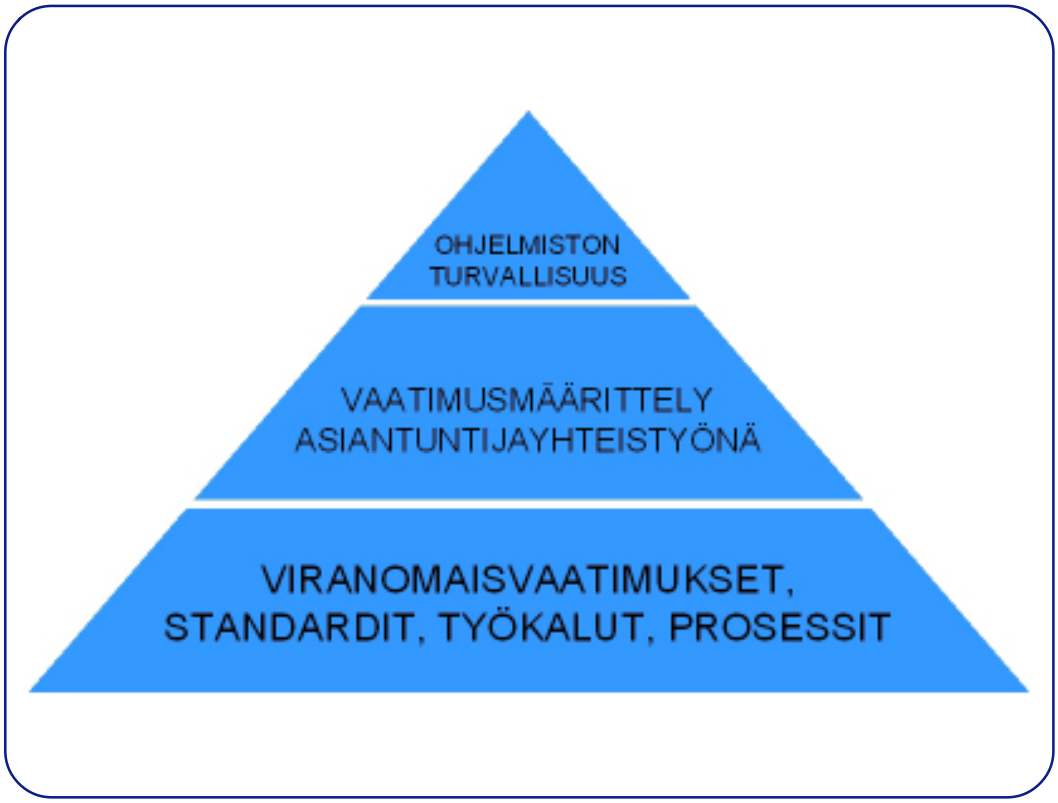
ESPOO 2004 VTT TIEDOTTEITA 2263



Ilpo Pöyhönen & Kristiina Hukki

Riskitietoisen ohjelmiston vaatimusmäärittelyprosessin kehittäminen

VTT TIEDOTTEITA – RESEARCH NOTES 2263

Riskitietoisen ohjelmiston vaatimusmäärittelyprosessin kehittäminen

Ilpo Pöyhönen & Kristiina Hukki

VTT Tuotteet ja tuotanto

ISBN 951–38–6496–0 (nid.)
ISSN 1235–0605 (nid.)

ISBN 951–38–6497–9 (URL: http://www.vtt.fi/inf/pdf/)
ISSN 1455–0865 (URL: http://www.vtt.fi/inf/pdf/)

JULKAISIJA – UTGIVARE – PUBLISHER

VTT, Vuorimiehentie 5, PL 2000, 02044 VTT
puh. vaihde (09) 4561, faksi (09) 456 4374

VTT, Bergsmansvägen 5, PB 2000, 02044 VTT
tel. växel (09) 4561, fax (09) 456 4374

VTT Technical Research Centre of Finland, Vuorimiehentie 5, P.O.Box 2000, FIN–02044 VTT, Finland
phone internat. + 358 9 4561, fax + 358 9 456 4374

VTT Tuotteet ja tuotanto, Tekniikankatu 1, PL 1306, 33101 TAMPERE
puh. vaihde (03) 316 3111, faksi (03) 316 3365

VTT Industriella System, Tekniikankatu 1, PB 1306, 33101 TAMMERFORS
tel. växel (03) 316 3111, fax (03) 316 3365

VTT Industrial Systems, Tekniikankatu 1, P.O.Box 1306, FIN–33101 TAMPERE, Finland
phone internat. + 358 3 316 3111, fax + 358 3 316 3365

Otamedia Oy, Espoo 2004

Pöyhönen, Ilpo & Hukki, Kristiina. Riskitietoisen ohjelmiston vaatimusmäärittelyprosessin kehittäminen [Development of risk-informed requirements specification process of software]. Espoo 2004. VTT Tiedotteita – Research Notes 2263. 36 s. + liitt. 9 s.

Avainsanat software requirements specification, risk-informed software engineering process, traceability, risk management, multidisciplinary expertise, expert interaction, knowledge transfer, integration of knowledge, shared thinking models

Tiivistelmä

Vaatimusmäärittely on ratkaisevan tärkeä vaihe ohjelmistoa sisältävien järjestelmien suunnittelussa. Tämän vuoksi vaatimusmäärittelyprosessin kehittämisen arvioidaan nostavan ohjelmistojen laatua merkittävästi. Raportissa tarkastellaan luotettavuus- ja turvallisuusvaatimuksia sisältävien ohjelmistojen vaatimusmäärittelyprosessin kehittämiseen liittyviä näkökohtia sekä teknisestä että prosessiin osallistuvien asiantuntijoiden vuorovaikutuksen näkökulmasta.

Kehittämiskohteiden valinta ja yhdenmukaistettujen standardien soveltuvuuden arviointi omaan toimintaan edellyttää yritykseltä kriittistä ja perinpohjaista tarkastelua. Raportissa esitetään keinoja, joiden avulla yrityksissä pystytään kehittämään yrityskohtaista riskitietoista vaatimusmäärittelyprosessia. Tarkastelun kohteena ovat standardien ja riskienhallinnan tarjoamaan tukeen perustuvat menettelytavat. Niiden avulla voidaan parantaa vaatimusmäärittelyn jäljitettävyyttä ja ohjelmiston luotettavan toiminnan kannalta kriittisten vaatimusten tunnistamista laajasta vaatimusjoukosta. Vaatimusmäärittelyn kehittämistä tukevina keinoina esitetään harmonisoitujen standardien soveltaminen suunnitteluprosessiin ja vaatimusmäärittelyyn. Tämän lisäksi ehdotetaan laadunhallintajärjestelmien ja riskienhallinnan integroimista osaksi suunnitteluprosessia.

Vaatimusmäärittelyprosessia tarkastellaan myös siihen liittyvien eri osapuolten vuorovaikutuksen näkökulmasta. Eri alojen asiantuntemusta edustavien osapuolten tiedonkululla on merkittävä vaikutus määrittelyprosessin onnistumisessa, koska ohjelmistoa koskevat vaatimukset muodostuvat niiden vuorovaikutuksen tuloksena. Raportissa käsitellään tiedonkulun merkitystä ohjelmiston turvallisuuden syntymisessä ja esitetään keinoja, joiden avulla voidaan edistää osapuolten keskinäistä ymmärrystä. Lisääntynyt keskinäinen ymmärrys helpottaa eri alojen asiantuntemuksen yhdistämistä vaatimusmäärittelyprojekteissa ja ohjelmistosuunnittelua koskevassa riskinhallinnassa.

Pöyhönen, Ilpo & Hukki, Kristiina. Riskitietoisen ohjelmiston vaatimusmäärittelyprosessin kehittäminen [Development of risk-informed requirements specification process of software] Espoo 2004. VTT Tiedotteita – Research Notes 2263. 36 p. + app. 9 p.

Keywords software requirements specification, risk-informed software engineering process, traceability, risk management, multidisciplinary expertise, expert interaction, knowledge transfer, integration of knowledge, shared thinking models

Abstract

Requirements specification is the most fundamental phase of software design. Therefore, the development of the requirements specification process is considered to contribute significantly to the quality of software. The report examines aspects related to developing the requirements specification process of safety-critical software. The consideration is made both from technical and expert interaction point of view.

The choice of the objects to be developed and the evaluation of the applicability of the harmonized standards to the company's activities require critical and thorough consideration. The report introduces procedures which help in developing the company-specific risk-informed requirements specification. The focus is on procedures which are based on the support provided by standards and risk management. These procedures make it possible to improve the traceability of requirements specification and to facilitate the identification of those requirements which are critical from the safety point of view. The suggested procedures are the application of the harmonized standards to the design process and to requirements specification, and, in addition, the integration of quality management system and risk management as part of the design process.

The requirements specification process is considered also from the viewpoint of different parties' interaction. The requirements are formed as the outcome of the experts' interaction. Therefore, the knowledge transfer between the experts, representing different disciplines, influences prominently the success of the specification process. The report examines the significance of knowledge transfer in the formation of software safety. In addition, procedures are introduced which help in enhancing the experts' mutual understanding. Enhanced mutual comprehension facilitates the integration of multidisciplinary expertise in the requirements specification projects and in the risk management of software engineering.

Alkusanat

Raportti kuuluu osana Trusted Software Technology -hankkeeseen (TRUST), jossa tutkitaan uusia menettelytapoja ratkoa turvallisuuskriittisiin ohjelmistoihin liittyviä ongelmia. Menettelytapojen avulla pyritään edistämään riskitietoisen ohjelmistotuotantoprosessin kehittämistä.

Kiitämme Risto Tuomista ja Mika Koskelaa hyödyllisistä kommenteista.

Tekijät

Sisällysluettelo

Symboliluettelo

ALARP	As Low As Reasonably Practicable
EN	European Norm, euronormi
EU	European Union, Euroopan Unioni
FMEA	Faut Modes and Effects Analysis, Vika- ja vaikutusanalyysi
FMECA	Fault Modes, Effect and Criticality Analysis, Vika-, vaikutus- ja kriittisyysanalyysi
FTA	Faut Tree Analysis, Vikapuuanalyysi
HRA	Human Reliability Analysis, Ihmisen luotettavuusanalyysi
IEC	International Electrotechnical Commission
IEEE	The Institute of Electrical and Electronics Engineers, Inc.
ISO	International Organization for Standardization
IVD	In-Vitro Diagnostic Directive 98/79/EEC
MDD	Medical Device Directive 93/42/EEC
NB	Notified Body, Ilmoitettu laitos
PHA	Preliminary Hazard Analysis, Vaara-analyysi
RAMS	Reliability, Availability, Maintainability and Safety
SFS	Suomen Standardoimisliitto r.y.
SRS	Software Requirements Specification, Ohjelmiston vaatimusspesifikaatio
TRUST	TRUsted Software Technology

1. Johdanto

Ohjelmistojen käyttö turvallisuuskriittisten järjestelmien osana on lisääntynyt nopeasti, minkä seurauksena myös tarve osoittaa tai varmistaa niiden turvallisuus on lisääntynyt. Turvallisuudella tarkoitetaan tässä yhteydessä sitä, että ohjelmisto soveltuu aiottuun käyttötarkoitukseensa, on riittävän suorituskykyinen ja toimii luotettavasti aiheuttamatta vaaroja käyttäjälle, ympäristölle, sivullisille tai sovellusaluekohtaisille kohderyhmille.

Vaatimusmäärittely on eräs tärkeimmistä osista luotettavuus- ja turvallisuusvaatimuksia sisältävien ohjelmistojen suunnittelussa. Huomattava osa ohjelmistossa esiintyvistä virheistä johtuu kuitenkin puutteellisesta vaatimusmäärittelystä. Tämän vuoksi vaatimusmäärittelyprosessin systemaattisen kehittämisen arvioidaan nostavan ohjelmistojen laatua merkittävästi. Ilman riittävän kattavia ja oikeita ohjelmistovaatimuksia ei voida laatia ohjelmistosuunnittelulta edellytettäviä riskienhallinta-, verifiointi- ja validointisuunnitelmia. Kehittämiskohteiden valinta ja yhdenmukaistettujen standardien sekä riskienhallinnan soveltaminen oman yrityksen toimintaan ei kuitenkaan ole helppoa, koska se edellyttää organisaatiolta omien tavoitteiden ja kriteerien kriittistä ja perinpohjaista tarkastelua.

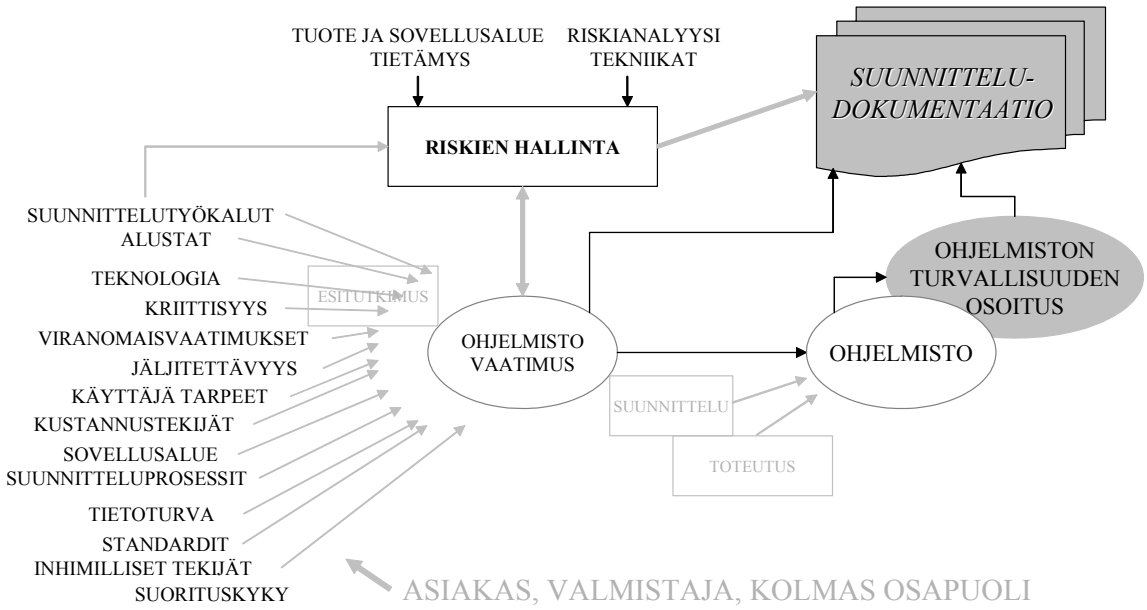
Raportissa tarkastellaan turvallisuuskriittisten ohjelmistojen vaatimusmäärittelyprosessin kehittämiseen liittyviä näkökohtia kuvaamalla ensin ohjelmistojen vaatimusmäärittelyn ongelmia. Tämän jälkeen esitetään standardien ja riskienhallinnan tarjoamia keinoja riskitietoisen vaatimusmäärittelyprosessin kehittämiseksi. Laadunhallintajärjestelmien, harmonisoitujen standardien ja riskianalyysin mahdollistamien keinojen avulla pystytään edistämään vaatimusmäärittelyprosessin systemaattisuutta ja jäljitettävyyttä ja tunnistamaan ohjelmiston luotettavan toiminnan kannalta kriittiset vaatimukset.

Vaatimusmäärittelyprosessia tarkastellaan myös siihen liittyvien eri osapuolten vuorovaikutuksen näkökulmasta. Eri asiantuntemusta edustavien osapuolten tiedonkululla on vaikutus määrittelyprosessin turvallisuuteen, koska ohjelmistoa koskevat vaatimukset muodostuvat osapuolten vuorovaikutuksen tuloksena. Raportissa käsitellään tiedonkulun merkitystä ohjelmiston toiminnallisen turvallisuuden syntymisessä. Lisäksi esitetään keinoja, joiden avulla voidaan lisätä vaatimusmäärittelyprosessiin liittyvien tahojen kokonaiskuvaa prosessista ja saada selville osapuolten näkemyseroista johtuvat puutteet tiedonkulussa. Tämän pohjalta voidaan edistää osapuolten keskinäistä ymmärrystä, mikä puolestaan helpottaa eri alojen asiantuntemuksen yhdistämistä suunnitteluprojekteissa. Vastaavien keinojen avulla voidaan myös tukea ohjelmistojen kriittisten vaatimusten tunnistamisessa tarvittavaa monialaista asiantuntijayhteistyötä suunnitteluprosessia koskevassa riskienhallinnassa.

2. Vaatimusmäärittelyn ongelmia

Vaatimusmäärittely on eräs suurimmista virhelähteistä ohjelmistosuunnittelussa. Ohjelmistoissa esiintyvistä virheistä jopa yli 50 % johtuu puutteellisesta vaatimusmäärittelystä [Kececi et. al.].

Ohjelmiston vaatimusmäärittely on erittäin vaikeaa, koska määrittelyyn vaikuttavat useat erilliset toisistaan riippumattomat tahot. Tärkeimpiä näistä ovat asiakkaan, valmistajan itsensä ja kolmannen osapuolen esittämät vaatimukset. Kolmannen osapuolen esittämät vaatimukset liittyvät usein tietyille sovellusalueille, joilla markkinoille saattaminen edellyttää ohjelmistolle jonkinlaista hyväksyntäprosessia tai lakisääteisten vaatimusten täyttämistä. Toisin sanoen kolmas osapuoli voi olla valvova viranomainen, mutta se voi olla myös esimerkiksi ohjelmiston myyjä. Lisäksi on huomioitava ohjelmiston sisäiset rajapinnat ja liitynnät, käyttöjärjestelmä ja laitteisto sekä sovellusalueen, käyttäjän ja ympäristön asettamat vaatimukset. Nämä suunnittelun lähtötietovaatimukset (ks. kuva 1) asettavat monenlaisia reunaehtoja, tavoitteita ja kriteerejä ohjelmiston vaatimusmäärittelylle ja ovat erittäin merkittäviä ohjelmiston suunnittelun kannalta.



Kuva 1. Onko kaikki määrittelyyn vaikuttavat lähtötiedot huomioitu?

Yrityksen omat vaatimukset ohjelmiston suunnittelulle ja vaatimusmäärittelylle riippuvat yrityksen laatupolitiikasta, eettisistä arvoista sekä käytössä olevasta suunnittelu- ja valmistustyökaluista. Kuinka hyvin vaatimusmäärittely saadaan onnistumaan, riippuu henkilöstön pätevyydestä, suunnittelun lähtötietojen riittävyydestä sekä vaatimusmäärittelyä tukevista riskienhallinta- ja laadunvarmistusprosesseista (huom. riskienhallinta on osa laadunvarmistusta tai se voi olla myös itsenäinen prosessi, tällöin yrityksen eri toi-

minnot kulkevat aina tämän prosessin kautta). Yleensä riskienhallinta, laadunvarmistus ja vaatimusmäärittely hallitaan parhaiten laadunhallintajärjestelmillä (ISO 9001, ISO 13485), jotka sisältävät myös vaatimukset suunnittelun dokumentoinnillekin. Osa vaatimusmäärittelyyn liittyvistä ongelmista johtuu organisaation laadunhallintajärjestelmän puutteista, puutteellisesta suunnitteludokumentaatiosta, tuotekehitysprojektin monimutkaisuudesta ja projektinhallinnan heikkoudesta sekä tiedonkulun puutteista ja näkemyseroista johtuvista ristiriidoista suunnittelutiimin eri asiantuntija- ja henkilöryhmien välillä.

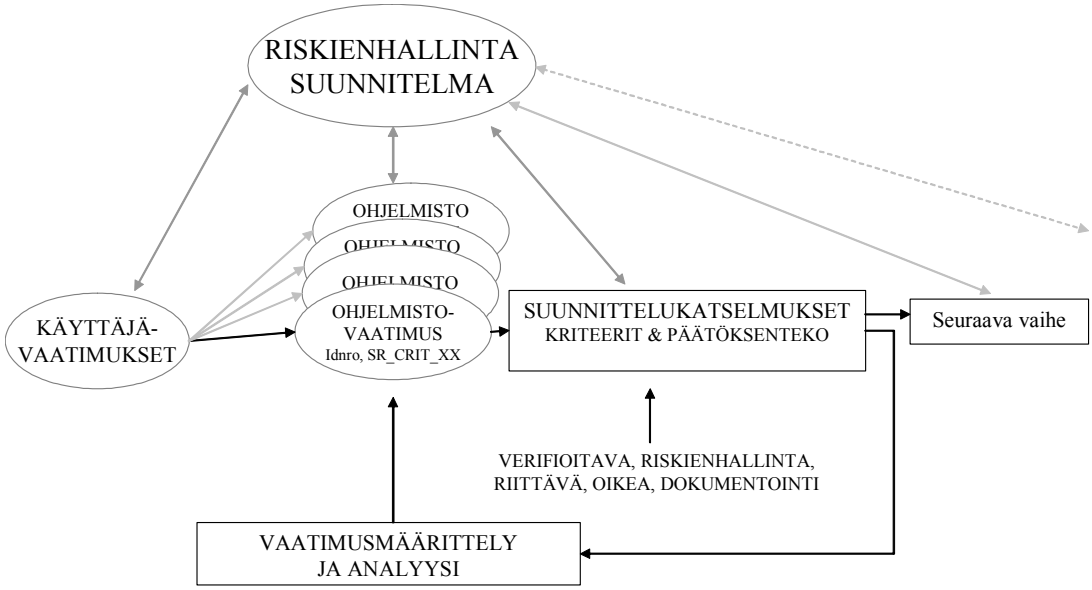
Käyttäjätarpeiden ymmärtämiseksi on tunnettava asiakkaan liiketoiminnan tarpeet ja kyseisen sovellusalueen tarpeet, jonka jälkeen voidaan määritellä ohjelmistoon liittyvät vaatimukset. Vaatimuksista tulee dokumentoida suorituskyky-, ympäristö- ja käyttöolosuhdevaatimukset sekä kaikki rajoitukset, joita ohjelmistolla ei voida toteuttaa. Kun kaikki tarvittavat käyttäjätarpeisiin liittyvät vaatimukset on dokumentoitu, voidaan näiden pohjalta laatia alustavat riskienhallinta-, verifiointi- ja validointisuunnitelmat.

Riskienhallinta ja riskianalyysit eivät ole enää uusia asioita, mutta niiden soveltaminen ohjelmiston vaatimusmäärittelyyn esimerkiksi asiakasvaatimusten analysoimiseksi saattaa olla uutta. Erityisesti turvallisuusvaatimuksia sisältäville ohjelmistoille riskienhallinnan puuttuminen aiheuttaa ongelmia tapauksissa, joissa ohjelmiston turvallisuutta joudutaan osoittamaan.

Heikosti tunnistetut ja toteutetut vaatimukset aiheuttavat tuotekehitysprojektiin huomattavia lisäkustannuksia sekä aikataulutuksellisia viiveitä, kun riittämättömiä tai virheellisiä vaatimuksia joudutaan määrittelemään, analysoimaan, koodaamaan ja testaamaan uudestaan. Pahimmassa tapauksessa ohjelman suorituskykyyn, vakauteen tai luotettavuuteen liittyvät ongelmat havaitaan vasta valmiista ohjelmistosta sen ollessa jo markkinoilla tai koekäytössä asiakkaalla.

Tärkeää vaatimusmäärittelyssä on kuitenkin tunnistaa laajasta vaatimusjoukosta ne vaatimukset, jotka ovat ohjelmiston turvallisuuden kannalta kaikkein kriittisimpiä. Tällöin suunnittelun riskienhallintaa, testausta ja validointia voidaan kohdistaa eniten näihin vaatimuksiin, kun taas vähemmän kriittisten vaatimusten analysointiin, testaukseen, todentamiseen ja kelpuutukseen käytettävää työpanosta voidaan pienentää. Tämä edellyttää yrityskohtaista päätöksentekoprosessia suunnittelukatselmuksineen, joissa riskienhallintasuunnitelmassa määriteltyjä tavoitteita ja kriteerejä vasten tehdään jako kriittisiin tai ei-kriittisiin vaatimuksiin. Suunnittelukatselmuksissa arvioidaan myös vaatimusten oikeellisuus ja riittävyys.

Riskienhallintasuunnitelman laatiminen alkaa hyvin aikaisessa määrittelyn vaiheessa, ja suunnitelmaa voidaan päivittää vielä määrittelyvaiheenkin jälkeen. (ks. kuva 2).



Kuva 2. Riskienhallintasuunnitelman tulee kattaa myös määrittely.

Hyvin usein dokumentoitujen suunnitelmien laadinta riskienhallinta-, verifiointi- ja validointisuunnitelmien osalta on puutteellista. Suunnitteluprosessin tuottama dokumentaatio on avainasemassa silloin, kun ohjelmistolta edellytetään ns. hyväksyntäprosessia sen markkinoille saattamiseksi (ks. esimerkki kappaleessa 3.1.4). Arviointi kohdistetaankin ohjelmiston suunnitteludokumentteihin, koska valmiin ohjelmiston vaatimustenmukaisuutta ei voida kaikilta osin todeta pelkästään ohjelmistosta. Mikäli ohjelmiston suunnitteludokumentaatio ei tässä vaiheessa täytä hyväksyntäprosessin edellyttämiä vaatimuksia, on varmaa, että aikataulutukselliset viiveet ja myös kustannustekijät nousevat huomattaviksi. Pahimmassa tapauksessa tuotteen suunnittelu joudutaan uusimaan osittain tai kokonaan.

Ohjelmistojen luotettavuuteen vaikuttavat tekijät eivät ole pelkästään teknisiä, vaan ihmisten toimintatavoilla on siihen merkittävä vaikutus. Silloin kun suunnitellaan ns. moniteknologisia laitteita ja järjestelmiä, suunnittelutiimi tai yksittäinen suunnittelija joutuu tekemään hyvinkin monimutkaisia ja kauaskantoisia päätöksiä. Näiden päätösten oikeellisuus punnitaan vasta tuotteen ollessa markkinoilla.

Edellä mainittujen seikkojen perusteella vaatimusmäärittelyn ongelmat voidaan kiteyttää laadunhallintajärjestelmiin ja riskienhallintaan sekä tiedonkulkuun ja osapuolten näkemyseroihin liittyviin ongelmiin ja puutteisiin.

3. Vaatimusmäärittelyprosessin kehittäminen

Tässä luvussa tarkastellaan toimintatapoja, joiden avulla yrityksessä voidaan kehittää riskitietoista vaatimusmäärittelyprosessia. Aluksi käsitellään laadunhallintajärjestelmien, harmonisoitujen standardien ja riskienhallinnan tarjoamia mahdollisuuksia edistää prosessin jäljitettävyyttä ja ohjelmiston luotettavan toiminnan kannalta kriittisten vaatimusten tunnistamista. Tämän jälkeen seuraa vaatimusmäärittelyprosessiin liittyvän tiedonkulun tarkastelu, jossa perustellaan tiedonvälitystapojen merkitys monialaisen asiantuntemuksen yhdistämisessä. Lisäksi esitetään lähestymistapa niiden kehittämiseksi tavalla, joka tukee asiantuntemuksen yhdistämistä.

3.1 Vaatimusmäärittelyprosessin kehittäminen standardien ja riskienhallinnan avulla

Ohjelmistojen turvalliseen ja luotettavaan toimintaan voidaan vaikuttaa pääasiallisesti onnistuneella vaatimusmäärittelyllä. Hyvä vaatimusmäärittely edellyttää ohjeistetun vaatimusmäärittelyprosessin lisäksi toimivaa ohjelmistotuotantoprosessia, riskienhallintaa, laadunhallintajärjestelmää suunnittelukatselmuksineen sekä selkeitä ohjeita tuottaa kulloinkin tarvittava suunnitteludokumentaatio.

Laajojen ja monimutkaisten ohjelmistojen suunnittelu ja vaatimusmäärittely sisältää useita toisistaan riippuvia vaiheita ja toimintoja. Kattavilla sovellusalueen vaatimukset huomiovilla menetelmäohjeilla vaatimusmäärittelyyn voidaan liittää tarvittavat suunnittelun lähtötiedot, kuten turvallisuus-, suorituskyky- jäljitettävyys- ja lakisääteiset vaatimukset. Näin ollen valmis ohjelmisto täyttää sille asetetut vaatimukset ja on turvallinen käyttää sekä on valmistunut sovitussa ajassa eivätkä suunnitteluaikaiset kustannukset ole ylittyneet.

Suunnittelun eri vaiheissa tehdään paljon päätöksiä (hyväksy /hylkää, riski/etu, riittävän suorituskykyinen, riittävän pätevä, lähtötiedot ovat riittävät), joihin eivät valmiit standardit voi antaa vastauksia. Nämä päätökset ovat aina yrityskohtaisia, ja niiden kriteerit voi asettaa ainoastaan yritys itse. Näiden kriteerien toteutumista valvotaan erillisissä suunnittelukatselmuksissa.

Harmonisoitujen standardien ja riskienhallinnan edellyttämien menetelmäohjeiden soveltaminen yrityksen omaan suunnitteluprosessiin tuo useita etuja ja mahdollisuuksia. Yrityksen tulee kuitenkin muistaa, että mitkään standardit tai ohjeet eivät välttämättä sellaisenaan sovellu yrityksen suunnitteluprosessin käyttöön, vaan standardien vaatimuksien rinnalle on lisättävä omat yrityskohtaiset ja sovellusaluekohtaiset lisävaatimukset.

3.1.1 Laadunhallintajärjestelmät

Laadunhallintajärjestelmät ohjaavat organisaation toimintaa prosessimaiseen toimintamalliin, jossa organisaation eri toiminnot kuvataan yksittäisiksi prosesseiksi. Prosessimaisen toimintamallin ansiosta esimerkiksi ohjelmistojen suunnittelu ja valmistus saadaan ohjeistettua ja systematisoitua, jolloin suunnittelutulokset ovat tasalaatuisempia, ja prosessissa havaittujen virheiden ja ongelmien korjaaminen on helpompaa. Laadunhallintajärjestelmien avulla kyetään helpottamaan myös tuotteen laadun mittaamista keräämällä ja analysoimalla organisaation eri prosesseista saatua tietoa, kuten asiakasvalitukset, tuotantohäiriöraportit, suunnitteluprosessien häiriöt sekä logistiset ongelmat. Sisäisillä auditoinneilla voidaan myös parantaa suunnittelun laatua, joka näkyy pienellä viiveellä myös parantuneena ohjelmiston laatuna.

Tällainen systemaattinen toiminta mahdollistaa myös organisaation eri yksiköiden, prosessien ja henkilöiden välisen paremman tiedonkulun, jolla voidaan ehkäistä paremmin inhimillisistä tekijöistä tai ristiriitaisista päätöksistä johtuvat haitalliset vaikutukset suunnitteluun tai vaatimusmäärittelyyn.

Jokainen organisaatio on joskus havainnut, että asiat eivät mene niin kuin on suunniteltu. Tästä syystä järjestelmään on integroitava mukaan ns. ongelmienratkaisuprosessi [EN 60601-1-4, ISO 9001, ISO 13485], joka kykenee ohjaamaan ja korjaamaan ohjelmiston suunnittelun, valmistuksen tai käytön aikana havaittuja ongelmia mahdollisimman tehokkaasti ja organisaation johdon haluamalla tavalla. Ongelmienratkaisuprosessi on menettelytapa, jolla havaitut ongelmat analysoidaan, tiedotetaan, ratkaistaan ja siirretään käyttöön ja tuotetaan tarvittava dokumentaatio. Suunnitelmat tai menettelytavat tulisi määritellä jokaiselle elinkaaren vaiheelle ja prosessi sekä käytetyt menettelytavat tulee dokumentoida.

Laadunhallintajärjestelmien avulla [ISO 9001] organisaatio kykenee osoittamaan, että

a) sillä on kyky toimittaa johdonmukaisesti tuotetta, joka täyttää asiakasvaatimukset ja soveltuvat lakisääteiset vaatimukset

b) se pyrkii lisäämään asiakastyytyväisyyttä soveltamalla vaikuttavasti järjestelmää, joka sisältää järjestelmän jatkuvan parantamisen prosessit sekä asiakkaiden ja soveltuvien lakisääteisten vaatimusten täyttämisen varmistamisen.

Laadunhallintajärjestelmän toimivuutta valvotaan määräajoin sisäisin tai ulkoisin auditoinnein. Auditointien tulee kattaa myös käytettyjen alihankkijoiden toimintaprosessit.

3.1.2 Harmonisoidut standardit

Laajojen ja monimutkaisten ohjelmistojen suunnittelu ja vaatimusmäärittely sisältää useita toisistaan riippuvia vaiheita ja toimintoja. Ilman kattavia menetelmäohjeita vaatimusmäärittelyyn ei kyetä liittämään kaikkia tarvittavia suunnittelun lähtötietoja, kuten turvallisuus-, suorituskyky- jäljitettävyys- ja lakisääteisiä vaatimuksia. Näin ollen valmis ohjelmistokaan ei välttämättä täytä sille asetettuja vaatimuksia ja aiheuttaa suunnitteluaikataulujen ja -kustannusten ylittymisen. Tästä syytä suunnitteluprosessi tuleekin ohjeistaa ja jakaa tiettyihin vaiheisiin ja osatehtäviin, joissa jokaisessa vaiheessa on omat selkeät dokumentointivaatimukset.

Soveltamalla suunnitteluprosessiin ja ohjelmiston vaatimusmäärittelyyn harmonisoituja standardeja[1] voidaan varmistua siitä, että suunnitteluprosessin eri vaiheet määritellään ja kullekin vaiheelle on omat selkeät tavoitteet, joiden onnistumista voidaan arvioida erillisissä suunnittelukatselmuksissa.

Harmonisoitujen standardien etuna on lisäksi se, että ne asettavat tiettyjä dokumentointivaatimuksia itse suunnitteluprosessille, vaatimusmäärittelylle sekä itse tuotedokumentaatiolle, jolloin ohjelmiston vaatimustenmukaisuus tietyillä sovellusalueilla asettaville viranomais- ja suorituskykyvaatimuksille voidaan osoittaa huomattavasti helpommin.

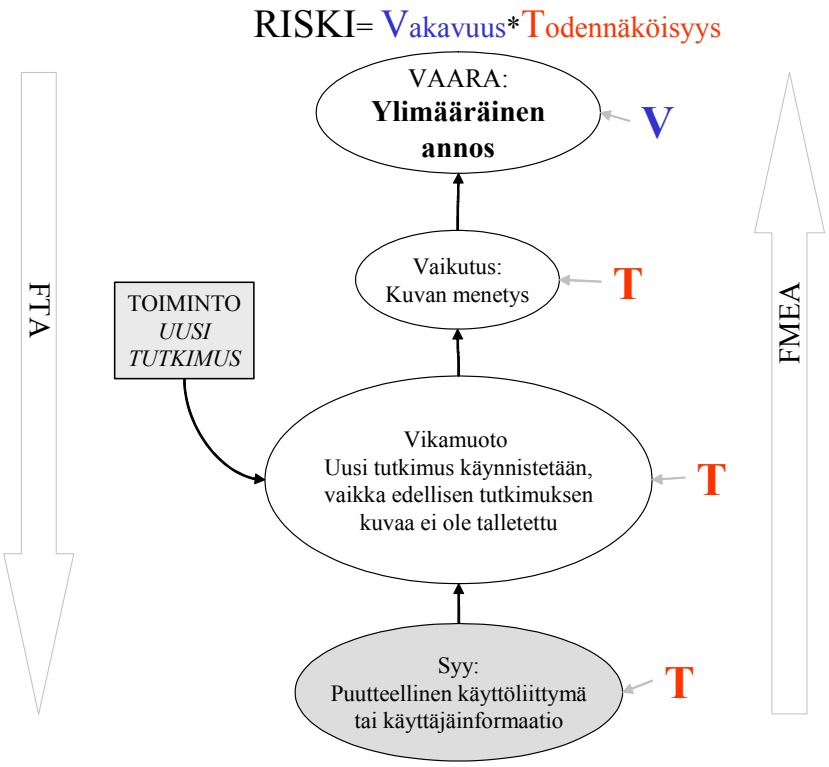
Valmistajalla on mahdollisuus käyttää myös omia suunnittelumetodeja. Näissä tapauksissa tuotteen turvallisuuden ja vaatimustenmukaisuuden osoittaminen lakisääteisten vaatimusten osalta saattaa olla hankalaa ja aikaa vievää, ja tämä on syytä huomioida ohjelmiston markkinoille saattamisessa.

3.1.3 Riskienhallinta ja riskianalyysi

Organisaatioiden toimintaan liittyy aina (talous, markkinointi, tukiprosessit, suunnittelu, valmistus ja henkilöstön toiminta) erilaisia riskejä ja vaaratekijöitä. Riskienhallinnan tavoite on ennalta ehkäistä, valvoa ja poistaa näiden toimintaa uhkaavien riskien toteutumista. Riskianalyysistandardit antavat puitteet riskienhallinnalle, mutta tehokas riskienhallinta edellyttää useita yrityskohtaisia ohjeita itse analyysin suorittamiseksi sekä yritysjohdon määrittelemiä periaatteita hallita yritystoiminnan riskejä, teknologiariskejä, tuoteriskejä ja projektinhallinnan riskejä. Analyysissä on huomioitava, että kaikki riskit eivät löydy välttämättä yhdellä menetelmällä, joten kattavan analyysin suorittamiseksi joudutaankin soveltamaan useampia toisiaan tukevia analyysimenetelmiä. FTA, FMEA ja

[1] Technical specification adopted by European Standards Organizations, developed under a mandate given by the European Commission and/or European Free Trade Association, in support of essential requirements of a New Approach Directives (http://www.cenorm.be/boss/glossary.asp#H).

FMECA soveltuvat hyvin teknisten ominaisuuksien analysointiin, kun taas HRA ja PHA soveltuvat paremmin inhimillisten tekijöiden ja virheellisen käyttäytymisen analysointiin. Liitteessä D on kuvattu riskienhallintaan ja analyysiin liittyvät vaiheet ja termit.



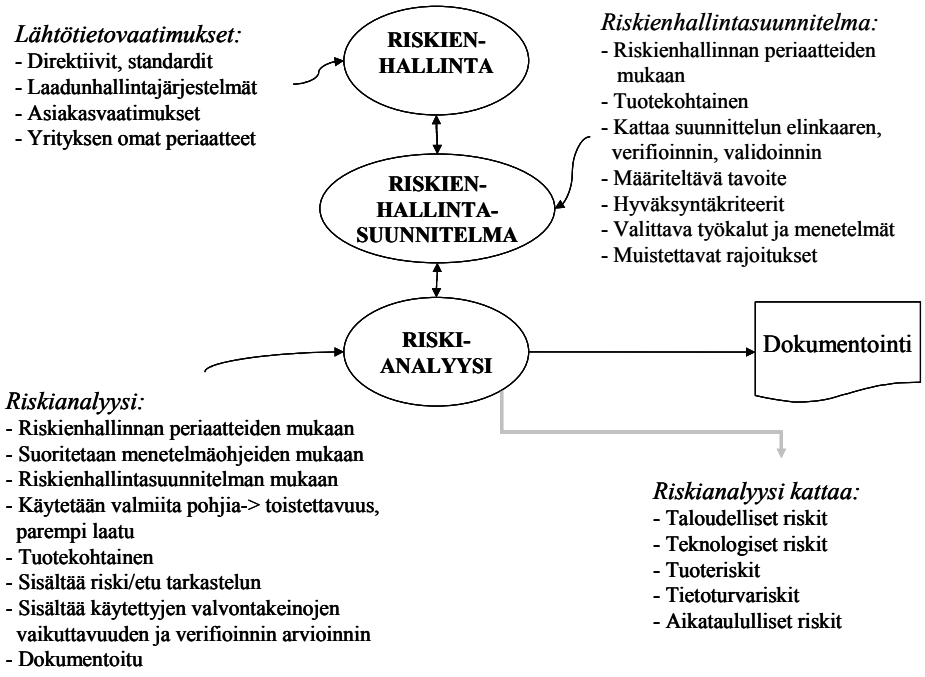
Kuva 3. Riskianalyysillä analysoidaan vaaran aiheuttavia tapahtumia.

Riskianalyysillä etsitään mahdollisia vaaran aiheuttavia tapahtumia tai syy-seurausketjuja (ks. kuva 3) tavoitteena määrittää ohjelmiston toiminnallinen turvallisuustaso.

Kun vaaran aiheuttava tapahtuma on tunnistettu, tehdään riskin suuruuden arviointi vaaran vakavuuden ja syyn, vikamuodon ja vaikutuksen todennäköisyyksien perusteella. Mikäli riski kasvaa kohtuuttoman suureksi, tehdään tarvittavat riskinpienennystoimenpiteet yrityksen hyväksymien riskinhallintaperiaatteiden mukaan sekä noudattamalla yleisesti hyväksyttyä ALARP-periaatetta (IEC 61508-5). Keinot riskin pienentämiseksi tulisi valita seuraavien periaatteiden mukaisessa järjestyksessä (MDD 1993, IEC 61508-5):

- Pyri poistamaan tai vähentämään riskiä mahdollisimman paljon luontaisesti turvallisella suunnittelulla tai rakenteella.
- Käytä riittäviä suojakeinoja niiden riskien yhteydessä, joita et voi poistaa, esim. hälytysjärjestelmät.
- Tiedota käyttäjää jäännösriskeistä, jotka johtuvat käytettyjen suojatoimenpiteiden vaikutuksesta.

Tuotekohtaista riskianalyysiä varten on aina tehtävä riskienhallintasuunnitelma, jossa määritellään tavoitteet ja rajoitukset itse analyysille. Riskienhallinta tapahtuu kuvan 4 mukaisessa järjestyksessä.



Kuva 4. Riskianalyysin työjärjestys.

Ohjelmiston tuotekehityksen elinkaaren eri vaiheissa voidaan käyttää erilaisia analyysimenetelmiä, esim. projektin alkuvaiheessa vaarojen tunnistamiseen soveltuu “top down”-tyyppinen menetelmä ja myöhemmässä vaiheessa taas analyyttisempi, tuotteen rakenteen huomioonottava ”bottom-up”-tyyppinen menetelmä. Käytettävät mentetelmät valitaan aina yrityskohtaisesti. Liitteessä E on kuvattu joitain riskianalyysimenetelmän valintaan vaikuttavia seikkoja.

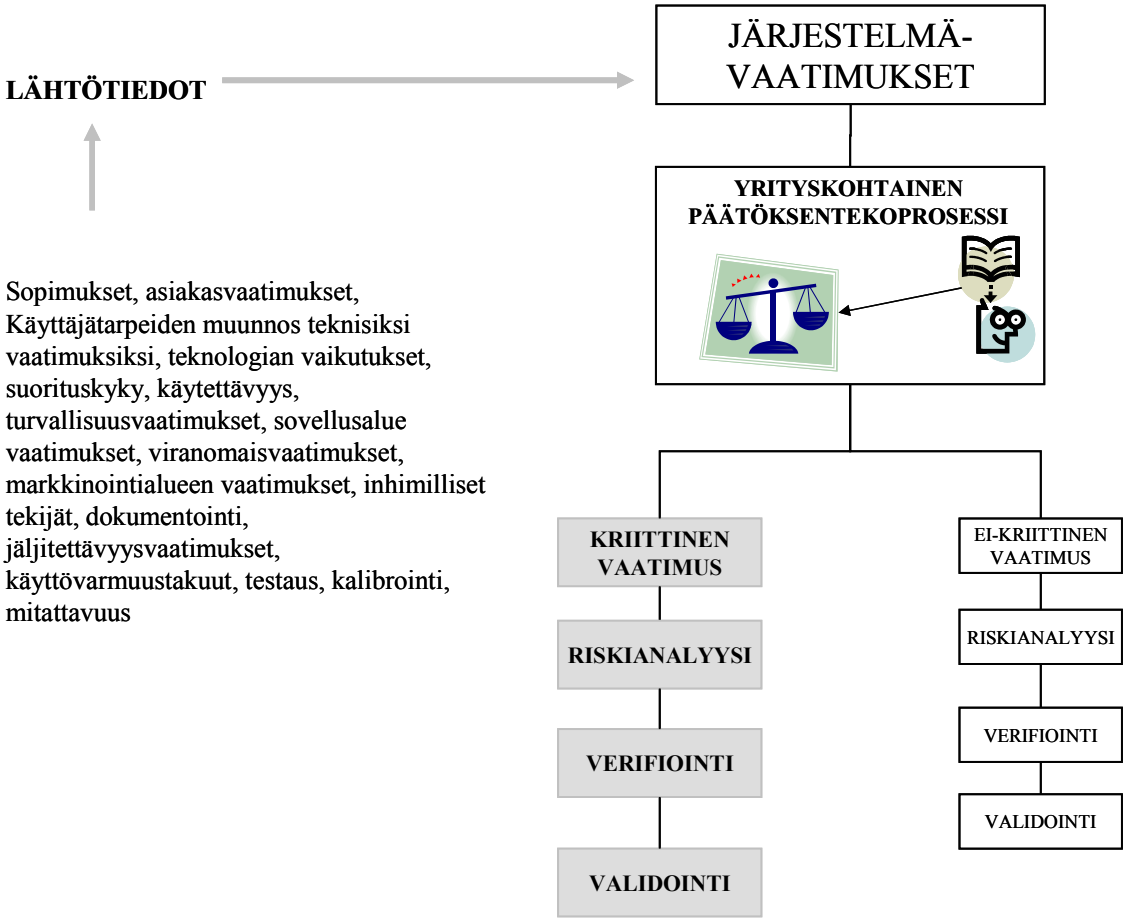
Ohjelmistoissa on lukematon määrä erilaisia vaatimuksia, jotka turvallisuuden kannalta voidaan luokitella kriittisiin tai vähemmän kriittisiin vaatimuksiin. Ohjelmiston vaatimusmäärittelyprosessiin tulee lisätä sellaisia toimintoja, joilla laajasta vaatimusjoukosta tunnistetaan ne kaikkein kriittisimmät vaatimukset. Kriittisten vaatimusten tunnistamiseen voidaan käyttää useampiakin toisistaan riippuvia tai riippumattomia keinoja, kuten riskienhallintaa, riskianalyysiä, menetelmäohjeiksi muutettua kokemusperäistä tietoa sekä sovellusalue- ja vaatimusanalyysejä.

Kriittisiksi vaatimuksiksi käsitetään ohjelmistosta ne vaatimukset, joiden virheellisen toiminnan seurauksena ohjelmiston suorituskyky heikkenee tai ohjelmisto siirtyy epästabiiliin tilaan tai suorittaa virheellisiä toimintoja ja aiheuttaa täten kohtuuttoman haitan käyttäjälle, sivullisille, ympäristölle tai terveydenhuollon tuotteissa potilaalle.

Kriittisten vaatimusten tunnistamiseen ei ole olemassa mitään takuuvarmaa keinoa. Usein pitkällisen kokemuksen perusteella tiedetään, että ohjelmiston tietyt osat aiheuttavat ongelmia käytössä ja näiden ominaisuuksien määrittelyyn ja testaukseen kiinnitetään erityistä huomiota. Tämä pätee niin kauan, kun ohjelmiston käytössä tai ympäristössä ei tapahdu muutoksia. Ongelmat alkavat silloin, kun ko. ohjelmistoa käytetään eri tavalla, sen käyttäjät vaihtuvat tai ohjelmiston käyttötarkoitusta muutetaan.

Tällainen subjektiivisen näkemykseen perustuva kriittisen vaatimuksen tunnistaminen ei voi olla kovin analyyttinen, tehokas ja toistettava, ja varmaa on ainakin se, että tällainen tapa ei kelpaa luotettavuuden ja turvallisuuden osoittamiseksi.

Kriittisen vaatimuksen syntymiseen vaikuttavat hyvin useat tekijät, jopa organisaation julistama laatu- tai tietoturvapolitiikka voi tehdä vaatimuksesta kriittisen. Esimerkiksi terveydenhuollon sovelluksessa kriittisiä vaatimuksia ovat ne vaatimukset, joiden avulla monitoroidaan potilaan vitaaliparametreja. Kriittsien vaatimuksen virhetilanteen seurauksena ohjelmisto antaa potilaan tilasta virheellistä informaatiota, jonka seurauksena voidaan tehdä virheellinen diagnoosi tai hoitopäätös.



Kuva 5. Yrityskohtainen päätöksentekoprosessi.

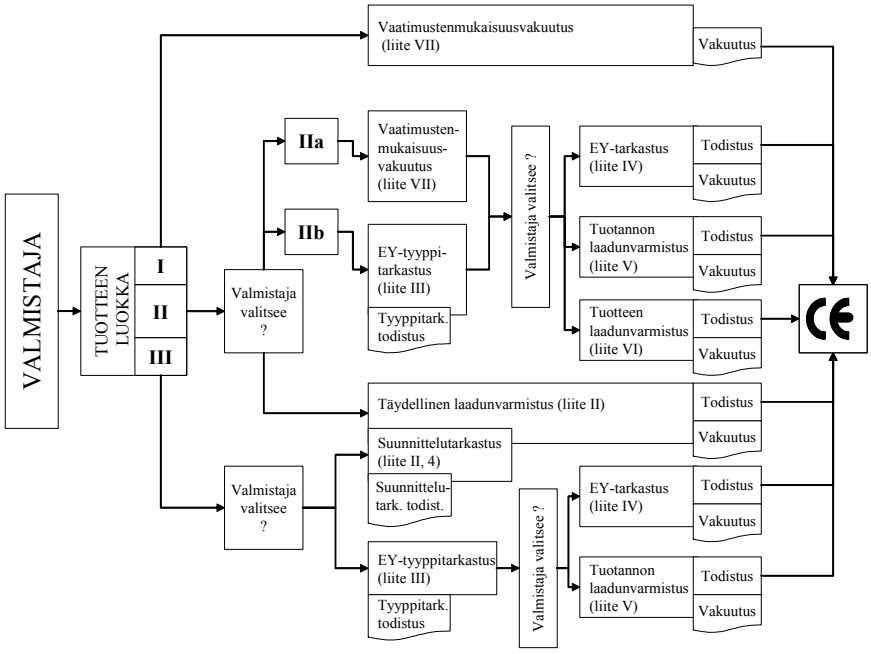
Menetelmät kriittisten ja ei-kriittisten vaatimusten tunnistamiseen poikkeavat eri aloilla (ks. kuva 5), koska ohjelmistoja suunnitellaan erilaisilla työkaluilla, eri tarkoituksiin, eri markkina-aluille ja niitä käyttävät erilaiset ihmiset. Tästä syystä jako kriittisiin ja ei-kriittisiin tehdään ohjelmistokohtaisesti ja se edellyttää yrityskohtaista päätöksentekoprosessia.

Riskianalyysit ja tulosten riski/etutarkastelu ovatkin eräs luotettavimmista keinoista tunnistaa sovelluksen kriittiset vaatimukset. Erityisesti tämä pätee silloin, kun ohjelmistosta tai sovellusalueesta ei ole olemassa kokemusperäistä tietoa.

3.1.4 Esimerkki standardien soveltamisesta terveydenhuollon tuotteeseen

Tässä luvussa tarkastellaan standardien soveltamista käyttäen esimerkkinä terveydenhuollon tuotetta, jolle asetetaan lakisääteisiä vaatimuksia. Esimerkkiä käsitellään ensin tuotteeseen kohdistuvien teknisten vaatimusten kannalta. Tämän jälkeen sitä tarkastellaan eri alojen ammattilaisten yhteistoiminnan näkökulmasta.

Ohjelmistoa sisältävän lääkinnällisen laitteen tai potilaan hoitoon tai tilan tarkkailuun käytettävän ohjelmiston markkinoille saattaminen tapahtuu EU-alueella direktiivin 93/42/EEC mukaisesti. Direktiivissä (MDD 1993) annetaan olennaiset vaatimukset tuotteen turvallisuudelle, suorituskyvylle ja käytettävyydelle sekä määritellään vaatimuksia kyseisten tuotteiden suunnittelu- ja valmistusprosesseja tuotteen vaatimuksenmukaisuuden varmistamiseksi. Siksi valmistajan on kyettävä osoittamaan, että ohjelmistotuotanto ja sen avulla tuotettu ohjelmisto täyttää direktiivin olennaiset vaatimukset.

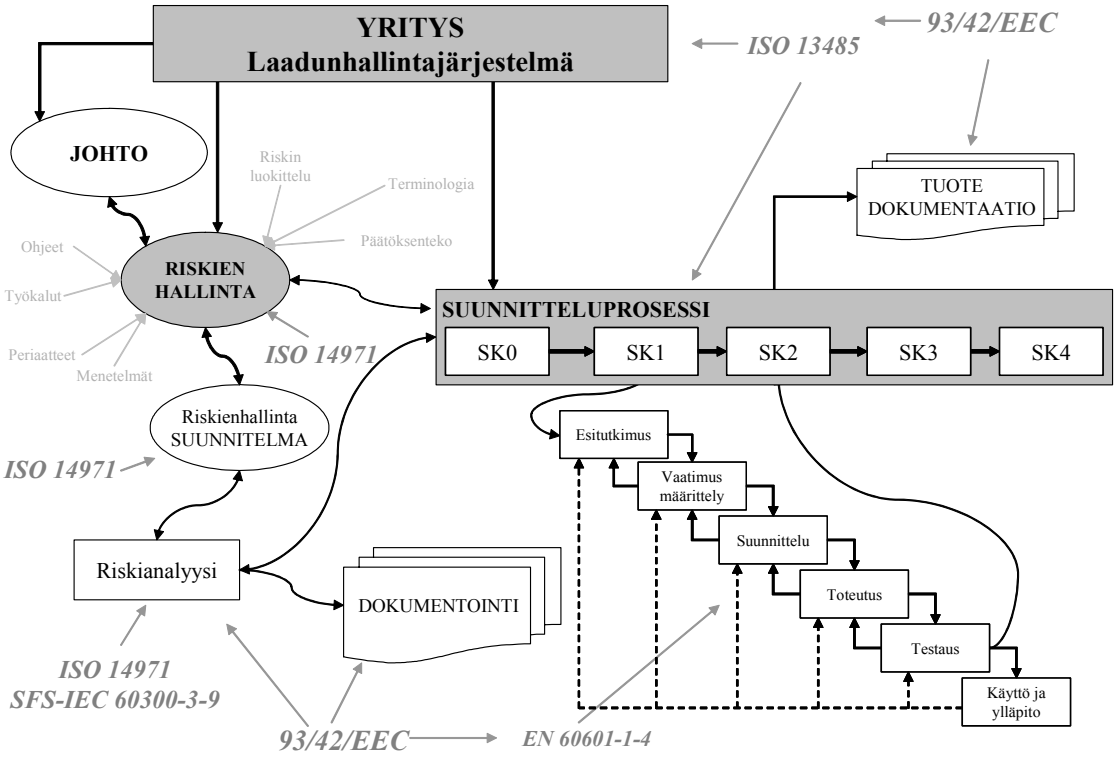


Kuva 6. Vaihtoehtoisia reittejä tuotteen markkinoille saattamiseksi.

Korkeamman riskiluokan omaavissa tuotteissa tämän vaatimuksenmukaisuuden arvioinnin suorittaa ilmoitettu laitos. Arvioinnin laajuus riippuu tuoteluokasta, johon tuote kuuluu. Valmistajalla on myös mahdollisuus valita kuvan 6 mukaisesti vaihtoehtoisia reittejä, joilla tuote voidaan saattaa markkinoille. Arviointi kohdistuu tuotteen suunnittelumenetelmiin, valmistukseen ja itse tuotteeseen.

Tuotteen tulee täyttää direktiivin (MDD 1993) liitteen 1 olennaiset vaatimukset. Lisäksi valmistajan on huolehdittava hyväksytyn laadunhallintajärjestelmän soveltamisesta tuotteiden suunnitteluun, valmistukseen ja lopputarkastukseen. Vaatimukset liittyvät yrityksen käytössä olevaan laadunhallintajärjestelmään, tuotteen suunnitteluun, valmistukseen, suorituskykyyn, rakenteeseen, materiaaleihin, merkintöihin sekä tuotteen suunnittelun, valmistuksen ja käytön aikaiseen riskienhallintaan, ja suunnittelu-, valmistus-, ja riskienhallintadokumentaatioon. Direktiivin artiklan 5 mukaan jäsenvaltioiden on pidettävä vaatimustenmukaisina tuotteita, jotka vastaavat yhdenmukaistettujen standardien vaatimuksia. Yhdenmukaistetut standardit löytyvät Euroopan yhteisöjen virallisesta lehdestä (http://europa.eu.int/comm/enterprise/newapproach/standardization/harmstds/reflist/meddevic.html).

Näin ollen terveydenhuollon tuotteen tai sen ohjelmiston suunnittelussa joudutaan soveltamaan standardeja, jotka kattavat laadunhallintajärjestelmän, riskienhallinnan, ohjelmistotuotannon sekä eri laiteryhmille tarkoitetut tuotespesifiset standardit (ks. kuva 7).



Kuva 7. Eri toiminnoille sovellettavia standardeja.

Ohjelmiston suunnittelun osalta tämä yhdenmukaistetun standardin käyttö tarkoittaa sitä, että kun valmistaja soveltaa ohjelmistotuotannolleen ja ohjelmistolle standardin EN 601-1-4:1999 vaatimuksia, täyttää ohjelmisto myös sille asetetut direktiivin olennaiset vaatimukset. Standardi edellyttää tiettyjen menettelytapojen noudattamista, koska pass/fail-testit eivät sovellu valmiin ohjelmiston testaamiseen. Standardin lähestymistapa on kertoa vaatimus, ja käyttäjän tehtävä on osoittaa, kuinka kyseinen vaatimus saavutetaan. Menettelytapa noudattaa yleisiä laadunohjauksen periaatteita (ISO 9000 -sarja).

Riskienhallinnan osalta joudutaan soveltamaan standardia ISO 14971:2000, joka määrittelee vaatimukset tuotteen suunnittelun, valmistuksen ja käytön aikaiselle riskienhallinnalle sekä vaatimukset itse riskienhallintasuunnitelman ja riskianalyysin suorittamiselle.

Laadunhallintajärjestelmän osalta joudutaan soveltamaan standardia EN ISO 13485:2003, joka määrittelee vaatimuksia yrityksen toimintaprosesseille ja näiden keskinäisille vuorovaikutuksille, dokumentointivaatimuksia laadunhallintajärjestelmälle sekä tuotteelle, dokumenttien valvonnalle, johdolle, laatupolitiikalle ja vastuujaolle, johdon katselmuksille, resurssien hallinnalle, kommunikoinnille, suunnittelulle, tuotekehitykselle, suunnittelukatselmuksille, suunnittelun lähtötiedoille (sisältäen tuotekohtaiset lakisääteiset vaatimukset), suunnistelun verifioinnille ja validoinnille, ostolle, tuotannolle, tuotteen asennukselle, tunnistettavuudelle ja jäljitettävyydelle, sisäisille auditoinneille, prosessien mittaukselle, control of nonconforming product, korjaaville ja ehkäiseville toiminnoille.

EU-alueella NB:n tekemät ohjelmistoarvioinnit suunnitellaan tapauskohtaisesti. Arviointitapoja on useita ja ne riippuvat sekä yksittäisen ohjelmiston valmiusasteesta ja kriittisyydestä että yrityksen ohjelmistotuotannosta ja -menetelmistä laadunvarmistuksineen (ks. kuva 6). Pääasiallisena tavoitteena kaikissa lähestymistavoissa on arvioida lääkintälaitteiden ohjelmistojen vaatimustenmukaisuus yleisesti hyväksyttyihin alan standardeihin nähden.

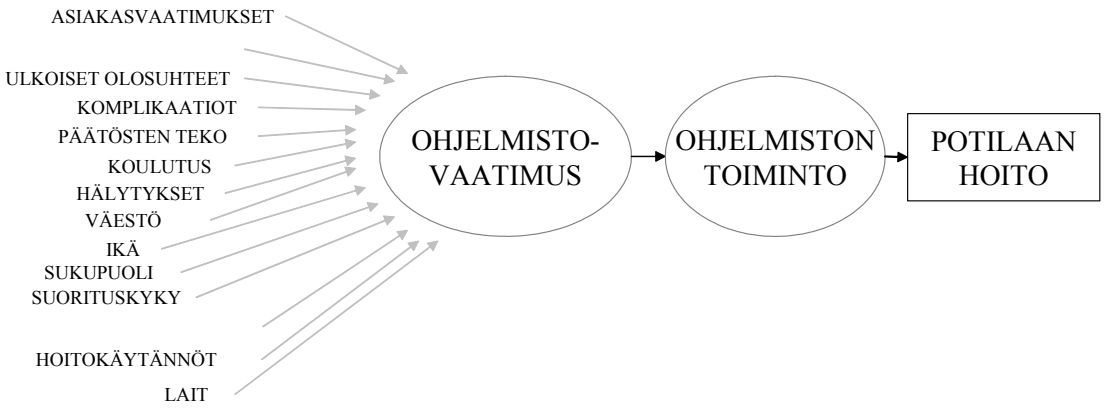
NB arvioi yrityksen laadunhallintajärjestelmän sekä tuotekehityksen menettelytavat ja tuotekohtaiset validointidokumentit ja myönteisessä tapauksessa antaa tästä todistuksen. Tämä todistus on osoitus siitä, että yritys on ohjelmistotuotannossaan käyttänyt riittävän tehokkaita riskienhallintamenetelmiä ja sillä on toimiva verifiointi- ja validointikäytäntö sekä ohjelmisto täyttää direktiivin (MDD 1993) olennaiset vaatimukset.

Esimerkki osoittaa, että laadunhallintajärjestelmien, harmonisoitujen standardien ja riskienhallinnan soveltaminen osana ohjelmiston vaatimusmäärittelyä on tärkeää. Alkuvaiheessa tiukat ohjeistetut toimintaprosessit dokumentointivaatimuksineen saattavat tuntua suunnittelua ja innovatiivisuutta rajoittavana tekijänä ja voivat aiheuttaa muutosvastarintaa. Mikäli ohjeet muuttuvat yrityskohtaisen koulutuksen ja opastuksen kautta

todellisiksi käytännöiksi ja tavaksi toimia, suunnittelijat hyväksyvät ne helpommin. Samalla myös vaatimusmäärittelyprosessi paranee, koska toiminnot, ratkaisut ja päätökset voidaan jäljittää määrittelystä eteen tai taaksepäin.

Standardien yrityskohtaisessa soveltamisessa tarvitaan eri alojen ammattilaisten asiantuntemusta. Osapuolten yhteistoiminnalla on tässä merkittävä vaikutus, koska monialainen asiantuntemus pitäisi saada yhdistymään tavalla, joka ei vaaranna ohjelmistotuotteen toiminnallista turvallisuutta.

Ohjelmistoa sisältävän lääkinnällisen laitteen tai potilaan hoitoon tai tilan tarkkailuun käytettävän ohjelmiston markkinoille saattaminen tapahtuu EU-alueella direktiivin 93/42/EEC mukaisesti. Merkittävimmät useita eri kohteita kattavan direktiivin (MDD 1993) vaatimuksista ovat: a) tuotteen tulee soveltua aiottuun käyttötarkoitukseensa, b) tuote on riittävän suorituskykyinen ja c) tuote on riittävän turvallinen. Jotta edellä mainitut kohdat voidaan täyttää, tulee laitteen sovellusalue ja käyttötarkoitus tuntea perinpohjaisesti.



Kuva 8. Sovellusalue vaikuttaa ohjelmistovaatimuksiin.

Ohjelmistolla toteutetun potilaan hoitoon, diagnoosiin tai tilan tarkkailuun tarkoitettu toiminto toteutetaan yhdellä tai useammalla ohjelmistovaatimuksella. Kuvan 8 mukaan kyseisen toiminnon mahdollistavan vaatimuksen tai vaatimusten laatimiseen vaikuttavat useat eri osatekijät. Taitavinkaan ohjelmistotuotannon ammattilainen ei kuitenkaan kykene tekemään toimivia ja luotettavia vaatimuksia, mikäli hänellä ei ole apunaan sovellusalueen asiantuntijaa.

Yrityksen tulisi tarvittaessa varmistaa riittävän laaja osaamispohja rekrytoimalla suunnittelutiimin tueksi lääkäreitä, hoitajia, käyttäytymistieteilijöitä ja käytettävyysasiantuntijoita.

3.2 Vaatimusmäärittelyprosessiin liittyvän tiedonkulun kehittäminen

Ohjelmistojen vaatimusmäärittelyprosessi perustuu siihen liittyvien osapuolten vuorovaikutukseen. Ohjelmistotuotetta koskevat vaatimukset muodostuvat ohjelmistojen kehittäjien, myyjien, hankkijoiden, käyttäjien jne. asiantuntemuksen yhdistymisen pohjalta Vaatimusmäärittelyä eri näkökulmista tarkastelevien asiantuntijoiden kokemuksen ja tietämyksen yhdistämisen tarkoituksena on mahdollisimman oikeiden ja turvallisten vaatimusten muodostaminen. Tavalla, jolla tieto välittyy osapuolten kesken, voi olla suuri merkitys lopputuloksen eli ohjelmiston laadun kannalta. Turvallisuusvaatimuksia sisältävien ohjelmistojen vaatimusmäärittelyssä tiedonvälitystavoilla on merkitystä myös tuotteen toiminnallisen turvallisuuden kannalta.

Myös standardien ja riskinhallinnan soveltaminen yrityksen olosuhteisiin edellyttää yhteistyötä. Ohjelmiston luotettavan toiminnan kannalta kriittisten vaatimusten tunnistaminen tapahtuu eri alojen asiantuntijoiden vuorovaikutuksen tuloksena.

Seuraavassa tarkastellaan ensin vaatimusmäärittelyprosessia ja sen yrityskohtaista kehittämistä tiedonvälityksen näkökulmasta. Tämän jälkeen käsitellään asiantuntijaosapuolten keskinäisen ymmärryksen merkitystä. Lopuksi esitellään keinoja asiantuntijoiden tiedonvälitystapojen riskitietoiseen kehittämiseen ohjelmistokehitysalan yrityksissä.

3.2.1 Vaatimusmäärittelyprosessi ja sen yrityskohtainen kehittäminen tiedonvälityksen näkökulmasta

Osa luvussa 2 esiintuoduista vaatimusmäärittelyn ongelmista näyttäisi johtuvan siitä, että tarvittava tieto ei aina välity määrittelyprosessiin liittyvien osapuolten kesken. Seurauksena siitä, että osapuolet tulkitsevat vaatimuksia omien näkökulmiensa valossa, kaikki olennainen tieto ei välttämättä tule esiin. Tämä voi johtaa siihen, että vaatimusmäärittely ei täytä turvallisuudelle asetettuja vaatimuksia, vaan luo mahdollisuuden suunnitteluvirheiden syntymiselle.

Vaatimusmäärittelyprosessin alussa vaatimukset ovat alustavia, joten niihin liittyy paljon epävarmuutta. Vaatimuksia voi tulla monelta taholta ja ne voivat olla moniselitteisiä ja ristiriitaisia (Haikala & Märijärvi 1998, Hull et al. 2002). Ohjelmiston kehittäjät eivät välttämättä tunne riittävästi sovellusaluetta ja tunnista asiakkaan tarpeita. Jotta vaatimukset olisivat helposti ymmärrettäviä, ne ilmaistaan yleensä luonnollisella kielellä. Haasteena on tällöin asiakkaan tarpeen tai ongelman muotoilu riittävän tyhjentävästi, mutta ilman ammattiterminologiaa (Hull et al. 2002). Asiakkaan vaatimukset käännetään ”insinöörikielelle”, mutta olennaisten asioiden välittyminen voi olla hankalaa, kos-

ka ei ole yhteistä käsitystä eikä yhtenäistä terminologiaa. Lisäksi tieto käyttäjätarpeista ei välttämättä kulje suoraan käyttäjältä suunnittelijalle, vaan jonkun muun osapuolen, kuten esim. myyjän, kautta. Käyttäjävaatimusten tunnistamisen ongelmiin kuuluu myös se, että ei ehkä kyetä arvioimaan vaatimusmäärittelyprosessin muutostarpeita sovellusalueen vaihtuessa.

Ohjelmiston kehittäjien ja myyjien välinen keskinäinen ymmärrys voi olla puutteellista näkökulmaerojen vuoksi. Kehittäjät tulkitsevat vaatimuksia ohjelmiston toteutettavuuden kannalta, mutta myyjät suuntautuvat talouden pohjalta. Osapuolet eivät välttämättä tunne riittävästi toistensa työn vaikuttimia, minkä seurauksena voi syntyä väärinkäsityksiä ja aukkoja tiedonkulussa. Vastaavasti suunnittelijoiden kesken voi olla erilaisista ajattelu- ja toimintatavoista johtuvia näkökulmaeroja. Osa suunnitteluun liittyvistä päättelyperusteista ja kokemusperäisestä tiedosta voi jäädä välittymättä toisille.

Myös yleisten standardien ja riskinhallintamenetelmien soveltaminen oman yrityksen vaatimusmäärittelyprosessiin asettaa haasteen asiantuntijatiedon välittymiselle. Standardien soveltamista hankaloittaa niiden yleisluontoisuus, minkä lisäksi ne voivat olla hyvin monimutkaisia ja laajoja. Toisaalta ongelmana voi olla standardien yhteensovittaminen tai se, että sopivaa standardia ei ole olemassa. Standardien soveltaminen vaatii yrityksen olosuhteita koskevaa tarkkaa kokemusperäistä tietoa ja eri alojen asiantuntijoiden asiantuntemuksen yhdistämistä.

Koska ohjelmistokehittämiseen vaikuttavat tekijät ovat yrityskohtaisia ja niiden erityispiirteiden tunnistaminen vaatii yrityskohtaista kokemusperäistä tietoa, ei ole mahdollista luoda yrityksille ulkoapäin tuotuja valmiita konsepteja, joilla voitaisiin räätälöidä standardivaatimukset yrityskohtaisiksi vaatimuksiksi. On tunnistettava itse, miksi ja miten tiettyä standardia pitäisi käyttää omassa yrityksessä. Yrityskohtaistamiseen tarvitaan yritysten omaa panosta ja eri alojen yhteistyötä. Raskaita ja laajoja toimintaprosseja ei myöskään voida ottaa käyttöön kerralla, vaan vähitellen, prosessimaisesti. Asiantuntijoiden on yhdessä muodostettava käsitys siitä, minkälainen vaatimusmäärittelyprosessin pitäisi kokonaisuutena olla. Tällöin lähdetään liikkeelle ohjelmiston turvallisuuden kannalta kriittisten vaatimusten tunnistamisesta. Riskianalyysin suorittamiseen osallistuu yleensä usean eri alan ammattilaisia, mutta heidän asiantuntemuksestaan huolimatta analyysi ei aina toteudu toivotulla tavalla. Sen lisäksi, että kriittisten vaatimusten tunnistamista vaikeuttaa standardien yleisluontoisuus, myös analyysiin osallistuvien erilaisista näkökulmista johtuvat tulkintaerot voivat hankaloittaa sitä. Asiantuntijoiden pitäisi pystyä muodostamaan yhteinen käsitys standardien valintaperiaatteista ja noudattamistavasta, mutta jos osapuolet eivät ymmärretä tarpeeksi toistensa näkökulmia, heidän asiantuntemuksensa ei yhdisty parhaalla mahdollisella tavalla.

3.2.2 Keskinäisen ymmärryksen merkitys asiantuntijoiden vuorovaikutuksessa

Yhteistyöhön liittyvät ongelmat eri alojen asiantuntijoiden vuorovaikutuksessa tulkitaan helposti yhteisen kielen puutteeksi. Yhteisen kielen puuttuminen on kuitenkin aina osoitus myös yhteisten ajattelumallien puuttumisesta ja näkemysten fragmentoitumisesta (Launis 1997). Ilman yhteistä tulkintakehystä, johon erilaiset käsitykset voisi suhteuttaa, näkökulmat eivät kohtaa riittävästi.

Kaarela (1996) on todennut laitossuunnittelun osalta, että suunnittelijoiden väliset keskustelut eivät välttämättä paranna suunnittelutiedon välittymistä, koska suunnittelun eri aloilla ei ole yhtenäistä käsitystä käsitteistä ja käytettävistä termeistä. Vaikeutta lisää se, että näissä keskusteluissa välitetty tieto kirjoitetaan hyvin harvoin muistiin muita varten. Informaation saanti on vaikeaa, koska suunnitteludokumentit eivät sisällä kaikkea aikaisempaan suunnitteluun liittyvää informaatiota. Tyypillinen esimerkki tällaisesta informaatiosta ovat puuttuvat suunnitteluperustelut.

Rakennesuunnittelua ja konetekniikkaa tutkineen Karsentyn (2000) mukaan suunnittelu on alue, joka erityisesti suosii rinnakkaisten näkökulmien olemassaoloa. Mikä tahansa suunnitteluratkaisu voidaan nähdä näkökulmien välisen neuvottelun tuloksena. Suunnittelua koskevassa yhteistyössä keskinäinen ymmärrys on usein ratkaisevan tärkeää tehtävien onnistumisen kannalta, mutta siihen ei ole kiinnitetty riittävästi huomiota. Karsentyn mukaan kysymys on siitä, että ei ole yhteistä käsitystä työn kohteena olevasta ongelmasta. Käsitysten eroavuutta ei aina huomata, minkä seurauksena voi syntyä ongelmia. Osa päätöksentekoprosessissa tapahtuvasta yhteistyöstä pitäisi kohdentaa yhteisen ongelmakäsityksen luomiseen, koska se auttaa ymmärtämään toisten näkökulmia ja helpottaa yhteistyön koordinointia.

Asioiden yhteisen käsittelyn tulisi arvojen, tavoitteiden ja yleisten määritelmien sijaan kohdistua mahdollisimman konkreettiseen kohteeseen (Launis 1997). Yhteisen ajattelumallin rakentaminen käsiteltävästä asiasta nostaa esiin näkökulmaeroista johtuvat ristiriitaisuudet. Niiden analysointi edellyttää erilaisten näkökulmien käsittelyä ja suhteuttamista toisiinsa.

Äänettömällä ammattitaidolla tarkoitetaan perinteisesti intuitioon ja kokemukseen perustuvan asiantuntijuuden näkökulmaa, mutta Launiksen (1997) mukaan monialaisen asiantuntijatyön yhteydessä tälle käsitteelle voitaisiin antaa toisenlainen sisältö. Yhteistoiminnan kannalta katsottuna äänetön asiantuntemus ei osallistu yhteisten ajattelumallien rakentamiseen.

Äänettömän tiedon ulkoistaminen näkyväksi on keskeistä luotaessa uutta tietoa organisaatioissa (Nonaka & Takeuchi 1995). Ulkoistaminen tapahtuu vuorovaikutuksen tuloksena, luomalla käsiteltävästä asiasta yhteinen malli ja muodostamalla sen pohjalta yhteisiä käsitteitä keskustelun avulla. Yhteinen käsitteellinen perusta, jonka avulla voidaan kehittää osapuolten ymmärrystä työn kohteesta ja työprosessista, auttaa vähentämään asiantuntijoiden kommunikointiongelmia (esim. Bechky 2003).

Borehamin (2002) mukaan työprosessia koskevalla tiedolla ei tarkoiteta osaamista, jolla yleensä viitataan kokemusperäiseen käytännön tietoon. Työprosessitieto sisältää enemmän, koska sen yhtenä osatekijänä on teoreettinen ymmärtäminen. Se syntyy kokemusperäisen ja teoreettisen tiedon yhdistymisen tuloksena. Työprosessitiedolla tarkoitetaan työssä tarvittavaa tietoa, joka voi muodostua vain itse työprosessissa ja joka ei ole aina ilmaistua tai edes ilmaistavissa (Norros & Nuutinen 2002). Työprosessia koskevat kokemukset eivät muunnu työprosessia koskevaksi tiedoksi automaattisesti (Rasmussen 2002). Niitä pitäisi reflektoida eli tarkastella ikään kuin ulkopuolelta, ne pitäisi ottaa keskustelun kohteeksi ja tulkita uudestaan yhteisesti muodostettujen käsitteellisten mallien avulla. Ei-formaalit keskustelut ovat tällöin tärkeitä.

Levesonin (2000) mukaan ei-formaaleilla tekniikoilla tulee aina olemaan suuri, ellei jopa suurin osuus monimutkaissa ohjelmistokehityshankkeissa, koska matemaattisilla malleilla ei pystytä käsittelemään kaikkia järjestelmän kehittämiseen liittyviä seikkoja.

Käsitellessään ohjelmistosuunnittelijoiden subjektiivisten arviointien perustana olevia henkilökohtaisia malleja Littlewood, Neil ja Ostrolenk (1995) korostavat näiden mallien ymmärtämisen tärkeyttä. Heidän mukaansa pitäisi saada esiin niiden taustalla olevat oletukset ja näkökulmat ja tehdä näkyväksi informaatio, jota ei ole otettu tarkastelun kohteeksi arviointia tehtäessä. Tarvitaan yhteisiä malleja, jotka perustuvat kollektiiviseen käsitykseen ja muodostuvat yhteiseksi tiedoksi. Yhteisiä malleja kehitettäessä voidaan tunnistaa henkilökohtaiset näkökulmat ja keskustella niistä. Henkilökohtaisten mallien välittämisessä muille tarvitaan jonkin verran ei-formaalia ilmaisua, jotta mallien edustamaa asiantuntemusta voidaan hyödyntää. Erityisesti suunnittelumenetelmissä mallin ilmaisevuus on vähintään yhtä tärkeä asia kuin sen formaalinen muoto. Kirjoittajien mukaan ehkä kaikkein eniten ohjelmistosuunnitteluyhteisössä tarvitaan järjestelmien kehittämistä, arviointia ja käyttöä koskevan kokemuksen jakamista ja uudelleenkäyttöä.

Yrityksissä kannattaisi kehittää tiedonvälitystapoja, jotka auttavat osapuolia ymmärtämään paremmin toisiaan ja helpottavat siten eri alojen asiantuntemuksen yhdistämistä. Koska on kysymys ohjelmistoista, joiden toimintaan kohdistuu turvallisuusvaatimuksia, näiden vaatimusten pitäisi ohjata vaatimusmäärittelyyn osallistuvien asiantuntijoiden työtä käytännön tasolla.

3.2.3 Keskinäisen ymmärryksen lisääminen asiantuntijayhteistyössä

Edellytyksenä keskinäisen ymmärryksen lisääntymiselle vaatimusmäärittelyssä on siihen liittyvien tiedonvälitystapojen merkityksen ymmärtäminen ohjelmistotuotteen turvallisuuden syntymisen kannalta. Yhtä keskeinen vaatimus on se, että monialaista asiantuntijatyötä teettävien yritysten pitäisi tukea tällaista ymmärtämistä luomalla sille edellytykset (Hukki & Pulkkinen 2003a, b, Hukki & Pulkkinen, valmisteilla). Yrityksissä kannattaisi pyrkiä selvittämään, mitä vaatimusmäärittelyyn osallistuvien asiantuntijoiden on ymmärrettävä ja mistä heidän on saatava tietoa, jotta he pystyvät välittämään toisilleen tietoa tavalla, joka lisää ohjelmistotuotteen turvallisuutta. Turvallisuustietoinen tiedon välittyminen tarkoittaa tässä sitä, että kaikki ohjelmiston toiminnallisen turvallisuuden syntymisen kannalta tarpeellinen tieto välittyy ja että tiedon välittyminen on mahdollisimman läpinäkyvää. Asiantuntijoiden kokemusperäisen tiedon esiin saamiseksi ja eri tahojen välisen tiedonkulun parantamiseksi tarvitaan yhteinen systeeminen ajattelumalli, jonka avulla vaatimusmäärittelyprosessiin osallistuvat tahot voivat muodostaa mahdollisimman yhtenäisen käsityksen vaatimusmäärittelyprosessista. Systeemiajattelu tarkoittaa kokonaisuuksien, kokonaisuuden sisältämien osien keskinäisten suhteiden ja niiden välisen dynamiikan hahmottamista. Sengen (1990) mukaan systeemiajattelu on tärkein osatekijä oppivan organisaation kehittymisen kannalta.

Vaatimusmäärittelyä koskevassa kirjassaan Hull, Jackson ja Dick (2002) korostavat kehitettävänä olevan järjestelmän mallintamisen tärkeyttä vaatimusmäärittelyprosessin onnistumisen kannalta. Järjestelmän mallintamisella tarkoitetaan sen käytön, toiminnan ja suorituskyvyn mallintamista. Monissa yrityksissä, etenkin sidosryhmätasolla, on lisäksi käytössä käsite avainvaatimukset (key requirements). Näillä vaatimuksilla tarkoitetaan sellaista pientä vaatimusjoukkoa, joka on pelkistetty vaatimusten kokonaisjoukosta ja jossa on kiteytettynä järjestelmän ”olemus”.

Vaatimusten hallinta ja järjestelmän mallintaminen tukevat toisiaan. Viimeksi mainittu helpottaa esim. kommunikointia asiakkaan kanssa ja lisää järjestelmää koskevaa keskinäistä ymmärrystä. Järjestelmän mallintaminen ja analysointi tehdään sidosryhmä- ja järjestelmätasoilla tapahtuvana asiantuntijayhteistyönä. Mallintamisen luonne muuttuu sovelluksesta toiseen. Mallit siis muuttuvat, mutta vaatimusten hallintaa koskevat periaatteet pysyvät yleisinä sovelluksista riippumatta.

Järjestelmän mallintamisen lisäksi kirjoittajat pitävät järjestelmän kehittämisprosessin mallintamista hyvin tärkeänä. Järjestelmän kehittäminen voidaan kuvata yleisenä prosessina, jota voidaan käyttää moniin tarkoituksiin. Kirjassa esitetään vaihtoehtoisia tapoja prosessin kuvaamiseen.

Asiantuntijoiden kokemusperäisen tiedon esiin saamisen kannalta on kuitenkin olennaista, että järjestelmän kehittämisprosessin mallintamiseen liitetään myös tiedonkulun systeeminen tarkastelu. Tässä voidaan soveltaa menetelmää, jota kutsutaan tiedonkulun systeemiseksi analyysiksi. Sen avulla on mahdollista tunnistaa tiedon välittymisen puutteet vaatimusmäärittelyssä ja määritellä kriteerit riskitietoisille tiedonvälitystavoille. Kehitteillä oleva menetelmä perustuu lähestymistapaan, joka on luotu VTT:llä aiemman tutkimuksen yhteydessä muussa turvallisuuskriittisessä ympäristössä (Hukki & Pulkkinen 2003a, b). Tutkimuksen kohteena oli eri aloja edustavien asiantuntijoiden tiedonkulun kehittäminen ydinjätteen loppusijoitusratkaisua koskevassa tutkimustyössä. Lähestymistapa ja menetelmä on kuvattu tarkemmin muualla (Hukki & Pulkkinen, valmisteilla).

Menetelmä tarjoaa periaatteet, joiden avulla yrityksessä voidaan muodostaa yhteinen käsitys tiedon välittymisestä kohteena olevassa työprosessissa tai sen osassa, esim. yksittäisessä työketjussa. Käsityksen muodostaminen tapahtuu systeemisen kuvauksen avulla. Analyysi toteutetaan ryhmätyönä, yhteisten keskustelujen pohjalta. Tarkoituksena on, että siihen osallistuvat kaikki ne asiantuntijat, joiden työ liittyy työprosessiin, tai henkilöt, jotka on valittu edustamaan eri alojen asiantuntemusta.

Yhteisenä ajattelumallina toimiva systeeminen kuvaus auttaa havainnollistamaan tiedonvälitystapojen merkityksen ohjelmistotuotteen turvallisuuden syntymisessä ja saamaan esille tiedonvälitystapaa koskevat vaatimukset. Vaatimusten tunnistaminen ja määrittely konkrettisiksi tiedontarpeiksi auttaa saamaan esiin asiantuntijoiden kokemusperäistä tietoa. Yhteisen mallin muodostaminen luo edellytykset keskinäisen ymmärryksen lisääntymiselle ja sen myötä yhtenäisemmän terminologian kehittämiselle yrityksessä.

Seuraavassa on lyhyt kuvaus analyysistä sovellettuna ohjelmistotuotannon vaatimusmäärittelyprosessiin.

Analyysin ensimmäisessä vaiheessa tunnistetaan vaatimusmäärittelyprosessin tärkeimmät tavoitteet ohjelmiston avainvaatimusten muodostamisen näkökulmasta ja kuvataan prosessin eri vaiheiden yhteys näihin tavoitteisiin kaavion avulla. Tuloksena muodostetaan yhteinen käsitys avainvaatimusten muodostumisprosessista.

Toisessa vaiheessa tehdään kaaviokuvaus vaatimusmäärittelyprosessista työnjaon näkökulmasta. Kaavion avulla muodostetaan yhteinen käsitys vaatimusmäärittelyprosessiin osallistuvien asiantuntijoiden tehtävien keskinäisistä riippuvuuksista. Tuloksena tunnistetaan kunkin osapuolen työn merkitys osana avainvaatimusten muodostumisprosessia. Tehtävien merkityksen ymmärtäminen on ensimmäinen perusedellytys tiedonvälitystapojen merkityksen ymmärtämiselle.

Kolmannessa vaiheessa muodostetaan yhteinen käsitys asiantuntijoiden tehtävärajapintoihin liittyvistä tiedollisista riippuvuuksista. Tiedontarpeiden merkityksen ymmärtäminen avainvaatimusten muodostumisessa on toinen perusedellytys tiedonvälitystapojen merkityksen ymmärtämiselle. Tuloksena tunnistetaan tiedon välittymisen merkitys vaatimusmäärittelyprosessissa yleisellä tasolla.

Neljännessä vaiheessa muodostetaan rajapintakohtaisesti yhteinen käsitys tärkeimmistä keskinäisistä tiedontarpeista tunnistamalla rajapintoihin liittyvät tiedon välittymisen puutteet. Tarkastelun kohteena on toisaalta tietyn asiantuntijatehtävän suorittamisen kannalta tarpeellinen muilta saatava tieto ja toisaalta muille asiantuntijoille välitettävä tieto, jota he tarvitsevat häneltä oman työnsä suorittamiseen. Tiedon välittymisen puutteet koskevat todennäköisesti tietoa, joka ei ole substanssitietoa vaan tietoa, joka auttaa osapuolia ymmärtämään paremmin toistensa näkökulmia. Tämäntyyppistä tietoa, jota kutsutaan tässä oheistiedoksi, on tarkasteltu yksityiskohtaisemmin muussa yhteydessä (Hukki & Pulkkinen 2003a, b, Hukki & Pulkkinen, valmisteilla). Oheistiedossa on kysymys välitettävän tiedon perusteluista tai sen tuottamiseen liittyvistä taustatekijöistä, jotka jäävät helposti välittymättä, mutta joilla voi olla suurikin merkitys toisen osapuolen työn suorittamisen kannalta. Analyysissä muodostetaan käsitys siitä, mihin tehtävärajapintoihin liittyy tiedon läpinäkyvyyden lisäämistarpeita. Tarkastelemalla puutteellisen tiedon yhteydessä eri osapuolten asiantuntemusaloille ominaisten ajattelutapojen ja käytäntöjen eroja voidaan tunnistaa oheistietoa koskevat puutteet. Ottamalla huomioon nämä puutteet voidaan määritellä tehtävärajapintoja koskevat keskeiset tiedontarpeet. Samalla voidaan yhtenäistää käytössä olevaa terminologiaa.

Analyysin tuloksena pystytään muodostamaan käsitys tiedon välittymistä koskevista vaatimuksista ja määrittelemään yrityskohtaiset kriteerit turvallisuustietoisille tiedonvälitystavoille. Ne auttavat päivittämään sovelluskohtaisia vaatimustietokantoja tavalla, joka lisää ohjelmiston turvallisuutta. Kriteerit voidaan sisällyttää ohjelmistojen vaatimustenhallintaan ja osaksi koulutusta ja yrityksen toimintaprosessien kuvausta.

Tiedonkulun systeemisen analyysin soveltaminen ohjelmistokehitystä koskevaan vaatimusmäärittelyyn edellyttää kohdeyritysten kanssa tehtäviä yrityskohtaisia kehittämishankkeita. Analyysi on tarkoitus tehdä kertaluontoisesti ja sen tuloksia voidaan hyödyntää yrityksen myyjä-asiakasrajapinnan vaihtuessa.

On myös mahdollista kehittää jatkotutkimuksen pohjalta yleiset kriteerit, joiden avulla yritykset voivat itsenäisesti tunnistaa tiedonkulkuun liittyviä puutteita ja kehittää keskinäistä ymmärrystä lisääviä ja terminologiaa yhtenäistäviä tiedonvälitystapoja.

Vaatimusmäärittelyyn osallistuvien asiantuntijoiden keskinäistä ymmärrystä voidaan edistää myös lisäämällä ohjelmiston elinkaarimalliin tiedon välittymistä tukevia vaihei-

ta. Osana esisuunnittelua vaatimusmäärittelyä voisi edeltää tapaaminen, jossa asiantuntijat tarkastelisivat projektia yhdessä ja sopisivat yhteisistä käsitteistä. Vastaavasti prosessin viimeisenä vaiheena voisi olla tapaaminen, jossa tehtäisiin yhdessä projektin toteutumisen kriittinen arviointi.

Edellä kuvatun tiedonkulun systeemisen analyysin tulokset tukevat myös ohjelmiston kriittisten vaatimusten tunnistamista vaatimusmäärittelyä koskevassa riskinhallinnassa. Tämän lisäksi riskinhallintaa voidaan helpottaa soveltamalla muussa turvallisuuskriittisessä ympäristössä luotua menetelmää, joka on kehitetty tukemaan monialaista asiantuntijayhteistyötä. Menetelmä perustuu ydinvoimalaitosten palotilanteiden hallintaa koskevaan tutkimukseen (Hukki & Holmberg 2004). Sen tarjoamien periaatteiden avulla on luotu yhteinen ajattelumalli valvomotoiminnan tukemiseksi turvallisuutta uhkaavassa tilanteessa. Malli helpottaa erilaisten näkökulmien hyödyntämistä ja monialaisen asiantuntijatiedon yhdistämistä. Sen avulla voidaan määritellä tilanteen arvioinnin kannalta kriittiset arviointitehtävät ja tunnistaa tilannekohtaisesti riskitietoiset toimintatavat.

Menetelmä on sovellettavissa asiantuntijayhteistyöhön, jota tarvitaan ohjelmiston luotettavan toiminnan kannalta kriittisten vaatimusten tunnistamisessa. Sen avulla voidaan luoda yhteinen ajattelumalli, joka auttaa muodostamaan käsityksen riskitietoisesta tavasta priorisoida ohjelmistotuotteiden vaatimuksia riskinhallintaprojekteissa. Tämän menetelmän soveltaminen ohjelmistokehitykseen edellyttää jatkotutkimusta.

4. Yhteenveto

Raportissa on kuvattu turvallisuuskriittisten ohjelmistojen vaatimusmäärittelyprosessiin liittyviä ongelmia ja esitetty keinoja ohjelmistojen toiminnallista turvallisuutta lisäävien toimintatapojen kehittämiseksi yrityksissä. Tarkastelun kohteena on vaatimusmäärittelyprosessi, jota on käsitelty toisaalta teknisestä ja toisaalta asiantuntijoiden vuorovaikutuksen näkökulmasta.

Valmiin ohjelmiston vaatimustenmukaisuutta ilman tuotekehitysprosessin aikana syntynyttä tuotedokumentaatiota on käytännössä mahdotonta arvioida ohjelmiston koon, laajuuden tai monimutkaisuuden takia. Standardien ja riskienhallinnan yrityskohtainen soveltaminen tuo konkreettista apua ohjelmiston suorituskyvyn, luotettavuuden ja turvallisuuden osoittamiseen. Raportissa esitettyjä näkökohtia voidaan käyttää apuna, kun kehitetään yrityskohtaista riskitietoista vaatimusmäärittelyprosessia. Seuraavat seikat ovat tärkeitä kehittämisen kannalta:

a) Dokumentoinnin merkitys

   Ohjelmiston turvallisuuteen voidaan vaikuttaa eniten ohjelmiston vaatimusmäärittelyprosessissa. Jos ohjelmistolta edellytetään lakisääteisten vaatimusten täyttämistä, joudutaan arviointi kohdistamaan ohjelmistoa suunnittelevan ohjelmistotuotantoprosessin ja vaatimusmäärittelyprosessin tuottamien dokumenttien arviointiin. Arviointi suoritetaan prosessikuvausten, menetelmäohjeiden ja erilaisten riskianalyysi-, testi-, verifiointi- ja validointiraporttien pohjalta, joten suunnittelun dokumentaatio ja erityisesti vaatimusmäärittely on avainasemassa vaatimustenmukaisuuden osoittamisessa. Tällöin laadunhallintajärjestelmän ja harmonisoitujen standardien soveltaminen osana vaatimusmäärittelyä antaa hyvät puitteet laatia sellainen suunnitteludokumentaatio, joka täyttää yrityksen, asiakkaan ja ulkoisen arvioijan vaatimukset.

b) Ohjelmistovaatimusten jäljitettävyys

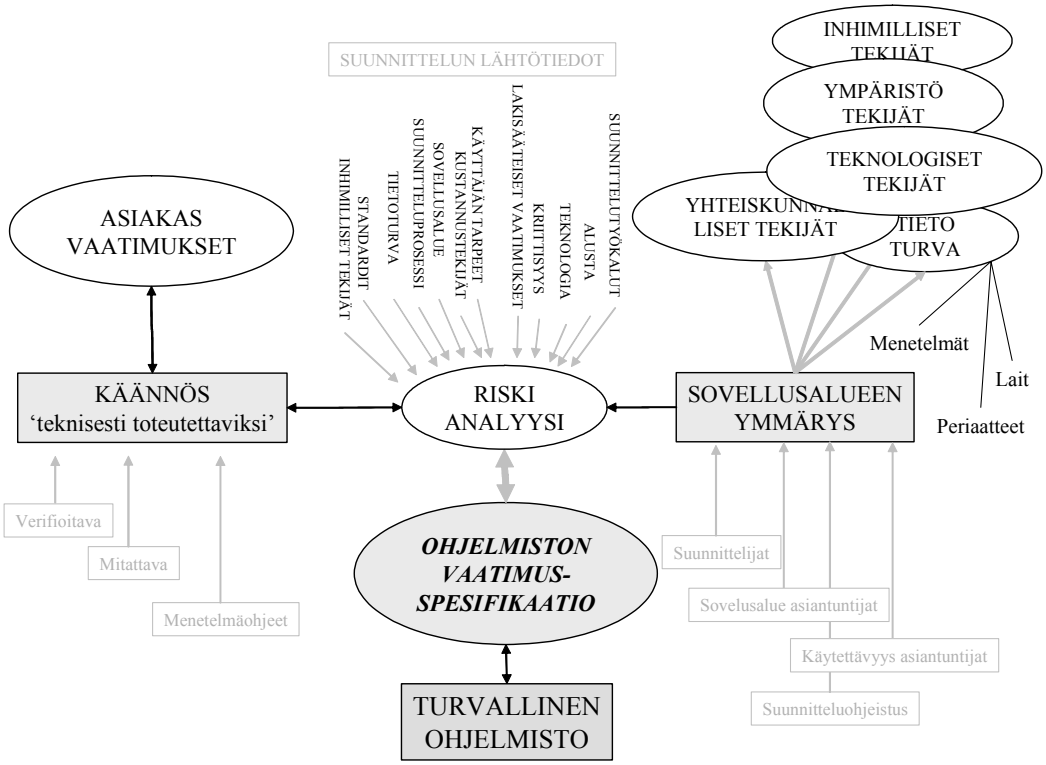
   Ohjelmistovaatimusten jäljitettävyydellä helpotetaan mm. ohjelmiston ylläpitoa ja turvallisuuden osoitusta, ja jäljitettävyys tuo muutostilanteissa kustannustehokkuutta. Jäljitettävyyteen tarvitaan kuitenkin monenlaisia ratkaisuja riippuen ohjelmistoprojektien suuruudesta, käytössä olevista suunnittelutyökaluista, tuotteiden kriittisyydestä ja toimialasta.

   Ohjelmistovaatimusten jäljitettävyys voidaan toteuttaa ainoastaan dokumentoiduista vaatimuksista. Tämä edellyttää ohjelmistotuotannon elinkaarimallin käyttöönottoa, jossa ohjelmistosuunnittelun eri vaiheille asetetaan tarkat tavoitteet sisältäen vaatimukset myös kulloinkin suunniteltavan ohjelmiston suunnitteludokumentaatiolle.

Osatehtävien tavoitteiden ja vaatimusten tulee olla niin hyvin ohjeistettuja, että niiden avulla voidaan yksittäinen vaatimus jäljittää mistä tahansa suunnittelun vaiheesta eteen- tai taaksepäin.

c) Suunnittelun aikainen riskienhallinta

Määrittelyvaiheen riskianalyysi edellyttää laajaa kokemusperäistä tietoa omasta ohjelmistosta ja sen käyttöympäristöstä sekä sitä tuottavista prosesseista sekä ohjeita ja valmiita tarkastuslistoja ohjelmiston vikamuodoista (ks. kuva 9). Analyysiä on harjoiteltava useamman kerran ja havaitut ongelmat on muutettava kirjallisiksi ohjeiksi analyysiä tukeviin menetelmäkuvauksiin.



Kuva 9. Osaamiseen ja riskienhallintaan perustuva yrityskohtainen vaatimusmäärittely.

d) Tiedonkulun kehittäminen

Turvallisuuskriittisten ohjelmistojen kehittämisessä tapahtuvaa asiantuntijayhteistyötä vaikeuttavat tiedonkulun puutteet ja osapuolten näkemyserot. Keskeisenä tekijänä ongelmien taustalla on riittämätön keskinäinen ymmärrys, jonka syynä on yhteisten ajattelumallien puuttuminen. Ilman yhteistä tulkintakehystä, johon erilaiset käsitykset voisi suhteuttaa, näkökulmat eivät kohtaa riittävästi. Yrityksissä kannattaisi kehittää tiedonvälitystapoja, jotka auttavat osapuolia ymmärtämään paremmin toisiaan ja helpottavat siten eri alojen asiantuntemuksen yhdistämistä. Koska on ky-

symys ohjelmistoista, joiden toimintaan kohdistuu turvallisuusvaatimuksia, näiden vaatimusten pitäisi ohjata vaatimusmäärittelyyn osallistuvien asiantuntijoiden työtä käytännön tasolla.

Edellytyksenä keskinäisen ymmärryksen lisääntymiselle vaatimusmäärittelyssä on se, että ymmärretään tiedonvälitystapojen merkitys ohjelmistotuotteen turvallisuuden syntymisen kannalta. Yhtä keskeinen vaatimus on, että yrityksen pitäisi tukea ymmärtämistä luomalla sille edellytykset. Yrityksissä kannattaisi pyrkiä selvittämään, mitä vaatimusmäärittelyyn osallistuvien asiantuntijoiden on ymmärrettävä ja mistä heidän on saatava tietoa, jotta he pystyvät välittämään toisilleen tietoa tavalla, joka lisää ohjelmistotuotteen turvallisuutta. Tiedonvälitystapojen kehittämiseksi tarvitaan yhteisten ajattelumallien luomista.

Kehitettävän järjestelmän ja sen kehittämisprosessin mallintaminen luo hyvän pohjan asiantuntijayhteistyölle, mutta asiantuntijoiden kokemusperäisen tiedon esiin saamisen kannalta on olennaista, että kehittämisprosessia tarkastellaan myös sitä koskevan tiedonkulun kannalta. Soveltamalla tiedonkulun systeemiseksi analyysiksi kutsuttua menetelmää on mahdollista tunnistaa tiedon välittymisen puutteet vaatimusmäärittelyssä ja määritellä kriteerit riskitietoisille tiedonvälitystavoille.

Menetelmän avulla voidaan luoda yhteinen ajattelumalli, jonka systeeminen kuvaustapa tarjoaa havainnollisen keinon kuvata vaatimusmäärittelyprosessin rakenne ja tiedon välittymisen dynamiikka. Tehtäväriippuvuuksien tarkastelu ja tehtävärajapintoihin liittyvien tiedollisten riippuvuuksien tunnistaminen suhteessa ohjelmiston avainvaatimuksiin auttaa ymmärtämään tiedonvälitystapojen merkityksen ja tunnistamaan osapuolten näkemyseroista johtuvat tiedonkulun puutteet ja kehittämistarpeet yrityksessä. Ohjelmistojen turvallisen toiminnan kannalta on ratkaisevan tärkeää, perustuuko asennoituminen yrityksissä ohjelmistojen vaatimuksenmukaisuuden täyttämiseen vai niiden turvallisuuden rakentumisen ymmärtämiseen.

Yhteisten ajattelumallien kehittäminen luo edellytykset turvallisuustietoisen tiedonkulun kehittämiselle. Systeeminen tarkastelutapa edistää keskinäistä ymmärrystä ja luo perustan yhtenäisemmän terminologian luomiselle ja tiedon paremmalle välittymiselle. Käsitysten yhtenäistyminen helpottaa eri alojen asiantuntemuksen yhdistämistä suunnitteluprojekteissa.

Tiedonkulun systeemisen analyysin soveltaminen ohjelmistokehitystä koskevaan vaatimusmäärittelyyn edellyttää kohdeyritysten kanssa tehtäviä yrityskohtaisia kehittämishankkeita. Analyysi on tarkoitus tehdä kertaluontoisesti ja sen tuloksia voidaan hyödyntää yrityksen myyjä-asiakasrajapinnan vaihtuessa. Tiedonkulun systeemistä analyysimenetelmää voidaan hyödyntää vaatimusmäärittelyprosessia kos-

kevien tiedonvälitystapojen kehittämiseen myös ei-turvallisuuskriittisissä ohjelmistonkehitysympäristöissä.

On myös mahdollista kehittää jatkotutkimuksen pohjalta yleiset kriteerit, joiden avulla yritykset voivat itsenäisesti tunnistaa tiedonkulkuun liittyviä puutteita ja kehittää keskinäistä ymmärrystä lisääviä ja terminologiaa yhtenäistäviä tiedonvälitystapoja.

Yrityksen vaatimusmäärittelyprosessia voidaan kehittää myös parantamalla asiantuntijoiden kokemusperäisen tiedon välittymistä siten, että lisätään ohjelmiston elinkaarimalliin tiedon välittymistä tukevat vaiheet. Esisuunnitteluvaiheessa tarvitaan keskustelua, jossa sovitaan yhteisistä käsitteistä, ja lopuksi on tärkeää tehdä yhdessä projektin toteutumisen kriittinen arviointi.

Vaatimusmäärittelyprosessia koskeva tiedonkulun systeeminen analyysi hyödyttää myös ohjelmiston kriittisten vaatimusten tunnistamista vaatimusmäärittelyä koskevassa riskinhallinnassa. Tämän lisäksi riskinhallintaa voidaan helpottaa soveltamalla menetelmää, joka on kehitetty tukemaan monialaista asiantuntijayhteistyötä. Menetelmän avulla voidaan sovelluskohtaisesti muodostaa käsitys riskitietoisesta toimintatavasta priorisoitaessa ohjelmistotuotteiden vaatimuksia riskinhallintaprojekteissa. Tämän menetelmän soveltaminen ohjelmistokehitykseen edellyttää jatkotutkimusta.

Yrityksen vaatimusmäärittelyprosessin kehittäminen raportissa kuvatuilla keinoilla tarjoaa seuraavanlaisia etuja:

- Systemaattisten suunnitteluprossien hyödyntäminen vaatimusmäärittelyssä mahdollistaa määrittelyprosessin toistettavuuden ja sen, että ohjelmiston vaatimustenmukaisuus voidaan osoittaa yrityksen omalle laadunvarmistukselle tai ulkopuoliselle arvioijalle. Riskienhallinnan avulla tapahtuva ohjelmiston kriittisten vaatimusten tunnistaminen ja määrittely lisää ohjelmiston turvallisuutta.

- Eri asiantuntijatahojen keskinäisen ymmärryksen lisääminen auttaa kehittämään virheellisiä suunnitteluratkaisuja vähentäviä ja kustannustehokkuutta lisääviä tiedonvälitystapoja suunnitteluprojekteissa. Tämän lisäksi se helpottaa monialaisen asiantuntemuksen yhdistämistä suunnitteluprosessia koskevassa riskienhallinnassa.

Lopuksi voidaan todeta, että muutokset viranomaisvaatimuksissa ja teknologiassa voivat aiheuttaa muutospaineita yrityskohtaiseen vaatimusmäärittelyprosessiin. Yritysten tuleekin tietyin määräajoin arvioida suunnitteluprosessin vastaavuutta vaatimuksiin nähden.

Lähdeluettelo

Bechky, B. 2003. Sharing Meaning Across Occupational Communities: The Transformation of Understanding on a Production Floor. Organization Science, Vol. 14, No. 3, s. 312–330.

Boreham, N. 2002. Work process knowledge in technological and organizational development. Teoksessa: Boreham, N., Samurcay, R. & Fischer, M. (toim.). Work Process Knowledge. London: Routledge.

EN 60601-1-4. EN 60601-1-4:1996. Medical electrical equipment - Part 1–4: General requirements for safety - Collateral standard: Programmable electrical medical systems, Amendment A1:1999 to EN 60601-1-4:1996.

Haikala, I. & Märijärvi, J. 1998. Ohjelmistotuotanto. Suomen ATK-kustannus Oy.

Hukki, K. & Holmberg, J. 2004. Development of management of Nuclear Power Plant Fire Situations. Proceedings of the International Conference on Probabilistic Safety Assessment and Management PSAM7 / ESREL ’04, Berlin, June 14–18.

Hukki, K. & Pulkkinen, U. (valmisteilla) Safety-Informed Expert Interaction in Nuclear Waste Management.

Hukki, K. & Pulkkinen, U. 2003a. Cognitive prerequisites for safety-informed organizational culture. Proceedings of the European Safety and Reliability Conference ESREL 2003, Maastricht, June 15–18.

Hukki, K. & Pulkkinen, U. 2003b. Enhancing Transparency in Multidisciplinary Expert Communication. Proceedings of the 3rd Symposium on Values in Decisions on Risk VALDOR 2003, Stockholm, June 9–13.

Hull, E., Jackson, K. & Dick, J. 2002. Requirements Engineering. London: Springer-Verlag.

IEC 61508-5:1998. Examples of methods for the determination of safety integrity levels.

ISO 13485. ISO 13485:2003. Medical devices - Quality management systems - Requirements regulatory purposes.

ISO 9001. SFS-EN ISO 9001:2001 Laadunhallintajärjestelmät, vaatimukset. Quality management systems. Requirements.

Kaarela, K. 1996. Enhancing communication of plant design knowledge. Espoo: VTT. 110 s. + liitt. 81 s. (VTT Publications 272).

Karsenty, L. 2000. Cooperative work: The role of explanation in creating a shared problem representation. Le Travail Humain, Vol. 63, No. 4, s. 289–309.

Kececi, N., Smidts, C., Modarres, M. & Hu, Y.-S. 1999. System-Software Interfaces for Safety-Related Digital I&C Systems [http://www.enre.umd.edu/ctrs-lab/esrel-paper.PDF]

Launis, K. 1997. Moniammatillisuus ja rajojen ylitykset asiantuntijatyössä. Teoksessa: Kirjonen, J., Remes, P. & Eteläpelto, A. (toim.). Muuttuva asiantuntijuus. Jyväskylä: Jyväskylän yliopisto.

Leveson, N. 2000. Intent Specifications: An Approach to Building Human-Centered Specifications. IEEE Software. Vol. 26, No. 1, s. 15–35.

Littlewood, B., Neil, M. & Ostrolenk, G. 1995. The role of models in managing the uncertainty of software-intensive systems. Reliability Engineering and System Safety 46, s. 87–95.

MDD 1993, Council Directive 93/42/EEC of 14 June 1993 concerning medical devices.

Nonaka, I. & Takeuchi, H. 1995. The Knowledge Creating Company. How Japanese Companies Create ther Dynamics of Innovation. New York: Oxfor University Press.

Norros, L. & Nuutinen, M. 2002. The concept of the core task and the analysis of working practices. Teoksessa: Boreham, N., Samurcay, R. & Fischer, M. (toim.). Work Process Knowledge. London: Routledge.

Rasmussen, L. 2002. Work process knowledge and creativity in industrial design. Teoksessa: Boreham, N., Samurcay, R. & Fischer, M. (toim.). Work Process Knowledge. London: Routledge.

Senge, P. 1990. The Fifth Discipline: The Art and Practice of the Learning Organization. New York: Currency Doubleday.

Liite A: Standardin IEEE-830 mukainen pohja ohjelmiston vaatimusspesifikaatiolle

Alla on standardin IEEE 830 mukainen esimerkki ohjelmiston vaatimusmäärittelyn laatimiseksi. Vaatimusmäärittelyn sisältö riippuu esitystavasta, jonka mukaan vaatimusmäärittely laaditaan.

A.1 Template of SRS Section 3 organized by mode: Version 1

3. Specific requirements

3.1 External interface requirements

3.1.1 User interfaces

3.1.2 Hardware interfaces

3.1.3 Software interfaces

3.1.4 Communications interfaces

3.2 Functional requirements

3.2.1 Mode 1

3.2.1.1 Functional requirement 1.1

.

.

3.2.1.n Functional requirement 1.n

3.2.2 Mode 2

.

3.2.m Mode m

3.2.m.1 Functional requirement m.1

.

.

3.2.m.n Functional requirement m.n

3.3 Performance requirements

3.4 Design constraints

3.5 Software system attributes

3.6 Other requirements

Liite B: 10 periaatetta ohjelmiston vaatimus-spesifikaation laadintaan

Ohjelmiston vaatimusmäärittelyn suunnittelussa on huomioitava vaatimuksiin vaikuttavat tekijät. Taulukossa B1 on eritelty 10 perusperiaatetta, jotka saattavat antaa uusia näkemyksiä ohjelmiston vaatimusmäärittelylle:

Taulukko B 1. 10 turvallisuusperiaatetta vaatimusspesifikaation laadintaan.

Periaate	Selitys / perustelu
1. Tunne sovellusalue ja käyttöympäristö [Identify the application area and environment]	Eri sovellusalueet poikkeavat toisistaan huomattavasti. Jollain sovellusalueella käytettävyys on ensisijainen, jollain toisella alueella taas suorituskyky on ensisijainen vaatimus. Eri sovellusalueilla työskentelee eri koulutusalan ja kulttuurin ihmisiä, laitteilta edellytetään erilaisia toimintoja, käyttäjän ja laitteiston väliset rajapinnat ovat erilaisia vaihdellen yksinkertaisesta erittäin monimutkaiseen rajapintaan tai käyttöliittymään, jossa järjestelmän tarjoamaa monimutkaista informaatiota on paljon. Vaatimusmäärittelyssä on tunnistettava sovellusalueen asettamat vaatimukset.
2. Määrittele lähtötiedot [Define the design input requirements]	Järjestelmän käyttötarkoituksesta riippuu, miten eri osapuolien vaatimukset ja toiveet vaikuttavat vaatimusmäärittelyyn. Mikäli näitä vaatimuksia ei tunnisteta määrittelyvaiheessa, seurauksena on todennäköisesti puutteellisesti toimiva ohjelmisto. On oletettavaa, että myös viiveet hyväksyntäprosesseissa tai markkinoille saattamisessa voivat kasvaa huomattavasti puutteellisten lähtötietojen takia. Lähtötiedot vaikuttavat myös riskienhallinnan ja suunnitteludokumentaation kattavuuteen ja laajuuteen.
3. Myönnä rajoitukset [Recognize limitations of application]	Yhdellä ohjelmistolla ei voi tehdä maailman kaikkia asioita. Myönnä tämä ja suunnittele järjestelmä siten. Osa rajoituksista osoitetaan riskianalyyseillä, osa mukana järjestelmän mukana seuraavalla käyttöohjeistuksella. Osaa järjestelmän rajoituksista voidaan määritellä ainoastaan opastuksella ja riittävällä koulutuksella. − rajoita, kiellä, kuvaile, analysoi, dokumentoi, opasta, kouluta Mikäli rajoituksia ei tuoda esiin kouluttamalla tai estämällä niitä teknisin ratkaisuin, tämä mahdollistaa ohjelmiston tahattoman tai tahallisen väärinkäytön.
4. Vaatimusmäärittelyn tulee olla ohjeistettu [Requirements specification process needs a code of practice to follow]	Systemaattinen, toistettava vaatimusmäärittely edellyttää tuekseen riittävän kattavat ohjeet sekä ohjelmistotuotantoprosessin, jossa toiminnot on vaiheistettu, ja eri vaiheita tukevat tarvittavat tukiprosessit ja katselmuskäytännöt.
5. Erotettava vaatimukset ja toiveet [Differentiate between requirements and desires]	Parantaa suunnittelun kustannustehokkuutta, kun päästään tekemään juuri sitä mitä pitikin.

6. Vaatimusten jäljitettävyys [Traceability of requirements]	Ainoastaan dokumentoidut vaatimukset voidaan jäljittää. Määrittele vaatimukset ja tavoitteet ohjelmistotuotannon dokumentaatiolle ja jäljitettävyydelle.
7. Vaatimusten mitattavuus [Measurability requirements]	Verifiointi ja validointi edellyttää mitattavia, verifioitavia ja validoitavia vaatimuksia. Muista tämä vaatimusmäärittelyssä (ristiriidattomuus, suorituskyky, toiminnallisuus). Esimerkiksi vaatimusta ”10 % nopeampi ja parempi kuin markkinoilla olevat vastaavat tuotteet” on erittäin vaikea verifioida.
8. Kriittisten vaatimusten tunnistus [Identification of critical requirements]	Järjestelmässä on usein useita erilaisia vaatimuksia. Osa näistä vaatimuksista on järjestelmän luotettavan, turvallisen ja suorituskykyisen toiminnan kannalta vähemmän merkityksellisiä kuin toiset. Suunnittelijan kannalta asian tekee vaikeaksi se, että monimutkaisessa järjestelmässä voi olla useita tuhansia vaatimuksia, jotka lisäksi jakaantuvat mekaniikan, sähkön, elektroniikan ja ohjelmiston osalle. Lisäksi vaatimuksen kriittisyys vaihtelee sovellusaluekohtaisesti. Esimerkiksi jonkin algoritmin puutteellinen suorituskyky voi olla tekstinkäsittelysovelluksessa ainoastaan kiusallista, kun taas samaisen algoritmin puutteellinen suorituskyky esimerkiksi ydinvoimalan reaktorin ohjausosassa on todennäköisesti katastrofaalista. Onko suunnittelijalla sitten keinoja tunnistaa kriittisiä vaatimuksia? Varmin tapa tunnistaa kriittisiä vaatimuksia on riittävän kattava riskianalyysi. Mikäli suunnittelijalla on pitkäaikainen kokemus ko. sovellusalueen järjestelmäsuunnittelusta, voidaan apuna käyttää mahdollisesti ns. vaatimuspaletteja, joiden avulla kokemusperäiseen tietoon perustuen on tehty jakoa kriittiseen ja ei-kriittiseen toimintaan liittyvien vaatimusten tunnistamisesta.
9.Suunnitelmallisuus [Need to proceed systematically]	Vaatimusmäärittely ja sen tukiprosessit edellyttävät suunnitelmallisuutta. Muista suunnitelmien teko ja päivitä suunnitelmia tarvittaessa: − Riskienhallintasuunnitelma − Verifiointisuunnitelma − Validointisuunnitelma − Testaussuunnitelma
10. Ohjelmistot ovat monimutkaisia [Recognize that software modules are complicated]	Ohjelmistot ovat monimutkaisia. Tämän takia niiden suunnittelua tai vaatimusmäärittelyä ei voida tehdä pelkästään ohjelmoijien toimesta. Suunnittelutiimissä on oltava useiden eri alojen osaajia (käytettävyys, riskienhallinta, ohjelmistoteknologiat, sovellusalueosaajia, psykologeja, matemaatikkoja, algoritmikehittäjiä jne.). Ainoastaan riittävän laajalla osaamispohjalla voidaan varmistaa ohjelmiston suorituskyky ja soveltuvuus käyttötarkoitukseensa.
10+. Älä uraudu. [Do not get into a rut]	Monet vaatimusmäärittelyn ristiriidoista saavat alkunsa suunnittelijoiden asenteesta. ”Ei siellä eilenkään ollut ongelmia ja kait sitä käytetään niin kuin ennenkin”. Suunnittele ohjelmia niin, kuin sinulla olisi aina uusi mielenkiintoinen koodaus edessäsi. Älä oleta mitään, testaa, tee vikasietoisia ratkaisuja.

Liite C: Lyhyt esittely standardista IEEE-830

Ohjelmiston vaatimusmäärittelylle tarkoitettu standardi IEEE 830 määrittelee suosituksia ohjelmiston vaatimusmäärittelyn [SRS] kirjoittamiselle. Ohje avustaa asiakasta kuvaamaan tarkasti, mitä he haluavat saada ja toisaalta valmistajaa ymmärtämään, mitä asiakkaat haluavat.

SRS:n kirjoittaminen edellyttää systemaattista menettelytapaa, jossa määritellään ohjelmiston vaatimusmäärittelylle formaatti sekä sisältö, ja itse ohjelmiston vaatimusmäärittelyn kirjoittamista tuetaan erilaisilla tarkastuslistoilla tai tyylioppailla.

Hyvä vaatimusmäärittely tuo useita etuja [IEEE 830], esimerkiksi

- Tarjoaa perustan laatia sopimus asiakkaan ja toimittajan välillä siitä, mitä ohjelmiston on tehtävä.
- Pienentää kehityskustannuksia ja panostusta.
- Tuottaa lähtökohdan arvioida kustannuksia ja aikatauluja.
- Tuottaa vetailukohdan verifioinnille ja validoinnille.
- Tarjoaa perustan laajennettavuudelle.

SRS on tarkoitettu tietylle ohjelmistotuotteelle, ohjelmalle, tai joukolle ohjelmia jotka suorittavat toimintoja tietyssä ympäristössä. Spesifikaatiota voi kirjoittaa joko ohjelmatoimittaja tai sitä voidaan kirjoittaa yhteistyössä toimittajan ja tilaajan kanssa.

SRS:ään ei saa sijoittaa suunnittelu- tai projektivaatimuksia, mutta sen tulee osoittaa seuraavat seikat:

a) Toiminnallisuus, mitä ohjelmiston on tarkoitettu tekevän?

b) Ulkoiset rajapinnat, kuinka ohjelmisto kommunikoi ihmisten, järjestelmän laitteiden, muiden laitteiden ja muiden ohjelmistojen kanssa?

c) Suorituskyky, mikä on nopeus, saatavuus, vasteaika, eri ohjelmatoimintojen toipumisaika jne.?

d) Attribuutit, mitä ovat siirrettävyys, virheettömyys, ylläpidettävyys, tietoturva, näkökohdat jne.?

e) Toteutuksen aiheuttamat suunnittelurajoitukset, edellytetäänkö joitain standardeja jotka vaikuttavat toteutuskieleen, tietokantojen eheyden periaatteisiin, resurssirajoituksiin, toimintaympäristöön jne.?

Hyvän ohjelmiston vaatimusspesifikaation tulisi olla oikea, yksiselitteinen, täydellinen, johdonmukainen, rankattu kriittisiin ja ei-kriittisiin vaatimuksiin, verifioitava, modifioitava ja jäljitettävä.

Jotta SRS täyttäisi yllä mainitut ominaisuudet, joudutaan ohjelmiston vaatimusmäärittelyprosessi liittämään kiinteäksi osaksi ohjelmistotuotantoprosessia ja asettamaan tavoitteet määrittelylle. Näiden tavoitteiden toteutumista voidaan valvoa suunnittelukatselmuksissa. Lopputuloksena on parempi vaatimusmäärittely ja tätä kautta luotettavampi ja suorituskykyisempi ohjelmisto, jonka ylläpidettävyys ja uudelleenkäytettävyys helpottuvat.

Standardi IEEE 830 määrittelee erittäin kattavasti ohjelmiston vaatimusmäärittelyyn vaikuttavia tekijöitä, kuten ympäristöä, suorituskykyä, tietoturvaa, luotettavuutta, käyttörajoituksia, työkaluja ja rajapintoja sekä tarjoaa valmiin mallin laatia ohjelmiston vaatimusspesifikaatio.

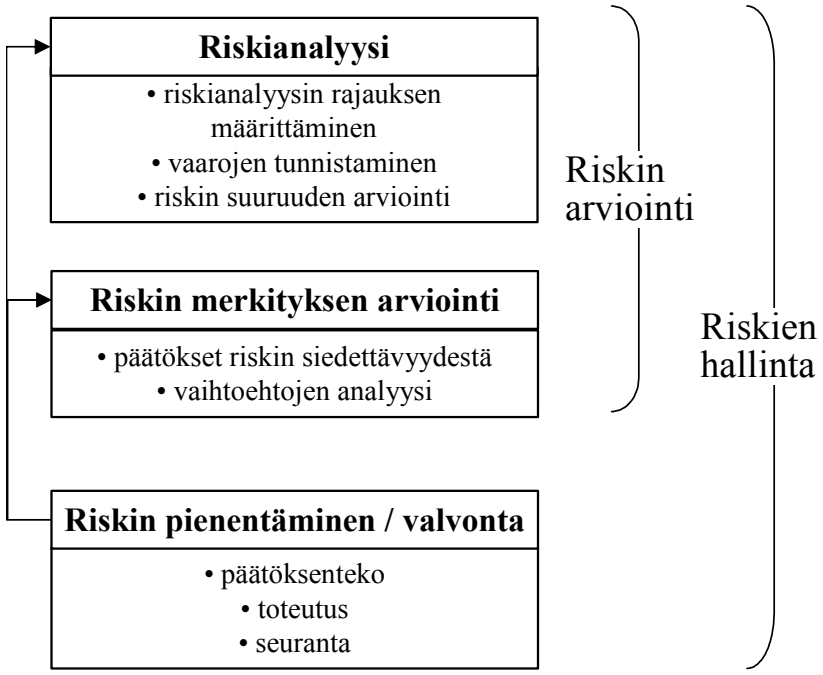
Standardin soveltamien tarjoaa valmistajalle kustannustehokkaan ja systemaattisen mallin ohjelmiston vaatimusmäärittelyprosessille. Mallin edut korostuvat silloin, kun ohjelmistolta edellytetään erilaisia lakisääteisiä hyväksyntöjä ennen markkinoille saattamista, ja ohjelmiston luotettavuus, suorituskyky, dokumentointi ja vaatimusten jäljitettävyys halutaan arvioida ulkoisen arvioijan toimesta.

Lisätietoa harmonisoiduista standardeista löydät www-osoitteesta:
http://ts.nist.gov/ts/htdocs/210/gsig/eu-guides/sp951/contents.htm

Liite D: Riskianalyysiprosessi

Organisaatioiden toimintaan liittyy aina (talous, markkinointi, tukiprosessit, suunnittelu, valmistus ja henkilöstön toiminta) erilaisia riskejä ja vaaratekijöitä. Riskienhallinnan tavoite on ennalta ehkäistä, valvoa ja poistaa näiden toimintaa uhkaavien riskien toteutumista.

Riskienhallintaprosessi on kokonaisvaltainen yrityksen toimintaa ohjaava tukiprosessi, jolla voidaan kartoittaa yrityksen toimintaan tai tuotteen suunnitteluun liittyviä riskejä. Riskienhallintaprosessi [SFS-IEC 60300-3-9] sisältää kuvan D1 mukaiset vaiheet.



SFS-IEC 60300-3-9:2000

Kuva D 1. Riskianalyysin ja muiden riskin hallintatoimintojen yksinkertaistettu riippuvuus.

Standardi [SFS-IEC 60300-3-9] on käyttänyt seuraavia määritelmiä termeistä:

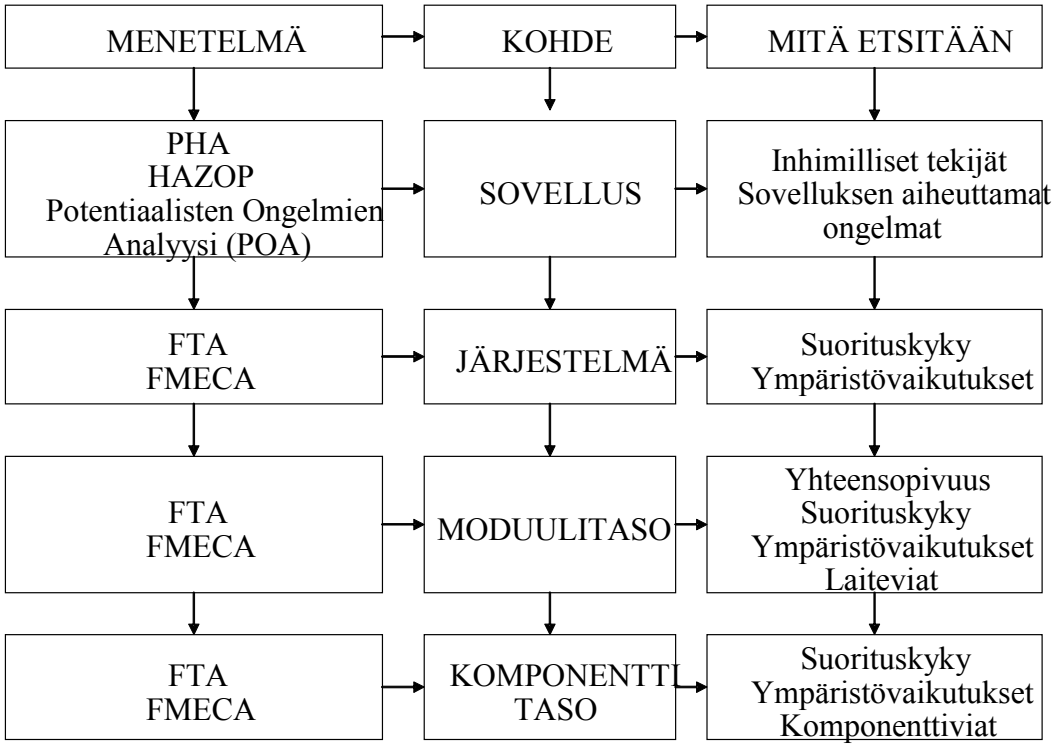
Riskianalyysi	Saatavissa olevan tiedon järjestelmällistä käyttämistä vaarojen tunnistamiseksi sekä ihmisiin tai väestöön, omaisuuteen tai ympäristöön kohdistuvan riksin suuruuden arvioimiseksi.
Riskin arviointi	Riskianalyysin ja riskin merkityksen arvioinnin kokonaisprosessi.

Riskin valvonta	Päätöksentekoprosessi riskin hallitsemiseksi ja/tai pienentämiseksi; päätöksentekoprosessin toteuttaminen, täytäntöönpano ja uudelleen arviointi aika ajoin, käyttäen arvioinnin tuloksia yhtenä lähtötietona.
Riskin suuruuden arviointi	Prosessi, jolla mitataan analysoitavien riskien taso. Riskin suuruuden arviointi koostuu seuraavista vaiheista: taajuusanalyysi, seurausanalyysi, ja niiden yhdistäminen.
Riskin merkityksen arviointi	Prosessi, jossa tehdään päätökset riskin siedettävyydestä riskianalyysin perusteella ottamalla huomioon esimerkiksi sosioekonomiset ja ympäristölliset näkökohdat.
Riskienhallinta	Johtamisperiaatteiden, menettelytapojen ja käytäntöjen järjestelmällistä hyväksikäyttämistä riskien analysoimiseksi, merkityksen arvioimiseksi ja valvomiseksi.

Liite E: Analyysimenetelmien valintaan vaikuttavia tekijöitä

Valittavien riskianalyysimenetelmien on oltava tunnustettuja ja tieteellisesti päteviä, ja niiden pitää soveltua analysoitavaan järjestelmään. Usein 2–3 analyysinmenetelmän hyvä tuntemus on riittävä. Menetelmien käyttöön on koulutettava riittävä määrä henkilöitä, jotta menetelmää voidaan käyttää työkaluna siten, että analyysi on jäljitettävissä, toistettavissa ja verifioitavissa, ja se tuottaa tulokset muodossa, joka auttaa riskin luonteen ymmärtämisessä ja valvonnassa.

Riskienhallinnassa tutkittava kohde voidaan jakaa useisiin osatekijöihin, jolloin kuhunkin tarkasteltavaan osatehtävään voidaan helpommin valita oikea menetelmä (ks. kuva G1).



Kuva G 1. Riskianalyysimenetelmän valinta riippuu analysoitavasta kohteesta.

Analyysimenetelmän valintaan vaikuttavat analysoitavan kohteen lisäksi seuraavat tekijät:

- saatavilla olevan tiedon määrä ja laatu (kvantitatiivisten menetelmien käyttöä voi joissain tapauksissa rajoittaa saatavilla olevan tiedon määrä tai luotettavuus, jolloin on tapauskohtaisesti arvioitava käytetäänkö kvantitatiivisia - vai kvalitatiivisiä - menetelmiä)

- onko analyysin tai tuotteen kriittiset tekijät tunnistettu (vikamuodot, avainsanat, huipputapahtumat, tutkittava taso jne.)

- kohteen tarkka rajaus (analysoitavan kohteen koko, rajapinnat jne.).

Riskianalyysi edellyttää systemaattista ajattelutapaa, ja lähes poikkeuksetta sen suorittaminen vaatii usean ihmisen muodostaman tiimin. Tiimissä on oltava usean eri alan ammattilaisia, joilla on esimerkiksi vahva käytännön kokemus ja projektin vetokyky, sovellusalueosaaminen, ohjelmistotekniikan osaaminen sekä RAMS-teknologioiden tuntemus.

Siitä huolimatta joskus käy niin, että analyysin suoritus on epäonnistunut, mikä näkyy toteutuneina vaaratilanteina tai muina toimintahäiriöinä. Analyysin epäonnistumiseen vaikuttavia tekijöitä voivat olla:

- kokemusperäisen tiedon puuttuminen

- systemaattisen työtavan puuttuminen

- vikaketjusta poikkeaminen

- väärä rajaus tai kattavuus

- vikamuoto tai vika on tunnistettu väärin

- analyysiä on ruvettu tekemään liian alhaalta, vaaralliseksi tapahtumaksi on haettu järjestelmän sisäinen tapahtuma, jolloin järjestelmän ulkopuolella oleva vaikutus jää analysoimatta.

Menetelmien tulee sisältää myös tehtyjen riskienhallintatoimenpiteiden vaikuttavuuden arviointi, jossa arvioidaan tehtyjen toimenpiteiden onnistumista ja vaikuttavuutta.

Tässä yhteydessä ei käsitellä ALARP-periaatetta eikä riskin luokittelua. Näistä saa lisätietoja standardista IEC 606158-5.

Julkaisija

Julkaisun sarja, numero ja raporttikoodi

VTT Tiedotteita 2263
VTT–TIED–2263

Tekijä(t)
Pöyhönen, Ilkka & Hukki, Kristiina

Nimeke

Riskitietoisen ohjelmiston vaatimusmäärittelyprosessin kehittäminen

Tiivistelmä

Vaatimusmäärittely on ratkaisevan tärkeä vaihe ohjelmistoa sisältävien järjestelmien suunnittelussa. Tämän vuoksi vaatimusmäärittelyprosessin kehittämisen arvioidaan nostavan ohjelmistojen laatua merkittävästi. Raportissa tarkastellaan luotettavuus- ja turvallisuusvaatimuksia sisältävien ohjelmistojen vaatimusmäärittelyprosessin kehittämiseen liittyviä näkökohtia sekä teknisestä että prosessiin osallistuvien asiantuntijoiden vuorovaikutuksen näkökulmasta. Kehittämiskohteiden valinta ja yhdenmukaistettujen standardien soveltuvuuden arviointi omaan toimintaan edellyttää yritykseltä kriittistä ja perinpohjaista tarkastelua. Raportissa esitetään keinoja, joiden avulla yrityksissä pystytään kehittämään yrityskohtaista riskitietoista vaatimusmäärittelyprosessia. Tarkastelun kohteena ovat standardien ja riskienhallinnan tarjoamaan tukeen perustuvat menettelytavat. Niiden avulla voidaan parantaa vaatimusmäärittelyn jäljitettävyyttä ja ohjelmiston luotettavan toiminnan kannalta kriittisten vaatimusten tunnistamista laajasta vaatimusjoukosta. Vaatimusmäärittelyn kehittämistä tukevina keinoina esitetään harmonisoitujen standardien soveltaminen suunnitteluprosessiin ja vaatimusmäärittelyyn. Tämän lisäksi ehdotetaan laadunhallintajärjestelmien ja riskienhallinnan integroimista osaksi suunnitteluprosessia. Vaatimusmäärittelyprosessia tarkastellaan myös siihen liittyvien eri osapuolten vuorovaikutuksen näkökulmasta. Eri alojen asiantuntemusta edustavien osapuolten tiedonkululla on merkittävä vaikutus määrittelyprosessin onnistumisessa, koska ohjelmistoa koskevat vaatimukset muodostuvat niiden vuorovaikutuksen tuloksena. Raportissa käsitellään tiedonkulun merkitystä ohjelmiston turvallisuuden syntymisessä ja esitetään keinoja, joiden avulla voidaan edistää osapuolten keskinäistä ymmärrystä. Lisääntynyt keskinäinen ymmärrys helpottaa eri alojen asiantuntemuksen yhdistämistä vaatimusmäärittelyprojekteissa ja ohjelmistosuunnittelua koskevassa riskinhallinnassa.

Avainsanat
software requirements specification, risk-informed software engineering process, traceability, risk management, multidisciplinary expertise, expert interaction, knowledge transfer, integration of knowledge, shared thinking models

Toimintayksikkö
VTT Tuotteet ja tuotanto, Tekniikankatu 1, PL 1306, 33101 TAMPERE

ISBN			Projektinumero
951–38–6496–0 (nid.) 951–38–6497–9 (URL: http://www.vtt.fi/inf/pdf/)			G3SU00058
Julkaisuaika Lokakuu 2004	Kieli Suomi, engl. tiiv.	Sivuja 36 s. + liitt. 9 s.	Hinta A
Projektin nimi TRUST		Toimeksiantaja(t) VTT	
Avainnimeke ja ISSN VTT Tiedotteita – Research Notes 1235–0605 (nid.) 1455–0865 (URL: http://www.vtt.fi/inf/pdf/)		Myynti: VTT Tietopalvelu PL 2000, 02044 VTT Puh. (09) 456 4404 Faksi (09) 456 4374	

Published by

Series title, number and
report code of publication

VTT Research Notes 2263
VTT–TIED–2263

Author(s)
Pöyhönen, Ilkka & Hukki, Kristiina

Title

Development of risk-informed requirements specification process of software

Abstract

Requirements specification is the most fundamental phase of software design. Therefore, the development of the requirements specification process is considered to contribute significantly to the quality of software. The report examines aspects related to developing the requirements specification process of safety-critical software. The consideration is made both from technical and expert interaction point of view.

The choice of the objects to be developed and the evaluation of the applicability of the harmonized standards to the company's activities require critical and thorough consideration. The report introduces procedures which help in developing the company-specific risk-informed requirements specification. The focus is on procedures which are based on the support provided by standards and risk management. These procedures make it possible to improve the traceability of requirements specification and to facilitate the identification of those requirements which are critical from the safety point of view. The suggested procedures are the application of the harmonized standards to the design process and to requirements specification, and, in addition, the integration of quality management system and risk management as part of the design process.

The requirements specification process is considered also from the viewpoint of different parties' interaction. The requirements are formed as the outcome of the experts' interaction. Therefore, the knowledge transfer between the experts, representing different disciplines, influences prominently the success of the specification process. The report examines the significance of knowledge transfer in the formation of software safety. In addition, procedures are introduced which help in enhancing the experts' mutual understanding. Enhanced mutual comprehension facilitates the integration of multidisciplinary expertise in the requirements specification projects and in the risk management of software engineering.

Keywords
software requirements specification, risk-informed software engineering process, traceability, risk management, multidisciplinary expertise, expert interaction, knowledge transfer, integration of knowledge, shared thinking models

Activity unit
VTT Industrial Systems, Tekniikankatu 1, P.O.Box 1306, FIN–33101 TAMPERE, Finland

ISBN	Project number
951–38–6496–0 (soft back ed.) 951–38–6497–9 (URL: http://www.inf.vtt.fi/pdf/)	G3SU00058

Date	Language	Pages	Price
October 2004	Finnish, Engl. abstr.	36 p. + app. 9 p.	A

Name of project	Commissioned by
TRUST	Technical Research Centre of Finland VTT

Series title and ISSN	Sold by
VTT Tiedotteita – Research Notes 1235–0605 (soft back edition) 1455–0865 (URL: http://www.vtt.fi/inf/pdf/)	VTT Information Service P.O.Box 2000, FIN–02044 VTT, Finland Phone internat. +358 9 456 4404 Fax +358 9 456 4374

Vaatimusmäärittely on ratkaisevan tärkeä vaihe ohjelmistoa sisältävien järjestelmien suunnittelussa. Tämän vuoksi vaatimusmäärittelyprosessin kehittämisen arvioidaan nostavan ohjelmistojen laatua ja turvallisuutta merkittävästi.

Hankkeessa tutkitaan luotettavuus- ja turvallisuusvaatimuksia sisältävien ohjelmistojen vaatimusmäärittelyprosessiin liittyviä ongelmia sekä teknisestä että prosessiin osallistuvien asiantuntijoiden vuorovaikutuksen näkökulmasta.

Lisäksi esitetään keinoja, joiden avulla yritykset voivat kehittää omaa riskitietoista vaatimusmäärittelyprosessia. Tarkastelun kohteena ovat standardien ja riskienhallinnan tarjoamat menettelytavat prosessin kehittämiseksi sekä keinot, joilla voidaan parantaa suunnitteluprosessiin liittyvien henkilöstöryhmien välistä tiedonkulkua.

Tätä julkaisua myy
VTT TIETOPALVELU
PL 2000
02044 VTT
Puh. (09) 456 4404
Faksi (09) 456 4374

Denna publikation säljs av
VTT INFORMATIONSTJÄNST
PB 2000
02044 VTT
Tel. (09) 456 4404
Fax (09) 456 4374

This publication is available from
VTT INFORMATION SERVICE
P.O.Box 2000
FIN–02044 VTT, Finland
Phone internat. + 358 9 456 4404
Fax + 358 9 456 4374

ISBN 951–38–6496–0 (nid.)
ISSN 1235–0605 (nid.)
ISBN 951–38–6497–9 (URL: http://www.vtt.fi/inf/pdf/)
ISSN 1455–0865 (URL: http://www.vtt.fi/inf/pdf/)