



KUNGL. TEKNISKA HÖGSKOLAN

Royal Institute of Technology
Numerical Analysis and Computing Science

TRITA-NA-D9811 • CID-38, KTH, Stockholm, Sweden 1998

Temadag på CID
Användarcentrerad systemutveckling och kravhantering
Inger Boivie, Jan Gulliksen och Ann Lantz



CID
Centre for
User Oriented IT Design

Inger Boivie, Enator AB och CID

Jan Gulliksen, Uppsala Universitet och CID

Ann Lantz, NADA och CID

Temadag på CID – Användarcentrerad systemutveckling och kravhantering

Report number: TRITA-NA-D9811, CID-38

ISSN number: ISSN 1403-073X

Publication date: December 1998

E-mail of author: inger.boivie@enator.se, jan.gulliksen@hci.uu.se och alz@nada.kth.se

URL of author:

http://cid.nada.kth.se/cid/jml.cgi/m_view.jml?lang=se&userName=IngerBoivie,

<http://www.hci.uu.se/contact/jg.html>

<http://best.nada.kth.se:8080/iplab/jml.cgi/personView.jml?userName=alz>

Reports can be ordered from:

CID, Centre for User Oriented IT Design

Nada, Dept. Computing Science

KTH, Royal Institute of Technology

S-100 44 Stockholm, Sweden

telephone: + 46 8 790 91 00

fax: + 46 8 790 90 99

e-mail: cid@nada.kth.se

URL: <http://www.nada.kth.se/cid/>

INNEHÅLL

FÖRORD	2
UPPLÄGG	3
SAMMANFATTNING	4
SLUTSATSER OCH FORTSÄTTNING	6
PLANERINGSORIENTERAD KONTRA ERFARENHETSBASERAD UTVECKLINGSMODELL	7
PANELDEBATT 1 - HUR FUNGERAR DET I PRAKTIKEN	10
Situationsanpassning	11
Användarcentrerad systemutveckling	14
Visioner, önskemål och krav	17
Iterationer - hur gör man	18
GRUPPÖVNING	19
Praktikfall 1	21
Praktikfall 2	23
Praktikfall 3	24
PANELDEBATT 2	25
Behov kontra krav - Torbjörn Näslund	26
ACSU och kravhantering i offertprocessen - Lars Dahlbom	28
Kommunikationskvalitet i informationssystem - Owen Eriksson	29
Designprinciper för nya grupper och tjänster - Hans Marmolin	31
REFERENSER	32
DEFINITIONER	32
DELTAGARE	33

Förord

För några år sen kom jag i kontakt med kravhantering på ett seminarium anordnat av VI (Sveriges Verkstadsindustrier). Där diskuterades numrering, spårbarhet, formella språk, etc - dvs, hur man hanterar kraven när man väl fått in dem. Man förutsatte att de var väl definierade och kända från början. De, i mitt tycke, svåra frågorna diskuterades aldrig, t ex:

- Hur samlar man in kraven?
- Hur vet man att det är rätt krav?
- Hur vet man att man fått in alla krav?
- Hur vet vi att vår tolkning av kraven stämmer överens med kravställarens?
- Hur hanterar vi förändringar?

Sedan dess har jag arbetat mycket med användarcentrerad systemutveckling (ACSU) och missionerat mycket om Enators ACSU-modell (PAS). Men inte heller ACSU har svaren på alla frågor eller lösningen på alla problem.

Är det så att kravhantering och ACSU står för två skilda synsätt på systemutveckling?

- I kravstyrda modeller strävar man efter struktur och ordning. Sådana modeller bygger på sekvensiellt arbete och antagandet att man först kan analysera problemet och sedan designa en lösning. Förändringar ses som något som bör undvikas eller i vart fall nogsamt kontrolleras - ett nödvändigt ont?
- I ACSU arbetar man iterativt, förändringar ses som en nödvändig förutsättning för att komma framåt. Problemet analyseras och utvecklas samtidigt som man skapar lösningen.

Med tanke på dessa skillnader ställde jag följande fråga inför temadagen:

ACSU och kravhantering - går de överhuvudtaget att förena?

Inger Boivie, Enator AB

Upplägg

- 9.00-9.30 Ankomst, fika
- 9.30-10.00
- Inledning - Inger Boivie,
 - Föredrag: Sten-Erik Öhlund, SISU
- 10.00-11.30 Paneldebatt 1 -Hur fungerar det i praktiken? Ordförande: Joachim Karlsson, Linköpings Universitet och Focalpoint
- 11.30-12.30 Lunch
- 12.30-14.30 Ett fiktivt praktikfall - föredragare: Jan Gulliksen, CMD, Uppsala Universitet. Gruppdiskussioner - fika under tiden
- 14.30-16.00 Paneldebatt 2 - Användarcentrerad utveckling och kravhantering - går de att förena. Ordförande: Sten-Erik Öhlund, SISU

Sammanfattning

Inlednings anförande

Sten-Erik Öhlund, SISU, inledde dagen med en kort historik samt en diskussion om utveckling ur ett produktperspektiv.

Sten-Erik drog paralleller med produktutveckling där man sett ett paradigmskifte under 80- och 90-talet - från tungrodda, välkontrollerade processer (over-the-wall-engineering) till snabba, effektiva, innovativa processer. Kan vi lära från detta skifte? Det finns likheter mellan de problem man har inom systemutveckling och de som finns/fanns inom produktutveckling.

Bland annat kan vi lära oss att det finns många olika sätt att bli snabb, och att effektiviteten beror på uppgiften och situationen.

Några av de frågor som diskuterades

Diskussionen täckte många olika områden, bl a nedanstående frågor:

- Användarcentrerad systemutveckling - vad innebär det egentligen?
- Situationsanpassning - vilka faktorer styr valet av arbetssätt?
- När kan man/ska man arbeta användarcentrerat/iterativt?
- Hur arbetar man iterativt?
- Kan man ersätta krav med andra sätt att beskriva behoven?
- Vad innebär kommunikationskvalitet?
- Kan man tillämpa användarcentrerade principer och arbetssätt när man utvecklar tjänster inom t ex digital-TV?

Användarcentrerad utveckling - vad är det

Någon konsensus om vad användarcentrerad utveckling är nådde vi aldrig. Definitionerna varierade från modell-baserad utveckling till användarstyrd utveckling, dvs att användarna bygger sina egna system.

Att användarna ska involveras var däremot de flesta överens om, dock ej i vilken omfattning och hur. Detta beror naturligtvis också på typen av projekt och system.

Det finns också andra intressenter än användarna, vars behov måste tillgodoses, t ex beställare och kundens kund. De olika gruppernas intressen kan stå i konflikt med varann.

Situations- anpassning

Vilka faktorer styr huruvida man kan arbeta användarcentrerat/iterativt? De faktorer som diskuterades var storlek på system/projekt, typ av system, leverantörsförhållande och andra begränsningar. Den viktigaste begränsningen ansågs tillgången till användare vara. Om man inte kan få tillgång till användarna ska man inte arbeta användarcentrerat.

När kan man arbeta användarcentrerat	Ett användarcentrerat och iterativt förfarande ställer mycket stora krav på relationen mellan beställaren och leverantören. Finns inte denna tillit krävs istället utfästelser, t ex garantier om leverans mot en frusen kravspecifikation. Man måste vara överens om ett iterativt arbetssätt från början.
	Det kan därför vara svårt eller omöjligt att arbeta användarcentrerat t ex vid offentlig upphandling där kraven på formella processer är höga, vid fastprisuppdrag, om det redan finns en frusen kravspecifikation, etc.
Hur arbetar man iterativt	Hur styr, planerar och prissätter man en iterativ process? Hur vet man när man är klar? Svaren skiljde sig: ställ upp mätbara mål, låt kunden bestämma, formulera en frågeställning.
Kan man ersätta krav	Kan man lösa problemen genom att ersätta kravspecifikationer med visioner och önskemål? Skrota kravspecifikationen och formulera visioner istället. Dessa kan sedan brytas ner i önskemål, som är lättare att förändra och omförhandla.
	Om man måste arbeta med kravspecifikationer är det viktigt att de hålls på en hög, strikt vad-orienterad nivå.
Kommunikationskvalitet	Räcker det att fokusera på användaren och användbarheten? Owen Eriksson införde begreppet kommunikationskvalitet. Han menade att informationssystem är i grunden sociala och att man därför måste utveckla för och utvärdera kommunikationskvaliteten i systemen, inte enbart användbarheten.
Tjänster inom t ex digital TV	Kan man tillämpa användarcentrerade principer och arbetssätt när man utvecklar tjänster inom t ex digital TV - dvs upplevelsebaserade system? Hans Marmolin menade att man här måste arbeta med traditionell användbarhet såväl som attraktivitet. Den sociala situationen blir viktigare än uppgiften. En tredje aspekt är samverkande kompetenser i utvecklingsprojekten - personer med estetisk bakgrund ska samarbeta med tekniker och beteendevetare.
Grupparbete	Ett inslag var gruppdiskussioner av tre olika fiktiva praktikfall som diskuterades i sex grupper. Grupperna fick ett antal frågor angående användarcentrering och krav att diskutera under ca en timme. Sammanfattningsvis kan sägas att de flesta grupper valt olika angreppssätt. Angreppssätten skiljde sig framförallt mellan de olika praktikfallen, där användargruppens sammansättning och storlek varierade. Men även för samma praktikfall hade grupperna valt olika angreppssätt.

Slutsatser och fortsättning

Har vi lärt något sen Våren 1997 anordnade CID en temadag runt ämnet användarcentrerad systemutveckling i praktiken. En av slutsatserna från denna dag var att svårigheterna med användarcentrerad systemutveckling beror på attityder och kommunikationsproblem och att utbildningsväsendet har ett stort ansvar att bearbeta detta problem.

På temadagen hösten -98 diskuterade vi fortfarande delvis samma problem, vilket leder till slutsatsen att de långt ifrån är lösta. Även om diskussionen har mognat avsevärt. Det är inte lika svårt att plädera för användarcentrerad systemutveckling och användbarhet längre. Däremot finns det fortfarande hinder och svårigheter när man försöker tillämpa tankarna i praktiken.

Obesvarade frågor Debatten täckte ganska många frågor av olika karaktär. Några av dessa frågor hann vi inte diskutera klart, bl a följande:

- Hur integrerar man användarcentrerad utveckling i andra systemutvecklingsmodeller?
- Hur gör man om man inte har tillgång till användarna?
- Hur gör man om man vill arbeta iterativt p g a osäkerheten i projektet, men inte får p g a externa krav på formella processer?
- Hur hanterar man ACSU i affärsprocessen?
- Måste man arbeta annorlunda med användarcentrerad systemutveckling av ny teknik?

Andra frågor, t ex några av de som stod med i inbjudan, kom aldrig upp till diskussion.

Hur fortsätter vi

Vi känner ett behov av att fortsätta diskussionen. Ett sätt vore att välja ut ett antal frågor av de som kom upp under temadagen, och att fortsätta debatten runt dessa. Vi hoppas kunna arrangera ytterligare en temadag under våren 99 - med ett mer fokuserat tema.

Om du har idéer eller frågor du vill diskutera, hör gärna av dig till någon av författarna till denna rapport.

Planeringsorienterad kontra erfarenhetsbaserad utvecklingsmodell

Inledning	Sten-Erik Öhlund, SISU (<i>Ref 1</i>), inledde dagen med lite historik samt en diskussion om utveckling ur ett produktutvecklingsperspektiv.
Har jag inte hört det förut	<p>Sten-Erik gav en kort historik över hur systemutvecklingsprocessen förändrats - från 60-talet till idag:</p> <ul style="list-style-type: none">• Slutet av 60-talet: iterativ utveckling, spiralmodell, avgränsade system, hålkort, batch, personberoende• Början av 70-talet: ordning och reda, kontroll, styrning, metod, personoberoende, dokumentation, SIS-RAS, skivminnet, organisations-övergripande system, central stordator.• Början av 80-talet: fler användare i org, högre interaktivitet, gränssnitt, experimentell systemutveckling, APL, CS5• Slutet av 80-talet, början av 90-talet: PC, standardprogram, massmarknad av användare, CASE verktyg, objektorientering, formella specifikationer• Början av 90-talet: Internet, fokus på kommunikation, nya tjänster, bättre interaktionsmiljöer, användarcentrering, iterativ utveckling, ad hoc-utveckling utan metoder (webb), ny generation systemutvecklare
Kravhantering - definition	<p>Sten-Erik visade flera definitioner av kravhantering (requirements engineering), bl a följande:</p> <p><i>"Definition of software requirements-the software engineering process of determining what is to be produced-and the products generated in that definition. The process involves all of the following: (1) requirements identification, (2) requirements analysis, (3) requirements representation,(4) requirements communication, (5) development of acceptance criteria and procedures. The outcome of requirements definition is a precursor of software design"</i>.</p> <p>Brackett, J. Boston University (<i>Ref 2</i>)</p> <p>Andra definitioner vidgar begreppet till att omfatta:</p> <p><i>"The area of knowledge concerned with communicating with organisational actors with respect to their visions, intentions, and activities regarding their need for computer support, and developing and maintaining an adequate requirements specification of an information system. This definition suggests that Requirements Engineering includes managerial, organisational, economic, social, as well as technical issues and problems"</i></p> <p>Janis Bubenko Jr, Challenges in Requirements Engineering 95 (<i>Ref 3</i>)</p>

Användarcentrerad systemutveckling - definition

Definitionen av användarcentrerad utveckling varierar också - en definition baseras på fyra principer enligt Gould och Lewis (1983) (Ref 4):

- Tidig fokus på användare
- Tidig och kontinuerlig utvärdering av användbarhet
- Iterativ utveckling
- Integrerad utveckling

Kan man jämföra

Kan man jämföra kravhantering och ACSU? Kan man diskutera i termer av användarcentrerad kravhantering eller kravhantering med användarfokus? Finns det motsättningar och i så fall var?

- analytisk vs experimentell?
- modell- och dokumentorienterad vs användning av protyper, tester?
- sekvensiell vs integrerad?
- linjär vs iterativ?
- beställarfokus vs användarfokus?

Produktutveckling

Kan vi lära av produktutveckling? Det finns intressanta likheter med användarcentrerad utveckling enligt Gould och Lewis. En ny paradigm har slagit igenom på bred front, med följande "buzz words", se :

- Concurrent Engineering
- Simultaneous Engineering
- Integrerad Produktutveckling
- Lean Production

På grund av konkurrenssituationen inom den produktutvecklande industrin har man tvingats gå från 80-talets tungrodda "over-the-wall-processer" till att bli snabba, effektiva och innovativa produktframtagare. Man fokuserar på följande faktorer:

- tung projektledning med produktkompetens - projektledarna är produktframtagare snarare än projektadministratörer
- integration - man jobbar med "framladdad" utveckling (ändringar tidigt i projektet) och i tvärvetenskapliga team
- visioner
- kommunikation - successiv informationsöverföring

Sten-Erik gjorde en jämförelse mellan sk planeringsorienterad och erfarenhetsbaserad utveckling.

Planeringsorienterad produktutveckling - vad är det *Planeringsorienterad utveckling* (kompressionsmodellen) antar att produktutveckling är förutsägbar i ett antal steg som kan komprimeras. Man försöker alltså snabba upp processen genom att komprimera stegen. Man försöker förutse de problem som kan förutses. Tidplaner är viktiga och man belönar att dessa hålls.

Erfarenhetsbaserad produktutveckling - vad är det *Erfarenhetsbaserad utveckling* bygger på att snabbhet inte kan nås genom att accelerera existerande processer eftersom produktutveckling är en högst osäker resa med ständigt förändrad marknad och teknologi. Man itererar ofta för att öka sannolikheten att träffa rätt och för att öka kunskapen om produkten. Man testar ofta för att upptäcka fel så tidigt som möjligt. Erfarenhetsbaserad utveckling kräver frekventa milstolpar för att kontrollera att man är på rätt spår och för att undvika kaos. Det krävs också tung projektledning.

Planeringsorienterad kontra erfarenhetsbaserad I en undersökning genomförd 1995 (*Accelerating Adaptive Processes: Product Innovation in the Global Computer Industry*, Eisenhardt, Tabrizi, *Stanford University, 1995, Administrative Science Quarterly, 40*) ingick 36 företag inom dataindustrin (stordator, minidator och PC) 72 produktutvecklingsprojekt. I undersökningen gjordes en jämförelse mellan planeringsorienterad och erfarenhetsbaserad utveckling.

Resultatet av undersökningen var:

- Planeringsorienterad utveckling fungerar enbart i situationer där förutsägbarheten är hög
- Planering och belöning av tidhållande => *långsammare* utveckling - speciellt vid stor osäkerhet
- Erfarenhetsbaserad modell effektivast vid osäkerhet

Systemutveckling Kan vi tillämpa samma idéer inom systemutvecklingen? Det finns många likheter:

- Samma utmaningar: teknik, marknad, användare - i ständig rörelse
- Från systemutveckling till produkt- och tjänsteutveckling
- Ökat partnerskap - beställare och leverantör
- Design i fokus - produktkvalitet = upplevd kvalitet

Slutsats Från produktutveckling kan vi lära oss att:

- Det finns många olika sätt att bli snabb
- Planeringsorienterad utveckling - effektiv organisering av process
- Erfarenhetsbaserad - accelererat lärande
- Effektivitet beroende av uppgift
- Rationalitet kontra intuition och improvisation

Paneldebatt 1 - Hur fungerar det i praktiken

Deltagare

Följande personer deltog i panelen:

- Ordförande: Joachim Karlsson, Focal Point
- Deltagare
 - * Gunilla Stoberski, Cap Gemini
 - * Ingrid Ottersten, LinnéData
 - * Lena Andersson, Enator
 - * Kjellåke Henriksson, RSV

Inledning

Joachim inledde med en diskussion runt produktstrategier. Hur kan vi utveckla bästa möjliga produkt på minsta möjliga tid? Hur ska vi prioritera användarnytta kontra kostnad?

Standish Group genomförde 1995 en undersökning som visar att 16 % av de undersökta projekten lyckades, dvs producerade det förväntade resultatet inom den uppsatta tiden och kostnaden. En ESPITI-undersökning visar att problemen som uppstår vid systemutveckling framförallt handlar om att kartlägga det verkliga behovet. Båda dessa undersökningar pekar på att det är mycket viktigt att involvera användarna och ha kontroll över kraven. Andra råd är att minska storleken på projekten och att arbeta iterativt.

Situationsanpassning är också viktigt - förutsättningarna skiljer sig dramatiskt från situation till situation. I bilindustrin är det oerhört viktigt att veta exakt vad man gör - i web-världen råder motsatsen ”Ta fram nåt kul så får vi titta”.

Frågeställningar

Följande punkter berördes under debatten.

- situationsanpassning - vad passar när
- användarcentrerad systemutveckling - vad är det, användarmedverkan, svårigheter, etc
- visioner, önskemål och krav - kan man byta ut krav mot visioner/önskemål
- iterationer - hur styr och planerar man en sådan process

Situationsanpassning

Kritiska variabler När kan/bör man arbeta användarcentrerat/iterativt? Följande fyra kritiska variabler diskuterades

- storlek på system/projekt
- typ av system
- leverantörsförhållande
- begränsningar

Storlek på projekt Frågan om det går att bedriva iterativ utveckling i stora projekt fick inget entydigt svar. Ju större projekt, ju högre komplexitet - kan man då ytterligare öka komplexiteten genom att arbeta iterativt, och därmed minska kontrollerbarheten i projektet? Ja, sa vissa. Nja, sa andra. Å andra sidan är stora projekt svåra att bedriva även med hjälp av traditionella modeller.

För att kunna arbeta iterativt måste man dela upp stora projekt och köra delarna parallellt. Dock kan detta vara mycket svårt på grund av för stora inberoenden (dvs att de olika delarna är beroende av varann) mellan de olika delarna - ibland helt omöjligt.

I de fall man ska dela upp projekten krävs en inledande analys av avgränsningarna så att gränstorna blir så tydliga som möjligt. Sedan ställs det stora krav på planering, styrning och samordning av de olika delprojekten.

Microsoft nämndes som exempel på ett företag som anammat ett iterativt arbetssätt i stora projekt. Där itererar man dagsvis - dvs man kodar på morgonen, kompilerar och testar på eftermiddagen och rättar på kvällen.

Typ av system Enligt panelen kan man arbeta arbetarcentrerat i stort sett vad gäller alla typer av system, utom rena funktionskonverteringar, där det gamla systemet i princip är kravspecen. Detsamma gäller inbäddade system där interaktion med användare inte förekommer.

Dock skiljer sig sättet att arbeta användarcentrerat för *arbetsrelaterade* system och *upplevelsebaserade* system. Se mer Arbetsrelaterade kontra upplevelsebaserade system, sid 16.

Leverantörsförhållanden

Det finns ingen motsättning mellan kontraktsupphandling och iterativ utveckling. Om man ska arbeta användarcentrerat/iterativt *måste* man dock vara överens med beställaren om detta arbetssätt. Man måste, t ex, ha diskuterat igenom för och nackdelar. Om man inte har kommit överens om ett iterativt arbetssätt är risken för konflikter *mycket* stor.

Är man överens med kunden om att arbeta iterativt kan man koncentrera sig på att bygga det kunden vill ha - inte nödvändigtvis det som står i kravspecen. Det kan ta längre tid och kosta mer, men kunden blir nöjd. Linnédata, t ex, styr med hjälp av parametrarna tid, ambition och kostnad. Ska man lägga till mer pengar, mera tid eller skära i något.

Vid offentlig upphandling kan man inte arbeta iterativt, på grund av juridiska krav på offertprocessen.

Begränsningar

Det som styr huruvida man kan arbeta användarcentrerat är tillgången på användare. Har man inte tillgång till användare - t ex på grund av användarna inte anses ha tid - ska man inte arbeta användarcentrerat. Senare under debatten kom frågan upp hur och när användarna ska involveras - se mer Användar-medverkan, sid 15.

Andra faktorer

En annan faktor som nämndes vad gäller val av systemutvecklingsmodell är systemets livslängd. Om det är ett system med kort livslängd behövs inte en detaljerad kravspecifikation - då kan man verifiera det i drift och snabbt komma ut med ett nytt system..

Exempel - RSV

På RSV har man arbetat med tre projekt, där den sammanhållande faktorn har varit en rumsmetafor. De tre delprojekten, ett stort, ett mellanstort och ett litet har alla tre producerat applikationer med sju, tre resp ett rum. De olika rummen ser därför lite olika ut.

I det lilla projektet, vars resultat blev ett enda rum, har man arbetat mer iterativt än i de övriga projekten. Det är också där man har lyckats bäst med att leverera något som användarna är nöjda med. Nöjda är de användare som nästan uteslutande använder denna applikation. De användare som använder alla tre applikationerna är inte nöjda eftersom den designmässigt m m avviker från övriga applikationer.

Dock har man kunnat konstaterat att det inte alltid är så att användarna vet bäst. Man har också haft problem med urvalet av användarrepresentanter. Det är inte alltid de mest förändringsbenägna som deltagit i processen, vilket har negativ effekt på resultatet.

Ledningen på RSV har beslutat att inga projekt ska vara längre än 6 månader. ”Detta löser man genom att dela upp projekten.” Sa RSV-representanten ironiskt.

En försvårande faktor på RSV är att man har absoluta slutdatum för sina utvecklingsprojekt. Den 1/7 varje år träder ett antal nya lagar och lagändringar i kraft. Då måste man ha systemstöd för dessa - man kan inte skjuta på leveranserna.

Användarcentrerad systemutveckling

Diskussionspunkter Användarcentrerad systemutveckling (ACSU) diskuterades vid ett flertal tillfällen under paneldebatten. Följande diskussionspunkter kom fram vad gäller ACSU:

- användarcentrerad systemutveckling - vad är det
- användarmedverkan- när och hur
- flera intressenter - hur tar man hänsyn till andra intressenter än användarna
- relationer - tillit mellan leverantör och beställare
- svårigheter - vilka problem är vanliga vid ACSU
- byta ut ”användarcentrerad” - skulle det lösa problemen
- skillnader mellan arbetsrelaterade system och upplevelsebaserade

Vad är användarcentrerad systemutveckling

Vad är användarcentrerad systemutveckling (ACSU)? Panelen och åhörarna lyckades aldrig nå konsensus om en definition av ACSU. Följande synpunkter kom upp:

- ACSU är en modell som utgår från de psykologiska teorierna - det innebär inte att man frågar användarna vad de vill ha
- ACSU styrs av de principer som fastställs i ISO-13407 - Human-centred design process for interactive systems
- ACSU kan definieras med hjälp av de fyra principer som specificerades av Gould & Lewis, 1983
 - * Tidig fokus på användare
 - * Tidig och kontinuerlig utvärdering av användbarhet
 - * Iterativ utveckling
 - * Integrerad utveckling
- ACSU innebär att man arbetar iterativt i nära samarbete med användare
- ACSU, eller användarstyrd utveckling, innebär att användarna själva bygger sina system. Det användarna vill göra är det som sedan kan bli betydelsefullt för företaget.

Användar- medverkan

Kan man och ska man fråga användarna vad de vill ha? Att användarna ska medverka i systemutveckling var alla relativt överens om - men hur och i vilken utsträckning lyckades man inte ena sig om. Svaren skilde sig - allt från ”a good user is a dead user” till att användarna själva ska bygga sina system.

Följande punkter diskuterades:

- Att hävda att användarna inte vet vad de vill ha är en nedlåtande attityd. Många gånger är det endast användarna som har den verksamhetskunskap som krävs när man utvecklar stödsystem. Man kan inte lämna dem ute.
- Användarnas roll i processen är oerhört viktig som drivkraft - det användarna vill ha är också bra för företaget. Systemutvecklingen måste ses som en del i verksamhetsutvecklingen. Och användarna vill ha kontroll över sin verksamhet och sina system.
- Vid användarmedverkan finns alltid risken att cementera kostigar - vi måste utveckla metoder för att undvika detta. Det är också viktigt hur användarrepresentanter väljs ut. Förändringsvilja är en viktig egenskap hos de deltagande användarna.
- Användarna känner t ex inte alltid till den nya tekniken. Det är upp till systemutvecklarna att analysera användarnas önskemål och presentera lösningsförslag.

Flera intressenter

Användarnas medverkan är inte tillräcklig. Nyckeln till ett lyckat resultat ligger i att man arbetar i team. Andra intressenter kan vara:

- beställaren - ekonomiska ramar, verksamhetsmål, etc
- utvecklaren - tekniska begränsningar
- kunden - dvs kundens kund
- säljpersonal - man kanske måste ta hänsyn till ”points of sale”. Ett exempel är vitvaror där undersökningar visar att ju fler knappar desto högre ”säljbarhet” - till förfång för användbarheten.

I många fall står de olika gruppernas önskemål och krav i konflikt med varann. Ett exempel är bank på Internet - kunden kanske önskar sig möjligheten att avsluta ett konto och flytta alla sina pengar till en annan bank. Detta står i direkt strid med bankens intresse - så vi lär nog aldrig få se den funktionen hos någon Internet-bank. För att kunna tillgodose alla målgrupper, eller ställa dem mot varann, kan man kort formulera och mäta produktmål för respektive målgrupp.

Konceptuella modeller är det enda sättet att få till stånd något som användare och övriga intressenter kan kommentera och komma med nya synpunkter på.

Relationer Grunden för att lyckas med ett iterativt arbets sätt är tillit och förtroende mellan leverantör och beställare. Man måste skapa ett åtagande som sedan är grunden för en vidareutveckling av dialogen.

Om denna tillit saknas måste man arbeta med kravspecifikationer. Ju mindre tillit desto mer detaljerade krav.

Svårigheter Vilka är då svårigheterna med ACSU? Följande frågor och problem togs upp, dock utan att vi fick några färdiga svar:

- Det är svårt att integrera användarcentrering i vanliga systemutvecklingsmodeller.
- Hur gör man när projekten är så hemliga att inte användarna kan involveras?
- Hur gör man när tiden styr?
- Hur ser man på godkännandekriterier - dvs hur kontrollerar man att leveransen är fullbordad?
- Hur gör man när det inte går bra och kunden inte är nöjd med det han fick?

Orienterad mot vad? Kan man lösa problemen genom att byta ut termen användarcentrerad? Följande alternativ diskuterades.

- användningsorienterad - vi har kompetensen och ska kunna ta fram systemen
- verksamhetscentrerad - vad vill vi uppnå i verksamheten
- problemorienterad - analysera nytta i relation till kostnad. Pekar på det faktum att det bara är vissa problem som kan beskrivas verbalt, av t ex användare. Andra kan man bara upptäcka vid observationer.
- kommunikations-/informationsorienterad - arbeta mer med kommunikationskvalitet än användbarhet (se Kommunikationskvalitet i informationssystem - Owen Eriksson, sid 29)

Arbetsrelaterade kontra upplevelsebaserade system Det finns en klar skiljelinje mellan arbetsrelaterade system och upplevelsebaserade system.

- Vid utveckling av arbetsrelaterade system är användardeltagande en av flera nyckelfaktorer för ett lyckat resultat. Enligt en undersökning (Standish Group, 1995) av över 8000 IT-projekt i 365 amerikanska företag genomfördes mindre än 20% av projekten enligt planerna. Resten förändrades, eller las ner. Två av huvudfaktorerna för framgång var användarmedverkan och ordentliga kravspecifikationer.
- Vid utveckling av upplevelsebaserade system måste man förlita sig på modeller som baseras på kunskaper om människan - kombinerat med beställarens önskemål.

Visioner, önskemål och krav

Alternativ och/eller komplement till krav Kan man arbeta på andra sätt än med kravspecifikationer? Eller ska man arbeta med komplement till krav? Följande punkter togs upp:

- Byta ut krav mot visioner och mål
- Byta ut krav mot önskemål
- Behålla krav

Visioner och mål istället för krav

Skrota kravspecen och arbeta istället med en vision på en högre nivå. Visioner handlar om målsättning och upplevelser medan kravhantering handlar om funktionella krav. Man måste analysera nuläge och börläge. Börläget blir sedan visionen som ligger som underlag för systemutvecklingen.

Man måste kommunicera ut visionen och skapa en dialog runt denna innan man diskuterar krav. Om man gör en detaljerad kravspec är denna en översättning av visionen - visionen är dold.

Önskemål istället för krav

Ska man byta ut termen krav mot önskemål istället? Detta för att "mjukgöra" processen. Önskemål är lättare att omförhandla än krav.

Att arbeta på detta sätt ställer dock höga krav på tilliten mellan parterna - det går bra så länge man kan prestera lyckade resultat. Men om en konfliktsituation uppstår är det bättre att ha en traditionell kravspec. Som leverantör behöver man kravspecen mer än beställaren - det är ett sätt att beskriva lösningen för sig själv.

Krav

Om man ändå vill arbeta med krav - hur gör man då? Uppkomsten av krav-orienterade arbetssätt och kravspecar härrör från den militär världen. Där använder man kravspecar i anbudsförfarandet för att samla in krav från användarna och kommunicera ut dem till flera leverantörer samtidigt och på samma villkor. Och för att kunna kontrollera utvecklingsprocessen.

Det är viktigt att kravspecar hålls på en hög, strikt vad-orienterad, nivå.

Iterationer - hur gör man

När är man klar?	<p>Hur vet man när man är klar i en iterativ process? Denna fråga fick följande svar:</p> <ul style="list-style-type: none">• mätbara mål• tid, kostnad, ambition - kunden styr• frågeställning
Mätbara mål	<p>Man måste ställa upp mätbara mål - när man uppnått målen är man klar. Målen ligger före kraven i processen. Krav kan formuleras först efter målen och fastställer hur man ska mäta målen för att kontrollera att de har uppnåtts.</p>
Tid, kostnad, ambition	<p>Man har en relation där man beskrivit vad man vill åstadkomma i termer av tid, kostnad och ambitionsnivå. Sedan prioriterar <i>kunden</i> hela tiden vad det är han vill ha, och avgör också när man är klar. Man måste tillåtas styra mot det som kunden vill ha vilket inte nödvändigtvis är det som står i kravspecen.</p>
Frågeställning	<p>Ofta har man bestämt sig för att arbeta iterativt men inte gjort klart för sig varför - då är risken stor att man itererar i all oändlighet. Syftet med iterationer är att man har en frågeställning där enda sättet att finna svaret är att arbeta experimentellt med prototyping och iterationer. Man måste alltså formulera sin frågeställning innan man kan avgöra när man är klar. Frågorna kan handla om t ex användbarhet, verksamhetsnytta, tekniska möjligheter.</p>
Jaha - och...	<p>Hur gör man då om tiden styr? Som i RSVs fall. Där kan man inte vänta med en leverans bara för att man inte uppnått målen.</p> <p>Naturligtvis är det mer komplext men man måste fortfarande ha mätbara mål. Har man då bara uppnått 80% av målet, har man i vart fall lärt sig något nytt. Annars hamnar man i att pengarna tar slut.....</p>
Värdebaserad prissättning	<p>Hur sätter man priset när man arbetar iterativt? Ett förslag var att arbeta med värdebaserad prissättning och undvika långa fastprisuppdrag. Vad är lösningen värd för användaren/beställaren? Detta kräver dock att kunden är mogen för den typen av prissättning - att man arbetar snarare som partners än beställare/leverantör. Goda affärsrelationer och stort förtroende är A och O.</p>

Gruppövning

Beskrivning	Gruppövningen gick ut på att i grupper om 6-8 deltagare diskutera ett av tre föreslagna praktikfall. Som grund för diskussionen fanns ett antal frågor.
Praktikfallen	Nedan beskrivs de fiktiva praktikfallen. <ol style="list-style-type: none">1. Utveckling av ett mindre gruppkonversationsverktyg för att underlätta arbetsmöten, på distans, för en liten känd användargrupp.2. Systemförnyelse av ett tidigare system med en känd, stor och geografiskt spridd användargrupp med en mängd olika arbetssätt.3. Elektroniska tjänster, okänd, mycket stor användargrupp.
Frågorna	Nedan listas frågorna som skulle tas upp under diskussionerna. <ol style="list-style-type: none">1. Hur väljer man ut användare och hur ska de involveras?2. Hur kartlägger man användarnas verkliga behov/krav (utan att cementera kostigar)?3. När och hur utformar man lösning/lösningsförslag?4. Hur hanterar man förändringar i behovet och förändrade förutsättningar under resans gång?5. Hur avgör man när man är klar?6. Vilka problem och risker kan uppstå? Vilka problem kan man inte ta hänsyn till med vald metod?
Diskussionerna	Varje praktikfall diskuterades i två olika grupper. De flesta grupperna valde att konkretisera praktikfallen. Resultaten presenteras nedan.

Diskussion

Under en kort summering och diskussion efter presentationer kom följande synpunkter och frågor upp:

- Man hade valt ett antal olika tillvägagångssätt - de flesta relativt adhoc. En aspekt som alla dock tog upp var projektledning. ”Projektledning är alltid krishantering”.
- Attityden är viktig - vi bygger för användarna och att de ska använda systemet. Kvalitetsaspekter måste vara med från början.
- Det är viktigt att identifiera de flesta kraven tidigt - innan man börjar med design. I ett tidigt skede bör man arbeta med systemegenskaper snarare än funktionalitet och utseende.
- Den stora svårigheten ligger i hoppet från krav till design. Risken med att lägga fram designidéer för tidigt är att man då fastnar i dessa idéer. Å andra sidan kan man använda pappersskisser (whiteboard) av lösningsidéer för att generera nya krav.
- Det är viktigt att komma ut med något snabbt - bättre att släppa något med begränsad, men bra, funktionalitet än att vänta för länge. Riktig verifiering på lösningen kan man bara få i skarp användning.

Praktikfall 1

Praktikfallet

Utveckling av ett mindre gruppkonversationsverktyg för att underlätta arbetsmöten, på distans, för en liten känd användargrupp.

Grupp 1

Grupp 1 valde en liten styrelse, 8-10 personer, som användargrupp. En styrelse utmärks av att personerna i den är ”kinkiga” och inte har tid.

- Hela styrelsen deltar i användargruppen. Man skapar också en artificiell referensgrupp med surrogatanvändare för tester av lösningsidéer. Man räknar dessutom med att kunna iterera två gånger mot styrelsen.
Det är också viktigt att tänka på supportroller, t ex sekreterare.
- För att kartlägga behovet kan man titta på existerande verktyg. Man kan också studera faktiska möten. Efter mötena intervjuar man deltagarna. Studierna genomförs av en person med kognitionsvetenskaplig kompetens.
- Man utformar lösningsidéer och testar dessa på referensgruppen. När man kommit en bit på väg, testar man också på ett riktigt styrelsemöte. Feedback från testerna med styrelsen återkopplas till referensgruppen för att man ska kunna iterera vidare mot den.
- En risk med det här arbetssättet är att man har hunnit ganska långt i utvecklingen innan man kan stämma av med styrelsen att man är på rätt väg.

Grupp 2

Grupp 2 valde att diskutera runt en tänkt utveckling av en kommersiell produkt för gruppkonversation för en specificerad målgrupp, t ex systemutvecklare.

- Man kan försöka etablera partners för samarbete - t ex genom att kontakta intresseföreningar för målgruppen. Med dessa kan man diskutera hur man kan använda verktyget och visa på möjligheter.
- Man definierar en affärsidé och visioner för användning. Sedan kan man använda en testplattform som man lägger ut för tester med användare. Man använder intervjuer, observationer och feedback för att studera användningen.
- God kontakt med användarna under utvecklingen är viktigt - och att användarna är intresserade av utvecklingen. Brist på motivation kan äventyra projektet.
- Under arbetets gång stämmer man av mot mål och visioner. Klar är man när man är eniga enligt någon metanivå (tid, ambition, kostnad).
- Risker i den här typen av projekt kan vara kompetensbrist eller otydliga målformuleringar.

Praktikfall 2

Praktikfallet

Systemförnyelse av ett tidigare system med en känd, stor och geografiskt spridd användargrupp med en mängd olika arbetssätt.

Grupp 3

Grupp 3 valde ett resebyrå-system som exempel.

- Man börjar med att ställa frågan varför man arbetar på ett flertal olika sätt.
- Man diskuterar olika mål med olika målgrupper
 - * strategiska mål - företagsledningen
 - * taktiska mål - resebyråtjänstemän
 - * operativa mål - kunden, biljettköparen
- För att kartlägga behovet identifierar man de kontor där man har problem. Man samlar in information med hjälp av intervjuer och enkäter.
- Att utforma lösningen är en kontinuerlig process. Man måste prioritera - vad ger mest nytta/nöje till minst kostnad. Här menar man både upplevd och verklig nytta/nöje. Vid förändringar måste nya krav prioriteras i förhållande till de gamla.
- Man kan avgöra när man är klar genom att mäta - t ex om man har ett konkret problem med biljettförsäljning och vill minska den genomsnittliga tiden från en halvtimme till tio minuter.
- Vad gäller risker är allting oförutsett. Man måste prioritera mera.

Grupp 4

Grupp 4 valde att diskutera ett 20 år gammalt biljettbokningssystem som ingen vågat röra de senaste 3 åren.

- Man väljer att skapa användargrupper med olika profiler. Det är viktigt att ta fram resurskontrakt med användarna - så att man verkligen har tillgång till dem under utvecklingen. Man bör också verifiera resultaten genom att intervjua slutkunder - dvs den person som köper biljetten.
- Vid en systemuppdatering bör man koncentrera sig på det nuvarande systemet - med tillägg av positiva egenskaper och bortdrag av negativa.
- Eventuella konfliktämnen bör tas upp och prioriteras tidigt - t ex att försäljningspersonal kommer att sägas upp på grund av förändringar i försäljningsprocessen.
- En risk vid systemuppdateringar är att man missar de positiva egenskaperna i det gamla systemet.

Praktikfall 3

Praktikfallet

Utveckling av elektroniska tjänster för en okänd, mycket stor användargrupp.

Grupp 5

Grupp 5 valde att diskutera bank på Internet.

- I den här typen av utveckling är det viktigt att ringa in målgruppen - dvs egenskaper hos potentiella användare, t ex personliga egenskaper, bankbeteende. Man kan ringa in typindivider samt typsituationer för användning av respektive tjänst. Gränssnittet måste tillgodose behoven hos alla typanvändare.
- Det finns en tydlig målkonflikt - bankens affärsmål kontra användarnas. Det är viktigt att tydliggöra denna konflikt tidigt.
- Man vill testa lösningsidéer på riktiga användare - t ex genom att fånga in dem i bankkön. Vid testerna använder man simulerade användningssituationer (framtdsverkstad).
- Genom att formulera och mäta användbarhetsmål kan man avgöra när man är klar. Risker finns dock att man missar kvalitativa mål.

Grupp 6

Grupp 6 valde att diskutera en helt ny tjänst för samåkning - man har 1 miljon kronor och 6 veckor på sig att bli klara.

- Man börjar genom att genomföra en sammanhangsanalys (context of use analysis). Detta ger de mål man vill uppnå.
- Alla aktörer måste vara med - beställare, användare, etc. Målen man identifierar kan t ex vara mindre biltrafik. Tre huvudanvändargrupper identifieras.
- Med dessa genomför man en behovsanalys - t ex genom fokusgrupper. En önskelista tas fram och från den kan ett antal kärnkrav fokuseras.
- Det är viktigt att kravarbetet är relativt klart innan man börjar designa. Sedan tar en iterativ process mellan krav och design vid. Man kan t ex iterera enkla designförslag med användargrupperna.
- Den stora svårigheten ligger i hoppet mellan krav och design - det är det hoppet som avgör huruvida produkten blir bra.

Paneldebatt 2

Deltagare

Följande personer deltog i panelen:

- Ordförande: Sten-Erik Öhlund, SISU
- Deltagare
 - * Torbjörn Näslund, Astra Arcus
 - * Lars Dahlbom, Redina Informatik
 - * Owen Eriksson, Högskolan Dalarna och forskningsgruppen VITS
 - * Hans Marmolin, UI Design

Frågeställningar

Varje deltagare hade förberett en kort presentation:

- Behov kontra krav - Torbjörn Näslund
- ACSU och kravhantering i offertprocessen - Lars Dahlbom
- Kommunikationskvalitet - Owen Eriksson
- Designprinciper för nya grupper och tjänster - Hans Marmolin

Behov kontra krav - Torbjörn Näslund

Krav avviker från behov	<p>Behoven är det som ”verkliga” behövs. Tanken är att kraven ska avspegla dessa behov, men i och med specificeringen av krav introducerar man systematiska avvikelser enligt följande:</p> <ul style="list-style-type: none">• Krav är <i>verbaliserade</i> behov• Krav är verbaliserade behov <i>så som man tror innan projektet knappt startat</i>• Krav är verbaliserade behov så som man tror innan projektet knappt startat, och uttryckta som <i>singulära, mätbara, krav</i>. <p>Varje punkt här gör att kraven avviker från de verkliga behoven, att kravspecifikationen är en mer eller mindre bra approximation.</p>
Varför	<p>Anledningen till att man ändå gör dessa avvikelser är att man då får avgörbarhet, stabilitet och personoberoende så att man vet vad man ska göra, när man är klar och hur mycket det kostar. Iterativ, användbarhetsorienterad utveckling står i motsats till detta.</p>
När	<p>Kravdriven utveckling fungerar när behoven är väl förstådda, verksamheten är stabil och man har ett gemensamt språk. Det fungerar inte vid hög osäkerhet, när man inte kan förklara behoven och målet är rörligt.</p> <p>Då kan man välja att ha kontroll över processen men inte uppfylla behoven, eller uppfylla behoven men med en okontrollerad process. Eller att kryssa sig fram mellan ytterligheterna, vilket ställer stora krav på kompetensen.</p>
Krav på processen: Exempel	<p>Torbjörn tog upp två exempel med externa krav på en kontrollerad process samtidigt som osäkerheten är mycket hög. Detta innebär att kraven på kontroll motverkar det egentliga syftet att höja kvaliteten på resultatet. Man har formella bevis på att processen fungerar men dåliga eller inga resultat.</p> <p><i>Exempel 1: Militär verksamhet</i> Det första exemplet är taget från militär verksamhet - vilket innebär statlig upphandling och fastprisåtaganden. En kravspecifikation togs fram men den speglade inte alls visionen hos de verksamhetsrepresentanter som deltog. På grund av detta drev kravbilden, konflikter uppstod och man hade svårt att få ihop alla delar i projektet. Resultatet blev en oanvändbar produkt.</p> <p><i>Exempel 2: Läkemedelsforskning</i> Det andra exemplet är taget från läkemedelsindustrin där läkemedelsverken ställer höga krav på en väl kontrollerad utvecklingsprocess. Acceptanstesten måste spegla kravbilden som tas fram tidigt i processen. Man kan inte experimentera med systemutvecklingsprocesser och kombinera dem.</p>

Vad gör man?

Vad gör man när man hamnar i sådana här situationer? Bluffar - dvs fixar till dokumentation och kravspecifikationer i efterhand, försöker få undantag, tar en risk eller dubbelarbetar?

Att låta bli att frysa kravspecen är inget alternativ eftersom kraven på processen kommer utifrån. I militär verksamhet kan kravspecen vara en produkt av 100-tals personers arbete under 10 års tid. Detta kan man inte ändra på.

ACSU och kravhantering i offertprocessen - Lars Dahlbom

Inledning	Lars Dahlbom tog upp ett antal frågor och möjligheter vad gäller ACSU i offertprocessen. Tyvärr fanns det inte tillräckligt med tid att diskutera alla.
Offentlig upphandling	<p>Är ACSU ett realistiskt alternativ i offentlig upphandling?</p> <p>Nej, enligt RSVs representant, med flera. Inte om det innebär att man delar ut en påse med pengar utan att få garantier om resultatet. Man har valet att arbeta normreglerat - dvs man koncentrerar sig på att följa reglerna istället för på resultatet - eller att bryta mot reglerna.</p> <p>Ett sätt att arbeta i två steg beskrevs - först enligt den frusna kravspecen, där resultatet garanterat kommer att bli misslyckat men inför nästa fas har man gjort sig oumbärlig. Moraliskt tveksamt och förbundet med stora risker.</p>
Prissättning	<p>Kan man arbeta användarcentrerat i ett fastprisuppdrag? Finns det andra sätt att prissätta?</p> <ul style="list-style-type: none">• kundvärde snarare än kostnad och tid - men pris och tid styr snarare än kvalitet• pris per iteration <p>Alternativ prissättning kräver dock att man har en etablerad relation med kunden. Man måste utbilda kunden.</p>
Nya kunder	<p>Kan man sälja in ACSU hos nya kunder? Hur etablerar man en relation med kunden? ACSU kräver en god relation till kunden - att kunden har ett gott förtroende för leverantören. I brist på förtroende måste man garantera resultatet - utfästelser istället för förtroende.</p> <p>Det är otroligt viktigt att känna någon hos kunden. Lars Dahlbom gick så långt som att ge upp kunder där man inte har någon som helst relation. Då offererar man inte.</p>
Planering	<p>Hur planerar man ACSU-projekt? Hur arbetar man med ACSU i stora projekt? Dessa frågor hade redan berörts något tidigare under dagen. Svaret var att dela upp projektet - om det låter sig göras. Det är också viktigt att komma med tidigt i utvecklingskedjan.</p>
Befintlig kravspec	<p>Vad gör man med den kravspec som föreligger i anbudsunderlaget? Den kan vara en täckmantel, dvs att kunden egentligen redan bestämt sig för en leverantör, men vill ta in kontrollofferter, eller är tvingad av lagen att gå ut med en anbudsförfrågan.</p> <p>Om kravspecen är fryst bör man inte arbeta användarcentrerat - det är meningslöst.</p>

Kommunikationskvalitet i informationssystem - Owen Eriksson

Inledning	Owen Eriksson diskuterade begreppet kommunikationskvalitet i informationssystem och hur fokusering på detta begrepp kan påverka utvecklingsprocessen.
Vad är kommunikationskvalitet	<p>Kommunikationskvalitet definierades på följande sätt:</p> <p>”Kommunikation som är <i>förståelig</i> och <i>acceptabel</i> och bidrar till att etablera interpersonella relationer som bygger på samförstånd.”</p> <p>Mottagaren måste förstå och acceptera kommunikationen för att våga ingå i ett åtagande. Owen ställde upp följande kriterier för kvalitet i kommunikation.</p> <ul style="list-style-type: none">• tydligt informationsinnehåll• tydlig handlingsaspekt• förståelse• trovärdighet• möjlighet till kontroll/kritik
Kommunikation är social	<p>Kommunikation är social - i betydelsen att det krävs minst två aktörer för att kommunicera. Kommunikation bygger på överenskommelser, åtaganden och aktörsrelationer.</p> <p>Vi handlar när vi kommunicerar. Vi utför viktiga handlingar - erbjudanden, frågor, löften - som skapar åtaganden som är styrande för framtida handlingar.</p> <p>Kommunikation kan alltså bidra till att skapa affärsrelationer och informationssystem kan stödja aktörerna i detta, t ex ett säljstödssystem för bilhandlare. Vi använder systemen för att kommunicera. I och med att kommunikationen i grunden är social är också informationen i våra system i grunden social.</p>
Ett kommunikativt perspektiv på IS	<p>Informationssystem kan hjälpa sändaren i rollen som utförare med att registrera, lagra, förändra, ta bort, utföra informationsbehandling, sända och presentera meddelanden.</p> <p>Informationssystem kan hjälpa mottagaren (uttolkaren) med att ta emot, söka, presentera, lagra och ta bort meddelanden.</p>

- Frågeställningar Owen tog upp följande frågeställningar:
- Vi går från intern till extern användning av informationssystem. Detta innebär att vi går:
 - * från information till kommunikation
 - * från informationskvalitet till kommunikationskvalitet
 - * från systemrationalitet till kommunikativ rationalitet
 - Användarbegreppet är problematiskt
 - * det är för snävt
 - * det är svårt att identifiera användaren
 - Användarnytta och kravhantering är problematiskt
 - * vad är nytta?
 - * nytta för vem?
 - * vems krav?

Owen menade också att användbarhetsbegreppet och andra kvalitetsbegrepp är för snäva - de fokuserar inte på kommunikationsaspekten.

Förtroende och tillit Förtroende och tillit är viktigt för hög kommunikationskvalitet. Hur skapar vi tillit när vi lägger ut tjänster på webben?

Det finns exempel där man misslyckats - ett bokningssystem där de få bokningar man fick in ändå gjordes per telefon. Systemet lyckades inte skapa förtroende och tillit hos mottagaren - något brast i kommunikationskvaliteten.

Exempel: Säljstödssystem för bilhandel Owen beskrev ett exempel från ett säljstödssystem för en bilhandel. Kundens och säljarens intressen kan vara i konflikt med varann. Det räcker därför inte att enbart utgå från användarnytta (i detta fall försäljarens nytta). När man utvecklar och utvärderar informationssystem måste man istället utgå från interaktion och kommunikation som sker mellan försäljare (användare), system och kund. Det är denna kombination som styr kommunikationskvaliteten. En kvalitet som är viktig både för kunden och för försäljaren.

Designprinciper för nya grupper och tjänster - Hans Marmolin

- Inledning** Hans Marmolin diskuterade designprinciper vid utveckling av tjänster inom digital TV - informations-, underhållnings- och köptjänster.
- Traditionella system** Vid utveckling av traditionella system arbetar man ofta på följande sätt:
- Mål: användbarhet
 - Metodik: krav, uppgift, lösning
 - Process: faser, specifikationer, iterationer, konkurrerande metoder (antingen eller)
- Nya tjänster** Vid utveckling av nya tjänster måste man ändra sina arbetsätt:
- Mål: användbarhet och attraktivitet. Användarna måste vilja använda de nya tjänsterna. Dessa två mål står ofta i konflikt med varann.
 - Metodik: koncept, situation, realisering - krav uppstår runt konceptet, den sociala situationen blir viktigare än uppgiften. Man måste realisera stödet i en social situation.
 - Process: kreativitet, dynamik, flexibilitet, samverkande metoder (kombination) och kompetenser.
- Vad finns kvar av användarcentrering** Man kan aldrig bortse från att det fortfarande är individer som ska använda tjänsten - men kan de gamla metoderna överföras till tjänsteutveckling?
- Ingrid Ottersten menade att användarcentrering har mer att göra med en tankemodell och att det finns plats för denna modell i tjänsteutveckling (upplevelsebaserade system).
- Kulturkonflikter** Ett problem är kulturkonflikter. Personer med estetisk bakgrund ska samverka med tekniker, beteendevetare och användare.
- Innehållsproducenterna har en kommunikativ roll men interaktionsproducenterna producerar den traditionella designen. Dessa måste samarbeta väldigt nära. Dock är vår utbildning innehålls- *eller* interaktionsbaserad, men man kan inte separera.
- Kommunikationen** Kommunikationsaspekten måste läggas till den traditionella användbarhetsaspekten.

Referenser

Ref 1

Utveckling av ett utvärderingsinstrument (Prodevo) - ett hjälpmedel för att utveckla tjänste-och produktutveckling med IT. Öhlund, Licentiatsuppsats, Data och Systemvetenskap, SU/KTH, December 1997, No. 97-020, ISSN 1101-8526

Ref 2

Software Requirements. Brackett, J.W. (1990). Technical Report SEI-CM-19-1.2, Software Engineering Institute.

Ref 3

Challenges in Requirements Engineering, Janis Bubenko Jr, abstract of invited talk at RE'95, Second IEEE International Symposium on Requirements Engineering, 27-9 March, 1995, York, England.

Ref 4

Designing for usability - key principles and what designers think. Gould, J. och Lewis, C. (1985), Communications of the ACM, 28, 3, 300-311.

Definitioner

- Concurrent Engineering
”Concurrent Engineering is a systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, schedule, and user requirement.”
CERC (Concurrent Engineering Research Center) [Cleetus 1992]
- Integrerad Produktutveckling
Ett synonymt begrepp till Concurrent Engineering är Integrerad produktutveckling (IPU) [Andreasen och Hein 1987]. En pionjär inom detta område var Fredy Olsson vid Lunds universitet. Hans modell för Integrerad produktutveckling finns redovisad i *Integrerad produktutveckling – arbetsmodell Sveriges Mekanförbund*, (1985). [Olsson, Carlqvist et al 1985]¹
- Lean Production
De japanska bilföretagen har utvecklat en helt ny produktionsteknik och nya sätt att utveckla produkter som skilde sig från de banbrytande tekniker som Henry Ford införde på 20-talet i sina fabriker i USA. Forskarna i IMVP [Womack, Jones et al 1990] benämner den nya produktionstekniken inom bilindustrin för Lean Production

Deltagare

Torbjörn Näslund	Astra Arcus	Torbjorn.Naslund@arcus.se.astra.com
Magnus Wånge	Astrakan SU	mawan@astrakan.se
Ann Lantz	CID	alz@nada.kth.se
Fredrik Winberg	CID	fredrikw@nada.kth.se
Peter Petrov	CID	ppetrov@nada.kth.se
Åke Walldius	CID	aakew@nada.kth.se
Eva Hallman	CAP Gemini	eva.hallman@capgemini.se
Gunilla Stoberski	CAP Gemini	gunilla.stoberski@capgemini.se
Bengt Göransson	Avd. för MDI, Uppsala Universitet	Bengt.Goransson@hci.uu.se
Erik Borälvs	Avd. För MDI, Uppsala Universitet	erik.boralv@hci.uu.se
Jan Gulliksen	Avd. För MDI, Uppsala Universitet/CID	jan.gulliksen@hci.uu.se
Jens Jonasson	D-linjen, KTH	d92-jjo@nada.kth.se
Anna Norman	Enator	anna.norman@enator.se
Lena Andersson	Enator	lena.l.andersson@enator.se
Måns Reventberg	Enator	mans.reventberg@enator.se
Inger Boivie	Enator	inger.boivie@enator.se
Nils-Erik Gustafsson	Ericsson Utvecklings AB	Nils-Erik.Gustafsson@uab.ericsson.se
Joachim Karlsson	Focal Point	Joachim.Karlsson@focalpoint.se
Owen Ericsson	Högskolan Dalarna	oer@blg.du.se
Peo Orvendal	Ineo Konsult	Peo@ineo.se
Ingrid Ottersten	LinnéData	Ingrid.Ottersten@lig.linnedata.se
Maj Johansson	LinnéData	maj.johansson@lig.linnedata.se
Torbjörn Lind	LO	Tojje1@algonet.se
Magnus Rönnäng	Meringue Systems/Folkuniversitet	ronnang@etek.chalmers.se
Nigel Claridge	Nomos Management	nigel.claridge@nomos.se
Tomas Berns	Nomos Management	tomas.berns@nomos.se
Dan Sjögren	Nutek	dan.sjogren@nutek.se

Deltagarlista, forts

Lars Dahlbom	Redina Informatik	lars.dahlbom@redina.se
Kjellåke Henriksson	RSV	kjellake.henriksson@rsv.rsv.se
Sigrid Schmidt	RSV DataService	sigrid.schmidt@ds.rsv.se
Malin Pettersson	RSV DataService	malin.pettersson@ds.rsv.se
Ulla Rohlen-Johansson	SAS Data Group	ulla.rohlen-johansson@sas.se
Sten-Erik Öhlund	SISU	ster@sisu.se
Anders Jansson	Telia/Avd. för MDI, Uppsala Universitet	anders.k.jansson@telia.se
Carl-Erik Löfgren	Telia	carl-erik.e.lofgren@telia.se
Ole L Larsson	Telia	Ole.L.Larsson@telia.se
Olle S Bergström	Telia	Olle.S.Bergstrom@telia.se
Patrik J Skoglund	Telia	Patrik.J.Skoglund@telia.se
Hans Marmolin	UI Design	Hans.Marmolin@uidesign.se
Cecilia Katzeff	WM-data/SISU	cekat@wmdata.com
Pia Strandberg	AB Sandvik Information Systems	pia.strandberg@sandvik.com