

Ville Lamminaho

LOKALISOINTI TEKNISESTÄ NÄKÖKULMASTA



JYVÄSKYLÄN YLIOPISTO
TIETOJENKÄSITTELYTIETEIDEN LAITOS
2013

TIIVISTELMÄ

Lamminaho, Ville

Lokalisointi teknisestä näkökulmasta

Jyväskylä: Jyväskylän yliopisto, 2013, 27 s.

Tietojärjestelmätiede, kandidaatintutkielma

Ohjaaja: Halttunen, Veikko

Lokalisointi voidaan määrittää kieli- ja teknologiaosaamisen yhdistämisenä kielelliset ja kulttuuriset rajat ylittävän tuotteen aikaansaamiseksi. Lokalisointi voidaan käsittää osana globalisointia, johon kuuluu myös kansainvälistäminen ja kääntäminen. Lokalisoinnin eri työvaiheissa käytetään apuna tietokoneavusteisia työkaluja, kuten käännösmuisteja ja terminologiatyökaluja. Tuotteen lokalisointi on monitasoinen ja monialainen projekti, jonka menestyksekkääseen suorittamiseen tarvitaan paljon eri alojen asiantuntijoita.

Tässä kandidaatintutkielmassa pyritään vastaamaan kysymykseen ”Mil-lainen on lokalisointiprosessi teknisestä näkökulmasta?”. Tutkielman lopputuloksena voidaan todeta, että lokalisointiprosessin tekniset työvaiheet voidaan jakaa kolmeen kokonaisuuteen, jotka ovat lähdemateriaalin analysointi, tiedos-tojen valmistelu ja tiedostojen viimeistely. Näiden kokonaisuuksien sisältämät työvaiheet riippuvat projektikohtaisesta lokalisoitavasta sisällöstä ja tiedosto-formaateista.

Asiasanat: lokalisointi, kansainvälistäminen, tietokoneavusteiset käännöstyöka-lut, käännösmuistit, lokalisointi-insinööri

ABSTRACT

Lamminaho, Ville
Technical Perspective on Localisation
Jyväskylä: University of Jyväskylä, 2013, 27 p.
Information Systems, Bachelor's Thesis
Supervisor: Halttunen, Veikko

Localisation can be defined as combining language and technology to produce a product that can cross cultural and language barriers. Localisation can be seen as a part of globalisation, which includes also internationalisation and translation. Using computer-aided translation tools, such as translation memories and terminology tools, can be helpful in different phases of the localisation process. Localising a product is a multi-level and multidisciplinary project, and to perform it successfully, variety of experts from different fields are needed.

This Bachelor's thesis tries to provide answer to the question "Which are the technical perspectives on localisation process?" As a result, the technical phases of localisation process can be divided into three categories, which are analysing of the materials, preparation of the files and post processing of the files. These categories can include many different tasks, which vary from project to project and depend on localisable content and file formats.

Keywords: localisation, internationalisation, computer-aided translation technology, translation memories, localisation engineering

SISÄLLYS

TIIVISTELMÄ	2
ABSTRACT	3
SISÄLLYS.....	4
1 JOHDANTO.....	5
2 LOKALISOINNIN MÄÄRITELMÄ	7
2.1 Mitä lokalisointi on?	7
2.2 Globalisoinnin infrastruktuuri.....	9
2.2.1 Globalisointi	10
2.2.2 Kansainvälistäminen.....	10
2.2.3 Lokalisointi.....	11
2.2.4 Kääntäminen	11
3 TIETOKONEAVUSTEISET KÄÄNNÖSTYÖKALUT	12
3.1 Käännösmuistit	12
3.1.1 Käännösmuistien toimintaperiaate.....	12
3.1.2 Käännösmuistien edut ja haitat.....	14
3.2 Muut käännöstyökalut.....	15
3.2.1 Terminologiatyökalut	15
3.2.2 Tagieditorit.....	16
3.2.3 Kohdistustyökalut.....	16
3.2.4 Konekäännöstyökalut.....	17
4 LOKALISOINTIPROSESSI TEKNISESTÄ NÄKÖKULMASTA	18
4.1 Lokalisointiprojekti.....	18
4.2 Lähdemateriaalin analysointi	20
4.3 Tiedostojen valmistelu	21
4.4 Tiedostojen viimeistely	22
5 YHTEENVETO	24
LÄHTEET	26

1 JOHDANTO

Viimeisen kahden vuosikymmenen aikana digitaalisen sisällön, kuten web-sivujen, ohjelmistojen ja tietokonepelien, tuotanto on moninkertaistunut. Digitaalinen vallankumous yhdessä taloudellisen globalisaation kanssa on johtanut käännettyjen digitaalisten tekstien määrän dramaattiseen kasvuun (Jiménez-Crespo, 2009). Käyttäjät työskentelevät mieluiten sellaisen sisällön parissa, joka on heidän omalla äidinkielellään. Tarve lokalisoida sisältö monelle eri kohdeympäristölle on johtanut lokalisoituyritysten merkittävään kasvuun. Lokalisointi konseptina on varsin nuori ja sen käsitteiden määrittely on vielä kesken, mutta lokalisointi pohjimmiltaan tarkoittaa kieli- ja teknologiaosaamisen yhdistämistä sellaisen tuotteen aikaansaamiseksi, joka ylittää kulttuuriset ja kielelliset rajat. (Esselink, 2003a.)

Digitaalisen sisällön lokalisoinnissa käytetään useimmiten tietokoneavusteisia käännoistyökaluja, kuten käännoismuisteja ja terminologiatyökaluja. Niiden käyttö nopeuttaa lokalisoituprosessia merkittävästi, auttaa ylläpitämään ristiriidatonta terminologiaa ja pienentää lokalisoinnin kustannuksia. (Esselink, 2000.) Lokalisointiprojekti on monitasoinen ja monialainen prosessi. Ohjelmistojen ja dokumentaation lokalisoinnissa ja kustomointiin tarvitaan monen eri alan asiantuntijoita, kuten kieli-asiantuntijoita, lokalisointi-insinöörejä, testaajia ja projektipäälliköitä (Hau, 2008). Lokalisointi-insinööri on vastuussa lokalisoituprosessin teknisistä työvaiheista (Esselink, 2000). Tekniset työvaiheet voidaan koota kolmeen kokonaisuuteen: tiedostojen analysointiin, tiedostojen valmisteluun ja tiedostojen viimeistelyyn. Näihin kokonaisuuksiin kuuluu laaja kirjo erilaisia tehtäviä, jotka riippuvat projektikohtaisista komponenteista ja tiedostoformaateista.

Tässä tutkielmassa pyritään vastaamaan kysymykseen ”Millainen on lokalisoituprosessi teknisestä näkökulmasta?” Koska lokalisoituprosessin ymmärtäminen vaatii tietoa lokalisoinnista käsitteenä sekä tuntemusta tietokoneavusteisten työkalujen toimintaperiaatteesta, tutkielmassa tarkastellaan niitä ennen varsinaista lokalisoituprosessin käsittelyä. Tutkielman sisältöosa on jaettu johdannon ja yhteenvedon lisäksi kolmeen sisältölukuun. Luvussa 2 tarkastellaan lokalisoitua yleisellä tasolla ja selvitetään, miten lokalisoinnin käsitteet ja infra-

struktuuri on määritelty kirjallisuudessa. Luvussa 3 esitellään tietokoneavusteisia käännöstyökaluja, niiden toimintaperiaatteita ja käydään läpi, millä tavoin niitä voidaan käyttää hyödyksi lokalisoinnissa. Luvussa 4 tarkastellaan lokalisointiprosessia teknisestä näkökulmasta. Siinä käydään ensin läpi prosessin yleiskuvaus ja projektin toteuttamiseen vaadittavat erilaiset roolit, minkä jälkeen esitellään lokalisointiprosessin tekniset työvaiheet.

2 LOKALISOINNIN MÄÄRITELMÄ

Tässä luvussa tarkastellaan, miten lokalisointi on määritelty kirjallisuudessa. Ensimmäisessä alaluvussa käydään läpi lokalisoinnin tavallisimmat määritelmät, tarkastellaan sen historiaa sekä lokalisointia toimialana. Toisessa alaluvussa käsitellään globalisoinnin infrastruktuuria, jonka osa-alueeksi lokalisointi useimmiten määritellään.

2.1 Mitä lokalisointi on?

Käyttäjät ympäri maailman haluavat, että ohjelmistot ovat saatavilla heidän omalla äidinkielellään. Se ei johdu käyttäjien kansallisylpeydestä, vaan siitä, että se helpottaa ohjelmistojen omaksumista ja parantaa tuottavuutta niitä käytettäessä. Käyttäjät, jotka ymmärtävät ohjelman toiminnot läpikotaisin ovat taitavampia sen käytössä ja pystyvät paremmin välttämään virheiden tekemistä. (Hau, 2008.) Interaktiivisissa teksteissä, kuten web-sivustoissa tai ohjelmistoissa, käännettyjen tekstien ymmärrettävyys on suoraan verrannollinen siihen vuoro-vaikutukseen, joka syntyy tekstin ja sen käyttäjän välillä (Jiménez-Crespo, 2009).

On miltei mahdotonta käyttää sellaista tuotetta tai web-sivustoa, jonka ohjeet, painikkeet ja valikot ovat käyttäjälle vieraalla kielellä. Paulsonin (2001) mukaan vuonna 2001 vain alle puolet internetin käyttäjistä osasi englantia sujuvasti, ja on turvallista olettaa, että internetin leviämisen myötä tänä päivänä reilu vuosikymmen myöhemmin tilanne ei ole ainakaan parempi. Ei olekaan yllättävää, että esimerkiksi monet Aasian maat ovat selkeästi Euroopan tai Pohjois-Amerikan maita jäljessä tietotekniikan omaksumisessa työpaikoilla, kouluissa ja kotona. Monesti ihmisten, jotka haluavat käyttää tietokoneita, täytyy ensin opiskella englantia. Se on erityisen hankalaa kehittyvissä maissa, joissa on paljon ihmisiä, jotka osaavat lukea huonosti. Informaatioteknologian omaksuminen olisi tällaisissa maissa tärkeä avaintekijä kehityksen edistämiseksi. (Souphavanh & Karoonboonyanan, 2005.)

Esselinkin (2003a) mukaan lokalisointi tarkoittaa tiivistetysti kieli- ja teknologiaosaamisen yhdistämistä kulttuuriset ja kielelliset rajat ylittävän tuotteen aikaansaamiseksi. Lokalisoinnin kantasanaa locale, suomeksi kohdeympäristö, käytetään useimmiten teknisessä kontekstissa, jossa se edustaa kielen, alueen, merkistökoodauksen sekä kulttuuristen ja muiden erityispiirteiden yhdistelmää. Lokalisointiin kuuluu tekstuaalisen sisällön kääntäminen kohdeympäristölle sekä ei-tekstuaalisen sisällön (esimerkiksi värien, kuvakkeiden ja näppäimistöasettelujen) muokkaaminen sellaiseksi, että se ottaa huomioon kohdeympäristön kulttuuriset, tekniset ja lain määrittelemät erityispiirteet (Dunne, 2006). Voidaan ajatella, että lokalisoituja tuotteita pitäisi käyttäjän näkökulmasta pysyvä pitämään alkuperäisteoksina, eikä käyttäjän kuuluisi olla eksplisiittisesti tietoinen siitä, että hän käyttää lokalisoitua ohjelmistoa (Jiménez-Crespo, 2009).

Lokalisointi toimialana on varsin nuori. Ennen 1980-luvun alkupuolta ohjelmistoyritykset eivät nähneet juurikaan tarvetta kääntää tuotteitaan usealle kielelle. Kuitenkin 1990-luvun puolivälistä eteenpäin erityisesti internetin kasvu on helpottanut tuotteiden markkinointia ja jakelua ympäri maailman. (Esselink, 2000.) Digitaalinen sisällöntuotanto on viimeisen kahden vuosikymmenen aikana lisääntynyt eksponentiaalisesti, mikä on johtanut dramaattiseen kasvuun käännetyn tekstisisällön määrässä (Jiménez-Crespo, 2009). Lokalisaatio onkin kehittynyt suureksi toimialaksi viime vuosikymmenten taloudellisen globalisaation ja informaatioteknologian kasvun myötä (Yao, 2010).

Lokalisointi on monesti merkittävä tekijä kansainvälisen tuotteen myönteisessä vastaanottamisessa ja menestyksessä (Esselink, 2000). Monien tuotteiden kohdalla ohjeistuksen ja ohjelmiston lokalisointia voidaan pitää kriittisenä osana tuotteen menestystä. Joissain tapauksissa voidaan jopa todeta, että jos ohjelmistoa ei ole lokalisoitu useille kielille, sitä on turha edes julkaista (Sprung & Jaroniec, 2000). Kun englanninkielisellä alueella toimiva yritys lokalisoi web-sivustonsa sisällön kiinaksi, sen on mahdollista tavoittaa tuotteelleen paljon suurempi markkina-alue kuin pelkällä englanninkielisellä sisällöllä. Kun tehdään monikielisten web-sivustojen ylläpitäminen ja luominen mahdollisimman helpoksi sellaisille ihmisille, jotka eivät ole edistyneitä tietotekniikan käyttäjiä, mahdollisuudet globaaliin kommunikaation lisääntyvät merkittävästi (Huang & Tilley, 2001).

Lokalisaatio saattaa tuottaa yritykselle suuria säästöjä. Ohjelmistojen tukipalveluiden tarvitsijoista, ja samalla kustannusten aiheuttajista, jopa puolet saattaa olla sellaisia käyttäjiä, jotka eivät ole ymmärtäneet englanninkielistä dokumentaatiota kunnolla (Watkins, 2002). Liiketoimintanäkökulmien lisäksi monet yritykset lokalisoivat tuotteitaan lakiteknisistä syistä. Monissa maissa sellaisten tuotteiden maahantuonti on kiellettyä, joita ei ole käännetty maan viralliselle kielelle. Useissa maissa laki edellyttää vähintäänkin käyttöoppaan kääntämistä. (Esselink, 2000.)

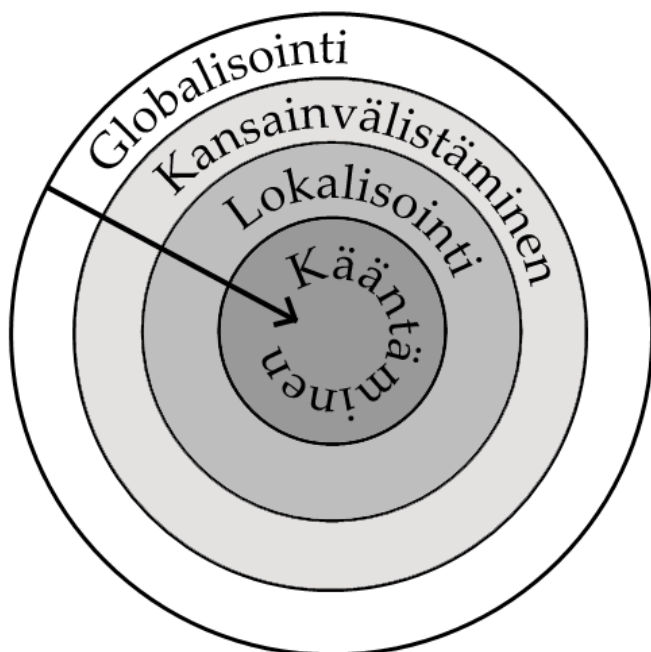
Lokalisointi ei kuitenkaan aina ole yksiselitteisen hyvä ja helposti toteutettavissa oleva asia. Ohjelmistojen lisääntyvä monimutkaisuus vaikeuttaa niiden lokalisointia. Se hankaloittaa uusien lokalisointiyritysten tuloa markkinoille ja lisää lokalisointiprosessin haastavuutta. (Cyr & Lew, 2003.) Lokalisointi on

myös tietoturvakysymys: ohjelmistojen joustavat tekstuaalisen sisällön syöttö- ja tulostusmenetelmät lisäävät tietoturvahyökkäysten riskiä (Abufardeh & Magel, 2009).

2.2 Globalisoinnin infrastruktuuri

Lokalisointi on kasvanut vuosien varrella merkittäväksi teollisuudenhaaraksi. Siitä huolimatta alan käsitteistö on vielä jossain määrin epäselvää. Raja esimerkiksi kansainvälistämisen ja lokalisoinnin välillä ei ole selkeä ja teorioiden muodostaminen akateemisella tasolla on kesken (O'Hagan, 2009). Tavallisimman määrittelyn mukaan lokalisointi on osa globalisointia: useimmiten voidaan ajatella ohjelmistojen mukauttamisen alkuperäisestä kohdeympäristöstä jollekin toiselle koostuvan neljän eri alakäsitteen kokonaisuudesta, jotka ovat globalisointi, kansainvälistäminen, lokalisointi ja kääntäminen.

Käsitteiden keskinäisiä suhteita voidaan havainnollistaa kuvion 1 avulla. Termien keskinäinen riippuvuus kulkee ympyrän ulkoreunasta sisäänpäin: globalisointi käsitteenä sisältää kansainvälistämisen, kansainvälistäminen lokalisoinnin ja lokalisointi kääntämisen.



KUVIO 1 Globalisoinnin infrastruktuuri

Dunnen (2006) mukaan tietoisuus näistä käsitteistä ja niiden merkityksestä kulki kuitenkin historiallisesti ympyrän sisältä ulospäin. Varhaiset pyrkimykset

muokata ohjelmistoja ja niiden dokumentaatiota kieleltä toiselle johtivat ymmärrykseen, että pelkällä kääntämisellä ei saada aikaan luonnollisesti toiseen kohdeympäristöön soveltuvaa tuotetta, mikä johti lokalisoinnin syntymiseen. Lokalisointipyrkimyksiä myötä havaittiin, että jo ohjelmistojen suunnitteluvaiheessa pitäisi ottaa huomioon asioita, jotka helpottaisivat lokalisointiprosessia, minkä myötä kansainvälistäminen syntyi. Tarve saattaa koko ohjelmiston kehityskaari sellaiseksi, että ohjelmistojen julkaisu monella kielellä ja eri alueilla olisi mahdollisimman helppoa, syntyi yritystason huomiosta, että lokalisoitujen ohjelmistojen mahdollinen tuottavuus on paljon suurempi.

2.2.1 Globalisointi

Esselinkin (2000) mukaan Localisation Industry Standards Association määrittelee globalisoinnin sellaisten liiketoiminnallisten näkökulmien huomioonottamiseksi, jotka liittyvät tuotteen saattamiseen maailmanlaajuisiksi. Korkean teknologian tuotteissa se tarkoittaa lokalisoinnin ja kansainvälistämisen integroimista kaikkiin yrityksen osa-alueisiin. Watkinsin (2002) mukaan globalisointi tarkoittaa tuotteen tekemistä sellaiseksi, että pienillä muutoksilla se voidaan myydä missä päin tahansa maailmaa. Globalisointi liittyy keskeisesti markkinointiin, ja se on erityisen tärkeä asia kulutustuotteiden kohdalla.

Globalisointia käytetään kuitenkin terminä monella tavalla. Yleisimmällä ja tunnetuimmalla tasolla globalisointi merkitsee, että yritykset laajentuvat toimimaan maailmanlaajuisesti. Tiukemman määrittelyn mukaan se tarkoittaa yritysten levittäytymistä fyysisesti useisiin maihin perustamalla maatoimistoja. Kapeimmin määritettynä se tarkoittaa yksittäisen ohjelmiston tai web-sivuston saattamista monelle eri kielelle. (Esselink, 2000.) Mazurin (2009) mukaan on suositeltavaa määritellä globalisointi vain sen tunnetuimmalla tasolla sekaannuksien välttämiseksi. Hänen mukaansa se pitäisi määritellä tilanteena, jossa yrityksellä on toimintoja kotialueensa ulkopuolella ja se markkinoi tuotteitaan useassa eri maassa.

2.2.2 Kansainvälistäminen

Esselinkin (2000) mukaan Localisation Industry Standards Association määrittelee kansainvälistämisen prosessina, jossa tuote tehdään suunnittelutasolla niin yleispäteväksi, että se pystytään muokkaamaan monille kielille ja kulttuurien asettamille vaatimuksille ilman tuotteen rakenteellista uudelleensuunnittelua. Kansainvälistämiseen sisältyy kulttuuristen ja kielellisten osien eristäminen muusta tuotteesta riippumattomaksi kokonaisuudeksi (Dagiene & Laucius, 2004).

Watkins (2002) määrittelee kansainvälistämisen tuotteen suunnittelemiseksi teknisestä näkökulmasta sellaiseksi, että se pystytään lokalisoimaan mahdollisimman helposti ja tehokkaasti. Se voi tarkoittaa esimerkiksi painikkeiden suunnittelemista sellaisiksi, että niihin mahtuu alkuperäiskieltä pidempi teksti tai sitä, että ohjelmisto tukee monenlaisia merkistökoodauksia. Kansainvälistä-

minen on esivaatimus lokalisoinnille ja sen tarkoitus on helpottaa ja nopeuttaa lokalisointia sekä pienentää sen kustannuksia. (Esselink, 2003a). Ihannetapauksessa suunnittelijat ottavat kansainvälistämisen huomioon alusta lähtien, eli he pyrkivät olemaan koodaamatta tekstuaalisia ja muita kieli- tai kulttuurispesifejä elementtejä siten, että niitä on vaikea muokata toisiin kohdeympäristöihin (Wang, Zhang, Xie, Mei & Sun, 2009).

2.2.3 Lokalisointi

Esselinkin (2000, 4) mukaan Localisation Industry Standards Association määrittelee lokalisoinnin tuotteen saattamisena kielellisesti ja kulttuurisesti soveliaaksi kohdeympäristöön, jossa sitä myydään ja käytetään. Watkinsin (2002) määritelmän mukaan lokalisointi tarkoittaa tuotteen mukauttamista sellaiseksi, että käyttäjät saavat sellaisen vaikutelman, että ohjelma on suunniteltu nimenomaan heidän kohdeympäristölleen. Lokalisointiin sisältyy tuotteen ominaisuuksien lisääminen tai poistaminen kohdeympäristön vaatimusten mukaan, käyttöliittymän kääntäminen, muuntyyppisen datan (kuten kuvien ja äänien) kääntäminen, päivämäärien, numeroiden ja rahayksiköiden esitystavan muokkaaminen sekä sen testaaminen, että tuote toimii näiden muokkauksien jälkeen odotetulla tavalla (He, Bustard & Liu, 2002).

2.2.4 Kääntäminen

Kääntäminen on prosessi, jossa teksti muutetaan yhdeltä kirjoitetulta kieleltä toiselle. On olennaista, että lähdemateriaalin sisältö saadaan käännettyä tarkasti ja alkuperäinen viesti säilyttäen, mutta kuitenkin siten, että huomioidaan kohdeympäristön kulttuurilliset tyyli- ja yksityiskohdat (Mazur, 2009). Kääntäminen on hyvin olennainen osa ohjelmistojen lokalisointia, ja lokalisointiprojektin budjetista jopa 80 prosenttia saattaa kuluu käännökseen toteuttamiseen (Esselink, 2000).

Perinteiseen kääntämiseen verrattuna kääntäminen lokalisointiprojektissa sisältää materiaalin analysointia, aikataulutusta ja budjetointia, päivitysten prosessointia ja muita ylimääräisiä toimintoja (Esselink, 2000). Liiketoimintamallin näkökulmasta katsottuna kääntäminen on vain osa lokalisointia, koska lokalisointiin kuuluu huomattavasti pelkkää kääntämistä laajempi kirjo prosesseja (Biau Gil & Pym, 2006).

3 TIETOKONEAVUSTEISET KÄÄNNÖSTYÖKALUT

Käännöstyön teknisiä apuvälineitä kutsutaan tietokoneavusteisiksi käännöstyökaluiksi, ja niiden toimintaperiaatteen tuntemus on edellytys lokalisointiprosessin ymmärtämiselle. Tietokoneavusteisista käännöstyökaluista olennaisin on käännösmuisti. Kaikki muut käännöstyökalut liittyvät sen toimintaan tavalla tai toisella. Sen vuoksi tässä luvussa käydään käännösmuistin toimintaperiaate läpi mahdollisimman huolellisesti ja muut tietokoneavusteiset käännöstyökalut pintapuolisemmin.

3.1 Käännösmuistit

Biau Gilin ja Pymin (2006) mukaan teknologian hyödyntäminen nykypäivän käännöstyössä on välttämätöntä. Tietokoneavusteisista käännöstyökaluista puhuminen tuntuu heidän mukaansa hieman hölmöltä, koska kääntämisessä käytetään nykypäivänä käytännössä aina apuna tietokoneita. Vaikka käännösosalalla painotetaan sitä, että nopeat muutokset teknologiassa ovat arkipäivää ja niihin täytyy sopeutua, käännösmuistiohjelmistojen toimintaperiaate on pysynyt valtaosin muuttumattomana 1990-luvun alun MS-DOS-ohjelmistoista lähtien (Reinke, 2013). Burns (1998) mukaan on arvioitu, että jopa 40–80 prosenttia lokalisointiprojektien lähdeteksteistä on toisteista. Toisteista materiaalia käännettäessä muistiohjelmat lisäävät kääntäjien tuottavuutta huomattavasti ja auttavat ylläpitämään ristiriidatonta terminologiaa sekä kirjoitustyyliä.

3.1.1 Käännösmuistien toimintaperiaate

Käännösmuistiohjelmit ovat laajimmin käytettyjä ohjelmistoja digitaalisen informaation lokalisoinnissa. Niiden ydinosan eli käännösmuistin toimintaperiaate on varastoida alkuperäiskielinen ja sitä vastaava käännetty teksti tietokantaan. (Reinke, 2013.) Käännösmuistin toiminta pohjautuu lähdekielisen tekstin

jakamiseen käännösyksiköiksi. Käännösyksikkö on tekstielementti, jonka kään-
nösmuistiohjelma on määritellyt pienimmäksi mahdolliseksi käännettäväksi
tekstiyksiköksi, joka on useimmiten yksi virke. (Esselink, 2000.)

Kun kääntämisprosessin aikana aletaan käsitellä alkuperäiskielistä kään-
nösyksikköä, käännösmuistiohjelma tarkistaa tietokannasta, onko sinne aiem-
min tallennettu identtisiä tai samankaltaisia käännösyksiköitä. Jos on, niin oh-
jelma lisää kohdekielisen vastineen käännettävään dokumenttiin automaattises-
ti. (Biau Gil & Pym, 2006.) Kääntäjä voi hyväksyä ohjelman löytämän käännök-
sen, muokata sitä, tai luoda kokonaan uuden käännöksen. Kääntäjällä on siis
tekstikokonaisuus koko ajan hallinnassa, eikä hänen tarvitse välttämättä hyväk-
syä muistista tulevia käännösyksiköitä sellaisenaan. (Burns, 1998.) Kun kääntäjä
on hyväksynyt käännösmuistista tulleen tekstin, muokannut sitä tai luonut läh-
detekstile täysin uuden vastineen, käänös voidaan tallentaa välittömästi tie-
tokantaan (Esselink, 2003b). Sen vuoksi käännösmuistista on hyötyä myös sil-
loin, kun ei ole olemassa valmista tietokantaa dokumentin aiemmista versioista
tai muuten samankaltaisesta materiaalista. Käännösmuistityökalu auttaa myös
käännöksen ristiriidattomuudessa. Jos tekstissä on useita samankaltaisia lähde-
kielisiä käännösyksiköitä, se hakee muistista jo aiemmin saman dokumentin
sisällä käännetyn vastineen (Watkins, 2002).

Käännösmuistista löytyvät vastineet voidaan jakaa kahteen luokkaan:
täydellisiin ja osittaisiin vastineisiin. Täydellinen, eli sataprosenttinen vastine,
tarkoittaa sitä, että dokumentissa oleva lähdekielinen käännösyksikkö on täysin
sama kuin tietokantaan tallennettu. Osittainen vastine tarkoittaa sitä, että kään-
nösmuistiin on tallennettu dokumentissa olevaan käännösyksikköön verrattuna
samankaltainen, muttei kuitenkaan täysin yhtäläinen vastine. (Mazur, 2009.)

Täydellisten vastineiden tapauksessa käänös on useimmiten turvallista
hyväksyä sellaisenaan. Joissain tapauksissa on kuitenkin otettava huomioon
konteksti. Lyhyiden käännösyksiköiden kohdalla se on erityisen tärkeää, koska
ne ovat monesti avoimia monitulkintaisuudelle. Esimerkki monitulkintaisuus-
desta löytyy vaikkapa tämän tutkielman aihepiiristä: "Ohjelma käännettiin" voi
tarkoittaa sekä ohjelmasisällön kielellistä kääntämistä että ohjelman kokoamista.
Kontekstista riippuen siis oikeellinen englanninkielinen käänös voi olla usean-
lainen, joten täydellistä vastinetta ei aina voi hyväksyä sellaisenaan. Jos muistin
sisältö periytyy jostain toisesta projektista, täydellisen vastineen sisältämä ter-
minologia kannattaa aina tarkistaa.

Osittainen vastine on käännösmuistista löytyvä alkuperäiskielistä tekstiä
muistuttava, muttei kuitenkaan aivan samanlainen käännösyksikkö, esimerkik-
si lause, jossa vain yksi sana on muuttunut. Osittaisia vastineita täytyy muokata,
jotta ne saadaan vastaamaan lähdetekstiä. Muokkauksen jälkeen osittaiset vas-
tineet tallentuvat käännösmuistiin täydellisinä vastineina. (Esselink, 2003b.)
Oletetaan, että dokumentissa on käännettävä lause "Koira juoksi tien yli.". Esi-
merkkinä osittaisesta vastineesta sille voi olla vaikka käännösmuistista löytyvä
lause "Kissa juoksi tien yli." ja sen englanninkielinen käänös "A cat ran across
the road.". Käännösmuistista voidaan lisätä osittainen vastine käännettävään

dokumenttiin, ja kääntäjän tehtäväksi jää muuttaa siitä vain "A cat" muotoon "A dog".

Käännösmuistien tehokkaan toiminnan avaintekijä on hyvä malli osittaisten vastaavuuksien määrittelyyn (Somers, 2003). Osittaisten vastineiden yhtäläisyyttä lähdetekstin kanssa mitataan yleensä prosentuaalisella arvolla: 95-prosenttinen vastaavuus on hyvin samankaltainen lähdetekstin kanssa, kun taas 70-prosenttisessa vastineessa saattaa olla paljonkin eroavaisuuksia. Kaikissa järjestelmissä vastaavuuksien määrittely perustuu ensisijaisesti vastaavuuksien kirjain- ja merkkitasolla, mutta monet järjestelmät mahdollistavat myös esimerkiksi muotoiluerojen ja tiettyjen sanojen merkittävyyden sisällyttämisen määrittelyyn. (Garcia, 2006.)

3.1.2 Käännösmuistien edut ja haitat

Esselinkin (2000) mukaan käännösmuistijärjestelmien käyttämisessä apuvälineinä lokalisointiprojekteissa on monia etuja. Ohjelmistojen dokumentaatio on usein toisteista, ja kun virke on käännetty kerran, käännös lisätään dokumenttiin automaattisesti, kun kääntäjä kohtaa saman virkkeen seuraavan esiintymän. Ohjelmistoja myös tavallisesti päivitetään usein ja olemassa olevia käännöksiä voidaan kierrättää ohjetiedostojen tai käyttöoppaiden uusissa versioissa. Käännösmuistien käyttö mahdollistaa myös käännösprosessin aloittamisen jo ennen lähdetekstin lopullisen version valmistumista. Olemassa olevia käännösmuisteja voidaan käyttää myös samankaltaisten uusien tuotteiden käännösten pohjana. Mitä pidempään kääntäjä työskentelee saman käännösmuistitietokannan kanssa, sitä hyödyllisempi siitä tulee (Garcia, 2006).

Useat kääntäjät voivat työskennellä samanaikaisesti yhden projektin parissa, eikä heidän tarvitse opetella useiden monimutkaisten ohjelmien käyttöä. Yleensä käännösmuistiohjelman avulla dokumentit konvertoidaan standardeihin tiedostoformaatteihin ja itse kääntäminen tapahtuu esimerkiksi Wordissa tai käännösmuistiohjelmiston omassa tekstinkäsittelytyökalussa. Tiedostokonvertoinnin avulla pystytään suojaamaan käännettävän tekstin ulkopuolinen sisältö kuten ohjelmointikoodi tai rakenteisen dokumentaation tagit muutoksilta, jolloin kääntäjä ei pysty vahingoittamaan tahattomasti dokumentin rakennetta. Kaiken tämän ansoista kääntäjien tuottavuus lisääntyy keskimäärin 30–50 prosenttia ja projektin kulut voivat laskea jopa 15–30 prosenttia. (Esselink, 2000.)

Käännösmuistien käyttö ei kuitenkaan ole täysin ongelmaton. Suurin osa käännösmuistityökaluista ei näytä dokumentin muotoilua tai taittoa, jonka vuoksi saattaa olla vaikeaa arvioida, miltä teksti näyttää lopullisessa dokumentissa. Käännöksiin tehdään myös usein viime hetken muutoksia ilman käännösmuistin apua, jolloin käännösmuistien pitäminen ajan tasalla muuttuu hankalaksi. Sen lisäksi tiedostojen konvertointi käännöstyökalujen ymmärtämään muotoon lisää aikaavieviä työvaiheita lokalisointiprosessiin. Tiedostokonversiofilterit eivät aina ole ajan tasalla eivätkä tue tiedostomuotojen uusimpia versioita, jolloin ne eivät välttämättä tunnista käännettäväksi tarkoitettua osaa tekstistä tai käsittävät esimerkiksi tagit tai muotoilut käännettäväksi tekstiksi.

Konvertointifilttereiden luominen sellaisille tiedostoformaateille, joita käännös-ohjelma ei tue oletuksena, voi olla hankala tehtävä. Käännösmuistien käyttö myös rajoittaa kääntäjien mahdollisuutta muokata tekstin rakennetta rajattomasti. Esimerkiksi virkkeiden järjestyksen vaihtaminen kappaleen sisällä ei ole mahdollista. (Esselink, 2000.)

Lommelin (2006) mukaan ongelmana on liiketoimintanäkökulmasta myös se, että käännösmuistiohjelmat sitouttavat yrityksen yhden tuottajan ohjelmiston käyttöön ja sitä myötä pitävät tavallaan yrityksen immateriaalista tekstiomaisuutta ”panttivankina”. Järjestelmän vaihtaminen on tiedosto- ja tietokantamuotojen vuoksi vaikeaa, ja jos järjestelmän toimittaja tekee konkurssin, yrityksellä saattaa olla käsissään valtava määrä dataa järjestelmässä, jota ei enää ylläpidetä tai päivitetä vastaamaan tulevaisuuden vaatimuksia. Tätä tilannetta helpottaa kuitenkin avoimien muistitietokanta- ja käännöstiedostomuotojen, kuten OpenTagin, TMX:n ja XLIFF:n olemassaolo (Burns, 1998).

3.2 Muut käännöstyökalut

Käännösmuistin lisäksi tyypilliseen käännösohjelmistoon sisältyy Reinken (2013) mukaan muun muassa seuraavat komponentit:

- Tekstinkäsittelyohjelma, jolla pystytään lukemaan lähdeteksti ja kääntämään se kohdekielelle.
- Terminologianhallintaohjelma, jolla ylläpidetään termitietokantaa, jonne tallennetaan aihepiiri-, asiakas- ja projektiriippuvaiset terminologiat.
- Tilastointiohjelma, jolla voidaan selvittää käännösmuisteista löytyvän uudelleenkäytettävän tekstin määrä kun käännetään uutta lähdedokumenttia.
- Kohdistustyökalu, jolla saadaan aiemmin ilman käännösmuistia käännetty materiaali sellaiseen muotoon, että siitä voidaan luoda käännösmuisti.

Tekstinkäsittelyohjelmasta käytetään tässä luvussa esimerkkinä tagieditoria. Reinken määrittelemien komponenttien lisäksi käsitellään lyhyesti konekäännöstyökaluja, koska käännösmuistit sekoitetaan tavallisesti niihin. Tilastointiohjelmaa käsitellään myöhemmin tutkielman luvussa 4 tiedostojen valmistelun yhteydessä.

3.2.1 Terminologiatyökalut

Terminologiatyökalut on yleensä integroitu käännösmuistijärjestelmien kanssa ja niitä voidaan käyttää kaksikielisten sanastojen sekä monikielisten termitieto-

kantojen luomiseen ja ylläpitämiseen (Watkins, 2002). Terminologiatyökalujen käyttäminen helpottaa kääntäjiä ylläpitämään ristiriidattomuutta terminologiassa. Se helpottaa käännettyjen dokumenttien lukemista ja ehkäisee väärintymärryksiä. (Esselink, 2003b.)

Terminologiaprosessi käsittää termien tunnistamisen, prosessoinnin ja määrittelyn. Termien tunnistaminen on suuri osa mitä tahansa käännösprojektiä. Useimmat lokalisointiyritykset ylläpitävät tietokantoja ydintermologiasta, jotka sisältävät tiettyjen tuotteiden käyttämät perustermit monella kielellä sekä niihin liittyvää metatietoa. Kohdatessaan avaintermejä käännettävässä tekstissä kääntäjät hakevat vastineita sekä termitietokannoista että kaksikielisistä sanastoista, jotka voivat sisältää käyttöliittymään, käyttöjärjestelmään ja projektiin liittyvää terminologiaa. (Esselink, 2000.)

3.2.2 Tagieditorit

Tagieditorit ovat käännösmuistia hyväksikäyttäviä tekstinkäsittelyohjelmistoja, jotka on suunniteltu erityisesti rakenteisen tekstin kääntämiseen. Tagieditoreilla voidaan kääntää esimerkiksi SGML-, XML ja HTML-dokumentteja suoraan konvertoimatta niitä ensin muihin tiedostoformaatteihin. Tagieditorit suojaavat rakenteisen dokumentaation sisältämiä tageja, eli estävät kääntäjää muuttamasta niitä tarkoituksella tai tahattomasti. Esimerkiksi HTML-tiedostoista tagieditorit tunnistavat useimmiten suoraan käännettäväksi tarkoitetut tekstit, ja muiden tiedostumuotojen kohdalla on mahdollista määritellä, mikä sisällöstä on tarkoitettu käännettäväksi ja mikä ei. (Esselink, 2000.)

3.2.3 Kohdistustyökalut

Kohdistus tarkoittaa lähdetekstin ja käännetyn tekstin kohdistamista käännösyksikkö käännösyksiköltä pareiksi (Somers, 2003). Kohdistustyökaluja käytetään useimmiten silloin, kun aikaisempaa tekstiä ei ole käännetty käännösmuistin avulla. Kun olemassa olevat tekstit on kohdistettu, voidaan lopputulos viedä tietokantaan ja täten muodostaa käännösmuisti. (Esselink, 2000.)

Useimmat kohdistustyökalut tekevät pohjatyön automaattisesti. Yleensä automatiikan tekemän kohdistuksen lopputulos ei ole riittävä, vaan kääntäjän pitää muokata sitä. Jos käännös on suoraviivainen, kohdistaminen on suhteellisen yksinkertainen toimenpide. Somersin (2003) mukaan on kuitenkin useita tekijöitä, jotka tekevät kohdistamisesta vaikeaa. Ensinnäkin joskus on vaikea tunnistaa, mistä virke alkaa ja mihin se päättyy. Toiseksi yksi virke jollain kielellä ei aina vastaa yhtä virkettä toisella kielellä. Kolmanneksi kääntäjät ovat saattaneet vaihtaa virkkeiden keskinäistä järjestystä.

3.2.4 Konekäännöstyökalut

Konekäännös ja käännösmuisti sekoitetaan usein keskenään. Niiden ero on kuitenkin helppo määritellä: konekäännösjärjestelmässä tietokone kääntää tekstin kokonaisuudessaan, kun taas käännösmuisti vain tallentaa käännetyt virkkeet. Siinä missä konekäännösjärjestelmä pyrkii korvaamaan kääntäjän, käännösmuistijärjestelmä pyrkii toimimaan kääntäjän tukena käännöstehtävissä. (Esselink, 2000.) Konekäännösjärjestelmät eivät tule vielä pitkään aikaan korvaamaan ihmiskääntäjiä, koska niiden tuottama teksti on usein heikkolaatuista. Se johtuu ensisijaisesti siitä, että koneet eivät pysty käsittelemään monitulkintaisuuksia tai ymmärtämään pitkiä ja monimutkaisia virkkeitä. (Mazur, 2009.)

Hyödyllisintä onkin käyttää konekäännöstä ylimääräisenä apuvälineenä lokalisointiprosessissa, sen sijaan, että pyrittäisiin tekemään sen käytöstä itseisarvo. Jos käännösmuistista ei löydy lähdetekstille vastaavuutta, kääntäjä voi ensin kääntää tekstin konekäännösjärjestelmällä, ja jos lopputulos osoittautuu riittävän hyväksi, kääntäjä voi muokata sitä tarpeen mukaan ja tallentaa sen käännösmuistiin tulevaa käyttöä varten. (Reinke, 2013.)

4 LOKALISOINTIPROSESSI TEKNISESTÄ NÄKÖKULMASTA

Tässä luvussa pyritään selvittämään millainen on lokalisointiprosessi teknisestä näkökulmasta. Ensimmäisessä alaluvussa tarkastellaan lokalisointiprojektia ja siihen vaadittavia rooleja yleisemmällä tasolla, jonka jälkeen käydään tarkemmin läpi lokalisointiprosessin tekniset työvaiheet. Ne on koottu kolmeen kokonaisuuteen, jotka ovat lähdemateriaalin analysointi, tiedostojen valmistelu ja tiedostojen viimeistely.

4.1 Lokalisointiprojekti

Lokalisointiprojektin menestyksekkääseen suorittamiseen tarvitaan monen eri alan asiantuntijoita, kuten kielispecialisteja, lokalisointi-insinöörejä, testaajia ja projektipäälliköitä (Hau, 2008). Projektipäällikön vastuulla on projektin etene-
misen valvominen, aikataulutus, yhteydenpito asiakkaaseen, resurssien- ja laadunhallinta sekä taloudelliset seikat. Kielispecialistit ovat vastuussa tyylioppaiden ja terminologian hallinnasta sekä kääntäjien tekemän työn tarkistamisesta. Lokalisointi-insinöörit ovat vastuussa lokalisointiprojektiin liittyvistä teknisistä työvaiheista, esimerkiksi tiedostojen konvertoinnista ja ohjetiedostojen koostamisesta. Lokalisointi-insinöörien ei tarvitse olla ohjelmoijia tai ohjelmistokehittäjiä. Perustiedot ohjelmoinnista, testaamisesta ja sovellusten koostamisesta riittävät monessa tapauksessa. Tarve tietämykseen ohjelmien toiminnasta kooditasolla riippuu kuitenkin paljon lokalisoitavan ohjelmiston monimutkaisuudesta sekä käytettävistä tiedostoformaateista. (Esselink, 2003b.)

Lokalisointiprojektin teknisiin työvaiheisiin kuuluu projektin arviointi ja valmistelu, ohjelmien koostaminen eli lähdekoodin kokoaminen toimivaksi ohjelmaksi, päivitysten prosessointi, kosmeettinen testaus, projektipäälliköiden ja kieliasiantuntijoiden konsultointi, kääntäjien tukeminen teknisissä asioissa, graafinen editointi ja taitto sekä monia muita asioita. Lokalisointi-insinööri käytännössä pilkkoo kehitysvaiheessa olevan ohjelmiston tai dokumentin, ottaa

siitä erilleen käännettävät osat ja niiden kääntämisen jälkeen kokoa tuotteen uudelleen. Koska lokalisoituyrityksillä on usein suuri määrä asiakkaita, lokalisointi-insinöörien pitää tuntea valtava määrä erilaisia kehitysalustoja, työkaluja ja tiedostoformaatteja. (Esselink, 2000.)

Ei olekaan yllättävää, että lokalisoituyritykset vaativat teknisiltä asiantuntijoiltaan monenlaisia taitoja. Esselinkin (2000) mukaan näitä ovat esimerkiksi yleisimpien käyttöjärjestelmien asennusympäristön ja konfiguroinnin tuntemus, kokemus ohjetiedostojen kehitysympäristöistä, laaja tietous merkkäuskielistä sekä yleisimmistä tekstieditoreista, taitto-ohjelmien käyttö sekä tyyli-tiedostojen hallinta, eri kielten vaatimien merkistöjen tuntemus, kokemus makroista ja kommentosarjakielistä sekä säännöllisistä lausekkeista, tieto lokalisaatiomalleista ja aikataulutuksesta, kokemus tietokoneavusteisista käännoistyökaluista, tieto web-teknologiasta ja yhteyskäytännöistä ja hyvät taidot vieraisissa kielissä. Lokalisointi-insinöörin toimenkuvaan voi kuulua myös esimerkiksi työkalukehitystä, konsultointia, koulutuksien järjestämistä ja paljon muuta. Toimenkuva riippuu paljon lokalisoituyrityksestä ja myös insinöörin omasta osaamisesta. Usein lokalisoituyrityksessä onkin useita teknisten työvaiheiden asiantuntijoita, joiden erityisosaamisalueet täydentävät toisiaan.

Tyypillinen lokalisoituprojekti koostuu ohjelmistokomponentin, ohjetiedoston ja painettavan materiaalin lokalisoimisesta (Esselink, 2003a). Ohjetiedosto on tyypillisesti projektin suurin lokalisoitava osa. Painetun materiaalin määrä on vähentynyt vuosien mittaan merkittävästi. Useimmiten tuotteet sisältävät painettuna ainoastaan aloitusoppaan, ja kaikki muu dokumentaatio on sähköisessä muodossa. Yksinkertaistetusti lokalisoituprosessi sisältää projektin valmistelun, sisällön kääntämisen, kielitarkistuksen, tuotannon, laadunvarmistuksen ja projektin päättämisen. (Esselink, 2003b.) Tuotteen lokalisointi on usein monimutkainen ja monia erilaisia työvaiheita sisältävä prosessi.

Teknisestä näkökulmasta erilaisia lokalisoituprosesseja on yhtä paljon kuin erilaisia projekteja ja lokalisoitavia tiedostoformaatteja. Jokaisella projektilla formaatilla on omat erityispiirteensä, jotka täytyy ottaa huomioon tiedostoja käsitellessä. Monet prosessien vaiheet ovat kuitenkin universaaleja, ja ymmärtämällä yhdenkaltaisen sisällön lokalisoituprosessi, on mahdollista saada peruskäsitys siitä, mitä lokalisoinnissa teknisestä näkökulmasta tapahtuu. Yaon (2010) mukaan ohjelmistojen lokalisoituprosessin teknisiin työvaiheisiin kuuluu muun muassa käännettävän sisällön erottelu, ei-käännettävän sisällön suojaaminen, käännoistiedostojen teknisen laadun tarkistaminen, käyttöliittymän tai dokumentin ulkoasun muokkaaminen ja ohjelmiston koostaminen. Lokalisoituprosessin tekniset työvaiheet voidaan jakaa kolmeen eri osa-alueeseen, jotka ovat lähdemateriaalin analysointi, tiedostojen valmistelu ja tiedostojen viimeistely.

4.2 Lähdemateriaalin analysointi

Kun lokalisoitavat tiedostot saapuvat asiakkaalta, ne täytyy ensin analysoida. Lähdemateriaalin analysointi on tärkein vaihe tiedostojen valmistelussa. Analyysivaiheen tarkoitus on havaita mahdolliset ongelmat lähdemateriaalissa ennen käännettä. (Esselink 2003b.) Tyypillinen ohjelmiston lokalisointiprojektin tiedostosetti sisältää Esselinkin (2000) mukaan toimivan version ohjelmistosta alkuperäiskielellä, ohjelman kehitysympäristön, jossa on mukana lähdekooditiedostot, dokumentaation taittotiedostot ja mahdolliset aiemmin lokalisoidut versiot tuotteesta sekä sanastot ja käännösmuistit. Käytännössä tiedostosetti voi kuitenkin sisältää paljon muutakin. Sisältö riippuu paljolti siitä, mitä työtehtäviä asiakas on ulkoistanut lokalisoinnin toimittajalle. Asiakas voi esimerkiksi haluta itse koostaa ohjelmiston, jolloin lokalisointiin toimitetaan vain ohjelmistosta eriteltyt tekstitiedostot.

Yaon (2010) mukaan on olemassa kolme erilaista metodologiaa ohjelmiston lokalisoinnille: koostamisenaikainen, linkityksenaikainen ja suorituksenaiikainen lokalisointi. Koostamisenaikainen metodologia vaatii sovelluksen muuttamista lähdekooditasolla. Linkityksenaikaisessa metodologiassa käytetään yhtä versiota lähdekoodista, mutta linkitetään siihen erilaisia ohjelmakirjastoja eri lokaaleille. Suorituksenaiikaisessa metodologiassa käytetään yhtä kansainvälistettyä versiota tuotteesta, joka voi käyttää erilaisia dynaamisia ohjelmakirjastoja tai käyttäjän määrittelemiä paikallisia resurssitiedostoja. Lokalisointitoimittajalle päätyvät tiedostot riippuvat siis paljolti myös siitä, millä tavalla ohjelmistokehittäjä on päättänyt integroida monikielisyys-tuen ohjelmistoonsa.

Esselinkin (2000) mukaan lokalisoitavalle tiedostosetille kannattaa tehdä ainakin seuraavat tarkistukset:

- Onko tiedostosetti täydellinen, eli sisältääkö se kaikki tiedostot, jotka tarvitaan sovelluksen lokalisointiin ja koostamiseen?
- Onko tarvittava ohjeistus mukana?
- Ovatko kaikki tiedostot valideja ja viruksettomia?
- Onko tiedostosetissä mukana tuntemattomia tiedostoformaatteja tai sellaisia tiedostoja, joita ei pysty avaamaan?
- Sisältääkö tiedostosetti tiedostoja, jotka ovat relevantteja projektipäällikön kannalta, kuten aikatauluja tai sanamääräanalyysseja?
- Sisältääkö tiedostosetti tiedostoja, jotka ovat tärkeitä kääntäjille, kuten tuotteen aiemmin lokalisoituja versioita tai sanastoja?

Analyysivaiheessa kannattaa myös testata, koostuvatko asiakkaan toimitamat lähdekooditiedostot oikein. Potentiaaliset ongelmat on hyvä saada selville ennen projektin loppuvaiheessa tapahtuvaa varsinaista koostamista. Testi-

koostaminen kannattaa tehdä pseudokäännöksellä, joka sisältää esimerkiksi kiinankielistä tekstiä, että nähdään, tuottavatko alkuperäiskielestä poikkeavat merkistöt ongelmia. Jos tiedostot on rakenteista dokumentaatiota, on syytä tarkistaa ovatko tiedostot valideja ja hyvin muodostettuja. Validiteetilla tarkoitetaan sitä, että tiedosto on dokumenttityypin mukainen ja hyvin muodostetulla tiedostolla sitä, että dokumentin sisältämät tagit eivät mene ristiin eikä esimerkiksi niiden lopetuselementtejä puutu (Esselink, 2000). HTML- ja XML-tiedostoille on olemassa erilaisia tarkistustyökaluja, joilla pystytään havaitsemaan useimmat ongelmat tiedostojen rakenteissa.

4.3 Tiedostojen valmistelu

Kun lähdemateriaali on analysoitu, mahdolliset ongelmanaiheuttajat havaittu ja testikoostaminen tehty, lähdetiedostot voidaan valmistella kääntöön. Esselinkin (2000) mukaan tiedostojen valmisteluvaiheessa on tärkeää ottaa huomioon seuraavat seikat:

- Mitkä tiedostot pitää kääntää?
- Onko käännösmuisteissa valmiita käännöksiä, joita voidaan hyödyntää?
- Mikä on tehokkain lähestymistapa ja tiedostomuoto käännökselle?
- Täytyykö tiedostoihin lisätä merkintöjä tai kommentteja kääntäjiä varten?
- Kuinka ohjelmistokoodi voidaan suojata muutoksilta?

Jos lähdemateriaali on valmisteltu hyvin, kääntäjien on helpompi työkennellä ja voidaan välttyä jo aiemmin käännetyn tekstin turhalta uudelleenkäännökseltä. Kunnollinen valmistelu on tärkeää myös siksi, että monikielisisä projekteissa mahdolliset ongelmatapaukset kertautuvat kohdekielien määrällä. (Esselink, 2003b.) Valmistelussa kannattaa siis ottaa huomioon se, monelleko eri kielelle lähdetiedostot käännetään. Mikäli kohdekieliä on vain yksi, käännettävien tekstien kopioiminen yksi kerrallaan paikalleen saattaa olla nopein ja tehokkain tapa kokonaistyömäärän kannalta. Jos samat lähdetiedostot käännetään kymmenelle eri kielelle, kannattaa todennäköisesti luoda makro helpottamaan tiedostojen loppukäsittelyä ja valmistella tiedostot sen mukaisesti.

Käännettävän tekstin pitää olla sellaisessa muodossa, joka on yhteensopi- va käytettävän käännöstyökalun kanssa ja kääntäjälle mahdollisimman selkeä ja helppokäyttöinen (Watkins, 2002). Kääntäjillä ei tyypillisesti ole teknistä taustaa, joten käännettävän tekstin muokkaaminen lähdekoodin seassa voi johtaa virheisiin ja ei-toivottuihin muutoksiin ohjelmakoodissa ja tageissa (Parr, 2006). Sen vuoksi tiedostot pitää konvertoida sellaiseen muotoon, että kääntäjät eivät pysty muuttamaan käännettävän tekstin ulkopuolista sisältöä (Watkins, 2002).

Käännösmuistiohjelmistot sisältävät tyypillisesti apuvälineitä, joilla tiedostot voidaan konvertoida käännösmuistien ymmärtämään muotoon. Näitä apuvälineitä kutsutaan tiedostokonversiofilttereiksi, ja ne konvertoivat lähdemateriaalin sen alkuperäisestä, usein suojatusta, tiedostoformaatista käännöstyökalun ymmärtämään muotoon. (Esselink, 2000.) Useimmissa käännösmuistiohjelmistoissa on valmiina filtrit yleisimmin käytetyille standardimuotoisille tiedostoformaateille, kuten HTML:lle, RTF:lle ja InDesignin INDD:lle. Monet ohjelmistot tarjoavat lisäksi mahdollisuuden kustomoida filtreitä, joiden avulla voidaan saattaa epästandardit tiedostoformaatit käännöstyökalun ymmärtämään muotoon. Kustomoituja filtreitä voidaan myös käyttää, jos standardista tiedostoformaatista halutaan jättää jotain tavallisuudesta poikkeavaa kääntämättä, esimerkiksi jos HTML-tiedostosta halutaan jättää otsikkoelementti alkuperäiskielelle.

Kun tiedostot on konvertoitu käännöstyökalujen ymmärtämään muotoon, on hyvä tarkistaa, onko käännösmuisteissa valmiita käännöksiä, joita voidaan hyödyntää. Kaikki käännösmuistiohjelmat sisältävät tilastointityökaluja, joilla voidaan selvittää, kuinka suuri osa materiaalista on jo aiemmin käännetty. Käännösmuistiohjelmilla on mahdollista analysoida käännöstiedosto muistia vasten ja tuottaa raportti, joka sisältää kokonaissanamäärän sekä käännösyksiköiden, tiedostojen sisäisten toistojen, täydellisten vastineiden, eritasoisten osittaisten vastineiden sekä sellaisten sanojen määrän, joille ei löytynyt käännösmuistista vastinetta. (Esselink, 2000.) Käännöstiedostot on myös mahdollista esikäntää, mikä tarkoittaa täydellisten vastineiden lisäämistä valmiiksi tiedostoihin. Kääntäjä voi tarkistaa sataprosenttiset vastineet ja parannella niitä tarvittaessa, tai ne voidaan lukita kääntäjältä, mikäli niitä ei haluta muutettavan.

4.4 Tiedostojen viimeistely

Kun tiedostot saapuvat kääntäjältä, ne pitää saattaa takaisin alkuperäiseen tiedostomuotoon ja kehitysympäristöön. Jos lähdemateriaali on valmisteltu huolellisesti, lokalisointi-insinöörin tehtäviin kuuluu Esselinkin (2000) mukaan viimeistelyvaiheessa muun muassa seuraavat asiat:

- Sen tarkistaminen, että kaikki tiedostot ovat palautuneet käännöksestä ja valideja sekä oikeissa tiedostoformaateissa.
- Sen tarkistaminen, että kaikki tarvittavat komponentit ja tekstit on käännetty.
- Tiedostojen kopioiminen kehitysympäristöön ja alkuperäiskielisten tai pseudokäännettyjen tiedostojen korvaaminen niillä.
- Ohjelmiston lokalisoidun version koostaminen, sen testaus ja ulkoasun muokkaaminen.

Kaikki materiaali ei aina valmistu samaan aikaan. Esimerkiksi käyttöoppaat sisältävät usein ruutukaappauksia, jotka sisältävät lokalisoitavaa tekstiä. Käyttöoppaan sisällä saattaa olla viittauksia ruutukaappauksiin, jolloin kontekstin ylläpitämiseksi on tarpeen ensin kääntää ohjelmisto, koostaa se ja luoda käännettyä tekstiä sisältävät ruutukaappaukset ja vasta sen jälkeen lähettää dokumentaatio kääntöön. (Esselink, 2003a.)

Käännösmuistiohjelmistot sisältävät useimmiten tarvittavat työkalut tiedostojen laaduntarkistukseen. Niillä voidaan havaita kääntämättä jääneiden käännösyksiköiden määrä sekä niiden sijainti tiedostojen sisällä. Joissain tapauksissa lokalisointi-insinöörin on mahdollista korjata kääntämättä jääneet käännösyksiköt muistityökalun avulla, esimerkiksi silloin, jos sama teksti esiintyy useaan kertaan käännöstiedostossa ja joku esiintymistä on käännetty. Useimmiten puutteelliset tiedostot joudutaan kuitenkin palauttamaan kääntäjälle korjattavaksi. Käännösmuistiohjelmiston työkaluilla voidaan havaita myös, jos tiedoston rakenne on kääntämisen aikana rikkoutunut. Jos esimerkiksi tageja on siirretty tarpeettomasti, lisätty tai poistettu, validointityökalut huomauttavat siitä. Tällaiset tapaukset on useimmiten mahdollista korjata ilman kääntäjän apua.

Kun käännöstiedosto on rakenteellisesti ja sisällöllisesti oikeellinen, se voidaan siivota, viedä sen sisältö käännösmuistiin ja konvertoida se takaisin alkuperäiseen tiedostomuotoon. Tiedoston siivoaminen lokalisointikontekstissa tarkoittaa lähdekielisten tekstien poistamista tiedostosta siten, että siihen jää jäljelle vain käännettyt tekstit (Esselink, 2000). Tiedosto konvertoidaan takaisin alkuperäiseen muotoon filterien avulla. Joidenkin tiedostomuotojen tapauksissa konvertoinnissa tarvitaan alkuperäiskielistä tiedostoa, toisten kohdalla alkuperäisen muodon saavuttamiseen riittää pelkkä käännöstiedosto.

Kun lokalisoitu sisältö on takaisin alkuperäisessä formaatissa, se viedään kehitysympäristöön, korvataan lähdekieliset tiedostot ja ohjelmisto koostetaan, tai se dokumentaation tapauksessa taitetaan siten, että sen ulkoasu on mahdollisimman yhtäläinen lähdekielisen tiedoston ulkoasun kanssa. Kummassakin tapauksessa lokalisointi-insinöörin tehtävä on varmistaa, että kaikki käännettyt tekstit näkyvät oikein muokkaamalla tekstiä sisältävien elementtien kokoa ja sijaintia tarpeen mukaan. (Esselink, 2003b.)

5 YHTEENVETO

Lokalisointi tarkoittaa kieli- ja teknologiaosaaminen yhdistämistä sellaisen tuotteen aikaansaamiseksi, joka ylittää kielelliset ja kulttuuriset rajat. Lokalisointi toimialana alkoi kehittyä 1990-luvulla digitaalisen sisällöntuotannon valtavan kasvun sekä taloudellisen globalisaation myötä. Yritykset näkivät tarpeen laajentua kotimarkkinoidensa ulkopuolelle, ja koska ihmiset ympäri maailman haluavat käyttää tuotteita mieluiten omalla äidinkielellään, globaalin menestyksen edellytys oli, että tuotteet käännettiin monille eri kielille. Käsitteellisesti lokalisointi usein ajatellaan osana globalisointia, jonka infrastruktuuri voidaan jakaa neljään osaan: globalisointi, kansainvälistäminen, lokalisointi ja kääntäminen. Globalisointi sisältää lokalisoinnin markkinointinäkökulmat, kansainvälistäminen tuotteen tekemisen suunnitteluvaiheessa sellaiseksi, että se on mahdollisimman helppo lokalisoida ja kääntäminen tekstuaalisen sisällön siirtämistä yhdeltä kieleltä toiselle.

Käytännössä kaikessa kääntämisessä käytetään nykypäivänä apuna tietokoneavusteisia käännoistyökaluja. Käännoistyökaluohjelmisto sisältää yleensä useita eri ohjelmia eri tarkoituksiin. Tärkein osa ohjelmistoa on käännösmuisti, jonka toimintaan muut työkalut tavalla tai toisella liittyvät. Muita komponentteja ovat muun muassa terminologiatyökalut, erilaiset tekstieditorit, kohdistustyökalut ja konekäännösjärjestelmät. Käännösmuistien toimintaperiaate on varastoida alkuperäiskielinen teksti ja sitä vastaava käännös tietokantaan. Tekstiä käännettäessä vastaavuuksia voidaan hakea käännösmuistitietokannasta, jolloin vältytään turhalta uudelleenkiääntämiseltä ja pidetään yllä ristiriidatonta terminologiaa. Tietokantaan tallennetut vastineet voidaan jakaa täydellisiin vastineisiin, jolloin käännösmuistista löytyvä teksti on täysin samanlainen kuin käännettävä teksti ja osittaisiin vastineisiin, jolloin teksteissä on yhtäläisyyksiä, mutta ne eivät kuitenkaan ole identtiset.

Lokalisointi on monitasoinen ja monialainen projekti. Tyypillisesti projektin menestyksekkääseen toteuttamiseen vaaditaan ainakin kieli- ja kulttuuriantuntijoita, teknisiä asiantuntijoita, testaajia ja projektipäälliköitä. Lokalisointiprojektien teknisistä työvaiheista vastuussa olevaa asiantuntijaa kutsutaan myös lokalisointi-insinööriksi. Tyypillinen lokalisointiprojekti koostuu ohjelmistokom-

ponentin, ohjetiedoston ja painettavan materiaalin lokalisoimisesta. Projekteihin saattaa kuitenkin kuulua valtavasti erilaisia komponentteja ja tiedostotyyppisiä, ja jokainen niistä sisältää omat käytäntönsä ja haasteensa.

Tutkimuksen tuloksena voidaan todeta, että lokalisointiprosessin tekniset työvaiheet voidaan jakaa kolmeen kokonaisuuteen, jotka ovat lähdemateriaalin analysointi, tiedostojen valmistelu ja tiedostojen viimeistely. Lähdemateriaalin analysointi on projektin tärkeimpiä vaiheita. Sen tarkoitus on havaita mahdolliset virheet tiedostoissa ennen niiden kääntämistä, sillä monikielisissä projekteissa virheet kertautuvat käännettävien kielten määrällä. Lähdemateriaalin analysointiin kuuluu muun muassa tiedostojen validiteetin tarkistaminen sekä testikoostaminen. Lähdemateriaalin analysoinnin jälkeen käännettävät tiedostot voidaan valmistella tiedostokonversiofilterien avulla käännoityökalujen ymmärtämään muotoon. Tavoitteena on saada aikaan sellainen tiedosto, joka olisi kääntäjälle mahdollisimman helppokäyttöinen. Tiedostosta pitää suojata sellainen sisältö kääntäjältä, jonka muuttaminen aiheuttaa tiedoston rakenteen rikkoutumisen. Käännöksen jälkeen tiedostojen oikeellisuus tarkistetaan ja varmistetaan, että kaikki tarvittava sisältö on käännetty. Tiedostot konvertoidaan takaisin alkuperäiseen tiedostomuotoon ja viedään kehitysympäristöön. Käännettujen ohjelmistojen ja dokumenttien ulkoasu tarkistetaan, jonka jälkeen lokalisointiprosessi on valmis ja lokalisoitujen tiedostot voidaan palauttaa asiakkaalle.

Lokalisointia teknisestä näkökulmasta on tutkittu akateemisella tasolla melko vähän. Aiheesta on olemassa tieteellisiä artikkeleita jonkin verran, mutta suurin osa niistä viittaa yleiskäsitteiden osalta samoihin alan perusteoksiin, eikä tuoretta lokalisoinnin teknistä puolta kartoittavaa tutkimusta ole olemassa. Siinä mielessä jatkotutkimukselle olisikin tarvetta. Perusteokset ovat jo yli kymmenen vuotta vanhoja, joten olisi aiheellista tutkia, miten käytännöt ovat muuttuneet niiden kirjoittamisen jälkeen.

LÄHTEET

- Abufardeh, S. & Magel, K. (2009). Software Internationalization: Crosscutting Concerns across the Development Lifecycle. *New Trends in Information and Service Science, NISS '09* (s. 447-450).
- Biau Gil, J.R. & Pym, A. (2006). Technology and Translation (a pedagogical overview). Teoksessa A, Pym, A. Perestrenko. & B. Starink (toim.), *Translation Technology and its Teaching* (s. 5-19). Tarragona: Intercultural Studies Group.
- Burns, W (1998). Opentag and TMX: XML in the localization industry. *Proceedings of the 16th annual international conference on Computer documentation* (s. 137-142). New York, NY: SIGDOC.
- Cyr, D. & Lew, R. (2003). Emerging challenges in the software localization industry. *Thunderbird International Business Review*, 45(3), 337-358.
- Dagiene, V. & Laucius, R. (2004). Internationalization of open source software: framework and some issues. *Information Technology: Research and Education, ITRE 2004* (s. 204-207).
- Dunne, K. J. (2006). A Copernican revolution. Teoksessa K. J. Dunne (toim.), *Perspectives on Localization* (s. 1-13). Philadelphia, PA: John Benjamins.
- Esselink, B. (2000). *A Practical Guide to Localization* (uud. painos). Amsterdam/Philadelphia: John Benjamins.
- Esselink, B. (2002). Localization Engineering : The Dream Job? *Tradumàtica. Bellaterra : Servei de Publicacions de la Universitat Autònoma de Barcelona*, 1. Haettu 2.7.2013 osoitteesta <http://www.fti.uab.es/tradumatica/revista/articles/besselink/besselink.PDF>
- Esselink, B. (2003a). The evolution of localization. *Multilingual Computing & Technology*, 14(5), 4-7.
- Esselink, B. (2003b). Localisation and translation. Teoksessa H. Somers (toim.), *Computers and Translation: A translator's guide* (s. 66-86). Philadelphia, PA: John Benjamins.
- García, I. (2006). Translators on translation memories: a blessing or a curse? Teoksessa A, Pym, A. Perestrenko. & B. Starink (toim.), *Translation Technology and its Teaching* (s. 97-105). Tarragona: Intercultural Studies Group.
- Hau, E. (2008). Software internationalization and localization in web based ERP. *Proceedings of the 26th annual ACM international conference on Design of communication* (s. 175-180). New York, NY: ACM.
- He, Z., Bustard, D. W. & Liu, X. (2002). Software internationalisation and localisation: practice and evolution. *Proceedings of the inaugural conference on the Principles and Practice of programming and Proceedings of the second workshop on Intermediate representation engineering for virtual machines, 2002* (s. 89-94). Ireland: National University of Ireland Maynooth.

- Huang, S. & Tilley, S. (2001). Issues of content and structure for a multilingual web site. *Proceedings of the 19th annual international conference on Computer documentation* (s. 103-110). New York, NY: ACM.
- Jiménez-Crespo, M. A. (2009). The evaluation of pragmatic and functionalist aspects in localization: towards a holistic approach to quality assurance. *The Journal of Internationalisation and Localisation*, 1, 60-93.
- Lommel, A. (2006). Localization standards, knowledge- and information-centric business models, and the commoditization of linguistic information. Teoksessa K. J. Dunne (toim.), *Perspectives on Localization* (s. 223-240). Philadelphia, PA: John Benjamins.
- Mazur, I. (2009). The Metalanguage of Localization. Teoksessa Y. Gambier & L. Doorslaer (toim.), *Metalanguage of Translation* (s, 145-166). Amsterdam: John Benjamins.
- O'Hagan, M. (2009). Evolution of user-generated translation: fansubs, translation hacking and crowdsourcing. *The Journal of Internationalisation and Localisation*, 1, 94-121.
- Parr, T. (2006). Web application internationalization and localization in action. *Proceedings of the 6th international conference on Web engineering* (s. 64-70). New York, NY: ICWE.
- Paulson, L.D. (2001). Translation technology tries to hurdle the language barrier. *Computer*, 34(9), 12-15.
- Reinke, U. (2013). State of the Art in Translation Memory Technology. *Translation: Computation, Corpora, Cognition*, 3(1). Haettu 9.9.2013 osoitteesta <http://t-c3.org/index.php/t-c3/article/view/25>
- Somers, H. (2003). Translation memory systems. Teoksessa H. Somers (toim.), *Computers and Translation: A translator's guide* (s. 31-48) . Philadelphia, PA: John Benjamins.
- Souphavanh, A. & Karoonboonyanan, T. (2005). *Free/Open Source Software: Localization*. India: APDIP.
- Sprung, R. C. & Jaroniec, S. (2000). *Translating Into Success: Cutting-edge strategies for going multilingual in a global age*. Amsterdam/Philadelphia: John Benjamins
- Wang, X., Zhang, L., Xie, T., Mei, H. & Sun, J. (2009). Locating need-to-translate constant strings for software internationalization. *Proceedings of the 31st International Conference on Software Engineering* (s. 353-363). Washington, DC: IEEE Computer Society.
- Watkins, J. (2002). *The Guide to Translation and Localization: Preparing Products for the Global Marketplace*. Portland: Lingo Systems.
- Yao, Y. (2010). An Ontology-Based Translation Memory Model in Localization Translation. *Third International Symposium on Information Science and Engineering* (s. 340-344). Shanghai, China.