSF-projekt.

Med detta som utgångspunkt och med tyngdpunkten på användarna och OSF-applikationen identifierar vi ett antal motsättningar i ett OSF-projekt (se tabell: 3.1). Nivåerna av motsättningar är beskrivna i fig 3.4.

Några av de motsättningar som vi har identifierat kan kopplas direkt mot hur verktyget utformas och den process som utvecklar verktyget. Dels har vi motsättningar i hur vår applikation är utformad: skall applikationen ha få eller många funktioner och då bli enkelt eller komplicerad att använda och hur mycket bättre blir applikationen med fler funktioner något som antagligen innebär att vi får en större utvecklingskostnad; skall alla ha åtkomst till dokument eller skall den som har skapar dokumentet ha rättigheterna. Vi har även den instrumentproducerande verksamheten där vi kan se: motsättningar att använda Lotus Domino som grund för applikation vs att fritt utforma ett gränssnitt; hur användarrepresentanterna för prioritet i sitt engagemang dagliga arbetet vs utvecklingsarbetet, där arbetet med att leverera Luna3 och hade en hög prioritering.

### 3.3 Relaterade program

Följande tre program har analyserats i syfte att samla erfarenheter som kan vara användbara som grund för vår design: ContactMap [18], Presto [5] och UMEA [9].

Contact Map har utgått från personers kontaktnät för att organisera kommunikation men är även tänkt att användas till andra uppgifter exempelvis att organisera tillgången till dokument och filer.

UMEA utgår från projekt för att få tillgång till resurser kopplade till projektet, resurserna kan vara dokument, mappar, webbsidor eller kontakter. De har arbetat med olika vyer för att visa programmet: Målet har varit att integrera personlig informationshantering, kommunikation och hanteringen av verktyg. Hanteringen har utgått från

Tabell 3.1: Identifierade motsättningar i ett OSF-projekt.

Verksamhetsdel	Motsättning	Kommentar
<b>Nivå 1</b>		
Projektledare	Insatt arbete vs utfall	Vad kan tas betalt för?
Projektgrupp-medlemmar	Förändringar i egen roll och arbetsuppgifter både under arbetes gång och efter genomfört projekt	Vad vinner gruppmedlemmen att vara med ett projekt? Lön, status vs arbetsinsats
OSF-applikationen	Individuellt vs gemensamt program Vilka funktioner?	Vad är effektivast?  Vad behövs? Tid och resurser?
<b>Nivå 2</b>		
Projektledare vs projektgrupp	Vad är viktigt Tid, plats och regler för möten och kontakter Förstå varandras "språk"	
Projektledare vs OSF-applikationen	Automatiserat vs styrt av användaren Tid med programmet vs vilken tid som sparas	
Projektgrupp vs OSF-applikationen	Automatiserat vs styrt av användaren Lättillgängligt vs svårtillgängligt	
<b>Nivå 3</b>		
OSF-projekt vs effektivare OSF-projekt	Effektivare verktyg OSF-projekt vs linjeorganisationen	Tid tilldelning Personal tilldelning
<b>Nivå 4</b>		
Objekt	Motsättningar om vad som är problem i organisationen	
Instrument	OSF-applikationens nytta för uppgiften Användarrepresentantens prioritering i utvecklingen vs andra uppgifter Programmets status vs andra program	
Subjekt	Projektledaren insats vs andra uppgifter	
Regler	Arbetsätt OSF-projekt vs linjeorganisationen och andra projekt	

det arbete som användaren utför och administrationen av hanteringen har minimerats. Problem som har visats sig är, merarbetet med att rensa bort irrelevanta resurser som inte skall kopplas till projektet

Presto låter användaren organisera och söka efter dokument genom att olika attribut som uttrycker egenskaper, kopplas till ett dokument.

I dessa lösningar kan vi se gemensam ansats att frångå det traditionella mappsystemet. Vi kan även se en strävan att ge användaren flexibilitet genom att erbjuda tillgång till dokument och resurser på olika sätt.

### 3.4 Sammanfattning

Sammanfattningsvis är det ett antal aspekter som vi tagit med i fortsatt utvecklingsarbete. Vi frångår inte den layout som återfinns i många Lotus program, som att arbeta med tabeller med flikar och att ha en knapprad i lägst upp på formuläret. Den ansats som andra utvecklare har haft att ge användaren olika sätt att visa och söka dokument har även vi arbetat efter. Vi anammar verksamhetsteorins synsätt och ser vårt program som ett verktyg att användas i arbetet. Ett verktyg som kan utvecklas vidare men också ge upphov till nya sätt att arbeta på, vilket fordrar ett dynamiskt program. Vidare skall programmet enligt vad Bardram [1] menar stödja grupparbetsaktiviteter enligt verksamhetsteorins hieratiska aktivitetsstruktur, och den dynamiska övergången mellan dessa. Vi skall även sträva efter att inte ha en alltför tekniskt styrd kommunikation i programmet, eftersom vi har den uppfattningen att många användare kan anse sådana funktioner enerverande.

## Kapitel 4

# Kravspecifikation

Med utgångspunkt i innehållskrav från Lotsa 1.1, användbarhetsmål och användbarhetsprinciper utarbetade av Nielsen [15], samt vår förstudie, utarbetades krav för applikationen. Kraven indelades i fem olika områden: applikationskrav, krav för att stödja processen (OSF-projektet), krav för att stödja kunden, krav för presentation mot kunden samt programmeringskrav.

### 4.1 Användbarhetsmål

Eftersom användaren skall stå i fokus i utvecklingsarbetet använder vi dessa användbarhetsmål, beskrivna av Preece [15], som utgångspunkt vid utformandet av vår kravspecifikation.

- *effectiveness*: Produktivt att använda, kan uppgiften som systemet stödjer utföras effektivare?
- *efficiency*: Är systemet effektivt för användaren?
- *safety*: Hindrar systemet användaren att göra fel, eller tillåter systemet att användaren gör fel och låter användaren ångra åtgärden?
- *utility*: Är systemet effektivt i vad det utför, skapas rätt utfall?
- *learnability*: Hur lätt är det att lära sig systemet?
- *memorability*: Hur stöds möjligheten att komma ihåg hur man utför arbetsuppgifter?

[15]

### 4.2 Användbarhetsprinciper

Andra punkter som vi stödjer oss på i utformandet av kravspecifikationen är tio användbarhetsprinciper utvecklade av Nielsen och hans kollegor [15].

1. *Synliggör status*: Håll användaren informerad om vad som händer, ge feedback.
2. *Likhet mellan systemet och verkligheten*: Använd användarnas vardagspråk, använd ord, fraser och koncept kända av användarna, istället för systemtermer.

3. *Användarkontroll och frihet*: Tillåt användarna att på ett enkelt sätt navigera tillbaka i programmet.
4. *Konsekvent och standardiserat*: Låt inte användaren undra om olika ord, situationer eller handlingar betyder samma sak.
5. *Hjälp användaren känna igen, hitta och avhjälpa fel*: Använd vanligt språk för att beskriva fel.
6. *Skydda mot fel*: Undvik att programmet tillåter fel.
7. *Igenkännande istället för ihågkommande*: Gör objekt, handlingar och val synliga.
8. *Flexibelt och effektivt att använda*: Använd genvägar som är osynliga för nybörjaren men som möjliggör för mer vana användare att utföra en uppgift snabbare.
9. *Estetisk i minimalistisk design*: Undvik information som ej är nödvändig.
10. *Hjälp och dokumentation*: Erbjud hjälpinformation som är lättåtkomlig och erbjuder hjälp stegvis så den blir lätt att följa.

### 4.3 Valfrihet för användaren

Vi vill ge användaren valfrihet genom att erbjuda många olika sätt att nå information. Detta möjliggörs till viss del av att vi arbetar med databaser, vilket medför att vi med sökning och filtrering kan visa upp information på olika sätt. Vi skall även ge användaren möjlighet att välja graden av information som visas.

Denna teknik kallas *progressive disclosure*, olika mycket av gränssnittet och funktioner visas, beroende av hur mycket användaren vill se eller vad som programmet tror att användaren vill se. Meningen är att denna begränsning av vad som visas skall göra det lättare för användaren att uppfatta den information som visas, en annan aspekt är att plats kan sparas för det som skall visas på dataskärmen.

### 4.4 Specifikation

Vilka krav som ställs på gränssnittets utformning redogörs nedan tillsammans med hur varje krav ska tillmötesgå i designen. Dessutom strävar vi efter att arbeta efter de tankegångar som framkom i vår förstudie (se avsnitt 3.4).

#### Applikationskrav

Här är allmänna krav som är tänkta att göra programmet lätt att använda för användarna.

- *Användarvänligt*: Synligt vad som kan göras, navigering med tangentbord med tabblägen och kortkommandon och det skall vara lätt att lära sig, genom att programmet är logiskt uppbyggt och har ett naturligt flöde och en liknande struktur i hela applikationen.
- *Enhetlighet i layout färger och typsnitt*: Samma uppgift placeras på samma ställe i olika formulär.



- *Navigation*: Knappar och menyer placeras på samma ställe logiskt. Tabbningslägen skall följa arbetsgången genom formulären.
- *Gruppering*: Uppgifter som hör ihop skall grupperas tillsammans. Avskiljning kan förstärkas med linjer och ramar. Återkommande element återanvänds om möjligt på samma plats.
- *Redigering*: Visning av uppgifter skall i första hand ske i labels. Redigerbara fält skall skilja i utseende från icke redigerbara. Om många uppgifter skall matas in skall det gör i ett särskilt redigeringsläge.
- *Synlighet av status*: Sparad eller ej sparad, skillnad på redigering och visningsläge och indikation var användaren befinner sig. Feedback vid val och på att systemet jobbar (om en process tar lång tid).
- *Effektivera*: Uppgifter som skrivs in av användaren skall ärvas eller arv ges som valmöjlighet.
- *Hjälp*: Hjälp skall ges i programmet med "tooltips" och hjälpavsnitt.

### Stödja processen

Här ingår krav som stödjer CSCW och ett OSF-projekt.

- *Översikt*: Översikt i tre kategorier: personlig eller grupp, tidsmässig och projektoversikt. Vi skall använda den flexibilitet som en databas tillhandahåller och ge användaren både ett antal fasta sätt att komma åt dokument men även möjlighet till egna sökningar.
- *Effektivera*: Automatisera behandling av indata och skapandet av dokument, skall lösas genom automatisering av rutinprocesser och mallar för återkommande uppgifter.
- *Underlätta kontakter*: Det skall finnas personregister med e-post och telefonnummer och kunskapsområden. Programmet skall vara en kanal för att få in och ut information från projektmedlemmar och vid undersökningar. Möjliggöra Internet baserade undersökningar och automatisering av inkommande materiel.
- *Ärendehantering*: Tillgång till stöddokument och länkar, mallar till dokument, då både fotnoter och hela dokument. Ge översikt av ärendehantering och automatisera flödet.
- *Utvecklingsbart*: Mallar och stöddokument skall på ett enkelt sätt kunna ärvas till nya projekt.
- *Kontroll och tillgänglighet*: åtkomst över Internet genom inloggning möjliggör god tillgänglighet.
- *Säkerhet*: Dokument på server möjliggör central säkerhetskopiering.

### Stödja kunden

Dessa krav stödja den del som skall lämnas vidare till kunden.

- *Stöd för arbetet*: Ge stöd för kunden att genomföra lösningar genom att ge en bra översikt över vem som skall göra vad, tidsram och tillgång till relevanta dokument.
- *Underlätta kontakten med projektmedlemmar*: Detta ska åstadkommas både genom systemets funktioner och att projektmedlemmars telefonnummer och e-post läggs in i personformuläret.

### Stödja presentation mot kunden

- *Effektivisering*: Mallar för presentation.
- *Underlätta kontakter*: Programmet skall vara en kanal för att distribuera information ut till alla projektmedlemmar.

### Programmeringskrav

- *Konsekvent namngivning*: Inga förkortningar, börja med liten bokstav, stor bokstav vid sammansättningar.
- *Återanvändning av kod*: Kod skall återanvändas från Luna3, om möjligt även inom det nya programmet.

#### 4.4.1 Krav på ingående element

Formulärens innehåll kallar vi element. De kan matas in i formuläret, ärvas från andra formulär eller skapas av systemet. Förslag på element till de olika formulären visas i tabell 4.1.

Tabell 4.1: Formulärens *Element*.

Formulär	Element	Kommentar
Organisation	Namn Adress Telefon Organisationsnummer Platser	
Plats	Namn Adress Telefon Personer	Namn på tex avdelning
Person	Efternamn Förnamn Telefon Telefon mobil e-postadress Arbetsstittel Intresseområden Kunskapsområden Foto Adress Arbetsplats	Arbetsrelaterat Arbetsrelaterat
Person intern	Personnummer Anställningsnummer Tidsrapportering	
Grupp	Namn Medlemmar Skapad av Övriga uppgifter	Gruppens namn Personer
Aktivitet	Namn Mål Ansvarig Initierat av Starttid Klart till Status Medlemmar Avvikelser Kommentarer Kopplade dokument Kopplade styrdokument Web-resurser Redovisas för Godkänns av Aktivitet Olika utskicks status  Svarsstatus Loggdokument Typ av process	Möjlighet att skapa grupp  Länkar  Gemensam och privat För kännedom Svar krävs (till datum) Kommentera gärna (till datum)
Ärende	Samma som aktivitet Flödesmallar	



## Kapitel 5

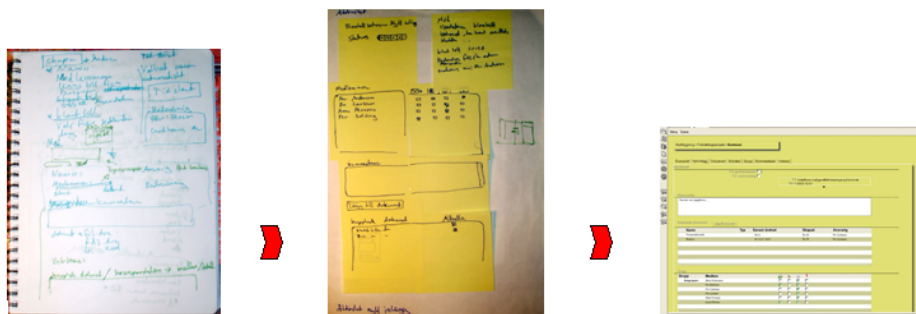
# Utveckling

Utvecklingen av programmet har gjorts genom att arbeta med skisser och prototyper med utgångspunkt från konceptuell design, interaktionsdesign, informationsdesign och grafisk design. Där konceptuell design är de övergripande frågorna hur gränssnittet skall vara uppbyggt, interaktionsdesign rör frågor om hur användaren samspelar med programmet, informationsdesign hur informationen visas och delas upp och grafisk design behandlar färg och form och val av typsnitt.

Prototyputvecklingen har innefattat tre nivåer; A4 med post-it lappar, formulär i Domino och utveckling av dominoformulär med Macromedia Flash MX.<sup>1</sup>

Två scenarios har tagits fram för att ge en bild av vad programmet kan användas till (se Bilaga A).

Utvärdering har utförts löpande dels genom egen utvärdering där vi har gått igenom olika sekvenser av programmet som representerar olika moment, samt genom "Quick and Dirty" testning med representanter från Lotsa.



Figur 5.1: Arbetsgång, visar tre faser i utvecklingsarbetet: skiss, A4 och Lotus Domino formulär.

<sup>1</sup><http://www.macromedia.com/>

## 5.1 Skisser

Skisserna är i första hand till för konceptuell design och interaktionsdesign, vi vill visa hur användaren kommer till olika formulär i programmet. Även utreda hur användaren på ett förståeligt sätt kan använda de olika och funktionerna i programmet. Arbetssättet är att helst göra flera olika skisser för att få olika idéer och infallsvinklar.

## 5.2 Post-it lappar

I nästa steg användes post-it lappar med olika element på som fästs på A4 papper och därigenom på ett lätt sätt kan flyttas runt för att pröva olika lösningar, A4 arken representerar is in tur formulär eller formuläruppslag så detta medger att man kan pröva att navigera mellan olika formulär. Det riktar sig mot konceptuell design, interaktionsdesign och informationsdesign. Den problemställning som visar sig vara svårast är hur åtkomsten till olika aktiviteter skall vara. Skall alla i gruppen ha tillgång och kunna ändra på innehållet i en aktivitet eller skall ansvarig för aktiviteten äga rättigheter och andra gruppmedlemmar enbart komma med utlåtanden. I större drag skall programmet underlätta respons på individuellt arbete eller också skall hela gruppen kunna arbeta med aktiviteten och dess kopplade dokument.

## 5.3 Formulär i domino

Härifrån gjordes fristående formulär i Domino med frågor runt interaktionsdesign, informationsdesign och grafisk design. Arbetet i Domino gav både fördela och nackdelar. Fördelen var att vi ser vad som kan göras och vi får ett färdig liknande resultat. Men det är svårt att få vissa funktioner som vi tänker att de skall vara då vi begränsas i viss mån av kunskaperna om verktyget. En annan nackdel är att vyerna, som är kanske den viktigaste delen av applikationen inte fungerar innan hela systemet är klart.

## 5.4 Vidareutveckling i Macromedia Flash

För att kunna visa hur vyerna skall fungera, användes Macromedia Flash. Vi använde de formulär som har utarbetats i Domino som bakgrund och lade in text och symboler för att visa hur det är tänkt att fungera i den färdiga applikationen.

## 5.5 Dokumenthantering

Applikationen är tänkt att hantera olika sorts dokument dels det vi kallar biblioteksresurser bestående av referenslitteratur såsom styrdokument eller sparade länkar till webbresurser. Sen har vi de dokument som det arbetas med, skapade av användaren eller importerade in i systemet. En viktig aspekt är hur dessa dokument skall hanteras, vi har i huvudsak tittat på två lösningar som kan integreras med vårt system. Det ena är Lotus som har en färdig databas, Microsoft Offices Library som hanterar Offices dokument. Denna ger möjlighet att öppna dokumenten i Lotus miljön. Det andra är SWING Integrator<sup>®</sup> som öppnar dokumenten fristående, men har fördelen att kunna utveckla integrationen mellan Lotus i Microsoft Word och Excel<sup>®</sup> dokument. Vi väljer därför SWING Integrator<sup>®</sup> se avsnitt 1.2.2). Inledningsvis var användarens dokument kopplade

till aktiviteten och SWING Integrator<sup>®</sup> integrerat med aktiviteten men för att kunna få en flexiblare hantering samlar vi alla dokument i en egen databas så dokumenten får en lösare koppling till aktiviteten. Dokumentformuläret öppnas upp integrerad i aktivitetsformuläret men kan även öppnas upp i ett eget fönster något som kan vara lite förvirrande men ger fördelan att vi får en större arbetsyta.





# Kapitel 6

## Resultat

### 6.1 Uppbyggnad

För varje nytt projekt skapas en ny version av programmet dvs. en ny databas, här finns möjligheten att ära dokument och mallar från tidigare projekt. Programmet indelas i två databaser, en för att organisera personer och deras aktiviteter och en för alla dokument. Denna uppdelning underlättar användandet av SWING Integrator<sup>®</sup> och underlättar även möjligheten att överföra dokument till nya projekt.

### 6.2 Utformning

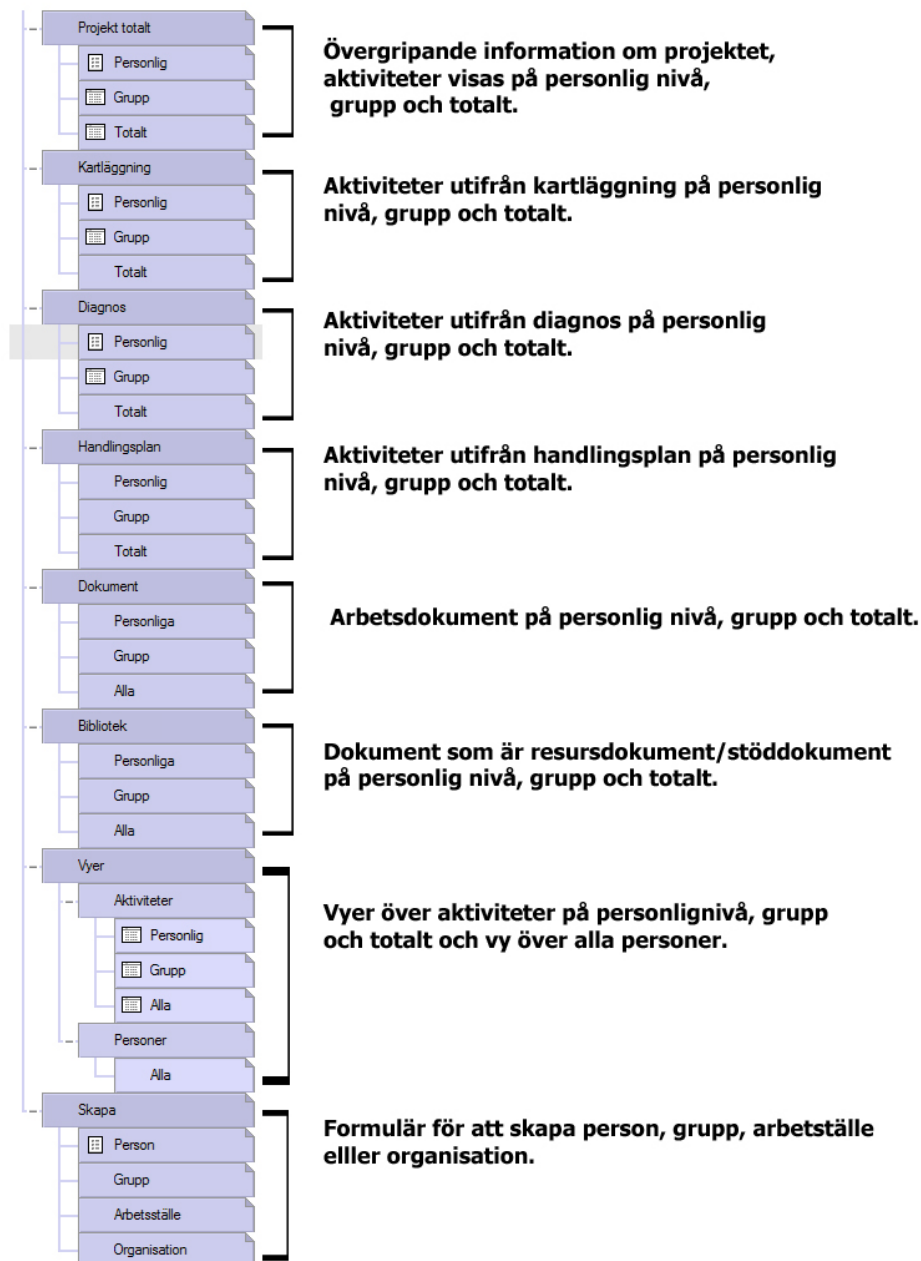
Den indelning i olika faser som arbetas efter i OSF-projekt återfinns i programmet. Detta genom att aktiviteter bildas och återkoms under diagnos, kartläggning eller handlingsplan (se bild 6.1). Denna indelning påverkar då även arbetsdokument och styrdokument eftersom de ligger kopplade till aktiviteterna. Användarna kan även direkt komma åt alla dokument genom att gå in på dokument och dess undernivåer. Dessa faser innehåller även skilda kategorier av användare exempelvis ska handlingsplanen avskiljas och användas av kunden som stöd för att genomföra den beslutade handlingsplanen. Det betyder att vissa användare bara har tillgång till aktiviteter under handlingsplan, det blir då även den enda del som synliggörs i navigeringsträdet. Vid informationsinhämtning från t.ex. kunder till kommunen förmedlas formulär i html version. Där inloggningsuppgifter och instruktioner skickas ut med e-post.

#### 6.2.1 Övergripande utformning

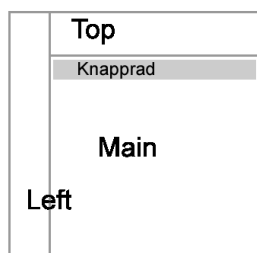
Applikationens normala layout består av tre fönster (se bild 6.2) *Left*, *Top* och *Main*. *Left* är till för navigering av vad som visas i *Main-fönstret* och i *Top-fönstret* visas system medelanden, samt att det skall även vara möjligt att ställa in färg och textstorlek på applikationen. Vi använder *progressive disclosure* i applikationen, först gällande vad som visas i vyerna där vi med en personlig sida bara visar vad som är aktuellt för personen. I aktivitet formuläret har vi en utfällbar informationsruta där användaren kan välja graden av information.

*Aktiviteter*, *personer*, *plats* och *organisation* nås genom att de öppnas från någon vy. Vyer är Lotus sätt att visa upp innehållet i databasen på. De kan byggas expanderbara, då visas en pil till höger om namnet på den kategori som går att expandera. Vyerna ger

## Översikt över navigeringsträd



Figur 6.1: Navigeringsträd, med programmets olika projektfaser.



Figur 6.2: Layouten på programmet.

möjlighet att filtrera vad som skall visas för användaren. Dessa vyer finns integrerade i någon sida eller kan nås direkt genom vy mappen i navigeringsträdet. Integrerade vyer skall även kunna öppnas upp så de upptar hela *Main-fönstret*.

Lotus använder en knapprad i överkant av huvudfönstret något som kan anses som standard i lotus applikationer. Vi går inte emot denna standardisering utan placerar knappar för val i denna knapprad men för att få ett naturligare flöde genom en del formulär sätter vi in knappar även i formuläret.

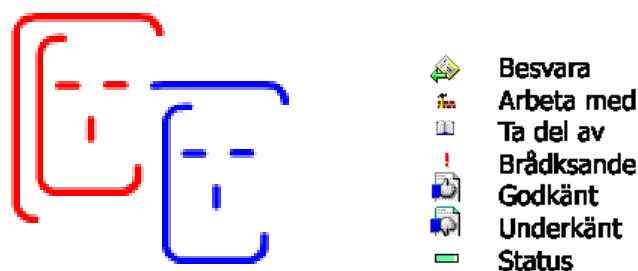
Organisationen av personer innefattar tre nivåer *Organisation*, *Plats* och *Person*. Överordnad nivå måste ha skapats före det att en undernivå skapas. Underliggande nivåer skapas från överliggande, men det är också möjligt att skapa exempelvis en *Person* och välja *Plats* tillhörighet i *Person* formuläret. Även fristående personer kan också kopplas till aktiviteten de får då uppdatering med e-post vid varje inlägg.

*Ärende* och *Aktiviteter* skapas hierarkiskt med ett ärende på första nivå. Uppbyggnaden av ett *ärende* och *aktivitet* är lika det är bara namnet som skiljer beroende på i viken nivå de skapas. Normalfallet är att den som skapar en *aktivitet* blir ansvarig för den. Men det är även möjligt att skapa en aktivitet och tilldela en annan person som ansvarig. Till aktiviteten kan bibliotek resurser kopplas från den fristående databasen samt nya arbetsdokument skapas eller läggs in från externa källor. För förståelsen bör vi nämna att alla dokument är ett Lotusformulär med i många fall ett Microsoft Word eller Excel<sup>®</sup> dokument kopplat till sig. Arbetsdokument kan även skapas från mallar, och då fältvärden föras direkt in i valt dokument.

Den personliga sidan i projekt total är som startsida när programmet startas. Här visas vilka aktiviteter som är aktuella att arbetas med och vad andra användare förväntar sig att användaren skall bistå med. Aktiviteter kan även nås direkt från vyer.

### 6.2.2 Ikoner, fonter och färger

Vi har valt att använda de ikoner som finns färdiga i Lotus förutom de som indikerar aktivitetens förlopp och programikonen (se figur 6.3). Fonter, textstorlek och färg kommer att kunna sättas av användaren något som det finns stöd för i utvecklingsverktyget.



(a) Bild a. Programikon

(b) Bild b. Ikoner för aktiviteter

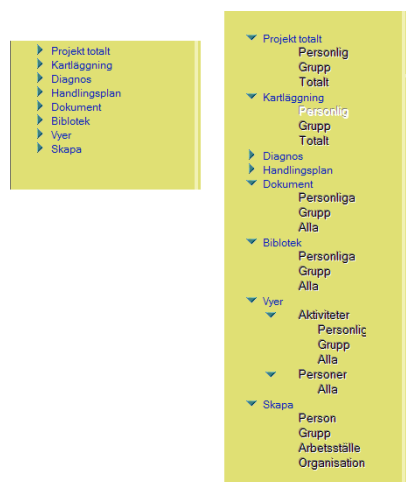
Figur 6.3: Ikoner i applikationen.

### 6.2.3 Datum

För att få en bättre tidsuppfattning runt aktiviteter och dokument visar vi datum i olika format beroende på närheten i tiden. Datum inom en vecka visas veckodag och månad: **Tis 21**, emedan övriga datum visas år, månad och dag: **06-02-21**.

### 6.2.4 Dokumentdatabas

Dokumentdatabasen är fristående och integrerad med SWING Integrator<sup>®</sup> vilket ger den SWING Integrators<sup>®</sup> lösningar. Både Dokument och Biblioteksresurser ligger i denna databas med de visas i skilda integrerade vyer. Varje i vy man kan även öppnas upp i som ett fristående fönster.



Figur 6.4: Navigeringsträd, visat komprimerat och expanderat.

### 6.2.5 Navigering

Navigering i navigeringsträdet sker med pil-tangenter eller mus. Den länk som är aktuell länk blir vit och lite fetare. I knappraden formulären stöds navigeringen med att när

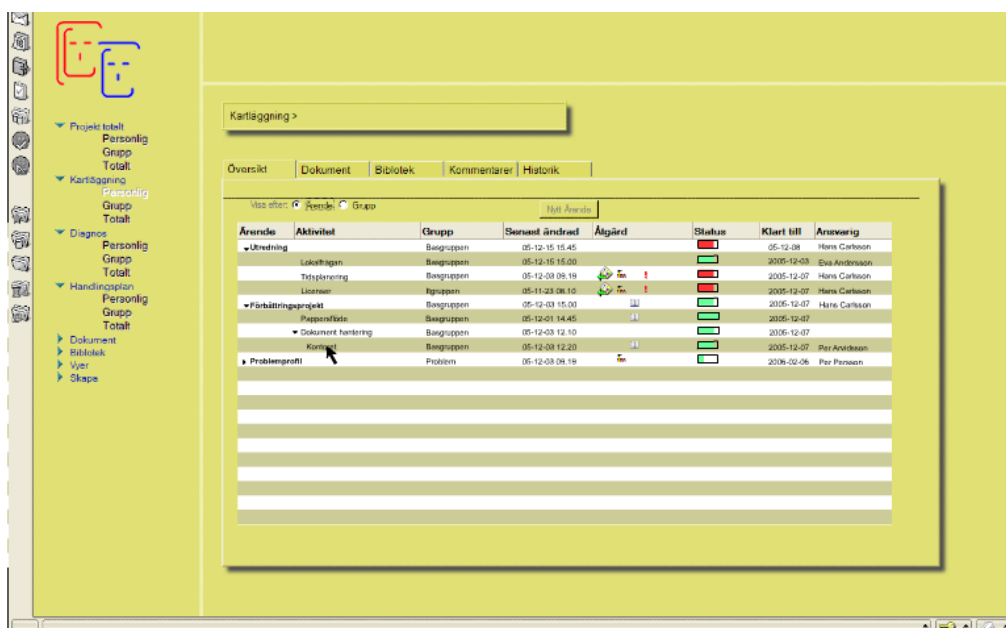
alt-knappen hålls ned visas tangent val för respektive knapp som en textruta (tooltip). Det gäller även de är inbakade formulären vid dokumenthantering som får en egen knapprad överst i formuläret. I övrigt skall tab-tangenten följa det naturliga flödet genom formulären, vilken möjliggör att navigera enbart med tangentbordet. På de val i navigeringsträdet som har flera undernivåer borde de undernivåer som inte är någon länk vara i samma font som översta nivån för att visa att det inte är någon länk, men programmet tillåter inte den inställningen.

## 6.3 Formulär

Vi går här igenom uppbyggnaden av de viktigaste formulären i applikationen; varje formulär består av ett antal element (se tabell 4.1). De formulär som vi tittar på är *Personlig översikt*, *Aktivitet*, *Person*, *Plats* och *Organisation*.

### 6.3.1 Formulär: Personlig översikt

Detta formulär är den personliga utgångspunkten för att arbeta med aktiviteter och dokument. Den ger användaren olika vägar för att få en översikt och åtkomst till aktiviteter och dokument. Formuläret har ett orienteringsfönster och en tabell med flikar bestående av *Översikt*, *Dokument*, *Bibliotek*, *Kommentarer* och *Historik* (se figur 6.5).



Figur 6.5: *Personlig översikt*, här visad med en översikt över *Ärende* och *Aktiviteter*.

### 6.3.2 Formulär: Aktivitet

Aktivitetsformuläret har en knapprad, orienteringsfönster och en tabell med ett flertal flikar. Den främre fliken i tabellen visar översikt av aktiviteten. Övriga flikar är *Nytt Inlägg*, *Dokument*, *Bibliotek*, *Grupp* och *Historia*.

Tabell 6.1: Översiktsformulärets innehåll.

Flik	Element	Kommentar
Översikt	Visa efter	Radioknapp för val av visningsläge Ärende eller Grupp.
	Nytt Ärende	Knapp, öppnar nytt fönster med nytt Ärende, om översikten är i projekt totalt öppnas ett dialog fönster för att välja projektfas.
	Integrerad vy	Integrerad vy med aktiviteter som personen är medlem i aktiviteterna visas efter Ärende eller Grupp.
Dokument	Integrerad vy	Dokumenterna är sorterade med ärende och datum som förval, men det går även att sortera efter dokumentets namn, senast ändrad eller dokumentets ägare. Valt dokument öppnas upp i eget fönster. Om ett dokument har en fil kopplat till sig visas det med en ikon, filen kan öppnas direkt genom att välja ikonen.
Bibliotek	Integrerad vy	Selekterade resursdokument sorterade efter samma princip som under dokumentfliken.
Kommentarer	Text fält	Kommentarer som är listade efter datum, med namn på aktivitet och kommenterare.
Historik	Text fält	Text fält som är listade efter datum, med namn på aktivitet och vem som senaste ändrade aktiviteten.

Kartläggning > Förbättringsprojekt >

Översikt | Nyttinlägg | Dokument | Bibliotek | Bibliotek Grupp | Kommentarer | Historia

Ny Aktivitet

Aktivitetens namn:

Standarddatum: 2006-01-19

Grupp:

Redovisat för:

Kart till:

Godkänns av:

Mål:

Ansvare: Patrik Löfgren

Varningsläge:  Publik  Privat

Spara aktivitet

Figur 6.6: Aktivitetsformulär, här visad vid registrering an *Ny Aktivitet*.

Aktiviteterna är hierarkiskt ordnade med ett *Ärende* som högsta nivå, denna hierarki visas i orienteringsfönstret. Vidare finns ett fönster med aktivitetsinformation, den visar kort fakta om aktiviteten och återfinns i varje flik. Informationen finns i två versioner, en kortfattad med mål och deadline. Samt en utökad version med, *Redovisas för*, *Godkänns av* samt datum för respektive och ansvarig för aktiviteten. Den kortfattade versionen är förval medan den utökade versionen väljs av användaren.

När en ny aktivitet skapas visas ett formulär för inmatning av uppgifter för den nya aktiviteten. Aktiviteten skapas som en undernivå till en aktivitet eller ett ärende och ärver ett flertal uppgifter och dessa ges som förval för vissa fält i den nya aktiviteten.

Under fliken *Översikt* visas en översikt av aktiviteten, (se figur 6.7). Här visas aktivitetens status och den personliga aktivitet andra vill att användaren skall utföra, denna information förmedlas med hjälp av olika ikoner. En statusikon visar statusen på aktiviteten, givet av en procentsats som sätts av ansvarig över aktiviteten. Statusikonen kan vara grön eller röd beroende om tidschemat ser ut att hållas (grön) eller inte (röd). Färgen sätts av systemet beroende på den tid som finns kvar mot den del som är kvar att göra och den totala tiden för aktiviteten. *Publik* eller *privat* visas bara för ansvarig och sätts av ansvarig. Ikoner som visar väntad personlig aktivitet är de samma som återfinns i aktivitetsvyn. Dessa neutraliseras (skuggas) av användaren genom att trycka på ikonerna och försvinner när aktiviteten sparas.

Aktivitet	Grupp	Senast ändrad	Åtgärd	Status	Klart till	Ansvarig
Arbetsutveckling	Kontoret	05-12-21 16:51			09	Per Andersson
Famers	Kontoret	05-12-21 12:21			21	Per Andersson

Figur 6.7: Aktivitetsformulär, här visad med *Översikt*.

Under fliken *Nytt inlägg* (se figur 6.8) kan användaren göra kommentarer och se de dokument som är kopplade till aktiviteten. Här sätts även vad andra gruppmedlemmar skall göra med aktiviteten alternativt är: *Besvara*, *Arbeta med* och *Ta del av*. Här kan även statusen *Brådskande* sättas. Aktiviteten kan även skickas för *Godkännande* eller för *Redovisning* genom att kryssa i respektive kryssruta

Stäng Spara

Kartläggning > Förbättringsprojekt > **Kontoret**

Översikt | Nytt inlägg | Dokument | Bibliotek | Grupp | Kommentarer | Historia

Kontoret

För godkännande    
 För redovisning

Må indelbara möjliga effektiviseringar på kontoret  
Sestil 2005-12-07

Kommentar

Titta över uppgifterna om ...

Kopplade dokument

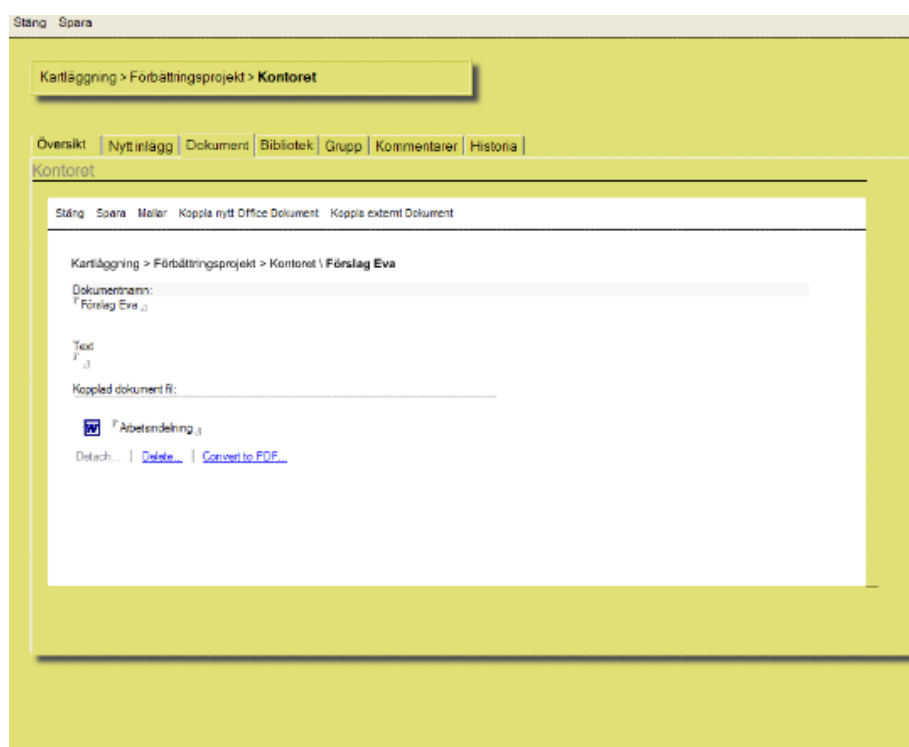
Namn	Typ	Senast ändrad	Skapad	Ansvarig
Personallösning		06:21	Tis 09	Per Carlsson
Rutiner		05-12-21 16:51	Tis 09	Per Carlsson

Grupp

Grupp	Medlem				
Besgruppen	Sina Andersson	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Per Bohman	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Per Carlsson	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Pia Larsson	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Mats Persson	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Anna Öhrman	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figur 6.8: Aktivitetformulär, här visad med *Nytt Inlägg*.





Figur 6.9: *Aktivitetsformulär*, här visad med fliken *Dokument* med sparad dokument.

Under fliken *Dokument* visas de dokument som är kopplade till aktiviteten. När ett nytt dokument skapas öppnas det upp integrerad i aktivitetsformuläret med sin egen knapprad överst, men det kan även öppnas upp i ett eget fönster om det krävs större arbetsyta. Ett redan sparad dokument öppnas genom att det klickas på i vyn, innehåller det ett kopplat dokument kan detta öppnas direkt utan att formuläret öppnas genom att filikonen markeras istället för filnamnet (se figur 6.9).

Under fliken *Bibliotek* visas olika resurser som är kopplade till aktiviteten. Det kan vara styrdokument, exempel på en tidigare utförd aktivitet eller webbresurser. Nya resurser kan även kopplas till aktiviteten.

Under grupp visas gruppmedlemmarna som är kopplade till aktiviteten.

Slutligen under *Historia* visas datum för ändringar av aktiviteten.

Figur 6.10: *Personformulär*, här visat i inmatningsläget *Ny Person*.

### 6.3.3 Formulär: Person

Formuläret *Person* (se figure 6.10) har en knapprad, orienteringsfönster och en tabell med ett flertal flikar. Orienteringsfönster visar personens namn och dennes hierarki i organisationen. Knappraden har knappar för stäng spara samt föregående och nästa. Formuläret är i första hand till för att införa personuppgifterna för en person. När en person öppnar upp sin egna personformulär första gången skall inmatningsläget visas.

### 6.3.4 Formulär: Plats

Formuläret *Plats* är det arbetsställe som personer är kopplade till. Formuläret har liknande uppbyggnad som formuläret *Person*. Med flikar *Ny Plats/Plats*, *Personer*, *Kommentarer* och *Historik*.

### 6.3.5 Formulär: Organisation

Formuläret *Organisation* visar uppgifter om organisationen, har *Platser* kopplade till sig. Består av knapprad informationsfönster och en tabell med olika flikar. Formulärets

Tabell 6.2: *Aktivitetsformulärets innehåll.*

Flik	Element	Kommentar
Ny Aktivitet	Aktivitets namn	Namn på aktivitet.
	Grupp	Medlemmar i aktiviteten, val från lista där arv är förval medlemmar kan både vara personer skapade i applikationen eller fristående personer som väljs in från Lotus Notes adressbok.
	Klart till	Klart till datum.
	Mål	Mål med Aktiviteten.
	Startdatum	Datum när Aktiviteten skall påbörjas, dagens datum som förval.
	Redovisas för	Namn på Grupp eller Person som aktiviteten skall redovisas för, val från lista där tom är förval.
	Datum	Datum för redovisning.
	Godkänns av	Namn på Person eller Grupp som skall godkänna Aktiviteten.
	Datum	Datum för godkännande.
	Ansvarig	Namn på ansvarig för Aktiviteten, skaparen som förval.
	Visningsläge	Val av Aktiviteten skall var publik eller privat.
Översikt	Status	Statusikon visar status.
	Visa	Radioknapp med val för vad som skall visas i vyn och bestämmer mål för knapp.
	Visavyn	Visar Dokument eller Biblioteksresurser.
	Skapaknapp	Ett snabbval för att skapa nya versioner av respektive val gjort med knapp, aktivitet visas som förval.
	Redigera aktivitet	Knapp för att uppdatera uppgifterna för Aktiviteten, samma formulär som när aktiviteten skapades visas, här kan de tidigare givna uppgifterna ändras.
Dokument	Koppla nytt Officedokument	Knapp med undermeny som skapar val med Officedokument. Andra typer av dokument kan inte skapas härifrån, de skapas externt och kopplas till aktiviteten.
	Koppla externt Dokument	Knapp som öppnar en "koppladokument" ruta för att välja det dokument man vill koppla till aktiviteten.
	Dokumenticon	Visas till vänster om filnamnet, öppnar dokumentet i ett frisående fönster.
	Detach...	Öppnar en dialogruta för att spara ned ett kopplat dokument på annan plats.
	Delate...	Öppnar en dialogruta för att ta bort dokument.
	Convert to PDF	Öppnar dialogruta för konvertera till pdf format med valmöjligheten att spara på hårddisken, skicka med e-post, spara som nytt kopplat dokument eller att ersätta existerande kopplade dokument.
	Nytt Dokument	Öppnar ett nytt formulär dokument i ett inbäddat skrifvfält. Visas enbart när vyn med dokument visas.
Bibliotek	Ny bibliotekresurs Vy	Lägger till en resurs till biblioteket. Vy med kopplade bibliotekresurser.
Grupp	Redigera grupp	Redigerar grupp.
	Vy	Vy över gruppmedlemmarna.
Kommentarer	Text	Fönster som visar kommentarer, varje kommentar har datum och författare.
Historia	Text	Fönster som visar datum för ändringar av aktiviteten.

Tabell 6.3: *Personformulärets* innehåll.

<b>Flik</b>	<b>Element</b>	<b>Kommentar</b>
Ny Person	Förnamn	Förnamn på personen.
	Efternamn	Efternamn på personen.
	Telefon	Telefon till arbetsställe.
	Mobil	Telefon nummer mobil.
	E-post	E-postadress.
	Titel	Titel på Person.
	Arbetsplats	Arbetsplats, arv som förval värdet är beroende under vilken plats ny Person öppnas.
	Adress	Adress till personens arbetsplats, arv som förval.
	Postnummer/Ort	Postnummer och ort.
	Intresse områden	Checkbox med intresseområden.
	Lägg till foto	Möjlighet att lägga in ett foto.
	Spara uppgifter	Sparar Personen, visar visningsläge.
	Aktiviteter	Vy
Dokument	Vy	Integrerad vy med Dokument som Personen är ansvarig för.
Grupper	Vy	Integrerad vy med Grupper som Personen är medlem i.
Kommentar	Fält	Fält med kommentarer om Personen Tex. om en Person kommer att vara tillgänglig vissa tider.
	Ny kommentar	Knapp med dialogruta för att skriva nya kommentarer.
Historiam	Fält	Fält med senast ändrad och av vem.

Tabell 6.4: *Platsformulärets* innehåll.

<b>Flik</b>	<b>Element</b>	<b>Kommentar</b>	
Ny Plats/ Översikt Plats	Namn	Namn på Platsen.	
	Organisation	Namn på Organisationen, arv som förval	
	Adress	Adress till Platsen, arv som förval.	
	Telefon	Adress till Platsen, arv som förval.	
	Kontaktperson	Öppnar dialogruta för att hämta namn.	
	Bekrivning	Beskrivning av Platsen.	
	Personer	Ny Person	Knapp som öppnar nytt fönster med Person.
		Inbakad Vy	Vy med personer tillhörande Platsen.
	Kommentarer	Fält	Kommentarer om Platsen
		Knapp	Dialog för nya kommentarer.
Historik	Fält	Visar när Platsen blev skapad och av vem, samt senast ändrad och av vem.	

uppbyggnad är liknande som formuläret *Person*.

## 6.4 Utvärdering mot kravspecifikationer

Vi går här igenom huruvida våra kravspecifikationer kan anses uppfyllda och vad som bör arbetas vidare med.

### Applikationskrav

Applikationskraven är uppfyllda förutom kravet på hjälpdelen som inte är genomarbetad.

### Krav för att stödja processen

Här återstår arbete för att användaren på ett enkelt sätt ska kunna välja ut de dokument och aktiviteter som skall överföras till nya projekt.

### Krav för att stödja kunden

Målet att ge kunden stöd och underlätta kontakter med projektmedlemmar kan anses uppfyllda. En grafisk översikt över tid och inplanerad verksamheten exempelvis ett Gantschema skulle vara önskvärt.

### Krav för att stödja presentation mot kunden

Grundförutsättningarna för målen är uppfyllda, däremot måste en uppsättning mallar utarbetas till programmet.

### Programmeringskrav

Har följts enligt kraven.



## Kapitel 7

# Diskussion

Målet med vårt utvecklingsarbete är skapa ett program som skall vara ett verktyg för organisationsutveckling, som ger möjlighet att organisera aktiviteter och tillhörande dokument i databaser. Detta ger ett kraftigare verktyg än att bara samla dokument i en katalog. Vidare genom att möjliggöra automatisering av olika moment fås ett verktyg som användaren kan bygga på löpande. Detta är en funktionalitet som bedöms ha en stor efterfrågan på många arbetsplatser där datorstöd används i arbetet i olika former. Vi vill poängtera att programmet inte är något heltäckande organisationsprogram för en arbetsplats utan riktar sig till att hantera mindre återkommande projekt med ett mindre antal medlemmar. Vilka mervärden ger då programmet användarna? På den personliga nivån ger systemet användaren att alla dokument sparas centralt, vilket underlättar säkerhetskopiering, och eftersom dokumenten sparas i databasformat så är urval och olika sökmeter lätt att utföra för att underlätta återkomsten. En annan funktion är att återkommande arbetsuppgifter på olika sätt kan automatiseras. På grupparbetsnivå ger programmet möjlighet att arbeta mot gemensamma dokument och distribuera information på ett enkelt sätt. Vidare ger det projektarbetet en struktur, olika stödfunktioner och en historia som ger möjlighet att utveckla projekten på ett strukturerat sätt.

### Arbetsättet

En av huvudtankarna med ID modellen är att det inte ska kosta alltför mycket tid eller resurser för att få fram den första modellen. Den kan då utvärderas och omformas vidare. Läggs för mycket arbete ned på den första iden kan det vara svårt att överge den för något annat. En nedlagd ide' skall dessutom inte ses som ett misslyckande utan snarare som en grund för fortsatt utvecklingsarbete.

Trots att mycket tid har lagts på inläring av Lotus Domino, har vissa saker ändå varit svåra att få som vi vill utforma dem. Då uppstår frågan är det begränsningar i kunskaperna om verktyget eller är det begränsningar i verktyget? Detta kan vara en nackdel med att utforma designförslag i Domino då vissa begränsningar kan uppstå, men fördelen är att det som har utformats kan implementeras. De delar som har lagts till i Macromedia Flash har antingen implementerats i liten skala eller återfunnits i andra Lotus applikationer eller förankrats hos andra programmerare.

En av de stora frågeställningarna har varit hur flöden med aktiviteter skall hanteras. Skall alla gruppmedlemmar ha till alla dokument? Skall tiden för responsen på ett inlägg tidsbegränsas av programmet? Skall den sättas för varje aktivitet för sig? Skall påminnelse ske med automatik? Skall de även göras över e-post? Frågorna är många, och

svaren ligger nog mest i arbetskulturen på arbetsplatsen. En sak är klar att vi i Sverige jämförelsevis har en mycket platt organisation jämfört med andra länder. Därför kan ett allt för reglerat program kan nog anses otidsenligt av många.

### Gränssnitt

När det gäller gränssnitt för nya användare är det svårt att gå emot det som kan anses vara standard, en ny lösning måste vara bra för att motivera att frångå den som användaren redan är van. På senare år har vi sett ett visst fokus gå över från novisen till de mer professionella användarna. Om en person använder ett program sällan måste ett gränssnitt innehålla uppenbara felaktigheter för att det skall uppfatta dåligt. Däremot kan små felaktigheter ge upphov till onödiga belastningar och dålig ergonomi för en användare som arbetar med ett program dagligen. Ofta är det så att båda kategorierna av användare måste fångas upp, något som kan lösas med skilda gränssnitt *progressive disclosure* 4.3. Detta arbetssätt är ingen nyhet, men tekniken blir mer och mer aktuell då många program har en stor uppsättning av funktioner som kanske aldrig används av normalanvändaren. Svårigheten är dock att veta vad som är lämpligt att visas för olika användare, konsekvensen kan bli att det blir svårare att nå de funktioner eller den information som användaren vill åt. Urvalet kan grunda sig på olika algoritmer något som ibland uppfattas förvirrande för användaren. Ett annat sätt är att låta användaren aktivt välja visningsläge.

Vi använder *progressive disclosure* i olika former i vår design. Först gällande vad som visas i vyerna där vi med en personlig sida bara visar vad som är aktuellt för personen. I aktivitet formuläret har vi en utfällbar informationsruta där användaren kan välja graden av information.

### Tillämpning av teorier

I utvecklingsarbetet har vi använt Engeströms modell för utvecklande arbete. Men tankarna i verksamhetsteorin har även influerat vårt tankesätt. En del i verksamhetsteorin som inspirerat oss är grundtankarna om utveckling och att ingenting är statiskt. Vilket ger en positiv inställning till utveckling, allting kan göras bättre och det faktum att en verksamhet har utförts på ett vist sätt under en längre tid behöver inte betyda att de är det optimala sättet. Behovet av ett nytt sätt att genomföra en verksamhet kan vara beroende på att förutsättningarna har förändrats, men kanske även för att man har varit fel ute i början.

En annan intressant aspekt är det sätt som verksamheter vandrar mellan medveten handling och automatisering. Hur kan detta överföras till datormiljön vandrigen mellan automatisering och medvetenhet hur kan denna process stödjas på bästa sätt? Det skulle innebära att en nybörjare och en expert för att utföra en verksamhet bör utföra samma steg av aktiviteter. Men för expertens del övergår det till en omedveten sekvens av handlingar, med möjligheten att snabbt kunna gå tillbaka till den högre nivån om det sker någon störning. Detta är något som till viss del stöds av kortkommandon med olika tangent kombinationer, som finns i många program. Och detta skulle kunna stärkas med standardisering mellan olika program för kommandon som utför återkommande vanliga funktioner. Om vi tar kortkommandot för att stänga av en Windowsdator (Windows-tangenten + ä-tangenten + ä-tangenten) om vi gör det i snabb följd stänger vi ner operativsystemet utan någon dialog, däremot om vi göra samma sekvens sakta ser vi de valbara alternativen och vi kan även göra det med musen och det innefattar samma sekvenser av valbara alternativ. Används det tillräckligt ofta av en användare borde det



övergå till att vara en operation, det behövs ingen medveten handling. Blir det då en störning, tex att vi vill sätta datorn i viloläge kan vi använda samma kortkommandon, men vara medveten om vad vi gör och välja v-tangenten istället för ä-tangenten när den dialogrutan för att stänga av eller sätta datorn i viloläge dyker upp.

I avsnitt 3.2.6 har vi identifierat potentiella "breakdowns" i ett OSF-projekt denna analys skulle även kunna användas som en utgångspunkt för att utvärdera OSF-applikationen. Detta genom att utvärdera om OSF-applikationen minskar eller eliminerar motsättningar från ett användarperspektiv.

Något som ligger utanför uppgiften men som kan vara värt att ta i beaktning vid införandet av ett nytt samarbetsprogram är att kombinera införandet med utbildning och någon slags nystart. Detta för att få alla berörda att använda systemet och dessutom ge möjlighet att införa nya rutiner så att systemet potential används så långt det är möjligt. Det är också viktigt att synliggöra fördelarna på den personliga nivån för användarna för att motivera att de skall använda programmet.

## 7.1 Begränsningar

Möjligheten att göra en användarundersökning har varit begränsad då bara en av de personer som har jobbat med OSF-projekt finns kvar i bolaget. Men eftersom vi har arbetat i samma lokal så har olika frågeställningar kunnat diskuteras löpande under utvecklingsprojektet.

Målet med examensarbetet var att implementera av vissa delar av programmet. Detta har delvis begränsats eftersom den programmering som sammanbinder de olika formulärens och är grunden för hur olika vyer uppvisas skall byggas på ett tidigare projekt. Denna begränsning har vi försökt att kringgå genom att vidareutveckla prototyper i Macromedia Flash<sup>®</sup>, och därigenom har vi kunnat visa en helhet av programmets design.



## Kapitel 8

# Sammanfattning

Vi har med utvecklingsverktyget Lotus Domino Designer som grund utvecklat ett designförslag för ett databasbaserat program som skall stödja samarbetet runt mindre projekt. Applikationen ger medlemmarna möjlighet att arbeta mot gemensamma dokument, spara dokumenten centralt, och på ett översiktligt sätt få tillgång till sina dokument och övriga resurser som stödjer projektet. Vidare uppnås med en integration med SWING Integrator<sup>®</sup> möjligheten att skapa mallar och läsa in återanvändbar text till dokument i Microsoft Word och Excel<sup>®</sup>. Text som fått värden från fält i Lotus, kan även sparas tillbaka till Lotus dokumenten. Vårt utvecklingsarbete har genomförts efter "Interaction Design"-modellen. En kravspecifikation ställdes upp baserat på en förstudie där vi ur ett teoretiskt perspektiv studerade datorstöd för kooperativt arbete. Vårt designförslag uppfyller till stora delar de krav som ställts upp, vissa delar kräver dock fortsatt utveckling.

En studie har genomförts på grupparbetsteorier för att få teoretiska kunskaper om området, studien behandlar i stora delar verksamhetsteorin, vi är övertygad om att verksamhetsteorin har mycket att tillföra utvärderingen av datorinteraktion under programutveckling och då speciellt program som stödjer kooperativt arbete och de speciella krav som ställs här. Detta för att teorin tar hänsyn till omgivande faktorer. Vi har tagit upp de grundläggande tankarna och modellen för verksamhet utvecklad av Engeström, vilken även tillämpats i analysen av verksamheten som varit i fokus för detta projekt.



## Kapitel 9

# Framtida arbete

Nästa steg i utvecklingsarbetet är att integrera vårt utvecklingsarbete med den programupbyggnad som har används i Luna3 för att få en körbar applikation. När detta är gjort bör ytterligare tester genomföras. Senare skall ytterligare två moduler tillföras applikationen, en med *systemkonfiguration* och den andra med *resurser*. Systemkonfigurationen ska innehålla *egenskaper*, *anpassningar* och *integration*. Innehållet i resurser ska omfatta bokningsbara objekt, tjänster/produkter och inventarier. Vidare skall aktivitet och person kunna ha tidkort för möjlighet att kunna rapportera in tid.

Till ärende har vi funderat på att kunna ha ett gantschema för att få en grafisk översikt över aktiviteternas utsträckning i tiden, något som dock kan tänkas innebära en stor arbetsinsats att implementera.

Ett antal mallar bör vidare kopplas till aktiviteterna men det är även något som kan utvecklas löpande av Lotsas projektrepresentanter.

Något som har framkommit under arbetets gång men inte hanterats i denna beskrivna lösning är möteshantering, vilken är en viktig funktion eftersom projekt ofta innefattar ett stort antal möten med olika inblandande parter kallade till möten, förberedelser och beslutade aktiviteter. Vi föreslår att hantera möten som specialfall av aktivitet. Dessa specialfall av aktivitet skulle kunna innefatta given mall för hantering av möten och där möten skulle ha sin egen sorteringsordning.

När utvecklingsarbetet har kommit på en nivå med ett fungerande system bör en grundlig utvärdering med användare genomföras för att undersöka om systemet uppfyller användbarhetsmålen specificerade i avsnitt 4.1 och hur systemet fungerar i den arbetsprocess som är beskriven i avsnitt 3.2.6. Denna utvärdering kan förutom med utgångspunkt i kravspecifikationen göras utifrån den lista av potentiella motsättningar som är beskrivna i tabell 3.1.



# Kapitel 10

## Tack

Tackar Mats Johansson och Hans Lennestål på Lotsa, för möjligheten till ett intressant exjobb, samt för stöd och hjälp i utvecklingsarbetet. Tackar även Helena Lindgren min handledare på datavetenskap.





# Referenser

- [1] J. Bardram. Designing for the dynamics of cooperative work activities. *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, ACM Press New York, NY, USA, pages 89–98, 1998.
- [2] O. W. Bertelsen and S. Bødker. Activity theory. In J M Carrol, editor, *HCI Models, Theories, and Frameworks towards a Multidisciplinary Science*. M. Kaufmann, 340 Pine Street, Sixth Floor San Francisco, 2003.
- [3] T. Brinck. Groupware: Design issues. <http://www.usabilityfirst.com/groupware/designissues.txt>, last visited 2005-12-12, 1998.
- [4] G. Dista. Activity theory as a theoretical foundation for information system research. In George Dista, editor, *Information Managment: Support Systems & Multimedia Technology*, pages 192–228. Idea Group Inc., Hershey, PA, USA, 2003.
- [5] P. Dourish, W.K. Edwards, A. Lamarcka, and M. Salisbury. Presto: an experimental architecture for fluid interactive document spaces. *ACM Press*, 6(2):133 – 161, 1999.
- [6] Y. Engeström. *Learning by Expanding: An activity-theoretical approach to developmental research*. Orienta-Konsutit Oy, Helsinki, 1987.
- [7] Y. Engeström, K. Brown, L.C. Christopher, and J. Gregory. Coordination, cooperation and communication in the courts. In M. Cole, Y. Engeström, and O. Vasques, editors, *Mind Culture and Activity*, pages 369–387. Cambridge University Press, 1997.
- [8] V. Kaptelinin. Activity theory: Implication for human - computer interaction. In B. A. Nardi, editor, *Context and consciousness. Activity theory and human computer interaction*. Cambridge University Press, 1996.
- [9] V. Kaptelinin. Umea: Translating interaction histories into project contexts. *CHI 2003:New Horizons*, 5(1):353–360, 2003.
- [10] V. Kaptelinin and B. A. Nardi. The activity checklist: A tool for representing the “space” of context. *ACM Press*, 6(4):27–39, 1999.
- [11] K. Kuutti. Activity theory as a potential framework for human -computer interaction. In Bonnie A. Nardi, editor, *Context and consciousness. Activity theory and human computer interaction*, pages 17–41. Cambridge University Press, 1996.
- [12] A.N. Leontiev. *Activity, consciousness, and personality*. Prentice-Hall, Englewood Cliffs, 1978.

- 
- [13] B. A. Nardi. Studying context: A comparison of activity theory, situated action models, and distributed cognition. In B. A. Nardi, editor, *Context and consciousness. Activity theory and human computer interaction*, pages 69–102. Cambridge University Press, 1996.
  - [14] B. A. Nardi. Concepts of cognition and consciousness: Four voices. *Journal of Computer Documentation*, 22(1):31–48, 1998.
  - [15] J. Preece, Y. Rogers, and H. Sharp. *Interacton Design beyound human-computer interaction*. John Wiley and Sons,Inc, USA, 2002.
  - [16] S. Ron and D. Spencer. *Understanding IBM Workplace Strategy and Products: Featuring Lotus Workspace*. Maximum Press, Gulf Breeze, FL USA, 2005.
  - [17] L.S. Vygotsky and M. Cole. *Mind in society*. Harvard University Press, Cambridge, 1978.
  - [18] S. Whittaker, Q. Jones, B. A. Nardi, M. Creech, L. Terveen, E. Isaacs, and J. Hainsworth. Contactmap: Organizing communication in a social desktop. *ACM Press*, 11(4):445 – 471, 2004.

# Bilaga A

## Scenario

### A.1 Scenario 1

Stina och Leif jobbar tillsammans med ett projekt mot en mindre kommun i inlandet. De har nyligen påbörjat sitt arbete. Förra träffen hade de en kort pratstund med de sex från kommunen som ingår i arbetsgruppen som skall vara med i projektet, då förde Stina in uppgifterna om användarna som behövdes på medlemmarna. Hon fick också ett antal namn och e-postadresser på personer som är intressanta för en kommande undersökning. I denna inledande fas är det bara Stina och Leif som skall arbeta med systemet. Stina har tittat i den databas som de använde vid ett tidigare tillfälle hos en annan uppdragsgivare därifrån har hon kompletterat några dokument och webbadresser till de styrdokument som redan finns i databasen. Stina har haft i uppgift att utföra en mindre undersökning. Hon skapade en ny aktivitet under kartläggning importerade en mall med frågeformulär skrev in sina frågor, det blev tio frågor med sifferuppskattning som svar och fem frågor med fritext som svar. Sedan gjorde hon ett inlägg där hon bad Lars titta på frågorna och komplettera med frågor. Lars har nu gjort ett inlägg där han har markerat att Stina skall arbeta med aktiviteten. Stina öppnar upp aktiviteten och ser att Lars har lagt till några frågor. Stina skickar nu en länk och inloggningsuppgifter till frågeformuläret som är tillgängligt på webben till de fristående användarna och ber de svara på frågorna.

### A.2 Scenario 2

Projektet är för Stina och Leif inne i slutskedet, tillsammans med arbetsgruppen har de kommit fram till ett antal bra lösningar som enligt beslut av personalchefen kommer att införas i organisationen. De har tillsammans med personalchefen bestämt en lämplig tidsplan och lägger ut de olika uppgifterna som ärenden på medarbetarna. Här läggs in vem som skall vara ansvarig och vilka medlemmar som skall ingå i arbetet vidare kopplas de handlingsplaner och andra styrdokument som är aktuella till respektive aktivitet.