

GÖTEBORGS UNIVERSITET
Institutionen för informatik

INTERNATIONALISERING -
problem vid framställning av programvara till Japan och övriga världen

av Ulrika Ernestål

C-UPPSATS PÅ ADB-PROGRAMMET
Höstterminen 1997
Handledare: Agneta Ranerup

INTERNATIONALISERING - problem vid framställning av programvara till Japan och övriga världen

Ulrika Ernestål

1. Abstrakt

Jag har i denna rapport försökt ge en inblick i vilka problem som kan uppstå när ett företag vill sälja produkter som mjukvara och tillhörande dokumentation på den internationella marknaden, och då främst på den japanska. Syftet med rapporten var att undersöka hur mycket extra arbete, tid och ändringar i systemen som krävs för att en mjukvara ska kunna förstås och användas av människor i ett annat land. Ett av delsyftena var att läsa delar av det litteraturutbud som finns på den japanska och svenska marknaden för att få en överblick över problematiken vid internationalisering av dataprojekter. Ett annat delsyfte var att under en vistelse i Japan ta del av den japanska kulturen och försöka få en inblick i deras marknad genom att bl.a. samtala med personer på NEC och genom att besöka en datormässa i Tokyo. Mina slutsatser var att det ofta krävs mycket arbete, pengar och tid för att sälja en produkt internationellt och kanske speciellt på den japanska marknaden då det krävs en del extra arbete som t.ex. att anpassa programmet för deras teckensystem och att omvandla gränssnitten så att dessa passar de japanska användarna. Beroende på hur stor del av en produkt som ska ändras till inköparnas språk och specifika karaktärsdrag så krävs olika insatser av produktutvecklarna.

Allt fler företag väljer numera att vidga sin verksamhet genom att sälja sina produkter till andra länder runt om i världen. För många mindre företag kan detta vara ett stort steg att ta och det kan vara svårt att få en uppfattning om hur mycket pengar och arbetsinsatser som krävs för att en produkt ska accepteras i länder där andra språk talas eller där andra kulturella förutsättningar finns. En mycket väsentlig fråga är om programmet ska säljas i en ursprunglig (som ofta är en engelsk) version eller om det lönar sig att helt ändra användargränssnittet och dokumentationen med mera till förmån för den speciella målgruppen. Jag hade för avsikt att i denna rapport ge en överblick över vilka problem som kan tänkas dyka upp i samband med internationaliseringsprocessen. Det finns sedan tidigare en hel del litteratur inom området som tar upp problemet från både användarnas och tillverkarnas perspektiv.

2. Metod

För att framställa denna rapport har jag främst använt mig av den litteratur som jag hittat inom ämnet internationalisering av mjukvara och dokumentation som ska säljas på den globala marknaden. Vidare har jag tillbringat ett par månader i Japan och under den tiden försökte jag ta del av deras kultur och vardagsliv. Dessutom har jag testat ett japanskt ordbehandlingsprogram i Macintoshmiljö för att försöka få en inblick i hur deras mjuk- och hårdvaror fungerar. Som komplettering till detta har jag talat med anställda på NEC i Tokyo (Aston Bridgman på Public Relation Division) och Osaka (Toshiyuki Asahi som är chefsforskare på C & C Media Research Laboratories) i Japan för att få en inblick i deras nuvarande och eventuella framtida marknad vad angår gränssnittsforskning men även vad gäller den japanska datamarknaden i stort.

3. Problembeskrivning

Anledningen till att jag valt internationalisering som ämne för denna rapport är att fler och fler företag berörs av problematiken som ofta kan uppstå vid internationaliseringsprocessen och syftet är att försöka visa ungefär hur mycket som behövs ändras i en dataproduct för att den ska vara gångbar på den internationella marknaden. Hur mycket jobb och tid går det egentligen åt för att producera ett bra gränssnitt och medföljande dokumentation?

Vi har under de senaste åren fått uppleva en otroligt snabb utveckling inom dataindustrin som betytt mycket för många dataföretag runt om i världen. För många har datoriseringen betytt att deras företag gått från att ha ett fåtal anställda och en liten budget till att bli ett multinationellt storföretag med tusentals anställda medan andra har valt att hålla sitt företag. Jag tror att det är av stor vikt att personer inom databranschen inser att det kan krävas en hel del arbete för att sälja en produkt på en marknad som inte helt liknar hemmamarknaden men även för studenter kan det vara bra att få en uppfattning om hur användare på olika platser uppfattar olika produkter.

Till att börja med vill jag visa på att det inom mjukvarutillverkning, oavsett vem produkterna ska säljas till, är viktigt att producenterna har en viss uppfattning om hur människan och datorerna samverkar. Det finns grundläggande faser i denna interaktion och en viktig fråga är om datoranvändare världen över skiljer sig åt vad gäller grundläggande kognitiva fakta. Eftersom arbetet främst är inriktat på den japanska marknaden kommer sekvenser med beskrivningar om den japanska marknaden och dess användare att finnas i rapporten.

Huvuddelen av rapporten ägnas dock åt problematiken med internationaliseringen som finns inom ett flertal områden. Förutom förklaringar på vad termen internationalisering betyder så ges även exempel på hur mycket av en produkt som bör internationaliseras. Är det lönsamt att översätta hela produkten inklusive gränssnitt och dokumentation eller ska man satsa på att bara översätta delar av produkten?

Vidare beskrivs problematiken med internationalisering vid själva framställandet av en produkt. Det är inte bara de mest synliga skillnaderna som att användarna har olika alfabet, annorlunda datumformat, sorteringsordningar osv som kan ställa till problem utan även mer djupgående kulturella frågor som om användarna uppfattar gränssnitten på samma sätt världen över eller om t.ex. ikoner och färger ska ändras helt för att produkten ska godtas. Hur djupgående förändringar behövs egentligen och hur kontrolleras att användarna förstår produkten på ett korrekt vis? Är förändringar på alla nivåer av internationaliserings-problematiken lika viktiga?

Frågor som berör vilka faktorer som bör tas hänsyn till när produkter ska säljas just till Japan och om användarna där skiljer sig från användarna i t.ex. Sverige tas även upp. Är det mest japanernas skriftspråk som ställer till svårigheter eller kan också mer djupgående skillnader som kulturell bakgrund vara avgörande?

Delsyften med rapporten är att förklara dessa frågor och jag tar hjälp av material som jag fått vid intervjuer i Japan samt av de kunskaper jag har inom det japanska språket och av de observationer jag gjort under min vistelse i Japan. Jag tar naturligtvis även hjälp av delar av den information som jag fått från studerad litteratur när jag besvarar dessa och övriga frågor som ställs i rapporten. Med hjälp av den information som jag fått fram från detta så kommer jag att dra slutsatser och diskutera de fakta angående internationalisering och de problem som berör detta som jag kommit fram till i resultatdelen av rapporten.

4. Resultat

Nedan presenteras de resultat som framkom under arbetets tid.

4.1 Allmänt om datamarknaden och datoranvändande i Japan

I Japan tillverkas och säljs en enorm mängd datorutrustning. NEC har de senaste åren varit det ledande märket på marknaden och ungefär hälften av alla datorer som säljs kommer från NEC. Tyvärr finns det enligt Aston Bridgman på NEC i Tokyo problem då NECs produkter inte är kompatibla med många andra produkter som säljs på marknaden.

Enligt Toshiyuki Asahi som är chefsforskare inom mediadesign på NEC i Osaka handhar företaget inte frågor som berör internationalisering så värdefull information angående detta kunde han inte ge men han bistod emellertid med data som handlar om hur långt deras forskning för tillfället kommit inom bl.a. ämnet människa- datorinteraktion.

I Japan utbildas inte så många programmerare som den japanska marknaden kräver. Många företag har p.g.a. detta börjat med internutbildningar för att minska efterfrågan lite. Utöver detta använder många företag sig av utländska mjukvarukonstruktörer som ofta snabbt och enkelt får visum utfärdade. Många tar chansen att åka till Japan under en period då lönerna är relativt höga.

Japanskan lånar många tekniska termer från engelskan men dessa ord betyder inte alltid samma sak som på originalspråket. Ett exempel som del Galdo och Nielsen (1996) tar upp i *International User Interfaces* är ordet copy som på engelska betyder att kopiera papper, kopiera en textsträng (som i copy/paste) eller att läsa från ett papper och sedan skriva in det lästa via tangentbord in i en dator. Det japanska copy (eller kopii som de stavar ordet) betyder bara de första två sakerna.

Trots att japanerna läser engelska i minst 6 år i skolan så har många svårt för att prata engelska. (Vanligtvis förstår de skriven engelska hyfsat.) Detta innebär att japanerna ofta vill ha mjukvara på sitt eget språk och kraven på att densamma ska vara korrekt är ofta rätt höga. I *Internationalization - Developing Software for Global Markets* (Luong, Lok, Taylor, Driscoll, 1995) så beskrivs det även att det finns många mindre utvecklare av mjukvara i Japan och marknaden är väldigt serviceinriktad och detta innebär att de japanska användarna väntar sig mycket av sina produkter. Eftersom marknaden dessutom är användarorienterad så är det viktigt med lättanvända produkter. Japanerna är vidare ofta villiga att betala mer för att få ett program som är så perfekt som möjligt.

4.2 Människa- datorinteraktion

Till en början tänkte jag redogöra lite kort för det mest grundläggande fakta inom datoranvändandet och det är den människa-datorinteraktion som ständigt sker utan att användarna över huvud taget reflekterar över det.

Enligt Toshiyuki Asahi på NEC så kommer det ständigt nya hjälpmedel som förbättrar och förenklar kommunikationen mellan datorer och människor. NEC har kommit en bit på väg vad gäller exempelvis röst- och skrivstilsigenkänning då de framställt datorer som klarar av att förstå en hel del ord och tecken, men Asahi tror att det fortfarande kommer att dröja innan dessa hjälpmedel kan vara fullt utvecklade och användbara.

Nedan beskrivs olika steg i perceptionsprocessen som sker när en människa använder en dator och dess mjukvara. I *International User Interfaces* (del Galdo et al., 1996) delas människa-datorinteraktion in i (A) *listening mode* och (B) *speaking mode* som i sin tur indelas i ett antal olika steg. Hur människor hanterar stegen i processen skiljer sig från person till person, och det går även att utrona mer omfattande skillnader som gör att det går att dra gränser mellan användare i olika kulturer. Detta ger naturligtvis en stereotyp bild av användarna men för att få en uppfattning om vad som sker i interaktionen krävs vissa generaliseringar. Jag har tagit upp de japanska användarna som exempel nedan i den mån jag hittat information angående dessa i del Galdo et al. (1996).

(A) *Listening mode*

Här undersöks vad som händer med användaren när information från ett program presenterats på en skärm. Listening mode indelas i tre faser.

1. Perceptionsfasen

I denna fas märker användaren att något förändrats på skärmen och här uppmärksammas intuitiv information som färg och form.

Denna fas har inte så stor kulturell skillnad då perceptionen inte beror på användarens språk eller på den kulturella bakgrunden utan på individernas sensoriska känslighet. I *Människa-datorinteraktion - Ett psykologiskt perspektiv* skriver Allwood (1991) att ögat och örat räknas till människans sensoriska buffertar och om för mycket information visas på en skärmbild samtidigt kan det störa den information som redan finns inkodad i de sensoriska buffertarna. Allwood (1991) menar vidare att det är bra att tänka på att människor i allmänhet är duktiga på att minnas information i visuell form och att det kan vara fördelaktigt att använda bilder och ikoner för att förmedla information till användare som inte har specialkunskaper inom ett område.

2. Associationsfasen

Användaren associerar det de mottagit i perceptionsfasen med verkligheten. Exempelvis kan individen se att en text är skriven på svenska och den är ett felmeddelande.

Denna fas är mer kulturkänslig än den första men problemen som uppstår kan vanligtvis hanteras med enklare medel som t.ex. ändringar av färger, språk, tids- och datumformat och ikoner. Här inkluderas problem som att japanska tecken kräver mer plats vid lagring och presentation på skärmen än engelska, färger betyder olika saker i olika länder (t.ex. lila betyder värdighet och adlighet i Japan medan det i det gamla Grekland betydde död och ondska). Textriktning skiljer sig också om man jämför Sverige och Japan då japanerna skriver från höger till vänster och uppifrån och ner och ibland kräver användarna dessutom att texten ska kunna skrivas på det sätt som vi är vana med i Sverige. Användande av metaforer kan även underlätta i många fall.

3. Resonerandefasen

Under denna fas resonerar användaren med sig själv angående den presenterade informationen.

Denna fas berörs mest av de kulturella skillnaderna. Formellt resonemang som matematiska formler och logik är rätt universella medan kognitivt resonemang beror på de sociala normer och den kulturella bakgrund som användaren har. Allwood (1991) förklarar att "Individens förståelse är summan av de bakgrundserfarenheter individen för med sig till den nya situationen." (s.32). Vanligtvis skiljer sig bakgrundserfarenheterna mindre mellan folk från ett och samma land än om man jämför med människor från andra länder och kulturer.

(B) Speaking mode

Detta är instruktioner som skickas från användaren till datorn genom exempelvis tangentbord eller touchscreen och händelseföljden i speaking mode kan indelas i fyra faser.

1. Beredandefasen

I denna fas identifierar användarna vad de kan göra med den information som visas på skärmen. De kollar exempelvis var de kan fylla i information och om de kan klicka på någon knapp eller ikon.

Den kulturella påverkan på beredandefasen är inte så stor då det helt beror på hur en användare förstår och uppfattar skärmbilder i allmänhet, och detta skiljer sig inte så markant från land till land.

2. Tillämplighetskontroll

Här utvärderar användaren sitt handlingsval och frågar sig om vald händelse kommer att få datorn att göra det som avsågs till en början.

Användarnas inställning när de kontrollerar dessa saker skiljer sig från kultur till kultur. I Japan använder man sig inte av trial and error i samma utsträckning som t.ex. är vanligt i Sverige utan där läses manualer och annan dokumentation i allmänhet mer noggrant. Svenskarna provar sig hellre fram till olika lösningar.

3. Utförande med förväntningar

Detta är fasen där användaren utför den valda händelsen. Hon klickar t.ex. på en knapp.

För kulturella skillnader inom detta område finns inga empiriska bevis men det är allmänt vedertaget att vissa människor vill att förloppen i ett arbete ska ske parallellt medan andra föredrar att arbeta sekventiellt. Japanerna har en förkärlek för att först identifiera önskat objekt för att sedan bestämma vilken händelse som bör användas för detta objekt medan sekvenserna oftare går mer parallellt i västerländska kulturer.

4. Bekräftandefasen

Efter att ha genomgått de tre tidigare faserna ska användaren slutligen bekräfta att den valda händelsen utfört det som avsågs till en början.

Viktigt här är hur snabbt det system som används är och hur användaren uppfattar systemet men även hur stort tålamod som visas vid användande är väsentligt. I Japan är

det viktigt att systemen svarar snabbt men japanerna visar ofta större tålamod när det gäller att uppnå långsiktiga mål. De avsätter som sagt var även längre tid till manualläsning innan de börjar använda en produkt. I andra länder kan ett lite långsammare system godtas till förmån för andra faktorer som t.ex. ett i övrigt lättarbetat program och bra arbetsmiljö.

4.3 Internationalisering och lokalisering

I det här avsnittet kommer jag att förklara begreppen internationalisering och lokalisering och distinktionen mellan dessa två. Vidare kommer jag kortfattat att förklara varför internationalisering och lokaliseringsprocesserna är viktiga.

Internationalisering har med åren blivit allt viktigare då exempelvis mer än hälften av de större Amerikanska mjukvaruföretagen säljer sina produkter på världsmarknaden (Luong et al., 1995). De flesta programmen görs för engelsktalande men med den växande marknaden så ökar kraven på mjukvaran successivt och användarna kräver i fler och fler fall att produkten ska utvecklas på användarnas språk och med deras egna standards.

Många som är nya inom området internationalisering tar för givet att när den ursprungliga produkten är testad och klar på hemmamarknaden så behövs bara en sista översättning för att produkten ska vara användbar i andra länder, men det kan vara mycket svårt och tidskrävande att t.ex. ändra ikoner som passar användarna, ändra tids- och datumformat och att tillverka felmeddelanden som användarna inte uppfattar som hotande och skrämmande.

I Internationalization - Developing Software for Global Markets (Luong et al., 1995) görs en distinktion mellan internationalisering och lokalisering. Inom internationaliseringsprocessen tas strukturen och stommen fram. Med hjälp av denna stomme sker lokaliseringsprocessen mer effektivt. Lokaliseringen är den process där produkten anpassas för den lokala marknaden genom att programmet översätts och att för landet specifika funktioner läggs till. Ett exempel som Luong et al. (1995) tar upp för att klarare visa på skillnaden mellan internationalisering och lokalisering är att en bil från Sverige har ratten på vänster sida medan en bilförare i Japan vanligtvis sitter på höger sida. Om varan endast hade varit lokaliserad så krävs olika tillverkare för att tillverka de olika bilarna, men vid internationalisering hade man t.ex. kunnat ha en ratt som kan förflyttas och användas på både höger och vänster sida. Målet med internationaliseringen är att kunna skapa mjukvara som kan användas överallt i världen utan att källkoden ska behöva ändras. Endast när programmet körs bestäms vilket språk produkten ska använda.

Luong et al. (1995) tar upp följande fem punkter som förklarar varför internationaliserings-processen är viktig:

- Produkterna kan säljas i hela världen.
- Man får en mer effektiv utveckling av kvalitetsprodukter då det är enklare att t.ex. skapa en svensk produkt om den engelska produkten redan fungerar med svenska data.
- Det går snabbare att lansera en produkt då det inte behövs någon ytterligare

utveckling. Lokaliseringsprocessen kan påbörjas parallellt med den nationella utvecklingen.

- Med hjälp av internationaliserade produkter krävs mindre resurser, tid och pengar för lokaliseringen.
- Single-sourcing av kod (dvs att kod som är enhetlig för alla språkversioner lagras på ett ställe) är möjligt och detta gör underhåll enklare och billigare.

Lokalisering kan även ske genom att all ursprunglig kod byts ut mot det nya användarlandets kodspråk. Detta förfarande kan verka snabbast till en början men är i själva verket ofta långsammast och det blir ofta fel vid översättningen. Proceduren med att byta ut koden måste dessutom gås igenom för varje språk som programmet ska översättas till vilket kan vara mycket tidskrävande. Det är alltså bättre med internationalisering och att i ett tidigt stadium lära ut rätt teknik för detta. Utvecklaren bör med andra ord lägga alla data som är specifik för varje användarregion på ett speciellt ställe utanför koden, och dessa data läses vid körning av mjukvaran.

4.4 Hur mycket bör internationaliseras/lokaliseras?

Det finns olika nivåer inom internationalisering/lokaliseringsprocessen. De olika nivåerna kan vara bra att tillämpa vid olika tillfällen. Vilken nivå ett företag väljer att lägga sig på beror dels på vilka resurser som finns att tillgripa men till största del bör användarnas behov och krav tillgodoses. Vidare är det även avgörande om produkten redan finns på ett språk, om den bara ska tillverkas för ett enda användarland eller om det är en produkt som ska nyttillverkas för ett flertal länder på den globala marknaden. (Luong et al., 1995). De olika nivåerna är följande:

- Endast engelsk produkt. Här väljer man att inte alls lokalisera produkten. Detta innebär oftast att den kan användas i engelskspråkiga länder men de flesta engelskspråkiga länderna har olika format på ex datum och har olika valutor vilket kan ställa till problem om användarna inte är medvetna om vilka standards som används. Exempelvis så fungerar inte kalkylprogram alltid så bra när de används i olika länder då monetära enheter och räknesätt osv skiljer sig.
- En engelsk produkt som handhar europeiska data. Denna produkt är inte lokaliserad utan gränssnitt och manualer är skrivna på engelska. Dessa program ska klara av de flesta bokstavsuppsättningarna och de ska även klara de flesta kulturella konventioner som finns i Europa.

- Engelsk produkt som hanterar data som används i Fjärran Östern. Detta är svårare än de tidigare nämnda nivåerna då det ofta innebär en ändring av den ursprungliga logiken som finns i programmet och att det kan krävas en helt ny arkitektur.
- En engelsk produkt som hanterar text som både går från vänster till höger och från höger till vänster. Inom arabiskan har man nytta av detta då de läser från höger till vänster för det mesta men en del ord som läggs in i texten löper från vänster till höger. Det är alltså inte helt ovanligt att i texter se en mix av dessa skrivsätt.
- Full eller delvis översättning av det ursprungliga användargränssnittet och dokumentationen. Utöver det faktum att mjukvaran ska hantera olika alfabet så ska gränssnitt med menyer, dialogrutor, hjälptext, felmeddelande osv och dokumentation översättas. Priset för detta kan bli högt och i många fall kan det löna sig att använda sig av delvis översättning för att hålla kostnaderna nere. Delvis översättning sker ofta vid utveckling av produkter som är avsedda för programmerare och andra som anses vara vana vid datorer och som är mer bekanta med den engelska dataterminologin. De produkter som däremot ska användas av "vanliga användare", dvs allt från noviser till experter, kräver ofta full översättning av såväl gränssnitt som dokumentation.
- Full lokalisering plus specifika lokala särdrag för marknaden räknas som den sista nivån i internationaliseringsprocessen. Amerikanska produkter kan ofta säljas till Europa efter att bara ha ändrat den språkliga biten. Den asiatiska marknaden kräver däremot ofta mer förändringar för de specifika format och kulturella variationer som finns och för att t.ex. ge bättre användning av de produkter som redan finns på marknaden. Exempelvis så finns i Japan Kiri som är en populär DOS databasprodukt. Med tanke på att Kiri har över 100 000 användare i Japan så kan det vara en avgörande faktor att produkter som säljs till Japan är kompatibla med Kiri för att mjukvaran ska nå framgång. Vidare bör speciella funktioner och hjälpmedel i programmen som t.ex. ikoner och färger ses över så att de passar de användare som produkten tillverkas för.

Fortsättningsvis kommer jag att använda mig av ordet internationalisering som ett samlingsord för både internationaliserings- och lokaliseringsprocessen.

4.5 Problem vid internationalisering

I detta avsnitt har jag för avsikt att mer ingående förklara vilka problem som kan uppstå i internationaliseringsprocessen. I det första skedet ges en sammanfattande bild av de nivåer som problemen kan indelas i och dessa förklaras senare mer ingående i de närmast följande avsnitten.

Det kan vara svårt att få ett grepp om hur mycket arbete, pengar och tid som krävs för att utveckla ett program som ska säljas utomlands, och då speciellt för produktutvecklare som aldrig utvecklat liknande mjukvara tidigare. I *International User Interfaces* (Del Galdo et al., 1996) indelas de problem som ofta uppstår vid design av ett användargränssnitt och översättning av dokumentation i tre nivåer.

1. De problem som troligtvis är mest uppenbara för tillverkarna är att

internationaliseringen ofta innebär att man handhar olika språk vilket i vissa fall även kan innebära olika alfabet, antal byte för att representera varje tecken, olika datumformat osv. Det är ofta inte några större problem att förstå att det krävs många förändringar, men problemen är inte alltid så enkla att hantera för utvecklarna.

2. Den andra nivån innebär att producera ett gränssnitt och information till användaren som kan förstås och användas på språket det översätts till. Problemet är här inte bara att programmen ska utvecklas med hjälp av vanliga användbarhetsmetoder där diverse riktlinjer som finns inom gränssnittsdesign (t.ex. hur många färger som bör användas, hur design av fel- och hjälpmeddelanden bör ske osv) och dokumentation bör följas, utan även att detta ska anpassas till användarnas specifika kultur som i vissa fall kan skilja sig väsentligt från den som tillverkarna är vana vid.
3. Den tredje nivån kan vara mycket svår att ge sig i kast med då den innefattar att försöka få en inblick i ett land och dess invånares beteenden och tankesätt. System som passar användarnas speciella karaktäristika är viktigt vid internationalisering. Här inkluderas frågor som hur gör man affärer i landet, hur kommunikation sker, vad som anses artigt och inte osv.

Dessa punkter kommer jag nedan att beskriva mer ingående.

4.6 Nivå 1

I nivå 1 inkluderas internationaliseringsproblem som t.ex. översättning av all text i en produkt, sorteringsordningar, monetära format och liknande. Vissa av punkterna nedan berörs olika mycket av förändringar beroende på vilket land som produkten utvecklas för. Exempelvis kan sägas att översättningar från engelska till japanska kan ställa till problem då japanska tecken är större än de romanska och det försvårar även lagring av tecken och sortering av deras specifika alfabet medan översättningar från engelska till svenska endast berörs i viss mån av samma problem då det endast är å, ä och ö som skiljer sig alfabetmässigt.

Komponenter som är involverade i denna del av internationaliseringen är enligt Luong et al. (1995) följande:

1. Användargränssnitt, on-line- och pappersdokumentation

Här inkluderas all den text som användaren ser när programmet används. Nämnas bör felmeddelanden, hjälptext, menyer och grafik. Texten bör lagras utanför källkoden och man ska även ha i åtanke att översättningar ofta tar olika stor plats i jämförelse med ursprungsspråket och texterna kan expandera både horisontellt och vertikalt. När översättning till japanska sker så bör man tänka på att de japanska kanji-tecknen (se Appendix 1) är både bredare och högre än de romanska bokstäverna som vi använder.

Det är i allmänhet fördelaktigt om både on-line- och pappersdokumentationen internationaliseras i samma grad som mjukvarukomponenterna.

Grafik är i ett program lika viktigt som text. Ikoner som används ska helst vara kulturellt neutrala, om inte annat är det en fördel om de är enkla att byta ut. Text som används i ikoner bör sparas separat från själva grafiken då texten oftast behövs bytas ut.

2. Klassifikation av bokstäver

I engelskan finns som bekant 52 bokstäver ([A-Z], [a-z]) och alla andra tecken anses vara icke alfabetiska. Exempelvis så är svenskans å inte alfanumeriskt enligt det engelska alfabetet och kan ej användas som alfanumeriskt i t.ex. en engelsk databas utan dessa orsakar felmeddelanden om de används. För européer är detta problem inte så stort då det endast är ett fåtal bokstäver som inte kan användas men för japanerna är det ett större problem då deras tecken inte är alfanumeriska utan de är fonetiska och ideografiska (dvs skrifttyp där skrivtecknen betecknar begrepp istället för språkljud).

3. Teckenomvandling

I denna punkt inkluderas bl.a. att göra om små bokstäver till stora och vice versa. För att kunna göra om 'å' till 'Å' så krävs speciella funktioner i ett program då ett engelskt program inte klarar av detta.

4. Numeriska format

Här hittas problem som hur siffror grupperas i aktuellt land. Några exempel nedan:

USA	8,263.44
Tyskland	8.263,44
Frankrike	8 263,25

Både . , och blanksteg används med andra ord. I Japan används dessutom olika tecken för olika stora summor:

万	10.000
億	10000.000.000

5. Monetära format

Enhets- och siffergruppering är vanligtvis detsamma som gruppering av numeriskt format men valutasymbolerna och deras placering varierar från land till land. Som exempel kan nämnas:

Sverige	5 Kr
Japan	¥5
USA	\$5

I Portugal så sätts valutasymbolen i mitten av summan för att separera heltal från bråktalen. Det kan även hända att den lokala valutasymbolen som används av ett land

skiljer sig från den symbol som används internationellt.

6. Datum och tidsformat

Här finns en del olika sätt att representera datum och tid. I västvärlden används för det mesta den julianska kalendern när datum ska representeras men trots att samma kalender används så skrivs datumen på många olika sätt. T.ex.:

USA Wed Aug 11 10:13:42 PDT 1997
Italien mer, 11 ago 1997 10.13.42
Tyskland Mi., 11. Aug 1997 10:13:42 PDT

Mellanöstern går efter månkalendern som baseras på månens cykler. I Japan används den julianska kalendern, men de har även ett eget datumformat som grundar sig på det antal år som aktuell kejsare regerat. 1998 är i Japan Heisei 10. I Taiwan och Kina baseras datum på antal år som gått sedan nuvarande era startade.

7. Jämförelser

Vanligt är att sortering sker med hjälp av ASCII-kodens ordning vilket inte alltid är tillfredsställande då alla bokstäver går före de små. Exempelvis:

<u>ASCII sortering</u>	<u>Ordlistesortering</u>
BENGT	anna
Barbro	Barbro
anna	BENGT
david	david

För icke engelska språk blir detta problem mer omfattande då speciella tecken som svenskans å, ä och ö och franskans é inte hamnar på rätt ställe om man använder sig av ASCII-tabellerna. Utöver detta finns andra problem som:

- 1 till 2 tecken representationer. T.ex. ß som sorteras som ss i tyskan.
- 2 till 1 tecken representationer. T.ex. ch som kommer mellan c och d i spanskan.
- S.k. "Don't care bokstäver". T.ex. bindestreck och apostrofer.
- Primär- och sekundärsortering.

I japanskan finns flera sorteringsmöjligheter. Bl.a. kan sortering ske efter fonetisk representation, vilken "klass" tecknet tillhör eller/och antal streck som kanjitecknet innehåller.

8. Förekommande uttryck

Detta är mönster som används för att söka efter text i data. Även här används oftast ASCII-koden som grund när matchning sker. Inom engelskan skulle [A-Z] betyda alla stora bokstäver men svenskans Å, Ä och Ö skulle vara exkluderade vilket kan leda till

stora problem.

9. Kodtabeller

Kodtabeller illustrerar ett till ett representationer mellan en bokstäver/tecken och datorns bit representationer. Vanligast är att ASCII-tabellen används men detta fungerar bara på det engelska alfabetet. Med hjälp av en tilläggsstabell på 256 tecken i teckensetet kan ASCII-tabellerna även användas för de europeiska språken. Japanskan har 1000-tals tecken i sitt teckenset vilket betyder att en byte inte räcker till för att representera dessa. För att lösa detta problem har bl.a. Double Byte Character Set (DBCS) utvecklats. DBCS kodscheman använder 16 bitar för att representera varje tecken och det är kompatibelt med ASCII. I verkligheten använder man sig inte alltid av dubbelbyte för att representera tecken utan det varierar mellan enkelbyte och dubbelbytetecken. Alla kanjitecken representeras av dubbla bytes medan de romanska tecknen och vissa av kanatecknen representeras av en enkelbyte. Det kan lätt uppstå problem på grund av användandet av både enkla och dubbla bytes då det kan vara svårt att veta när en byte och inte två används. Enligt Luong et al. (1995) så är det viktigt att kontrollera att t.ex. hela tecknet suddas ut vid användande av Delete. Detta innebär att man måste kolla om tecknet som suddas ut lagras med en eller två bytes. Ett annat exempel är när man förflyttar sig i en textmassa. Det är vid dessa tillfällen viktigt att markören inte kan ställas mitt i ett tecken utan det ska alltid bara kunna stå framför eller bakom ett tecken. Endast ett helt tecken ska kunna markeras också.

10. Textriktning

De flesta latinbaserade skriftsätten skrivs från vänster till höger medan hebreiska och arabiska skrivs i två riktningar. Deras egna tecken skrivs från höger till vänster men nummer och latinska tecken, som blandas med de arabiska tecknen, skrivs från vänster till höger. I Japan så skrivs text vanligtvis lodrätt från höger till vänster men det ska även gå att skriva vågrätt från vänster till höger i deras skrivprogram. Specifikt för japanska textriktningen är bl.a. att markören ska flyttas till vänster när raden är slut eller om Enter nedtrycks. Vidare så ska texten flyttas horisontellt om PageUp eller PageDown (som även används på japanska datorer) trycks ner. Detsamma gäller för Home och End. Dessutom sker markering av text kolumnvis och inte radvis som vi är vana vid.

Ett annat problem som uppstår när japanska används är att reglerna skiljer sig angående var i en text en brytning ska ske vid t.ex. radbyte. I svenskan används mellanrum mellan ord som en naturlig brytpunkt men japanskan använder inte mellanrum i sina texter. Japanskan kan bryta texterna i princip var som helst, även mitt i ord, men med ett fåtal undantag. Exempel på detta är att början på citat inte får finnas sist i en kolumn och slut av citat och små kanatecken inte får finnas i början av en kolumn.

11. Bortre Österns input metoder

Eftersom det inte går att ha tangentbord som täcker de tusentals kanjitecken som finns i japanskan så måste användarna förses med Input Method Editors (IMEs). Det japanska

tangentborden innehåller som vanligt de romanska bokstäverna och siffrorna 0-9, men de har även kanatecknen representerade. Text skrivs antingen in med hjälp av romaji eller kana. Om romaji används så översätts dessa till kana allt eftersom texten skrivs in. Kana omvandlas sedan till kanjitecken genom att man väljer det rätta tecknet från en lista som visas (se även Appendix 1).

4.7 Nivå 2

Det krävs ofta mycket av ett program som ska säljas på den internationella marknaden då det utöver de vanliga användarvänlighetsteknikerna, där diverse riktlinjer som finns inom gränssnittsdesign och dokumentation bör följas, även krävs att gränssnitt och dokumentation anpassas till de speciella företeelser som finns inom den kultur där programmet ska användas. I detta avsnitt kommer jag att i sammanfattande form förklara vissa grundläggande punkter angående användarvänlighet, och hur man bör gå till väga för att kontrollera att produkten är användbar i användarlandet. Vidare kommer jag att beskriva hur ikoner, symboler, dokumentation och dokumentationsformat kan ändras för att underlätta i internationaliseringsprocessen.

4.7.1 Användarvänlighet

Användarvänlighet är en mycket viktig fråga som har avhandlats i en mängd böcker. Jag tänker här bara ta upp de allra viktigaste punkterna som direkt kan kopplas till internationalisering av gränssnitt.

Ett väl designat användargränssnitt tänker man inte på utan användaren ska kunna koncentrera sig på de uppgifter hon har för avsikt att utföra. Mjukvaran accepteras om det bl.a. innehåller bra meddelanden, layout och färgsättning.

Felmeddelanden kan te sig som vaga, hotande och obegripliga om de inte designas på ett korrekt sätt. Om programmet inte kan göra en ordentlig diagnos av varför felet uppstår så blir meddelandena oftast inte bra heller. Enligt Allwood (1991) så bör programmen innehålla mer hjälpsamma och vänliga felmeddelanden. Språket ska vara enkelt och tydligt och fördömande ord bör undvikas i största möjliga mån. Vilken språkform som används i felmeddelanden varierar även från kultur till kultur och detta bör noggrant undersökas innan produkten börjar användas.

En annan viktig punkt som tas upp av Schneiderman i *Designing the User Interface* (1992) är att det ska råda konsistens i programmet och detta gäller både mellan olika skärmbilder och inom en och samma skärmbild. Detta innebär att knappar, färger, texturor osv bör vara liknande på olika ställen i programmet så att det är enkelt att navigera sig och att användaren enkelt kan se och förstå vad det är som ska göras vid olika tillfällen. På NEC i Osaka har det enligt Toshiyuki Asahi utvecklats ett program som kontrollerar att skärmbilderna är konsistenta och att färger, rubriker, meddelanden osv överensstämmer från skärmbild till skärmbild. Med hjälp av nuvarande program kan

ungefär 30% av ett programs gränssnitt kontrolleras.

Vad gäller färgsättning så ska man vara försiktig. Man bör inte använda för många färger utan vanligtvis bör det vara 4-7 stycken genom hela systemet. Vidare är det viktigt att tänka på att olika färger kan representera olika saker i olika länder och färger kan även representera olika saker inom olika yrken (del Galdo et al. 1996). Exempelvis:

Bilister:	Röd - stopp, Gul - varning
Kemister:	Röd - varmt, Blå - kallt
Ekonomer:	Röd - finansiell förlust, Svart - vinst
Karttillverkare:	Blå - vatten, Grön - skog, Gul - öken

Information kan även grupperas med hjälp av färger genom att ge all information som hör ihop liknande färg. Enligt del Galdo et al. (1996) finns det numera en kunskapsbas på Internet som ger förklaringar vad olika färger betyder i olika kulturer och tips om hur dessa bör användas. Det finns mängder av oskrivna regler som bör följas vid design av gränssnitt men detta får vidare behandlas vid ett annat tillfälle.

4.7.2 Testning

För att kontrollera om gränssnittet passar för den speciella kulturen där det ska användas krävs grundliga undersökningar. Omfattande kontroller om användarna i andra länder förstår och på ett enkelt sätt kan använda sig av applikationen är viktiga. Detta kan enligt del Galdo et al. (1996) ske på flera olika sätt.

- Ett av dessa är att människor från användarlandet ser över programmet och försöker kontrollera om och vad som kan tänkas ställa till problem i hemlandet. Dessa testare kan vara antingen användbarhetskonsulter eller helt vanliga datoranvändare som ger en första reaktion på mjukvarans funktioner. Detta kallas internationell inspektion.
- Internationell användartestning går ut på att personer som är tänkta som slutanvändare av produkten sätts att testa diverse funktioner. Detta kan ske genom att (a) någon åker till mållandet och övervakar testen. Detta tillvägagångssätt ger försöksledaren en god inblick i vad som verkar svårt för användaren och vilka funktioner som tar lång tid att genomföra. Ett annat sätt är att (b) testen körs på distans. Detta är ett billigare alternativ men det kan vara svårare att kommunicera med dem som testar produkten under testens gång. Ett tredje alternativ är att (c) hyra in lokala användbarhetskonsulter som utför testen i sitt eget hemland och på sitt eget språk. Dessa konsulter kan tillverka egna test som passar för just de användare som ska kontrolleras. Om konsulterna inte har full insikt i hur programmet fungerar till en början så är det naturligtvis viktigt att ge klara instruktioner hur testet ska utföras och vad man vill ha ut för resultat. Ett annat alternativ som är möjligt om företaget har ett lokalt kontor i landet är att (d) använda folk från den egna branschen som uppför testsituationer. Detta blir ett billigare alternativ än att anställa en konsult men å andra sidan är dessa personer oftast inte tränade inom användbarhet så det gäller att förklara grundligt vad de ska leta efter när testen utförs.

4.7.3 Ikoner och symboler

Del Galdo et al. (1996) gör en distinktion mellan ikoner och symboler genom att beskriva ikoner som familjära symboler och som lättförståeliga och konkreta representationer av objekt eller människor, medan en symbol är något mer abstrakt som ofta kräver diverse instruktioner för att de ska förstås. Vid användande av ikoner och symboler är det viktigt att ha i åtanke att dessa måste kunna användas på olika skärmstandards. Konsistens, tydlighet, enkelhet och igenkännande är nyckelord i designprocessen och det bör bara finnas ett begränsat antal ikoner och symboler i ett system då det kan vara svårt att skapa, lära ut och förstå ikonerna annars.

Något som kan vara mycket viktigt att tänka på när ett gränssnitt skapas till användare i andra länder är grafik och text som används till ikoner och symboler. Bilder och färger har inte alltid samma betydelse i olika kulturer vilket betyder att detta måste ses över noggrant innan systemet tas i användning. Väldesignade ikoner är ofta dyra men är de slarvigt gjorda eller dåligt genomtänkta så kan de skapa förvirring och sänka produktiviteten. Naturligtvis kan inte alla i ett land ha samma bakgrundserfarenheter men allmänna grundprinciper som t.ex. vilken färg brevlådorna i landet har och hur en vanlig soptunna ser ut bör beaktas. Man har t.ex. haft stora problem vid val av bild till e-mailfunktionen i diverse program då postlådorna har olika utseenden överallt i världen. Kuvert har blivit en smidigare lösning på detta då deras utseende inte skiljer sig lika mycket från land till land.

Ikoner är bra att använda då de:

- drar till sig uppmärksamhet och är tilltalande.
- ofta tar mindre plats än ett helt ord vilket kan vara praktiskt vid platsbrist på skärmen.
- kan förstås över språkgränserna om de är väldesignade.
- inte bara är bra att använda på skärmen utan även i dokumentation.
- om de är välgjorda är enklare och snabbare att förstå än motsvarande ord.

4.7.4 Dokumentation

En annan viktig fråga vid översättning av produkter berör dokumentationen. Det kan uppstå stora problem när företag ska ta fram rätt dokument i rätt format till kunder internationellt och därför är det viktigt att noggrant ta reda på vad kunderna förväntar sig av hjälpmaterial och övrig dokumentation. Ett exempel som kan nämnas i sammanhanget är Kanada som ofta kräver flera språk på sin dokumentation då det både talas engelska och franska där.

Många använder sig av flerspråkiga dokument när de säljer en produkt till flera länder. Luong et al. (1995) skriver att flerspråkiga dokument kan komma till nytta när (1) det finns en lands- eller regionsspecifik lagstadga om att dokumentationen ska finnas på flera språk. Detta är fallet i bl.a. Kanada (franska och engelska), Belgien (franska, flamländska

och tyska) och Finland (finska och svenska). I t.ex. Frankrike, Spanien och Danmark finns lagstadgat att viss specifik dokumentation måste finnas på det lokala språket. Många internationella organisationer och företag kräver flerspråkiga dokument för sina användare. (2) På internationella företag kan produktiviteten höjas om dokumentationen finns tillgänglig på både modersmålet och på det språk som används där företagets huvudkontor finns. Modersmålet är naturligtvis bra då användarna enklare förstår den nya produkten, det blir enklare med problemlösning och felhantering. Med hjälp av att dokumentationen finns på "företagets språk" blir det enklare att kommunicera med filialer i andra länder då orden finns tillgängliga på minst två språk samtidigt. (3) Flerspråkiga dokument kan verka kostnadssparande för företag då de kan höja produktiviteten, men detta bör undersökas noggrant innan översättning sker då framställning och paketering kan bli kostsamt. (4) Flerspråkiga dokument gör även att företag kan anställa den bästa arbetskraften de kan hitta på den internationella marknaden oavsett språk.

I många fall läses dokumentationen trots allt på annat än modersmålet p.g.a.:

- Mjukvarutillverkarna tror att användarna kan läsa manualer på främmande språk (oftast engelska).
- Användarna vill börja använda produkten direkt och det kan ibland betyda innan dokumentationen är översatt.
- Tillverkarna väntar ibland med att översätta dokumentation tills varan redan finns i butikerna.
- Det är inte lönsamt att översätta dokumentation till produkter som bara säljs i ett fåtal upplagor.
- Även om mjukvaran, övningsuppgifterna och referensdokumentationen översätts så kan andra av systemets delar som hjälpdatabaser osv fortfarande vara på originalspråket.

4.7.5 Dokument format

Efter att ha beslutat sig för att använda sig av flerspråkiga dokument så återstår beslutet angående vilket dokumentformat som ska användas. Det finns en hel del olika modeller som var och en kan vara bra vid olika tillfällen. Exempelvis är installationsmanualer, som innehåller många illustrationer, effektiva om man väljer att ha en och samma bild som används till alla språk som finns i manualen. Nedan kommer jag att ta upp några av de vanligaste formaten som används i olika typer av dokumentation (del Galdo et al., 1996).

Collated

Detta är en samling av de olika språkversionerna som finns av ett och samma dokument. Manualerna på de olika språken presenteras ett efter ett.

Denna modell är enklast att framställa och publicera då det mest liknar ett enspråkigt dokument. Metoden används till nästan alla dokument som är under 150 sidor. Det är viktigt att noggrant markera varje ny språksektion så att användarna enkelt och snabbt kan hitta sitt eget språk. Det bör finnas en titel för varje språk och denna ska inte bara finnas inne i boken utan även visas på dokumentets framsida.

Facing pages

Denna modell går ut på att ett språk presenteras på udda sidor och ett annat presenteras på de jämna och kan således endast användas på tvåspråkiga dokument.

Här delas ingen information mellan språken utan det kan sägas vara två skilda publikationer. Det är bra om sidorna som visas samtidigt när en bok öppnas innehåller samma information eftersom det då är enkelt att jämföra språken och då det kan underlätta vid kommunikationen mellan personer med olika språk.

Sida vid sida

I sida vid sida modellen visas vanligtvis två, men ibland även tre språk på en och samma sida som indelas i kolumner.

Om möjligt kan en bild användas till båda språken. Ska t.ex. diagram användas skall texten på dessa finnas på alla språk som används i dokumentationen. Det kan ibland uppstå problem med att matcha texterna då vissa översättningar tar mer plats än andra. Text kan expandera med 30-100% och detta innebär att det blir en hel del tomrum i texten på vissa språk.

Stacked

Här staplas språken vertikalt på en och samma sida.

Denna modell används främst då det är lite text som ska representeras. Den är fördelaktig då bilder ska förklaras eftersom samma bild kan användas till alla språk och detta utnyttjas ofta då t.ex. installationsguider och säkerhetsinformation dokumenteras. Denna typ av dokument bör inte användas till mer än fyra språk men om texterna är väldigt korta går det bra att ta med ända upp till tio språk. I denna modell är det inte alltid bra att ta med språk som japanska eller arabiska då deras texter löper åt olika håll. Det kan kännas onaturligt för dessa användare att läsa text som går från vänster till höger och att skriva ut text som de är vana att läsa den kan skapa stora problem i layouten. Även användande av olika alfabet kan ställa till trubbel vid layout och tryckning.

Tumble

Dessa dokument är alltid tvåspråkiga och det finns två varianter av tumble.

1. Horizontal tumble

Detta används när två språk med samma alfabetstyp som t.ex. engelska och svenska ska representeras.

I exemplet ovan läses den engelska delen från framsidan av boken medan den svenska påbörjas bakifrån. Vid läsning av den svenska texten kommer boken att vändas upp och ned i jämförelse med när man läser den engelska delen.

2. Vertical tumble

Detta används främst till språkpar som t.ex. svenska och japanska som skriver och läser åt olika håll.

I denna modell så påbörjas den svenska delen som en vanlig bok, dvs framifrån medan den japanska delen läses bakifrån. Detta är en smidig lösning eftersom japanerna alltid

läser sina böcker på det sätt som vi anser vara bak och fram.

Tumble modellerna är bra då varje språk alltid får var sin framsida. Ett tjockt papper som markerar var de båda böckerna slutar kan även rekommenderas.

4.8 Nivå 3

Den tredje och sista nivån är det lätt att bortse från när utveckling av ett system påbörjas då det inte alltid är så lätt att få grepp om vad den egentligen innebär. Att framställa ett system som passar ett speciellt lands karaktäristika och användarnas krav kräver att åtminstone en i systemutvecklingsgruppen vet eller har haft chans att undersöka landet som produkten ska säljas i. Allt från grundläggande programstrukturer till små detaljer som ikoner och färgsättning kan vara viktigt när mjukvara produceras.

Ett sätt att undersöka ett lands specifika drag är att följa antropologernas idéer och skapa kulturella modeller. Detta går ut på att jämföra likheter och skillnader mellan två eller flera kulturer genom att använda vad Luong et al. (1995) kallar internationella variabler som kan beskrivas som kategorier som organiserar kulturella data. Dessa kulturella data kan reflektera en nations kultur, kollektiva kulturer, kulturella skillnader mellan grupper av användare, internationella marknaden osv. Internationella variabler kan t.ex. fokuseras på (1) skillnader som är enkla att hitta som t.ex. politiska, ekonomiska, olika skriftsystem eller (2) på subjektiv information som värderingar, beteende, intellektuella system osv. Dessa sistnämnda faktorer är förstås mycket svårare att undersöka och ett mer komplext undersökningssätt krävs.

Orsaker till att kulturella modeller används är bl.a. följande:

- Att inför internationalisering försöka identifiera global information som är relevant för många kulturer.
- Identifiera kulturella fördomar genom att applicera den kulturella modellen på sin egen kultur.
- Identifiera effektiva kulturella metaforer.
- Hitta vilken grad av lokalisering som krävs.
- Undvika kulturella misstag som kan väcka anstöt hos användarna eller som kan förvränga informationsframställningen.
- För att utvärdera effektiviteten av ett internationellt gränssnitt.

Följande är ett exempel på en metod som används för att skapa en kulturell modell. Det finns naturligtvis en mängd olika metoder som kan användas men jag valde denna, som finns beskriven i Luong et al. (1995), då den på ett bra sätt visar hur en kulturell modell kan se ut. Upphovsmannen är Geert Hofstede som grundade och styrde the Personnel Research Department på IBM från 1965-1971. En mängd studier inom IBM ledde fram till hans kulturella modell och de internationella variabler som han använde till att identifiera skillnader i tänkande och handlande.

Hofstede använde följande variabler:

Maktdistans

Här görs försök att mäta hur anställda på ett företag reagerar på makthavande och ledande personer på firman. Hofstede skapade ett index för att mäta maktdistans och från detta index kan utläsas att i t.ex. Latin amerikanska länder, Frankrike, Spanien, Asien och Afrika så tenderar underordnade att vara rädda för sina chefer. De överordnade frågar inte övriga anställda om råd och cheferna är ofta enväldiga härskare. Detta kallas hög maktdistans. Låg maktdistans kan hittas på företag där underordnade tenderar att ställa sig frågande till ledarnas beslut och där de gärna utmanar sina överordnade. I dessa länder konsulterar cheferna oftare sina anställda. Detta sätt att arbeta kan man hitta i t.ex. USA, Storbritannien, Sverige, Tyskland, Norge, Holland, Danmark, Nya Zeeland och Israel.

Kollektivism - Individualism

Detta avser vilka band som finns mellan individer i ett samhälle. I en kultur som är kollektivistisk så utvecklar individerna starka band mellan varandra och det förväntas att de ska vara lojala mot gruppen och företaget där de samverkar. Här kan nämnas Japan, Mexiko, Korea och Grekland som är starka kollektiva kulturer. I en individualistisk kultur förväntas det att var och en ska sköta sina affärer på egen hand. USA, Frankrike, Tyskland, Kanada och Syd Afrika är exempel på individualistiska länder.

Femininitet - Maskulinitet

Hofstede kom i sina undersökningar fram till att män och kvinnors mål skiljer sig åt markant. Männen värdesätter exempelvis att få den uppmärksamhet som förtjänas, ha möjlighet att tjäna mycket pengar, ha möjlighet att avancera inom företaget medan kvinnorna såg goda relationer med överordnade, bra samarbete med övriga anställda, och säkerhet att få jobba vidare på arbetsplatsen så länge man vill som viktiga punkter. Länder som tenderar uppskatta de kvinnliga önskemålen är Sverige, Spanien, Korea, Frankrike, Finland, Danmark, Indonesien osv, medan ett maskulint index föredras i länder som Japan, USA, Storbritannien, Italien, Hong Kong och Tyskland.

Osäkerhetsundvikande

Här försöker Hofstede sig på att mäta till vilken grad olika människor känner sig hotade av osäkra eller okända situationer. Enheterna för denna variabel är stark och svag. Vid starkt osäkerhetsundvikande så finner personer att okända situationer är hotande och undviker därav dessa i största möjliga mån. Exempel på länder där "annorlunda är farligt" är Japan, Sydkorea och de Latinamerikanska länderna. Svagt osäkerhetsundvikande kan hittas i Holland, Tyskland, USA, Storbritannien osv, och här känner sig folket mindre hotade av okända situationer. Ovisshet anses vara mer normalt och man är mer nyfiken på det annorlunda.

Långsiktighet - Kortsiktighet

Detta index visar på huruvida människor i en kultur har lång- eller kortsiktig planering i livet. Där långsiktighet (angelägenhet inför framtiden) begagnas finns nyckelord som ihärdighet, uthållighet, respekt för hierarkiska relationer, sparsamhet osv. Kina, Hong Kong, Taiwan, Japan och Indien är länder där långsiktig livsplanering används. Kortsiktig planering, eller oro för då- och nutid, hittas i bl.a. USA, Australien, Storbritannien, Filippinerna, Pakistan. Här är nyckelorden säkerhet och stabilitet, rädda ditt rykte, respekt för traditioner och reciprocitet.

De resultat som uppnås vid när dessa internationella variabler undersöks ska senare användas som vägvisande när produkter skapas för den internationella marknaden.

5. Diskussion

Som resultatdelen i rapporten visar på så krävs det en hel del arbete för att marknadsföra produkter utomlands. Hur mycket tid och pengar som går åt är delvis beroende av hur mycket man valt att internationalisera produkten men även andra faktorer som vilket land som är inköpare och om produkten redan finns i andra versioner på marknaden är även väsentliga. I denna diskussion tänker jag försöka dra allmänna slutsatser och besvara frågor som uppkommit tidigare i rapporten.

5.1 Människa- datorinteraktion

Svaret på frågan om datoranvändare världen över skiljer sig åt vad gäller grundläggande kognitiva fakta kunde delvis utrönas redan under rubriken människa- datorinteraktion i resultatdelen av rapporten. Sociala normer och kulturell bakgrund skapar som sagt var människans förståelse och detta sätter sin prägel även på det sätt som en användare ser på ett gränssnitt och medföljande dokumentation. Man får en känsla av att skillnader som inte märks så tydligt vid en direkt inblick i ett annat land med ett annat språk ofta får stiga åt sidan för mer uppenbara problem som textriktningar och sorteringsordningar. Jag tror dock att ett program där de grundläggande riktlinjerna inom människa- datorinteraktion inte anammats förlorar användare då dessa system ofta tenderar att inte bli så intuitiva och lättanvända som många användare kräver att dagens program bör vara.

5.2 Internationaliseringsproblem

Frågan om det är lönsamt att översätta en hel produkt med gränssnitt, dokumentation och allt eller om bara delar av produkten bör översättas är inte en enkel fråga att besvara. Som jag skrivit tidigare i rapporten så beror detta dels på vad användarna väntar sig av produkten. Å ena sidan kan en engelsk produkt användas i nästan hela världen då de

flesta behärskar det engelska språket, men om så är fallet får användarna göras medvetna om att funktioner som användande av specialtecken som å, ä och ö försvinner och att man kan vänta sig en annan sorteringsordning än den man är van vid. Kalkylprogram som t.ex. Excel kan vara svåra att använda om inte användarna är uppmärksammade på att valutor, termer och räknesätt inte fungerar som det gör på deras hemspråk. Å andra sidan finns de produkter som är helt översatta för att passa specifika slutanvändare på marknaden. I de fall där mjukvaran ska användas av noviser på området brukar det vara fördelaktigt att använda sig av full lokalisering plus specifika lokala särdrag för marknaden då dessa produkter oftast får mer positiva reaktioner och då det är enklare att förstå och använda dessa program. (Luong et al., 1995). Mellan dessa två ytterligheter finns som sagt en mängd nivåer som man kan lägga sig på när det gäller internationalisering. Företagets samlade kompetensnivå och ett projekts budget är även de faktorer som bestämmer på vilken nivå internationaliseringen kan läggas.

Det är viktigt att vara medveten om att det finns både för och nackdelar med att internationalisera en produkt. Fördelar som att produkten kan säljas på den globala marknaden och att program som säljs i flera länder kan utvecklas parallellt måste vägas upp mot nackdelar som att produktionen är både mer tidskrävande och dyrare än att utveckla produkter för den nationella marknaden.

Om man tagit beslut att en produkt ska översättas för användande i ett annat land så återstår frågor som hur mycket som verkligen behövs ändras i ett program för att det ska vara användbart. Att problemet inte består i att bara översätta gränssnitt och dokumentation när det ursprungliga programmet är testat och klart kan vi nog vara överens om. Mer djupgående förändringar behövs för att en produkt ska vara användbar. Jag skulle vilja säga att alla de tre nivåer av internationaliseringsproblem som finns bör tas hänsyn till då alla dessa nivåer hjälper till i framställningen av ett komplett och intuitivt lättanvänt system.

Ändringarna som innefattas i den första nivån kan nog sägas vara de som kan ställa till mest förtret om de inte är korrekt genomförda då hela systemet är beroende av att flera av dessa funktioner fungerar som de ska. Om exempelvis de monetära enheterna inte är korrekta går systemet förmodligen att använda i alla fall men det kan vara en mycket störande faktor att hela tiden se fel format presenteras. Händer det att man i ett japanskt ordbehandlingsprogram kan radera halva tecken och ställa sig mitt i ett tecken på grund av att programutvecklingen inte skett på ett korrekt vis så kan detta ha förödande konsekvenser då programmet troligtvis blir mer eller mindre oanvändbart. Testning av program är naturligtvis ytterst väsentligt då minsta lilla fel i en produkt kan ställa den i en dålig dager.

Nivå 2 i problematiken berör som sagt var bl.a. användarvänlighet, ikoner och dokumentation och att allt som rör dessa komponenter bör ändras så att de passar de föreställningar och erfarenheter som användarna i inköpslandet har. Det räcker inte här att bara utgå från de allmänna regler och hänvisningar som finns inom gränssnitts- och dokumentdesign utan det är även viktigt att se till att just de särdrag som kan hittas hos inköpslandet tas tillvara och att det noggrant testas så att användarna verkligen uppfattar

produkten som bra och enkel att använda. Jag tror att en mjukvaras gränssnitt gärna anses vara färdigarbetat när den första nationella versionen av en produkt godkänts och det är säkerligen lätt att fränse de mindre förändringar som kanske inte alltid är nödvändiga för att ett program ska kunna förstås i andra länder. Det kan dock vara dessa förändringar som är skillnaden mellan ett program som är enkelt att förstå och använda och ett program som är tungarbetat och som användarna lätt stör sig på. Att framställa passande dokumentation kan också vara avgörande faktorer då användare i vissa länder som t.ex. Japan är noggranna med läsande av manualer innan själva mjukvaran börjar användas. Om manualen inte är väl genomtänkt tror jag att hela produkten kan få ett dåligt anseende.

Den tredje nivån går förmodligen inte alltid att påvisa om man lyckats med eller inte i ett program. En kulturell modell bör användas som förebild och som en fingervisning för kulturella skillnader mellan grupper av användare i olika länder eller på olika företag. Jag tror att man endast genom att göra undersökningar på den specifika användargruppen som produkten är avsedd för kan kontrollera huruvida de kulturella modellerna haft effekt eller inte. Jag kan tänka mig att en populär och välanvänd produkt har lyckats i de flesta avseenden vad gäller alla nivåer i internationaliseringsproblematiken.

Vilken av undersökningsmodellerna som används för att kolla om produkten har fått positiva eller negativa reaktioner i ett land är även det en fråga som företagen får ta ställning till. Det kan naturligtvis vara ett billigt alternativ att använda sig av folk i inköpslandet för att hjälpa till att kontrollera testsituationerna men detta får inte alltid önskat resultat då dessa testare inte alltid har kontroll på vad programmet utför och vad som väntas av testen. Att istället skicka en testare från produktionsteamet kan verka bättre då dessa personer ofta har mer kontroll på programmet i sig men språkbarriärer och resekostnader kan ställa till problem här. De olika möjligheterna som finns bör ses över noggrant innan beslut tas.

5.3 Japanska användare

Amerikanska produkter brukar ofta kunna säljas till europeiska länder utan att allt för många förändringar krävs, men vanligtvis kräver den asiatiska marknaden betydligt fler förändringar. Vad kan då detta tänkas bero på vad gäller den japanska marknaden?

Viss del av problematiken uppstår eftersom det japanska skriftsättet kräver så många förändringar i uppläggningsarna i ett program. Listan på sånt som kan kopplas ihop med de språkliga skillnaderna kan som jag visat på i resultatdelen göras lång och detta är tidskrävande förändringar som måste utföras vid utveckling av produkter. Jag tror även att problemen med den språkliga biten kan försvåras då de använder ett skriftsystem som vi har svårt att kontrollera om det som skrivs är korrekt eller inte. Det krävs i många fall att det finns japaner med i gruppen som utvecklar produkten som kan kolla om det verkligen är riktiga tecken som skrivs, att hela tecken raderas och att sorteringsordningar och andra speciella funktioner fungerar som de ska osv.

En annan faktor som bör nämnas är de program som redan finns på den japanska marknaden. Program som kommer från andra länder behöver ofta modifieras mycket för att anpassas till de japanska programmen medan det inte krävs lika mycket för att ett program från USA ska vara kompatibelt med t.ex. ett program från Sverige. Vidare så finns det som sagt var få programmerare i Japan och detta innebär att företagen antingen anställer folk från andra länder eller att fler program köps från den internationella marknaden. I båda dessa fallen är det ytterst viktigt att de som tillverkar programmen har en förståelse för vad slutanvändarna verkligen vill ha ut av sin produkt. Det kan kanske vara enklare för en person som redan är stationerad i Japan att få en insikt i hur det japanska tankesättet skiljer sig åt i jämförelse med hur människor i t.ex. Sverige tänker.

Japanernas kunskaper i det engelska språket kan även ställa till problem ibland. Då många inte talar särskilt bra engelska så är det viktigt att produkter som säljs till Japan är bra översatta. Full översättning med specifika lokala särdrag verkar vara den nivå av internationalisering som bör användas i de flesta fallen och speciellt när användarna inte är experter.

För att förstå den kultur där en produkt ska säljas och användas måste designers först förstå en kulturs karaktärsdrag. Detta kan ske antingen genom att tala med förmodade användare och undersöka deras förväntningar och krav genom t.ex. frågeformulär, eller så kan man exempelvis undersöka produkter som redan varit framgångsrika på den japanska marknaden och på så sätt få en inblick i vilka särdrag som bör tas hänsyn till. Jag tror att de japanska användarnas tänkande kan skilja sig mycket från t.ex. det västerländska tänkandet då religion, kultur, arbetsförhållanden med mera är väsensskilt i många fall och detta kan ha effekt på hur de använder och uppfattar gränssnitt och dokumentation. Det finns ännu så länge inte universella gränssnitt som kan användas i vilken kultur som helst.

6. Appendix 1

I detta avsnitt tänker jag kortfattat ge en beskrivning av det japanska skriftsystemet och hur japanska orbehandlingsprogram och tangentbord fungerar.

I Japan - kultur och historia (Fujio Düring, 1985) kan man se att det japanska skriftsystemet inte innehåller ett alfabet, som vi är vana vid, utan de använder sig av ideografiska tecken (kanji) där skrivtecknen betecknar begrepp istället för språkljud och av fonogram som är ljudskrift (hiragana och katakana). Japanerna lånade på 500-talet in kinesiskans kanji-tecken som innehåller många tusen tecken men de nöjde sig inte med detta utan de utvecklade dessutom två egna alfabet, hiragana och katakana, som med ett gemensamt namn kallas för kana och är förenklingar av olika kanji-tecken. De sistnämnda alfabeterna innehåller 46 tecken vardera. De tre skriftsätten används tillsammans med det romanska alfabetet i en salig blandning när text skrivs. Katakana används till största delen för att representera lånord, men eftersom alla tecken förutom a, i, u, e, o och n representerar två eller tre bokstavs-kombinationer så kan de utländska orden oftast inte översättas på ett korrekt sätt utan de flesta blir "japaniserade". T.ex. så

blir computer konpyuutaa på japanska, julen är kurisumasu och själv så heter jag Ururika. Hiragana representerar som katakana även det en, två eller tre bokstäver och detta skriftsätt används oftast för att t.ex. böja verb, sätta prepositioner och även för att förklara hur kanji-tecknen ska uttalas. Man kan med andra ord omskriva alla kanji-tecken med hjälp av hiragana och detta använder man sig av i ordbehandlingsprogrammen.

När ordbehandlingsprogrammen utvecklades byggdes dessa på skrivmaskinsmetaforen, dvs man skapade dem så att de skulle likna en skrivmaskin med tabbar, marginaler osv och detta skulle underlätta då de flesta använt sig av en skrivmaskin vid olika tillfällen. Detta var dock inte fallet i Japan då det inte varit möjligt att skapa en skrivmaskin som kunde hantera alla de tusentals tecken som behövdes utan det var i stället papper och penna som fick användas i de flesta fallen. Datoriseringen har förstås gjort det möjligt att använda alla de nödvändiga tecknen men metoden är lite annorlunda än den vi är vana vid. I Japan kan man antingen välja att skriva in orden med det vanliga romanska alfabetet som vi är vana vid, eller så kan tangentbordet användas direkt för att skriva in hiragana-tecknen. Om romanska alfabetet används omvandlas dessa bokstäver direkt till hiragana om ett korrekt ord skrivits. Från hiragana kan orden sedan omvandlas till kanji genom att användaren väljer ut de hiragana-tecken som ska göras om till kanji för att sedan automatiskt få upp en lista över möjliga kanji-kombinationer som man har möjlighet att välja från.

7. Referenser

- Allwood, C.M. (1991). Människa-datorinteraktion Ett psykologiskt perspektiv. Lund: Studentlitteratur.
- del Galdo, E., & Nielsen, J. (1996). International User Interfaces. New York: John Wiley & Sons, Inc.
- Fujio Düring, C. (1985). Japan - kultur och historia. Lund: Studentlitteratur.
- Luong, T.V., & Lok, J.S.H., & Taylor, D.J., & Driscoll, K. (1995). Internationalization. Developing Software for Global Markets. New York: John Wiley & Sons, Inc.
- Pettersson, M. (1996). Intuitive User Interfaces. Unpublished manuscript, Skövde högskola, Institutionen för datavetenskap.
- Shneiderman, B. (1992). Designing the User Interface. Reading, Massachusetts: Addison-Wesley Publishing Company.