UPTEC STS07 027
Examensarbete 20 p
Juni 2007

Telia CallGuide Admin

En användbarhetsutvärdering

Erika Wimelius

UPPSALA
UNIVERSITET

Teknisk- naturvetenskaplig fakultet
UTH-enheten

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
http://www.teknat.uu.se/student

Abstract

Telia CallGuide Admin - en användbarhetsutvärdering

Telia CallGuide Admin - a usability evaluation

Erika Wimelius

This thesis concerns the subject of human-computer interaction (HCI) and the adoption of available HCI methods and theories in order to improve the usability of an existing, commercial interactive software. The software in question is a Contact Center platform called Telia CallGuide and the focus for this investigation has been its administrative client, Telia CallGuide Admin. The purpose has been to study the use of Admin by real end users in order to identify usability flaws and suggest improvements to the user interface. The evaluation methods has been Contextual interviews and heuristic evaluation. Another task was to map the existing system and menu structure and propose a different structure that would improve the usability.

The results from the different evaluation methods ends in an overall recommendation to reduce the uncertainty experienced by the users and to improve the efficiency of the interaction between user and software. Suggested means to accomplish this are for instance a more consequent way of implementing basic functions like add and remove and to make the navigation easier by saving user adjustments between sessions. The suggestion for a new menu structure is based on theories that recommend width over depth and the ambition has been to reduce the number of steps by making the choices more straight forward. A more theoretic conclusion is that existing HCI theory and practical guidance have little to offer in the case where usability activities are needed to improve an existing, commercial software in the opposite of developing a new and tailored product. Usability theories need in fact to become more usable in order to gain influence in the world of software companies.

Handledare: Anna Grahn-Östlund, Madelaine Thor
Ämnesgranskare: Anders Jansson
Examinator: Elísabet Andrésdóttir
ISSN: 1650-8319, UPTEC STS07 027
Tryckt av: Ångströmlaboratoriet, Uppsala

Sammanfattning

Detta examensarbete har utförts på uppdrag av Teliasonera i Uppsala och syftar till att förbättra användbarheten hos en av företagets produkter, Telia CallGuide. CallGuide är ett så kallat Contact Center, ett datorprogram som används av kundtjänster för att hantera in- och utgående kontakter i form av exempelvis telefonsamtal, fax och brev. Sammantaget består CallGuide av fem olika delprogram eller klienter men examensarbetet handlar uteslutande om CallGuide Admin, den klient som används för att administrera och konfigurera inställningarna och användarna i Contact Centret. Efter hand som CallGuide vuxit och utökats med nya funktioner har gränssnittet i Admin också blivit allt mer omfattande. På grund av detta behövdes en utvärdering för att ta reda på om klienten kan förbättras så att den hjälper användarna att utföra sina arbetsuppgifter på ett mer ändamålsenligt, effektivt och tillfredsställande sätt., d v s. om den kan bli mer användbar.

Ett av målen med examensarbetet har varit att studera hur CallGuide Admin används av de administratörer som nyttjar verktyget i sitt dagliga arbete. För att undersöka detta har intervjuer med verkliga systemanvändare genomförts ute på deras respektive arbetsplatser. Intervjuerna hölls vid användarnas egna datorarbetsplatser så att de både kunde visa och berätta hur de använder Admin i sitt arbete. Det samlade intrycket är att användarna på det stora hel är mycket nöjda med CallGuide som system, men att de ibland känner osäkerhet i samband med sitt eget arbete i Admin. Användarna upplever bland annat att klienten är plottrig och svår att överblicka, att vissa handlingar är omständliga genomföra, samt att det ofta förekommer inkonsekvens mellan verktygets olika delar. Nästa uppgift var att göra en kartläggning av klienten. Denna kartläggning innehåller dels en översiktskarta över menystrukturen, dels en pärm med bilder på alla de individuella fönster som Admin består av. Med hjälp av dessa bilder har även en systematisk genomgång av gränssnittet genomförts för att komplettera användarnas kommentarer och synpunkter.

Utvärderingen har lett fram till en rekommendation som går ut på att minska den osäkerhet som användarna upplever, samt att öka effektiviteten i interaktionen mellan system och användare. För att minska osäkerheten bör man till exempel se till att grundläggande funktioner som att lägga till och ta bort fungerar på liknande sätt i hela Admin. Att öka effektiviteten handlar till stor del om att minska antalet musklick och detta kan bland annat åstadkommas genom att spara användarinställningar som kolumnbredder och fönsterpositioner. Rekommendationen innehåller även ett förslag till ny menystruktur som grundar sig på teorier där bredd förordas framför djup. Detta innebär i praktiken en struktur med färre nivåer och fler val på varje nivå. Ambitionen har varit att minska antalet steg för att ta sig från en plats till en annan.

Parallellt med utvärderingen gjordes även en genomgång av ett urval av litteratur inom människa-datorinteraktion för att se vilka råd och riktlinjer som finns när en undersökning av detta slag skall genomföras. Slutsatsen blev att utbudet av litteratur visserligen är stort men att merparten av informationen rör nyutveckling av system som därtill är skräddarsydda för att passa ett specifikt företag. När det gäller förbättringar av ett befintligt och kommersiellt gångbart system som CallGuide finns det idag inte mycket konkret att hämta från teorierna. Teorierna om användbarhet behöver således själva bli mer användbara.

Innehållsförteckning

1 Inledning 3

1.1 Telia CallGuide 3
1.2 Problemformulering 4
1.3 Teoretisk utgångspunkt 4
1.4 Syfte 5
1.5 Avgränsningar 5
1.6 Målgrupp 6
1.7 Inför läsningen 6

2 Telia CallGuide 7

2.1 Med kundtjänstens behov som utgångspunkt 7
2.2 Centrala begrepp 7
2.2.1 Organisationsområden och delområden 7
2.2.2 Användare, kompetenser och behörigheter 8
2.2.3 Styrning, köer, väntelistor och eskalering 9
2.3 Telia CallGuide Admin 10
2.3.1 Grundproblematik och utformning 10

3 Teori 11

3.1 Människa-datorinteraktion 11
3.1.1 Sätt användarna i första rummet 11
3.2 Användbarhet 12
3.2.1 Ett relativt begrepp 13
3.3 Användarcentrerad systemdesign 13
3.4 Exempel – Contextual Design 14
3.4.1 Contextual Inquiry och Work Modeling 15
3.4.2 Consolidation och Work redesign 15
3.4.3 User Environment Design, Paper Mock-ups och Intervjuer 16
3.5 Med strukturen i fokus 16
3.5.1 User Environment Design 16
3.5.2 Hierarkiska menystrukturer 18
3.6 Vidareutveckling av befintliga, kommersiella system 19
3.6.1 Skilj på olika utvecklingssituationer 20
3.6.2 Fokus på kritiska strukturella element 20

4 Metod 22

4.1 Förberedelsefas 22
4.1.1 Val av angreppspunkt 22
4.1.2 Val av utvärderingsmetoder 22
4.2 Datainsamling 24
4.2.1 Informantintervjuer 24
4.2.2 Contextual Inquiry 24
4.2.3 Heuristisk utvärdering 28
4.3 Kartläggning 32
4.4 Analys och rekommendationer 33

5 Resultat **34**
5.1 Informantintervjuer 34
5.2 Observationsintervjuer – Contextual Inquiries 35
5.2.1 Presentation av användarna och deras respektive organisationer 35
5.2.2 Generella kommentarer 38
5.2.3 Fönsterspecifika synpunkter 40
5.2.4 Egna iakttagelser 49
5.3 Kartläggning av CallGuide Admin 50
5.4 Heuristisk utvärdering av CallGuide Admin 52
5.4.1 Synlig systemstatus 52
5.4.2 Anpassning mellan system och verklighet 52
5.4.3 Användarkontroll och användarfrihet 53
5.4.4 Konsekvens och standarder 53
5.4.5 Förhindra fel 55
5.4.6 Igenkänning framför återminnelse 55
5.4.7 Flexibilitet och effektivitet 55
5.4.8 Estetisk och minimalistisk design 56
5.4.9 Felkorrigering 56
5.4.10 Hjälp och dokumentation 56

6 Analys och rekommendationer **57**
6.1 Gränssnittsutformning 57
6.1.1 Rekommendationer 57
6.2 Menystruktur 58
6.2.1 Planritningen – långa korridorer och många små rum 59
6.2.2 Analys 60
6.2.3 Rekommendation 60

7 Avslutande diskussion **66**
7.1 Behov av teoriutveckling 66
7.1.1 Kostnad, nytta och acceptans 67
7.1.2 En inneboende tröghet 67
7.1.3 Gör användbarhet mer användbart 68
7.2 Fortsatt utveckling av CallGuide Admin 68

8 Referenslista **70**
8.1 Internetkällor 70
8.2 Intervjuer 71

Bilaga A – Intervjuguide **72**

1 Inledning

Visst har du också undrat? När du sitter där vid skrivbordet med telefonluren fastklämd mellan axel och öra och lyssnar till den datoriserade rösten som berättar att du för närvarande har plats 27 i telefonkön och en beräknad väntetid på 13 minuter, visst har du också undrat hur det egentligen fungerar? Vem är det egentligen som svarar i andra änden och hur håller de ordning på alla samtal?

Kundservice och kundvård har kommit att bli ett par av dagens viktigaste konkurrensmedel och att inrätta en kundtjänst är ett verkningsfullt sätt för företag och organisationer att ge kunder eller medborgare bra service med hög tillgänglighet. Under det senaste decenniet har kundtjänsterna också vuxit fram som en egen bransch att ta på allvar. Eftersom verksamheten inte är beroende av närhet till sina kunder har det blivit vanligt att kundtjänsterna lokaliseras till glesbygden där både lokaler och arbetskraft är mer prisvärda. En kundtjänst är ofta en mycket diversifierad arbetsplats där man kan stöta på personer med mycket olika bakgrunder, utbildningar och förutsättningar. Själva arbetsplatsen har i många fall formen av ett landskap där ett större antal arbetsplatser inrymts i ett och samma rum. Man har ofta monitorer uppsatta så att alla medarbetare kontinuerligt hålls uppdaterade i frågan om hur många samtal och e-brev som står i kö och väntar på betjäning. Arbetet är dessutom ofta prestationsbaserat med strikta scheman och målrelaterad utvärdering.

En viktig förutsättning för att kunna bedriva ett effektivt kundtjänstarbete är att man har tillgång till ett välfungerande system som kan hantera all trafik till och från kundtjänsten. Det är dock ingen trivial uppgift att utveckla ett kommersiellt gångbart system som är användbart i den här typen av användningssammanhang där personer med så vitt skilda kunskapsbaser är hänvisade till samma datorbaserade hjälpmedel och stödsystem för att utföra sina arbetsuppgifter.

1.1 Telia CallGuide

Telia CallGuide är ett verktyg för att hantera inkommande och utgående kontakter i en kundtjänst. Verktyget hanterar såväl samtal och faxmeddelanden som e-post och vanliga brev. CallGuide är således ett Contact Center, vilket är en utvidgning av ett Call Center där man endast hanterar in- och utgående samtal.

Telia CallGuide är den största Contact Center-lösningen i Skandinavien med över 14 000 agenter fördelade på cirka 250 kunder. Varje år hanterar systemet i runda tal 15 miljoner samtal. De största kundgrupperna återfinns dels inom den offentliga sektorn i kommuner, landsting och statliga myndigheter, dels bland företag i bank-, finans- och försäkringsbranschen, inom servicesektorn, samt i resebranschen. Målgruppen är medelstora till stora kundtjänster med fler än 10 handläggare.

Telia CallGuide kan levereras till kunderna på två olika sätt. Det ena sättet innebär att kunden köper en tjänst i Telias nät, ett virtuellt kontaktcenter (VCC) och sedan betalar i relation till i vilken utsträckning tjänsten nyttjas. I det fallet ansvarar Telia för tekniken och driftövervakningen. Den andra varianten levereras som en kundplacerad lösning (Enterprise) där kunden själv äger hårdvara och licenser och även sköter drift och övervakning på egen hand. Oavsett vilken lösning kunden väljer så dirigeras all

trafik via ett talsvar där inkommande samtal styrs till rätt avdelning och rätt handläggare genom att personen som ringer gör ett antal val via sin tonvalstelefon. Man kan även få instruktionen att knappa in ett medlems- eller telefonnummer och på så sätt tillföra mer information som används av systemet för att ytterligare finjustera samtalsstyrningen.

1.2 Problemformulering

Från början utvecklades CallGuide Enterprise och CallGuide VCC parallellt och de ursprungliga gränssnitten har således tagits fram på två olika håll. Vid en senare tidpunkt valde man att smälta samman dessa båda spår till ett och samma utvecklingsprojekt, vilket inneburit att gränssnittet idag bland annat innehåller snarlika och bitvis överlappande funktioner. Vidare så har utvecklingen av gränssnittet till största del skett med utgångspunkt i produktens funktionalitet, vilken har vuxit och utökats i många olika omgångar. Detta har bland annat medfört att samma typ av grundfunktionalitet implementerats på olika sätt i olika delar av gränssnittet. Idag finns det därför ett behov av att undersöka om gränssnittet speglar hur systemet används i verkligheten. Från företagets sida önskar man således utvärdera produktens användbarhet som ett led i det kontinuerliga arbetet med att förbättra verktygets kvalitet.

Avdelningen för Customer Integrated Solutions (CIS) har tidigare samarbetat med utomstående konsulter med expertis inom människa-datorinteraktion (MDI) vid vidareutvecklingen av agentklienten i CallGuide och man ser det som en naturlig del i utvecklingsarbetet att ta del av och anpassa produkten till rådande teorier och standarder inom MDI och interaktionsdesign. Under hösten 2006 startade CIS ett råd där man behandlar olika aspekter kring produktens grafiska gränssnitt, eller produktens *Graphical User Interface* (GUI), som det kallas för på branschspråk. Detta GUI-råd har till uppgift att granska de olika gränssnitten i CallGuide och formulera övergripande produktspecifika riktlinjer för bättre enhetlighet och konsekvens mellan de olika klienternas gränssnitt. Som ett led i detta arbete skall grundliga genomgångar och utvärderingar av de befintliga gränssnitten genomföras. Uppfattningen är att en sådan inventering med fördel bör göras av någon som inte tidigare arbetat med CallGuide och därför är fri från förutfattade meningar och dold kunskap om produkten. Uppgiften anses därför vara synnerligen väl lämpad för ett examensarbete.

1.3 Teoretisk utgångspunkt

Den teoretiska utgångspunkten för examensarbetet hämtar jag från ämnesområdet människa-datorinteraktion. Inom MDI utgår man från interaktionen mellan användarna och de datoriserade system de använder. Men även om interaktionen är själva studieobjektet, så är det naturligt att utgå från användarna och sätta dem i centrum för utvecklingen eftersom det är lättare att utforma och ändra tekniska system än det är att försöka ändra på människors kognitiva förutsättningar för att hantera dessa. En annan anledning till att utgå från användarna är att människan också påverkas, både positivt och negativt, av olika interaktionsmiljöer, och med användarna i fokus blir det möjligt att ta hänsyn till exempelvis belastningsskador och stress som kan ha sitt ursprung i själva interaktion med datorsystemet. I linje med detta har det producerats mängder med böcker där författarna redogör för hur man kan anpassa systemutveckling till ett mer användarcentrerat arbetssätt med en användbar slutprodukt som resultat. Min uppfattning är dock att de flesta författare som utvecklat

metoder för att beskriva någon form av användarcentrerad systemdesign arbetar utifrån förutsättningen att det är ett nytt system som ska designas och implementeras.

I verkligheten ser det däremot ofta annorlunda ut – man har istället en färdig produkt som man vill förbättra. Av naturliga skäl borde denna typ av utvecklingssituation också öka i framtiden då mycket av tekniken idag är tillgänglig för alla. Det finns redan ett antal konkurrerande produkter på marknaden inom de flesta applikationsområden och mjukvaruföretagens arbete skiftar fokus från utveckling av nya produkter till underhåll och uppgraderingar av existerande produkter. Det finns däremot inte alls särskilt mycket litteratur som tar upp problematiken kring hur användbarhetsarbetet bör utformas då man redan har en befintlig produkt som är spridd på marknaden och där användandet ökar som är fallet med Telia CallGuide.

Användbarhetsbegreppet, vilket förklaras mer ingående i rapportens teorikapitel, bygger på att ett system ska vara ändamålsenligt, effektivt och tillfredsställande för användarna. Att systemet ska vara effektivt handlar i det här sammanhanget om *användarnas* och inte datorns effektivitet. Användarna ska med andra ord kunna utföra sina arbetsuppgifter på ett smidigt och verkningsfullt sätt. Att systemet möjliggör effektivt arbete är värdefullt både för den enskilda användaren och verksamheten som helhet. Mänsklig arbetskraft kostar bra mycket mer än maskiner och en ökning av datorernas prestanda motsvarar sällan en ökning av den mänskliga produktiviteten, även om det är här de stora kostnadsspararna finns. Betydande effektivitetsförbättringar kan dock bara åstadkommas genom att ändra den grundläggande strukturen hos ett system, inte genom den ytliga designen av systemets gränssnitt. (Tognazzini, 2003) Min uppfattning är att det finns stora effektivitetsvinster att göra genom att strukturera om gränssnittet för CallGuide Admin.

1.4 Syfte

Syftet med examensarbetet är att:

1. Undersöka hur verkliga systemanvändare nyttjar verktyget idag.
2. Identifiera användbarhetsbrister i det nuvarande gränssnittet och ge förslag till förbättringsåtgärder.
3. Göra en kartläggning av systemets nuvarande uppbyggnad och menystruktur, samt föreslå en alternativ menystruktur som kan förbättra användbarheten.
4. Undersöka hur rådande teorier inom människa-datorinteraktion kan tillämpas i samband med vidareutveckling av befintliga system.

1.5 Avgränsningar

Då Telia CallGuide är ett stort system med fem olika klienter ansågs det inte rimligt att genomföra grundliga användbarhetsundersökningar på alla fem gränssnitten inom ramen för ett och samma examensarbete. I samråd med GUI-rådet valdes därför CallGuide Admin ut som den klient undersökningen skulle fokusera på. Vidare begränsades arbetet till de delar av Admin som finns tillgängliga i Enterprise-lösningen. Detta innebär med andra ord att de delar av CallGuide Admin som *endast* finns för tjänstelösningen inte har behandlats. VCC-lösningen har heller inte beaktats i de fall där de båda varianterna löser en uppgift på olika sätt. På det stora hela ser dock gränssnitten likadana ut och därför har ingen åtskillnad dem emellan gjorts under informationsinsamlingen.

Eftersom produkten vidareutvecklas och uppgraderas kontinuerligt har utredningen i detta examensarbete utgått ifrån version 5.2.2.9 av CallGuide Admin. Brister som förekommer i denna version men redan åtgärdats i en efterföljande version har inte särbehandlats. Resultatet baseras helt och hållet på referensversionen av Admin.

Vidare så har uppgiften varit att endast titta på applikationen och gränssnittet, och inte på manualer och utbildningsverktyg även om dessa delar är betydelsefulla för systemets totala användbarhet. Det ligger också en avgränsning i det faktum att syftet inte är att införa användarcentrerade moment i utvecklingsprocessen, även om användarcentrerad systemdesign anses vara förutsättningen för användbara system.

1.6 Målgrupp

De två huvudsakliga målgrupperna för arbetet är dels personal inom TeliaSonera AB och avdelningen för Customer Integrated Solutions, dels studenter och lärare inom STS-programmet och vid institutionen för människa-datorinteraktion. Förhoppningen är dock att rapporten skall ha ett läsvärde även utanför denna grupp.

1.7 Inför läsningen

För att läsaren skall få en bättre och djupare förståelse för de problemområden som är förknippade med domänen och de utmaningar som en utredning av systemet innebär, fortsätter rapporten i kapitel 2 med en introduktion till CallGuide och CallGuide Admin. I kapitel 3 följer sedan en presentation av det teoretiska ramverk som varit utgångspunkt för de metodologiska och analytiska verktyg som använts under arbetets gång. Kapitel 4 är ett metodkapitel och innehåller mer explicita redogörelser för studiens genomförande. Där finns återgivet vilka metoder och vilket tillvägagångssätt som använts för att samla in information, för att bearbeta och analysera denna information, och slutligen för att presentera resultatet. Resultatet från datainsamlingen presenteras i kapitel 5, medan kapitel 6 innehåller analys och rekommendationer till åtgärder och förändringar. Rapporten avslutas med en diskussion i kapitel 7.

Språkmässigt har inga egna svenska översättningar gjorts, utan i de fall då det inte finns vedertagen svensk terminologi har de ursprungliga, engelska produkt-, domän- och metodnamnen använts genomgående i rapporten. Ord som inte översätts är exempelvis CallGuide, Contact Center och Contextual Inquiry. Synonymer som använts är bland annat agent/handläggare och Contextual Inquiry/intervju/ observationsintervju/utfrågning.

2 Telia CallGuide

Kapitlet syftar till att försöka ge läsaren en översiktlig inblick i utformningen av CallGuide med fokus på centrala begrepp och den grundproblematik som måste hanteras vid utvecklingen av en Contact Center-lösning av CallGuides typ. Informationen i kapitlet är hämtad från dokumentation, marknadsmaterial och utbildningsmaterial om CallGuide och CallGuide Admin. Värdefull information har även inhämtats i samtal med mina handledare och annan personal på Telia.

2.1 Med kundtjänstens behov som utgångspunkt

Det en kundtjänst framför allt är ute efter är ett redskap för att öka sin service och kundnöjdhet, men minst lika viktigt är det att få stöd för bemanningsplanering då optimering mellan kundnöjdhet och bemanning är centralt för en kundtjänst. Eftersom en kundtjänst sällan genererar konkreta intäkter till företaget gäller det att hålla kostnaderna på en rimlig nivå i relation till det mervärde servicen skapar för kunderna. Målet för kundtjänstverksamheten är helt enkelt att optimera servicen till så låg kostnad som möjligt. För att kunna göra detta behöver kundtjänsterna hjälp med information av olika slag och i detta sammanhang är statistik i alla dess former en mycket viktig informationskälla. Det gäller att kunna få ut rätt siffror för rätt gruppering. Många kundtjänster genomför egna mätningar och studier och en av Telias kunder har till exempel konstaterat att väntetid inte går att generalisera, utan att det beror på vilket ärende kunden har hur lång tid denne är beredd att vänta innan han eller hon lägger på luren. (Grahn-Östlund, intervju 2006-12-15)

Som plattform innefattar CallGuide fem olika klienter eller applikationer där CallGuide Agent är arbetsverktyget som används av handläggarna ute i kundtjänsterna. För planering och uppföljning finns två olika klienter där CallGuide Pulse ger realtidsstatistik och visar hur arbetssituationen i kundtjänsten ser ut ett visst ögonblick, medan CallGuide Stat ger statistisk information ur ett mer historiskt perspektiv. Användaren kan skräddarsy egna rapporter och få ut värdefull information för uppföljning. I CallGuide Admin konfigureras inställningarna för själva CallGuide-systemet och i CallGuide Configuration Utility sätts de mer tekniska inställningarna för varje enskild anknytning.

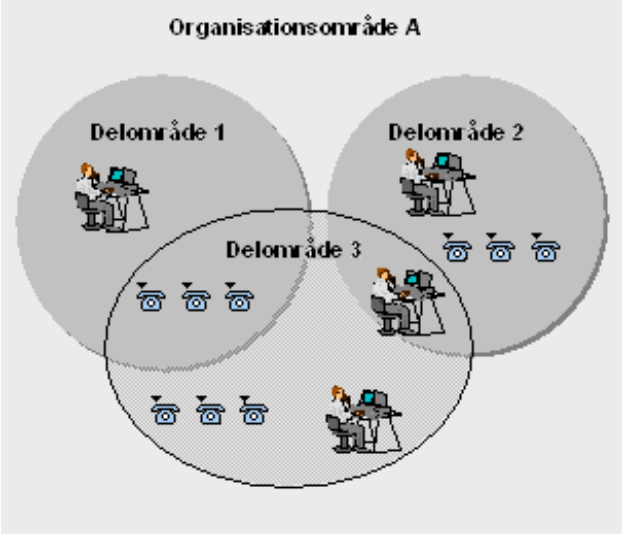
2.2 Centrala begrepp

De följande avsnitten försöker mycket översiktligt förklara grundstrukturen i CallGuide och behandlar begrepp som organisationsområden, delområden, kompetenser, agenter, behörighetsfunktioner, behörighetsroller, grupper, styrning, köer och väntelistor samt eskalering.

2.2.1 Organisationsområden och delområden

Den grundläggande strukturen i CallGuide utgörs av organisationsområden och delområden. Ett organisationsområde kan i princip ses som ett separat Contact Center med egna köer, väntelistor, grupper, kompetenser och agenter där statistik, realtidsinformation och konfigurationer är helt åtskilda från andra organisationsområden. Indelningen i organisationsområden gör det möjligt för flera fristående enheter inom samma organisation att dela på en och samma CallGuide-lösning.

Ett organisationsområde är i sin tur möjligt att dela in i mindre delområden. Dessa delområden kan överlappa varandra så att de till viss del omfattar samma köer, väntelistor, grupper, kompetenser och agenter men de kan också byggas som unika enheter. Avsikten med delområden är till exempel att göra det möjligt för en kundtjänstledare att filtrera ut den del av informationen som endast rör den egna delen av kundtjänst-organisationen. På detta sätt går det även att följa upp olika avdelningar inom organisationen som om de vore helt autonoma företag.



Figur 1: Relationen mellan organisations- och delområden i Telia CallGuide

Hur organisationen väljer att bygga upp den egna strukturen styrs av vilka behov som finns av att kunna följa upp, utvärdera och övervaka olika delar av verksamheten.

2.2.2 Användare, kompetenser och behörigheter

Varje enskild användare tillhör med andra ord alltid ett, och endast ett, organisations-område, men kan ingå i inget, ett eller flera delområden. För CallGuide som plattform har man urskiljt tre olika användargrupper: handläggare/agenter, kundtjänstledare och systemadministratörer. Agenterna är kundtjänstens ansikten utåt, de arbetar med kärnverksamheten i Contact Centret och tar emot och besvarar de inkommande kon-takterna. Kundtjänstledarna arbetar mest med övervakning, bemanningsplanering och uppföljning och har Pulse och Stat som sina huvudsakliga arbetsredskap. System-administratören arbetar i sin tur mest med Admin då arbetsuppgifterna består i att justera inställningar, administrera och konfigurera systemet.

Samtliga CallGuide användare har en behörighetsprofil och kan även ges en kompetensprofil. Behörighetsprofilen är kopplad till användarens möjlighet att hantera själva CallGuide-systemet, medan kompetensprofilen hänger ihop med vilka kontakter som skall styras fram till agenten i kundtjänsten.

Behörighetsprofilen definieras av att en behörighetsroll väljs och att ett antal behörig-hetsområden specificeras. En behörighetsroll är i sin tur uppbyggd av ett antal behörighetsfunktioner. Dessa behörighetsfunktioner anger vilka funktioner i Call-Guide som en användare med just den rollen har möjlighet att nyttja. En användare som har behörighetsfunktionen *modifiera användare* får till exempel ändra befintlig information om redan inlagda användare. För att få lägga till eller ta bort användare krävs däremot den mer kraftfulla behörigheten *äga användare*. Behörighetsrollen definierar med andra ord *vad* användaren får göra i systemet.

Behörighetsområdena utgörs av de organisations- och delområden som användaren har rätt att övervaka eller konfigurera. Behörighetsområden används för att hindra användare från olika organisationsområden eller delområden att få tillträde till

varandras konfigurationer eller övervakningsstatistik. En användares behörighetsområden anger således *var* i systemet användaren får utöva sin behörighetsroll.

En agents kompetensprofil definieras av de kompetenser och grupper som agenten har knutna till sig. En kompetens utgörs av kunskap som är relevant för arbetet i någon del av kundtjänsten. Det kan röra sig om allt ifrån kunskap om fakturor eller avtal till språkfärdigheter eller lokalkännedom. Vilka kompetenser som används definieras utifrån organisationens egen verksamhet. Till varje kompetens kan man dessutom välja att specificera en kompetensnivå och på så sätt gradera kunskapen. Grupper är att betrakta som en typ av kompetenser med den begränsningen att grupptillhörighet inte kan graderas; antingen så är man medlem i gruppen eller så är man *inte* medlem i gruppen. Ett bra exempel där grupper används istället för kompetenser är då kundtjänsten är uppdelad rent geografiskt och det skall finnas möjlighet att styra samtal till agenter på en specifik plats.

2.2.3 Styrning, köer, väntelistor och eskalering

Styrningen av inkommande kontakter i CallGuide är kompetensbaserad. Detta innebär att kundens identitet och/eller ärende styr vilken kö eller väntelista kunden hamnar i och i slutänden vilken agent som besvarar kundens ärende. När man som kund ringer in till ett CallGuide-system så kommer man först till ett talsvar. I talsvaret får man möjlighet att identifiera sig och sitt ärende genom att exempelvis mata in kund- eller telefonnummer och välja det menyalternativ som bäst passar mot det egna syftet med samtalet. Den här informationen används sedan för att koppla fram samtalet till en ledig kundtjänstmedarbetare som har rätt kompetens för att besvara just denna typ av ärenden och alltså är den som kan ge kunden bäst service.

Samtliga inkommande kontakter hamnar antingen i en kö eller i en väntelista. I en kö fördelas kontakterna automatiskt till lediga agenter med rätt kompetens, medan väntelistor låter agenterna själva välja vilken kontakt som ska betjänas förutsatt att de har rätt kompetens. Det går bra att blanda köer och väntelistor i samma lösning och organisationen väljer själv det arbetssätt som passar verksamheten bäst. I enlighet med visionen att arbeta lokalt och känna sina kunder har till exempel Länsförsäkringar Bank valt att arbeta med väntelistor så att lokala agenter kan gå in och plocka samtal från den egna regionen oberoende av hur samtalet är placerat i kö.

Ibland händer det att en kund väljer fel menyval och hamnar hos en agent som inte kan bistå kunden i dennes ärende. Kontakten måste då styras vidare till rätt kö. För att kunden inte ska behöva börja köa från början finns ofta en sidoingång till huvudkön som används i dessa fall. Här eftersträvas en blixtlåsprincip av samma typ som tillämpas vid påfarter till motorvägar i rusningstrafik. Det gäller att båda köerna betjänas rättvist utan att påfarten blir en gräddfil eller att huvudleden bara kör förbi utan att släppa fram samtalen från sidoingången.

Att bedriva effektiv kundtjänstverksamhet innebär en ständig avvägning mellan att ge kunden *snabb* service och att ge kunden *rätt* service. Ibland inträffar det till exempel händelser som gör att många människor ringer in till samma enhet inom kundtjänsten utan att det finns kapacitet att besvara alla samtal inom en rimlig tid, det blir helt enkelt trafikstockning i systemet. Vid dessa tillfällen kan kontakter automatiskt flyttas över till andra köer för att kunden åtminstone ska komma fram och få svar. Detta kallas för eskalering. Eskalering kan, precis som i exemplet, användas då kötiden

överskridit en på förhand specificerad längd, men det finns även andra situationer då detta systembeteende kan vara användbart och önskvärt.

2.3 Telia CallGuide Admin

CallGuide Admin används för att administrera och konfigurera inställningarna och användarna i Contact Centret. I CallGuide Admin ges de parametrar som behövs för att styra systemet. Via Admins grafiska gränssnitt kan man bland annat konfigurera öppettiderna i talsvaret, skapa nya köer, och skicka meddelanden till agenterna. Här görs dessutom all administrering av systemanvändarna. Agenter läggs till och tas bort och nya kompetenser, grupper och behörighetsroller skapas för att passa organisationens behov. I Admin ser man också till att alla anknytningar är inlagda på rätt organisationsområde så att samtalen över huvud taget kan styras fram. I en annan del av applikationen skapas och administreras de köer och väntelistor som behövs för att styrningen skall kunna hanteras på det sätt som organisationen önskar och själva styrningen av inkommande och utgående kontakter sköts till sist från ett tredje fönster. Vid behov kan man även lägga upp nya organisations- och delområden. Klienten innehåller utöver detta en mängd olika konfigurationsmöjligheter vad gäller exempelvis, telefonbok, kampanjer, callbacksamtal, e-post, statistik och inställningar av gränssnittet i CallGuide Agent.

Figur 2: Huvudfönstret i CallGuide Admin. Fönstret består av sex menyval och nio snabbvalsknappar, vilka leder till dialoger som nyttjas mer frekvent.

2.3.1 Grundproblematik och utformning

Givet den grundkomplexitet som ges av de krav som ställs på produkten från olika delar av kundorganisationerna har Telia fått lov att fatta ett beslut om hur man ska handskas med den variationsbredd som uppstår i användandet av produkten. Företaget har som princip valt att hålla nere kostnaderna för kunden genom att låta gränssnittet möjliggöra flertalet konfigurationer och kundanpassningar direkt i systemet. På detta sätt måste man inte bygga helt nya moduler för att kunna göra mer avancerade kundanpassningar av produkten. Detta vägval leder dock samtidigt till en ökad komplexitet. Allt fler delar av systemet som tidigare hanterats med hjälp av ren programmering lyfts efter hand in i gränssnittet för att kunderna själva ska kunna göra sina egna inställningar.

Utformningen och användandet av Admin påverkas också till viss del av klientens modala fönsterstruktur. Modalitet i det här sammanhanget innebär att det endast går att ha ett fönster öppet och aktivt åt gången. För att kunna öppna och arbeta i en annan del av klienten måste användaren med andra ord stänga ner det fönster som är öppet för stunden. Modaliteten har ansetts nödvändig i det här fallet på grund av att det arbete som utförs i Admin ofta innebär att användaren är inne i och modifierar uppgifter i systemets databas. Om det fanns möjlighet att arbeta i två olika fönster som använder samma del av databasen samtidigt skulle det lätt kunna uppstå problem då systemet får svårt att avgöra vilka uppgifter som är aktuella.

3 Teori

Kapitlet inleds med en presentation av ämnesområdet människa-datorinteraktion och de viktiga begreppen användbarhet och användarcentrerad systemdesign som utgör fundamenten till de teorier som är intressanta för examensarbetet. Läsare som redan är välorienterade inom diskursen kan med fördel gå direkt till avsnitt 3.5 för en utvidgning av problemställningen som ligger till grund för den teoretiska vinklingen av arbetet.

3.1 Människa-datorinteraktion

Människa-datorinteraktion är ett tvärvetenskapligt ämnesområde med fokus på människor, datorer och det sätt på vilket de interagerar med varandra. Mer precist så intresserar sig MDI för design, utvärdering och implementering av interaktiva datorsystem för mänskligt användande med syftet att utveckla eller förbättra säkerheten, nyttan och användbarheten i system som inkluderar datorer. (Lindgaard, 1994) Forskningen och arbetet inom MDI strävar efter att förstå de psykologiska, ergonomiska, organisatoriska och sociala faktorer som reglerar mänskligt handlande. Med bakgrund i denna förståelse försöker man sedan ta fram verktyg och tekniker för att hjälpa systemutvecklare i arbetet med att utforma datorsystem som är anpassade till de aktiviteter som systemen ska användas till. Målet är att åstadkomma effektiva, ändamålsenliga och säkra interaktioner och fler användbara slutprodukter. (Preece, Rogers, Sharp, Benyon, Holland, & Carey, 1994)

Människa-datorinteraktion myntades som begrepp i början av 1980-talet och terminologin som sådan är viktig eftersom den visar att intresset inte stannar vid ytliga egenskaper så som gränssnittsdesign, utan fokuserar på alla aspekter som relaterar till interaktionen mellan människor och datorer. Ämnet bygger på kunskaper och metoder från många olika discipliner där de båda huvudspåren utgörs av människa-maskininteraktion och ergonomi. Därtill är det i huvudsak datavetenskap, kognitionspsykologi och samhällsvetenskap som bidragit med en stor kunskapsbas, men även områden som lingvistik, datorgrafik och industriell design är relevanta i sammanhanget. (Gulliksen & Göransson, 2002; Preece et al., 1994) Idag är MDI ett brett område som, enligt Gulliksen och Göransson (2002), spänner över allt ifrån datorstött samarbete och systemutveckling till teknikens sociala villkor eller samhälleliga effekter.

3.1.1 Sätt användarna i första rummet

Dix, Finlay, Abowd och Beale, (1998) definierar termen människa-datorinteraktion lite närmare utifrån sina tre beståndsdelar. Ordet *Människa* står i det här fallet antingen för en individuell användare, en grupp av användare som arbetar tillsammans, eller sekvenser av användare i en organisation där var och en hanterar olika delar av arbetsprocessen. Begreppet *Dator* specificeras här till att omfatta samtliga teknologier från den enskilda persondatorn till ett storskaligt datorsystem, ett processkontrollsystem eller ett integrerat system. Systemet som sådant kan även inkludera icke-datoriserade delar, inklusive andra människor. *Interaktion* står i sin tur för all form av kommunikation mellan människa och dator, oavsett om den är direkt eller indirekt.

Det finns ingen generell, entydig förståelse av disciplinen, men en av hörnstenarna i all MDI-forskning är uppfattningen att sätta de människor som ska använda datorsystemet, användarna, i första rummet. Deras behov, fallenheter och preferenser bör genomsyra det sätt på vilket ett datorsystem designas och implementeras. En annan grundläggande princip är att människor använder datorer för att åstadkomma något, för att utföra en specifik uppgift. Ett användbart datorsystem stöttar användaren i arbetet med att utföra denna uppgift. Om systemet däremot tvingar användaren att anpassa sig till ett oacceptabelt arbetssätt så är systemet inte användbart. Utgångspunken är med andra ord att människor inte ska behöva anpassa sig till systemet, utan att systemet från början skall utformas för att möta deras anspråk. (Dix et al., 1998; Preece et al., 1994)

3.2 Användbarhet

I folkmun har *användarvänlighet* kommit att bli ett vanligt begrepp som används för att beskriva önskvärda egenskaper hos ett datorsystem. Dessa egenskaper kan handla om att systemet ska vara intuitivt, lätt att lära sig och att vem som helst kan använda det. I utvecklingssammanhang fyller användarvänlighet som begrepp däremot inte någon funktion eftersom det är oklart vad användarvänlighet egentligen innebär. Detta medför att användarvänlighet heller inte går att kravställa utifrån: Vem avgör vad som är användarvänligt eller inte och på vilka premisser? (Gulliksen & Göransson, 2002)

Inom MDI har man därför övergått till att använda sig av begreppet *användbarhet* (eng. usability) när man pratar om den här typen av systemegenskaper. Det råder dock en viss oenighet även kring användbarhetsbegreppet och de åsikter och formuleringar som framförts av olika författare och teoretiker diskuteras både ingående och väl av Gulliksen och Göransson (2002). En skiljelinje bland teoretikerna är huruvida ett systems funktionalitet ska anses vara en del av användbarheten eller inte. Här går Gulliksen och Göransson på samma linje som ISO, den internationella organisationen för utformning av standarder, vilken menar att användbarhet innefattar såväl funktionella som icke-funktionella egenskaper. Enligt ISO definieras användbarhet som:

”den utsträckning till vilken en specificerad användare kan använda en produkt för att uppnå specifika mål, med ändamålsenlighet, effektivitet och tillfredsställelse, i ett givet användningssammanhang.”

(ISO 9241-11, 1998, enligt Gulliksen & Göransson, 2002)

Standarden omfattas även av definitioner av de ingående begreppen:

- **Ändamålsenlighet** – *noggrannhet och fullständighet med vilken användarna uppnår givna mål.*
- **Effektivitet** – *resursåtgång i förhållande till den noggrannhet och fullständighet med vilken användarna uppnår givna mål.*
- **Tillfredsställelse** – *frånvaro av obehag, samt positiva attityder vid användningen av en produkt.*
- **Användningssammanhang** – *användare, uppgifter, utrustning samt fysisk och social omgivning i vilken produkten används.*

Enligt Gulliksen och Göransson (2002) har ISO-standarden ett antal positiva egenskaper jämfört med andra förklaringsmodeller. Till att börja med är ISO-definitionen en internationell standard som enar begreppsapparaten och utgör en gemensam

utgångspunkt för diskussioner kring användbarhetsfrågor. Standarden ger även en konkret definition av begreppet som gör det möjligt att mäta användbarhet kvantitativt, att formulera explicita användbarhetskrav, samt att ange användbarheten i relation till någon annan produkt. Även Lindgaard (1994) menar att fördelen med användbarhet som begrepp är att det uttrycks i kvantifierbara, mätbara termer med hjälp av vilka man kan fastställa när ett ”bra” system är ”tillräckligt bra”. En tredje styrka är i Gulliksen och Göranssons mening att definitionen försöker ”ta ett helhetsgrepp om problematiken då den innefattar hela systemets bredd ur användarens perspektiv från funktionalitet till upplevelsen av de estetiska värdena.”

3.2.1 Ett relativt begrepp

Det är viktigt att inse att användbarhet inte är en absolut storhet, utan ett relativt begrepp. Användbarhet är med andra ord ingen objektivt observerbar, inneboende produktegenskap, utan en kvalitetsegenskap hos interaktiva produkter som visar sig i samspelet mellan en produkt och dess användare över en tidsperiod. (Gulliksen & Göransson, 2002; Ottersten & Berndtsson, 2002)

Användbarhet är dessvärre en egenskap som ofta tas för given eller som förväntas uppstå ur tomma intet utan att man egentligen behövt ägna särskilt mycket tanke, energi eller medel åt att frambringa det (Gulliksen & Göransson, 2002). För att skapa en användbar produkt behöver man enligt Ottersten och Berndtsson (2002) utforma produkten med hänsyn både till det mänskliga systemet, dvs. de egenskaper som bärs av individerna som använder produkten, och till de fysiska, psykiska, sociala och organisatoriska sammanhang produkten skall användas i. Som en tredje faktor vill författarna betona behovet av att även ha produktens förväntade nytta i åtanke – både för dem som använder, och för dem som tillhandahåller produkten. Till skillnad från ISO-definitionen där huvudfokus ligger på att målgruppernas behov ska tillgodoses menar Ottersten och Berndtsson att produkten även ska uppfylla beställarens syften för att anses som användbar. Som exempel på ett par olika nyttor som ett bra användbarhetsarbete kan leda till nämner författarna ökad produktivitet, minskad inlärningstid, ökad tillfredsställelse för användaren, engagemang hos beställare och användare, minskade kostnader och kortare tid för utveckling, minskade livscykelkostnader, samt ett stärkt varumärke.

3.3 Användarcentrerad systemdesign

Användarcentrerad systemdesign (ACSD) är ett samlingsnamn för de systemutvecklingsprocesser som tagits fram med avsikt att producera användbara datorsystem. En användarcentrerad utvecklingsprocess riktar uppmärksamheten mot slutanvändarna, deras arbete och arbetsmiljö genom hela designprocessen och fokuserar på hur teknologin bäst kan utvecklas och designas för att hjälpa användarna att nå sina mål. Syftet med ett datorsystem är att tillhandahålla service åt användaren och det finns inget egenvärde i att använda en specifik teknologi eller att producera snygg och prydlig kod. I en användarcentrerad utvecklingsprocess är det viktigt att såväl funktionella som icke-funktionella systemkrav specificeras med användarna i centrum och att hela systemdesignen utgår ifrån deras behov, mål och förväntningar. (Gulliksen & Göransson, 2002; Preece et al., 1994)

ISO har tagit fram en standard även för användarcentrerad systemdesign men den representerar snarare ett övergripande koncept än en entydig utvecklingsmodell och är

med andra ord inte lika precist formulerad som standarden för användbarhet. Enligt denna ISO-standard innebär användarcentrerad systemdesign:

1. Aktiv involvering av användare och en tydlig förståelse av användarens och uppgiftens krav
2. En lämplig allokering av funktion mellan användare och teknik
3. Iterering av designlösningarna;
4. Tvärdisciplinär design.

I samband med detta definierar ISO även fyra användarcentrerade designaktiviteter som skall utföras inom ett systemutvecklingsprojekt, nämligen att förstå och specificera användningssammanhanget; att specificera användarnas och organisationens krav; att producera designlösningar; samt att utvärdera designen gentemot krav. (ISO 13407, 1999, enligt Gulliksen & Göransson, 2002) Ottersten och Berndtsson (2002) har utgått ifrån den nyss beskrivna ISO-definitionen och funnit att många befintliga utvecklingsmetoder uppvisar brister i förhållande till standarden. Även om olika metoder har olika brister så finns det gemensamma drag. Exempel på detta är att användarnas begreppsvärld och olika användningssituationer inte beskrivs i tillräcklig omfattning, att interaktionsdesign inte ses som en medvetet styrd process och att ingen utvärdering genomförs för att säkerställa produktens användbarhet.

Så som Preece et al. (1994) beskriver det är också utvecklingen av användarcentrerad systemdesign i mångt och mycket en reaktion på de tillkortakommanden som tillskrivs flertalet av de traditionella utvecklingsprocesserna som exempelvis vattenfallsmodellen. Merparten av dessa systemutvecklingsmodeller är i huvudsak orienterade mot utvecklingen av stora programvaror med fokus på systemets funktionalitet. De bygger dessutom på principen att varje processteg tar indata från steget ovanför, bearbetar den med hjälp av någon uttalad metod och sedan lämnar vidare ett välpaketerat resultat till steget nedanför. Enligt författarna är den mest grundläggande svagheten att det i verkligheten är fullständigt omöjligt att helt och hållet förstå och formulera krav innan projektet kommit en bit på väg och det finns utkast till designen. Användarcentrerad systemdesign erkänner däremot betydelsen av täta användartester och en iterativ utvecklingsprocess som ger återkoppling och möjlighet att korrigera felaktiga antaganden i ett tidigt skede.

De huvudsakliga principerna i användarcentrerad systemdesign kan sammanfattas i att bibehålla fokus på användaren och användbarhet genom hela designprocessen, att kontinuerligt genomföra tester och utvärderingar med verkliga användare, samt att tillämpa ett iterativt arbetssätt. (Preece et al., 1994)

3.4 Exempel – Contextual Design

En av de mest kända och använda metoderna inom MDI och användarcentrerad systemdesign är *Contextual Design* (Beyer & Holtzblatt, 1998). Följande avsnitt innehåller en kortfattad genomgång av de olika metodstegen i denna system-utvecklingsmodell och informationen är i sin helhet hämtad från upphovsmakarnas bok samt deras hemsida www.incent.com (2007-05-03).

Contextual Design såg dagens ljus för första gången i mitten av 1990-talet. Som namnet antyder vill författarna betona vikten av att ta hänsyn till användnings-

sammanhanget genom att föra utvecklingsarbetet närmare slutanvändarna och den miljö eller kontext som arbetet utförs i och där även datorstödet måste passa in. Enligt författarna själva handlar Contextual Design till stor del om att belysa och konkretisera viktiga designmoment som redan utförs i det tysta av duktiga designers. Genom att göra dessa moment explicita och tillgängliga för andra kan de istället utföras gemensamt av ett helt designteam. När fler personer kan delta i tanke- och beslutsprocessen är förhoppningen att fler aspekter kan täckas in redan från början.

Contextual Design fokuserar på designfasen i ett utvecklingsprojekt och urskiljer ett antal olika designsteg som ska genomföras *innan* man över huvud taget börjar med att koda och programmera. Som modell är Contextual Design i sin helhet optimerad för storskaliga, komplexa projekt men har även med framgång applicerats på mindre projekt. Nedan följer en kortfattad beskrivning av de delar som ingår i Contextual Design.

3.4.1 Contextual Inquiry och Work Modeling

Det första designsteget handlar om att skapa en förståelse för användarna, deras behöv, önskemål och inställning till sitt arbete genom att utföra enskilda intervjuer med användare på deras arbetsplats medan de utför sina arbetsuppgifter. Contextual Inquiry (CI) avslöjar dolda strukturer i arbetet och förser designteamets deltagare med en mycket innehållsrik datamängd att basera sina designidéer och lösningar på. De enskilda intervjuerna följs sedan upp av tolkningssessioner där hela teamet deltar.

Under tolkningssessionerna samlas den inhämtade informationen i olika arbetsmodeller, vilka tillhandahåller en konkret representation av det arbete som varje individuell användare utför. Det finns fem olika typer av arbetsmodeller, vilka bidrar med olika perspektiv på arbetssituationen: *Flödesmodellen* beskriver de olika roller som finns på arbetsplatsen samt de formella och informella vägar som rollerna använder för att kommunicera med varandra. *Sekvensmodellen* skildrar arbetsgången för olika arbetsuppgifter utifrån ett visst mål samt den sekvens av handlingar som utförs för att nå fram till målet. *Artefaktmodellen* beskriver hur användarna skapar eller tar hjälp av olika föremål i sitt arbete. *Den kulturella modellen* visar hur både organisationskultur och samhällskultur påverkar enskilda personers arbete, medan *den fysiska modellen* beskriver hur användaren förflyttar sig och utför sina uppgifter i den fysiska arbetsmiljön.

3.4.2 Consolidation och Work redesign

Eftersom ett system byggs för att passa en hel grupp av användare behöver arbetsmodellerna arbetas ihop till en enhetlig och integrerad modell. Detta görs med hjälp av så kallade affinitets- eller relationsdiagram och konsoliderade arbetsmodeller där gemensamma strukturer i arbetet fångas upp.

Nästa steg är att analysera och utvärdera de konsoliderade arbetsmodellerna utifrån det arbetssätt som organisationen skulle vilja ha i framtiden. Med utgångspunkt i denna framtidsvision fattas beslut om hur arbetet kan förbättras och hur tekniken kan användas för att bättre bistå användarna i den nya arbetsformen. Ambitionen är att explicit utforma systemet så att positiva strukturer i arbetet förstärks medan negativa motverkas. Det ideala arbetssätt som man tar fram fångas sedan med hjälp av

storyboards, vilka beskriver hur användare går tillväga för att lösa en viss uppgift i systemet som en serie av stillbilder eller seriestripar.

3.4.3 User Environment Design, Paper Mock-ups och Intervjuer

Contextual Design framhåller vikten av att systemstrukturen bestäms innan man går vidare med konkret gränssnittsdesign. För att kunna stödja arbetsflödet måste systemet i första hand ha lämpliga funktioner men de måste även struktureras på ett lämpligt sätt. User Environment Design (UED) är en representation av systemets arbetsmodell som liknar planritningen till ett hus. Här framgår vilka delar systemet är uppbyggt av, vilka funktioner som hör ihop med och kan användas i de olika delarna, samt hur förflyttning sker från en del till en annan. User Environment Design beskrivs närmare i avsnitt 3.5.1.

Att visualisera olika förslag till designlösningar och diskutera dessa med användarna är ovärderligt i ett utvecklingsprojekt. Att testa med hjälp av mock-ups och prototyper i papper är ett kostnadseffektivt sätt att iterera den tänkta designen med användare och på så sätt kunna korrigera brister innan designen har implementerats. Det är generellt accepterat att ju tidigare man finner problem, desto mindre kostar det att rätta till dem.

Sist men inte minst gäller det att omsätta designen i ett fungerande gränssnitt så att användarna med tillfredsställelse kan interagera med systemet och utföra sina arbetsuppgifter på ett ändamålsenligt och effektivt sätt.

3.5 Med strukturen i fokus

Det här avsnittet innehåller två olika delar där den första är en fördjupning av metodsteget User Environment Design från Contextual Design och det andra sammanfattar mer generell teori i ämnet hierarkiska menystrukturer.

3.5.1 User Environment Design

Beyer och Holtzblatt (1998) anser att det är en utmaning att designa system som överensstämmer med slutanvändarnas förväntningar och är ett verkligt stöd i deras dagliga arbete. Ett sådant system behöver vara en sammanhängande enhet som avspeglar och tillmötesgår användare som arbetar i vitt skilda verksamheter och använder många olika stödsystem och som dessutom har olika bakgrund, utbildning, arbetsuppgifter, behov och önskemål. Att användaren upplever systemet som en sammanhängande enhet har delvis med konsekvens och stöd för olika arbetsflöden att göra, men det räcker inte. Ett koherent system behöver dessutom tillhandahålla en välordnad och logisk miljö för användaren att utföra sitt arbete i. För att åstadkomma detta måste alla olika delar av systemet byggas på en gemensam organisation och struktur som stödjer användarens naturliga förflyttningar i arbetet och samtidigt möjliggör nya arbetssätt som inte förutsetts av systemdesignern. Systemstrukturen byggs upp av tre element: en *plats* är en del av systemet där relaterade arbetsuppgifter utförs; *funktioner* är de verktyg som finns till hands för att utföra uppgifterna och *länkar* tillåter användaren att förflytta sig mellan olika platser.

Att anpassa systemet efter användarnas förväntningar och arbetssätt är med andra ord en fråga om att strukturera systemet på rätt sätt. Diskussioner om hur arbetet i systemet ska struktureras behöver ett eget forum så att de tydligt kan särskiljas från

diskussioner om gränssnittet eller den interna systemstrukturen och de rent tekniska lösningarna som ligger bakom det användarna upplever. Utan något hjälpmedel för att fokusera diskussionen är det däremot ofta lätt hänt att man börjar prata om specifik dialogdesign och gränssnittsutformning innan den övergripande strukturen är klar. Avsaknaden av en tydlig struktur kan exempelvis leda till att olika delar av systemet utvecklats åt olika håll och att systemet i förlängningen inte representerar en sammanhängande lösning. Att se systemstrukturen som en helhet är därför särskilt viktigt i de fall då förändringsarbete genomförs på en mindre del av systemet. Trots goda intentioner att förbättra just den delen måste man efter varje förändring ta ett steg tillbaka och se hur förändringen påverkar resten av systemet. Enligt Beyer och Holtzblatt är det lätt hänt att system blir mer ostrukturerade med tiden och att det som från början var ett rimligt och elegant gränssnitt efter hand förvandlas till ett gytter av funktioner och länkar utan någon urskiljbar struktur. Problem av det här slaget blir särskilt påtagligt i stora utvecklingsprojekt där många olika personer är involverade och arbetar på olika delar av systemet. Beyer och Holtzblatt menar därför att det behövs en extern representation av systemet som kan fungera som utgångspunkt i diskussioner och överenskommelser för att utvecklingsarbetet ska leda fram till en enhetlig och koherent produkt.

3.5.1.1 Med planritningen som utgångspunkt

Enligt Beyer och Holtzblatt finns det stora likheter mellan att arbeta i ett datorsystem och att bo i ett hus. Ett datorsystem består, precis som ett hus, av platser eller rum där man kan utföra olika handlingar, funktioner eller verktyg som kan användas för att utföra dessa handlingar, samt dörröppningar och korridorer som länkar ihop de olika delarna och låter husägarna förflytta sig mellan olika rum. Gränssnittsdesign liknar de på samma sätt vid inredningen i ett hus – den spelar stor roll för upplevelsen och tillfredsställelsen men om den grundläggande strukturen är fel finns det inget gränssnitt som kan ställa saken till rätta. Vid utformningen av ett hus använder sig arkitekter allt som oftast av en *planritning* för att ge en överskådlig bild av den tänkta planlösningen. Som diagram ligger planritningen till grund för diskussioner om hur väl designen stöder en viss livsstil och är ett effektivt hjälpmedel för arkitekten som kan jämföra detta med behoven och önskemålen från husets framtida ägare. Med utgångspunkt i ovanstående jämförelse menar författarna att en planritning med fördel skulle kunna användas som representation även för den övergripande strukturen hos ett datorsystem.

Den representation som Beyer och Holtzblatt baserar på ovanstående husanalogi kallar de för User Environment Design (UED). En svensk översättning skulle bli ungefär ”design av användningsförhållanden”. Byggstenarna i en UED kallas för Focus areas (fokusområden) och de representerar enhetliga platser i systemet, vilka stödjer utförandet av en arbetsuppgift. De utgör systemets ”rum”. Precis som verkliga rum skall dessa fokusområden vara välutrustade för att på bästa sätt hjälpa användaren att utföra sitt arbete genom att tillhandahålla de funktioner som är nödvändiga för att göra detta – och endast de funktioner som är nödvändiga. Varje fokusområde ska ha ett syfte, en kortfattad beskrivning av det arbete som fokusområdet stöder. Om syftet inte går att sammanfatta i en enda mening på grund av att fokusområdet innehåller så många olika funktioner är systemet troligtvis dåligt strukturerat. Precis som en planritning innehåller UEDn även information om de länkar som fokusområdet har till andra delar av systemet. En UED visar helt enkelt systemet som användaren upplever det med alla delar och deras inbördes förhållanden, oberoende av gränssnitt.

Precis som man förväntar sig att människor tillbringar en viss tid i ett rum innan de går vidare till nästa bör användarna av ett system kunna arbeta i ett och samma fokusområde under en sammanhängande tidsrymd. Att som systemutvecklare lära sig att tänka i termer av fokusområden och länkar bidrar till att koncentrera relaterat arbete till ett och samma fokusområde istället för att sprida ut arbetet över olika delar i systemet. En UED hjälper med andra ord till att stävja en snabb förökning av olika fönster, rutor och dialogboxar och ser till att hålla systemstrukturen koncis och koherent. User Environment Designs kan enligt Beyer och Holtzblatt användas både vid nyutveckling och för att representera existerande system. I det senare fallet blir de ett hjälpmedel för att avslöja den underliggande strukturen och framhäva problemområden som exempelvis dubbel funktionalitet och sådant som saknas.

3.5.2 Hierarkiska menystrukturer

Menyer i form av hierarkiska trädstrukturer har förmågan att inrymma stora datamängder och en menyhierarki anses därför ofta vara nödvändig för att öka tillgängligheten till informationen i stora system (Nielsen, 1993; Shneiderman, 1998). Hierarkiskt nästlade menystrukturer utgör därför grunden för många interaktiva system. Men hierarkiska menystrukturer för även med sig vissa inneboende problem. Enligt Nielsen (1993) vore det bäst att undvika hierarkiska menyer helt och hållet eftersom de innebär att användaren inte har samtidig tillgång till alla olika alternativ och kräver speciella interaktionstekniker för navigering. Lindgaard (1994) menar att användare har en benägenhet att tappa bort sig i hierarkiska menystrukturer och ofta glömmer bort var någonstans i hierarkin de för närvarande befinner sig, vart de är på väg och varifrån de kom. I hans mening verkar de navigationsproblem som hierarkin medför snarare gör det svårare för användarna att få tillgång till den information som systemet tillhandahåller.

Enligt Dix et al. (1998) är det huvudsakliga problemet med menyer att avgöra vilka element som ska inkluderas och hur de i sin tur skall grupperas. Den grundläggande kompromissen vid gruppering av element i en hierarkisk menydesign ligger i valet mellan bredd och djup. En platt, bred struktur kräver inte att användaren tar sig igenom lika många nivåer som en djup meny och reducerar därmed behovet av navigering. Samtidigt blir varje nod i hierarkin mer komplex, vilket leder till att användaren måste göra fler val vid varje nivå. Eftersom både navigering och beslutsfattande tar tid, gäller det att hitta en rimlig utformning där hierarkin varken blir för bred eller för djup. (Nielsen, 1993) Många författare framhåller att man inte bör använda fler än 4-8 alternativ per meny, men samtidigt inte heller nyttja fler än 3 till 4 nivåer. I stora menyapplikationer måste avvägningar självklart göras även mellan dessa båda tumregler (Shneiderman, 1998).

3.5.2.1 Föredra bredd framför djup

Det har gjorts många empiriska studier med fokus just på avvägningen mellan bredd och djup i hierarkiska menystrukturer och det finns starka bevis för att bredd skall föredras framför djup (Shneiderman, 1998). Att öka menybredden före menydjupet har bland annat visat sig förbättra användarnas prestation och minska antalet fel (Lindgaard, 1994). Shneiderman (1998) hänvisar till en studie utförd av Kieger 1984 där 64 element på olika sätt delats in i menyer. Resultatet visade att en trädstruktur med 8^2 element, dvs. åtta alternativ på vardera två nivåer, fick höga betyg både vad

gäller snabbhet, träffsäkerhet och tillfredsställelse hos användarna. Ett smalt, djupt träd med två valalternativ på vardera sex nivåer, med andra ord 2^6 element, producerade som jämförelse ett långsamt, oprecist och icke-tilltalande alternativ till att organisera de 64 menyelementen. Över huvud taget finns det många empiriska belägg för att plattare menystrukturer med fler menyval åt gången men färre nivåer är att föredra framför djupa strukturer med få val och många nivåer.

Lindgaard (1994) tar upp problemet med att användare kan ha svårt att hitta i en menyhierarki på grund av att de korta fraser som beskriver de tillgängliga valen är formulerade på ett mångtydigt sätt. Även här måste designern sammanfoga två motsatta krav då varje menyval måste beskrivas tillräckligt generellt för att innefatta innehållet i hela den underliggande hierarkin och samtidigt formuleras tillräckligt specifikt för att på ett tydligt sätt vägleda användaren till information som lagrats på lägre nivåer. Enligt Nielsen (1993) måste man räkna med att nya användare som ännu inte är vana vid systemet kommer att göra fel, vilket föranleder ett behov av att låta de olika undermenyerna innehålla naturliga grupperingar av alternativ så att namnen på den högre nivån blir så lättbegripliga som möjligt. Shneiderman (1998) fortsätter på samma spår och menar att förflyttningen genom en meny kan utföras på ett par sekunder om grupperingen av alternativ på varje nivå är naturlig och begriplig för användarna. Å andra sidan är det inte alltid helt enkelt att gruppera menyelement på ett sätt som upplevs som naturligt för alla användare. Shneiderman liknar problemet vid hur man ordnar sina köksredskap; förskärarna ligger tillsammans på ett ställe och gafflar och skedar på ett annat, men var hör bordsknivarna och smörknivarna hemma? På samma sätt kommer menykategorier att överlappa varandra och tillsynes orelaterade element att hamna under samma meny. Det finns med andra ord ingen perfekt trädstruktur som matchar alla användares kunskap om applikationsdomänen, utan resultatet baseras på avvägningar mellan olika designbeslut. Den initiala designen bör dock förbättras genom feedback från användare.

3.6 Vidareutveckling av befintliga, kommersiella system

Efter att ha läst en hel del litteratur som behandlar användarcentrerad systemdesign har jag stött på mycket information om hur man bör gå till väga för att utvecklingsprocessen skall leda fram till ett användbart system. De flesta processer är dock utformade på ett sådant sätt att de lämpar sig bäst vid utveckling av helt nya system, antingen vid en intern IT-avdelning eller av en extern leverantör. I det här fallet handlar det däremot om att genomföra passande användbarhetsinsatser för att förbättra ett redan befintligt system. Även om det inte finns mycket information om detta i dagens litteratur så går det att hitta början till ett resonemang hos ett par författare.

En vanligt förekommande uppmaning i MDI-litteraturen är att genomföra användbarhetsinsatser så tidigt som möjligt då eventuella problem anses bli allt dyrare att rätta till efter hand som projektet fortskrider. Ottersten och Berndtsson (2002) poängterar dock att användbarhetsinsatser kan, och bör, göras i ett projekts alla faser och att man kan göra åtgärder som bidrar till att förbättra produktens kvalitet även i projekt som hittills genomförts utan användbarhetsaktiviteter. De aktiviteter som i deras mening är enklast att utföra och ta till sig i ett sådant projekt brukar vara expertutvärdering och användningstest.

3.6.1 Skilj på olika utvecklingssituationer

Enligt Preece et al. (1994) måste man skilja på de situationer då en produkt skall utvecklas för ett specifikt företag och de situationer då en kommersiell produkt skall tas fram. I det senare fallet skall produkten säljas på öppna marknaden och det finns ingen speciell användarorganisation i åtanke. Detta betyder att utvecklingsteamet är begränsade till mer generell information om produktens framtida användare som utgångspunkt för designen. Enligt författarna är det dessutom viktigt att skilja på utvecklingen av helt nya system och den sorts design som sker när ett system skall uppdateras eller underhållas. För stora system har underhållsfasen till och med uppskattats till att omfatta så mycket som 60 % av de totala arbetsinsatserna. Jenny Preece och hennes medförfattare menar att utvecklingsfasen i själva verket är ett av de viktigaste stadierna i en produkts livscykel. En mjukvarudesigner spenderar med andra ord generellt sett mest tid på att modifiera och uppgradera befintliga produkter, vilket betyder att en stor del av designaktiviteten egentligen handlar om att formge något på nytt. I dessa situationer är friheten hos designern kraftigt begränsad av beslut som tagits tidigare under designprocessen, antingen i originaldesignen av produkten eller i den produktlinje som produkten tillhör och det tillgängliga designutrymmet är litet.

3.6.2 Fokus på kritiska strukturella element

Beyer och Holtzblatt (1998) fokuserar i första hand på distinktionen mellan nyutveckling och vidareutveckling av en produkt och har lite mer konkreta förslag att komma med. De menar att Contextual Design i sin helhet bör användas i de fall då man ska utveckla en helt ny produkt. Eftersom varje projekt har sina egna unika förutsättningar bör utvecklingsprocessen dock anpassas till varje nytt designprojekt. Det gäller att avgöra vilka steg som ska ingå och vilka som kan uteslutas och att vara aktsam så att inte för många delar av processen inkluderas i onödan. Det är i annat fall lätt hänt att processen tar för lång tid och driver teamet till att fundera över aspekter som ändå inte kan adresseras inom projektets ramar.

När det, som i det här fallet, handlar om att vidareutveckla en redan befintlig produkt och omarbeta den med användarnas intressen i fokus behövs enligt Beyer och Holtzblatt en process som inkluderar de kritiska strukturella elementen av Contextual Design. De föreslår att teamet bör få till stånd Contextual Inquiries med 8-15 personer, samt genomföra anslutande tolkningssessioner. Syftet med intervjuerna är i det här läget att tillföra värdefull information om användarnas problem med och krav på systemet inför uppgraderingen. Utmaningen ligger i att se de grundläggande behov som ligger bakom ett specifikt önskemål och sedan designa en lösning som motsvarar dessa behov samtidigt som den passar in i det mer övergripande arbetsflödet och även bevarar systemdesignens integritet.

Ett projekt av den här typen handlar om att förbättra systemet efter en redan utstakad väg så inga flödesmodeller eller kulturella modeller behövs. Författarna menar däremot att sekvensmodellen, artefaktmodellen och den fysiska modellen bör byggas för att man ska kunna se hur systemet interagerar med användarnas värld rent strukturellt. För att sedan få en sammansatt bild av de intervjuade användarna kan man med fördel generera användningsscenarier istället för att ta fram konsoliderade arbetsmodeller. Ett relationsdiagram bör byggas som utgångspunkt för kreativa idédiskussioner och brainstorming kring olika synpunkter och designförslag. Nästa

steg är att välja ut ett antal nyckelsynpunkter att visionera utifrån och att bygga storyboards. När det är klart har designteamet tillräckligt mycket på fötterna för att gå direkt på gränssnittsdesign och prototyping. När det kommer till User Environment Design så rekommenderar Beyer och Holtzblatt att man i det här fallet använder sig av den tankemodell som är bakgrunden till UED för att organisera gränssnittet, och att man alltså kan avstå från att bygga explicita UED-diagram. På det här sättet kommer man snabbt från data till att organisera ett designförslag. När målet inte är att skapa ett nytt sorts system, utan att skapa nästa iteration av ett system som redan existerar är detta enligt upphovsmakarna ett lämpligt sätt att använda processen på.

4 Metod

Det här kapitlet innehåller närmare beskrivningar av de metoder som använts för att genomföra utvärdering och kartläggning av CallGuide Admin, samt för att analysera och bearbeta det insamlade och framtagna materialet.

4.1 Förberedelsefas

Förberedelsefasen inleddes med att söka reda på relevant litteratur inom ämnet människa-datorinteraktion med Uppsala universitets biblioteksresurser som den huvudsakliga källan. Det har även funnits mycket bra information att tillgå på olika Internetsidor. Nästa steg var att skapa sig en övergripande bild av CallGuide som plattform och då i synnerhet Admin-klienten. Det var till viss del ett dilemma eftersom jag inte ville lära mig för mycket om verktyget innan jag fick möjlighet att träffa verkliga användare. Samtidigt behövs en grundläggande kunskap om domänen för att kunna ställa relevanta frågor och förstå användarnas arbete. Till att börja med fick jag tillgång till samtliga manualer och en hel del utbildningsmaterial om systemet. Jag fick en kort demonstration av systemet i Telias demomiljö och deltog i en tretimmars fördjupningsutbildning i CallGuide Admin. På det här sättet kunde jag bekanta mig med terminologin och systemstrukturen utan att själv interagera med systemet, vilket jag anser vara en fördel i sammanhanget.

4.1.1 Val av angreppspunkt

Vetenskapliga metoder kategoriseras generellt som antingen kvantitativa eller kvalitativa. Båda typerna av metoder syftar till att ge en bättre förståelse av världen omkring oss, men där kvantitativa metoder omvandlar informationen till siffror och analyserar materialet statistiskt står forskarens uppfattning eller tolkning av informationen i förgrunden i de kvalitativa studierna. Kvantitativa metoder används i första hand om forskaren strävar efter hög precision och söker en förklaring till ett visst fenomen. Dessa metoder går på bredden och hämtar information från en stor population för att kunna beräkna fram genomsnitt och generaliserbara värden. Kvalitativa metoder lämpar sig å andra sidan bättre om forskaren intresserar sig för variation, sammanhang och struktur och vill skapa en ökad förståelse för en viss företeelse. Kvalitativa metoder går på djupet och samlar in riklig information om en liten population. (Holme, & Krohn-Solvang, 1997) Utifrån syftet med detta examensarbete ansågs en kvalitativ ansats vara en naturlig utgångspunkt för arbetets utformning.

4.1.2 Val av utvärderingsmetoder

Enligt Gulliksen och Göransson (2002) har mängden utvärderingsmetoder inom användarcentrerad utveckling traditionellt sett alltid varit mycket stor. Författarna menar att denna snårskog av olika metoder och tillvägagångssätt ofta tillåts stjäla för mycket uppmärksamhet från det som är viktigt, nämligen själva utvärderingsprocessen och inte de enskilda metoderna.

Inom området skiljer man på indirekta och direkta eller deltagande metoder. Indirekta metoder utgörs av olika typer av expertutvärderingar där utvärderingen görs utan medverkan av verkliga användare, till exempel heuristisk utvärdering. I motsats till detta så innebär direkt metod att riktiga användare involveras i utvärderingen på endera sättet. Exempel på deltagande metoder är bland annat användarobservationer,

intervjuer, enkäter, användningstester och laboratorieutvärderingar. (Gulliksen & Göransson, 2002; Ottersten & Berndtsson, 2002) Val av metod kan exempelvis bero på var i systemets livscykel man befinner sig, användarnas och uppgiftens karaktäristik, produkten eller systemet i sig, vilken typ av mätinstrument och information som finns till hands, omedelbarheten i responsen, vilken nivå av inblandning och resurser som krävs, olika typer av begränsningar som påverkar projektet, samt vilken grad av expertis inom MDI-området som finns inom organisationen eller i projektteamet. (Dix et al., 1998; Gulliksen & Göransson, 2002)

Gulliksen och Göransson rekommenderar att utvärderingen i första hand baseras på deltagande metoder och att enkla metoder bör föredras framför metoder av mer komplex karaktär. Därtill anser författarparet att enkäter är överskattade eftersom det oftast krävs mycket stor beteendevetenskaplig kunskap för att kunna sätta samman en enkät som ger de svar som eftersöks. Alltför ofta förbises även den stora tidsåtgång som krävs för att konstruera och prova ut enkäten samt för att analysera provsvaren.

Något som rekommenderas från flera olika håll är vikten av att inte förlita sig på en ensam metod, utan att kombinera olika metoder med varandra. På det här viset kan metodernas respektive för- och nackdelar väga upp varandra. Det är framför allt kombinationen mellan någon form av användartest eller användarobservation och expertutvärdering som förespråkas. Ett vanligt tillvägagångssätt är att man först gör en heuristisk utvärdering för att upptäcka och korrigera mer uppenbara användbarhetsproblem, och sedan testa systemet på verkliga användare för att på så sätt fånga upp deras synpunkter om både genomförda ändringar och kvarvarande problem. (Gulliksen & Göransson, 2002; Nielsen, 1993)

Utgångspunkten i metodvalet har å ena sidan varit ovanstående rekommendationer, d v s. att basera utvärderingen på deltagande metoder, att kombinera olika metoder med varandra, samt att avstå från enkäter. Å andra sidan har jag tagit fasta på de riktlinjer som författarparen Beyer och Holtzblatt (1998) samt Ottersten och Berndtsson (2002) ger i anslutning till projekt av det här slaget (se avsnitt 3.6). Ottersten och Berndtsson föreslår de båda metoderna expertutvärdering och användningstest medan Beyer och Holtzblatt menar att det behövs en process som inkluderar de kritiska strukturella elementen av Contextual Design där Contextual Inquiry och User Environment Design utgör huvuddelarna. Eftersom användningstest och CI är två datainsamlingsmetoder som involverar verkliga slutanvändare har jag valt bort användningstest till förmån för Contextual Inquiry på grund av tids- och utrymmesbrist, vilket även hänger ihop med arbetets parallella fokus rörande menystrukturen.

En heuristisk utvärdering är enkel att göra och även förhållandevis billig, vilket också brukar framhållas som metodens största fördel. En nackdel är däremot att resultatet kan variera avsevärt beroende på utvärderarens kompetens. (Gulliksen & Göransson, 2002) Contextual Inquiries rekommenderas av Beyer och Holtzblatt för denna typ av undersökning och är i praktiken en blandning mellan intervju och observation. Metoden kallas därför ibland för observationsintervju, men eftersom den har ett eget teoretiskt ramverk som jag tycker är viktigt har jag valt att använda mig av originalnamnet. För att genomföra Contextual Inquiries behövs en lämplig population av användare. Ett systematiskt urval av användare förutsätter dock en omfattande kunskap om den grupp som undersökningsenheterna ska väljas från (Holme & Krohn-

Solvang, 1997). För att ytterligare öka mina kunskaper om CallGuide Admin, samt systemets användare, deras arbetsuppgifter och behov valde jag därför att börja datainsamlingen med att genomföra så kallade informantintervjuer med ett par anställda på Telia med god kännedom om systemets användare.

4.2 Datainsamling

De metoder som valts ut för utvärderingen eller datainsamlingen är informant-intervjuer, Contextual Inquiry och heuristisk utvärdering. Nedan följer en beskrivning av metoderna och hur de tillämpats inom ramarna för detta examensarbete.

4.2.1 Informantintervjuer

En informant är en person som själv står utanför den företeelse som studeras men ändå har värdefull kunskap eller information att dela med sig av (Holme & Krohn-Solvang, 1997). Det kan till exempel röra sig om en läkare som träffar sjuka och kan ha kunskap både om deras sjukdomsbild och situation utan att för den skull själv ha sjukdomen. I det här fallet utgörs informanterna av personal på Telia som på olika sätt kommer i kontakt med CallGuide-användare. För att få en så rik bild som möjligt var ambitionen att intervjua personer med olika befattningar som kan antas träffa olika typer av användare i olika sammanhang och dessutom ha olika infallsvinklar och förhållningssätt till produkten. Med bakgrund i detta valdes en handfull personer ut:

- Lena Falkman, ansvarig för utveckling och förvaltning av Telias egen kundtjänstplattform
- Jonas Fogelberg, affärsstöd
- Anna Grahn-Östlund, utbildningsansvarig för produkten CallGuide
- Jessica Strömberg, uppdragsbeställare åt kund
- Anna Törnqvist, leveransprojektledare för Contact Center-lösningar

Som underlag för intervjuerna togs en intervjuguide fram och den återfinns i bilaga A. Intervjuerna genomfördes under december månad 2006 i mitt eget kontorsrum i Telias lokaler på Strandbodgatan i Uppsala och varade mellan 30 till 45 minuter. Samtliga intervjuer spelades in med hjälp av en mp3-spelare och renskrevs därefter omsorgs-fullt. Det sammanfattade intervjumaterialet fungerade initialt som en kunskapskälla gällande CallGuide och dess användare och var till stor hjälp vid urvalet av användare till de fortsatta undersökningarna.

4.2.2 Contextual Inquiry

Enligt Holme och Krohn-Solvang (1997) förutsätter kvalitativ metod fysisk närhet, ömsesidig tillit och förståelse mellan forskare och undersökta personer. Utmaningen för forskaren är att i möjligaste mån försöka få fram de uppfattningar som är de intervjuade personernas egna. Författarna menar att detta är något man uppnår bäst genom att framstå som en engagerad men något naiv och ovetande forskare och gå in i rollen som den *intresserade lyssnaren*. Detta är något som Beyer och Holtzblatt tar fasta på i den metod som de kallar Contextual Inquiry (CI).

4.2.2.1 Contextual Inquiry enligt Beyer och Holtzblatt

Contextual Inquiry är en kvalitativ och ostrukturerad datainsamlingsteknik i fält som går ut på att ett fåtal noggrant utvalda användare studeras mer ingående för att skapa

en bredare och djupare förståelse för den arbetssituation och det arbetssätt som generellt kan sägas gälla för samtliga användare av systemet. Rent praktiskt är Contextual Inquiry en form av observationsintervju som vilar på fyra grundläggande principer, vilka intervjuaren ska ha i åtanke:

- **sammanhang** – åk ut till användarnas arbetsplats och se hur de utför sina arbetsuppgifter i sin egen arbetsmiljö.
- **partnerskap** – samtala med användarna om deras arbete och hjälps åt med att sätta ord på ditintills dolda aspekter med arbetet.
- **tolkning** – utveckla en delad förståelse med användarna rörande vilka aspekter med arbetet som är betydelsefulla.
- **fokus** – styr utfrågningen utifrån en tydlig förståelse av det egna syftet med utvärderingen. Det centrala är att intervjun handlar om det arbete som användaren utför snarare än hur systemet ska designas.

Contextual Inquiry handlar med andra ord om att åka ut till användarna, att observera dem i arbetet och att samtala med dem om det arbete de utför. Metoden tillhandahåller också riktlinjer för själva intervjusituationen, men istället för att sätta upp en lista med regler och instruktioner om vilka moment som måste ingå och vad som bör åstadkommas under utfrågningen riktar CI snarare in sig på hur man som intervjuare bör agera i samtalet med användaren.

Beyer och Holtzblatt benämner Contextual Inquiry som en relationsmodell, en modell som ger råd om hur man som utredare ska förhålla sig till respondenten, vilka relationer som är bra och vilka som bör undvikas. Författarna menar att olika relationer för med sig olika attribut och beteenden och att vissa beteenden inte är önskvärda i en forskningssituation. De exemplifierar med relationer som forskare/ forskningsobjekt, förälder/barn och besökare/värd. I och med att relationer är duala och självförstärkande kan dock forskaren genom att spela den ena parten få den andre till att ikläda sig rollen som den andra parten i en relation. Genom att vara uppmärksam på detta och välja en roll som är lämplig för sammanhanget kan utredaren skapa rätt förutsättningar för en givande intervju. När det gäller datainsamling lyfter Beyer och Holtzblatt fram relationen mästare/lärling som en god utgångspunkt. Precis som en hantverkslärling lär sig hantverket genom att studera sin mästare menar författarna att designteamet behöver skaffa sig kunskap om användarens arbete genom att studera denne. Användaren får vara experten som visar utredaren hur arbetet går till. En utredare som ikläder sig rollen som lärling uppvisar dessutom automatiskt den ödmjukhet, frågvishet och uppmärksamhet som behövs för att samla in bra data.

En utfrågning bör i vanliga fall ta någonstans mellan två och tre timmar. För att man ska få med så många olika perspektiv och infallsvinklar som möjligt hjälps därefter hela designteamet åt att analysera och tolka materialet. Beyer och Holtzblatt menar att det här är en metod som kan användas av såväl interna IT-avdelningar som kommersiella mjukvaruutvecklare för att komma till rätta med potentiella användbarhetsproblem genom att grunda designprocessen i detaljerad, pålitlig kunddata.

4.2.2.2 Metodanpassning

Det har inte funnits möjlighet att diskutera resultatet från intervjuerna tillsammans med designteamet, utan tolkningen har utförts av utredaren själv. Det har heller inte alltid varit möjligt att be användarna *visa* hur deras arbete kunde se ut på grund av

systemets komplexa natur och det faktum att vissa handlingar slår igenom på så många olika ställen i systemet och kräver många sekventiella operationer. Framför allt hade det kunnat ställa till onödigt arbete med att rensa upp i systemet efteråt. Detta gäller i första hand handlingar som kräver att man lägger till nya organisationsområden, delområden, kompetenser eller liknande. Detta har dock inte upplevts som något hinder i utvärderingen.

4.2.2.3 Urval av användare

Även om statistisk generalisering och representativitet inte är några centrala syften i kvalitativa metoder blir ändå urvalet av undersökningspersoner en avgörande del av undersökningen. Eftersom kvalitativa metoder syftar till att öka förståelsen och kunskapen om ett visst fenomen behöver urvalet göras systematiskt utifrån vissa medvetet formulerade kriterier. Detta kallas för att göra ett *stratifierat urval* istället för att låta slumpen eller tillfälligheter styra. En grundläggande utgångspunkt är att försöka få med så rik och nyanserad kunskap som möjligt. Genom att exempelvis säkra god variationsbredd i urvalet fås ett stort informationsinnehåll och kvaliteten på informationen blir högre om urvalet dessutom består av intervjupersoner som kan antas ha rikligt med kunskap om det som ska undersökas. Därtill är intervjupersonernas förmåga att uttrycka sig och villighet att delta viktiga faktorer att ta med i beräkningen. (Holme & Krohn-Solvang, 1997)

4.2.2.3.1 Urvalskriterier och rekommendationer

Målet med Contextual Inquiry är enligt Beyer och Holtzblatt att hitta en gemensam och underliggande arbetsstruktur som kan sägas gälla för hela kundbasen. För att åstadkomma detta på bästa sätt rekommenderar författarna att utfrågningen inkluderar ett antal mycket olika användarorganisationer och menar att urvalet bör baseras på skillnader i företagens och användarnas arbetssätt. Kontors- och terminalarbete som sådant är dock mycket lika oavsett vilket marknadssegment man tittar på, så för att hitta skillnader i arbetssätt är istället faktorer som affärsstrategier, kulturella skillnader, fysisk utbredning och storlek av större betydelse som urvalskriterier. (Beyer & Holtzblatt, 1998) Gulliksen och Göransson (2002) plockar ut ålder, kön, kultur, verksamhetserfarenhet, yrkeskategorier, organisationsstorlek, geografisk spridning och gruppstorlek som möjliga urvalsfaktorer och poängterar också vikten av att deltagandet är frivilligt. Ålder, kön och kultur nämns även av Ottersten och Berndtsson (2002) som därutöver lyfter fram kunskap, erfarenhet, arbetsuppgifter, användningsfrekvens, värderingar, mål och fysisk miljö.

När valet av urvalskriterier är klart gäller det att hitta 2-3 personer ur varje målgrupp som är villiga att delta i utvärderingen. Totalt sett bör antalet undersökningspersoner uppgå till någonstans mellan 10 och 20. I de fall då undersökningen fokuserar på en enda användarroll eller riktar sig mot detaljerad gränssnittsinteraktion istället för en övergripande arbetsprocess är det ofta tillräckligt med 6-10 personer. Om utvärderingen gäller en kommersiell mjukvara bör åtminstone 4-6 olika företag involveras för att säkra variationsbredden i det insamlade materialet. (Beyer & Holtzblatt, 1998)

Enligt Beyer och Holtzblatt kommer uppfattningen om vad som behöver tillkomma, vilken information som saknas, att ändras allt eftersom intervjuerna genomförs och de rekommenderar därför att alla intervjuer inte bokas in på en gång. Holme och Krohn-Solvang (1997) är inne på samma spår och menar att det alltid kommer en punkt då man blir mätt på information. Vid det tillfället bör också insamlingen av information

upphöra. Om det skulle visa sig att informationen inte var tillräcklig eller att den behöver preciseras är det alltid möjligt att gå tillbaka till undersökningspersonerna och be att få mer information vid ett senare tillfälle.

4.2.2.3.2 Mitt urval

De kriterier som använts för att välja ut lämpliga organisationer och användare för utvärderingen av CallGuide Admin har i första hand varit av pragmatisk karaktär och är förknippade med Gulliksen och Göranssons kommentar om frivillighet. Här kommer det in en annan aspekt kring utvärderingar av kommersiella produkter, nämligen att användarna måste ha tid och möjlighet och framför allt vilja att deltaga. Då resultatet av utvärderingen inte nödvändigtvis kommer kunden till godo inom en snar framtid krävs andra motivationsfaktorer. För att leverantören ska kunna be om en tjänst av det här slaget är det därför en grundförutsättning att relationen till kunden är god. En stark och positiv relation behöver dock inte betyda att kunden är alltigenom nöjd och inte har synpunkter på verktyget, utan snarare att det finns ömsesidigt engagemang parterna emellan. Med anledning av detta gjordes ett första urval som bestod av kunder som varit med ett tag och har en väletablerad relation till Telia.

En annan faktor som tagits med i beräkningen är typ av CallGuide-lösning och ambitionen var att få en jämn fördelning mellan användare av CallGuide Enterprise och CallGuide VCC. Personliga faktorer så som ålder, kön, utbildning och bakgrund har inte varit i fokus. Uppgifter om kön har däremot tagits med i presentationen för att ge läsaren möjlighet att skapa sig en tydligare bild av de enskilda användarna. Av de urvalskriterier som omnämnts i teorin baserades det fortsatta urvalet på i huvudsak tre faktorer. Innebörden av dessa faktorer har tolkats utifrån sammanhanget och kriterierna har i enlighet med detta tilldelats olika etiketter och mätinstrument:

- **organisationsstorlek** – antal licenser för Agent respektive Admin
- **arbetsuppgifter/målgrupp** = användargrupp – tekniker eller administratör
- **affärsstrategi** = organisationstyp – myndighet eller privat företag

Det slutgiltiga urvalet befinner sig inom ramarna för de rekommendationer av Beyer och Holtzblatt som presenteras på föregående sida. Sammantaget har sju intervjuer med åtta olika användare genomförts, varav fem är tekniker och tre administratörer. Användarna arbetar vid fem olika företag eller organisationer varav tre har valt CallGuide VCC och de resterande två kör CallGuide Enterprise.

Användarurval till Contextual Inquiries

	Kön	Användargrupp	Organisationstyp	Antal Agent-licenser	Antal Admin-licenser	CallGuide	Version
1	2 män	2 tekniker	myndighet	923	52	Enterprise	4.0
2	1 man	1 tekniker	myndighet	290	25	VCC	4.0.5.4
3	2 män	1 tekniker, 1 administratör	privat företag	175	6	VCC	4.0.5.5
4	2 män	2 administratörer	halvstatligt företag	2330	15*	Enterprise	5.1
5	1 kvinna	1 tekniker	myndighet	252	15	VCC	5.2.2.9

*100 licenser, men endast 15 användare har enligt uppgift behörighet att arbeta med Admin

Tabell 1: Tabell över användarurvalet till Contextual Inquiries.

4.2.2.4 Genomförande

Alla sju intervjuer genomfördes under en treveckorsperiod i månadsskiftet januari-februari 2007. I samtliga fall fanns möjligheten att åka ut till användarnas respektive arbetsplatser och genomföra intervjuerna i deras egen arbetsmiljö. En av intervjuerna hölls i ett konferensrum, alla övriga vid användarens egen datorarbetsplats. För att ingen information skulle gå förlorad dokumenterades sessionerna både med hjälp av stödordsanteckningar och inspelningsutrustning i form av en mp3-spelare.

Intervjumaterialet transkriberades sedan så noggrant som möjligt med ambitionen att lämna tolkningar och analys till ett senare skede. Det renskrivna materialet delades in i fyra olika avsnitt: information om företaget och användaren, generella system-kommentarer, fönsterspecifika synpunkter samt mina egna intryck. Läsaren kommer att finna att resultatdelen i denna rapport är disponerad på samma sätt. Parallellt med transkriberingen gjordes även kontroller av användarnas synpunkter gentemot referensversionen av CallGuide Admin (5.2.2.9). Synpunkter och upplevda brister som redan avhjälpts i denna version summerades i ett dokument och sändes tillbaks till respektive användare för att ge någon form av återkoppling och samtidigt tacka för deras tid och hjälp. Efter att alla utfrågningar genomförts och transkriberats sammanfattades slutligen alla kommentarer om gränssnittet, generella såväl som fönster-specifika, i ett och samma dokument. Detta dokument utgör stommen till huvuddelen av det redovisade resultatet från dessa intervjuer.

Intervjusituationen efterliknade i möjligaste mån Contextual Inquiry så som metoden presenteras av Beyer och Holtzblatt. De intervjuade användarna uppmuntrades att berätta om sitt arbete och att visa hur de går tillväga för att utföra sina dagliga arbetsuppgifter. Det var en mycket positiv upplevelse och erfarenhet för mig som intervjuare att gå in med den sanningsenliga hållningen att jag inte kunde så mycket men gärna ville lära mig mer. Längden på intervjuerna varierade från en halvtimme till uppemot två och en halv timmar. Riktlinjerna föreskriver visserligen att man bör lägga cirka två timmar per användare men jag upplevde inte att någon intervju blev för kort. När alla intervjuer var avklarade kändes det snarare som att fler intervjuer inte skulle ha tillfört någon ny information av vikt, att jag precis så som Holme och Krohn-Solvang beskriver det hade blivit mätt på information.

Efter önskemål från Telia genomfördes kompletterande telefonintervjuer med tre av användarna i mitten av mars. Dessa intervjuer handlade uteslutande om telefonboken i CallGuide och syftade till att få mer specifik information angående användarnas synpunkter och arbetssätt vad gällde denna funktion. Detta material arbetades sedan in i dokumentet med de sammanfattade användarintervjuerna.

4.2.3 Heuristisk utvärdering

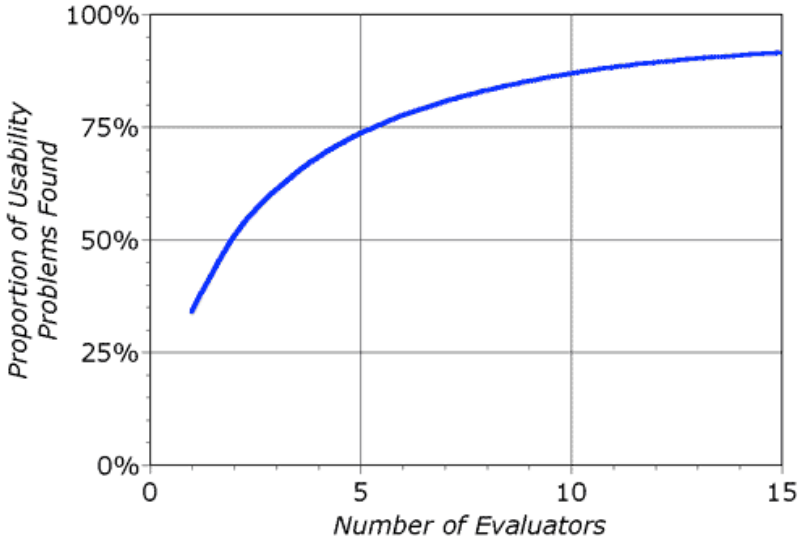
Adjektivet *heuristisk* betyder "som tjänar till att (på egen hand) upptäcka, uppfinna eller bilda ny kunskap". Ordet heuristik kommer ursprungligen från grekiskan och betyder just upptäcka. Något som betecknas som heuristiskt kan inte betraktas som strikt och slutgiltigt utan bara som möjligt och provisoriskt, med syfte att leda till upptäckten av det aktuella problemets lösning. Det är en mer eller mindre säker gissning som får duga innan den riktiga lösningen på problemet blivit fastställd. (Polya, 1979; susning.nu, 2003) I linje med detta är en heuristisk utvärdering en utvärdering som syftar till att upptäcka eller uppdaga *möjliga* användbarhetsproblem.

Att det verkligen är ett problem kan endast avgöras genom att studera och samla information om verkliga användare.

4.2.3.1 Teori och bakgrund

Den typ av heuristisk utvärdering som kommit att bli en välanvänd metod inom MDI utvecklades av Jakob Nielsen och Rolf Molich i början av 1990-talet. De presenterar en metod för att strukturera kritik mot ett system genom att göra en systematisk inspektion av användargränssnittet med hjälp av en samling relativt enkla och generella principer. (Dix et al., 1998) Målet med utvärderingen är att hitta och uppmärksamma användbarhetsproblem i gränssnittsdesignen så att de kan tas itu med som en del av den iterativa designprocessen. Heuristisk utvärdering tillhandahåller däremot inget systematiskt sätt att generera lösningar eller något sätt att bedöma kvaliteten av ny design. Nielsen menar dock att många användbarhetsproblem har relativt uppenbara lösningar när de väl lyfts fram i ljuset. (Nielsen, 1993)

En heuristisk utvärdering går till så att en expert eller utvärderare går igenom gränssnittet flera gånger och tittar på olika aspekter med designen och jämför dem med listan över erkända användbarhetsprinciper. En session bör ta någonstans mellan en till två timmar beroende på gränssnittets omfattning och syftet med utvärderingen. Resultatet av utvärderingen är en lista med användbarhetsproblem kommenterad med referenser till de användbarhetsprinciper som gränssnittsdesignen bryter mot. (Nielsen, 1993)



***Figur 3**: Graf över hur stor del av förekommande användbarhetsbrister som förväntas bli upptäckta i en heuristisk utvärdering i relation till antalet medverkande experter.*

Enligt Nielsen har erfarenheten visat att olika experter hittar olika typer av problem och för att få ett så fylligt material som möjligt bör tre till fem utvärderare göra varsin utvärdering av gränssnittet utan att diskutera sinsemellan. Först när varje enskild utvärdering är genomförd sammanfattas resultatet. Bäst resultat fås om alla deltagande utvärderare har kunskap både inom MDI och den aktuella domänen. En sådan expert hittar uppemot tre gånger fler problem än en nybörjare som endast har generell datakunskap, men ingen specifik kunskap om vare sig MDI eller domän. Heuristisk utvärdering kan dock utföras av personer utan egentlig expertkunskap och även om fler utvärderare kommer att hitta fler problem så är en bättre än ingen (Stanford University, 1999-2001).

En heuristisk utvärdering kan genomföras när som helst i designprocessen. Det vanligaste är dock att den används på ett tidigt stadium för att ge vägledning om mer eller mindre färdiga designförslag, eller i ett senare skede för att utvärdera en befintlig produkt. (Nielsen, 1993) De fördelar som lyfts fram är med heuristisk utvärdering är att det är en snabb och relativt billig metod för att få feedback på ett designförslag eller ett befintligt system. Att utvärderingen görs utan användare bidrar till att hålla kostnaderna nere, men samtidigt är det en nackdel eftersom de problem som uppdagas inte baserar sig på användarnas *behov*. (U.S. Department of Health and Human Services)

4.2.3.2 Val av designheuristiker

Det finns många olika författare som gett sig i kast med att ta fram riktlinjer och principer för användbarhet och i detta arbete har ett flertal olika samlingar med principer studerats och begrundats. De designheuristiker som Nielsen och Molich presenterade 1990 reviderades ett par år senare av Nielsen och är en återkommande syn i en stor del av både den samtida och den nutida MDI-litteraturen. Det finns dock även de som är kritiska mot Nielsens principer. I ett nyhetsbrev på en välrenommerad Internetsida om användbarhet skriver man till exempel att det inte finns några vetenskapliga bevis för att ett gränssnitt verkligen kan förbättras genom att tillämpa och följa Nielsens principer. Här lyfter man istället fram en samling principer som tagits fram av Jill Gerhardt-Powals (1996), vilka baseras på empirisk forskning och har sin utgångspunkt i människans kognitiva funktioner. (Human Factors International Inc., 1999) Efter att ha läst igenom sagda principer blev jag däremot inte övertygad av deras lämplighet som utgångspunkt för en heuristisk utvärdering, då fokus ligger på kognition och frågor som hur man bör organisera och presentera data för att underlätta arbetet för den mänskliga hjärnan. Andra bitar som också är viktiga i interaktions-sammanhang som exempelvis fel och felhantering, feedback och flexibilitet har lämnats utanför. Min uppfattning är därför att Gerhardt-Powals lista lämpar sig bättre som komplement till andra riktlinjer att stämma av emot i ett utvecklingsprojekt.

En samling designprinciper som får anses vara relativt ny i sammanhanget är Bruce Tognazzinis First Principles of Interaction Design från 2003. Tognazzini har gjort ett grundligt arbete och presenterar en lista över 17 olika principer tillsammans med ingående tolkningsförklaringar (Tognazzini, 2003). Även Jakob Nielsen har uppmärksammat detta nytillskott inom diskursen och hans kommenterar är att: *"The list is slightly too long for heuristic evaluation but serves as a useful checklist"* (Nielsen, 2005). Att listan är för lång för att användas i en heuristisk utvärdering är också min egen uppfattning. Det skulle framför allt bli mycket tidskrävande att koppla ihop varje problem med rätt princip och frågan är om utvärderingens kvalitet verkligen skulle bli bättre. Samma sak gäller för de principer som presenteras av the Usability Body of Knowledge på hemsidan www.usabilitybok.org (Usability Professionals' Association, 2005). Här har man gjort ett bra försök att sammanfatta sju olika uppsättningar med designprinciper av bland andra just Nielsen, Gerhardt-Powals och Tognazzini. Denna lista är dock av naturliga skäl ännu mer omfångsrik och lämpar sig därför inte heller som utgångspunkt för en heuristisk utvärdering.

4.2.3.3 Nielsens 10 designheuristiker

Med bakgrund i ovanstående resonemang fann jag att Nielsens principer trots allt var mest användbara av de alternativ som stod till buds och har därför valt att använda

dem i den heuristiska utvärdering som genomförts på CallGuide Admin. Följande lista är en egen översättning av dessa designheuristiker som de presenteras på Jakob Nielsens hemsida www.useit.com (Nielsen, 2005):

1. Synlig systemstatus	Systemet skall alltid hålla användaren informerad om vad som händer genom passande feedback inom rimlig tid.
2. Anpassning mellan system och verklighet	Systemet skall tala användarens språk och utgå från de ord, fraser och koncept som användaren är bekant med och använda sig av konventioner från den riktiga världen istället för systemorienterade termer. Information skall visas i en naturlig och logisk följd.
3. Användarkontroll och användarfrihet	Användare väljer ofta väljer ofta funktioner i systemet av misstag och behöver tydligt markerade vägar som på ett snabbt och enkelt sätt leder användaren tillbaks till utgångsläget. Stöd funktioner för att ångra och göra om.
4. Konsekvens och standarder	Användare ska inte behöva fundera över om olika ord, situationer eller handlingar betyder samma sak. Följ gällande plattforms-standard.
5. Förhindra fel	Vad som är ännu bättre än välformulerade och instruerande felmeddelanden är om designen utformas så omsorgsfullt att felen inte uppstår i första taget.
6. Igenkänning framför återminnelse	Gör objekt, handlingar och val synliga. Användaren ska inte behöva komma ihåg information från en del av dialogen till en annan. Instruktioner och anvisningar för hur systemet skall användas skall vara synliga eller lätta att komma åt vid behov.
7. Flexibilitet och effektivitet	Erbjud genvägar till den avancerade användaren genom att implementera olika former kortkommandon, osynliga för den mer ovana användaren. Låt användaren själv skräddarsy de vanligaste arbetsmomenten.
8. Estetisk och minimalistisk design	Dialoger skall inte innehålla information som är överflödig i sammanhanget eller information som sällan används. All redundant information i en dialog gör det svårare för användaren att urskönja och sortera ut den relevanta informationen.
9. Felkorrigering	Hjälp användaren att känna igen, diagnostisera och korrigera fel. Felmeddelanden skall inte innehålla koder som användaren i sin tur måste tolka, utan formuleras på ett klart och tydligt sätt, precisera problemet och föreslå en konstruktiv lösning för att avhjälpa problemet.
10. Hjälp och dokumentation	Även om det är bättre ifall systemet kan användas utan dokumentation så kan det vara nödvändigt att tillhandahålla sådana funktioner. All sådan typ av information skall vara sökbar, fokusera på användarens uppgift, lista konkreta steg som behöver utföras, samt inte vara allt för omfattande.

Tabell 2: Nielsens 10 designheuristiker.

4.2.3.4 Genomförande

På grund av systemets höga komplexitet genomfördes den heuristiska utvärderingen efter att kartläggningen av Admin hade avslutats. På detta sätt fanns gränssnittet tillgängligt i två olika kompletterande format, vilket underlättade arbetet avsevärt. Kartläggningen blev i själva verket en smygstart av utvärderingen eftersom jag vid detta tillfälle kom i kontakt med alla delar av systemet för första gången och kunde anteckna mina allra första intryck. När den heuristiska utvärderingen genomfördes fungerade kartläggningen som ett referensmaterial att bläddra i för att hitta skillnader

och likheter mellan olika delar av systemet. Med menyhierarkin som naturlig utgångspunkt gjordes flera systematiska genomgångar av klienten och noggranna anteckningar fördes parallellt. En första genomgång syftade till att hitta dialog-specifika problem, medan ytterligare granskning avsåg att åskådliggöra mer övergripande mönster.

Jag tror precis som man uttrycker det på www.usability.gov att resultatet från den heuristiska utvärderingen måste ses i ljuset av hur användare interagerar med gränssnittet på riktigt så att fokus inte hamnar på fel saker och energi läggs på att ändra sådant som egentligen inte behöver ändras. (U.S. Department of Health and Human Services) Syftet med den heuristiska utvärderingen har därför varit att sätta in användarnas kommentarer och synpunkter i ett större sammanhang och att visa vilka principer som bör fokuseras på i första hand i vidareutvecklingen av systemet. Resultatet av den heuristiska utvärderingen redovisas därför som en sammanfattning av förekommande användbarhetsproblem med bakgrund i de valda designprinciperna.

4.3 Kartläggning

Under förarbetet med att sätta upp ramarna för examensarbetet visade det sig mycket tidigt att det skulle behöva göras en explicit kartläggning av systemets nuvarande struktur på grund av gränssnittets höga komplexitet. Efter att ha sökt i litteraturen och inte funnit någon teoretiskt förankrad metod att utgå ifrån blev det kvarvarande alternativet att ta fram en egen procedur för detta. Kartläggningen bygger därför på ett mer handfast tillvägagångssätt med syftet att helt enkelt fånga de många fönstren och systemkomplexiteten på papper. Detta har åstadkommits genom att systematiskt utforska systemets olika delar och med hjälp av programmet PaintShop Pro 4 ta skärmdumpar av hela gränssnittet. Dessa har sedan skrivits ut och sorterats in i en pärm med en flik för varje menyval i huvudmenyn. För att få en helhetsbild av klientens olika menyval innefattar kartläggningen även en schematisk skiss över menystrukturen, vilken utförts med hjälp av diagramtypen o*rganisationsschema* i Microsoft Office Word 2003.

För att även ge något sorts mått på systemets komplexitet innehåller kartläggningen uppgifter om hur många noder och löv systemet är uppbyggt av. Terminologin är lånad från datalogin och ämnesområdet datastrukturer där den appliceras just på strukturformer så som hierarkiska träd. (Weiss, 1999)

- *Nod* – ett fönster eller en dialogruta som länkar vidare till andra dialogrutor.
- *Löv* – ett fönster som inte länkar vidare, utan bara har en koppling till noden ovanför i hierarkin.

De löv som räknats är de som kräver någon form av inmatning från användaren. Därutöver finns med andra ord en mängd fönster som endast innehåller information till användaren, samt dialoger där användaren ombeds bekräfta en inmatning eller ett val genom att klicka okej eller avbryt. Även om felmeddelanden är viktiga ur använd-barhetssynpunkt anses de inte tillföra något i detta sammanhang då syftet snarare är att försöka ge en bild av storleken på användarens klientinterna arbetsplats.

Resultatet av kartläggningen var i första hand tänkt att ligga till grund för arbetet med att ta fram en ny menystruktur för klienten. På grund av systemets många delar och

höga komplexitet fick denna artefakt även en viktig funktion i arbetet med den heuristiska utvärderingen då den utgör en fysisk representation av systemet där det är lätt att göra anteckningar i direkt anslutning till en dialogruta och även lätt att snabbt gå igenom hela gränssnittet och jämföra olika delar på ett sätt som inte är möjligt i programvaran på grund av dess modala grundstruktur.

4.4 Analys och rekommendationer

Resultatet från intervjuerna och den heuristiska utvärderingen kommer att fogas samman till en lista och förses med rekommendationer till förändring av gränssnittet för att förbättra användbarheten. En övergripande analys har gjorts med utgångspunkt i ISO-definitionen för användbarhet (se avsnitt 3.2). Därutöver baseras analysarbetet i det här fallet inte på någon uttalad metod, utan tar istället avstamp i ett uttalande av Jakob Nielsen. Denna profil inom användbarhetsområdet menar att många användbarhetsproblem egentligen har relativt uppenbara lösningar när de väl lyfts fram i ljuset. Avsaknaden av analysmetod ses därför inte som något större problem i sammanhanget. Det viktiga är istället att försäkra sig om att de problem som upptäcks verkligen är relevanta ur ett användarperspektiv.

Med bakgrund i kartläggningen och användarnas synpunkter om navigering och struktur kommer en ny menystruktur att tas fram med stöd av grundläggande teori om hierarkiska menystrukturer samt den av Beyer & Holtzblatt framtagna metoden User Environment Design.

5 Resultat

Resultatet presenteras i kronologisk ordning och inleds därför med en sammanfattning av informantintervjuerna, därefter följer resultatet från Contextual Inquiries, kartläggningen och den heuristiska utvärderingen i nämnd ordning.

5.1 Informantintervjuer

I följande avsnitt ges en sammanfattad presentation av de intervjuer som genomförts med personal på Telia i syfte att inhämta mer kunskap om CallGuide Admin, klientens användare och deras arbetsuppgifter.

För hela CallGuide som plattform finns det tre olika användargrupper, nämligen systemadministratörer, kundtjänstledare och agenter. När det gäller CallGuide Admin urskiljer intervjupersonerna däremot endast två olika grupper där den ena utgörs av system- eller driftsansvariga och den andra gruppen består av personer med ett mer administrativt ansvar. I de flesta fall är dessa grupper synonyma med huvudgrupperna systemadministratörer respektive kundtjänstledare. Administrativa roller kan däremot även finnas på olika nivåer inom en och samma organisation på så sätt att en del personer behöver ha överblick över hela systemet, medan andra bara ska ha koll på sina egna anknytningar och handläggare.

De mest grundläggande funktionerna i Admin som alla användare behöver kunna är att administrera användare och anknytningar, ändra öppettider samt modifiera kompetenser och behörigheter. Om man har ett system som tuffar och går och är hyfsat väl anpassat efter verksamheten så klarar man sig långt med dessa bitar, men kan inte gå in och skriva på systemet för att optimera det. Vilka funktioner som tillkommer i nästa steg beror på hur pass avancerat kundens system är och alla har inte så invecklade system att de måste lära sig alla finesser. De områden som mer avancerade användare intresserar sig för i första hand verkar dock vara styrning av kontakter, konfiguration av köer och väntelistor samt inställningar av användargränssnittet i CallGuide Agent.

Hur mycket tid de olika kundorganisationerna lägger på administrering och konfiguration i Admin varierar också mycket beroende på organisationens verksamhet och teknikmognad. I de allra flesta fall utgör CallGuide bara ett av många olika stödsystem som kunderna använder för att bedriva sin verksamhet och som leverantör arbetar Telia med att visa kunderna vilka fördelar systemet kan ge om det används på rätt sätt. För de organisationer som lär sig att se och använda de olika funktioner som CallGuide erbjuder kan systemet bli mer av ett verksamhetsstöd än ett verktyg för att hantera inkommande och utgående kontakter. En del går till exempel in och skruvar i styrningen nästan dagligen i sin strävan efter att utforma mer avancerade flöden för att hitta den optimala brytpunkten mellan väntetid och kompetens.

Andra styrkor med Admin som intervjupersonerna upplever att användarna uppskattar är bland annat att allt från systemparametrar och licensnycklar till öppettider och användarnamn finns i ett och samma verktyg. Dessutom framhålls det värdefulla i att systemet faktiskt levereras på svenska med svensk branschterminologi. Genom sina olika kontakter med användare får intervjupersonerna också ta del av användarnas

synpunkter på verktyget. Ett par vanligt förekommande kommentarer är bland annat att man efterfrågar en sökfunktion i klienten och önskar att det fanns möjlighet att exportera mer information till Excel. Vidare har det framkommit synpunkter på att klienten är svår att överblicka och att man i vissa lägen skulle vilja kunna ha flera fönster öppna samtidigt. Ett par av intervjupersonerna menar att de själva skulle önska att systemet var mer självinstruerande och att det tillhandahöll mer interaktiv information och tydligare felmeddelanden för att vägleda användandet.

5.2 Observationsintervjuer – Contextual Inquiries

Resultatet av de observationsintervjuer som genomförts med användare av CallGuide Admin presenteras i fyra olika delar. Först kommer en kort presentation av var och en av de fem *organisationer* som besökts. Därefter följer två avsnitt med dels generella, dels fönsterspecifika synpunkter som fångats upp under dessa observationsintervjuer. Intervjumaterialet är sammanfattat för samtliga användare och har förvisso blivit föremål för ansenlig tolkning, men presenteras här utan analys. För att underlätta för läsaren och på ett tydligt sätt särskilja mina egna tankar och åsikter från användarnas synpunkter och kommentarer summeras mina iakttagelser i ett separat och avslutande avsnitt. I de fall då användare citerats hänvisar siffran inom hakparentes till respektive användare. Se referenslistan för mer information.

5.2.1 Presentation av användarna och deras respektive organisationer

Denna presentation är tänkt att ge en bild av vilka behov som finns inom respektive verksamhet, vilka arbetsuppgifter användarna har och hur deras arbetsmiljö ser ut. Här finns också uppgifter om vilka delar av systemet som användarna enligt egen utsaga använder mest och vad som helt enkelt verkar vara viktigast för dem.

5.2.1.1 Statlig förvaltningsmyndighet

De båda användarna som deltar i intervjun arbetar som kommunikationsansvariga på myndigheten, vilket innebär att de ser till att såväl CallGuide som interna telefonilösningar fungerar som de ska. Vad gäller CallGuide så är det deras uppgift att tillhandahålla tekniken och att strukturera upp systemet vad gäller organisations- och delområden, talsvar, talsvarsdialoger, köer och styrning. Administrationen av handläggare och kompetenser sköts centralt av en avdelning med controllers på ett annat kontor och de är även systemägare. Teamledarna ute på de enskilda kontoren *använder* CallGuide och har inte tillgång till Admin, utan jobbar främst med Pulse. Myndigheten har valt att köra en enterprise-lösning och sköter med andra ord mycket av arbetet med systemet på egen hand. Vid intervjutillfället har myndigheten version 4.0 av CallGuide men ska uppgradera till 5.2 under våren 2007. Enligt användarna har de inga större problem med Admin egentligen, utan är mycket nöjda på det stora hela.

Arbetsuppgifterna i Admin initieras ofta av förslag från controllergruppen. De berättar vad de vill ha och det åligger de intervjuade användarna att berätta hur man kan göra det – *om* man kan göra det – och så försöker man hitta något som passar för alla. Man har till exempel nyligen genomfört en stor förändring i CallGuide där man delat upp verksamheten i andra delområden än innan. Idag jobbar samtliga handläggare tillsammans i en stor organisation oberoende av var i landet de befinner sig, vilket innebär att alla som ringer in hamnar i en och samma kö.

De vanligaste arbetsuppgifterna som dessa båda användare har är bland annat att rätta till problem med användare, vilket oftast innebär att controllergruppen själva missat något moment när de lagt till en ny användare. De hjälper också till att ändra öppettider när det behövs och går ibland in i styrningstabellen och styr om köerna vid hög belastning eller stryper flödet vid extrema arbetstoppar. De får också mycket att göra vid större organisationsförändringar då det ska byggas om köer, menyval, organisationsområden, kompetenser och lite av varje.

5.2.1.2 Central förvaltnings- och tillsynsmyndighet

Den användare som intervjuats är systemförvaltare vad gäller telefonilösningar för myndighetens kontor i hela Sverige. Han har jobbat rätt mycket med Admin, men eftersom myndigheten har valt att köra CallGuide VCC så sköts en hel del av Telia. Totalt sett har man cirka 80-90 Admin-användare och kör för närvarande version 4.0.5.4 av CallGuide men kommer att uppgradera till en nyare version under våren 2007. Som systemförvaltare arbetar användaren på uppdrag åt myndighetens nationella kundtjänstgrupp. De bestämmer vilka förändringar som ska genomföras efter att först ha verifierat önskemålen med olika delar av verksamheten. Användarens vanligaste arbetsuppgifter i Admin berör anknytningar, platser och accesskoder, användarkonton, telefonboken, e-postkonton och styrning av kontakter.

Systemförvaltaren är nöjd med att ha Telia som helhetsleverantör och tycker på det stora hela att det fungerar jättebra. Han tycker att det är skönt att ha en utomstående leverantör som äger systemet eftersom det betyder att han själv inte alltid behöver vara på topp med sin egen kunskap, utan istället kan lita på att expertkunskapen finns hos leverantören. Som systemförvaltare ser han också helst att det är Telia som sköter all konfigurering av styrning och liknande för då blir det aldrig någon diskussion om vems fel det är ifall något i systemet inte fungerar som det ska. Om det är något han däremot skulle vilja trycka särskilt på är det att få in mer hjälp och tydligare instruktioner i systemet så att man inte behöver vända sig till manualen i lika stor utsträckning som nu.

5.2.1.3 Internationellt transport- och reseserviceföretag

På det här företaget har både en systemadministratör och en teamledare intervjuats. Organisationen har valt att köra CallGuide VCC och har idag version 4.0.5.5 av Admin. En uppgradering är dock att vänta under första halvåret 2007. Avdelningen som systemadministratören jobbar för har hand om samtalsstyrningen samt utarbetar metoder för arbetssättet. Han har själv ingen som helst teknisk utbildning bakom sig, utan är i grund och botten humanist. Däremot har han alltid varit intresserad av teknik och tycker det är roligt att lära sig nya saker, att ha en idé om hur saker och ting fungerar. Enligt egen utsaga är han väldigt petig vad det gäller datorer och anser att en dator ska göra exakt det han själv vill göra och inte vad någon tekniker har bestämt att han ska vilja göra. På det stora hela tycker han att CallGuide fungerar rätt okej, men menar att det finns saker att jobba på för att göra användargränssnitten bättre. Agent är i hans mening den enda av CallGuide-applikationerna som är riktigt bra då den är både genomtänkt och användarvänlig.

Vid sidan av CallGuide arbetar systemadministratören mestadels med Teleopti, Officepaketet, en terminalemulator och lite olika webbapplikationer. Callguide Admin används mest för att ändra tidsramarna i styrningen, att omdirigera samtal och ändra eskaleringsvillkor, och användaren menar att 99% av den tid han tillbringar med att

arbeta i Admin läggs just på styrningen av kontakter. Att uppdatera användarkontona är också något som behöver göras lite mer rutinmässigt och dessa båda funktioner använder han allra mest. Andra funktioner nyttjar han i princip en eller någon gång om året.

Teamledarens huvudsakliga arbetsuppgift är att ha hand om den dagliga driften i kundtjänsten. Han arbetar därför mest med schemaläggningsprogrammet Teleopti men tycker inte att de bitar av CallGuide som han använder är särskilt svåra att förstå sig på. I Admin är hans arbete i princip begränsat till fönstret Användarkonton. Han ser till att hålla listan uppdaterad genom att lägga upp och ta bort konton, och ser inga svårigheter med detta. Det händer att han även använder sig av funktionen att lägga in inloggningsmeddelande. Däremot mejlar eller pratar han oftast direkt med folk istället för att använda funktionen Skicka meddelande. När systemet var nytt använde han däremot även den funktionen mer.

5.2.1.4 Nationell trafikledningscentral för kundtjänst

Det här företaget har en liten grupp Admin-användare som sköter all administrering av handläggare samtidigt som de konfigurerar styrning av kontakter och övervakar de köer och väntelistor som finns i systemet. Enskilda intervjuer har gjorts med två av personerna i gruppen. De har båda jobbat på företaget sedan slutet på 90-talet och började sin bana som handläggare i kundtjänsten. Företaget har arbetat med CallGuide sedan år 2000 och har just nu version 5.1 så de har båda arbetat med många olika upplagor av systemet under resans gång.

Trafikledarna har ansvar för att olika arbetsuppgifter blir utförda men *vad* som faktiskt ska göras bestämmer kundtjänstledarna ute verksamheten. Om det exempelvis är en ny grupp av användare som ska använda systemet så ska det läggas upp nya handläggare, anknytningar, kompetenser, köer och styrrader så att allt fungerar.
Gruppen använder själva CallGuide Agent för att svara i telefon och CallGuide Pulse för att övervaka köer och agenter. Om användarna ute i kundtjänsten har problem med CallGuide Stat kan det hända att trafikledarna får hjälpa till även med det. E-post-program, Internet, MSN, Word och Excel är andra hjälpmedel som de ofta använder, men mest är det CallGuide och Admin som gäller.

För den ena av användarna innebär huvuddelen av arbetet att administrera användare och anknytningar. Utöver anknytningar menar han att det finns mycket under Media-menyn som han och hans kollegor aldrig använder eller är inne och kollar på. E-brev använder företaget sig till exempel i princip inte av över huvud taget.

Den andra intervjuade trafikledaren jobbar i första hand med viktning och prioritering av köerna för att få dem att flyta på så bra som möjligt. Det verktyg han använder allra mest i Admin är således fönstret Köer och väntelistor. Direktiven om vilka köer som är viktigast och alltså ska prioriteras i första hand kommer uppifrån. Oftast är det volymköerna, dvs. de köer som det kommer in mest trafik på som ska prioriteras. De flesta köer har dessutom varsin priokö som används när kunderna gjort ett felaktigt val och kommit till en avdelning där de inte kan få hjälp och därför måste bli omkopplade. Gruppen har sammanställt en lathund i Word för grundkonfiguration av viktning och prioritet. Under dagen ändrar de lite varstans och korrigerar sedan tabellen i enlighet med lathunden innan de går hem på kvällen så att det ska se rätt ut

på morgonen. Vid sidan av Admin har de som ett av sina primära hjälpmedel ett egenutvecklat webbaserat stödsystem som ger en översiktlig bild av de största köerna.

En annan viktig arbetsuppgift är övervakning av handläggarna där trafikledarna kollar så att det inloggade antalet agenter stämmer överens med schemat – i första hand om någon kö blir ovanligt lång. Trafikledarna agerar helt enkelt trafikpoliser. Idag vet alla handläggare att de är övervakade och de måste följa sina scheman. De får bonus efter hur väl de följer sitt schema.

5.2.1.5 Statlig myndighet med bredd

Myndigheten har under hösten genomfört en omstrukturering av personalen i lokalerna och eftersom systemförvaltaren numera delar arbetsrum med en kollega genomförs intervjun i ett litet sammanträdesrum där det finns möjlighet att koppla upp sig mot alla olika stödsystem. Myndigheten har kört Telia CallGuide VCC sedan den första oktober 2004. Tre veckor innan intervjun uppgraderades systemet till version 5.2.2.9. Traditionellt har myndigheten varit indelat i ett antal olika regioner med varsin chef och därför fanns det från början ett stort motstånd inom organisationen till att införa ett gemensamt telefonisystem av CallGuides typ. Drygt två år efter systembytet har man dock lyckats samordna alla enheter under ett och samma tak.

Systemförvaltarens huvudsakliga arbetsuppgifter i Admin är att ladda nya fraser och styra samtal. Eftersom myndigheten har funktionsavtal på CallGuide så behöver hon aldrig lägga upp några delområden, kompetenser eller köer på egen hand. För att få ett system som speglar verksamheten samarbetar hon däremot mycket med teknikerna på Telia när standardkonfigurationen skall utformas. Verksamheten bestämmer sedan när förändringar och tillägg i systemet behöver genomföras. I de fall då ändringar ska göras menar systemförvaltaren att CallGuide Pulse är ett viktigt komplement till Admin. Det är det enda sättet att få återkoppling och i realtid se följderna av det man ändrat och hon tycker inte att man kan använda Admin fullt ut om man inte har Pulse.

Ute i verksamheten finns också ett tiotal Admin-användare, vilka med sin personkännedom har ansvar för att se till att agenterna ligger under rätt delområde och har rätt kompetenser. Dessa personer har behörighet att administrera användare, anknytningar och kompetenser och kan även *se* t.ex. informationen för samtalsstyrning men har inte behörighet att *modifiera* den. Systemförvaltaren menar att ”om man ska ha ett helhetsansvar för systemet kan man inte låta vem som helst ändra i det.”

5.2.2 Generella kommentarer

Med generella kommentarer avses användarnas helhetsintryck av CallGuide Admin och deras totala upplevelse av systemet. Många användare uttrycker en allmän belåtenhet med systemet och menar att de är nöjda med CallGuide på det stora hela. De generella synpunkter som användarna trots allt har kan grovt sett grupperas i tre olika områden, nämligen navigering och struktur, handhavande samt kommunikation.

5.2.2.1 Navigering och struktur

Vad gäller systemets informationsstruktur så menar en av användarna att gränssnittet i Admin är onödigt kompakt. När man är inne i Admin sitter man enligt honom och administrerar och det gör absolut ingenting om man tar upp hela skärmytan för detta ändamål – tvärtom. En annan användare hade synpunkter på den hierarkiska meny-

strukturen och den mängd av olika fönster som klienten är uppbyggd av. Han menade att det känns som om det alltid krävs ytterligare ett steg, ett besök i ännu ett fönster, för att utföra en önskad handling.

När det kommer till navigeringen mellan systemets olika delar så hade en av användarna synpunkter på att man alltid kommer tillbaka till huvudmenyn när man stängt ner ett fönster. Någon annan ansåg att det ofta blir mycket klickande fram och tillbaka mellan olika menyer och att det saknas en naturlig och logisk ordningsföljd mellan olika systemdelar.

Det faktum att man egentligen inte behöver gå in i alla delar av klientgränssnittet för att upprätta ett fullt fungerande CallGuide-system var något som kommenterades av ett par olika användare. De upplever att många av menyvalen leder till funktioner som är överflödiga för deras egen del. En av användarna menade att man snarast ”blir förvirrad av en massa menyval som man ändå inte har behov av och aldrig kommer att använda. Det finns ju faktiskt ställen där man kan ställa till rejält med skada om man inte vet vad man håller på med.”

5.2.2.2 Handhavande

Användarna upplever även att vissa element i systembeteendet gör deras arbete mer omständligt och energikrävande än det skulle behöva vara. Ett exempel på detta är att systemet idag inte sparar de fönsterinställningar som användaren gör, utan det mesta måste göras om igen när fönstret öppnas på nytt. Från användarhåll kommer därför önskemålet att inställningar såsom fönsterstorlek, kolumnbredder, sorteringsordning och liknande skall sparas, åtminstone under en och samma session. Därtill menar ett par användare att CallGuide över lag skulle kunna anpassas bättre till Windows-standarder och i synnerhet vanliga kommandon som att öppna med Ctrl+O, dubbelklicka för att göra ett val eller markera flera val samtidigt genom att hålla inne Ctrl-knappen.

Det som generellt sett upplevs som ett av de mest besvärliga handhavandeproblemen är dock systemets skrollningsfunktion, vilken kommenteras av ett flertal användare och upplevs vara bristfällig genomgående i hela klienten. En av användarna berättade att han och hans kollegor till och med blivit ålagda att över huvud taget inte använda skrollern eftersom ”man då kan ändra på sådant som man inte *ska* ändra på.”

5.2.2.3 Kommunikation

Enligt ett par olika användare finns det många olika ställen i systemet där saker och ting inte riktigt fungerar så som man förväntar sig. De tycker att systemet ger dålig hjälp och vägledning till användarna och den information som ges kan vara svår att tyda. Många gånger får man gissa sig till vad som händer. Någon menar till exempel att det bland annat saknas möjlighet att avbryta på flera ställen i CallGuide och att det inte alltid är självklart vad det innebär att klicka okej. Väljer man att då att istället stänga fönstret med hjälp av krysset uppe i fönstrets högra hörn så vet man inte om det är detsamma som att klicka okej eller som det hade varit att klicka avbryt om en sådan knapp funnits.

Vidare menar ett flertal användare att det genomgående i CallGuide saknas möjlighet att få hjälp, både i fråga om isolerade händelser och mer omfattande operationer som kräver en sekvens av olika handlingar. De tycker att systemet borde bli bättre på att

hjälpa dem som användare att göra rätt från början. Från användarhåll vill man därför framhålla vikten av att få mer stöd av systemet i form av fler stödfunktioner, bättre instruktioner och wizards så att man slipper vända sig till manualen eller gissa.

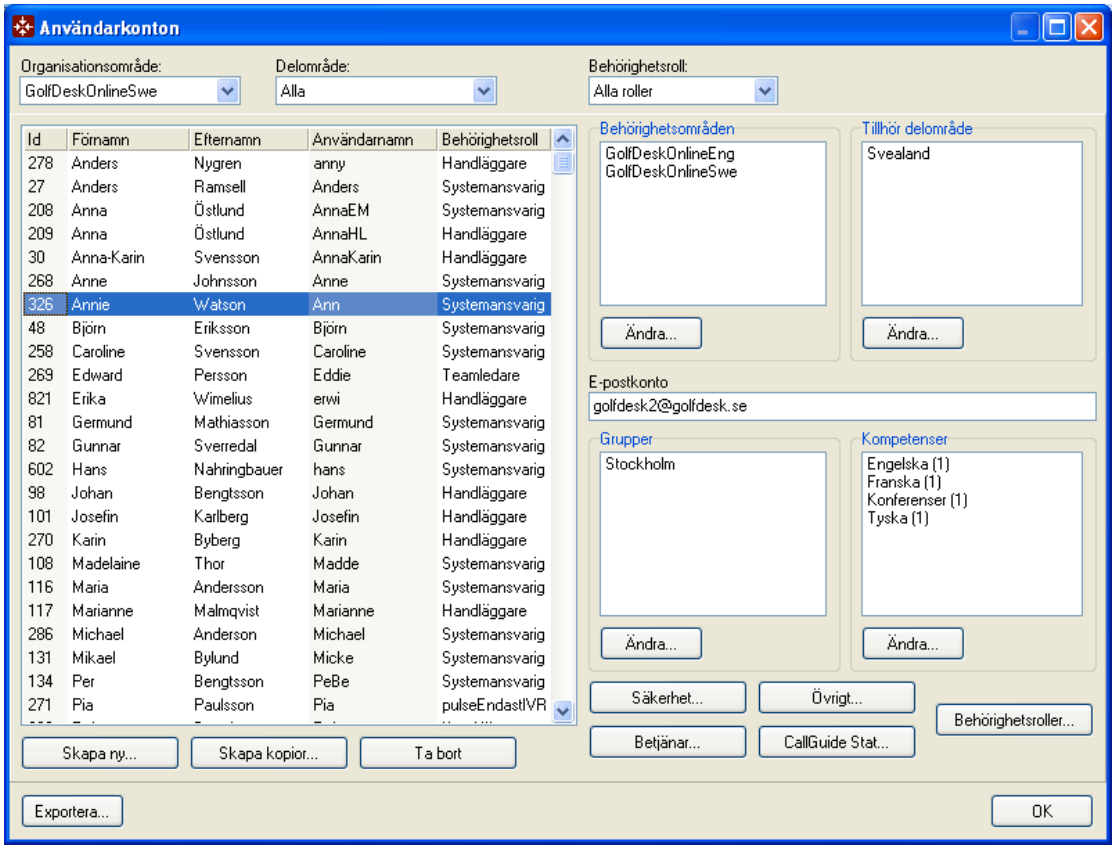
En av användarna uttrycker en känsla av att systemet är väldigt teknikerinriktat och inte så mycket inriktat på användarna. ”De som gör systemen är oftast inte användare själva, utan jätteduktiga tekniker som visserligen kan programmera väldigt bra, men som inte har användargränssnittet som sin högsta prioritet.” Han menar vidare att man skulle behöva vara expert på Admin för att verkligen förstå sig på alla delar, men att man har så mycket annat att göra också att man inte *hinner* bli expert.

5.2.3 Fönsterspecifika synpunkter

Användarna hade också mycket åsikter om just de delar av gränssnittet som de använde sig mest av i sitt arbete. Eftersom det dagliga arbetet ofta är koncentrerat till ett mindre antal fönster kom dessa mer konkreta användarsynpunkter att fokuseras till ett mindre antal dialoger, närmare bestämt tolv stycken. Dessa tolv dialoger presenteras nedan i varsitt avsnitt. Avsnitten är ordnade efter huvudmenyn i Admin (se exempelvis figur 14 på sid.63 för en komplett uppställning av samtliga menyval).

5.2.3.1 Användare | Användarkonton

Det här är en central del av systemet som alla Admin-användare är välbekanta med. Det hindrar däremot inte att de arbetar på olika sätt och har olika åsikter om hur pass väl den här dialogen fungerar.



Figur 4: Dialogen ”Användarkonton” i CallGuide Admin. Här administreras användare, deras behörighetsområden, delområden, grupper och kompetenser.

Överlag tycker användarna att den här delen av applikationen är lätt att använda och inte särskilt svår att förstå sig på. De tycker bland annat att det är smidigt och enkelt att man kan lägga upp nya användare genom att skapa kopior av befintliga användare och på så sätt återanvända alla inställningar och bara ändra namn och användarnamn. Det är också mycket uppskattat att man kan se vilka köer en användare betjänar. En användare tycker dock att hela användarfönstret borde förenklas, att det är för rörigt som det är idag. Någon tycker att det besvärligt att det inte finns något sätt att avbryta när man märker att man gjort fel, medan en tredje skulle vilja att det fanns möjlighet att göra ett mer specificerat urval över vilka användare man vill få fram i tabellen. Det påpekas också att Användarkonton är ett av de fönster där man kan ställa till riktig oreda om man använder skrollern samtidigt som en flervalsruta är vald i tabellen.

5.2.3.1.1 Ändra

Användarnas uppfattning är att det är lite för bökigt att ändra behörighetsområden, delområden, kompetenser och grupper. För att genomföra en ändring måste man klicka på den ändra-knapp som är associerad med respektive funktion. När man gör detta öppnas ett nytt fönster och en av användarna tycker att det är svårt att få överblick över informationen i dessa fönster. Samtliga fyra fönster är uppbyggda på samma sätt. Till vänster finns en ruta med tillgängliga alternativ, till höger en ruta med valda alternativ och däremellan en vänster- och en högerriktad pil. För att göra en förändring måste man markera en rad i någon av rutorna och ”klicka över” informationen till det andra fönstret via en av pilarna.

Det går visserligen att välja fler alternativ åt gången genom att hålla nere kontrollknappen (Ctrl) och markera fler rader, men det finns användare som anser att det även borde gå att dubbelklicka för att göra sitt val. ”Det här är så typiskt: utvecklare lägger in fem användare, men ska man hålla på att lägga in 200 då är det inte kul, då är varje klick värdefullt för handen.”

5.2.3.1.2 Export

Möjligheten att i det här fönstret även exportera tabellen med alla användare i Call-Guide-systemet till en excelfil uppskattas av användarna och de tycker att det är en väldigt bra funktion. Eftersom den samlade informationen endast finns att tillgå genom Admin är det bra att man kan få ut en lista så att lokala chefer också kan få ta del av dem.

En av användarna tycker däremot att det konstigt att man inte får ut allt man vill när man exporterar. Det han saknar är att få med vilka delområden och grupper användarna tillhör och vilka kompetenser de har. Om man ska kunna använda den exporterade tabellen som en back-up fullt ut så *måste* man enligt honom ha den informationen också. I nuläget går det inte att få ut den informationen ur systemet på något enda sätt, utan vill man veta så får man sitta med papper och penna och skriva upp det. Ibland händer det till exempel att agenter ute i kundtjänsten tycker att de får många fler samtal än någon annan och då vill man ha tillgång till kompetenserna i ett jämförbart format för att kunna kontrollera om det finns några skillnader mellan två givna personer.

5.2.3.1.3 Sökfunktion

Vad gäller möjligheten att söka reda på en vald användare så skiljer sig användarnas åsikter åt en hel del. Någon tycker att det fungerar utan någon specifik funktion eftersom man kan sortera den kolumn man vill söka i och sedan leta sig fram till rätt användare med hjälp av skrollern på musen eller rullningslisten i tabellen. En annan användare som administrerar en stor organisation med över 10 organisationsområden och drygt tvåtusen användarnamn tycker dock att det är en stor nackdel att det inte finns någon sökfunktion i det här fönstret. En medarbetare som ringer in från något lokalkontor har för det mesta inte någon aning om vilket organisationsområde han eller hon tillhör. Eftersom systemet inte tillhandahåller information för fler än ett organisationsområde åt gången blir det ofta ett rätt omständligt arbete att hitta fram till rätt rad i rätt tabell. För att underlätta sitt eget arbete har just den här administratören därför skapat ett excelblad över samtliga handläggare så att han bara behöver söka på ett ställe för att hitta rätt organisationsområde innan han går in i Admin för att exempelvis göra begärda ändringar.

5.2.3.1.4 Ta bort

En användare som nyligen uppgraderat Admin ansåg att ta-bort-funktionen blivit mycket bättre eftersom man numera får en motfråga där man ombeds bekräfta att man verkligen vill ta bort den aktuella användaren. Flera användare med henne tycker däremot att det är störande att en användare som man väljer att ta bort försvinner ur listan *innan* man faktiskt bekräftat detta. Användaren skall inte plockas bort ur listan förrän man klickat ja i bekräftelserutan, annars blir man förvirrad. Arbetet försvåras ytterligare av att motfrågan innehåller användarnamnet istället för att ge information om användarens egennamn i form av för- och efternamn. Som administratör har man sällan koll på vilket användarnamn en viss person har och eftersom informationen för den användare man valt att ta bort inte längre syns i tabellen skapar detta en osäkerhet om man verkligen håller på att radera rätt användare i listan.

5.2.3.2 Användare | Användare med kompetenser och grupper

Ett par av Admin-användarna som ofta administrerar stora grupper av agenter tycker att det här fönstret är ett bra verktyg eftersom det ger möjlighet både att ta reda på exakt vilka handläggare som har en viss kompetens och att ge nya kompetenser till en hel grupp av handläggare på en och samma gång. Innan den här möjligheten fanns var det enda alternativet att gå in under Användarkonton och lägga till den nya kompetensen per handläggare, men detta tillvägagångssätt är mycket smidigare. Det är visserligen oftare man får ändring beställd på en person, men i de fall man får långa listor så är detta mycket användbart.

En användare som inte arbetar just i den här delen av systemet mer än någon gång i månaden uttrycker också sin uppskattning och menar att även om han inte använder denna funktion så ofta, är det ändå tillräckligt ofta för att han ska uppskatta det. ” Sånt här är både smart och lätt att använda.”

5.2.3.3 Användare | Skicka meddelande

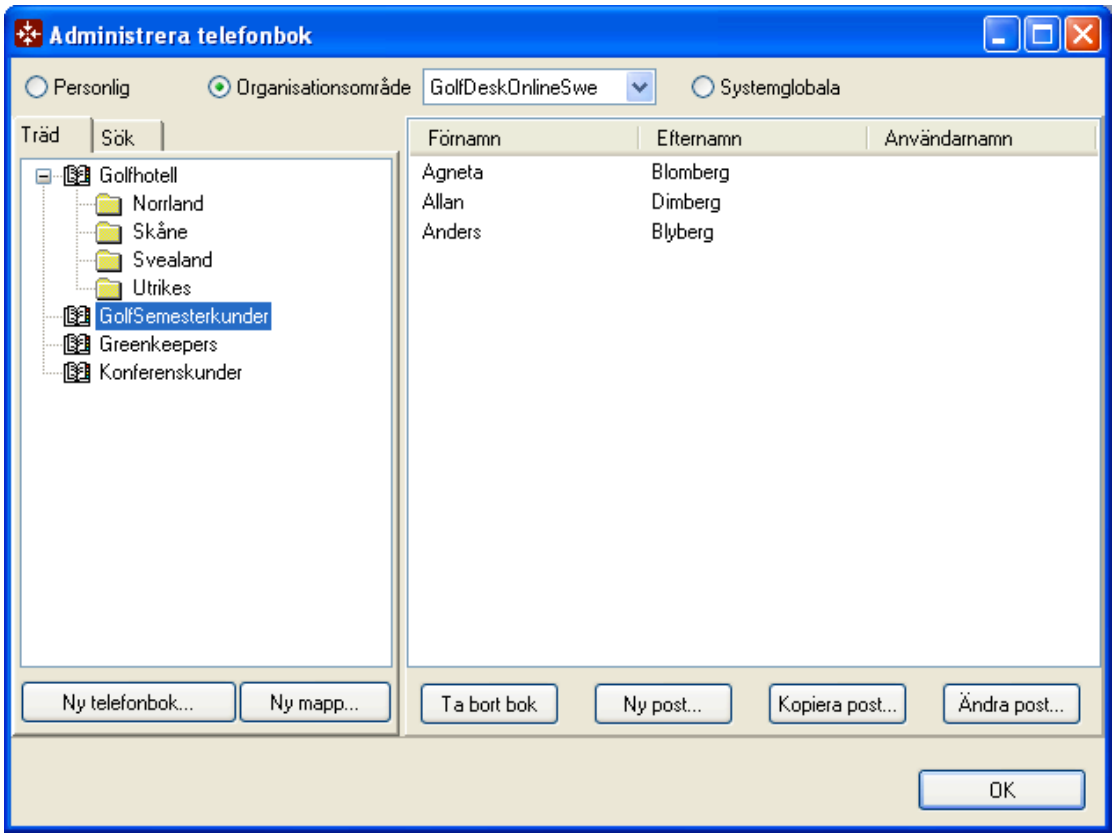
Att kunna skicka ut meddelanden till alla användare i CallGuide-systemet är en funktion som uppskattas av många Admin-användare. De tycker bland annat att det är bra att man kan välja mellan så många olika systemnivåer. I en organisation används den till exempel om det är ett stort fel någonstans eftersom det är smidigare att skicka

ut meddelandet via systemet istället för att mejla alla berörda. Meddelandet kommer upp direkt på skärmen, alla som är inloggade får informationen, och de slipper i sin tur att ringa och fråga vad som är fel.

För en av användarna utgör arbetet med att skicka ut meddelanden en viktig del av hennes samlade arbetsuppgifter, medan ett par andra användare menar att de inte tänker på att använda funktionen trots att den är bra. När systemet var nytt använde däremot även de den funktionen mer. Idag mejlar eller pratar de oftast direkt med folk istället för att skicka ett meddelande.

Skillnaden mellan Skicka meddelande och den närbesläktade funktionen Inloggnings-meddelande är att ett vanligt meddelande endast ses av de användare som är inloggade just när det sänds, medan ett inloggningsmeddelande visas varje gång användaren loggar in i systemet. Ett önskemål som framförts är möjligheten att kunna skicka ett engångsmeddelande som är en *kombination* av inloggningsmeddelande och vanligt meddelande så att de som inte är inloggade vid utskicket får meddelandet när de loggar in och de som redan är inloggade får det omedelbart och dessutom att alla bara får det *en* gång.

5.2.3.4 Användare | Telefonbok



Figur 5: Dialogen "Administrera telefonbok" i CallGuide Admin. Här kan man lägga till och ta bort böcker, mappar och poster i telefonboken, samt kopiera och modifiera befintliga poster.

Telefonboken är en uppskattad funktion och användarna tycker över lag att själva strukturen både är smidig, lättarbetad och enkel att förstå. Så som informationen i telefonboken är organiserad idag tycker dock ett flertal användare att telefonboken i grunden inte att är anpassad till stora företag där det många gånger är vanligare att man ringer andra företag och företagsväxlar istället för enskilda personer.

Den information som visas i den samlade tabellen är nämligen förnamn, efternamn, och användarnamn. Tre av de fem besökta kundorganisationerna har därför gjort egna work-arounds för att systemet ska passa deras behov. En användare berättade att de lagt in telefonnummer i fältet för användarnamn, medan de båda andra organisationerna valt att använda förnamnskolumnen för funktion, skriver in öppettider i kolumnen för efternamn och lämnar användarnamnskolumnen helt tom. Användarna önskar med andra ord att den information som visas i telefonboken skulle vara mer flexibel och gå att anpassa till den egna organisationens behov. Den främsta anledningen till detta är att agenterna ute i kundtjänsterna behöver få en överblick över rätt typ av information för att snabbt kunna koppla vidare samtal som hamnat fel. Den information som efterfrågas är framför allt funktion/ort, öppettider och telefonnummer.

5.2.3.5 Användare | Information Guide

Information Guide är en stödfunktion i CallGuide Agent, en hjälp för handläggarna där de kan slå upp sådant de undrar över under ett pågående samtal. Funktionen anpassas helt och hållet efter kundens egen verksamhet i och med att man efter hand fyller den med matnyttig information. Konfigureringen sköts helt och hållet via Admin och åsikterna går isär

En av användarna tycker att funktionen är ganska enkel att administrera. Hans enda egentliga synpunkt är att man inte kan koppla ett befintligt dokument till posten i guiden. En annan användare tycker dock att Information Guide hör till den del av systemet som är svårast att förstå sig på. Enligt honom lämnar användargränssnittet i det här fallet mycket kvar att önska. Ett exempel på detta är att man i någon konfigurationsfas måste ha två Admin-klienter öppna samtidigt eftersom man behöver informationen under Media | E-post | Metoder för kategorisering för jämförelse.

5.2.3.6 Media | Telefoni | Anknytningar

Till att börja med frågar sig ett antal användare varför anknytningarna aldrig är sorterade när man öppnar fönstret. Om man letar efter något speciellt måste man alltid sortera först. Det här fönstret går man antingen in i för att se om en anknytning finns inlagd, eller för att lägga till eller ta bort en anknytning och det gäller att hålla reda på vilket organisationsområde man ska arbeta med.

5.2.3.6.1 Lägg till

När man har skrivit in en ny anknytning räcker det inte med att klicka ok för att den ska sparas i systemet, utan man måste *först* klicka ”Lägg till”. En av användarna berättar att han gjort fel i det här fönstret hur många gånger som helst och att det är mycket irriterande när man upptäcker att den anknytning man just lade till trots allt inte fungerar. Han kan se att det är befogat i de fall då man vill lägga till fler anknytningar innan man stänger fönstret men menar ändå att det är inkonsekvent i relation till andra delar av systemet. I de båda dialogerna Kontaktdata och Styrning av kontakter lägger man till exempel till ett nytt tabellelement genom att mata in information direkt i den tomma raden nederst i tabellen, medan man i fönstret Köer och väntelistor öppnar en ny dialogruta då man klickar ”Lägg till” och där blir guidad genom ett antal olika steg för att lägga till en ny kö. Om detta åtminstone skedde på samma sätt skulle man lära sig och det skulle bli intuitivt. ”Det går ju an när man har

lärt sig – allt är ju enkelt när man kan det – men det ska ju vara lätt att lära sig också.” En annan användare poängterar att man som användare dessutom inte får någon feedback då man klickar ”Lägg till”. Systemet reagerar till synes inte alls, de inmatade siffrorna står kvar i rutorna och man får ingen bekräftelse via tabellen heller. Man får helt enkelt sortera och söka i tabellen för att se om den nya anknytningen verkligen *har* lagts till.

5.2.3.6.2 Ta bort och export

Här stöter man på samma problem som för Användarkonton, nämligen att den markerade raden försvinner från tabellen innan användaren bekräftat att detta faktiskt var ett medvetet val och att raden *ska* tas bort. En av användarna är särskilt brydd över detta och menar att det är ett generellt feltänk i hela systemet. I och med att markeringen flyttas till nästa post i tabellen blir man enligt henne dessutom osäker och frågar sig om det är den rad som är markerad i tabellen som man håller på att ta bort. I den organisation hon arbetar för går man med jämna mellanrum igenom alla anknytningar och gallrar bort dem som blivit inaktuella och hon menar att man alltid undrar om man verkligen fick med dem man skulle eller inte.

I och med den här osäkerhetsfaktorn som hon upplever så tycker samma användare att det är mycket positivt att man kan exportera ut den här tabellen till ett excelark. På så sätt kan man hålla reda på hur det såg ut innan man började gallra bland anknytningarna, något hon själv också alltid gör.

Även om man har klickat ja i bekräftelserutan för ”Ta bort” så upplever användaren att man ändå inte kan vara säker på att det verkligen går igenom om man stänger på krysset. Man måste öppna fönstret en gång till och dubbelkolla att det verkligen är borta, vilket leder till en massa klickande och kollande hit och dit.

5.2.3.7 Media | Telefoni | Platser och accesskoder

Det här är en del av systemet där användande skiljer sig markant beroende på organisationernas respektive verksamhet. En av användarna berättar till exempel att han inte kan förstå varför det finns en snabbvalsknapp för att komma till det här fönstret eftersom han själv aldrig har behov av att använda det. En annan menar på att detta är ganska krångligt att administrera och att de oftast inte gör just den här typen av konfigurering själva, utan låter leverantören sköta den biten. För en tredje användare är det här däremot en så viktig funktion i hans eget arbete att det är det första han tar upp under intervjun.

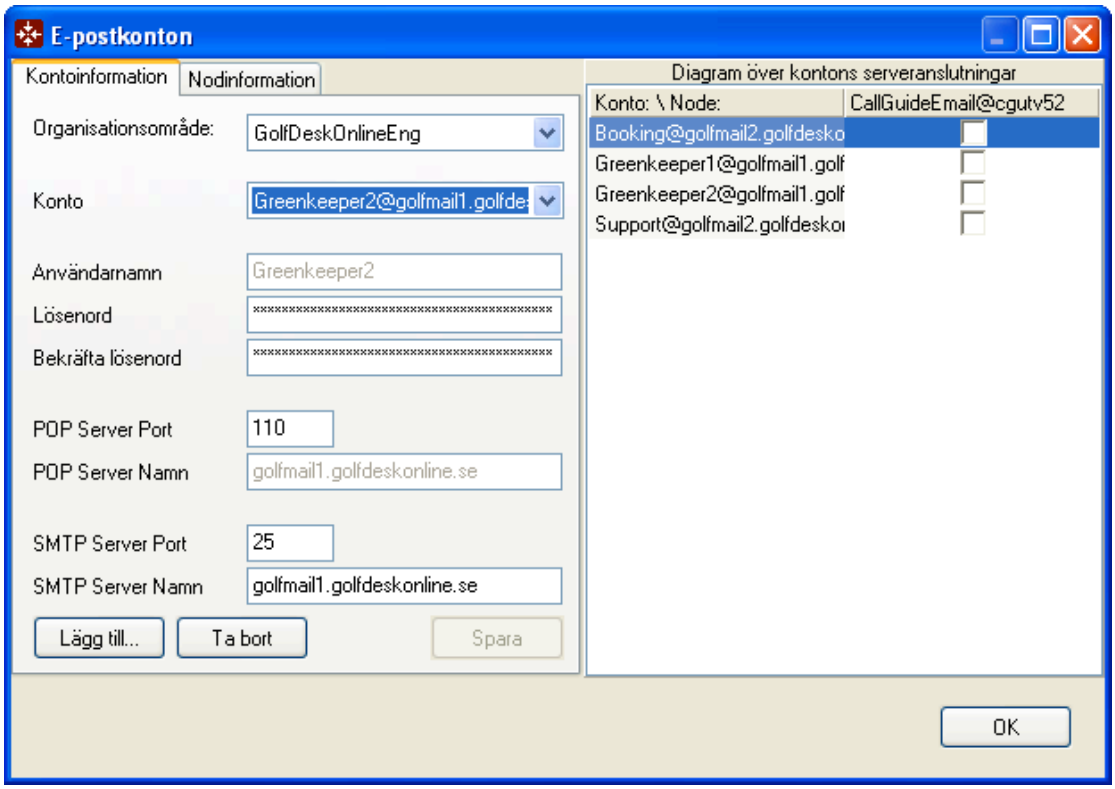
Han berättar bland annat att det inte är helt lätt att lägga till en ny plats i systemet. När man har matat in de uppgifter som är aktuella och klickat ”Lägg till” så blir man automatiskt förflyttad till fliken *Accesskoder, anknytningsplatser* där den nya platsen skall associeras samman med alla andra platser i systemet så att de vet hur de ska nå den nya platsen och vice versa. Rader för detta skapas automatiskt och är rödmarkerade, men de sorteras in utspritt i tabellen. De skapas dessutom med ett initialvärde som måste ändras manuellt för varje enskild rad och ett prefix för varje rad måste också matas in. Detta blir enligt användaren väldigt omständligt och meckigt om systemet är stort och har många olika platser som i hans fall.

För att underlätta arbetet skulle han tycka att det vore toppen att få tillgång till exportfunktionen även i dessa fönster. Om man kunde exportera tabellen så skulle

man kunna se på en gång om något är fel. Han skulle också vilja att man som en säkerhetsåtgärd får en motfråga när man stänger på krysset i de fall då man gjort ändringar.

5.2.3.8 Media | E-post | E-postkonton

Den här dialogen upplevs som lite svår och användarna uttrycker en rädsla för att göra fel. En användare berättar att han väldigt sällan är inne och ändrar något i det här fönstret och de gånger det hänt har det alltid trasslat till sig så att personal på Telia har fått lov att gå in och rätta till saker och ting.



Figur 6: Dialogen "E-postkonton" i CallGuide Admin. Här administreras de e-postkonton som är kopplade till CallGuide-systemet.

Användarna tycker att grundinställningarna i fönstret är satta så att mycket av informationen inte visas på ett överskådligt sätt. Flervalsrutan för kontoinformation går till exempel inte att göra större, vilket medför att man inte ser allt som står i den. Den borde vara större från första början, eller vara möjlig att anpassa efter informationsinnehållet. En annan synpunkt är att markeringen borde överensstämma mellan de båda rutorna i fönstret, att det är det enda logiska beteendet. Det konto som väljs i flervalsrutan borde markeras i den intilliggande tabellen och tvärtom.

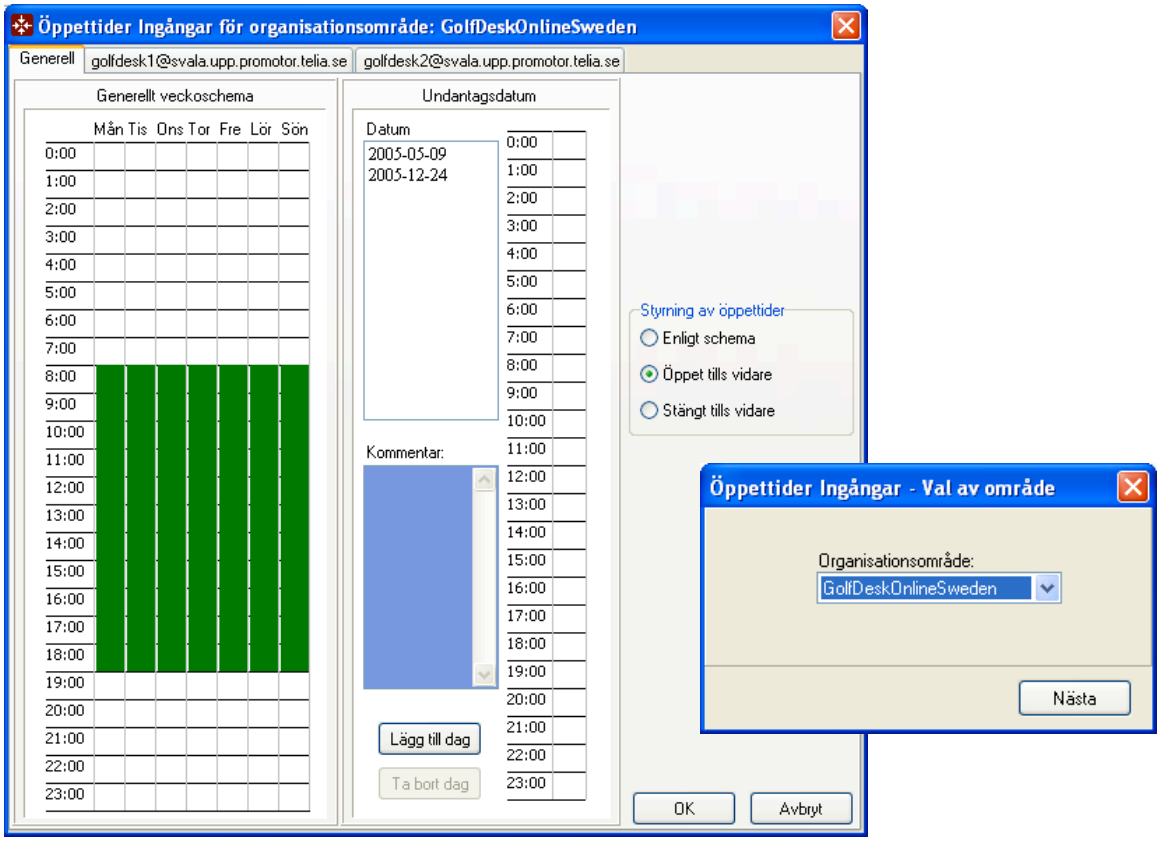
5.2.3.9 Styrning | Öppettider

Användarna tycker generellt att det är enkelt att justera öppettiderna och att man kan välja att styra antingen på ingångar eller på menyval. Däremot måste man själv hålla reda på om öppettiderna är inlagda på ingångar eller menyval då det inte går att få fram ingångar och menyval per kompetens eller grupp ur systemet. I större organisationer kan vissa grupper ha det ena, medan andra grupper har det andra, det beror lite på. Andra kommentarer av mer övergripande karaktär är att man tycker det är smidigt

att tidslisten är klickbar och mycket bra att flikarna numera är sorterade i bokstavsordning. Däremot skulle användarna vilja kunna maximera fönstret till helskärm.

En av användarna har synpunkter på arbetsgången då man lägger till ett nytt undantagsdatum. Enligt honom är det logiskt och naturligt att omedelbart efter att man valt datum även specificera öppettiden för den aktuella dagen. Som det är nu måste man själv markera den nyss tillagda dagen för att göra detta – den borde istället markeras automatiskt. En annan användare tycker också att det skulle vara toppen om man kunde kopiera inställningarna för ett undantagsdatum till ett annat.

När man sedan ska ta bort undantagsdatum ur listan så går det inte att markera fler än ett datum per gång. Det uppleves av användarna som frustrerande och inkonsekvent i jämförelse med andra delar av systemet som exempelvis dialogen för att ändra kompetenser, där detta går hur bra som helst. En av användarna menar att man tydligt känner att det är olika personer som har programmerat olika delar av systemet.



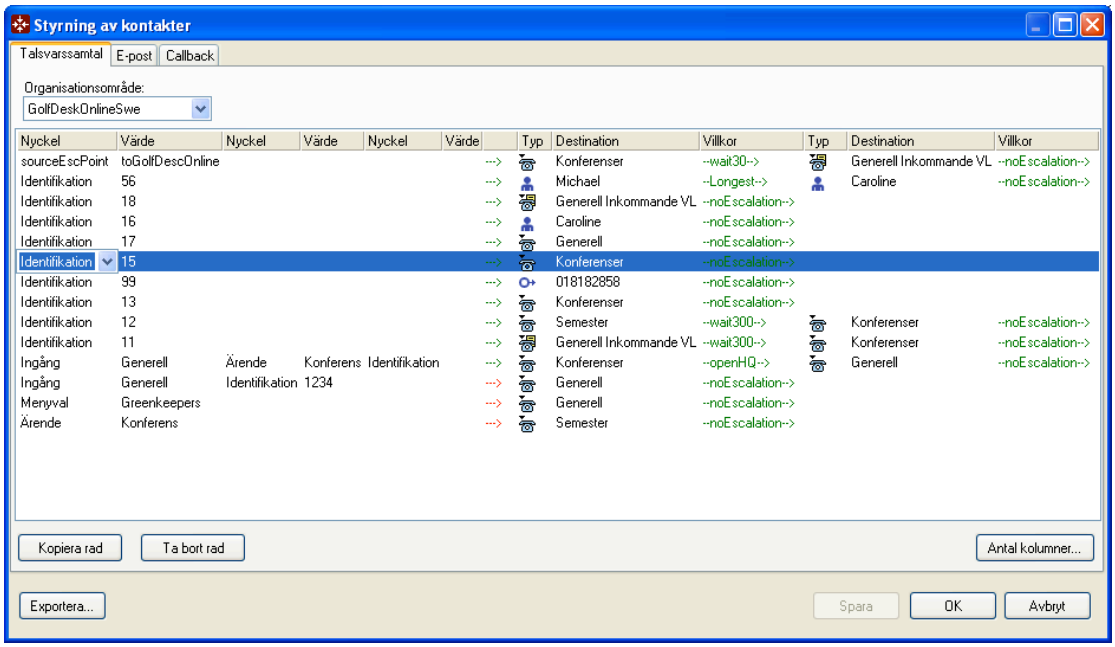
Figur 7: Dialogen "Öppettider" i CallGuide Admin. I det här fönstret administreras öppettiderna i talsvaret via ett generellt veckoschema och undantagsdatum för dagar som har avvikande öppettider, exempelvis helgdagar. Dialogen ser likadan ut för menyval. Det lilla fönstret till höger måste passeras innan användaren får fram den större dialogen.

Användarna har också förslag på förändringar som skulle kunna underlätta deras arbete avsevärt. Från en stor organisation med många användare och organisationsområden kommer bland annat önskemålet om en möjlighet att lägga till undantagsdatum på mer övergripande nivåer än bara för ingångar och menyval. När rörliga helgdagar som exempelvis nyårsafton infaller på en vardag måste man i dagsläget gå in på varje ingång och vartenda menyval för att lägga till detta, vilket kan ta mellan en halv till en hel arbetsdag.

5.2.3.10 Styrning | Styrning av kontakter

Styrningen är en av de mest centrala delarna av ett CallGudie-system och en användare berättar att man på grund av känsligheten exporterar styrraderna en gång i veckan och sparar dem i ett par veckor för att kunna gå tillbaks och ställa saker och ting till rätta ifall nåt råkar gå snett av misstag. Enligt användarna är det dessutom relativt lätt hänt att man av misstag råkar förstöra saker och ting genom att använda rullmusen där man inte borde. En av administratörerna har till och med förbjudit alla andra Admin-användare att gå in i styrningen eftersom han märkt att folk omedvetet ändrat i tabellen, vilket ställer till stora problem. Samma användare tycker dessutom att både fönstret i sig och informationen är svår att överblicka, vilket gör det svår-arbetat. Samtidigt menare en tredje användare att det mesta ändå är ganska självklart på så sätt att det är hyfsat intuitivt vad man ska göra och hur man ska göra det.

Eskaleringen fungerar enligt användarna precis som den ska, men däremot så står allt på engelska och användaren måste själv hålla reda på vad de olika villkoren betyder. Någon tycker att det är ”lite si och så med namngivningen.” En del villkor har genom-tänkta namn som gör det lätt att förstå vad de innebär, medan andra namn inte snarare upplevs som ”hieroglyfer.” En av användarna menar att det kanske inte är så lätt för en användare som aldrig har jobbat i systemet tidigare. För att bli ett bättre verktyg som är enklare att arbeta med skulle man som användare behöva få en beskrivning av vad de olika villkoren innebär. Som det är nu så användes inte alla de olika val som finns tillgängliga, även om det skulle kunna vara lämpligt.



Figur 8: Dialogen ”Styrning av kontakter” i CallGuide Admin. Här konfigureras vilken typ av samtal som skall styras till vilken kö, samt vilka villkor som skall gälla för eskalering av samtal.

En av användarna tycker att det är en stor fördel att man kan exportera hela fönstret till en excelfil. Däremot menar han också att det är en väldig nackdel att man sedan inte kan importera samma fil; att den exporterade tabellen inte går att använda rakt av för att återställa ett önskat tillstånd. Man måste *veta* hur det var eller kontrollera mot ett gammalt excelark. Excel har enligt användaren den fördelen att man lätt kan filtrera och sortera informationen. Det hade i hans mening varit toppen om man hade

kunnat konfigurera detta ark och sedan importera det i CallGuide istället för att mickla direkt i styrraderna.

Vissa organisationer har en verksamhet som till stor del påverkas av olika händelser. En av användarna berättar att han ändrar styrningen av inkommande samtal beroende på vilken säsong det är. På sommaren kommer det in så många samtal att det i princip inte spelar någon roll hur de styrs. Då handlar det mer om att se till att alla kunder får svar. Under lågsäsong styr man däremot mer på kompetens, och har alltså i princip två scenarier; hög- och lågsäsong och ibland även mellansäsong. Enligt honom skulle det vara riktigt bra om det fanns möjlighet att konfigurera upp ett par färdiga alternativ att välja mellan så att man slapp ändra på egen hand varje gång.

5.2.3.11 Styrning | Köer och väntelistor

I det här fönstret lägger man till nya köer och väntelistor och bestämmer vilka kompetenser som ska krävas för att betjäna dem. Om en kö även skall vara synlig i CallGuide Pulse är det här man gör den inställningen. I den här tabellen konfigurerar man även handläggarordning och sätter vilken sorteringsregel som ska gälla per kö. En användare berättar att han arbetar mycket med att ändra prioritet och viktning på köerna beroende på belastningen under arbetsdagen. Den organisation han arbetar för har ett komplext system med många köer och organisationsområden. Eftersom hans arbete innebär övervakning av köer i fler än ett organisationsområde så skulle han önska att man kunde lägga in ett ”Eget val” där man själv väljer vilka köer man vill kunna se i samma vy, oavsett organisationsområde, så att man slipper byta organisationsområde och leta sig fram till rätt köer stup i kvarten. Eftersom tabellen ändras lite varstans i under dagens lopp har man sammanställt en lathund i Word för grundkonfiguration av viktning och prioritet. Innan de går hem för dagen ändrar de tillbaks alla värden i enlighet med denna lista.

5.2.3.12 Organisera | Grupper och kompetenser

Här har det enligt en av användarna hänt något i samband med uppgraderingen. Informationsrutorna till höger skall dels visa vilka användare som har respektive kompetens eller hör till en viss grupp, dels vilka delområden som kompetensen eller gruppen finns representerade i. Men efter uppgraderingen ändras inte informationen om man väljer en ny rad i tabellen. Det blir svårt att se vad man gör och inte gör. I användarens egen mening är det viktigt att det inte blir sådana här fel om man vidareutvecklar verktyget. Det som händer enligt henne är att man bygger in ännu mer osäkerhet i systemet och att man som användare undrar om det är en själv eller systemet det är fel på.

5.2.4 Egna iakttagelser

Innan intervjuerna var jag lite fundersam över om användarnas egna kommentarer egentligen skulle handla så mycket om användbarhetsfrågor eller om de mest skulle kretsa kring vilka funktioner, knappar och knapplaceringar man skulle vilja ha eller ändra på. Det visade sig dock att samtliga användare hade mycket synpunkter i frågor som gäller just klientens användbarhet och intervjumaterialet är både innehållsrikt och tankeväckande.

Det samlade intrycket är att användarna uppskattar CallGuide som system och i synnerhet de stora anpassningsmöjligheter som verktyget tillhandahåller. Samtidigt

känner de osäkerhet och förvirring i samband med sitt eget arbete i Admin. De upplever att klienten är plottrig och svårt att överblicka, att det är svårt att navigera mellan alla fönster, samt att det förekommer mycket inkonsekvens mellan verktygets olika delar. Beroende på verksamhet används dessutom olika delar av Admin i olika stor utsträckning och det finns många delar av systemet som ingen av de intervjuade användarna egentligen har särskilt stor användning för i sitt dagliga arbete. Stora delar av gränssnittet används bara vid större förändringar och åtgärder i systemet som att bygga om och bygga nytt. Den större delen av tiden tillbringar användarna dock med underhåll och drift av systemet men de fönster och dialoger som faktiskt används dagligen är utspridda i hela gränssnittet och det är inte alltid lätt att hitta rätt.

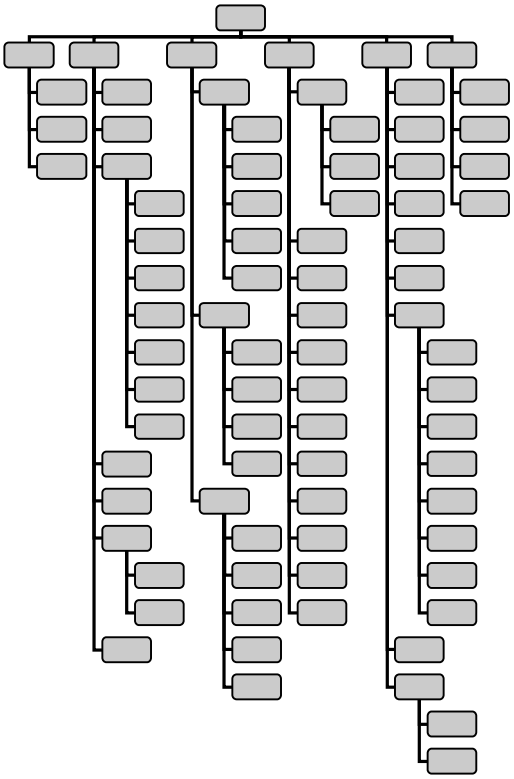
Resultatet är att det *blir* mycket klickande fram och tillbaka för att hitta rätt bland menyerna, för att ändra fönsterinställningar, samt för att dubbelkolla att handlingar såsom att lägga till eller ta bort verkligen fungerat som användaren tänkt sig. En vanlig företeelse är att användaren söker efter en viss funktion men inte är helt säker på under vilken meny den ligger. Han tar sig därmed ner i någon av menyerna och öppnar ett fönster som skulle kunna stämma in på hans önskemål. Om det inte blev rätt på första försöket måste användaren stänga fönstret för att kunna gå vidare och måste då börja om från början eftersom Admin inte ger någon indikation om var i systemet han senast befann sig. Det kan vara svårt för användaren att minnas vilket fönster han nyss öppnade och därför händer det att han väljer samma menyval igen vilket är både tidsödande och frustrerande. Användarna blir också frustrerade över andra saker och har ibland jobbat förbi upplevda problem genom att som i telefonboken mata in den information som de önskar se i de fält som faktiskt visas, eller genom att göra egna listor i program som Word och Excel över sådan information som systemet inte tillhandahåller. Administratörsarbetet verkar dessutom vara mindre styrt av den bakomliggande strukturen med organisations- och delområden och det finns ett behov av att kunna åsidosätta detta för att underlätta arbetsflödet för de mer avancerade användarna.

Något som blev tydligt allteftersom intervjuerna genomfördes är att skrollfunktionen verkligen ställer till mycket problem i kombination med tabellernas flervalsrutor. Detta leder till stor osäkerhet och försiktighet hos användarna när de arbetar i dessa fönster. En användare, som vid ett tillfälle råkade markera en cell i styrningstabellen så att en flervalsmeny öppnades, stängde till och med hela fönstret och öppnade det på nytt för att vara säker på att inte av misstag ändra något som inte skulle ändras.

5.3 Kartläggning av CallGuide Admin

Arbetet med att kartlägga menystrukturen och den underliggande informationsordningen i gränssnittet har resulterat i två olika artefakter. Samtliga fönster i Admin inklusive menyer, noder och aktiva löv men exklusive felmeddelanden, bekräftelse- och informationsrutor finns numera utskrivna och organiserade i en pärm som en fysisk representation av det nuvarande gränssnittet. Kartläggningen som sådan har utgjort en stor del av examensarbetets totala arbetsinsats då pärmen innehåller strax över 150 skärmdumpar. Pärmen och dess innehåll har dock inget egenvärde för läsaren av denna rapport då den i examensarbetet snarare fungerat som ett extra hjälpmedel vid den heuristiska utvärderingen, samt som utgångspunkt i arbetet med att föreslå en ny struktur. Den utgör dock ett värdefullt arbetsmaterial som lämnas till företaget att använda i den fortsatta vidareutvecklingen av Admin.

Även den nuvarande menystrukturen har fångats på papper och representeras av figuren till höger. Totalt sett finns idag 80 olika menyval fördelade på tre nivåer. Huvudmenyn innehåller sex olika alternativ och mellannivån består av sammanlagt 38 menyval varav åtta är noder. På den tredje nivån finns det slutligen totalt 36 menyval.
Menyvalen leder i sin tur till olika platser i systemet och de kan se ut på mycket olika sätt. Ofta har en ny funktion fått ett eget menyval och många platser består därför av ett enda fönster där användaren kan utföra en eller ett par olika handlingar. Admin rymmer också ett par mer utbyggda dialoger som tillhandahåller många olika funktioner samtidigt och även leder användaren vidare till andra dialogsekvenser. Exempel på detta är användarkonton, telefonboken, kampanjer, styrning av kontakter samt köer och väntelistor. Det finns även ett par olika menyval som öppnar samma dialog.



Figur 9: En något komprimerad bild av den befintliga menystrukturen i CallGuide Admin.

Som ett mått på systemets komplexitet har jag tagit fram uppgifter på hur många noder respektive löv som gränssnittet innehåller. Syftet med detta har inte varit att presentera någon ny metod, utan är ett försök att använda tillgänglig information för att skapa en större förståelse av gränssnittet som det ser ut idag.

		Antal menyval	Antal noder	Antal löv	Antal löv direkt under menyval
Arkiv		3	0	1	1
Användare		5	4	24	10
CallGuide Agent användargränssnitt		7	1	9	8
Telefonbok		2	2	15	1
Media					
Telefoni		5	0	15	15
Utgående telefoni		4	3	19	7
E-post		5	5	14	6
Styrning		10	5	19	10
Öppettider		2	2*	4	0
Organisera		7	5	11	2
Statistik		8	2	16	14
InteractionView		2	0	2	2
Inställningar		4	0	10	10
		64**	**27**	**159**	**86**

* Antal noder beror på hur många ingångar och hur många menyval som finns i systemet.
** Det totala antalet menyval är 64 + 14 (menynoder) + 2 (menyval endast för CallGuideVCC).

Tabell 3: Tabell över hur gränssnittet i CallGuide Admin är uppbyggt i termer av noder och löv.

Tabellen ovan är helt enkelt ett försök att följa Beyer & Holtzblatts rekommendation och måla upp en bild av den befintliga systemstrukturen utan att göra någon egentlig reverse UED. Frågor man ställer sig är givetvis om det skulle vara möjligt att minska antalet löv som ligger direkt under ett menyval genom att integrera funktioner i en och samma dialog. På detta sätt skulle användaren inte vara tvungen att stänga ner ett fönster och öppna ett annat för att utföra relaterade uppgifter.

5.4 Heuristisk utvärdering av CallGuide Admin

På grund av systemets komplexitet och det insamlade materialets omfattning har jag valt att presentera resultatet från den heuristiska utvärderingen med utgångspunkt i de 10 designprinciperna och ge konkreta exempel på vilka aspekter av CallGuide Admin som behöver förbättras med avseende på respektive princip.

5.4.1 Synlig systemstatus

Systemet skall alltid hålla användaren informerad om vad som händer genom passande feedback inom rimlig tid.

Vid genomgången av systemet visade det sig att det fanns ett par olika brister i återkopplingen i samband med de vanliga handlingarna ”Lägg till” och ”Ta bort”.
I fönstret Media | Telefoni | anknytningar ges exempelvis ingen feedback på att något verkligen har hänt då användaren klickat ”Lägg till”. Det som faktiskt händer är att de nya uppgifterna lagts till på nedersta raden i tabellen. Markeringen hoppar dock upp till den översta raden och användaren måste själv leta igenom tabellen för att kunna kontrollera att den nya anknytningen verkligen blivit inlagd. Ett liknande exempel kan hämtas från dialogen Organisera | Statistik | Ärendetyper. I detta fall sorteras den tillagda ärendetypen in i tabellen efter bokstavsordning, men eftersom raden inte markeras blir det ändå svårt för användaren att kontrollera resultatet av handlingen.

I dussinet olika dialoger hanteras grundfunktionen ”Ta bort” på följande sätt: När användaren klickar ”Ta bort” försvinner det markerade elementet ur tabellen och därefter kommer en fråga från systemet där användaren ombeds bekräfta borttaget genom att svara ja eller nej. Vid ”nej” läggs elementet tillbaks längst ner i tabellen. Det ges dock ingen feedback till användaren om att tabellelementet faktiskt är tillbakalagt, utan det kan verka som om det är borttaget fastän användaren valt att avbryta handlingen.

För att hjälpa användaren med att få informationen i systemet att stämma överens mellan olika delar så skapas vid ett par tillfällen automatiska handlingar kopplade till någon aktivitet som användaren utfört. Då man lägger till en ny eskaleringspunkt skapas till exempel en ny rad i styrningstabellen och när ett nytt e-postkonto skapas lagras motsvarande information som en ny ingång. Samma sorts åtgärd initieras även då en eskaleringspunkt eller ett e-postkonto tas bort. Det här är däremot något som händer utan användarens vetskap. Den enda gång användaren informeras om dessa automatiska handlingar är då en ny eskaleringspunkt läggs till.

5.4.2 Anpassning mellan system och verklighet

Systemet skall tala användarens språk och utgå från de ord, fraser och koncept som användaren är bekant med och använda sig av konventioner från den riktiga världen

istället för systemorienterade termer. Information skall visas i en naturlig och logisk följd.

Terminologin som används i CallGuide Admin är i de flesta fall klar och redig och har en tydlig koppling till branschen. I vissa fall skiner dock tekniken igenom som under menyn Media | E-post där användaren kan göra inställningar för olika former av automatisk kategorisering av inkommande e-brev. Under menyvalet ”Metoder för kategorisering” ges valet mellan antingen *patternmatching only* eller *CallGuide keyword*. Den svenska versionen innehåller dock även en översättning av pattern-matching i det intilliggande menyvalet ”Mönstermatchning” där också de aktuella inställningarna utförs. Kopplingen mellan dessa båda termer och fönster är idag inte helt uppenbar. Vidare så finns ett par termer som antingen är för vaga, som exempel-vis ”Tillhör” (Styrning | Köer och väntelistor) och ”Övrigt” (Användare | Användar-konton) eller har dålig överensstämmelse med det de används för, så som ”Kundunik” (Organisera | Grundkonfiguration av kontaktdata) och ”Bekräfta kompetensprofil” (Styrning | Köer och väntelistor).

Vad gäller att informationen skall visas i en för användaren naturlig och logisk följd så är dialogerna som följer under menyvalet Styrning | Öppettider ett bra exempel på hur det kan bli när detta brister. Rent systemtekniskt styrs öppettiderna för talsvaret antingen via ingångar eller menyval och denna åtskillnad har även använts som utgångspunkt i gränssnittet. För användaren är det dock mer logiskt att utgå från organisationsområde och därefter förfina urvalet till att omfatta antingen ingångar eller menyval. På grund av programmeringstekniska orsaker innehåller gränssnittet dessutom en passage, en dialog som endast innehåller val av organisationsområde innan fönstret med öppettider visas. Om användaren gjort fel i valet mellan ingångar och menyval krävs minst fem handlingar innan rätt fönster har öppnats.

Det finns ett par fönster till där informationsflödet avviker från det som får anses vara normen i CallGuide Admin. Det är dialogerna Metoder för kategorisering, Väntelistor i CallGuide Agent samt Köer och väntelistor. Dessa undantag eller avvikelser skapar osäkerhet då användaren inte kan lita på sin mentala bild av systemet.

5.4.3 Användarkontroll och användarfrihet

Användare väljer ofta väljer ofta funktioner i systemet av misstag och behöver tydligt markerade vägar som på ett snabbt och enkelt sätt leder användaren tillbaks till utgångsläget. Stöd funktioner för att ångra och göra om.

Som Admin ser ut idag kommer användaren alltid tillbaks till huvudmenyn när ett fönster stängs ner. Det går med andra ord inte att backa endast en nivå för att göra ett nytt val. Det finns vidare många fönster där det inte går att avbryta eller ångra en handling, utan användaren måste i så fall på egen hand försöka återskapa eller radera information som felaktigt tagits bort eller lagts in

5.4.4 Konsekvens och standarder

Användare ska inte behöva fundera över om olika ord, situationer eller handlingar betyder samma sak. Följ gällande plattformsstandard.

Vid en systematisk genomgång av gränssnittet blir det väldigt tydligt att systemet blivit spretigt allt efter hand som nya funktioner tillkommit under årens lopp. Detta märks framför allt på förekomsten av en mängd olika tillvägagångssätt för att utföra grundfunktionerna ”Lägg till” och ”Ta bort”. I det nuvarande gränssnittet finns det fem olika principer vardera för att utföra dessa båda operationer, samt ett par specialfall. För att lägga till ett nytt tabellelement finns fem följande tillvägagångssätt:

1. Klicka ”Lägg till” och nytt fönster öppnas där begärd information matas in.
2. Mata in begärd information i där för avsedda fält, klicka sedan ”Lägg till”.
3. Mata in begärd information direkt i den tomma raden längst ner i tabellen.
4. Klicka ”Lägg till” varpå en ny rad skapas i tabellen, mata därefter in begärd information.
5. Högerklicka i tabellen och få upp en meny och välj där ”Lägg till”, eller tryck ”Insert” på tangentbordet. En ny rad skapas i tabellen där begärd information matas in.

I ett fall har man dessutom implementerat en sorts mini-wizard för att lägga till, nämligen i fönstret Köer och väntelistor. Användaren måste passera en sekvens av tre olika fönster via ”Fortsätt”-knappar för att lägga till en ny kö eller väntelista. För att ta bort ett befintligt tabellelement finns på liknande vis en specialvariant som endast används i en enda dialog. För att ta bort ett element ur tabellen i dialogen Kontakt-datafönster för kampanj (i CallGuide Agent) under menyvalet Media | Utgående telefoni | Kampanjer ändrar man helt enkelt innehållet i en av kolumnernas flervals-ruta till blankrad (” ”). Bortsett från detta fönster har fem olika principer för ta bort identifierats. Den urskiljande faktorn är ofta om och när systemet ställer en motfråga till användaren. Med motfråga menas i detta fall en ja- eller nej-fråga i stil med: ”Vill du ta bort Anna Användare?”.

1. Klicka ”Ta bort” och det markerade elementet försvinner ur tabellen utan motfråga.
2. Klicka ”Ta bort” och det markerade elementet försvinner ur tabellen. Därefter kommer en motfråga och vid ”nej” läggs elementet tillbaks längst ner i tabellen.
3. Klicka ”Ta bort” och en motfråga ställs. Elementet ligger kvar på sin plats i tabellen och försvinner endast vid ”ja”.
4. Klicka ”Ta bort” och raden gråas ut men informationen är synlig. Först vid ”Uppdatera”, ”Spara” eller ”Okej” tas elementet bort från tabellen.
5. Högerklicka i tabellen och få upp en meny och välj där ”Ta bort”, eller tryck ”Delete” på tangentbordet. Tabellelementet försvinner utan motfråga.

På ett ställe förekommer dessutom två olika principer samtidigt, nämligen under Media | E-post | Mönstermatchning. Under de båda flikarna i detta fönster finns sammantaget tre olika typer av element att ta bort. Även fast implementeringen ser likadan ut så ges motfråga enligt princip 2 ovan i ett av fallen, medan de båda andra implementerats enligt princip 1. Detta skapar både osäkerhet och förvirring.

Konsekvensen brister även vad gäller placeringen av export- och lägg till-knapparna, vilken varierar från fönster till fönster. Verktyget skulle få ett mer enhetligt utseende och användarna skulle få betydligt lättare att hitta i olika fönster om placeringen ensades.

5.4.5 Förhindra fel

Vad som är ännu bättre än välformulerade och instruerande felmeddelanden är om designen utformas så omsorgsfullt att felen inte uppstår i första taget.

Skrollern på datormusen är näst intill oumbärlig då man ska leta reda på information i stora datamängder. I CallGuide Admin är det dock lätt hänt att man ändrar informationen i en flervalsmeny istället för att förflytta sig i listan när man skrollar och skrollfunktionen är en stor källa till oavsiktliga misstag. Det är också lätt hänt att information går förlorad i de fönster som innehåller ”Ta bort”-knappar som inte ger någon motfråga i kombination med att det inte finns någon knapp för att avbryta.

5.4.6 Igenkänning framför återminnelse

Gör objekt, handlingar och val synliga. Användaren ska inte behöva komma ihåg information från en del av dialogen till en annan. Instruktioner och anvisningar för hur systemet skall användas skall vara synliga eller lätta att komma åt vid behov.

CallGuide Admin levereras visserligen med en diger manual, men systemet behöver ändå tillhandahålla interaktiv information till användarna. Symboler, markeringar och förkortningar bör förklaras så länge som de inte kan antas vara självförklarande. I dialogen Köer och väntelistor används exempelvis en gul färgmarkering som inte är helt självklar och i Kampanjfönstret finns ett antal symboler som skulle behöva en kort förklaring.

Vidare så innehåller klienten mängder av olika inmatningsfönster där informationsformatet inte alltid är uppenbart. I de båda dialogerna Media | E-post | Automatsvar och Media | Utgående telefoni | Callbackposter skall användaren till exempel mata in värden som svarar mot en kontaktdatanyckel. Användaren får dock ingen hjälp med vilket format värdet måste vara på för att systemet ska kunna använda de inmatade värdena på rätt sätt. Dessa anvisningar ges till användaren först när ett fel har uppstått. Om kortnamnet blir fel när ett nytt organisationsområde ska läggas in så visas exempelvis en ruta med information om att ”Kortnamnet måste innehålla exakt tre tecken a-z.” Denna information borde istället ligga som en hint direkt i fönstret.

Högermenyer är bra och lättillgängliga hjälpmedel för mer avancerade användare, men all information bör även finnas tillgänglig på ett synligt sätt så att mer ovana användare kan lära sig att hitta och nyttja den. I CallGuide Admin är detta oftast inte ett problem. I fönstret för kampanjposter finns dock en del mycket bra funktionalitet endast att tillgå via en högermeny vilket gör den svårfunnen för mer ovana användare.

5.4.7 Flexibilitet och effektivitet

Erbjud genvägar till den avancerade användaren genom att implementera olika former av kortkommandon, osynliga för den mer ovana användaren. Låt användaren själv skräddarsy de vanligaste arbetsmomenten.

Människor fungerar på olika sätt och ett bra system tillhandahåller olika vägar att ta sig fram till samma mål. Det måste alltid finnas minst en tydligt markerad framkomlig väg, men därutöver kan olika typer av genvägar implementeras. I CallGuide Admin finns till exempel många fönster där användaren gör ett urval av tillgängliga kompetenser/behörighetsområden/delområden etc. genom att markera de önskade

valen i en lista och sedan klicka på en pil för att flytta över dem till en lista över aktuella val. För en van användare blir det med tiden frustrerande att behöva dela upp handlingen i två steg och man kan exempelvis tänka sig att dubbelklick flyttar över den markerade informationen direkt, eller att pilarna även finns tillgängliga i en högermeny. Ett annat exempel på bra och beprövade genvägar är acceleratorer. Det innebär att användaren håller nere Ctrl- eller Alt-knappen och sedan använder bokstäverna på tangentbordet för att göra knappval istället för att klicka med musen.

5.4.8 Estetisk och minimalistisk design

Dialoger skall inte innehålla information som är överflödig i sammanhanget eller information som sällan används. All redundant information i en dialog gör det svårare för användaren att urskönja och sortera ut den relevanta informationen.

Informationsöverflöd är ett större problem på menystrukturnivå än i själva dialogerna då menyval som används dagligen är uppblandade med menyval som används mer sällan eller till och med aldrig. Dialogerna i sig nästan är istället *för* sparsmakade i vissa fall och ett flertal skulle med fördel kunna integreras med varandra för att ge samtidig tillgång till den information som behövs för att utföra en viss handling.

5.4.9 Felkorrigering

Hjälp användaren att känna igen, diagnostisera och korrigera fel. Felmeddelanden skall inte innehålla koder som användaren i sin tur måste tolka, utan formuleras på ett klart och tydligt sätt, precisera problemet och föreslå en konstruktiv lösning för att avhjälpa problemet.

Felmeddelanden i CallGuide Admin är ofta formulerade på ett klart och tydligt sätt och preciserar problemet, även om någon konstruktiv lösning sällan föreslås. Det finns dock situationer där det kan vara bra att ge något mer information till användaren. Ett exempel på detta är det informationsmeddelande som kan komma upp vid försök att ta bort en behörighetsroll: ”Kunde inte ta bort roll p.g.a.: Rollen används”. Var den används eller hur man beter sig för att avhjälpa problemet framgår inte. Ett annat exempel gäller menyval och det felmeddelande som ges vid försök att ta bort menyval med definierade öppettider: ”Det finns öppettider definierade för menyvalet”. En mer konstruktiv formulering skulle kunna vara något i stil med: ”Det finns öppettider och/eller undantagsdatum definierade för menyvalet NN och det kan därför inte tas bort. Gå in under Styrning | Öppettider | Menyval och ta bort samtliga öppettider och undantagsdatum för NN innan du gör ett nytt försök.”

5.4.10 Hjälp och dokumentation

Även om det är bättre ifall systemet kan användas utan dokumentation så kan det vara nödvändigt att tillhandahålla sådana funktioner. All sådan typ av information skall vara sökbar, fokusera på användarens uppgift, lista konkreta steg som behöver utföras, samt inte vara allt för omfattande.

Att få tillgång till mer information direkt i klienten är något som efterfrågas av användarna och den heuristiska utvärderingen pekar åt samma håll. I och med att systemet är så stort, komplext och svåröverblickbart skulle användbarheten i ett flertal fönster förbättras bara genom en kort genomgång av vad användaren faktiskt förväntas utföra just där och vad resultatet av handlingen blir.

6 Analys och rekommendationer

Mitt eget samlade intryck är att användarna på det stora hela verkligen uppskattar CallGuide som system. De värdesätter i synnerhet de stora anpassningsmöjligheter som plattformen tillhandahåller, men upplever samtidigt osäkerhet, förvirring och frustration i samband med sitt eget arbete i systemets administrationsklient, CallGuide Admin. Det råder ingen tvekan om att plattformen innehåller en hel del bra och smarta funktioner, men det tar mycket tid för användarna handskas med och navigera runt i systemet för att hitta och utföra funktionerna. Med utgångspunkt i ISO-definitionen för användbarhet blir en första övergripande analys därför att:

- ändamålsenligheten är bra
- effektiviteten kan förbättras
- tillfredsställelsen kan förbättras

En förbättring av tillfredsställelsen handlar i det här fallet i första hand om att göra verktyget mer stabilt och robust så att användarna känner sig tryggare i sitt användande och kan göra sig av med den osäkerhet som idag leder till onödig stress och frustration i deras arbete. Genom att göra klienten mer pålitlig kommer även produktiviteten att förbättras eftersom osäkerheten också gör att användarna tar det mer försiktigt och använder mycket tid till att dubbelkolla sina handlingar. Grundstenen för mina fortsatta rekommendationer blir därför att man bör se till att minska osäkerheten och förbättra effektiviteten.

6.1 Gränssnittsutformning

Om man börjar på detaljnivå finns det helt klart ett antal enskilda fönster och dialoger i CallGuide Admin som skulle kunna förbättras med avseende på användbarhet. Att så är fallet blir tydligt inte minst när man läser igenom användarnas fönsterspecifika synpunkter på gränssnittet. Min bedömning är dock att man i det här skedet snarare bör fokusera på helheten och sätta in alla dessa små byggstenar i ett större sammanhang. Det handlar helt enkelt om att ta ett helhetsgrepp och se till att verktyget upplevs som en genomtänkt och sammanhängande enhet där användarna kan känna sig säkra och utföra sina arbetsuppgifter på ett effektivt sätt. Med utgångspunkt i detta resonemang har jag därför försökt att ta fram en lista över åtgärdsförslag som enligt min uppfattning både kan ge vägledning till hur det nuvarande gränssnittet bör omarbetas och användas som riktlinjer i vidareutvecklingen av klienten för att undvika liknande problem i framtiden.

6.1.1 Rekommendationer

Eftersom det lämnas till utredaren att avgöra vilken typ av problem som bör ligga till grund för vad förändringsarbetet ska fokusera på så är de åtgärdsförslag eller rekommendationer som presenteras i nästa avsnitt en sammanvägning av flera olika faktorer. Stommen utgörs helt och fullt av användarnas egna kommentarer och synpunkter på klienten, samt de iakttagelser jag själv har kunnat göra under datainsamlingen. Den heuristiska utvärderingen har bidragit med ett mer systematiskt helhetsperspektiv och sett till att samtliga delar av systemet har blivit föremål för utredning. De rekommendationer som sedan utkristalliserats är i slutänden ett resultat av mina egna bedömningar av vilka åtgärder som skulle göra mest nytta till lägst

kostnad, samt vad som faktiskt skulle kunna vara praktiskt genomförbart inom en rimlig tidsrymd.

MINSKA OSÄKERHETEN

- **Förbättra konsekvensen**
  - Det finns idag fem olika tillvägagångssätt vardera för operationerna ”Lägg till” och ”Ta bort”. Detta behöver homogeniseras. Välj ut en eller max två olika tillvägagångssätt för dessa funktioner och implementera dem genomgående i hela gränssnittet.
- **Förbättra återkopplingen**
  - Det är ibland svårt att se om en handling utförts vid exempelvis ”Lägg till” och ”Ta bort”. Se till att användaren alltid får återkoppling efter en handling genom att implementera tydliga signaler som är anpassade till situationen. Bibehåll god konsekvens genom att använda samma form av återkoppling i samma typ av situationer.
- **Förhindra uppkomsten av fel**
  - Skrollfunktionen är idag en källa till stor osäkerhet hos användarna. Gör denna funktion smidig och säker genom att ändra tabellinställningarna så att det krävs dubbelklick för att redigera innehållet i en cell.
  - Alla gör vi misstag ibland. Se till att ingen information kan gå förlorad på grund av att användaren klickar på fel knapp, utan att det alltid finns möjlighet att antingen avbryta eller att ångra en handling. I de fönster där man av tekniska skäl inte kan implementera avbryt-knappar bör motfrågor ställas så långt det är möjligt.

FÖRBÄTTRA EFFEKTIVITETEN

- **Underlätta navigeringen**
  - Idag lägger användarna mycket tid på att justera inställningarna i klienten för att få överblick över informationen. Minska antalet klick genom att låta systemet spara användarens fönsterinställningar så som fönsterstorlek, fönsterplacering, kolumnbredder och kolumnordning både under, men även mellan, sessioner.
  - Systemet upplevs generellt som onödigt kompakt. Låt systemet ta upp mer skärmyta och skala upp fönstren så att så mycket information som möjligt är synlig redan då användaren öppnar fönstret första gången.
  - Människor är olika och alla har sitt eget sätt att arbeta. Öka flexibiliteten genom att implementera fler genvägar för avancerade användare. Se dock samtidigt till att all funktionalitet är synlig och även går att komma åt via synliga kommandon.
- **Underlätta förståelsen**
  - CallGuide Admin är ett komplext verktyg och mycket konfigurering kräver en viss förståelse för hur saker och ting hänger ihop. Erbjud användaren mer interaktiv information och tydligare instruktioner direkt i klienten för att reducera tiden för informationssökning.
- **Förbättra menystrukturen:**
  - En tydligare menystruktur är en viktig del i att göra klienten mer lättnavigerad och sålunda mer effektiv. Ett förbättringsförslag till ny menystruktur presenteras i nästa avsnitt.

Tabell 4: Rekommendationer för att minska osäkerheten och förbättra effektiviteten för en användare av CallGuide Admin.

6.2 Menystruktur

Admin har idag en stor och delvis oöverskådlig menystruktur. Ur det material som inhämtats genom observationsintervjuer med användare pekar både iakttagelser och

direkta synpunkter på att den nuvarande menystrukturen påverkar användarnas arbete på ett negativt sätt. Kommentarer från användarna handlar bland annat om att systemet känns svårmanövrerat, klumpigt och innehåller en alltför omfattande mängd fönster och dialogrutor. En användare menar till och med att den hierarkiska uppbyggnaden med modala fönster adderar ett extra arbetsmoment då användaren hela tiden måste öppna och stänga olika fönster för att kunna utföra sitt arbete. Att det klickas mycket fram och tillbaka över skärmen och i menyerna blir också tydligt när man sitter med användarna och ser hur de arbetar.

6.2.1 Planritningen – långa korridorer och många små rum

Beyer och Holtzblatt rekommenderar att man vid vidareutveckling av befintliga gränssnitt kan avstå från att bygga explicita UED-diagram och istället ta hjälp av den tankemodell som är bakgrunden till UED för att organisera gränssnittet på ett bättre sätt. Med detta som utgångspunkt har jag gjort ett försök att måla upp en bild av menystrukturen i CallGuide Admin som om den vore en fysisk struktur. Figur 11 ger en översikt över den befintliga menystrukturen och kan med fördel användas som referens i följande avsnitt.

Som den befintliga menystrukturen ser ut tänker jag mig en institutionsbyggnad med ett system av ett antal långa korridorer, vilka innehåller ett varierande antal dörrar som leder in till små rum eller celler. Mellan dörrarna gömmer sig dock ibland även nya korridoröppningar och dessa korridorer innehåller i sin tur ännu fler rum. Exakt vad som döljer sig i dessa korridorer går dock inte att ta reda på om man inte går dit och läser skylten som är uppsatt på väggen utanför. De små rummen kan i de allra flesta fall endast nås via korridoren och det är endast i ett par enstaka fall som rummen försetts med dörrar mellan varandra. När en uppgift ska utföras måste användaren hitta till de rum som tillhandahåller de verktyg som krävs för att utföra just den uppgiften, och dessutom besöka rummen i rätt ordning. Oavsett om nästa rum ligger i samma korridor eller inte så måste användaren alltid ta sig tillbaks till huvudkorridoren innan han kan leta sig fram till sitt nya mål. Samma sak gäller om han går fel i korridorsystemet och öppnar en dörr av misstag. Det är bara att återvända till huvudkorridoren och börja om. Transportsträckorna som användaren tillryggalägger i systemet är många och långa och tiden skulle med fördel kunna användas till mer produktivt arbete.

Jämför den byggnad som beskrivs ovan med en vanlig villa bestående av hall, vardagsrum, kök, sovrum, badrum, tvättstuga och garage samt en trädgård med redskapsbod. I en välplanerad villa ligger hallen oftast så till att den fungerar som en sluss, en passage för förflyttning mellan husets olika rum. Ett hus där sovrummet endast gick att nå via badrummet skulle med största sannolikhet anses vara rätt opraktiskt och troligtvis inte få allt för många spekulanter vid en eventuell försäljning. Man räknar också med att de olika rummen i ett hus ska tillhandahålla vissa grundläggande funktioner som överensstämmer den livsstil som hör till ens egen kultur. Sängen ska finnas i sovrummet, soffan i vardagsrummet och bilen i garaget. Köket är det rum där man lagar och äter mat och därför är det också smidigt om även alla köksredskap och matvaror förvaras här. Tänk vad arbetsamt det skulle vara att behöva gå ut och hämta elvispen i redskapsboden varje gång den ska användas eftersom den klassificerats som redskap och därför placerats med resten av redskapen fastän det är i köket och inte trädgården den alltid används. En arkitekt som inte tar hänsyn till husägarnas livsstil och önskemål skapar ett hus som ingen vill bo i på samma sätt som

en mjukvarudesigner som inte anpassar systemstrukturen efter användarnas arbetssätt skapar ett system som inte är användbart.

6.2.2 Analys

Som Nielsen säger så ligger den grundläggande kompromissen vid gruppering av element i en hierarkisk menydesign i valet mellan bredd och djup och det gäller att hitta en rimlig utformning där hierarkin varken blir för bred eller för djup. (Nielsen, 1993). Menystrukturen i CallGuide Admin består av totalt 80 valalternativ fördelade över tre nivåer med sex menyval i huvudmenyn och hamnar därmed i princip inom riktlinjerna, vilka påbjuder 4-8 alternativ per meny och mellan 3-4 nivåer. Trots detta upplever användarna att det är krångligt att hitta rätt bland menyerna. Min slutsats är att menyvalen helt enkelt är för svåra att överblicka.

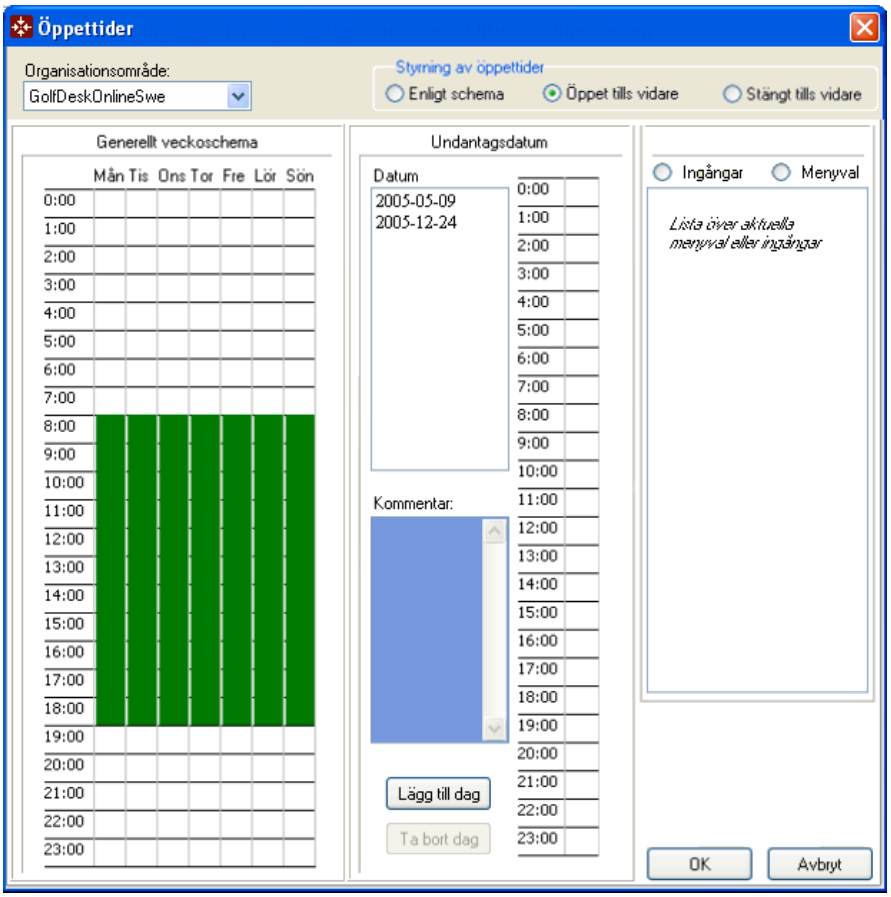
Klientens modala fönsterstruktur i kombination med det faktum att användaren alltid har huvudmenyn som utgångspunkt för sin navigering i menyerna medför att extra vikt bör läggas vid att göra förflyttningen mellan olika delar av systemet så enkel och smidig som möjligt. Något som skulle underlätta arbetet avsevärt är om antalet undermenyer kunde minskas för att ge användaren möjlighet att se så många menyalternativ som möjligt genom att bara föra muspekaren över menyraden. För att dra paralleller till husanalogin så behöver korridorerna ”rätas ut” för att ge användarna en bättre överblick över samtliga tillgängliga alternativ. Att tillämpa sådana åtgärder ligger även i linje med den sjätte punkten på Nielsens lista över designprinciper, vilken går ut på att främja igenkänning framför återminnelse. Användaren ska med andra ord inte behöva minnas vilka alternativ och funktioner som finns i en viss undermeny.

I arbetet med att ta fram ett förslag till ny menystruktur i CallGuide Admin har jag valt att ta fasta på rekommendationen att föredra bredd framför djup i hierarkiska menystrukturer och stöder mig även mot Kiegers empiriska försök som visade att en trädstruktur med 8^2 element var det bästa sättet att organisera 64 menyelement. (Shneiderman, 1998) Ambitionen har även varit att på samma gång se till att samla relaterade funktioner och dialoger under samma meny. Jag ser däremot inget egenvärde i att flytta på menyval bara för att komma ner i åtta alternativ eller färre per meny, utan har behållit en grundstruktur som användarna kan känna igen sig i. Vanans makt är i min mening något som inte bör förringas. Det förslag som presenteras här gör inte anspråk på att vara perfekt i någon mening, utan baseras som alltid på avvägningar mellan olika designbeslut, beslut som i det här fallet helt och hållet är mina egna. En grafisk presentation av förslaget återfinns i figur 12. Färgmarkeringar har använts för att läsaren bättre skall kunna avgöra vilka skillnader som föreligger mellan detta förslag och den befintliga strukturen i figur 11.

6.2.3 Rekommendation

Med utgångspunkt i resonemanget ovan är mitt förslag att en omorganisering av menystrukturen bör fokusera på att få bort samtliga menyval som för närvarande ligger på nivå tre. I ett första steg innebär detta att de menyval på nivå tre som ligger under mycket små undermenyer med endast två eller tre val integreras med varandra och byggs ihop till ett och samma fönster. Så som gränssnittet ser ut idag skulle detta tillvägagångssätt vara tillämpbart på tre olika ställen:

- **Telefonbok:** Idag finns de båda alternativen administrera respektive importera i en undermeny till telefonboken. Att administrera telefonboken är enligt uppgift en av användarnas vanligaste arbetsuppgifter och funktionen har visserligen en snabbvalsknapp, men för den som använder menyerna krävs det tre val för att öppna detta fönster. Dialogen för administrering är dessutom en nod med ett antal underliggande fönster. Menyvalet importera används däremot betydligt mer sällan och innehåller bara ett enda fönster. Genom att helt enkelt lägga in en knapp med rubriken ”importera” i dialogfönstret för att administrera telefonboken blir undermenyn överflödig och kan tas bort.
- **Öppettider:** Även öppettider ingår i användarnas generella arbetsuppgifter och administreringen sker idag via en passage där användaren först måste välja om han vill titta på öppettider för antingen ingångar eller menyval. Det här är ett mycket omständligt arbetssätt för användaren som måste återvända till huvudmenyn om valet råkade bli fel. Genom att integrera inställningarna för ingångar och menyval i samma fönster och låta huvudstyrningen ske på organisationsområden istället skulle arbetet för användaren underlättas avsevärt. Om menyvalet för scheman (VCC) dessutom flyttas ut i undermenyn Styrning och läggs som ett utgråat menyval i listan så behöver användaren inte ens gå in i en undermeny för att komma åt öppettiderna.
- **Interaction View:** Här behövs en kreativ lösning för att integrera de båda fönstren med varandra så att användaren kan göra alla nödvändiga inställningar i ett och samma fönster.

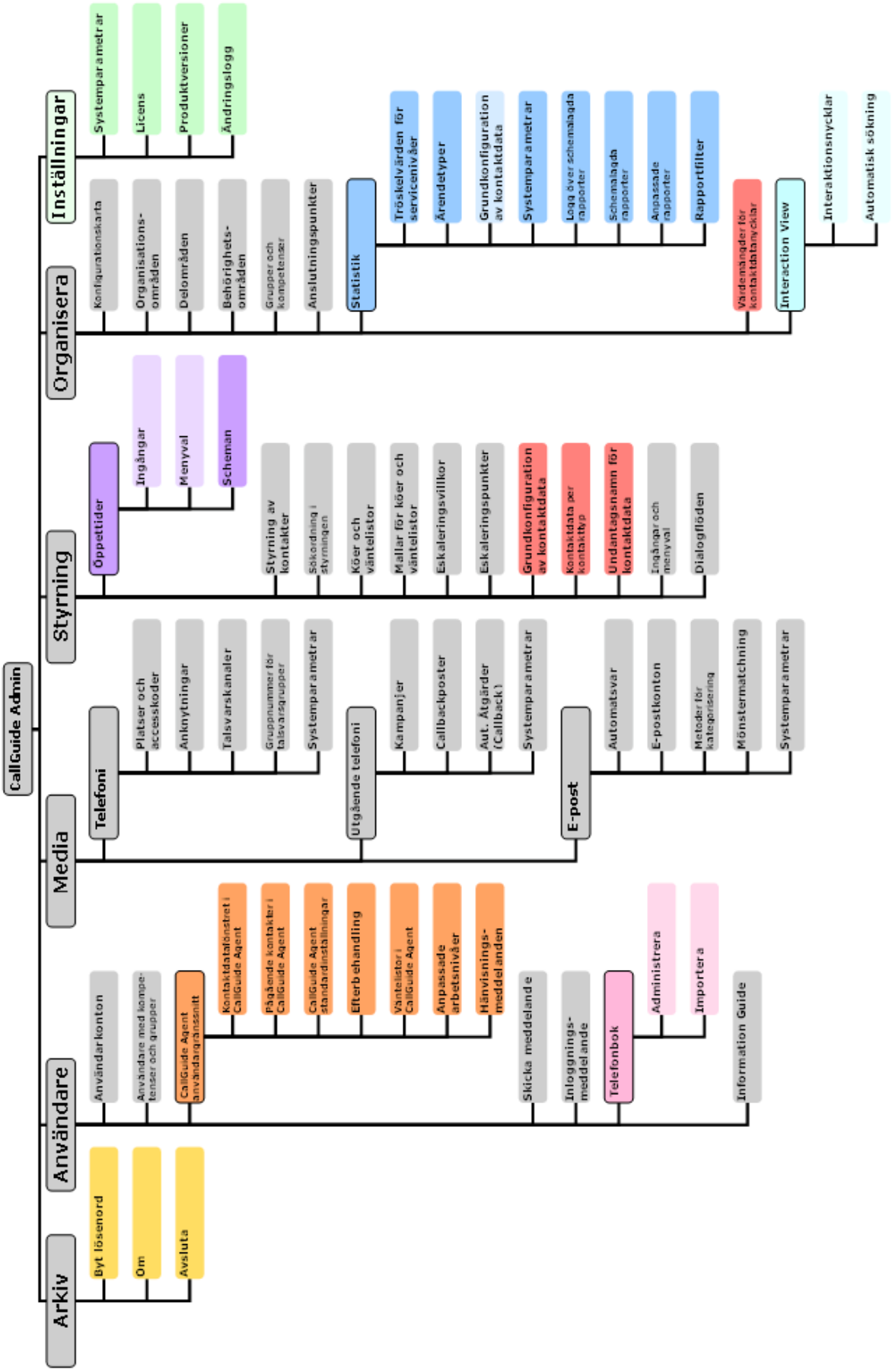


Figur 10: Ett första utkast till ny utformning av dialogen för öppettider i CallGuide Admin. Se figur 7 på sida 47 för jämförelse med dialogen som den ser ut idag.

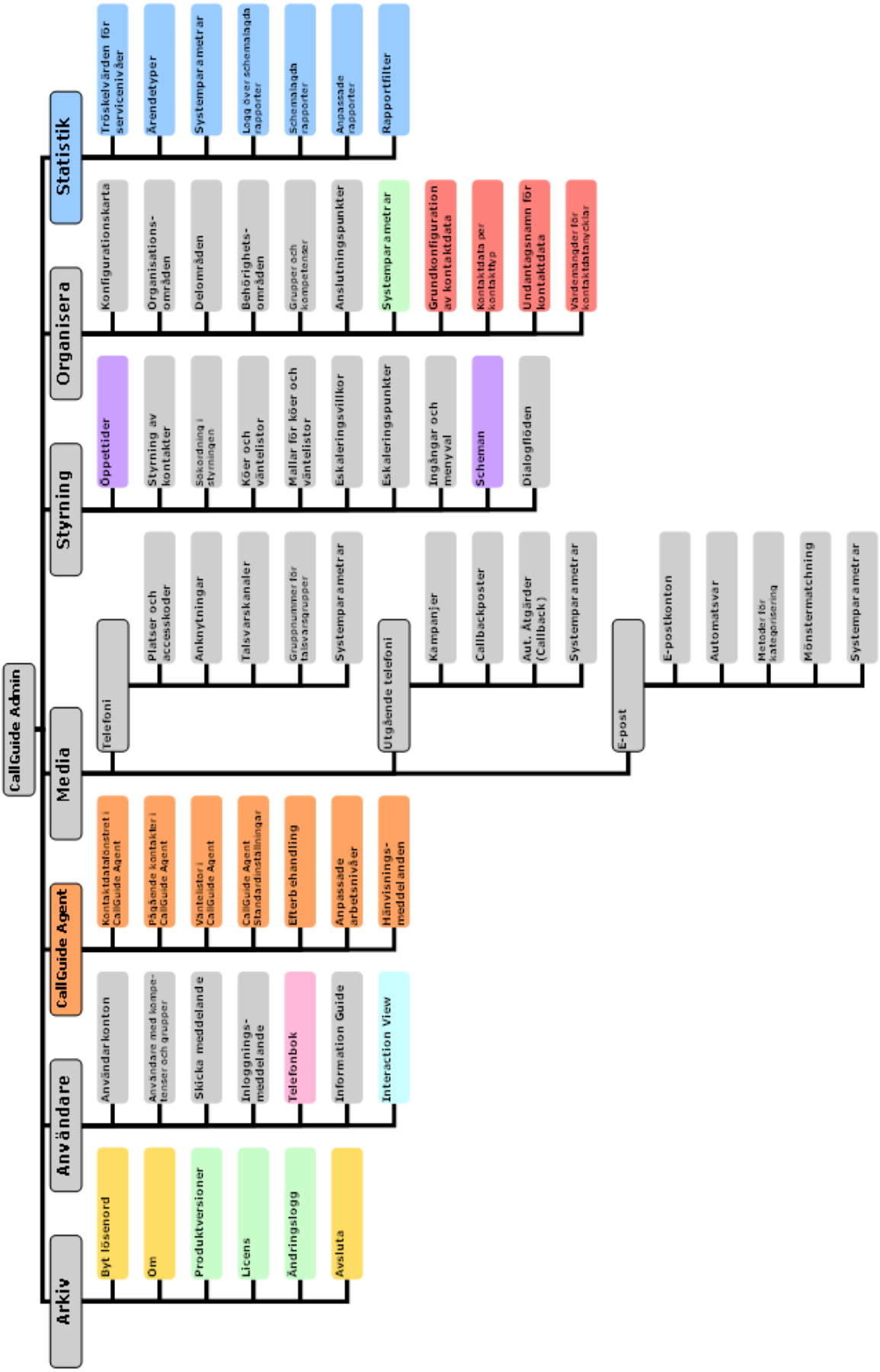
Att bygga om telefonboken enligt ovanstående rekommendation borde inte behöva medföra några större arbetsinsatser. De båda andra åtgärderna kommer däremot att innebära ett något större arbete vad gäller både design och programmering. Som inspirationskälla har jag tagit fram ett utkast till hur en framtida lösning av dialogen för öppettider skulle kunna se ut (se figur 10 ovan).

Det andra steget är att lyfta ut de två stora undermenyer som finns på nivå tre och lägga dem som egna alternativ i huvudmenyn, och samtidigt baka ihop de nuvarande huvudmenyvalen Arkiv och Inställningar. Rent praktiskt innebär detta att ”Användare | CallGuide Agent användargränssnitt” och ”Organisera | Statistik” blir till två nya alternativ i huvudmenyn, medan menyvalet ”Inställningar” försvinner. Ytterligare åtgärder har varit att flytta Interaction View från menyn Organisera till menyn Användare. Detta på grund av att Interaction View är en funktion som administreras i Admin men används i CallGuide Agent precis som Information Guide och Telefonboken, vilka redan ligger under menyn Användare. Vidare har samtliga menyval som är relaterade till kontaktdata samlats under menyn Organisera. För att underlätta mer översiktliga jämförelser mellan de båda menystrukturerna återfinns i figur 13 en schematisk presentation av dem sida vid sida.

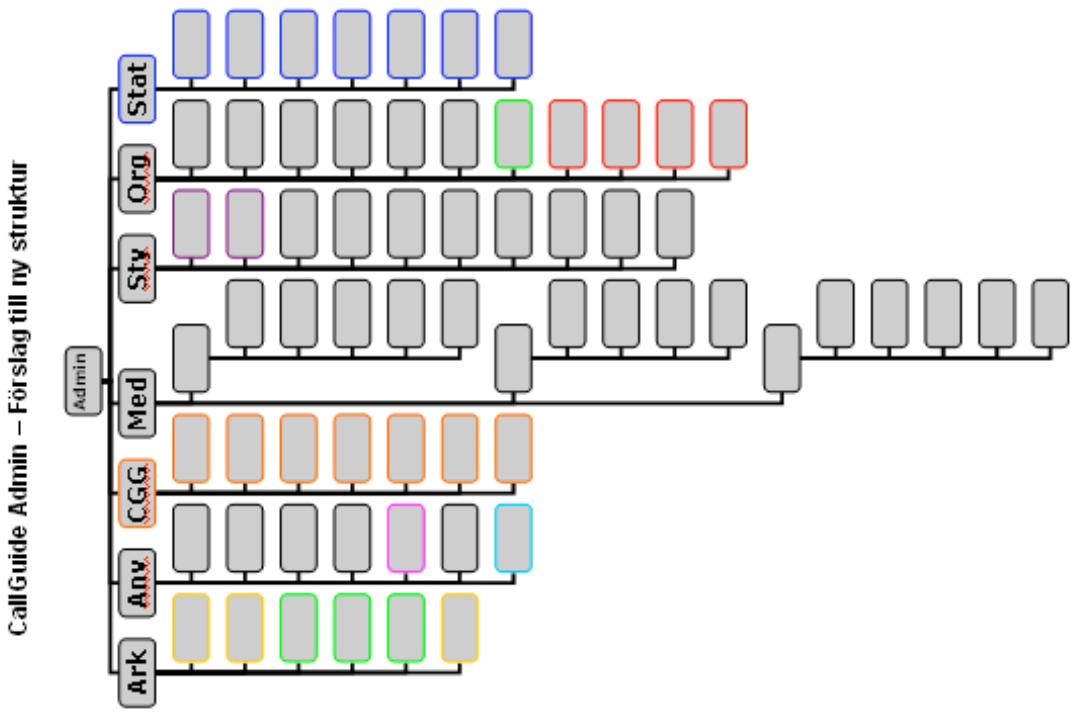
Efter dessa åtgärder har applikationen sju istället för sex menyval i huvudmenyn. Av dessa innehåller sex stycken bara direktval och har alltså inga undermenyer, medan den sjunde istället består av endast tre val som alla har undermenyer. En sådan förändring av menystrukturen kan öka systemets transparens avsevärt genom att synliggöra många fler menyval och samtidigt minska antal klick och musoperationer. På det hela taget skulle detta också bidra till en ökad effektivitet och minskad osäkerhet hos användarna.

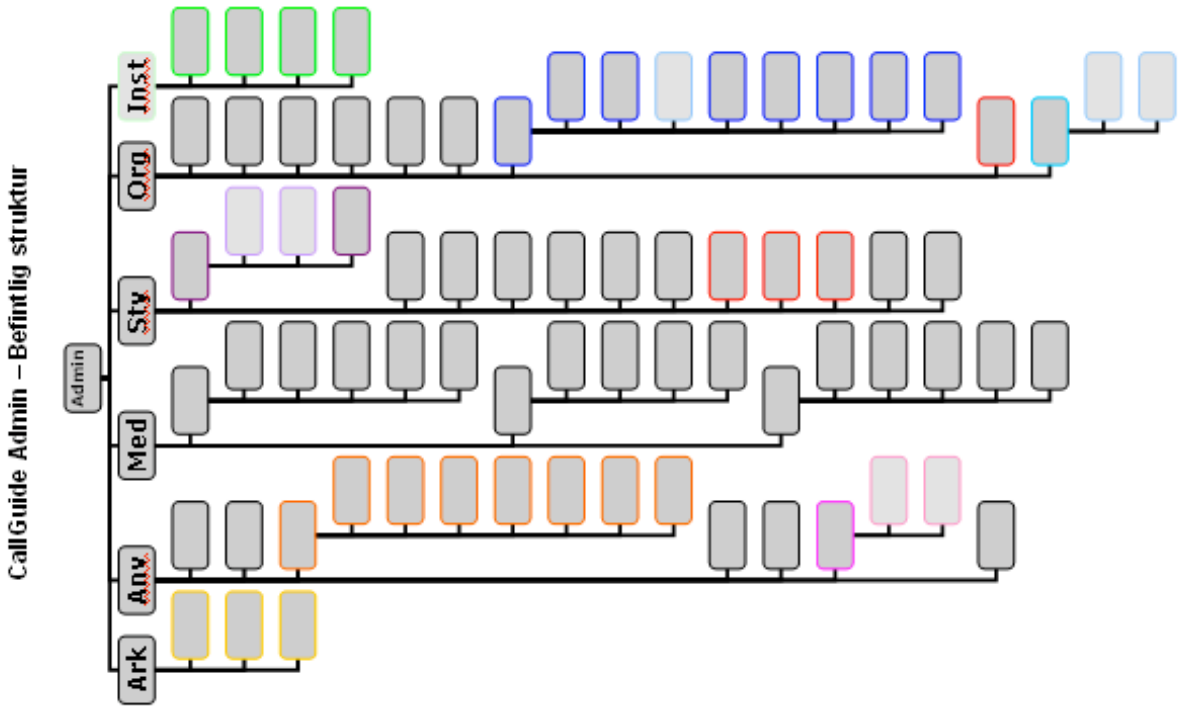


Figur 11: Grafisk representation av den befintliga menystrukturen i CallGuide Admin.



Figur 12: Grafisk representation av mitt förslag till en ny menystruktur i CallGuide Admin.





Figur 13: Jämförelse mellan den befintliga och den föreslagna menystrukturen i CallGuide Admin

7 Avslutande diskussion

När ett arbete av den här omfattningen skall avrundas behöver man alltid fråga sig om utredningen resulterat i ett utfall som kan anses ha både god validitet och hög reliabilitet säkerställd. Validitet handlar om att mäta rätt sak, att ringa in problemområdet och svara på de frågor som faktiskt ställts, medan reliabilitet handlar om att utföra mätningarna på rätt sätt, att använda rätt metoder och vara noggrann i bearbetningen av materialet. (Holme & Krohn-Solvang, 1997) Genom att kombinera två olika utredningsmetoder som rekommenderas till den här typen av utredningar borde validiteten i resultatet kunna sägas vara tämligen god. Reliabiliteten eller pålitligheten i resultatet är däremot svårare att uttala sig om eftersom kvalitativa studier till så stor del är beroende av utredarens tycke och smak. För mig har det därför varit viktigt att ge läsaren tillgång till så pass obearbetad data som möjligt genom att återge ett utförligt resultatet från mina möten med systemets slutanvändare så att läsaren själv kan följa med i resonemanget och ges utrymme att dra egna slutsatser.

7.1 Behov av teoriutveckling

Under arbetets gång har jag samlat på mig en hel del kritik mot de användarcentrerade utvecklingsprocesser som tillhandahålls i dagsläget. Tabell 5 visar schematiskt hur min uppfattning av teorifördelningen mellan olika typer av utvecklingssituationer ser ut. Till att börja med så verkar merparten av de användarcentrerade utvecklingsprocesserna i min mening vara mest lämpade för internutveckling eller kontraktsutveckling där allt från datainsamling till kodning sker inom en och samma organisation. När en kommersiell produkt utvecklas finns däremot ofta många olika kundorganisationer att ta hänsyn till. Det bör framhållas att det visserligen finns gott om metoder och riktlinjer som beskriver urvalet av användare till datainsamling även i dessa situationer, men det saknas ändå flera bitar i pusslet för att göra processerna mer anpassade till den verklighet som möter kommersiella utvecklingsföretag. Vidare så tycks i princip samtliga användarcentrerade utvecklingsprocesser fokusera på nyutveckling av programvara och upphovsmakarna framhåller ofta vikten av att integrera användbarhetsarbete i utvecklingsprocessen så tidigt som möjligt eftersom det blir både dyrare och svårare att genomföra genomgripande förändringar allt eftersom utvecklingsprojektet fortskrider och en färdig produkt börjar ta form.

A. / B.	ny produkt	befintlig produkt
riktad produkt	Merparten av all teori	En del teori
kommersiell produkt	Relativt mycket teori	Relativt lite teori

***Tabell 5**: Tabell över min egen syn på teorimängden för varje kombination av A: ny vs. befintlig produkt och B: riktad vs. kommersiell produkt.*

Det ligger dock någonting i det implicita påståendet att en färdig struktur bara går att bygga om till en viss gräns. För att återanvända den tidigare presenterade husanalogin så kan man göra en jämförelse med TV-programmet ”Extreme home make over” där det ofta går snabbare och ger ett bättre resultat att jämna det gamla huset med marken och bygga ett helt nytt från grunden än att göra ombyggnader av det gamla huset. Oavsett hur kreativa arkitekter och designers man har att göra med så går det inte flytta på bärande väggar. Något som ofta förbises är däremot att man samtidigt måste

väga in de enorma resurser som krävs för att åstadkomma något sådant och betänka att det i slutänden är användarna som får betala för kalaset. TV-programmet slutar ofta med att överförtjusta husägare uttrycker översvallande tacksamhet mot sina välgörare. Det skulle dock vara intressant att i en uppföljare få reda på hur de reagerar när de måste börja betala underhållet för dessa överdådiga hus.

7.1.1 Kostnad, nytta och acceptans

Hur man än vänder och vrider på det så går det inte att komma ifrån att verkligheten för ett vinstdrivande företag i slutänden ändå alltid handlar om pengar. Vad man väljer att göra är till stora delar en kostnadsfråga och det handlar om att avgöra vilken nivå man ska lägga sig på för att få tillräckligt stor nytta till så låg kostnad som möjligt; att hitta en optimal avvägning mellan kostnad och nytta. I det sammanhanget är det även mycket intressant att försöka ta reda på vilken typ av förändringar som ger störst effekt.

En annan mycket viktig fråga i sammanhanget är hur stora förändringar användarna egentligen är villiga att acceptera. Enligt Beyer och Holtzblatt (1998) ska en bra och innovativ systemdesign tillhandahålla en optimal matchning mellan användarnas befintliga arbetssätt och det arbetssätt som introduceras i och med införandet av det nya systemet. För att åstadkomma detta bör systemet erbjuda fördelar som förändrar arbetet tillräckligt mycket för att göra det mer effektivt, och samtidigt bevarar tillräckligt med kontinuitet så att användarna klarar övergången.

7.1.2 En inneboende tröghet

Ett annan infallsvinkel till problemställningen är frågan om hur stor kontroll ägaren av en kommersiell mjukvara, utvecklarorganisationen, egentligen har över en etablerad produkt. Vad gäller datorer, datortillbehör och mjukvara så finns ett klassiskt exempel på en produkt som visat sig vara omöjlig för branschen att göra sig av med trots att studie efter studie visar att betydligt mer användbara och ergonomiska alternativ finns tillgängliga på marknaden, nämligen Qwerty-tangentbordet. Enligt Shneiderman (1998) är detta ett intressant exempel som visar att även dokumenterade förbättringar kan vara svåra att sprida eftersom användarna inte upplever att ansträngningen med att byta tangentbord uppvägs av fördelarna. Ur ett STS-perspektiv ligger det nära till hands att dra paralleller till Thomas P. Hughes idéer om livscykeln hos stora tekniska system (Mayntz, R. och Hughes, T. P., 1988). Hughes baserar visserligen sina idéer på forskning om infrastrukturella system som el- och järnvägsnät men min uppfattning är att begreppsapparaten ändå går att tillämpa även på andra typer av system. Hughes menar att riktigt stora system har en inneboende tröghet (eng. *momentum*). Detta innebär i princip att när en tillräckligt stor massa uppnåtts, när systemet är tillräckligt stort, så fortsätter det att utvecklas längs den från början utstakade vägen utan att någon egentligen kan hejda det. Applicerat på IT-system skulle detta innebära det blir allt svårare att göra omvälvande förändringar av system som, likt Qwerty-tangentbordet, har många olika användare spridda över flera olika organisationer. Detta resonemang sammanfaller också med vad Preece et al. (1994) skriver om att designutrymmet är mindre vid vidareutveckling av en befintlig produkt eftersom produkten formats av beslut som fattats vid tidigare stadier under utvecklingsprocessen.

7.1.3 Gör användbarhet mer användbart

Problemen som man ställs inför då ett befintligt system skall uppgraderas och vidareutvecklas är med andra ord allt annat än triviala och behöver enligt min mening tas upp till diskussion av både teoretiker och praktiker inom användbarhetsområdet:

- Vad är *tillräckligt* bra?
- Vilken typ av förändringar kan antas ge mest nytta i relation till arbetsinsats och kostnad?
- Var går gränsen mellan att göra förändringar i det befintliga gränssnittet och att istället riva och bygga nytt?
- Hur stora förändringar kan göras utan att kontinuiteten går förlorad?

Jag anser att författare, praktiker och teoretiker inom MDI behöver bli bättre på att anpassa kunskapen, modellerna och metoderna så att de blir användbara för dem som faktiskt ska använda dem i sitt dagliga arbete, nämligen företagen. Verkligheten ute i ett vinstdrivande företag ser oftast inte ut på ett sådant sätt att modellerna är direkt applicerbara och de blir därför svåra att anamma. Ett bra exempel på detta är den vanligt förekommande upplysningen om att användbarhetsinsatser bör göras så tidigt som möjligt eftersom det blir dyrare att identifiera och korrigera eventuella användbarhetsproblem ju längre utvecklingen hinner fortskrida. Tanken må vara god, men detta är ändå ett slag på fingrarna, en knäpp på näsan för en mjukvaruorganisation som redan har ett befintligt och kommersiellt gångbart system, men som fått upp ögonen för att något kanske ändå borde göras. Budskapet blir att det är lika bra att ge upp innan man ens börjat för det kommer inte att löna sig. Med de knappa resurser som ofta finns, både vad gäller tid, personal och pengar, är det kanske heller inte läge att sätta sig ner och plöja igenom en större mängd litteratur för att själv hitta argument som kan visa på kostnadseffektiviteten samt välja ut de metoder, processteg och tillvägagångssätt som kan tänkas vara lämpliga att börja med – både för projektet, produkten och personalens del.

Användbarheten måste göras mer användbar. För att komplettera utbudet av användarcentrerade utvecklingsprocesser behövs i mina ögon en process som, istället för att ingående beskriva utvecklingsfaserna för ett *nytt* projekt, fokuserar på den *sista* fasen där nya uppgraderingar görs kontinuerligt. Projektledare inom IT är vana vid att planera uppgraderingar långsiktigt med relativt god framförhållning och borde därför ha lättare att ta till sig en användarcentrerad process som är anpassad till detta arbetssätt.

7.2 Fortsatt utveckling av CallGuide Admin

Under hela den tidsperiod som examensarbetet pågått så har intresset från Telias sida varit mycket stort och ett visst förändringsarbete, bland annat vad gäller bättre feedback, har redan påbörjats. De rekommendationer som presenteras i kapitel 6 är tänkta att vara ett första steg för att förbättra användbarheten i det nuvarande gränssnittet. I förlängningen skulle en utveckling mot ett mer integrerat gränssnitt med färre rum vara önskvärt och min förhoppning är att man tänker till en extra gång innan man skapar ett nytt menyval. GUI-rådet är ett utmärkt forum för diskussioner om plattformens olika gränssnitt och fler designbeslut skulle behöva förankras hos denna grupp innan designen implementeras. För att de ska kunna arbeta med mer

produktövergripande riktlinjer och rekommendationer behöver liknande utredningar som denna genomföras även på övriga klienter i plattformen.

Inom ramarna för examensarbetet har uppgiften varit att titta på applikationen och gränssnittsutformningen vilket medfört att tillhörande manualer och utbildnings-verktyg inte har inkluderats i utredningen. Det bör dock nämnas att många kommenterat manualen som ”ett kapitel för sig” och att ett framtida arbete bör fokusera på att göra informationen mer tillgänglig och lättanvänd för användarna.

Avslutningsvis vill jag föreslå att samtliga utvecklare inom CallGuide-projektet bör åka ut och träffa verkliga användare på deras arbetsplatser och sitta med och titta hur de arbetar med verktyget enligt den lärlingsmetod som förespråkas i Contextual Inquiry. Besöken bör med andra ord inte genomföras i grupp, utan en och en. På detta vis kan projektmedlemmarna få en bättre förståelse för hur användarna brukar produkten och de bitar man själv implementerat och samtidigt lära sig mer om en del av CallGuide som de kanske inte arbetar direkt med själva.

8 Referenslista

Beyer, H. & Holtzblatt, K., (1998). Contextual Design: Defining Customer-Centered Systems, Morgan Kaufmann Publishers, San Francisco

Dix, A., Finlay, J., Abowd, G. & Beale, R., (1998). Human-Computer Interaction, Second edition, Prentice Hall Europe, Hertfordshire

Gerhardt-Powals, J., (1996). Cognitive engineering principles for enhancing human-computer performance, International Journal of Human-Computer Interaction, 8(2), 189-211

Gulliksen, J. & Göransson, B., (2002). Användarcentrerad systemdesign, Studentlitteratur, Lund

Holme, I-M. & Krohn-Solvang, B., (1997). Forskningsmetodik; Om kvalitativa och kvantitativa metoder, Studentlitteratur, Lund

Lindgaard, G., (1994). Usability Testing And System Evaluation; A guide for designing useful computer systems, Chapman & Hall, London

Mayntz, R. & Hughes, T. P., (1988). The Development of Large Technical Systems, Westview Press, Boulder, Colorado

Nielsen, J., (1993). Usability Engineering, Academic Press, Boston

Ottersten, I. & Berndtsson, J., (2002). Användbarhet i praktiken; Praktiska handgrepp, grundbegrepp och tankemodeller, Studentlitteratur, Lund

Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S. & Carey, T., (1994), Human-Computer Interaction, Addison-Wesley publishing company

Shneiderman, B., (1998). Designing the User Interface: Strategies for Effective Human-Computer Interaction, 3rd ed., Addison Wesley Longman Inc., USA

Weiss, M. A., (1999). Data Structures and Algorithm Analysis in Java., Addison Wesley Longman Inc., USA

8.1 Internetkällor

InContext Enterprises Inc., (2005). Contextual Design, www.incent.com (2007-05-03)

Human Factors International Inc., (1999). Bob Bailey reviews Heuristic evaluations, www.humanfactors.com, (2007-03-08)

Nielsen, J., (2005). Ten Usability Heuristics, www.useit.com (2007-03-07)

Polya, G., (1970). Problemlösning – en handbok i rationellt tänkande. Elektronisk version av Bruno Kevius, www.kevius.com (2007-04-29)

Stanford University (1999-2001). Web-creators, Heuristic Evaluation, www.stanford.edu (2007-03-07)

susning.nu, (2003). Heuristik, susning.nu (2007-04-29)

Tognazzini, B., (2003). First Principles of Interaction Design, www.asktog.com (2007-03-08)

U.S. Department of Health and Human Services. Heuristic Evaluation, www.usability.gov (2007-03-07)

Usability Professionals' Association (2005). Usability Body of Knowledge, Principles for Usable Design, www.usabilitybok.org (2007-03-08)

8.2 Intervjuer

[1] Kommunikationsansvariga, statlig förvaltningsmyndighet (2007). Muntl. intervju 2007-01-24.

[2] Systemförvaltare, central förvaltnings- och tillsynsmyndighet (2007). Muntl. intervju 2007-01-31.

[3] Systemadministratör, internationellt transport- och reseserviceföretag (2007). Muntl. intervju 2007-02-06.

[4] Teamledare, internationellt transport- och reseserviceföretag (2007). Muntl. intervju 2007-02-06.

[5] Trafikledare nr.1, nationell trafikledningscentral för kundtjänst (2007). Muntl. intervju 2007-02-07.

[6] Trafikledare nr.2, nationell trafikledningscentral för kundtjänst (2007). Muntl. intervju 2007-02-07.

[7] Systemförvaltare, statlig myndighet med bredd (2007). Muntl. intervju

Falkman, Lena (2006). Ansvarig för utveckling och förvaltning av Telias egen kundtjänstplattform, muntl. intervju 2006-12-12.

Fogelberg, Jonas (2006). Affärsstöd, muntl. intervju 2006-12-15.

Grahn-Östlund, Anna (2006). Utbildningsansvarig för produkten CallGuide, muntl. intervju 2006-12-15.

Strömberg, Jessica (2006). Uppdragsbeställare åt kund, muntl. intervju 2006-12-14.

Törnqvist, Anna (2006). Leveransprojektledare för Contact Center-lösningar, muntl. intervju 2006-12-11.

Bilaga A – Intervjuguide

Praktisk information

- Är det okej att spela in intervjun?
- Är det okej att citera intervjumaterialet i den skrivna rapporten?

Personlig information

- Vilken befattning har du och vilka arbetsuppgifter innebär det?
- Hur länge har du haft den positionen?
- Vilken roll har du i företaget och utvecklingsarbetet?
- Vilken bakgrund har du vad det gäller utbildning och yrkesliv?

Övergripande Telia CallGuide

- Vilka använder Telia CallGuide?
- Hur ser din kontakt med olika användare av Telia CallGuide ut?

Användargrupper Admin

- Vilka använder Telia CallGuide Admin? Finns det olika typer av användare?
- Finns det någon användargrupp som du anser borde prioriteras framför andra?

Produktanvändning Admin

- Vilka arbetsuppgifter utförs med hjälp av Admin-klienten?
- Vilka är enligt dig de *vanligaste* arbetsuppgifterna som utförs i Admin?
- Hur arbetar användarna med Telia CallGuide Admin enligt din uppfattning?
  - sällan/ofta, fortlöpande/grundinställning vid uppstart
- Vilka upplever du som de vanligaste synpunkter som användare har på verktyget?
- Vilka anser du själv vara Admin-klientens styrkor och svagheter?