

# **Kriittisten tietojärjestelmien muutoksen hallinta**

Hannu Tanhuamäki

Tampereen yliopisto  
Tietojenkäsittelytieteiden laitos  
Tietojenkäsittelyoppi

Pro gradu -tutkielma  
Helmikuu 2006

Tampereen yliopisto  
Tietojenkäsittelytieteiden laitos  
Hannu Tanhuamäki: Kriittisten tietojärjestelmien muutoksen hallinta  
Pro gradu -tutkielma, 71 sivua, 1 liitesivu  
Helmikuu 2006

---

Tässä tutkielmassa tutkitaan kriittisten tietojärjestelmien muutoksen hallintaa. Kriittisiä tietojärjestelmiä ovat järjestelmät, joiden pettäminen johtaa menetyksiin, joita ei voida sallia. Tällaisia menetyksiä olisivat esimerkiksi ihmishenkien menetykset. Muutoksen hallinnalla tarkoitetaan prosessien, rakenteiden, tekniikan, henkilöstön ja kulttuurin muutosten hallintaa organisaation sisällä.

Tarkastelussa keskitytään järjestelmien turvallisuuteen vaikuttaviin asioihin. Tutkielmassa käsitellään ongelmien aiheutumista, ennakointia ja niihin reagointia. Pääkysymyksenä pohditaan, miten ohjelmiston turvallisesta käytöstä voidaan muutoksen aikana varmistua, jos yksikin vikaantuminen voi aiheuttaa liian suuria menetyksiä. Tutkielmassa arvioidaan teoreettisen näkökulman lisäksi todellisen kriittisen järjestelmän muutoksenhallintaprosessia.

Tutkielmassa käsitellään yksi tietojärjestelmän käyttöönottoprosessi. Tarkastelussa keskitytään niihin seikkoihin, joilla järjestelmä voitiin ottaa turvallisesti käyttöön. Tuloksena saatiin joukko näkökulmia, jotka tulee ottaa huomioon kriittisen järjestelmän muutosprosessin aikana. Näkökulmia tulee soveltaa järjestelmän kriittisyyden ja laajuuden mukaan sekä todellista käyttöympäristöä mukaillen.

Avainsanat: kriittiset tietojärjestelmät, muutoksen hallinta, turvallisuus, ongelmien ennakointi.

## Sisällys

1.	Johdanto .....	1
2.	Tutkimusongelma .....	4
3.	Tutkimusprosessi .....	6
3.1.	Tutkimuksen lähestymistapa .....	6
3.2.	Tutkimusmenetelmät .....	7
3.3.	Tiedonkeruumenetelmät.....	8
3.4.	Tutkimuksen toteutus .....	9
4.	Kriittiset tietojärjestelmät .....	10
4.1.	Määritelmiä .....	10
4.2.	Tietojärjestelmien kriittisyys .....	11
4.3.	Kriittisten tietojärjestelmien ongelmat.....	12
4.3.1.	Viat sekä vikojen ja vaaran aiheuttajat.....	13
4.3.2.	Luotettavuuden mittaaminen.....	15
4.3.3.	Ongelmien luokittelu.....	18
4.4.	Reagointi vaaran aiheuttajiin ja ongelmiin.....	18
4.4.1.	Vaaratilanteiden syntyminen .....	18
4.4.2.	Reagointi vaaraan.....	19
4.4.3.	Vaaratekijöiden poistaminen.....	20
4.5.	Kriittisten tietojärjestelmien tietoturvallisuus .....	21
4.5.1.	Tekninen turvallisuus.....	22
4.5.2.	Fyysinen turvallisuus .....	23
4.5.3.	Toiminnallinen turvallisuus .....	23
4.5.4.	Ongelmat kriittisten tietojärjestelmien turvaamisessa.....	24
4.6.	Turvallisuuden varmistaminen .....	25
5.	Muutoksen hallinta .....	27
5.1.	Muutostyypit .....	27
5.2.	Organisaatiotason muutoskohteet.....	28
5.3.	Muutos ohjelmistotuotteen kannalta .....	28
5.4.	Muutosten luokittelu .....	29
5.5.	Muutokset liiketoiminnan prosessien näkökulmasta.....	30
5.5.1.	Liiketoimintaprosessien uudelleenjärjestäminen.....	30
5.5.2.	Liiketoimintaprosessien parantaminen .....	31
5.5.3.	Liiketoimintaprosessien automatisointi.....	31
5.6.	Muutokset yksilön näkökulmasta .....	31
5.7.	Organisaation kulttuuri ja muutokset.....	33
5.8.	Muutoksen vaiheet .....	34

6.	Kriittisten tietojärjestelmien turvallisuusvaatimukset ilmailussa .....	36
6.1.	Lennonvarmistusohjelmistojen turvallisuusvaatimukset.....	36
6.1.1.	Yleiset turvallisuusvaatimukset.....	36
6.1.2.	Ohjelmistojen turvallisuuden varmistusjärjestelmä .....	37
6.1.3.	Ohjelmistojen turvallisuustasot .....	37
6.1.4.	Ohjelmistojen vaatimusten varmistaminen.....	37
6.1.5.	Ohjelmistojen toiminnan varmistaminen .....	38
6.1.6.	Ohjelmistojen asetusten hallinnan varmistaminen .....	38
6.1.7.	Ohjelmiston vaatimusten jäljitettävyys järjestelmän vaatimuksiin.....	38
6.2.	ESARR-vaatimusten toteuttaminen Suomessa.....	38
6.3.	Ilmailukenteen hallintapalvelun tekninen henkilöstö .....	38
6.4.	Järjestelmien käyttöönotto .....	40
6.4.1.	Käyttöönoton hyväksyntä.....	40
6.4.2.	Käyttöönottokriteerit .....	42
6.5.	Suunnittelun merkitys ilmailun tietojärjestelmissä .....	43
7.	Case: Kriittinen tietojärjestelmä Ilmailulaitoksessa.....	46
7.1.	Järjestelmä ja sen käyttötarkoitus .....	47
7.2.	Projektin vaiheet.....	49
7.2.1.	Ohjelmiston hankinta ja määrittelyt .....	49
7.2.2.	Testaus .....	49
7.2.3.	Koulutus .....	50
7.2.4.	Käyttöönotto .....	51
7.3.	Kriittisyyden huomioiminen.....	51
7.3.1.	Henkilöstön rooli toiminnan varmistamisessa .....	52
7.3.2.	Vaaratekijöiden tunnistaminen.....	52
7.3.3.	Vaaratekijöiden luokittelu.....	53
7.3.4.	Vaaratekijöiden kartoitus ja turvallisuustavoite.....	54
7.3.5.	Tekninen riskianalyysi.....	55
7.4.	Järjestelmän viat .....	55
7.5.	Operatiivinen toiminta vikatilanteissa.....	55
7.6.	Tekninen näkökulma .....	56
8.	Muutos ja sen hallinta Case Ilmailulaitoksessa.....	59
8.1.	Toiminnallisten prosessien muutos.....	59
8.2.	Organisaation muutoskohteet.....	59
8.3.	Muutos ohjelmistotuotteen kannalta .....	60
8.4.	Muutostyyppi ja luokittelu .....	60
8.5.	Muutos yksilön näkökulmasta.....	61
8.6.	Organisaation kulttuuri ja muutos .....	62

8.7. Arvio muutoksen hallinnasta.....	63
9. Tulokset.....	64
10. Yhteenveto ja suositukset.....	68

Viiteluettelo.....sivulla 69

#### Liitteet

Liite 1 Risk Assessment and Mitigation in AMT- ESARR 4

## 1. Johdanto

Tietojärjestelmien käyttö alueilla, joilla toimintahäiriö tai vikaantuminen voi aiheuttaa vaaraa ihmishengelle, omaisuudelle tai ympäristölle, on lisääntynyt viime vuosien aikana. Tietojärjestelmät ovat levinneet uusiin toimintaympäristöihin samalla, kun perinteisillä toiminta-alueilla järjestelmiä uusitaan muuttuvien palvelutarpeiden vaatimuksesta. Kehityksen ja kasvaneiden vaatimusten myötä toimintaprosessit ovat monimutkaistuneet, minkä seurauksena myös ohjaus- ja valvontajärjestelmät ovat monimutkaistuneet. Kriittisissä toimintaympäristöissä on tärkeää varmistua siitä, että järjestelmät toimivat oikein ja että mahdollisessa vikatilanteessa menetyksiä ei syntyisi.

Palveluvaatimusten muuttuminen luo tarpeen järjestelmien muuttamiseen ja uusien järjestelmien käyttöönottoon. Sujuva muutosprosessi on ensiarvoisen tärkeää silloin, kun järjestelmä toimii kriittisessä ympäristössä. Kriittisen järjestelmän vikaantuminen voi pahimmillaan johtaa ihmishenkien menetykseen. Tällaisten järjestelmien osalta ei riitä, että voidaan osoittaa vikojen esiintyvän harvoin. Siksi onkin voitava muulla tavoin varmistua siitä, että vika ei johda menetyksiin tai että vikoja ei esiinny lainkaan.

Kriittisessä ympäristössä muutosprosessiin liittyy monia tekijöitä, jotka vaikuttavat muutoksen onnistumiseen. Virhe uuden järjestelmän tai muutoksen suunnittelussa voi johtaa myöhemmin toimintavirheeseen järjestelmässä. Muutokseen liittyy suunnittelijoiden lisäksi monia henkilöstöryhmiä, joiden panos on merkittävä muutoksen onnistumisessa. Esimerkiksi työntekijöiden tai kouluttajien väärä asenne voi toimia toimintahäiriön aiheuttajana. Johdon oikeilla toimilla ja asenteella voidaan vaikuttaa vähentävästi muutokseen miltei aina kuuluvaan negatiiviseen suhtautumiseen. Tutkielmassa kartoitetaan kriittisiin järjestelmiin liittyviä erityispiirteitä. Järjestelmien osalta pohditaan, miten ja mihin asioihin ohjelmistojen suunnittelussa tulee keskittyä, jotta vikoja ei synny tai jos niitä syntyy, niistä ei aiheudu menetyksiä.

Kriittisiä järjestelmiä ja muutoksen hallintaa koskevia tutkimuksia on tehty runsaasti. Nämä käsittelevät kuitenkin pääsääntöisesti vain toista aihealuetta. Perehtymällä molempiin aihealueisiin ja käsittelemällä näiden erityispiirteitä, voidaan määritellä kriittisyyden vaikutus muutoksen hallintaan. Tutkimalla aitoa reaali maailman kriittisen järjestelmän muutoksen hallintaprosessia, voidaan teoreettisia löydöksiä verrata, arvioida ja täydentää.

Tutkimus pohjautuu sekä kirjallisiin lähteisiin että omaan kokemukseen työskentelystä kriittisten järjestelmien parissa. Olen työskennellyt kriittisten

tietojärjestelmien parissa yli kymmenen vuotta. Työtehtäväni Tampereen alueenjohtajana ovat olleet erittäin monipuolisia. Työkokemusta kertyi aluksi operatiivisten tietojärjestelmien käyttäjänä ja myöhemmin niiden suunnittelijana ja kehittäjänä. Tässä tutkielmassa esitellään eräs Ilmailulaitoksen tietojärjestelmä. Tietojärjestelmällä huolehditaan ilmatilavarausten koordinoinnista eri yksiköiden välillä ja näytetään sekä reaaliaikaista ilmatilannekuvaa että ennakkotietoa tulevista varauksista. Olen ollut mukana tässä projektissa huolehtien projektiin liittyvistä asioista teknisestä näkökulmasta katsoen.

Ongelmaa lähestytään tässä tutkimuksessa perehtymällä kriittisten järjestelmien ja muutoksen hallinnan erityispiirteisiin kirjallisuuslähteiden avulla. Tutkimuksessa esiteltävän tietojärjestelmän käyttöönotosta olen tehnyt omia havaintoja, jotka perustuvat tietoihin, joita sain osallistuessani projektin kokouksiin, testauksiin, asennuksiin ja varsinaiseen käyttöönottoon sekä uuden järjestelmän käytön seuraamiseen. Lisäksi käytetään myös muita asiakirjoja, kuten pöytäkirjoja, työohjeita ja määräyksiä.

Tutkimuksen tuloksena löytyi keinoja turvallisuuden varmistamiseen kriittisten järjestelmien muutoksenhallintaan liittyvissä tilanteissa. Varmistamiseen löytyi useita näkökulmia, joiden avulla järjestelmien turvallista käyttöä voidaan edistää ja varmistua niiden oikeasta, ennakoitavasta toiminnasta. Turvallisen toiminnan varmistaminen alkaa jo ennen ohjelmiston määrittelyä. Suunnitteluvirheet ovat hankalia, koska ne tyypillisesti löydetään vasta ohjelmiston operatiivisen käytön aikana. Ongelmia voi syntyä niin ohjelmiston virheellisen toiminnan johdosta kuin käyttäjien tekemien virheiden johdosta. Käyttäjät eivät kuitenkaan välttämättä ole vahingon varsinaisia aiheuttajia, vaan syynä voi olla ympäristöön liittyvät tekijät tai organisaatiotason asiat, jotka ovat mahdollistaneet haitalliset ympäristötekijät. Tunnistamalla vikatilanteet, mieluiten jo ennalta, voidaan estää järjestelmää menemästä vaaralliseen tilaan, kiertämään vaarallinen tila tai minimoimaan järjestelmästä aiheutuvat vahingot.

Johdannon jälkeen luvussa kaksi määritellään tarkemmin tutkimusongelma ja perehdytään aihealueeseen. Luvussa kolme kuvataan tutkimusprosessi. Siinä esitellään tutkimuksen lähestymistapa, menetelmät, tiedon kerääminen ja toteutuksen kuvaus. Seuraavaksi neljännessä luvussa määritellään kriittiset järjestelmät. Luvussa perehdytään kriittisyyden käsitteeseen, järjestelmien ongelmiin, niiden aiheuttajiin ja keinoihin, joilla niitä voidaan ehkäistä. Lopuksi käsitellään tietoturvallisuutta ja turvallisuuden varmistamista. Luvussa viisi käsitellään muutoksen hallintaa. Muutoksen hallinnassa tutustutaan eri muutostyyppeihin ja siihen, millä tavoin ne vaikuttavat organisaatioon. Lisäksi

tarkastellaan muutosta eri näkökulmista. Luvussa kuusi perehdytään ohjelmistoturvallisuuden ilmailussa. Siinä käsitellään sekä kansainvälisiä määräyksiä että kansallisia määräyksiä. Luvussa seitsemän esitellään tutkimuksessa käsitelty case ja luvussa kahdeksan käsitellään muutosta ja sen hallintaa Case Ilmailulaitoksessa. Luvussa yhdeksän esitellään tutkimuksen tulokset ja luvussa kymmenen tehdään yhteenveto, kuvataan rajoitukset, annetaan soveltamisohjeita ja jatkotutkimusaiheita.

## 2. Tutkimusongelma

Erilaiset tietojärjestelmät ovat keskeisessä roolissa yhä useamman kriittisen asian tai tehtävän hoitamisessa. Järjestelmillä voi olla tehtävän hoitamisen lisäksi kriittisten toimintojen valvontaan liittyviä tehtäviä. Kriittisiä järjestelmiä on perinteisesti tunnistettu olevan käytössä voimalaitoksissa, terveydenhuollossa, ilmailu- ja avaruusteknologiassa, mutta tänä päivänä niitä tunnistetaan myös muista ympäristöistä. Uusista alueista on tullut mukaan esimerkiksi puhelinverkot, jotka sinällään eivät välttämättä täytä kriittisille järjestelmille annettuja kriteereitä. Kuitenkin, jos niitä käsitellään esimerkiksi osana hätäkeskusten hälytysjärjestelmää, kukaan ei kiistä niiden kriittistä merkitystä pelastuspalvelun hälyttämisessä. Rakennettaessa tietoliikenneverkkoja ei vielä välttämättä tunneta palveluita ja niiden kriittisyyttä. Palvelun kriittisyys tekee järjestelmästä kriittisen. Ei riitä, että muutoksessa varmistutaan uuden järjestelmän turvallisuudesta. Arvioinnissa on oltava mukana myös vanhat järjestelmät, joita käytetään uudella tavalla tai tuottamaan uutta palvelua.

Uusien tietojärjestelmien käyttöönoton lisäksi vanhat järjestelmät tarvitsevat uusia ominaisuuksia muuttuvan ympäristön painostaessa sujuvampaan ja tehokkaampaan asioiden hoitamiseen. Järjestelmissä muutokset edellyttävät myös ympäristöltä ja käyttäjiltä sopeutumista uuteen tilanteeseen. Järjestelmiä uusittaessa joudutaan siirtymään usein ilman toimintakatkosta uusiin työtapoihin, samalla kun tehtävien hoitaminen siirtyy uuden järjestelmän tai ohjelmaversioon hoidettavaksi. Tähän muutosprosessiin liittyy monia epävarmuustekijöitä ja uhkia, joita tässä tutkielmassa pyritään ratkaisemaan. Avainkysymyksiä on, miten voidaan varmistua kriittisen järjestelmän turvallisesta käytöstä, jos yksikin vikaantuminen aiheuttaa sellaisia menetyksiä, joita ei voida sallia. Ratkaisevia tekijöitä ovat ongelmien tunnistaminen, estäminen, kiertäminen ja vahinkojen minimoiminen. Järjestelmän toiminnan hyvä tunteminen ja sitä kautta ennustettavuus voi mahdollistaa heikotasoisenkin järjestelmän turvallisen käytön.

Aluksi tässä tutkielmassa tutkitaan kriittisiä järjestelmiä yleensä, sekä selvitetään niihin liittyviä erityispiirteitä. Tutkielmassa etsitään ongelmien aiheuttajia sekä arvioidaan mittareita, joilla luotettavuutta yritetään mitata. Lisäksi pohditaan ongelmien ennakoimista ja poistamista.

Muutoksen hallintaa tutkittaessa tunnistetaan eri tapoja muuttaa järjestelmiä. Muutoksen hallinnasta pyritään löytämään vastauksia kysymyksiin, mitä ongelmia muutoksen hallintaan yleensä liittyy ja millä

muutoksen hallinnan toimilla ongelmia voidaan vähentää. Näiden erityispiirteitä arvioidaan suhteessa kriittisiin tietojärjestelmiin.

Kirjallisten lähteiden lisäksi käytetään esimerkkinä yhtä tapausta, jossa uusittiin kriittinen tietojärjestelmä. Tapausta tutkimalla pyritään löytämään ne käytännön toimet, joilla turvalliseen käyttöönottoon pyrittiin ja miten ne käytännössä onnistuivat.

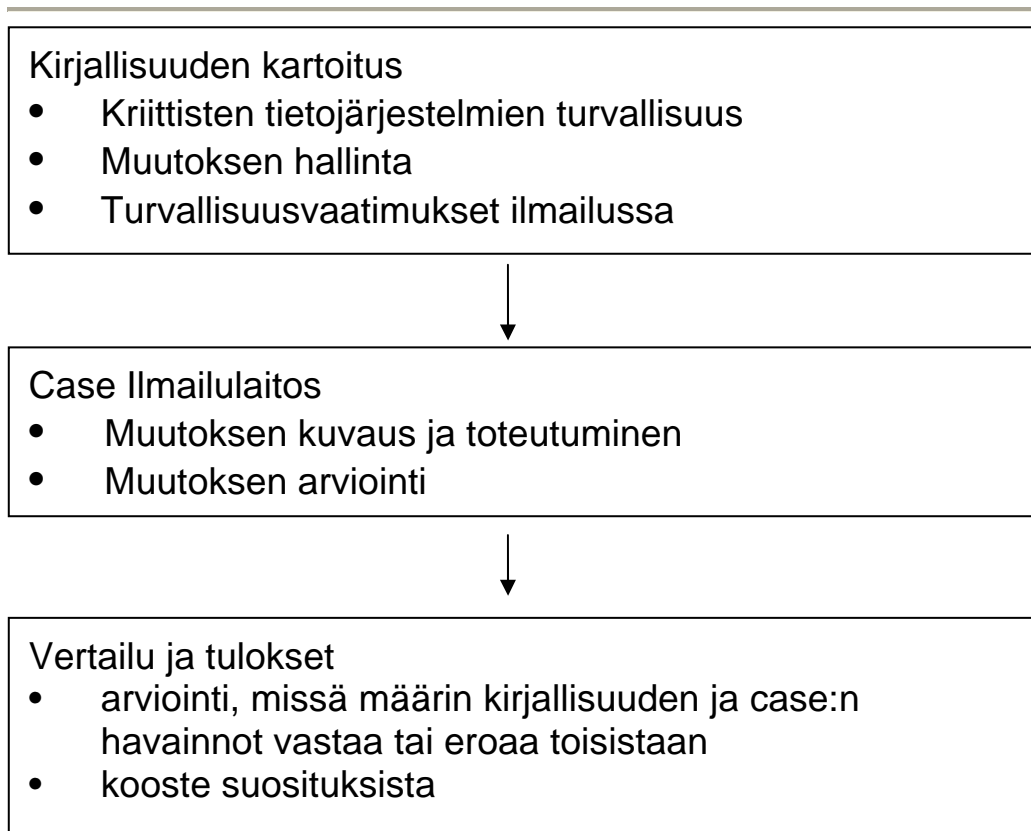
Lopuksi tutkielmassa kootaan yhteen ne seikat, jotka tulee ottaa huomioon tehtäessä muutoksia kriittisiin tietojärjestelmiin. Vertaamalla aiempia tutkimustuloksia case:n avulla saatiin tuloksiin syntyi lista niistä toimista, jotka tulisi ottaa huomioon suunniteltaessa kriittisiä tietojärjestelmiä ja niiden saattamisessa operatiiviseen käyttöön.

### 3. Tutkimusprosessi

Tässä luvussa tarkastellaan tutkimuksen lähestymistapaa, tutkimusmenetelmiä ja tiedonkeruumenetelmiä.

#### 3.1. Tutkimuksen lähestymistapa

Tässä tutkimuksessa tutkitaan, miten kriittisten tietojärjestelmien muutoksen hallinnassa toiminnan turvallisuudesta voidaan varmistua. Tutkimus jakaantuu kolmeen osaan (kuva 1, tutkielman rakenne). Tutkimuksessa käytetään hyväksi ja sovelletaan aiempien tutkimusten tuloksia. Niiden avulla yritetään ymmärtää ilmiöiden säännönmukaisuuksia ja piirteitä. Tietoja kerätään myös toteutuneen case:n tutkimisella. Lopuksi tuloksia arvioidaan, jotta tiedettäisiin, saavutettiinkö hyödyllinen tulos. Koska tuloksena syntyi lista käytännön toimista, annetaan lopussa suosituksia niiden soveltamiseksi. Kyseessä on konstrukttiivinen tutkimus, jolle on luonteenomaista uuden todellisuuden rakentaminen olemassa olevan (tutkimus)tiedon pohjalta [Järvinen ja Järvinen, 2000, s.102].



Kuva 1, Tutkielman rakenne

Tutkimuksella pyritään vastaamaan kysymyksiin, millä tavoin kriittiset järjestelmät eroavat muista järjestelmistä ja mitä tulee huomioida suunniteltaessa ja käyttöön otettaessa kriittisiä tietojärjestelmiä. Tutkimuksessa pyritään ymmärtämään aiempien tutkimuslöydösten soveltuvuutta tutkimalla teoreettisen katsauksen rinnalla todellista kriittisen järjestelmän käyttöönottoprosessia. Teoreettisen pohjan ja käytännön tapauksen avulla pyritään luomaan suosituksia, joiden käyttöä tulisi harkita kriittisen järjestelmän muutosprosessissa.

Tutkimuksen lähestymistapa on tässä tutkimuksessa deduktiivinen. Liikkeelle on lähdetty olemassa olevasta teoriasta. Teoriat tuottavat hypoteeseja, jotka ohjaavat havaintojen tekoa. Havaintojen avulla pyritään edelleen tuottamaan empiirisiä yleistyksiä. Tällä tavoin vertaamalla yleistykset saattavat edelleen aiheuttaa muutoksia teoriaan. Teoria on ymmärrettävissä tässä tutkimuksessa siis vahvistusta saaneena hypoteesina [Uusitalo, 2001, s.42].

### 3.2. Tutkimusmenetelmät

Tutkimusongelmana tässä tutkimuksessa etsitään käytännön keinoja, jotka huomioimalla voidaan parantaa kriittisten tietojärjestelmien muutoksen hallintaa siten, että vikaantuminen ei johtaisi korvaamattomiin menetyksiin.

Teoreettista osuutta verrataan todelliseen kriittisen järjestelmän käyttöönottoon. Vertailua varten aineistoa on kerätty määräyksistä, yrityksen sisäisistä asiakirjoista, tekemällä havainnot ja haastatteluja. Havainnoinnin etuna on se, että se tapahtuu tyypillisesti tutkimuskohteen luonnollisessa ympäristössä [Uusitalo, 2001]. Toiminnan ja käyttäytymisen kuvaaminen ja niiden ymmärtävä tulkitseminen on tärkeää. Havainnot tehtäessä kriittistä järjestelmää ei tulisi irrottaa ympäristöstään, koska tällöin järjestelmä voi menettää kriittiset piirteensä. Järjestelmien käyttöä ei voida mitenkään täysin ymmärtää ymmärtämättä niiden käyttöympäristöä. Järjestelmän vuorovaikutus ympäristön ja käyttäjien kanssa muodostaa tärkeän osan kokonaisuuden ymmärtämisessä. Vallitsevilla olosuhteilla voi joskus olla merkittävämpi rooli kuin itse järjestelmällä.

Tutkimusmenetelmäksi tähän tutkielmaan on valittu konstruktiivinen tutkimus. Tutkimuksen tavoitteena on uuden kattavamman tiedon tuottaminen jo olemassa olevan tutkimustiedon ja tutkitun case:n avulla. Aiempia tutkimustuloksia täsmälleen tämän tyyppisestä ongelmasta ei ole käytettävissä, ja siksi kriittisiä tietojärjestelmiä ja muutoksen hallinnan erityispiirteitä on jouduttu tarkastelemaan osittain erillisinä kokonaisuuksina. Case:n tutkiminen soveltuu hyvin tähän tilanteeseen, jossa perehdytään reaali maailman ongelmiin. Aidon käyttöympäristön käyttö ei muutoin olisi mahdollista. Tässä

tutkimuksessa verrataan missä määrin aiempien tutkimusten tiedot vastaavat tai poikkeavat tutkitun case:n avulla saaduista tiedoista. Tutkimusasetelmasta on pyritty luomaan mahdollisimman realistinen. Siksi tutkimuksen tekemiseen ja aineiston keräämiseen ei ole etukäteen tehty tarkkoja rajauksia. Yin [1989] määrittelee case-tutkimuksen : "Case-tutkimus on empiirinen tutkimusote, joka tutkii tämän päivän ilmiötä sen todellisessa kontekstissa, kun ilmiön ja kontekstin rajapinta ei ole selkeä, ja jossa käytetään monia evidenssin lähteitä". Tähän tutkimukseen on pyritty keräämään tietoa mahdollisimman monista erityyppisistä lähteistä.

Tutkimuksessa tarkastellaan yhtä käyttöönottoprosessia. Kriittisten järjestelmien käyttöönotot, siten kuin ne tässä tutkimuksessa on määritelty, on erittäin harvinaisia. Itse olen päässyt osallistumaan tällaiseen käyttöönottoon ja siten on ollut mahdollista tehdä havaintoja prosessin kulusta. Toimintatutkimuksessa tutkija osallistuu tutkittavan kohteen toimintaan tutkijan tai konsultin roolissa muutosaganttina [Järvinen ja Järvinen, 2000, s. 129]. Olen tehnyt osallistuvaa havainnointia, mikä tarkoittaa sitä, että tutkija osallistuu ryhmän toimintaan sen yhtenä jäsenenä [Uusitalo, 2001, s. 90]. Etuna tässä tavassa tutkia on se, että tutkija ei luo tutkimustilannetta. Etuna on myös se, että tutkija voi saada käyttöönsä enemmän taustatietoja tai tietää jo valmiiksi paljon enemmän tutkittavasta kohteesta kuin ulkopuoliselle paljastettaisiin. Havaintojen systemaattisuuteen on kiinnitettävä erityistä huomiota. Koska havainnot ovat helppo kirjata myös jälkeenpäin, niiden laatu ei saa kuitenkaan kärsiä.

### 3.3. Tiedonkeruumenetelmät

Aiempaa tutkimusaineistoa on jouduttu keräämään sekä tavallisista että kriittisistä järjestelmistä, koska tutkimuksen kannalta on olennaista ymmärtää, miten kriittisten järjestelmien muutoksen hallinta eroaa tavallisten järjestelmien vastaavasta tilanteesta. Muutoksen hallintaan liittyvää aineistoa on käytännössä saatavilla vain tavanomaisista järjestelmistä. Tutkimuksen kannalta harmillista ovat viitteet joidenkin kriittisten järjestelmien tutkimustulosten olemassaolosta, mutta valitettavan usein niitä ei ole mahdollista saada käsiin, koska niissä viitataan sotilasjärjestelmiin tai liikesalaisuuksiin. Tutkimuksen kannalta oli kuitenkin löydettävissä kattavasti kirjallisia lähteitä.

Tutkimuksen tapauksen tiedot on hankittu keskusteluiden, omien havaintojen ja arkistomateriaalien avulla. Omat havainnot on poimittu tutkijan omista muistiinpanoista, joita olin tehnyt käyttöönottoprosessin aikana. Arkistomateriaaleista on käytetty lähinnä kokouspöytäkirjoja, raportteja ja

tilastoja. Materiaalia täydennettiin projektia tukevilla tieteellisillä tutkimusraporteilla ja kirjallisuuslähteillä.

### **3.4. Tutkimuksen toteutus**

Tutkimusprosessi alkoi tutkimusongelman määrittelemisellä. Tässä tutkimuksessa pyrittiin löytämään ne seikat, jotka mahdollistavat menestyksellisen muutoksen hallinnan kriittisten järjestelmien osalta.

Seuraavaksi valittiin tutkittava Case. Työtehtäviini Etelä-Suomen lennonvarmistuskeskuksessa kuuluu tietojärjestelmien kehitysprojektit. Koska olin osallistunut ilmatilan varausjärjestelmän (AMC-Tool) käyttöönottoon, valitsin sen tutkimuskohteeksi.

Tämän jälkeen kerättiin tutkimusaineisto. Kerättyjen materiaalien avulla luotiin aluksi kuvaus kriittisistä tietojärjestelmistä ja muutoksen hallinnasta. Case:a tutkittaessa kuhunkin kohtaan pyrittiin tuomaan syvyyttä vertaamalla löydöksiä ja toimintatapoja kirjallisuuslähteisiin. Havaintojen avulla tehtyihin löydöksiin pyrittiin löytämään syyt ja taustat, miksi asia on niin kuin se on havainnoitu.

Tutkimusprosessi päättyi, kun osatekijät oli tunnistettu ja niiden vaikutus muutosprosessiin oli arvioitu. Lopuksi tehtiin yhteenveto tutkimuksen päätuloksista. Tuloksena syntyi lista suosituksista, jotka tulisi ottaa huomioon kriittisten tietojärjestelmien muutoksenhallintaprosessissa.

## 4. Kriittiset tietojärjestelmät

Seuraavassa tarkastellaan määritelmien kautta, mitä tarkoitetaan kriittisillä tietojärjestelmillä. Asiaa syvennetään pohtimalla, millaisia ongelmia ne voivat aiheuttaa. Sen jälkeen perehdytään ongelmien syihin ja niiden mittaamiseen. Lopuksi etsitään keinoja ongelmien poistamiseen ja yleensä miten ongelmiin tulisi reagoida.

### 4.1. Määritelmiä

Tässä tutkielmassa perehdytään kriittisiin tietojärjestelmiin. Tietojärjestelmä on ihmisistä, tietojenkäsittelylaitteista, tiedonsiirtolaitteista ja ohjelmista koostuva järjestelmä, jonka tarkoitus on tietoja käsittelemällä tehostaa tai helpottaa jotakin toimintaa tai tehdä toiminta mahdolliseksi [Hallinnon kehittäminen, tietoturvasanasto]. Järjestelmä koostuu niistä komponenteista, jotka toimivat yhdessä annetussa ympäristössä saavuttaakseen annetun päämäärän annetussa ajassa [Ridley, 1983]. Tutkielma käsittelee tietojärjestelmiä kokonaisuutena, johon kuuluu ohjelmistot, jotka suorittavat määriteltyä tehtävää sekä fyysistä laitteistoa, johon ohjelmisto on asennettu. Tietojärjestelmiin viitattaessa voidaan tästä eteenpäin käyttää lyhyesti termiä järjestelmä.

Käsiteltäessä järjestelmien tiloja, toimintoja ja ongelmia käytetään monia termejä, joilla voi olla asiayhteydestä riippuen hieman eri merkityksiä. Kriittisyys järjestelmässä korostuu, kun järjestelmään tulee toimintahäiriö tai sen uhka, joka voi aiheuttaa kohtuuttoman suurta vahinkoa. Toimintahäiriö (failure) esiintyy järjestelmässä, kun toimitettu palvelu eroaa määritellystä palvelusta, joka on sovittu kuvaus odotetusta palvelusta [Abbot, 1990]. Toimintahäiriö voi tarkoittaa määritellyn tehtävän suorittamista virheellisesti tai sen suorittamatta jättämistä.

Virhe (error) on järjestelmän tila, joka voi johtaa järjestelmävikaan (fault), jos mitään ei tehdä [Knight, 2003]. Virhe järjestelmässä voi johtaa toiminnan häiriöön, joka käynnistää tai muutoin edesauttaa toimintahäiriön syntymistä. Järjestelmävian (fault) aiheuttaja on siten virhe järjestelmässä [Abbot, 1990]. Kaikki virheet eivät automaattisesti aiheuta toimintahäiriötä. Järjestelmä voi jättää yksittäisen komponentin vikaantumisen huomioimatta tai se voi reagoida siihen esimerkiksi alustamalla sen uudelleen alusta. Vikoja sietävä (fault tolerant) järjestelmä estää järjestelmävikoja aiheuttamasta toimintahäiriötä [Abbot, 1990].

Kun ongelmien syntymistä käsitellään suhteessa niiden esiintymisen todennäköisyyteen, puhutaan riskistä. Riski (risk) on toiminto, joka voi johtaa

uhkaavaan tilanteeseen tai vahingon syntymiseen. Riskillä tarkoitetaan tässä kahta tilaa. Ensinnäkin sillä tarkoitetaan todennäköisyyttä saattaa järjestelmä vaaralliseen tilaan ja toisaalta sitä, että uhka johtaa menetyksiin [Leveson, 1986].

Käyttövarmuus (dependability) on yksi keskeisistä termeistä, kun puhutaan kriittisistä tietojärjestelmistä. Käyttövarmuus käsittää edelleen ominaisuuksia, joilla järjestelmien luonnetta voidaan arvioida. Knight [2003] jakaa käyttövarmuuden edelleen kuuteen ominaisuuteen: luotettavuus (reliability), saatavuus (availability), käyttöturvallisuus (safety), luottamuksellisuus (confidentiality), eheys (integrity) ja ylläpidettävyys (maintainability). Tietoturvallisuus (security) kuuluu myös olennaisena osana käyttövarmuuteen.

- Luotettavuus tarkoittaa, että järjestelmä toimii oikein silloin kun sitä käytetään,
- saatavuus, että palvelua on toiminnassa kun sitä tarvitaan,
- käyttöturvallisuus, että järjestelmä on turvallinen,
- luottamuksellisuus, että järjestelmää käyttää vain ne, joilla on siihen oikeus,
- eheys, että järjestelmää ei ole luvatta muutettu ja
- ylläpidettävyys, että sillä voi tehdä tarvittavat huoltotoimet.

Uudelle kriittiselle tietojärjestelmälle tulee määritellä käyttövarmuus vaatimukset. Ei ole aivan itsestään selvää, että kriittisten järjestelmien tulisi täyttää kaikki yllä olevat vaatimukset. Esimerkkinä voidaan tarkastella sydämen tahdistinta. Tahdistimella ei varmastikaan ole varsinaisia tietoturva-vaatimuksia (security), mutta käyttöturvallisuuden (safety) ja saatavuuden tulee olla hyvä. Vaikka luotettavuus on toivottava ominaisuus, ei sen tarvitse olla sitä sanan formaalissa merkityksessä. Jos laite vikaantuu ja yksi tahdistus jää puuttumaan, se ei vielä välttämättä ole ongelma, jos se laite käynnistyy uudelleen ja hoitaa jo seuraavan tahdistuksen normaalisti.

#### **4.2. Tietojärjestelmien kriittisyys**

Järjestelmien kriittisyyttä voidaan tarkastella useista näkökulmista. Helpoimmin tunnistettavissa ovat järjestelmät, jotka suoraan vaikuttavat turvallisuuden ylläpitämiseen osallistumalla kriittiseen toimintaan tai ohjaavat vaarallista kokonaisuutta, kuten ydinvoimalaa. Toisaalta ei-kriittisistä järjestelmistä tulee kriittisiä, jos niitä käytetään kriittisessä ympäristössä. Palvelun tuottajalle järjestelmä voi olla kriittinen, jos sen vikaantumisesta voi

seurata liiketoiminnan menettäminen. Seuraavassa käsitellään eri tyyppisiä kriittisyyksiä.

Turvallisuuskriittisiä järjestelmiä ovat ne, joiden vikaantuminen aiheuttaa ihmishenkien menetyksiä, merkittävää vahinkoa omaisuudelle tai ympäristölle. Kriittisiä tietojärjestelmiä on monilla eri toimialoilla. Niitä voi olla esimerkiksi terveydenhuollossa hengityskoneet, tahdistimet ja sädehoidon, tehohoidon ja leikkaussalien järjestelmät tai liikennepuolella junien ohjausjärjestelmät tai lennonvarmistuslaitteet. Toisaalta kriittisiä järjestelmiä syntyy kaiken aikaa myös uusiin ympäristöihin. Puhelinjärjestelmiä ei varmastikaan sellaisenaan voida pitää kriittisinä, mutta sitten kun niitä käytetään osana pelastuspalvelun hälytysjärjestelmää, ei kukaan kyseenalaista niiden kriittisyyttä.

Sommerville [2000] jakaa kriittiset järjestelmät kolmeen päätyyppiin:

- turvallisuuskriittiset
- tehtäväkriittiset ja
- talouskriittiset järjestelmät.

Turvallisuuskriittiset järjestelmät voivat ongelmatilanteessa aiheuttaa loukkaantumisia, henkien menetyksiä tai suuria ympäristöllisiä vahinkoja. Ongelma tehtäväkriittisessä järjestelmässä aiheuttaa tehtävän suorittamisen epäonnistumisen ja talouskriittisessä vastaavasti syntyy merkittäviä taloudellisia ongelmia.

Järjestelmä voidaan luokitella kriittiseksi monesta eri näkökulmasta. Yleisesti voidaan sanoa, että jos järjestelmän pettäminen johtaa menetyksiin, joita ei voida sallia, kutsutaan järjestelmään turvallisuuskriittiseksi [Knight, 2002].

Kriittisiä tietojärjestelmiä tarkasteltaessa ei voida tarkastella vain osaa ohjelmiston koodista. Tarkastelussa tulee ottaa huomioon kokonaisuus, jossa ohjelmisto on käytössä, ja ympäristötekijät, jotka voivat vaikuttaa järjestelmän toimintaan.

### **4.3. Kriittisten tietojärjestelmien ongelmat**

Kriittisten järjestelmien virheellinen toiminta voi johtaa onnettomuuteen, vahinkoon, uhkaan tai sen mahdollisuuteen. Joissain tilanteissa kriittisyyteen riittää mahdollisuus onnettomuuden syntymiseen. Esimerkkinä tällaisesta voidaan pitää esimerkiksi ilmailussa käytettävien turvallisuusminimien alittamista. Yksi ilmailussa käytettävistä minimeistä on porrastusminimi. Tällä tarkoitetaan pienintä etäisyyttä, jolla ilma-alus voi ohittaa toisen ilma-aluksen tai esimerkiksi muiden käytössä olevan ilmatilan osan. Porrastusminimin alittaminen johtaa onnettomuustutkintaan, vaikka niin sanotussa läheltä piti tilanteessa henkilö- tai materiaalivahinkoja ei olisi syntynytäkään.

Onnettomuus tai vahinko on perinteisesti määritelty odottamattomana tapahtumana tai tapahtumasarjana, joka johtaa ei sallittuihin menetyksiin, kuten kuolemiin, loukkaantumisiin, sairastumisiin, vahinkoihin tai laitteiden tai omaisuuden menetyksiin, tai ympäristöllisiin vahinkoihin [Leveson, 1990]. Tietokoneet itse harvoin räjähtävät, syttyvät tuleen tai muutoin aiheuttavat fyysisiä vammoja. Täten tietokoneita sellaisinaan voidaan pitää melko turvallisina. Kuitenkin tietokoneet osana kriittistä kokonaisuutta voivat vaikuttaa merkittävästi onnettomuuden syntymiseen. Koska tietokoneet vaikuttavat epäsuorasti onnettomuuksien syntyyn, ohjelmistojen turvallisuutta tulee tarkastella järjestelmäturvallisuuden näkökulmasta [Leveson, 1990].

#### **4.3.1. Viat sekä vikojen ja vaaran aiheuttajat**

Ongelmat järjestelmissä johtuvat virheistä järjestelmän toiminnassa. Ohjelmistojen kannalta virhe on olemassa käyttöönotosta lähtien. Se voi tulla esille ajan myötä tai se voi esiintyä tietyn järjestelmään annetun syötteen seurauksena. Jos mitään ei tehdä, virhe voi johtaa järjestelmävikaan. Viat järjestelmissä voidaan jakaa kulumisesta tai suunnitteluvirheistä syntyneisiin virheisiin [Knight, 2003].

Kulumisesta aiheutuvat viat voidaan usein ennustaa, ja siten niistä ei saisi syntyä esimerkiksi palvelun saatavuusongelmaa. Kulumista voi esiintyä esimerkiksi tietokoneiden fyysisissä osissa. Niiden vikaantumisten osalta voidaan tutkia ja mitata järjestelmän vikojen sietokykyä.

Suunnitteluvirheet ovat vikoja, jotka ovat olleet järjestelmissä jo niiden kehittämisestä asti. Näin ollen kaikki ohjelmistovirheet ovat tavallaan suunnitteluvirheitä, koska ohjelmistot eivät varsinaisesti voi rappeutua; niihin ei voi tulla kulumia. Tietoja voidaan menettää tai ohjelman suoritus voi keskeytyä esimerkiksi levyrikon tai jonkin muun korruptoitumisen seurauksena. Tällöin syynä on kulumisen tai sovelluksen ulkopuolelta tuleva häiriötekijä.

Jotta vaaratilanteet voitaisiin tehokkaasti estää, tulee tuntea niiden aiheuttajat. Etsittäessä vaaran aiheuttajaa usein ongelman syntymistä pyritään yksinkertaistamaan. Vaikka vaaran syntymiseen löydettäisiinkin useita myötävaikuttajia, saatetaan niistä valita kuitenkin vain yksi ja nimetä se syyksi. Aiheuttajaksi saatetaan virheellisesti valita tapahtumaketjun aloittaja tai viimeinen olosuhdetekijä, joka myötävaikutti ketjun käynnistymiseen. Jotta vaaran aiheuttajat voitaisiin tunnistaa oikein, on syytä tarkastella muutamia päätyyppisiä virheellisistä tavoista määritellä ongelman syy liian yksinkertaisesti. Leveson [1995] mainitsee seuraavat tavat ongelman syiden liialliseen yksinkertaistamiseen:

- lainopillinen lähestymistapa,
- käyttäjän virhe tai tekninen vika ainoana tekijänä,
- organisaatiotekijöiden huomioimatta jättäminen.

Lainopillisella lähestymistavalla tarkoitetaan lakimiesten tai vakuutustarkastajien tapaa etsiä asioille vain yksi syy, jotta vastuiden selvittäminen ja vaatimusten kohdistaminen olisi helpompaa ja paremmin ymmärrettävissä. Lakien, määräysten ja ohjeiden noudattaminen on toiminnan lähtökohta. Todelliset ongelmien syyt voivat kuitenkin jäädä selvittämättä, jos tutkinta loppuu määräyksen rikkojan löytymiseen eikä tutkinnassa selvitetä syitä, miksi määräystä ei noudatettu. Lainopillisten syiden selvittämisestä voi olla hyötyä, mutta teknisesti se ei aina auta todellisten syiden selvittämistä.

Yleinen tapa syiden yksinkertaistamisessa on käyttäjän syyllistäminen. Koska ihmiset toimivat järjestelmien käyttäjinä, on aivan liian helppo sanoa, että syy on operaattorin tekemässä valinnassa tai valinnan tekemättä jättämisessä. Jos tapahtumaketjusta löytyy käyttäjän vikaan myötävaikuttanut toiminto, valitettavan usein syiden tutkiminen pysähtyy siihen, vaikka esimerkiksi olosuhteiden tutkimisesta voisi löytyä avaintekijöitä tulevaisuuden onnettomuuksien välttämiseen.

Toinen yleinen ja merkittävä tapa yksinkertaistamisessa on vain teknisten syiden käsittely. Jos syitä tarkastellaan vain teknisestä näkökulmasta, voi muita tärkeitä seikkoja jäädä huomaamatta. Kapea-alainen syiden tutkinta voi estää tulevaisuudessa vastaavien onnettomuuksien estämisen. Syyt tulee siten ymmärtää osana laajempaa kokonaisuutta, eikä yksittäisen teknisen komponentin roolina.

Organisaatiotason asioita ei tule sivuuttaa syiden etsinnässä. Syyn aiheuttaja voi olla jossain konkreettisessa asiassa, mutta sen mahdollistaneet tekijät ja olosuhteet voidaan yleensä johtaa jollakin tavoin organisaation kulttuuriin, johtamiseen ja rakenteeseen. Esimerkiksi ihmisen tekemiä virhearviointeja voi olla hankala estää puuttumatta näihin myötävaikuttaviin tekijöihin.

Kun ajatellaan yksinkertaistaen kriittisiä järjestelmiä, on olemassa kolmen tyyppisiä järjestelmäkomponentteja, joilla on taipumus aiheuttaa vikoja ja vaaratilanteita. Näitä ovat fyysiset osat järjestelmissä, ohjelmistot ja ihmisten tekemät virheet [Sommerville, 2000]. Jos halutaan parantaa kriittisten järjestelmien turvallisuutta, tulee toiminnassa tarkastella kaikkia näitä osaluokkia. On väärin todeta, että vikoja ovat vain toiminnot vastoin ohjelmiston

määrittelyä. Suurin osa onnettomuuksista, joissa ohjelmistot ovat osatekijöinä, johtuvat virheistä ohjelmiston määrittelyssä [Leveson, 1986].

#### 4.3.2. Luotettavuuden mittaaminen

Ohjelmistojen luotettavuuden mittaaminen on erittäin vaikea tehtävä. Kriittisten järjestelmien osalta on usein kuitenkin välttämätöntä osoittaa, että järjestelmä on luotettava. Osoittaminen on välttämätöntä, koska järjestelmävirheen seurauksia ei voida hyväksyä ja koska asiakkaan tulee vakuuttua, että järjestelmän kaikki luotettavuusosa-alueet (saatavuus, luotettavuus, käyttö- ja järjestelmäturvallisuus) ovat vaaditulla tasolla [Sommerville, 2000, s. 468].

Numeerisia arvioita yritetään löytää tutkimalla yksittäisten komponenttien luotettavuushistoriaa tai vastaavien järjestelmien onnettomuushistoriaa. Vastaavien vertailukelpoisten tietojärjestelmien löytäminen on vaikeaa. Vaikka käytettävä ohjelmisto olisi sama, eroja löytyy todennäköisesti käytettävistä laitteista, ympäristöstä tai käyttäjistä. Onnettomuushistorian tietoja voidaan käyttää hyödyksi mekaanisten osien osalta. Standardiosille tietoa onkin yleensä saatavilla pidemmältä aikaväliltä. Näitä mittareita ei voida luotettavasti hyödyntää ohjelmistojen osalta. Jopa ohjelmistojen uudelleenkäytössä niiden rajapinnat vaihtelevat ja siten täysin samanlaisen ympäristön luominen on erittäin hankalaa. Laajoissakin järjestelmissä luotettavuutta tulee mitata kokonaisjärjestelmän kannalta. Mittaamalla yksittäiset järjestelmän osien luotettavuudet, ei niistä voida johtaa koko järjestelmän luotettavuutta. Ongelmat syntyvät usein ympäristön vaikutuksesta ja ohjelmiston eri osien välisestä kommunikoinnista.

Yksi olennaisimmista kysymyksistä on ratkaista, mikä tapa osoittaa luotettavimmin, mikä on järjestelmän luotettavuuden taso. Muodollisilla tavoilla on yritetty luoda kaavamaiset tavat laskea luotettavuutta. Kriittisten järjestelmien kehittämisessä niitä voidaan hyödyntää kahdella tasolla [Somerville, 2000, s. 469]. Muodollisella tavalla voidaan määritellä järjestelmä ja matemaattisesti analysoida epäjohdonmukaisuuksia ja yhteensopimattomuuksia. Muodollisesti voidaan varmistaa, että ohjelmiston koodi on yhtäpitävä määrittelyn kanssa. Tämä vaatii toimiakseen muodollisen määrittelyn.

Laajojen järjestelmien osalta tämä vaatii paljon aikaa ja olisi siten myös kallista. Ajan lisäksi tarvitaan myös erityisiä työkaluja teorioiden tuottamiseen ja erittäin hyvää matemaattista asiantuntemusta laskemiseen ja analysoimiseen. Vaikka tällä tavoin voidaankin saada luotettavampia ja turvallisempia järjestelmiä, käytännössä tästä ei ole varmuutta. Somervillen [2000, s. 470] mukaan on kolme syytä käytännön epävarmuustekijöistä: spesifikaatio ei

vastaa todellisia käyttäjien vaatimuksia, todistuksissa voi olla virheitä ja todistuksen olettama käyttömalli voi olla väärä. Haittapuolista huolimatta muodollisilla menetelmillä on tärkeä rooli kehitettäessä kriittisiä järjestelmiä.

Luotettavuus järjestelmissä on perinteisesti määritelty toimintona, jossa tarkastellaan todennäköisyyttä, että järjestelmä toimii oikein tietyn ajan. Se on siis aikaväli toimintahäiriöiden välillä [Abbot, 1990]. Ongelmana tässä tavassa on se, että se ei käsittele ongelmien seurauksia lainkaan ja kohtelee kaikkia ongelmia samalla tavalla.

On myös tärkeää tunnistaa ero luotettavuuden ja turvallisuuden välillä järjestelmissä. Järjestelmässä syntyvä virhe voi lopettaa ohjelman suorittamisen, minkä ei sinällään siis tarvitse johtaa menetyksiin. Jos kriittistä valvontaa suorittava laite tekee prosessin uudelleenkäynnistämisen ongelmatilanteessa, ei se välttämättä johda ongelmiin katkoksen jäädessä lyhyeksi. Turvallisin järjestelmä on joskus se, joka ei tee mitään [Leveson, 1986]

Seuraavassa käsitellään tapoja mitata tietojärjestelmien toimintaa. Mitattavia asioita ovat esimerkiksi:

- ongelmien esiintymistiheys,
- ongelmien välisen ajan mittaaminen,
- todennäköisyys suunnitteluvirheen testauksen läpäisemisestä,
- palvelun saatavuus,
- kykyä toimia palvelutilanteessa,
- virheiden määrä suhteessa ohjelmakoodiin ja
- luotettavuuden kasvua korjatussa ohjelmassa.

Analysoitaessa riskejä saatetaan laskea todennäköisyyksiä siitä, kuinka usein vaaratilanteita voi sattua. Laskennassa hyödynnetään malleja tapahtumista, jotka voisivat aiheuttaa vaaran. Käytännössä vain laskentaan mukaan otetut mallit voidaan mitata. Laskennassa käytetään ennakkoon kerättyä dataa ja olosuhteita määriteltäessä joudutaan tekemää oletuksia. Kaikkien yhdistelmien huomioiminen on mahdoton tehtävä. Usein on kuitenkin niin, että vaaran aiheuttaja on jokin ennalta arvaamaton tilanne tai syy. Tällaisten saaminen mukaan laskentaan on miltei mahdotonta. Ei ole mahdollista mitata todennäköisyyttä, että ohjelmisto ei toimisi tietyllä tavalla [Leveson, 1990]. Luotettavuuden lukuarvo voi olla harhaanjohtavaa. Helposti voi syntyä tilanne, että lukuarvoltaan suurimmat riskit käsitellään huolella ja samanaikaisesti pienemmät sivuutetaan, vaikka niiden seuraukset olisivat todennäköisempiä vakavammat.

Yksi tapa mitata numeerista arvoa luotettavuudelle on ajan mittaaminen kahden vian välillä [Parnas et al., 1990]. Tämän lukuarvon käyttökelpoisuus paranee sitä mukaan mitä kauemmin ohjelmisto on ollut käytössä. Vian aiheuttaja, jopa pitkänkin ajan kuluttua, voi olla syötejärjestys, jota ei ole aiemmin esiintynyt. Tällöin arvo kertoo enemmänkin syötteiden todennäköisestä esiintymisestä kuin itse ohjelmiston luotettavuudesta. Uudelta järjestelmältä luotettavan arvon saaminen on vaikeaa. Somervillen mukaan [2000, s.471] vaikeutena ovat tuotannollisen toiminnan mallintaminen testiin, testidatan tuottamisen korkeat kustannukset sekä statistinen epävarmuus silloin, kun vaaditaan erittäin korkeaa luotettavuustasoa.

Kriittisissä järjestelmissä mikään suunnitteluvirhe, joka voi johtaa vahingon syntymiseen, ei ole hyväksyttävissä. Monimutkaisista järjestelmistä, joista meillä ei ole tarpeeksi käyttöön perustuvaa tietoa, emme voi olla varmoja, että niissä ei ole suunnitteluvirheitä. Tällöin tulee arvioida todennäköisyys, jolla vakava suunnitteluvirhe läpäisisi ohjelmiston testauksen [Parnas et al., 1990]. Tämä voi kohottaa ohjelmiston luotettavuusarvoa.

Palvelun saatavuus on yksi mahdollisesti mitattavista arvoista. Saatavuudella tarkoitetaan aikaa, jolloin järjestelmä on käytettävissä ja valmiina toimimaan. Arvoon vaikuttaa omalta osaltaan myös palvelukatkojen kesto. Esimerkiksi kauanko kestää siitä, kun järjestelmä sammutetaan siihen, kun se on taas käytettävissä [Parnas et al., 1990].

Mitattavat arvot tulee suhteuttaa järjestelmien tuottamiin palveluihin. Jos esimerkiksi järjestelmä vastaa toimista poikkeustilanteissa, tulee mitata järjestelmän kykyä toimia niissä oikein.

Kirjallisuudessa käsitellään mittaustapaa, jossa mitataan virheiden määrä suhteessa ohjelmakoodiriveihin. Turvallisuusmielessä tätä ei ole tarkoituksenmukaista mitata, koska objektiivista tapaa laskentaan ei ole. Oletetaan tilanne, jossa on luotu kaksi ohjelmistokomponenttia tekemään samaa tehtävää. Molemmat toimivat väärin samassa tilanteessa, mutta toisen tekemiseen on käytetty enemmän koodirivejä. Jälkimmäinen ei silti ole luotettavampi kuin toinen.

Luotettavuuden kasvua voidaan myös yrittää ennustaa. Jos ohjelmiston luotettavuudesta on kerätty tietoa, voidaan vian korjauksen olettaa poistavan tietyn tyyppiset viat. Kriittisten järjestelmien arviointiin tätä periaatetta ei voida käyttää. Jokaisen muutoksen jälkeen ohjelmaa tulee käsitellä uutena ohjelmana, koska pienillä muutoksilla voi olla suuria seurauksia [Parnas et al., 1990].

Keskusteltaessa luotettavuuden mittaamisesta nousee esille kysymys ohjelmiston testauksesta. Testaamalla ei voi todentaa ohjelman oikeellisuutta [Parnas et al., 1990]. Testauksen tekee ongelmalliseksi ohjelmiston sisäisten

rakenteiden puuttuminen tai monimuotoisuus, ohjelmiston tilojen suuri lukumäärä ja syötekombinaatioiden suuri lukumäärä. Testaamalla löydetään koodausvirheitä, mutta suunnitteluongelmien etsimiseen testaus ei sovellu.

Järjestelmän luotettavuuden mittaaminen ei useinkaan johda suoraan luotettavuuden paranemiseen [Abbot, 1990]. Luotettavuuden mittaamisen ongelmana on riippuvuus ohjelmistolle annetuista syötteistä. Jos virheen aiheuttavaa syötettä käytetään toistuvasti, aiheuttaa se luotettavuuden laskun, vaikka järjestelmä itsessään ei ole muuttunut.

Ohjelmiston ei tarvitse olla virheetön, jotta sitä voidaan pitää luotettavana. Yleisesti ottaen ohjelmistoa, jonka todennäköisyys aiheuttaa ongelmia on erittäin pieni, pidetään luotettavana.

### **4.3.3. Ongelmien luokittelu**

Tarkasteltaessa ongelmia, joita tietojärjestelmien virheellinen toiminta voi aiheuttaa, havaitaan että niiden seuraukset ovat suuruudeltaan erilaisia. Ongelmat voidaan lajitella kategorioihin niiden aiheuttamien ongelmien suuruuden mukaan. Lajittelun perusteella syntyy riskitasot, kun mukaan arviointiin otetaan niiden esiintymisten todennäköisyydet.

Riskitasojen pohjalta voidaan yleisesti määritellä, minkä tasoisiin toimiin tulee ryhtyä, jos riski uhkaa tai se toteutuu. Vastatoimille voidaan edelleen määritellä vasteajat. Vasteajalla tarkoitetaan aikaa, jolloin viimeistään toimiin tulee ryhtyä.

Kaikkiin riskeihin ei voida määritellä korjaavia toimenpiteitä. Toimet riskin aktivoitumisen jälkeen voivat joskus olla riittämättömiä tai niillä ei enää saavuteta riittävää turvatasoa. Jos äärimmäinen riski on esimerkiksi hengen menetys, ei siitä voida enää toipua.

Osa riskeistä voidaan määritellä seuraamuksiltaan niin pieniksi tai epätodennäköisiksi, että ne hyväksytään sellaisenaan. Tällaiset riskit voidaan määritellä hyväksyttäväksi riskeiksi [Leveson, 1990]. Eri järjestelmissä hyväksyttävän riskin raja voi vaihdella suuresti.

## **4.4. Reagointi vaaran aiheuttajiin ja ongelmiin**

### **4.4.1. Vaaratilanteiden syntyminen**

Vaaratilanteet ovat miltei aina syntyneet useista eri syistä. Niiden osuus tapahtumien kulkuun ei useinkaan ole aivan selvää. Vaaratilanteen voidaan ajatella syntyvän, kun joukko tapahtumia sekoitetaan satunnaisella tavalla [Petersen, 1971] tai dynaamisena mekanismina, joka alkaa vahingon aktivoimisella ja kulkee järjestelmän läpi joukkona peräkkäisiä tai

samanaikaisia tapahtumia loogisessa järjestyksessä, kunnes hallinta menetetään ja menetyksiä syntyy (domino-teoria) [Malasky 1982].

Koska yksittäiset tapahtumat yksinään eivät useinkaan aiheuta vaaratilanteita, joudutaan ongelman aiheuttajaa etsittäessä pohtimaan monia tapahtumia sekä niiden välisiä suhteita. Näiden seikkojen selvittämiseksi tarvitaan paljon tietoa järjestelmän toimintalogiikasta sekä vallinneista ympäröivistä ja sisäisistä olosuhteista.

#### **4.4.2. Reagointi vaaraan**

Kun tietojärjestelmän havaitaan aiheuttavan tai myötävaikuttavan uhkaavan vaaratilanteen syntymisen, tulee vaaraan reagoida oikean tasoilla toimilla. Toimet luokitellaan yleisesti seuraavasti [Hammer, 1972]:

- vaaran poistaminen
- vaaran vähentäminen
- vaaran hallitseminen ja
- vahinkojen vähentäminen.

Reagointi vaaraan ei tarkoita vain yhtä yllä mainituista kohdista. Usein toimien tavoitteena on saavuttaa tuloksia muillakin tasoilla [Leveson, 1995]. Ensisijaisesti pyritään vaaran poistamiseen, mutta jos vaaran poistaminen ei ole mahdollista, tulee toimilla pyrkiä vaaran vähentämiseen. Voi olla myös mahdollista, että vaaran vähentäminen ei onnistu. Tällöin vaara pyritään hallitsemaan muilla keinoilla. Tällainen keino voisi olla esimerkiksi vaarasta tiedottaminen, jolloin osapuolet pystyvät ennakoimaan toimiaan vaaratilanteessa. Viimeinen taso on vaarasta aiheutuvien vahinkojen vähentäminen esimerkiksi rajoittamalla viallisen järjestelmän käyttöä.

Riski voidaan määritellä mahdollisuutena vaaran esiintymiselle, mahdollisuutena, että vaara johtaa onnettomuuteen ja onnettomuuteen liittyen pahimman mahdollisen menetyksen määränä [Leveson, 1990]. Riskiä voidaan vähentää vähentämällä yhtä tai useampaa näistä kolmesta riskitekijästä.

Riskien poistaminen ei ole ongelmaton. Riskien poistaminen johtaa usein siihen, että riski vaihtuu toiseksi, eikä siten poistu [Malasky, 1982]. Jos järjestelmää on muutettu riskien poistamiseksi tai vähentämiseksi, tulee riskitekijät arvioida riittävän kattavasti uudestaan muutoksen jälkeen.

Ohjelmistojen omat turvallisuusmääritykset ovat keskeisessä roolissa, kun uhkaavaan vaaraan reagoidaan. Ohjelmistolta voidaan vaatia estoa menemästä uhkaavaan tilaan, uhkaavan tilan huomaamista tai siirtymistä vaarallisesta tilasta turvalliseen [Lutz, 2000].

#### 4.4.3. Vaaratekijöiden poistaminen

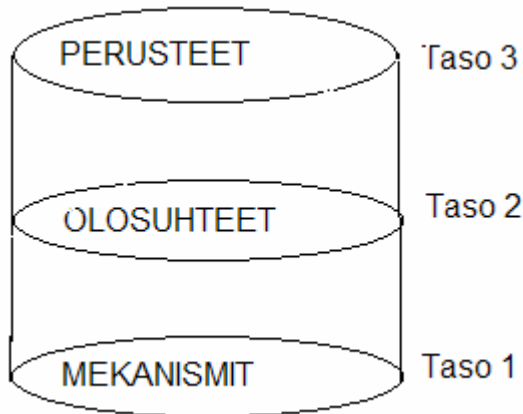
Monesti todetaan, että meidän tulee oppia virheistä. Jos ongelmat opettavat meitä menestystä enemmän, miten taataan turvallisuus sellaisissa kokonaisuuksissa, joissa yksikin vika aiheuttaa liian suuren menetyksen. Toisaalta on helppo myöntää, että kaikkia riskejä ei voida kokonaan poistaa. Tällöin tuleekin keskittyä myös riskeistä aiheutuvien vahinkojen vähentämiseen.

Kriittisten järjestelmien monimutkaisuus johtaa siihen, että vaara syntyy tavallisesti tapahtumaketjujen seurauksena. Riskien eliminoimisen kannalta tapahtumaketjut antavat hyvät edellytykset, koska tällöin voi olla monta kohtaa, jossa ei-toivottu tapahtumaketju voidaan havaita ja pysäyttää [Johnson, 1973]. Ketjuissa on myös usein kohta, josta ei enää ole paluuta. Tapahtumat seuraavat tämän jälkeen toisiaan ja johtavat lopulta vahingon syntymiseen.

On selvää, että harvoin onnettomuuksiin on vain yksi syy. Tapahtumia tulee tarkastella laajemmin kuin vain yhden tapahtumaketjun valossa. Lewycky [1987] esittää kolmitasoisien mallien onnettomuuksien syiden ymmärtämiseksi (Kuva 2, Hierarkkinen onnettomuusmalli). Alimmalla tasolla (taso 1) kuvataan tapahtumaketjut. Esimerkiksi ruuvien katkeamista seuraa pyörän irtoaminen ja irtoamista seuraa ohjauksen menettäminen.

Keskimmäisellä tasolla (taso 2) kuvataan vallinneiden olosuhteiden syy-yhteydet tapahtuman syntymiseen. Esimerkiksi suojuksen puuttuminen mahdollistaa kosteuden pääsyn ruuviin ja siten kosteuden pääsy ruuviin on mahdollista.

Ylimmällä tasolla (taso 3) kuvataan perusteet, jotka mahdollistavat tai sallivat tasolla 2 kuvatut olosuhteet. Taso sisältää perusteita teknisistä ja fyysisistä olosuhteista, sosiaalisista vuorovaikutteista ja ihmisten toimista, johtamisjärjestelmistä ja organisaatiotason kulttuurista sekä hallinnollisista ja sosiaalipoliittisista periaatteista. Tämän tason sanotaan sisältävän niin sanotut juurisyyt onnettomuuteen. Syiden poistamisessa syyllistytään usein alempien tasojen ongelmien poistoon puuttumatta juurisyihin. Koska onnettomuudet tapahtuvat harvoin täsmälleen samalla tavalla, juurisyiden merkitys korostuu. Jotta onnettomuuksien riskejä voidaan merkittävästi vähentää, tulee juurisyyt tunnistaa ja poistaa [Leveson, 1995].



Kuva 2, Hierarkkinen onnettomuusmalli, [Lewycky, 1987]

Yksinkertaisimmillaan ongelma voi olla ohjelmistovirhe. Ohjelmistovirheet ovat suunnitteluvirheitä, jotka voi välttää huolellisella suunnitteluprosessilla, poistamalla ne kehitysvaiheessa, korjaamalla suunnitelmat vika-analyysillä tai sallimalla ne käytössä Knight [2003]. Vaikka vikasietoisuuden tutkimukseen on kehitetty uusia tekniikoita estämään fyysisten komponenttien vikoja, käytännön keinoja suunnitteluvirheiden löytämiseksi ei ole löytynyt [Butler & Finelli, 1991].

#### 4.5. Kriittisten tietojärjestelmien tietoturvallisuus

Käytettäessä tietojärjestelmää kriittisessä ympäristössä korostuu tietoturvallisuuden vaikuttavien tekijöiden huomioiminen merkittävästi. Tietoturvallisuudella tarkoitetaan [Hallinnon kehittäminen, tietoturvasanasto]:

1. asiaintilaa, jossa tietojen, tietojärjestelmien ja tietoliikenteen luottamuksellisuuteen, eheyteen ja käytettävyyteen kohdistuvat uhat eivät aiheuta merkittävää riskiä ja
2. keinojen ja toimenpiteiden kokonaisuutta, joiden avulla pyritään varmistamaan tietoturvallisuus niin normaali- kuin poikkeusoloissa.

Tietoturvallisuuden toteuttamisessa erotetaan kahdeksan toimenpidealuetta [Hallinnon kehittäminen, tietoturvasanasto]: hallinnollinen, henkilöstö-, fyysinen, tietoliikenne-, laitteisto-, ohjelmisto-, tietoineisto- ja käyttöturvallisuus. Seuraavassa tarkastellaan näiden huomioimista kriittisissä tietojärjestelmissä.

#### 4.5.1. Tekninen turvallisuus

Tekninen turvallisuus toimenpidealueina käsittää laitteisto-, ohjelmisto- ja tietoliikenneturvallisuuden. Yleisesti nämä osa-alueet ovat hyvin tunnistettuja, mutta järjestelmien käyttö kriittisissä ympäristöissä vaatii näiden osa-alueiden täsmällistä tunnistamista ja tarvittavista toimista huolehtimista.

Laitteistoturvallisuus on tietoturvallisuuden alue, joka käsittää tietojenkäsittely- ja tietoliikennelaitteiden käytettävyyden, toiminnan, kokoonpanon, kunnossapidon ja laadunvarmistuksen [Hallinnon kehittäminen, tietoturvasanasto]. Kriittisissä tietojärjestelmissä yleensä käytettävyydelle asetetaan kovat vaatimukset. Laitteiden tulee toimia koko ajan, niiden huollot ja tarkastukset tulee olla ennalta suunniteltuja ja niiden toteutumista tulee valvoa.

Ohjelmistoturvallisuus on tietoturvallisuuden osa-alue, joka käsittää käyttöjärjestelmät, väliohjelmistot, sovellusohjelmat ja tietoliikenneohjelmistot [Hallinnon kehittäminen, tietoturvasanasto]. Alueeseen kuuluvat ohjelmistojen tunnistamis-, eristämisen-, pääsynvalvonta- ja varmistusmenettelyt, tarkkailu- ja paljastustoimet, lokimenettelyt, ohjelmistojen laadunvarmistus sekä niiden ylläpitoon ja päivitykseen liittyvät turvallisuustoimet. Kriittisissä tietojärjestelmissä ohjelmistot tulee olla yksilöityinä (mm. ohjelmistoversioiden hallinta). Ohjelmistot tulee eristää sekä toimintaympäristössään että käyttäjien suunnalta siten, että vain välttämätön pääsy sallitaan. Toiminta tulee varmistaa erilaisilla varamenetelmillä. Erityisen tärkeää kriittisten palveluiden osalta on niiden oikeasta toiminnasta varmistuminen. Se voidaan hoitaa valvontamekanismeilla, seuraamalla lokitiedostoja ja huolehtimalla ohjelmiston laatuun vaikuttavista seikoista. Ylläpidosta ja päivityksistä huolehtiminen on tärkeä osa turvallisuuden varmistamista. Vaikka esimerkiksi uusien käyttöjärjestelmäversioiden luvataan tuovan parempaa turvaa ja vakautta, ei toimittajien lupauksiin voida kuitenkaan täysin luottaa. Uudet ohjelmakomponentit vaativat aina testausta ja siksi on aina syytä varautua palaamaan vanhojen komponenttien käyttöön, jos ylipääsemättömiä ongelmia ilmenee.

Tietoliikenneturvallisuus on tietoturvallisuuden osa-alue, johon kuuluvat mm. tietoliikennelaitteiston kokoonpano, sen luettelointi, ylläpito ja muutosten valvonta, ongelmatilanteiden kirjaus, käytön valvonta, verkon hallinta, pääsyn valvonta, viestinnän salaaminen ja varmistaminen, tietoturvallisuuden kannalta merkityksellisten tapahtumien tarkkailu, kirjaus ja selvittäminen sekä tietoliikenneohjelmien testaus ja hyväksyminen [Hallinnon kehittäminen, tietoturvasanasto]. Kriittisissä tietojärjestelmissä tulee suojautua hyvin ulkoisia uhkatekijöitä vastaan. Tietoliikenteen turvaaminen on tässä tärkeässä roolissa. Ei riitä, että rakennetaan turvalliset yhteydet ja taataan tiedon eheys ja sen

saatavuus vain sitä tarvitseville. Toiminnan turvaaminen edellyttää lisäksi yhteyksien jatkuvaa tarkkailua ja vaaditun turvataso- testauksia.

#### **4.5.2. Fyysinen turvallisuus**

Fyysinen turvallisuus tarkoittaa henkilöiden, laitteiden, aineistojen, varastojen ja toimitilojen turvallisuutta tuhoja ja vahinkoja vastaan [Hallinnon kehittäminen, tietoturvasuunnitelma]. Fyysinen turvallisuus sisältää mm. kulunvalvonnan, teknisen valvonnan ja vartiointin, palo-, vesi-, sähkö-, ilmastointi- ja murtovahinkojen torjunnan sekä kuriirien ja tietoaineistoja sisältävien lähetysten turvallisuuden.

Uhka voi tulla monelta suunnalta. Vahingon aiheuttaja voi olla ihminen, tekninen vika tai luonnonilmiö. Kriittisten järjestelmien käytön yhteydessä tilojen lukitukset ovat yleensä itsestäänselvyyksiä, mutta tämän rinnalle tarvitaan riittävän tarkka seuranta henkilöiden kulkemisista. Siitä voidaan huolehtia esimerkiksi kulkemiset rekisteröivillä avainkortteilla ja kuvallisilla henkilökortteilla. Näilläkään ei saavuteta riittävää turvatasoa, mikäli niiden käytössä sallitaan poikkeuksia tai käyttöä ei riittävästi valvota.

Teknisiä vikoja ja luonnonilmiöitä vastaan kriittisissä järjestelmissä tulee suojautua hyvin. Tiloissa tulee olla sellaiset palontorjuntalaitteet, jotka eivät palon torjunnan lisäksi tuhoa laitteistoja tai vaaranna tiloissa työskenteleviä ihmisiä. Varalaitteita ei koskaan tule sijoittaa samoihin tiloihin varsinaisten laitteiden kanssa. Sama koskee luonnollisesti myös tietojen ja ohjelmistojen varmuuskopioita. Paloturvallisuuden lisäksi tulee huolehtia riittävästä ilmastoinnista, jotta laitteet eivät mene rikki ylikuumentumisen vuoksi. Sähkön saanti tulee myös turvata akuilla sähkökatkosten varalta.

Kaikkia fyysisiä uhkatekijöitä vastaan ei aina ole mahdollista suojautua. Tällöin tulee olla olemassa suunnitelma toiminnan alasajosta ja kriittisten palveluiden turvaamisesta alasajon aikana.

#### **4.5.3. Toiminnallinen turvallisuus**

Toiminnallinen turvallisuus toimenpidealueena käsittää hallinnollisen tietoturvallisuuden, henkilöstöturvallisuuden, tietoaineistoturvallisuuden ja käyttöturvallisuuden. Nämä osa-alueet luovat perustan turvallisuudelle toiminnalle. Lakien ja normien noudattaminen on lähtökohta turvallisuuden saavuttamisessa.

Hallinnollinen tietoturvallisuus on tietoturvallisuuden osa-alue, joka käsittää toimintalinjaukset, periaatteet, organisaatiojärjestelyt, henkilöstön tehtävien ja vastuiden määrittelyn sekä tietoturvallisuuteen tähtäävän ohjeistuksen, koulutuksen ja valvonnan [Hallinnon kehittäminen, tietoturvasuunnitelma]. Kriittisissä tietojärjestelmissä hallinnollinen tietoturvallisuus

luo eräänlaisen pohjan tietoturvaliselle toiminnalle. Laatumalla turvallisuukseskeiset toimintaperiaatteet ja huolehtimalla henkilöstön oikeasta toiminnasta koulutuksella, selkeillä vastuualueilla, ohjeistuksella ja valvonnalla voidaan jo sinällään estää monia turvallisuuatta uhkaavia toimia.

Henkilöstöturvallisuus on henkilöstöön liittyvien riskien hallintaa henkilöstön soveltuvuuden, toimenkuvien, sijaisuuksien, tiedonsaanti- ja käyttöoikeuksien, suojaamisen, turvallisuuksoulutuksen ja valvonnan osalta [Hallinnon kehittäminen, tietoturvasanasto]. Työskenneltäessä kriittisten tietojärjestelmien parissa niin organisaation toiminnan kuin yksittäisten työntekijöidenkin kannalta henkilöstön toimintaan liittyvät riskit tulee hallita kattavasti. Muistettava on, että samat koulutus-, valvonta- ja soveltuvuusvaatimukset koskevat aivan samalla tavalla tilapäistä työvoimaa kuin ne koskevat vakituksia työntekijöitä.

Tietoaineistoturvallisuus on tietoturvalisuuden osa-alue, joka käsittää asiakirjojen, tiedostojen ja muiden tietoaineistojen käytettävyyden, eheyden ja luottamuksellisuuden, keinoina mm. tietoaineistojen luettelointi ja luokitus sekä tietovälineiden asianmukainen hallinta, käsittely, säilytys ja hävittäminen [Hallinnon kehittäminen, tietoturvasanasto]. Tietoaineistojen käsittely on monessa mielessä eräänlaista tasapainoilua eri suunnilta tulevien määräysten ja toimintaperiaatteiden välillä. Vaikka oman toiminnan kannalta olisikin järkevää säilyttää tietoja mahdollisimman pitkään, lain mukaan niiden hävittämistä voidaan edellyttää tietyn ajan kuluttua. Esimerkiksi henkilöstölaki määrittelee millaisia toimia edellytetään organisaatioilta, jotka ylläpitävät henkilötietoja sisältäviä rekistereitä.

Käyttöturvallisuus on tietotekniikan käyttöön, käyttöympäristöön, tietojenkäsittelyyn ja sen jatkuvuuteen sekä tuki-, ylläpito-, kehittämis- ja huoltotoimintoihin liittyvä tietoturvalisuuden osa-alue [Hallinnon kehittäminen, tietoturvasanasto]. Kriittisten järjestelmien turvaamisessa tukitoimet hoidetaan yleensä ympärivuorokautisina päivystyksinä tai jatkuvasti toimintaa seuraavilla valvonnoilla. Jos järjestelmillä ei ole henkilöstöä seuraamassa niiden toimintakuntoa, tulee tarkoin harkita, kuinka nopeasti heidän pitää ehtiä paikalle tarkastamaan tilanne ja suorittamaan alustavat korjaustoimet.

#### **4.5.4. Ongelmat kriittisten tietojärjestelmien turvaamisessa**

Lähtökohta kriittisten järjestelmien turvaamisessa on se, että niiden toiminta turvataan mahdollisimman hyvin ulkoisia hyökkäyksiä vastaan. Uhkia vastaan voidaan suojautua eristäytymällä tai estämällä hyökkäykset palomuurien ja käyttäjätunnisteiden avulla. Verkkolaitteista, palomuuureista, ohjelmistoista tai käyttöjärjestelmistä paljastuu tietoturva-aukkoja miltei päivittäin. Turva-aukon

paikkaavat ohjelmistopäivitykset julkaistaan yleensä melko nopeasti ja turvallisuuden varmistamiseksi päivitykset tulisi asentaa mahdollisimman pian.

Kriittisissä järjestelmissä ohjelmistoihin usein tehtävät päivitykset eivät aina ole mahdollisia. Päivityksen asentaminen vaatii usein palvelukatkoa ja jopa järjestelmän uudelleen käynnistämistä. Kriittisissä järjestelmissä päivitysten asennukset tulee suunnitella hyvin. Tämä edellyttää korjauksen testausta ja varmistumista siitä, että päivitys ei aiheuta ongelmia varsinaisen palvelun tuottavan ohjelmiston toimintaan. Tästä syystä tulee aina varautua myös päivityksen poistamiseen, mikäli se aiheuttaa yhteensopivuusongelmia. Kriittisissä järjestelmissä päivityksessä on aina kyse muutoksesta, jonka asentaminen saattaa vaatia jopa lupamenettelyä.

#### **4.6. Turvallisuuden varmistaminen**

Turvallisuuden varmistamisessa ja luotettavuuden varmistamisessa on kyse melko lailla eri prosessista [Somerville, 2000, s. 476]. On mahdollista määritellä luotettavuusvaatimukset ja mittaustavat valmiin järjestelmän luotettavuuden mittaamiselle. Turvallisuutta ei voi tarkoituksenmukaisesti määritellä lukumääräisesti mitattavalla tavalla. Asiantuntijaperusteisesti se voi olla jotain hyvin korkean ja erittäin matalan väliltä.

Järjestelmissä, joilta vaaditaan erittäin pientä vikatiheyttä, luotettavuuden todentaminen perinteisin menetelmin on hyvin vaikeaa. Käytännössä riittävän tilastollisen aineiston kerääminen on usein mahdotonta otoksen jäädessä liian pieneksi, jotta sen perusteella voi tehdä luotettavia analyysejä.

Vaikka menetelmät ohjelmistojen oikeellisuuden todistamiseksi ovat olleet käytössä jo yli 25 vuotta, niitä käytetään hyvin vähän muualla kuin tutkimuslaboratorioissa. Oikeellisuuden todistaminen on hankalaa ja aikaa vievää. Joidenkin kriittisten ohjelmistojen osalta todistamista on käytetty menestyksekkäästi parantamaan järjestelmän turvallisuuteen vaikuttavia osia. Jotta järjestelmää voidaan käyttää kriittisessä tehtävässä, tulee vastuullisten tahojen olla varmoja sen oikeasta toiminnasta. Todistamisella voidaan lisätä luottamusta järjestelmän oikeasta toiminnasta ja siten mahdollistaa järjestelmän käyttö kriittisessä tehtävässä.

Ohjelmistolle voidaan määritellä turvallisuusperustelut osoittamaan, että ohjelmisto täyttää sille asetetut turvallisuusvaatimukset. Perustelujen tarkoitus on osoittaa, että ohjelmiston suoritus ei voi johtaa turvattomaan tilaan. Perusteluilla ei oteta kantaa, miten se ylipäätään vastaa ohjelmiston määrittelyä.

Tehokkain tapa osoittamiseen on kumoamisen tekniikka [Somerville, 2000, s. 477]. Pohjaksi otetaan turvallisuusanalyysien avulla saadut turvattomat

tilat, jotka voivat syntyä ohjelman ajon aikana. Tämän jälkeen ohjelman koodista analysoidaan kaikki ohjelmistopolut, joiden seurauksina turvaton tila voi syntyä. Edeltävät ohjelmiston tilat käsitellään siten, että niistä osoitetaan polun turvattomaan tilaan olevan mahdoton. Tämä tapa on huomattavasti nopeampi ja kustannustehokkaampi turvallisuusvaatimusten osoittamisessa kuin normaali osoittaminen, jossa tulee näyttää toteen, että ohjelmisto täyttää kaikki sille asetetut vaatimukset.

On tärkeää, että ohjelmistojen turvallisuusnäkökohdista huolehditaan sekä ohjelmiston kehitysvaiheissa että sen käytön aikana. Kehittämisen aikana tulee suorittaa tarkastuksia, jotka kohdistuvat käytettyihin tekniikoihin, dokumentointiin, algoritmeihin, laatuvaatimusten toteutumiseen jne. Ohjelmiston käytönaikaista tarkastusta varten on tarpeen suunnitella mekanismit, joilla ohjelmiston toimivuutta voidaan tarkkailla sisältä.

Kriittisen järjestelmän käyttöympäristöstä tulee myös varmistua. Turvallisuusnäkökohdat tulee ottaa huomioon jo määriteltäessä ohjelmistoja. Vaatimusten todentamiseen Sommerville [2000, s. 483] luettelee neljä lähestymistapaa: tarkastetaan järjestelmä tunnettuja hyökkäystapoja vastaan, käytetään turvallisuuden testaamiseen tarkoitettuja työkaluja (esim. salasanojen tarkastus), murtamiseen erikoistuneen tiimin käyttö ja muodollinen tarkastaminen turvallisuusvaatimuksia vastaan. Tarkastamisen tulee olla jatkuva prosessi, sillä sen tulee ottaa huomioon myös käytön aikana mahdollisesti tapahtuvat ympäristön muutokset.

## 5. Muutoksen hallinta

Muutoksen hallinta ymmärretään eri yhteyksissä eri tavoilla. Tietojenkäsittelyssä se usein liitetään ohjelmistojen komponenttien, konfiguraatioiden ja toimintatapojen hallintaan, mutta tällöin kyse on tuotteenhallinnasta (configuration management), joka on osa laajempaa muutoksen hallintaa. Tuotteenhallintaan kuuluu kolme osakokonaisuutta: 1) menetelmät, joilla saman komponentin eri versiot hallitaan, 2) menetelmät, joilla konfiguraatioita ja niiden versioita hallitaan sekä 3) versioita ja konfiguraatioita luotaessa ja muutettaessa noudatettavat toimintatavat [Haikala ja Märajärvi, 2002]. Muutoksen hallinnalla tarkoitetaan tässä tutkielmassa laajempaa kokonaisuutta eli prosessien, rakenteiden, tekniikan, henkilöstön ja kulttuurin muutosten hallintaa organisaation sisällä [Chaffey et. al., 2005].

Uusien tietojärjestelmäratkaisujen käyttöönotto vaikuttaa merkittävästi sekä yksittäisiin työntekijöihin että tiimien välisiin vuorovaikutussuhteisiin. Mikäli henkilöstön vaikutusta ei tunnisteta ja siihen reagoida, muutos aiheuttaa henkilöstössä vastustavan reaktion, joka vaikeuttaa merkittävästi järjestelmän käyttöönottoa.

Tietojärjestelmien kehittämiseen tai hankkimiseen voi olla monia syitä. Lähtökohdiana voi olla ulkopuolelta tulevat paineet tai organisaation sisäinen halu tehostaa toimintaa. Hallitun muutoksen tulee nojata yrityksen strategiaan. Strategian mukaisesti uudistuksen myötä voidaan vastata kilpailijoiden palveluiden kehittymiseen, muuttuneisiin määräyksiin, tuottavuuden lisäämiseen, vanhentuneen teknologian korvaamiseen tai toimintakustannusten laskemiseen. Doherty et. al.[1999] kuvaa tietojärjestelmien kehitysstrategian seuraavasti: prosessi, jossa tunnistetaan niiden kehitettävien tietokonepohjaisten sovellusten joukko, joka tulisi toteuttaa, joka tukee suuresti yrityksen strategiaa ja jolla on kyky luoda etu kilpailijoihin.

Kun uusi tietojärjestelmä otetaan käyttöön, työntekijöiden tulee oppia, kuinka järjestelmää käytetään. Tätäkin merkittävämpää työntekijöille on oppia uudet työskentelytavat. Uusien työskentelytapojen merkitys kasvaa sitä mukaan, mitä suuremmasta järjestelmä uudistuksesta on kyse. Esimerkki tällaisesta voisi olla kasvotusten tapahtuvan asiakaspalvelun muuttuminen Internetin välityksellä saapuvien lomakkeiden käsittelyksi.

### 5.1. Muutostyypit

Järjestelmämuutokset voidaan jakaa karkeasti kahteen päätyyppiin: jatkuviin ja ei jatkuviin. Jatkuvilla muutoksilla tarkoitetaan pieniä, peräkkäin tehtäviä

päivitysmuutoksia. Ei-jatkuvilla tarkoitetaan kerralla tehtävää suurempaa ei-jatkuvaa muutosta.

Jatkuvilla muutoksilla pyritään parantamaan yksittäisiä kohtia toimintalogiikassa. Muutos voi kohdistua toimintaa hidastavaan solmukohtaan, uusien lakien huomioonottamiseen, reagointia asiakkaan puolella tapahtuviin pidempiaikaisiin muutoksiin tai uuden tuotevariaation tuotannon aloittamiseen.

Ei-jatkuvilla isoilla muutoksilla reagoidaan toimintaympäristössä tapahtuviin isoihin muutoksiin. Nämä muutokset johtuvat liiketoiminnan perusteiden muuttumisesta.

## **5.2. Organisaatiotason muutoskohteet**

Muutokset vaativat yleensä muutostyypistä huolimatta muutoksia organisaation toimintaan. McKinsey kuvaa mallissaan neljä pehmeää ja kolme kovaa s-kohdetta, joihin muutokset kohdistuvat [Waterman et. al., 1980]. Kovia kohteita ovat strategiat, rakenteet ja järjestelmät. Pehmeitä kohteita ovat taidot, jaetut arvot, henkilöstö ja tyyli. Tietojärjestelmämuutokset vaativat yleensä myös muutoksia joihinkin näistä organisaatiotason kohteista.

Uusissa järjestelmissä, joissa muutokset ovat jatkuvia, taidot, järjestelmät ja henkilöstö tarvitsevat muutoksia. Ei-jatkuvien suurempien muutosten osalta muutoksia saattaa tarvita missä tahansa näistä kohteista.

## **5.3. Muutos ohjelmistotuotteen kannalta**

Ohjelmistotuotteen kannalta muutos voi kohdistua ohjelmiston koodin lisäksi laitteistoon, käytettäviin asetuksiin, käyttömenetelmiin, ohjeisiin, käsiteltävään tietoon tai käyttöympäristöön. Muutoksen kohdistuessa yhteenkin edellä mainituista kohteista tulee muutosta näkemykseni mukaan käsitellä muutoksen hallinnan näkökulmasta.

Ohjelmistotuote rakentuu suuresta määrästä erilaisia komponentteja. Niitä ovat esimerkiksi lähdekieliset ohjelmakomponentit, käännetyt ongelmakomponentit, määrittelydokumentit, testaussuunnitelmat ja ohjeet. Komponenteista kootaan suurempia kokonaisuuksia, joita kutsutaan konfiguraatioiksi. Ne kehittyvät ohjelmiston elinkaaren aikana, kun virheitä korjataan ja ominaisuuksia lisätään. Tuotteen pohjalta voidaan edelleen kehittää muita tuotteita. Tällöin tulee olla luotuna menetelmät, joilla tuotteen eri haaroja, uusia ja vanhoja versioita, hallitaan. Kokonaisuutena tästä huolehtii tuotteenhallinta.

Tuotteenhallinnan yhtenä tarkoituksena on estää tilanteita, joissa koordinaation puutteesta johtuen kehitetty ja testattu piirre katoaa, aiemmin

korjattu virhe ilmestyy uudelleen tai ohjelmistoon ilmestyy ei-toivottu ominaisuus.

#### 5.4. Muutosten luokittelu

Nadler et al. [1995] esittelevät tavan organisaation muutosten luokitteluun (Taulukko 1, Muutosten luokittelu). Mallissa luokitellaan muutokset jatkuviin ja ei-jatkuviin, sekä ennakoiviin ja reagoiviin muutoksiin. Ennakoivilla muutoksilla tarkoitetaan muutoksia, joissa pyritään parantamaan kilpailukykyä tai tehokkuutta. Reagoivat muutokset johtuvat yleensä muutoksista toimintaympäristössä.

Yhdistämällä ja tarkastelemalla näitä samanaikaisesti löydetään neljä eri muutosten luokittelutyyppiä. Näitä ovat säätäminen, sovittaminen, uudelleenjärjestely ja uudelleenluominen.

	Jatkuva muutos	Ei-jatkuva muutos
Ennakoiva	Säätäminen	Uudelleenjärjestely
Reagoiva	Sovittaminen	Uudenluominen

Taulukko 1, Muutosten luokittelu, Nadler et al. [1995].

Säätämällä tarkoitetaan jatkuvaa muutosta, jossa välitöntä tarvetta muutokseen ei ole. Muutostyyppillä pyritään tehostamaan toimintaa pienillä muutoksilla tai alentamaan tuotantokustannuksia. Näitä muutoksia voidaan tehdä osissa ja ajankohdallisesti silloin, kun se organisaation toimintaan parhaiten sopii.

Sovittaminen muutostyyppinä on jatkuvaa ja sillä reagoidaan ulkoiseen uhkaan tai sen mahdollisuuteen. Muutos voi johtua esimerkiksi kilpailijan uudesta tuotteesta, joka sinällään ei edellytä muutosta koko toimintaympäristöön.

Uudelleenjärjestelyssä kyse on merkittävästä muutoksesta organisaatiolle. Muutos toteutetaan kerralla, mutta sinällään pakkoa muutoksen välittömään tekemiseen ei ole. Muutoksella ennakoidaan tulevaa muutostarvetta tai siirrytään uuteen palvelumuotoon.

Uudelleenluomisella tarkoitetaan muutostapaa, jossa muutos toteutetaan kerralla ja siihen on välitön tarve. Muutos voi saada alkunsa toimintaympäristön äkillisestä muutoksesta, esimerkiksi lainsäädännön muuttumisesta tai uuden kilpailijan toimista.

## 5.5. Muutokset liiketoiminnan prosessien näkökulmasta

Yritysten toiminnan kannalta on tavanomaisempaa, että muutoksilla yritetään kohentaa tuottavuutta ennemmin kuin että yrityksen toimintaympäristö muuttuisi ja sitä kautta olisi pakotettu muuttamaan prosesseja. Parannuksia tavoiteltaessa tulee liiketoimintaprosessien vaiheet selvittää yksityiskohtaisesti. Liiketoimintaprosessien johtamisessa (business process management, BPM) kehitetään vaiheittain yrityksen liiketoimintaprosesseja.

Seuraavassa käsitellään kolme tapaa, joilla yrityksen toimintaprosesseja voidaan muuttaa [Chaffey et. al., 2005, s.387]. Näitä ovat liiketoimintaprosessien uudelleenjärjestäminen, kehittäminen ja automatisointi (Taulukko 2, liiketoimintaprosessien muuttaminen, [Chaffey et. al., 2005, s.388] ).

Nimitys	Käsittää	Tarkoitus (Tehokkuus)	Epäonnistumisen vaara
Liiketoimintaprosessien uudelleenjärjestäminen	Organisaation kaikkien prosessien uudelleenjärjestelyt	yli 100%	Suurin
Liiketoimintaprosessien parantaminen	Avainprosessien uudelleenjärjestelyt	alle 50 %	Keskinkertainen
Liiketoimintaprosessien automatisointi	Olemassa olevien prosessien automatisointi	alle 20 %	Pienin

Taulukko 2, liiketoimintaprosessien muuttaminen, [Chaffey et. al., 2005, s.388]

### 5.5.1. Liiketoimintaprosessien uudelleenjärjestäminen

Jos yrityksessä aloitetaan liiketoimintaprosessien uudelleenjärjestäminen (business process re-engineering, BPR), on se yrityksen kannalta todella mittava operaatio. Uudelleenjärjestely on perusteellista uudelleen ajattelua ja radikaalia liiketoimintaprosessien uudelleen suunnittelua tavoitteena saavuttaa dramaattisia parannuksia kriittisissä tuonaikaisissa tehokkuusarvoissa, kuten kustannukset, laatu, palvelu ja nopeus [Hammer et al, 1993].

Tehtävillä muutoksilla yritetään yleensä omasta tahdosta parantaa yrityksen tuottavuutta. Siten kyseessä on ennakoiva, mutta ei-jatkuva muutos.

### 5.5.2. Liiketoimintaprosessien parantaminen

Vähemmän radikaalia vaiheittaista prosessien muuttamista kutsutaan liiketoimintaprosessien parantamiseksi (business process improvement, BPI). Yrityksissä saatetaan miettiä, tulisiko muutosta tehtäessä uusia koko palveluketju, vai kannattaisiko toimintaa tehostaa uusimalla ketjusta vain kriittisimmät, toiminnan kannalta tehottomimmat tai hidastavat kohdat. Modernin ajattelumallin mukaan yksittäisten osien uusiminen on suositeltavampaa, vaikkakin tällöin ei ole mahdollista saavuttaa niin suuria tehon lisäyksiä kuin uudelleen järjestelyjen kautta olisi teoriassa mahdollista saavuttaa.

### 5.5.3. Liiketoimintaprosessien automatisointi

Jos toimintaprosessin kulkuun ei uskalleta puuttua tai jostain muusta syystä prosessien toimintamallit halutaan säilyttää, jää muutoksen kohteeksi automatisoida manuaalisesti suoritettavat prosessit. Teknologiasta yritetään ottaa hyödyt irti, mutta usein ilman prosessien kehittämistä toiminnalliset hyödyt voivat jäädä pieniksi ja muutoksiin tehdyt investoinnit ei ole tuottavia.

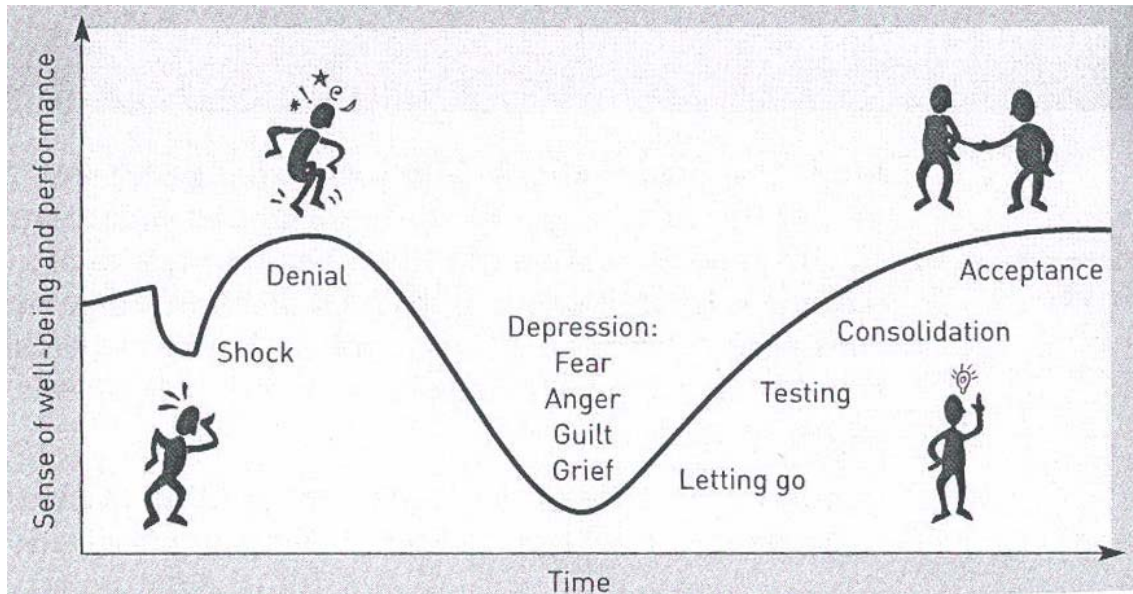
## 5.6. Muutokset yksilön näkökulmasta

Tietojärjestelmien muutokset vaikuttavat organisaation henkilöstön työtehtäviin, työympäristöön ja, kuten useat tutkimukset osoittavat, henkilöstön mielialoihin ja suhtautumiseen muutoksiin. Kriittisten järjestelmien ongelmien syyt voivat johtua olosuhteista, jotka mahdollistavat kriittisen tapahtumaketjun, tai organisaatiotason vaikutuksesta, jotka mahdollistavat olosuhteet myötävaikuttaen haitallisten olosuhteiden syntymiseen.

Muutoksista johtuvilta tunnereaktioilta ei varmastikaan voida kokonaan välttyä, mutta jotta niihin voitaisiin vaikuttaa, vaiheet on syytä tunnistaa muutosprosessin aikana. Adams et al. (1976) on tunnistanut seuraavat tunnetilojen vaihtelut (kuva 3, Henkilöstön reaktiot muutoksen aikana):

1. Tiedostaminen/shokki. Henkilöstö saa harvoin etukäteen tietoa muutossuunnitelmista. Tiedostamista seuraa hyvin nopeasti järkytys, jonka suuruus on riippuvainen siitä, kuinka hyvin yksilötasolla on valmistauduttu muutokseen.
2. Kieltäminen. Tähän sisältyy kieltämistä, negatiivisten seurausten korostamista sekä huomion kääntämistä muihin asioihin. Muutoksen vastustaminen on suurinta tässä vaiheessa.
3. Masennus. Masennus tulee, kun huomataan, että muutos on väistämätön. Tila ei välttämättä ole passiivinen, sillä se voi johtaa vihastumiseen ja muutoksen tarpeen kiistämiseen.

4. Periksi antaminen. Yksilö tuntee tarvetta siirtyä eteenpäin ja alkaa eriytyä edellisestä tilanteesta.
5. Lujittuminen. Yksilö alkaa kerätä positiivisia ajatuksia ja minimoi muutoksen negatiivisia puolia.
6. Hyväksyminen. Lopullinen hyväksyntä tulee, kun uusi tilanne nähdään normaalina.



Kuva 3, Henkilöstön reaktiot muutoksen aikana, [Chaffey et. al., 2005]

Muutosten aiheuttamiin tunnetiloihin ja niiden seurauksiin voidaan projektin johdon toimesta vaikuttaa ja varautua tunnistamalla vaiheet ja huolehtimalla etukäteissuunnittelusta. Suoraa uuden järjestelmän tuhoamista esiintyy harvoin, mutta pahimmillaan vastustus näkyy siten, että uutta järjestelmää ei käytetä tai sitä vältellään. Muutosvastarinnan esiintymisen syitä voivat olla ymmärryksen puute, vapauksien menettäminen (johdon kontrolli), havainnot hyötyjen jäämisestä kustannuksia pienemmiksi sekä epäonnistumisen pelko ja vajaatyöllisyys [Chaffey et. al., 2005, s. 393]. Yksilön näkökulman huomioonottaminen etukäteen on avainasemassa kriittisten järjestelmien muutoksenhallinnassa. Muutoksen jälkeen on hyvä tunnistaa niin onnistumiset kuin asiat, joita parantamalla olosuhteet olisi voitu tehdä suotuisammiksi.

## 5.7. Organisaation kulttuuri ja muutokset

Yksi huomioonotettavista osa-alueista, joka tulee ottaa huomioon muutosprosessia suunniteltaessa, on organisaation kulttuuri. Kulttuurin tyyppi on riippuvainen asenteista, arvoista ja uskomuksista, joita organisaation jäsenillä on [Chaffey et. al., 2005, 394]. Näiden lisäksi ei myöskään sovi unohtaa ulkoisia tekijöitä ja niiden vaikutusta kulttuuriin. Näkemykset muodostuvat siitä, miten organisaatio näkee itsensä suhteessa kilpailijoihin ja asiakkaisiin.

Johtamisella ja organisaation rakenteella on merkittävä rooli. Hierarkkinen organisaatio toimii yleensä formaalisti, kun taas matalatasoisissa organisaatioissa kulttuuriin vaikuttaa merkittävästi liiketoiminnan johtajan näkemykset.

Kyky reagoida muutokseen on osittain riippuvainen organisaation kulttuurista. Kulttuuriin liittyvien tekijöiden huomioiminen auttaa tunnistamaan muutoksen hallinnan menestystekijät [Schein, 1992]. Tunnistamalla mallit voidaan helpommin tunnistaa seikat, jotka tulee ottaa huomioon muutosprosessissa. Schein [1992] listaa kolmekohtaisen mallin organisaatioiden luokitteluun:

1. Olettamukset ovat näkymättömiä peruselementtejä. Organisaation sisällä vuosienkin aikana syntyneet näkemykset ovat merkittävä haaste muutoksen hallinnassa. Näitä olettamuksia pitää pystyä kyseenalaistamaan silloin, kun ne näyttävät muodostavan esteitä muutoksen syntymiselle.
2. Arvot ovat seikkoja, jotka ohjaavat käyttäytymistä. Suhtautuminen muutokseen on usein arvojen ohjaamaa. Käytöstä ohjaavat arvot välittyvät enemmän puhuttuna ja käytöksenä, kuin että niitä olisi kirjattuna jossakin. Samoin kuin olettamuksia, arvoja on vaikea muuttaa.
3. Perinteet ovat todellisia, usein kirjallisia, kulttuurin elementtejä. Kirjallisina säädöksinä niitä on helpoin muuttaa. Olettamuksista ja arvoista riippuu, miten niiden muuttamiseen reagoidaan organisaation sisällä.

Organisaation kulttuurin kannalta muutoksen onnistumiseen merkittävimmin vaikuttavat olettamukset ja arvot. Nämä ovat siten selvästi niitä seikkoja, joihin muutoksissa tulee panostaa kulttuurin kannalta.

Tarkasteltaessa yritysten kulttuurien suuntautumista voidaan ne jakaa neljään eri tyyppiin [Boddy et. al. ,2001]:

1. Selviytyjät. Matalan organisaation omaavat, asiakkaiden ohjaamat yritykset, joissa ympäristöllä on merkittävä rooli määriteltäessä yrityksen strategiaa.
2. Tuottavuuden ohjaajat. Hierarkkisen organisaatorakenteen omaavat markkinavetoiset yritykset, joiden liitynnät ulkoiseen ympäristöön ovat hyvässä järjestyksessä. Organisaatio toimii tyypillisesti hierarkkisen mallin mukaan.
3. Ihmissuhteiden ohjaajat. Yritykset, jotka toimivat kuten perheet ja joiden sisäinen vuorovaikutus on muodollista raportointia tärkeämpää. Organisaatorakenne on näillä matalampi ja esimerkiksi henkilöstön kehittäminen on tärkeää.
4. Vakaat. Yrityksissä ympäristö on pääasiallisesti jätetty huomioimatta johdon keskittyessä sisäisen tehokkuuden ylläpitoon. Organisaatorakenne on tällöin hierarkkinen.

Matalan organisaatorakenteen omaavilla organisaatioilla on kyky nopeampaan toimintaan muutosten läpiviemisessä. Toisaalta yrityksissä, joissa henkilöstöllisillä asioilla on suuri merkitys, muutokset voivat aiheuttaa suurta vastustusta, ellei henkilöstön näkökulmaa ole huomioitu muutosprosessin alkuvaiheista lähtien.

### **5.8. Muutoksen vaiheet**

Lähdettäessä muutosprosessiin tulee ymmärtää muutoksen taustat ja syyt, miksi muutokseen ylipäätään ollaan menemässä. Muutosstrategia tulee luoda. Siinä tulee olla määriteltynä mittarit laadun mittaamiseen, projektin etenemisen vaiheet ja aikataulut, tavoitteiden mittaaminen ja miten projekti organisoidaan.

Valmistauduttaessa muutokseen tulee selvittää muutokseen vaikuttavat ympäristötekijät. Tällöin kiinnitetään huomiota kriittisiin menestystekijöihin ja analysoidaan mahdollisia uhkatekijöitä. Valmistautumiseen kuuluu ennakoitavien tekijöiden lisäksi myös muutoksesta syntyvien seurausten selvittäminen. Viimeistään tässä vaiheessa muutoksesta tulee olla selkeä kuva, joka voidaan esitellä henkilöstölle.

Muutoksen toteuttaminen voidaan aloittaa tekemällä testimuutos. Tässä voidaan esitellä ja testata uusia työmenetelmiä, koulutusasioita ja lopullisen muutoksen eteenpäin viemistä. Usein työtehtävien määrittelystä, koulutusaikatauluista ja koulutuksen läpiviemisestä muodostuu odotettuakin suurempi työsaika. Tässä vaiheessa ei myöskään tule unohtaa yhteistyökumppaneiden, alihankkijoiden, asiakkaiden ja muiden sidosryhmien

tiedottamista. Uudet laitteet ja työmenetelmät voivat myös vaatia jossain tilanteissa tiedottamisvelvollisuuksia ja lupamenettelyjä viranomaisilta.

Organisaation sisällä muutos alkaa eri aikaan eri henkilöille. Toisille se alkaa siitä, kun prototyyppiä suunnitellaan, toisille testien alkamisesta tai vasta kun muutos tulee tuotantokäyttöön.

Päätös siitä, onko ohjelmisto valmis käytettäväksi, voi syntyä useiden eri henkilöstöryhmien toimesta. Sen voi tehdä ohjelmoija, hankkija, asentaja tai lisensoija. Kriittisissä järjestelmissä päätös ei voi syntyä monitahoisesti, koska tällöin vastuunkantajasta ei ole selkeää kuvaa.

Muutosprosessi ei pääty siihen, kun uusi järjestelmä on käytössä. Uudet työtavat voivat vaatia uudelleen järjestelyjä tai hienosäätöä. Muutoksen vaikutukset tulee mitata suunnitelmien mukaisesti. Tehokkuuden lisääntymisen mittaaminen heti käyttöönoton jälkeen ei vielä anna lopullista kuvaa toiminnan tehostumisesta, koska uudet työtavat eivät vielä välttämättä ole aivan niin sujuvia aluksi kuin ne ovat totuttelun jälkeen. Myös muutosvastarinnan heikentyessä tehokkuus lisääntyy.

## 6. Kriittisten tietojärjestelmien turvallisuusvaatimukset ilmailussa

Turvallisuus on tärkein kriteeri ja arvo ilmailussa. Euroopan lentoturvallisuusjärjestö Eurocontrol [Safety in air navigation, Eurocontrol] määrittelee turvallisuutta siten, että lennonvarmistuspalveluiden tärkeimpänä tehtävänä on varmistaa ilma-alusten turvallinen porrastaminen (riittävän turvaetäisyyden säilyttäminen) ilmassa ja maassa, samanaikaisesti ylläpitäen mahdollisimman hyviä operatiivisia ja taloudellisia olosuhteita.

SRC (Safety regulation commission, Eurocontrol) huolehtii Eurocontrol:in tehtävästä julkaista lennonvarmistuksen turvallisuussäädöksistä koko ECAC alueella. Tavoitteena on yhteen sovittaa muun muassa ohjelmistojen turvallisuuteen liittyvät kansalliset säädökset [Safety regulation commission, Eurocontrol].

SRC on julkaissut määräyskokoelman (ESARR, EUROCONTROL Safety Regulatory Requirement), jolla ohjataan jäsenmaiden kansallisia säädöksiä:

- Yleiskatsaus lennonvarmistuksen turvallisuuteen (ESARR 1)
- Turvallisuuteen liittyvä raportointi (ESARR 2)
- Turvallisuuden varmistusjärjestelmä (ESARR 3)
- Riskien hallinta (ESARR 4)
- Lennonvarmistuspalvelun henkilöstö (ESARR 5)
- Lennonvarmistusjärjestelmien ohjelmistot (ESARR 6)

### 6.1. Lennonvarmistusohjelmistojen turvallisuusvaatimukset

ESARR 6 (Eurocontrol safety regulatory requirement, Software in atm systems) on Eurocontrol:in jäsenvaltioille laatima ohjeisto, koskee ohjelmistojen käyttöä turvallisuuteen liittyvissä maanpäällisissä lennonvarmistusjärjestelmissä, joita käytetään lennonvarmistuspalveluiden tuottamiseen siviili-ilmailulle [ESARR 6, 2003]. Tärkeimpänä turvallisuustavoitteena pidetään ohjelmistojen osalta varmistumista siitä, että ohjelmistojen käytön riski saadaan vähennettyä sallittavalle tasolle.

#### 6.1.1. Yleiset turvallisuusvaatimukset

Lennonvarmistuspalvelun tuottajan tulee osana riskien hallintajärjestelmää laatia ohjelmistojen turvallisuuden varmistusjärjestelmä. Palvelun tuottajan tulee varmistaa, että varmistusjärjestelmä täyttää vähintään seuraavat minimivaatimukset:

- Ohjelmiston vaatimusten tulee osoittaa, mitä siltä vaaditaan, jotta se voisi täyttää riskianalyyseilla tunnistettavat turvallisuusvaatimukset.
- Kaikki ohjelmiston vaatimukset ovat jäljitettävissä.
- Ohjelmiston toteutus ei sisällä toimintoja, jotka epäsuotuisasti vaikuttaisivat turvallisuuteen.
- Lennonvarmistussovellus täyttää vaatimukset sillä luottamuksen tasolla, joka vastaa ohjelmiston kriittisyyttä.
- Vakuuttaa, että yllä olevat vaatimukset täytetään ja että ne ovat aina johdettavissa ajettavaan ohjelmaversioon, asetuksiin ja ohjelmistopaketteihin sekä kuvauksiin, joita on käytetty version tuottamiseen.

Lennonvarmistuspalvelun tuottajan tulee toimittaa vaadittavat todisteet viranomaiselle vaatimusten täyttämiseksi.

#### **6.1.2. Ohjelmistojen turvallisuuden varmistusjärjestelmä**

Ohjelmistojen turvallisuuden varmistusjärjestelmän tarkoituksena on varmistaa, että kaikki turvallisuustekijät ovat dokumentoituina osana yleistä riskien hallintaa. Tavoitteena on varmentaa, että ohjelmiston vaatimukset on laadittu oikein, ohjelmisto toimii vaatimusten mukaisesti, asetusten hallinta toimii ja ohjelmiston vaatimukset ovat jäljitettävissä.

Ohjelmistoille määritellään varmuustasot, jotka niiden tulee saavuttaa. Varmuustasosta määritellään, tuleeko ne saavuttaa itsenäisesti vai yleisesti. Toiminnan ja tasojen määrittelyjen oikeellisuuden varmistamiseksi vioista tulee raportoida.

#### **6.1.3. Ohjelmistojen turvallisuustasot**

Ohjelmiston varmuustason tulee vastata lennonvarmistusohjelmiston kriittisyyttä. Varmuustaso määritellään suhteessa vahinkoon, jonka ohjelmiston virheellinen toimivuus tai toimimattomuus voi aikaansaada. Lennonvarmistusohjelmistoissa ohjelmakomponentteja ei voida tarkastella erikseen. Varmuustaso määritellään aina kriittisimmän mukaan.

#### **6.1.4. Ohjelmistojen vaatimusten varmistaminen**

Ohjelmistojen vaatimusten osalta tulee varmistua siitä, että ohjelmiston toiminta vastaa käyttötarkoitusta tilanteissa, joissa järjestelmä toimii ylikuormitettuna tai tilanne on muulla tavoin epänormaali. Myös näissä tilanteissa tulee taata toiminnallisen suorituskyvyn, kapasiteetin ja kestävyysden säilyvyys.

### **6.1.5. Ohjelmistojen toiminnan varmistaminen**

Ohjelmiston toimintojen tulee vastata kaikilta osin ohjelmiston määrittelyjä. Vastaavuus todennetaan analyyseilla ja testeillä siten kun niistä on sovittu viranomaisen kanssa.

### **6.1.6. Ohjelmistojen asetusten hallinnan varmistaminen**

Lennonvarmistuspalvelun tuottajan tulee varmistua, että asetukset voidaan yksilöidä, jäljittää ja että muutosten kirjanpidosta huolehditaan. On tärkeää, että tunnistetaan, mitkä asetukset ovat kulloinkin olleet käytössä.

### **6.1.7. Ohjelmiston vaatimusten jäljitettävyyden järjestelmän vaatimuksiin**

Kaikki ohjelmistojen suunnittelutason vaatimukset tulee voida jäljittää järjestelmän vaatimuksiin.

## **6.2. ESARR-vaatimusten toteuttaminen Suomessa**

Suomi on Eurocontrol:in jäsenvaltio. Jäsenenä Suomen tulee laatia menetelmät vaatimusten täyttämiseksi. Eurocontrol seuraa jäsenten kykyä noudattaa määräyksiä tekemällä tarkastuksia.

Ilmailulaitos on tehnyt työtä määräysten täyttämiseksi. ESARR 6:n osalta työ menetelmien laatimiseksi on kuitenkin lennonvarmistuspalvelun tarjoajan kannalta vielä alkutekijöissä. Toisaalta voidaan myös todeta, että vaadittava turvallisuustaso saavutetaan jo kansallisten määräysten ja menetelmien turvin. ESARR tuo kuitenkin yhtenäisyyttä järjestelmien turvallisuuden määrittelyyn ja siten osaltaan varmistaa turvallisuustason säilymisen jatkossakin.

## **6.3. Ilmaliikenteen hallintapalvelun tekninen henkilöstö**

Lentoturvallisuushallinto on julkaissut 11.4.2005 voimaan tulleen ilmailumääräyksen koskien henkilöstöä, joka toimii lennonvarmistuksen operatiivisten järjestelmien turvallisuuteen liittyvissä tehtävissä, sekä ylläpitää ja huoltaa operatiivisessa käytössä olevia ilmaliikenteen hallinnan yhteydenpito-, suunnistus-, valvonta- sekä säähavaintolaitteita ja -järjestelmiä. Määräyksellä asetetaan ilmaliikenteen hallintapalvelun tarjoaja vastuuseen siitä, että henkilöt ovat päteviä suorittamaan heille kuuluvia ilmaliikenteen turvallisuuteen liittyviä tehtäviä [Ilmailumääräys, ANS M1-3, 2004]. Ilmailumääräyksen mukaan lennonvarmistusteknisen organisaation on:

- a) varmistettava, että lennonvarmistusteknisellä henkilöstöllä on asianmukainen koulutus ja pätevyys heille määrättyihin tehtäviin;
- b) varmistettava, että lennonvarmistustekninen henkilöstö tuntee työnsä vaikutukset palvelujen turvallisuuteen ja omaa riittävät

tiedot työskentelyrajoituksista, joita sovelletaan turvallisuuteen liittyvissä tehtävissä;

- c) varmistettava, ettei lennonvarmistusteknisen henkilöstön jäsen suorita operatiivisten järjestelmien turvallisuuteen liittyviä tehtäviä silloin, kun organisaatio tietää tai epäilee, että henkilö on fyysisen tai henkisen tilansa vuoksi kykenemätön hoitamaan näitä tehtäviä;
- d) ilmoitettava Lentoturvallisuushallinnolle poikkeamista, jotka vaarantavat, tai jos niihin ei puututa, vaarantaisivat ilma-aluksen, siinä olevien henkilöiden tai kenen tahansa muun henkilön turvallisuuden;
- e) varmistettava, että käytössä ovat asianmukaiset menetelmät sen varmistamiseksi, että lennonvarmistuksen operatiivisten järjestelmien turvallisuuteen liittyviin tehtäviin nimettävät henkilöt täyttävät määräyksen vaatimukset;
- f) varmistettava, että lennonvarmistusteknisen henkilöstön kelpoisuudesta ja pätevyydestä on olemassa todisteet, jotka esitetään viranomaiselle niin vaadittaessa.

Ilmailumääräyksen mukaan lennonvarmistuksen operatiivisten järjestelmien turvallisuuteen liittyvissä tehtävissä toimivien henkilöiden on:

- a) noudatettava organisaation kelpoisuusjärjestelmän mukaisia vaatimuksia pätevyyden varmistamiseksi;
- b) varmistettava, että he ovat saaneet riittävät tiedot tunteakseen työhön liittyvät palvelut ja niiden vaikutukset turvallisuuteen sekä mahdolliset työskentelyrajoitukset;
- c) pidättäytyttävä suorittamasta turvallisuuteen liittyviä tehtäviä, jos he tietävät tai epäilevät olevansa fyysisen tai henkisen tilansa vuoksi kykenemättömiä hoitamaan näitä tehtäviä;
- d) ilmoitettava Lentoturvallisuushallinnolle poikkeamista, jotka vaarantavat, tai jos niihin ei puututa, vaarantaisivat ilma-aluksen, siinä olevien henkilöiden tai kenen tahansa muun henkilön turvallisuuden;
- e) ilmoitettava muista poikkeamista siten kuin lennonvarmistusteknisen organisaation sisäisen ilmoittamisjärjestelmän ohjeistus velvoittaa;

- f) noudatettava niitä lisäehtoja, joita viranomaisen mahdollisesti asettaa.

Ilmailumääräyksellä asetetaan velvoitteita sekä organisaatiolle että tehtävissä toimiville henkilöille. Molemmat ovat omalta osaltaan vastuussa siitä, että henkilöstöllä on riittävä pätevyys ja työkyky tehtävien suorittamiseksi. Toinen merkittävä velvoite on se, että molempien tulee raportoida havaitsemistaan puutteista ja turvallisuutta vaarantaneista tilanteista.

#### **6.4. Järjestelmien käyttöönotto**

Seuraavassa kuvataan toimintamalli, jota noudatetaan Suomessa lennonvarmistuslaitteiden muutosprosessissa.

##### **6.4.1. Käyttöönoton hyväksyntä**

Lennonvarmistuslaitteiden käyttöönotto on tarkoin ohjeistettu prosessi. Laitteistojen asennus ja käyttö edellyttävät asiaan kuuluvien lupien saamista. Hyväksyntämenettely on riippuvainen laitteiston tai järjestelmän merkityksellisyydestä lentoturvallisuuteen [Käyttöönottohyväksyntä GEN130.01, 2001].

Hyväksyntämenettelyt voidaan jakaa kolmeen luokkaan hyväksyjän mukaan:

- lentoturvallisuushallinnon hyväksyntä,
- lennonvarmistusosaston hyväksyntä ja
- paikallinen hyväksyntä.

Lentoturvallisuushallinnon hyväksyttäväksi menevät järjestelmät, joilla voi olla välittömiä vaikutuksia lentoturvallisuuteen. Näitä ovat esimerkiksi navigointi ja tutkalaitteet. Vaikka lentoturvallisuushallinto antaisi hyväksynnän järjestelmän käyttöönotolle, menee hakemus aina myös lennonvarmistusosaston käsittelyn kautta.

Lennonvarmistusosasto vastaavasti hyväksyy järjestelmät, jolla käsitellään ilmailuun liittyviä tiedotteita tai viestiliikennettä. Paikallinen hyväksyntä riittää esimerkiksi lennonjohdon puhelinten ja kirjoittimien asennukseen. Paikallisesti hyväksyttävillä laitteilla ei saa olla lennonjohdollista merkitystä.

Käyttöönottolupaa haetaan hyväksyntäesityksellä. Esityksen tulee sisältää seuraavat asiakirjat [Käyttöönottohyväksyntä GEN130.01, 2001]:

- yhteenveto,
- yleiskuvaus,
- operatiiviset menetelmät,
- käyttäjien ja ylläpito henkilöstön valmius,
- laitteistojen toiminta,
- riskianalyysi ja
- käyttöönottosuunnitelma.

Operatiivisissa menetelmissä kuvataan, miten uusi laitteisto tulee muuttamaan toimintamenetelmiä, lentomenetelmiä, julkaisuja ja yhteistoimintasopimuksia. Käyttäjien ja ylläpito henkilöstön osalta tulee kuvata, miten koulutus on hoidettu, ja toimialapäällikön tulee antaa lausunto henkilöstön valmiuksista suoriutua tehtävistä. Perusedellytyksenä on, että henkilöstöä on riittävästi informoitu tulevasta tilanteesta sekä riittävästi koulutettu laitteiston käyttöön ja huoltomenetelmiin. Luotettavuusanalyysi vaaditaan vain niille laitteille, joille on esitetty numeerisia vaatimuksia [Käyttöönottohyväksyntä GEN130.01, 2001]. Analyysi on laskelma, jolla pyritään osoittamaan laitteiston käyttövarmuus. Laitteille, joille vaaditaan luotettavuuden todentamista (esimerkiksi ILS, Instrument landing system), on hyväksyntädokumentaatioon liitettävä kuvaus todentamismenetelmästä lopputuloksineen.

Toimintatesteillä on selvitettävä, kuinka laitteisto on todennettu toimivaksi ja kuinka on varmistuttu teknisten arvojen täyttymisestä [Käyttöönottohyväksyntä GEN130.01, 2001]. Ylläpitosuunnitelmassa määritellään, kuka laitteistoa huoltaa, kuinka laitteistoa huolletaan ja miten varmistetaan varaosien riittävyys.

Riskianalyysillä kartoitetaan vaikutuksia operatiiviseen toimintaan erilaisissa vikatilanteissa [Käyttöönottohyväksyntä GEN130.01, 2001]. Olennaisimmat kohdat analyysissä ovat, miten ja kuinka kauan toimintaa voidaan jatkaa vikatilanteessa ja millä järjestelmillä ja menetelmillä toiminta jatkuu.

Käyttöönottosuunnitelma antaa vastauksen seuraaviin kysymyksiin [Käyttöönottohyväksyntä GEN130.01, 2001]:

- kuinka laitteisto otetaan käyttöön,
- kuinka ja kenelle käyttöönotosta ilmoitetaan,
- kuka vastaa ja johtaa yliheittoa,
- miten varmistetaan uuden laitteiston toiminta yliheiton aikana,
- kuinka mahdollinen takaisinpaluu vanhaan järjestelmään tapahtuu ja kuinka kauan tällainen valmius ylläpidetään.

#### **6.4.2. Käyttöönottokriteerit**

Käyttöönottokriteereillä tarkoitetaan toimenpiteitä, joita järjestelmien käyttöönotolta edellytetään riippumatta siitä, kuka antaa käyttöönottoluvan [Käyttöönottohyväksyntä GEN130.01, 2001]. Käyttöönotot jaetaan kolmeen luokkaan: uusi laitteisto, muutettu laitteisto ja korjattu laitteisto. Merkittävää on huomata, että myös vikaantuneen laitteiston korjauksen jälkeen ollaan käyttöönottilanteessa, mikäli korjaus muuttaa laitteiston ominaisuuksia.

Uuden laitteiston käyttöönottaminen edellyttää seuraavia hyväksytyjä ja dokumentoituja toimenpiteitä [Käyttöönottohyväksyntä GEN130.01, 2001]: käyttöönottomittaus, käyttötestit, lentomittaus, huolto-ohje ja tekninen käyttöönottokatselmus.

Käyttöönottomittauksilla varmistetaan siitä, että laitteiston toimintaan vaikuttavat suureet ja asetukset ovat oikein ja merkittyinä mittauspöytäkirjaan, jonka asennustarkastaja hyväksyy. Tämän jälkeen asennetulla ja käyttöönottomitatulla laitteistolla voidaan aloittaa käyttötestit. Käyttötestien aikana laitteistoon ei saa tehdä muutoksia. Seurantajakson tulee olla riittävän pitkä luotettavuuden osoittamiseksi. Käytännössä jakson pituus on 30 vuorokautta keskeytymätöntä testausta. Testin aikana mitataan ja rekisteröidään säännöllisesti tärkeimmät parametrit sekä kirjataan ympäristöolosuhteet.

Teknisellä käyttöönottokatselmuksella varmistetaan, että asennustyö ja asennusympäristö täyttää asetetut vaatimukset ja että laitteistolla on edellytykset toimia kansainvälisten ja kansallisten vaatimusten mukaisesti [Käyttöönottohyväksyntä GEN130.01, 2001]. Katselmuksessa tarkastetaan laitteiston lisäksi myös muut toimintaan vaikuttavat laitteet, joita ovat mm. sähkö- ja lvi-laitteet.

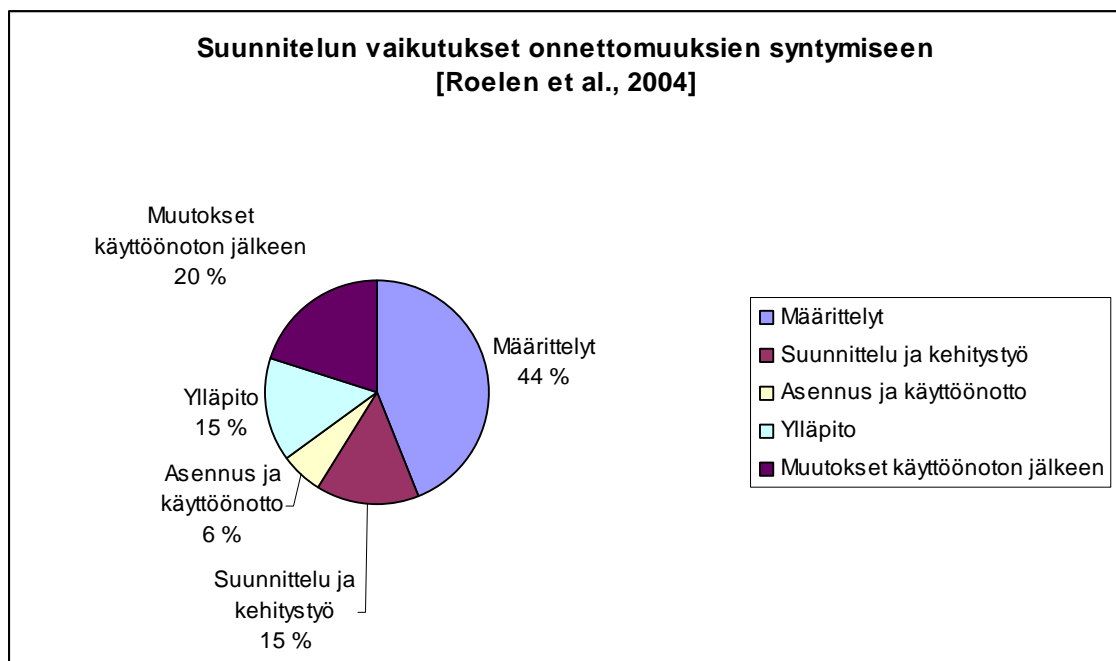
Muutettuja laitteistoja käsitellään periaatteessa uutena laitteistona ja sen hyväksyntämenettelyssä sovelletaan uuden laitteiston kriteereitä. Myös ohjelmapäivityksiä on käsiteltävä muutoksena.

### 6.5. Suunnittelun merkitys ilmailun tietojärjestelmissä

Jotta ilmailun kriittisten järjestelmien ongelmia voidaan ennakoida, tulee tuntee syyt niiden syntymiseen. Ilmailussa on laitettu paljon voimavaroja järjestelmien tekniseen kehittämiseen, mutta siitä huolimatta suunnittelusta johtuvia teknisiä vikoja syntyy.

Eurocontrol Experimental Centre (EEC) tekee työtä parantaakseen lennonvarmistuspalvelun turvallisuutta. EEC kehittää parempia turvallisuusmetodeita sekä sisäiseen että ulkoiseen käyttöön ja kehittää ja toteuttaa turvallisuuden oppimisprosesseja lennonvarmistusteollisuudelle. EEC:n tutkimuksen mukaan 59 % onnettomuuksista ja läheltä piti tilanteista syntyy suunnittelussa ja kehitysvaiheessa syntyneistä ongelmista (Taulukko 1, Onnettomuuksien syiden jakautuminen) [Roelen et al., 2004]. Tutkimuksessa tarkasteltiin ilmailun lisäksi ydinvoima- ja rautatieteollisuutta. Ilmailussa ja ydinvoimateollisuudessa suunnittelu oli syynä onnettomuuksiin yli puolessa tapauksista.

Taulukko 1, Onnettomuuksien syiden jakautuminen



Suunnittelu voidaan ymmärtää monella eri tavalla. On mielekkäämpää puhua suunnittelusta eikä suunnittelijoista. Suunnittelu on tyypillisesti niin laaja-alainen prosessi, että kaikki siihen osallistuvat eivät edes pidä itseään

suunnittelijoina, vaikka he osallistuisivat suunnittelutoimintaan. Suunnittelulla tässä tarkoitetaan vaihetta järjestelmän matalan tason vaatimusten keräämisestä järjestelmän toteutuksen alkamiseen. Onnettomuudella tässä tarkoitetaan vaaratilannetta, josta syntyi vahinkoa tai josta olisi voinut syntyä vahinkoa vastaavissa olosuhteissa. Onnettomuudet tyypillisesti sisältävät sekä virheitä että vikoja. Aina ei ole löydettävissä yhtä juurisyitä onnettomuuden syntymiseen. Juurisyys on tila, jonka syntyminen on välttämätöntä onnettomuuden syntymiselle [Roelen et al., 2004]. Jos onnettomuuden yhdenkin juurisyyn todetaan syntyneen suunnitteluvaiheessa, katsotaan koko onnettomuuden juurisyyn olevan suunnitteluvaiheessa, vaikka jokin muu onnettomuuteen liittyvä syy ei olisikaan suunnitteluvirhe.

Turvallisessa suunnittelussa tulee ymmärtää, mitä ovat hyvät suunnittelukäytännöt ja turvallisuusajattelu. Suunnittelun vaiheista tulee tunnistaa, missä erehdyksiä tapahtuu ja millaiset puutteet niitä voivat aiheuttaa. EEC:n raportti [Roelen et al., 2004] listaa joukon suunnittelussa tapahtuneita virhetyyppejä seuraavasti:

- suunnittelussa ei käytetty hyväksyttyä standardia,
- operaattorin väärinymmärtäminen tai tulkinta,
- käyttö suunnittelukehyksen ulkopuolella,
- paljastumattomat virheet ja viat,
- järjestelmän laajentaminen ymmärtämättä koko järjestelmän suunnittelukokonaisuutta,
- järjestelmän ei-tarkoituksenmukainen käyttö,
- puutteellinen varmuuskopiointi laitevian jälkeen,
- turvallisuustoiminnon ilmaisimen vika,
- alustan huonosta ymmärtämisestä johtuva komponentin vikaantuminen tai
- tuuleuksesta tai paloturvallisuudesta johtuvat pettämiset.

Ilmailun järjestelmät ovat monimutkaisia ja usein erittäin laaja-alaisia. Järjestelmien valvonta on tarkkaa ja laitteistot laadukkaita. Tästä syystä varsinaiset tekniset virheet ovat harvinaisia. Merkittäväksi syyksi jäävät siten ihmisten harhaluulot. Niitä voi syntyä operaattorien ja suunnittelijoiden välille, ja ne voivat syntyä operaattorien tarkoituksien ymmärtämättömyydestä ja operointiympäristöstä. Seuraavassa on EEC:n tutkimuksessa listattuja yleisiä eri aloilla syntyneitä väärinkäsityksiä ja uskomuksia:

- operaattorit etsivät aktiivisesti tietoja järjestelmän tilasta, vaikka he ovatkin usein passiivisia vastaanottajia,
- operaattorit opettelisivat uudet ohjeet, vaikka usein toimivatkin luottaen vanhaan osaamiseen,
- ympäristön muutoksilla ei ole juurikaan vaikutusta toimintaan, vaikka operaattorit toimivat eri tavoin eri olosuhteissa,
- operaattorit tuntisivat järjestelmän rajat, vaikka he eivät riskistä johtuen todellisuudessa uskalla kokeilla niitä,
- hätäolosuhteet olisivat vain tietyn kaltaisia, kun taas ne todellisuudessa ovat hyvinkin ennustamattomia,
- esivaroittimet lisääisivät luotettavuutta, kun operaattorit eivät rutiininomaisesti tarkasta varoittimia normaalikäytössä ja
- operaattorit ylläpitäisivät jatkuvasti korkeata valmiustasoa, kun heidän tarkkaavaisuutensa vaihtelee olosuhteiden mukaan.

Vaikka suuri osa juurisyistä löytyykin nimenomaan suunnittelumenetelmistä, ei riitä, että kehitetään suunnittelumekanismia ja standardeja. Erityistä huomiota tulee kiinnittää suunnitteluprosessiin osallistuvien henkilöiden vuorovaikutukseen, ja pyrkiä estämään heidän välilleen syntyviä väärinkäsityksiä. Käytännön ratkaisuja ongelmiin voidaan etsiä operatiivisten menetelmien ristiin kouluttamisella, yhteisillä suunnittelukokouksilla ja valittujen toimintatapojen huolellisella perustelemisella.

## 7. Case: Kriittinen tietojärjestelmä Ilmailulaitoksessa

Tässä tutkielmassa käsitellään teoreettisen osuuden lisäksi todellista kriittisen järjestelmän muutosta. Tässä luvussa käsitellään Ilmailulaitoksessa lennonvarmistuskäyttöön otettua AMC-Tool -järjestelmää. Tässä kuvatut asiat perustuvat kirjallisten lähteiden lisäksi omaan kokemukseeni, joka minulle syntyi osallistuessani projektityöskentelyyn.

Ilmailulaitos tuottaa asiakkailleen lentoasema- ja lennonvarmistuspalveluita. Ilmailulaitoksen toiminta-ajatus, arvot ja tahtotila (2004-2007) ovat kaikki hyvin turvallisuuslähtöisiä. Ilmailulaitoksen arvojen ensimmäisenä kohtana on turvallisuus, josta sanotaan:

”Ilmaliikenteen turvallisuus on toimintamme ehdoton lähtökohta. Turvallisuus syntyy henkilöstömme ammattitaidosta, yhteistyökkyvystä ja vastuullisuudesta.”

Etelä-Suomen lennonvarmistuskeskus on Ilmailulaitoksen yksikkö, jossa toimii Tampereen Aluelennonjohto. AMC-Tool -järjestelmän käyttöönotosta Ilmailulaitoksessa vastasi Etelä-Suomen lennonvarmistuskeskus.

Ilmailussa ja siten myös Etelä-Suomen lennonvarmistuskeskuksella on pitkät perinteet kriittisten tietojärjestelmien käyttöönotoista. Toimintaa ohjaavat monet eri määräykset, mutta näiden lisäksi noudatetaan monia suosituksia ja niin sanottuja hyväksi katsottuja periaatteita. Tätä tutkielmaa luettaessa on hyvä ymmärtää seuraavien toimintatapojen, periaatteiden ja menetelmien olemassaolo ja käyttö.

Laitejärjestelmät asennetaan lukittuihin laitetiloihin. Operatiivisia laitehuoneita on kaksi, jotka on varustettu identtisillä laitteistoilla ja yhteyksillä. Vikatilanteessa voidaan siten vastaava laite ottaa käyttöön toisesta huoneesta. Palvelimet asennetaan aina laitekaappeihin, joissa on ilmastointi ja rungon maadoitus. Sähkön saatavuus varmistetaan kytkemällä laitteet eri sähköryhmiin, ups-akuilla ja automaattisesti käynnistyvällä diesel-käyttöisellä generaattorilla.

Ilmailulaitoksen henkilöstöllä on kuvalliset henkilökortit ja eri tiloihin pääsy edellyttää avainkortin ja tunnusnumeron käyttöä. Ohjelmistojen ja laitteiden asennus ja käyttöönotto vaatii aina lupamenettelyn läpikäyntiä. Tärkeimmät palvelut sisältävät jo itsessään erilaisia varamenetelmiä tai viansietotiloja. Esimerkiksi tutkatietoa vastaanotetaan useilta tutkilta, ja jos joltain tutkalta ei tule tietoa tai sitä ei pidetä luotettavana, korvataan tieto toisen tutkan tiedoilla. Tutkatiedon esitysjärjestelmä on pääjärjestelmässä

kahdennettu, mutta silti tiedon saatavuus vikatilanteessa on varmistettu täysin pääjärjestelmästä erillään olevalla eri laitetoimittajan tekemällä järjestelmällä.

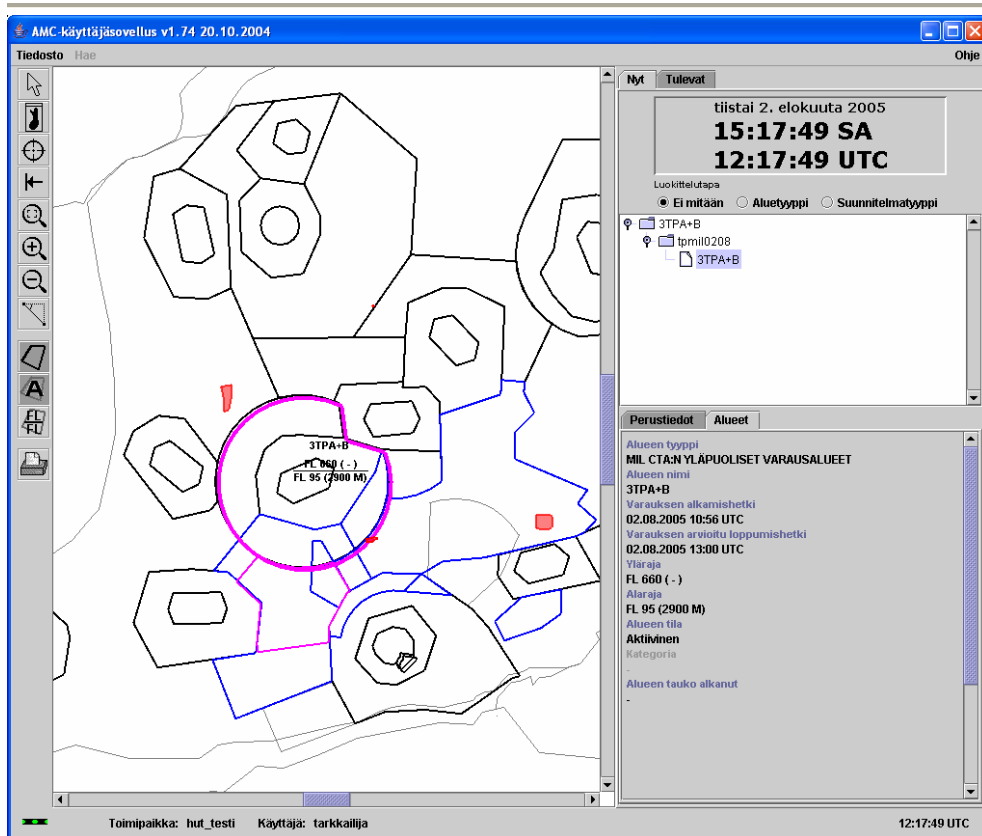
Seuraavassa käsitellään AMC-Tool -järjestelmän muutoksen etenemistä ja käyttöönottoa. Toiminnallisen kuvauksen lisäksi prosessia pohditaan suhteessa teoreettisiin malleihin, joita tässä tutkielmassa on aiemmissa luvuissa esitelty.

### **7.1. Järjestelmä ja sen käyttötarkoitus**

AMC-Tool- sovelluksen käyttöönotto kuuluu osana Ilmatila 2000+ -strategian toteutukseen. Ilmatila 2000+ implementaation myötä Suomessa otettiin käyttöön Eurocontrolin "Ilmatilan joustava käyttö"- konsepti (Flexible use of airspace). Konseptin käyttöönotto on suunniteltu tapahtuvaksi vaiheittain, ja seuraavassa vaiheessa odotetaan kansallisten palveluntarjoajien ottavan käyttöönsä AMC:n toimintaa ylläpitäviä ja tukevia teknisiä ratkaisuja. [AMC-Tool Turvallisuudenvarmistusasiakirja, 2004]

AMC-Tool- sovelluksen toiminnallisen ja teknisen määrittelyn lähtökohtana oli parantaa AMC:n ja Ilmavoimien yksiköiden yhteistoimintaa. Järjestelmää päätettiin kuitenkin laajentaa alkuperäisestä toimintaympäristöstä niin, että se kattaa kaikkien ilmatilan käyttäjien tarpeet. Tämä tuo toimintaympäristöön mukaan kaikki Suomen lennonjohdot, lennonneuvonnat, Puolustusvoimien ilma-ammuntoja suorittavat joukko-osastot sekä yleisilmailijat, esimerkiksi purjelentokerhot ja laskuvarjohyppykerhot. Näin mahdollistetaan nykyiseen ilmatilanhallintaan verrattuna ilmatilan hallinnan reaaliaikaisen kuvan levittäminen myös edellä mainituille kaikille ilmatilan käyttäjille ja edesautetaan kehittyneempien yhteistoimintamenetelmien luominen. [AMC-Tool Turvallisuudenvarmistusasiakirja, 2004]

Järjestelmän toiminnallisuus voidaan jakaa kahteen osa-alueeseen. Järjestelmällä voidaan esittää kulloinkin vallitseva ilmatilannekuva ilmatilavarausten osalta (Kuva 4, AMC-Tool käyttöliittymä). Nykytilanteen lisäksi voidaan tarkastella tulevia varauksia. Tiedot esitetään tekstimuotoisen tiedon lisäksi havainnollisena karttanäkymänä.



Kuva 4, AMC-Tool käyttöliittymä

Toinen merkittävä osa-alue on alueiden varaamiseen liittyvä lupamenettely. Tietyyntyyppiset varaukset vaativat luvan lennonvarmistusyksiköiltä, joita varaus koskee. Järjestelmä osaa kohdistaa pyynnöt oikeisiin yksiköihin varaukseen liittyvien alueiden perusteella (Kuva 5, AMC-Tool, aktivointipyyntö).



Kuva 5, AMC-Tool aktivointipyyntö

Järjestelmä koostuu palvelimesta ja asiakastyöasemista, joita järjestelmään kuuluu kymmeniä. Työasemia on alueenjohtajien lisäksi lentoasemien lennonjohdoissa sekä ilmavoimien yksiköissä. Työasemien kautta järjestelmään voidaan luoda ilmatilan käyttövarauksia ja lähettää niihin liittyviä aktivointipyyntöjä sekä lopetusilmoituksia.

## **7.2. Projektin vaiheet**

### **7.2.1. Ohjelmiston hankinta ja määrittelyt**

Amc-Tool -järjestelmän hankinta tapahtui pitkällisenä kehitystyönä. Hankinta alkoi Ilmavoimien ja Ilmailulaitoksen yhteisprojektina. Ohjelmiston omistaja on Ilmavoimat. Tällöin hankittavista ominaisuuksista päätti viimekädessä Ilmavoimat. Suunnittelu- ja määrittelykokouksissa oli edustajat molemmista organisaatioista. Käyttötarkoituksesta ja muista näkökulmista johtuen toiminnallisuuksia määriteltäessä jouduttiin usein päätyämään kompromissiratkaisuihin. Ensimmäisten versioiden valmistumisen jälkeen sovellusta yritettiin soveltaa käyttöympäristöön. Operatiivisesta ympäristöstä haettiin testitapauksia, joita yritettiin hoitaa uudella järjestelmällä. Tällöin havaittiin ongelmia, joita oli aavisteltu suunnitteluvaiheessa, mutta silloin niihin ei vielä reagoitu riittävästi.

### **7.2.2. Testaus**

Testauksen painopiste alkoi projektin edetessä siirtyä Ilmailulaitoksen hoidettavaksi. Samaan aikaan kun testeissä löydettiin useita testitapauksia, joita järjestelmä ei pystynyt käsittelemään, huomattiin toimintalogiikassa merkittäviä puutteita. Järjestelmän suoriutuessa perustehtävistä ihan hyvin niin sanotut poikkeukset ja ilmatilavarausten monimuotoisuus aiheutti ongelmia. Sovelluksen kehitystyö oli tällöin jo hyvin pitkällä. Tässä tilanteessa jouduttiin tekemään merkittävä valinta sen suhteen, että rajoitetaanko operatiivista toimintaa kieltämällä tietyt varaamismenettelyyn liittyvät joustot vai muutetaanko sovelluksen toimintalogiikkaa, jolloin se pitäisi suunnitella ja toteuttaa suurelta osin kokonaan uudestaan. Valinnassa päädyttiin sovelluksen muokkaamiseen.

Toiminnallisuuden muokkaamisesta seurasi automaattisesti suurehko aikataulullinen viive. Järjestelmä oli tarkoitus ottaa käyttöön osana suurempaa laite- ja työmenetelmä uudistusta. Valmistumisen siirtyminen voitiin sallia, koska myös muissa järjestelmä uudistuksissa oli kohdattu viivettä.

Lopullinen sovellus valmistui vain hieman ennen uutta aikataulullista takarajaa. Ennen käyttöönottoa suoritettiin valtakunnallisia testejä, joissa

havaittiin ongelmia, jotka olisivat estäneet sovelluksen operatiivisen käytön. Laitetoimittajan ja sovelluksen testaajien merkittävällä panostuksella ongelmista päästiin eroon ennen käyttöönottoa.

### 7.2.3. Koulutus

Järjestelmän kouluttaminen oli pitkälinen prosessi. Operatiivinen koulutus annettiin useassa eri vaiheissa. Varsinaista koulutusta edelsivät tietoiskut järjestelmän käytöstä ja sen ominaisuuksista. Ensimmäisessä vaiheessa annettiin niin sanottua laitekoulutusta. Koulutuksen tarkoituksena oli tutustuttaa käyttäjät järjestelmän ominaisuuksiin. Vaikka lopullisiin työmenetelmiin ei tässä vaiheessa otettu kantaa, heräsi oppilaille koko joukko avoimia kysymyksiä tulevista toimintatavoista. Näiltä ei voitu välttyä, koska ominaisuuksia esiteltäessä piti käyttää reaali maailman esimerkkejä. Hyvänä puolena menettelyssä oli, että oppilaiden kysymysten avulla voitiin löytää asioita, joita ei muutoin olisi vielä osattu ottaa huomioon.

Koulutusta annettiin paikan päällä, simulaattorissa ja puhelimen välityksellä tapahtuvana yhteiskäyttönä. Järjestelmän käyttöönoton viivästyessä koulutuksia jouduttiin antamaan useampaan eri otteeseen, koska työmenetelmät muuttuivat sovelluksen toiminnallisuuksien muutoksista johtuen.

Työmenetelmäkoulutukset annettiin myös useassa eri vaiheessa. Järjestelmän koulutuksenaikaisia puutteita jouduttiin paikkaamaan ohjeilla ja lisäkoulutuksilla. Lennonvarmistushenkilöstölle annetaan säännöllisesti kertauskoulutuksia. Näihin liitettiin mukaan AMC-Tool oppitunnit, joilla voitiin kerrata järjestelmän käyttöä ja tiedottaa muuttuneista työmenetelmäohjeista

Koulutettavat kokivat koulutuksen osittain liian lyhyeksi ja koulutuksessa olleet ohjeet ja työmenetelmät keskeneräisiksi [Koulutusraportti, 2004]. Käytetty aika koettiin liian lyhyeksi, koska koulutus oli niin laaja, että asioiden omaksumiseen ei jäänyt riittävästi aikaa. Ongelma olisi ollut koulutettavien mielestä korjattavissa lisäämällä harjoitustehtävien määrää.

Aikataulu ohjelmiston käyttöönotolle oli tiukka. Koska koulutettavia henkilöitä oli paljon ja heidän pääsemisensä operatiivisista vuoroista koulutuksiin oli vaikeaa, jouduttiin järjestelmän kouluttaminen aloittamaan hyvissä ajoin ennen käyttöönottoa. Koulutusten alkaessa ohjelmistosta ei ollut käytössä lopullista versiota, minkä vuoksi ohjeistukset ja työmenetelmät eivät myöskään olleet saaneet vielä lopullista sisältöään. Tästä seurasi se, että koulutettavat kokivat koulutuksen puutteelliseksi.

Koulutuksessa käytetty materiaali koettiin hyväksi ja kouluttajat kärsivällisiksi ja riittävän asiantunteviksi [Koulutusraportti, 2004]. Kouluttajat

kokivat ongelmalliseksi lähtötiedoiltaan hyvin eritasoiset koulutettavat. Käytettävissä ollut aika ei riittänyt paikkaamaan kenenkään henkilökohtaisia puutteita. Näitä puutteita käyttöönotossa olleet tukihenkilöt pyrkivät huomioimaan. Tukihenkilöinä toimivat projektin henkilöstö sekä kouluttajat.

Vaikka koulutuksen aikana koettiin muutosvastarintaa, positiivista oli, että koulutus saatiin vietyä läpi ilman suurempia poissaoloista johtuvia muutoksia. Kokonaisuutena koulutuksen koettiin olleen onnistunut [Koulutusraportti, 2004].

Koulutusta järjestelmän käytöstä annettiin runsaasti. Sitä kuitenkin haittasi järjestelmän koulutuksen aikaiset puutteet ja muutokset suunniteltuihin työmenetelmiin.

#### **7.2.4. Käyttöönotto**

Järjestelmän käyttöönotto sujui hyvin. Tiedot päivitettiin järjestelmään hyvissä ajoin ennen käyttöönottoa. Muissa samaan aikaan käyttöönotetuissa järjestelmissä ei ollut myöskään merkittäviä ongelmia. Lentoliikenteen määrää oli käyttöönottohetkellä rajoitettu ja myös ilmatilan varaajia ei ollut paljon. Vuorossa oli henkilöstöä enemmän kuin normaalisti. Henkilöstö oli käyttöönottohetkellä silmin nähden positiivisella mielellä. Henkilöstö turvautui järjestelmää päivittäessään aluksi työmenetelmäohjeistukseen, vaikka koulutuksissa käydyt asiat olivatkin suurelta osin hyvin opittuina. Yliheiton ja sitä seuranneiden lähipäivien ajaksi oli vuoroon varattu niin sanottuja tukihenkilöitä opastamaan käyttäjiä. Tukihenkilöinä toimivat järjestelmän pääkäyttäjät sekä kouluttajat. Menettely oli havaittu hyväksi aiempien käyttöönottokokemusten perusteella. Periaatteena yritettiin noudattaa sitä, että kukaan ei joutuisi tekemään ensimmäistä vuoroaan yksin ilman kouluttajaa. Käyttöönotossa Ilmavoimat ei ollut mukana, koska tietoverkkoihin liittyvät syyt viivästyttivät koulutuksen antamista.

#### **7.3. Kriittisyyden huomioiminen**

AMC-Tool -järjestelmän avulla eri lennonjohtoyksiköt saavat tiedon aktiivisista ilmatilavarauksista. Varaustietoon pohjautuen lennonjohdot pitävät muut johdetut ilma-alukset erossa alueista. Erittäin tärkeätä on, että kaikissa yksiköissä on sama reaaliaikainen tieto varauksista.

Järjestelmä on selvästi turvallisuuskriittinen, koska järjestelmän vikaantuminen voi aiheuttaa tilanteen, jossa ilma-alus virheellisesti, väärään tietoon pohjautuen ohjataan varausalueelle. Kuvattu tilanne ei välttämättä suoraan johda menetyksiin, mutta ilmailussa porrastusminimien alituskin käsitellään ja tutkitaan syiden osalta samalla tavalla kuin onnettomuus olisi syntynyt.

Ongelmien syntyminen kannalta vaara voi syntyä siitä, että lennonjohto menettää tilannekuvan ilmatilavarauksista tai lennonjohto saa virheellisen tiedon varaustilanteesta. Virheellisen tiedon pohjalta tehdyt ratkaisut voivat johtaa suoraan porrastusminimien alittumiseen ja siten vaaran syntymiseen. Tämän tyyppinen vika on pahin mahdollinen, koska tilanteen pitkittyessä virheelliseen tietoon perustuvia ratkaisuja voi tulla useita. Tilannekuvan hävitessä kokonaan lennonjohto pystyy huomioimaan ratkaisuisaan sen, että alueilla saattaa olla toimintaa ja käsitellä alueita kuten ne olisivat varattuina muulle toiminnalle. Vastaavat tiedot ovat osittain saatavilla myös muiden järjestelmien kautta.

### **7.3.1. Henkilöstön rooli toiminnan varmistamisessa**

Koska AMC-Tool -järjestelmä näyttää käyttäjälle tietoa ilmatilannekuvasta, tulee sen selvästikin indikoida myös oma toimintakuntonsa. Käyttäjä tekee havaintoja toimintakunnosta usein jopa sitä tiedostamatta. Tällaisia tarkastuksia ovat esimerkiksi hiiren liikuttaminen ja kellon seuraaminen.

AMC-Tool näyttää käyttäjälle vihreää merkkivaloa merkiksi palvelimen ja asiakasohjelman välisen tietoliikenteen toimivuudesta. Ohjelmistossa on haluttu varmistaa, että alueita ei aktivoitu käyttäjän sitä huomaamatta. Tämä toiminta on varmistettu kuittausmenettelyllä, jossa käyttäjälle tärkeiden alueiden aktivoituminen tai peruminen edellyttää dialogissa tehtävää valintaa. Menettelyn yhteydessä katkokset havaittaisiin nopeasti, vaikka ohjelmisto jostain syystä näyttäisi virheellisesti olevansa toimintakunnossa.

### **7.3.2. Vaaratekijöiden tunnistaminen**

AMC-Tool:in operatiiviseen käyttöön liittyvät vaaratekijät on pyritty systemaattisesti tunnistamaan ja listaamaan ne toimenpiteet, joilla saavutetaan hyväksytty turvallisuus- ja luotettavuustaso korkealaatuisen ilmaliikennepalvelun antamisen takaamiseksi [AMC-Tool Turvallisuudenvarmistusasiakirja, 2004]. Tunnistamisessa on otettu huomioon

- järjestelmän koko elinkaari,
- järjestelmä koko laajuudessaan sisältäen kaikki järjestelmän piiriin kuuluvat yksiköt ja
- kolme ATM-järjestelmään kuuluvaa elementtiä; laitteisto, menetelmät ja käyttäjät.

Vaaratekijöiden tunnistusprosessissa tekijöitä arvioidaan ja tarvittavat toimenpiteet määritellään. Prosessiin kuuluu järjestelmän laajuuden, rajojen, liittymäpintojen ja toimintaympäristön tunnistaminen. Vaatimusten mukaisesti

turvallisuustavoitteiden määrittelyyn sisällytettiin vaaratekijöiden ja vikatilanteiden tunnistaminen. Tunnistamisen pohjalta luotiin vaaratekijät eliminoivat työmenetelmät, joiden avulla saavutetaan määritellyt turvallisuustavoitteet. Turvallisuustarkastelu uusitaan aina sovellusmuutosten yhteydessä.

### **7.3.3. Vaaratekijöiden luokittelu**

Turvallisuustarkastelussa tunnistetut vaaratekijät luokitellaan niiden operatiivisten vaikutusten perusteella. Luokitusjärjestelmä perustuu ESARR 4:n mukaiseen kvalitatiiviseen vaaratekijän vakavuuden operatiivisen vaikutuksen arviointiin (Liite 1, Risk Assessment and Mitigation in AMT- ESARR 4).

Arvioitaessa luokittelua pyrittiin vaaratekijä arvioimaan suhteessa eri lennonvarmistuksen osa-alueisiin. Näitä ovat vaikutukset lennonjohtajiin (esim. työkyky tai työkuorma), vaikutukset ohjaamomiesthistöön, ilma-aluksen toimintakykyyn, lennonvarmistuksen toimintakykyyn ja turvallisen lennonvarmistuspalvelun tarjoamiseen. Näiden lisäksi huomioitiin joukko muita tekijöitä, kuten ilma-alusten lukumääräinen altistuminen ja toimintaympäristön luonne.

Arvioitaessa vaaratekijän vakavuusasteen ja ilmenemisen todennäköisyyden suhdetta tulee näiden yhteisvaikutuksen olla sellainen, että se sijoittuu turvallisuustasoa kuvaavan taulukon (kuva 6, Turvallisuustason määrittely) valkoiselle alueelle. Esimerkiksi jos vaaratekijän vakavuusasteen arvioidaan olevan sellainen, että se aiheuttaa luokkaan 3 (Major Incident) luokiteltavan vaaratilanteen, sen tulee olla ilmenemisen todennäköisyydeltään hyvin harvinainen (Very Rare) tai erittäin harvinainen (Extremely Rare), jotta tilannetta voidaan pitää hyväksyttävänä, ja turvallisuustason katsotaan olevan riittävä [AMC-Tool Turvallisuudenvarmistusasiakirja, 2004].

Severity of Hazard	Likelihood of Hazard Occurring			
	Possible	Rare	Very Rare	Extremely Rare
1				
2				
3				
4				
5				

Not Tolerable	Tolerable
---------------	-----------

Kuva 6, Turvallisuustason määrittely [AMC-Tool Turvallisuudenvarmistusasiakirja, 2004]

#### 7.3.4. Vaaratekijöiden kartoitus ja turvallisuustavoite

Vaaratekijöiden ja riskien kartoittamiseen sekä työmenetelmien suunnitteluun osallistui AMC-ylläpitöryhmän jäsenet. Kartoittamisesta syntyi taulukkomuotoinen riskiloki. Jokaisesta vaaratekijästä kirjataan seuraavat kohdat:

- vaaratekijä,
- vaaratekijästä aiheutuva riski,
- ensisijaiset riskiä vähentävät toimenpiteet,
- todennäköisyys ennen toimenpiteitä,
- vakavuusaste toimenpiteiden jälkeen,
- vakavuusasteen määräytymisen perusteet,
- todennäköisyys toimenpiteiden jälkeen,
- todennäköisyyden määräytymisen perusteet,
- toissijaiset riskiä vähentävät toimenpiteet ja
- viite (esimerkiksi ohjeistukseen).

Turvallisuustavoitteen saavuttamiseksi riskiluokituksen vakavuusasteen täytyy suoritettujen toimenpiteiden jälkeen olla kaikilta osin sellainen, että luokkaan 1 kuuluvia riskitekijöitä ei esiinny. Muiden vaaratekijöiden arvioidun vakavuusasteen ja oletetun ilmenemistiheyden välisen suhteen tulee pysytellä turvallisuustason määrittelytaulukon valkoisella alueella (Kuva 4, Turvallisuustason määrittely).

### **7.3.5. Tekninen riskianalyysi**

AMC-Tool -järjestelmästä tehtiin operatiivisen riskianalyysin lisäksi tekninen riskianalyysi. Analyysissä kartoitettiin järjestelmän eri osien vikaantumista sekä ympäristön muutosten vaikutuksia järjestelmälle. Ympäristön osalta arvioitiin esimerkiksi tietoverkkojen, turvaratkaisujen ja sähkösyöttöjen häiriöitä. Kun vikaantuvat osat oli lueteltu, arvioitiin niiden todennäköisyys ja vaikutus kokonaisuuden kannalta. Analyysillä varmistettiin, että vikaa vastaava operatiivinen varamenetelmä oli laadittu ja selvitetty, miten viasta toipuminen tapahtuu ja kuinka kauan se kestää.

### **7.4. Järjestelmän viat**

AMC-Tool -järjestelmässä ei ole noin vuoden kestäneen käytön aikana esiintynyt lainkaan fyysisiä laitevikoja. AMC-Tool:ssa näihin tilanteisiin on kuitenkin varauduttu siten, että laitteistosta on asennettuna täysin identtinen kokoonpano. Se voidaan ottaa käyttöön, mikäli järjestelmässä havaitaan fyysisiä ongelmia.

Käytön aikana sovelluksesta on löytynyt useita suunnitteluvikoja, jotka eivät ole kuitenkaan estäneen järjestelmän operatiivista käyttöä. Hankalimmat viat ovat olleet tietoliikenneongelmia. Näistä on tullut tieto joko käyttäjälle sovellusikkunaan tai tekniseen valvontamonitoriin. Osa ongelmista korjaantuu itsestään yhteyksien palautumisen jälkeen ja osa vaatii asiakassovelluksen uudelleenkäynnistystä.

### **7.5. Operatiivinen toiminta vikatilanteissa**

Toiminta vikatilanteissa on ohjeistettu alueenjohtajan toimintaohjeessa [AMC-varaohje, 2005]. Ohjeistuksen mukaan päätöksen varaohjeen ja sitä kautta varatyömenetelmien käytöstä tekee alueenjohtajan vuoro esimies. Päätöksen tekeminen vaatii normaalisti neuvottelua teknisen henkilöstön kanssa. Hajautettu järjestelmä ei tavallisesti lamaannu kokonaan. Tällöin joudutaan arvioimaan vian tai katkoksen laajuus. Jos katkos koskee vain joitain yksiköitä, on helpompaa ja turvallisempaa hoitaa koordinaatio näihin yksiköihin puhelimitse, kuin että ajettaisiin koko järjestelmä alas katkoksen ajaksi. Varamenetelmiin siirtyminen tarkoittaa miltei aina pienempää kapasiteettia kuin normaalitilanteessa.

Ohjeistuksessa kuvataan ne toiminnot, jotka tehdään eri tavalla kuin järjestelmän ollessa käytössä. Vikatilanteessa ohjeistuksesta tulee ottaa kopio, josta yliviivaamalla ja tekijän puumerkinnällä varmistetaan, että kaikki ohjeistuksessa kuvatut asiat tulee tehdyksi. Tehtävät toimenpiteet raportoidaan vuoron esimiehelle, joka edelleen koordinoi asiat muihin lennonjohtoyksiköihin. Jos järjestelmän toimimattomuus on pitkäaikainen,

suoritetaan esivalmistelut, jotka mahdollistavat pidempiaikaisen toiminnan ilman AMC-Tool:ia.

Tietojärjestelmän toimimattomuus aiheuttaa sen, että täyttä reaaliaikaista varaustilannetta ei ole käytettävissä. Merkittävin osa tiedoista on kuitenkin nähtävissä pääjärjestelmän tutkanäytöllä. Tiedot siirretään käsin valotaulukartoille, joita on jokaisessa lennonjohtopöydässä. Jokaiseen tauluun on siten tussilla piirrettävä varatut alueet sekä muut mahdolliset ilmoitukset. Ampuma-alueita koskevat varaustiedot on mahdollista tulostaa toisesta tietojärjestelmästä, joten niiden osalta jaellaan karttakopiot varaustilanteista lennonjohtopöytiin.

Vikatilanteessa toimenpiteiden kirjaaminen tapahtuu hieman eri tavalla kuin normaalisti. Esimerkiksi ammuntojen aloitus- ja lopetusilmoitusten kirjaamiseen käytetään eri kaavaketta kuin normaalisti. Tämä tehdään siksi, että vuoron esimiehen tulee kirjata tietojen vastaanottaminen henkilökohtaisesti ilmoituskavakkeeseen. Varauspyyntöjen käsittely AMC-Tool:in toimissa on hyvin pitkälle automatisoitua. Järjestelmä osaa esimerkiksi lähettää aktivointipyynnöt oikeisiin yksiköihin varattavien alueiden perusteella. Järjestelmän vikatilanteessa voidaan yksiköt katsoa listasta, mutta se on toimenpiteenä melko hidasta. Ilman järjestelmää varaukset joudutaan koordinoimaan vuoron esimiehen kanssa ja muilta lennonjohtoyksiköiltä luvat joudutaan pyytämään puhelimitse.

## 7.6. Tekninen näkökulma

Lennonjohdon tietojärjestelmäyksikkö ylläpitää AMC-Tool -järjestelmää. Ylläpito käsittää palvelimille tehtävät tarkastukset, joita ovat esimerkiksi levytilojen tarkastukset, prosessien tilojen tarkastaminen, tietokannan tarkastaminen ja varmuuskopioiden ottaminen. Asiakassovelluksille ei tehdä sovelluksen takia mitään normaalien työaseman huoltojen lisäksi. Palvelimia on kaksi. Ne on varustettu identtisillä ohjelmistoilla. Toinen palvelin voidaan vikatilanteessa tarvittaessa käynnistää ja siirtää siihen viimeisin varmuuskopio ohjelmiston tiedoista.

Ongelmien selvittäminen käynnistyy vian havaitsemisesta. Tieto voi tulla käyttäjän ilmoituksena tai valvontajärjestelmän kautta. Käyttäjien ilmoitukset vaihtelevat paljon. Hankalimpia ovat ilmoitukset ”.. yövuorolainen kertoi, että eilen iltapäivällä AMC oli jotenkin hitaalla...”. Ongelman alustava vakavuusaste on kuitenkin usein määriteltävissä suoraan kuvatusta ongelmasta. Vakavuusaste voidaan AMC-Tool -järjestelmässä luokitella karkeasti neljään eri luokkaan. Kyseessä voi olla jokin seuraavista ongelmatyypeistä:

- Vakava ohjelmistovika (tiedot eivät päivitty ja käyttäjä ei saa siitä heti tietoa),
- Järjestelmän ylikuormittuminen tai verkkoviiveet (järjestelmän tiedot päivittyvät hitaasti ja viestien perille menossa viiveitä),
- Työaseman jumiintuminen (hiiri ei liiku tai sovellus ei ota komentoja vastaan),
- Sovellus itse indikoi ongelmista (ilmoitus "ei yhteyttä palvelimeen" )

Lista ei kata kaikkia tapaustyyppisiä, mutta kokemukseni mukaan tämä luokittelu antaa hyvän lähtökohdan vakavuuden määrittelemiseksi. Ensimmäisen vikatyypin kohdalla, tulee koko järjestelmän käyttö kyseenalaistaa epäluotettavana. Toinen kohta ei välttämättä edellytä seuranta vaativampia toimia ja kahdesta viimeisestä selvittää yleensä työaseman tai sovelluksen uudelleen käynnistämällä. Järjestelmän käyttöympäristön ja sitä kautta siis järjestelmän kriittisyyden kannalta olennaisin seikka on, että huomaako käyttäjä, että järjestelmä ei toimi oikein.

Teknisen henkilöstön käytössä on valvontajärjestelmä, jolla valvotaan työasemia, verkkoja ja palvelimia. Perusvalvonta tarkkailee esimerkiksi miten paljon eri prosessit käyttävät tietokoneen prosessoria, kuinka paljon tietokantayhteyksiä on käytössä ja riittääkö tietokoneen levytila. Näiden lisäksi tarkkaillaan sovellusten tuottamia lokitiedostoja. Jos sovellus tuottaa lokiin viestin ongelmista, välittyy viesti keskitettyyn valvontajärjestelmään. Tekniikan vuoro esimies saa viestin työpisteeseensä ja voi siten välittömästi ryhtyä tarvittaviin toimiin. Valvontajärjestelmä välittää viestit eteenpäin, mutta toiminnan edellytyksenä on, että vian mahdollinen esiintyminen on tunnettu, ja että sovellus osaa välittää tiedon lokiin. Uusien, ennalta tiedossa olemattomien vikojen havaitseminen jää usein käyttäjälle.

Toiminta vikatilanteissa riippuu luonnollisesti viasta ja sen laajuudesta. Ensimmäinen tehtävä on kuitenkin aina tiedottaa operatiivista henkilöstöä mahdollisista ongelmista. Käyttäjät voivat tällöin nostaa valppaustasoa tekemällä ylimääräisiä tarkistuksia tai muutoin varmistua tehtyjen toimien oikeasta toiminnasta. Käytännössä, jos esimerkiksi järjestelmällä on lähetetty kuittauspyyntö, yksiköstä voidaan puhelimitse varmistaa viestin saapuminen. Operatiivinen henkilöstö osallistuu miltei aina vikojen laajuuden selvittämiseen.

Operatiivisen vuoron esimies tekee päätökset työmenetelmien käytöstä kuultuaan ensin teknisten henkilöiden arvion tilanteesta. Heti, kun tieto mahdollista ongelmista saadaan, siirrytään melko usein niin sanottuihin

puhelinvarmistuksiin. Tämä tarkoittaa, että mahdollisesti vikaantuneen järjestelmän käytöstä ei heti kokonaan luovuta, mutta tietojen oikeellisuus tarkastetaan puhelimella.

Kun vian laajuudesta on saatu luotettavampia tietoja, tekninen henkilöstö aloittaa vian korjaamisen samalla koko ajan tiedottaen toimistaan ja niiden mahdollisista vaikutuksista operatiiviselle vuoron esimiehelle. Vikojen korjaamisessa on tärkeää, että niistä tiedotetaan täsmällisesti järjestelmää käyttävälle henkilöstölle. Heidän avullaan saadaan tietoa korjaustoimien oikeellisuudesta ja tunnistetaan heti mahdolliset korjaustoimista syntyvät uudet ongelmat.

Paluu normaalitoimintaan ongelman korjaamisen jälkeen tehdään hallitusti ja tyypillisesti useassa eri vaiheessa. Ensimmäiseksi järjestelmän tiedot päivitetään vastaamaan operatiivista tilannetta. Sen jälkeen sillä tehdään riittävä määrä testejä, ei operatiivisilla tiedoilla. Onnistuneiden testien jälkeen siirrytään järjestelmän operatiiviseen käyttöön. Käytön aikana aluksi varmistetaan tietojen päivittymiset. Kun riittävä varmuus järjestelmän oikeasta toiminnasta on saatu, päätetään siirtymisestä normaaliin toimintaan.

## **8. Muutos ja sen hallinta Case Ilmailulaitoksessa**

Seuraavassa tarkastellaan Ilmailulaitoksessa käyttöön otetun AMC-Tool -järjestelmän muutosta ja sen hallintaa, miten toiminnalliset prosessit muuttuivat, mitkä organisaation osat olivat muutoksen kohteena, millainen muutos oli ohjelmistotuotteen kannalta. Lisäksi tarkastellaan muutostyyppiä, pohditaan millainen muutos oli yksilön ja organisaation näkökulmasta ja arvioidaan muutoksen toteutumista.

### **8.1. Toiminnallisten prosessien muutos**

Aluelennonjohto vastaa ilmatilavarausten myöntämisestä. Ilmatilaa haluavat käyttöönsä monet eri yksiköt. Näitä ovat yleisilmailijoista purjelentäjät, ja laskuvarjohyppääjät sekä puolustusvoimista ilmavoimat lentojen osalta ja maa- ja merivoimat ammutaharjoitusten osalta.

Toiminnallinen muutos tarkoitti siirtymistä käsin ylläpidettävistä tiedoista tietojärjestelmän käyttöön. Ennen muutosta ilmatilan varaukset ylläpidettiin ja myönnettiin ilman tietojärjestelmiä. Tiedot varauksista ylläpidettiin aluelennonjohdossa paperilla ja valotaulukartoilla, joihin ne piirrettiin erivärisillä tusseilla. Muilla lennonjohtoyksiköillä oli omat menetelmänsä, joilla he ylläpitivät tiedot heille merkityksellisistä ilmatilavarauksista. Uudessa tietojärjestelmässä tiedot esitetään havainnollisena karttanäkymänä ja tekstinä, jolloin valotaulumerkintöjä ja paperille kerättyjä tietoja ei tarvitse ylläpitää.

Varausten myöntäminen hoidettiin ennen aluelennonjohdon vuoron esimiehen toimesta soittamalla yksiköihin vuoron perään, ja pyytämällä heiltä suostumus varauksen myöntämiseen. AMC-Tool -järjestelmässä on toiminnallisuus, jolla voidaan lähettää järjestelmän sisäinen sähke, jolla lupaa pyydetään ja pyyntöön vastataan. Pyyntö ohjautuvat automaattisesti oikeisiin yksiköihin varattavien alueiden perusteella, ja yksiköt voivat vastata pyyntöön haluamassaan järjestyksessä. Kun kaikilta on saatu myöntävä vastaus, aktivoituu alue automaattisesti karttanäkymään.

### **8.2. Organisaation muutoskohteet**

Muutos oli strateginen, koska järjestelmän tavoitteena oli jakaa tietoja olemassa olevista ja suunnitelluista varauksista laajemmin, jotta välttyttäisiin päällekkäisiltä varauspyynnöiltä. Tavoitteen mukaisesti esimerkiksi eri lennostot voivat paremmin keskenään sopia käytettävistä alueista ilman, että aluelennonjohto toimisi välikätenä neuvoteltaessa, kuka alueen saa milloinkin käyttöönsä.

Suomessa oli kaksi lentotiedotusaluetta, joita valvoi kaksi eri alueenonjohtoa. Rakenteellisella muutoksella mahdollistettiin siirtyminen keskitettyyn järjestelmään, jossa varaukset hoidetaan yhtenäisesti yhden järjestelmän kautta. Toiminnot keskitettiin yhden alueenonjohdon hoidettavaksi.

### **8.3. Muutos ohjelmistotuotteen kannalta**

Järjestelmän kannalta muutos oli suuri, koska korvattavaa tietojärjestelmää ei ollut. Täysin uusi järjestelmä vaati suunnittelua työpisteiden osalta. Niin alueenonjohdossa kuin muissakin lennonjohtoyksiköissä uudelle näyttölaiteelle, näppäimistölle ja hiirelle piti löytää hyvä ja riittävä tila. Lennonjohtopöydät olivat ennestään täynnä laitteita, joten tilan löytäminen uudelle ei ollut ongelmaton.

Yksittäisiä työasemia oli ympäri Suomea. Vaikka ohjelmistoa jouduttiin päivittämään varsinaisen asennuksen jälkeen, ei se osoittautunut hankalaksi, koska sovelluksen käynnistyksen yhteydessä latautuu uusi ohjelmaversio automaattisesti.

### **8.4. Muutostyyppi ja luokittelu**

AMC-Tool:in tuomassa muutoksessa oli kyse prosessien rakenteiden, tekniikan, henkilöstön ja osin myös kulttuurin muutoksesta. Toiminnalliset prosessit suunniteltiin vanhojen työmenetelmien päälle, mutta silti muutokset olivat suuria. Esimerkiksi alueiden varaamisessa vuoron esimiehen rooli vähentyi, koska järjestelmään määriteltiin etukäteen kenelle pyynnöt tuli osoittaa ja purjelentovarausten myöntämiseen määriteltiin niin sanottuja vakiovarauksia, jotka voitiin suoraan myöntää. Tekniikan osalta muutos tarkoitti mittavaa muutosta, jossa puhelin, paperit ja valotaulujen tussimerkinnot korvattiin tietojärjestelmän sisäisillä viesteillä ja näyttöruudun karttanäkymillä. Henkilöstön osalta muutos tarkoitti siirtymistä yhden työpisteen tehtävistä kahden työpisteen yhteistoimintaan. Uudistuksen myötä perustettiin uusi AMC-työpiste, ja henkilöstö joutui opettelemaan uudet työmenetelmät. Muutoksessa oli osittain kyse myös kulttuurin muutoksesta. Suomessa oli kaksi lentotiedotusaluetta, joiden ilmatilan hallinta kuului niitä valvoville alueenonjohdoille. Muutoksen myötä molempien lentotiedotusalueiden ilmatilanhallinta siirtyi Tampereen alueenonjohdon AMC:n hoidettavaksi.

Muutoksessa oli kyse ei-jatkuvasta muutoksesta, koska suuri muutos tehtiin kerralla. Muutos johtui liiketoiminnan perusteiden muuttumisesta ja kohdistui McKinseyn määrittelemiin koviin ja pehmeisiin s-kohteisiin [Waterman et. al., 1980]. Muutos kohdistui järjestelmän lisäksi strategiaan ja rakenteisiin.

Pehmeistä kohteista muutos kohdistui taitoihin, henkilöstöön ja tyyliin, mutta arvojen muuttumista ei ollut tunnistettavissa.

Nadler et al. [1995] mukaan muutos voidaan luokitella uudelleenjärjestelyksi. Muutos oli tyypiltään ei-jatkuva ja ennakoiva. Muutos oli suuri, mutta mitään välitöntä pakkoa sen tekemiselle ei ollut. Muutoksella siirryttiin uuteen palvelumuotoon, jossa aluevarausten ennakkosuunnittelua voitiin helpottaa, ja varausten varaaminen voitiin hoitaa joustavasti.

Liiketoimintaprosessien kannalta muutoksessa oli kyse liiketoimintaprosessien uudelleenjärjestämisestä. Tehdyillä muutoksilla pyrittiin omasta tahdosta parantamaan joustavuutta, tiedon jakamista, automatisointia ja sitä kautta myös tuottavuutta. Liiketoimintaprosessien uudelleenjärjestämisessä epäonnistumisen vaara on suurin [Chaffey et al., 2005, s.388].

### 8.5. Muutos yksilön näkökulmasta

Uusi tietojärjestelmä vaati paljon uutta osaamista. Järjestelmän toimintojen oppimisen lisäksi työntekijöiden piti oppia uudet toimintatavat ja määräykset. Ennen alueiden piirtämiseen meni merkittävä osa ajasta, kun taas uudessa järjestelmässä piirtämistä ei tarvita lainkaan. Toisaalta alueiden täsmällinen syöttäminen järjestelmään vaatii aiempaa enemmän aikaa verrattuna tietojen kirjaamiseen paperille. Vuoron esimies vapautui alueiden myöntämiseen liittyvistä rutiineista, koska uusi järjestelmä osaa jakaa pyynnöt automaattisesti oikeisiin lennonjohtoyksiköihin. Lukumääräisesti henkilöstöä tarvitaan periaatteessa yhtä monta kuin aiemmin. Koska järjestelmä on uusi ja sen käyttö vaatii totuttelua, tarvitaan ruuhka-aikoihin enemmän henkilöitä. Toisaalta ei pidä unohtaa, että järjestelmällä käsitellään koko Suomen ilmatilavaraukset, ja sen voidaan olettaa vähentäneen työtä muista lennonjohtoyksiköistä.

Muutos vaikutti organisaation henkilöstön työtehtäviin ja työympäristöön. Muutosprosessin aikana oli selvästi aistittavissa muutoksen vaikutus heidän mielialoihinsa ja suhtautumiseensa muutokseen. Tunnereaktioiden vaihtelusta oli mahdollista tunnistaa eri henkilöillä piirteitä Adams et al. [1976] luokittelun mukaisista tunnetiloista. Vaikka tietojen jakaminen aloitettiin jo projektin alkuvaiheessa, tuli tieto muutoksesta toisille järkytyksenä. Tämä johtuu siitä, että henkilö ei valmistautunut muutokseen ja saattoi esimerkiksi pelätä, että hän ei opi uusia työtapoja. Aivan ensimmäisten koulutusten aikana oli havaittavissa kieltämistä. Negatiivisia asioita korostettiin kiinnittämällä huomiota järjestelmän puutteisiin ja keskeneräisyyksiin. Muutoksen vastustaminen oli tässä vaiheessa suurinta. Kieltämistä seurasi eräänlainen masennus ja periksi antaminen. Henkilöstö huomasi, että muutos on

väistämätön ja että se tulee vastustuksesta huolimatta. Vaiheeseen tyyppillistä muutoksen tarpeellisuuden kiistämistä ei esiintynyt. Periksi antamisessa siirryttiin eteenpäin kohti lujittumisen vaihetta, jossa järjestelmän negatiivisia puolia ei enää korostettu ja alettiin löytämään positiivisia puolia. Käyttöä myötä on siirrytty hyväksymisen tilaan, jossa tilanne nähdään normaalina.

### **8.6. Organisaation kulttuuri ja muutos**

Työskentelyn tyyli muuttui paperikirjaamisesta ja henkilöiden perässä juoksemisesta tietojärjestelmän käsittelyyn. Arvot ja asenteet eivät ole varausprosessia kohtaan varsinaisesti muuttuneet. Kaikkiin uusiin järjestelyihin suhtaudutaan kuitenkin aluksi melko varauksellisesti.

Kyky reagoida muutokseen on osittain riippuvainen organisaation kulttuurista. Kulttuurin tekijöiden tunnistaminen auttaa tunnistamaan muutoksen hallinnan menestystekijöitä. Shein [1992] mukaan organisaatiota voidaan luokitella olettamusten, arvojen ja perinteiden kautta.

Olettamukset ovat näkymättöminä peruselementteinä merkittävä haaste muutoksen hallinnassa. Lennonvarmistuksen henkilöstö on kokenut monia tietojärjestelmä uudistuksia vuosien aikana. Heille on muodostunut tietyt käsitykset siitä, miten järjestelmäuudistukset etenevät. Usein järjestelmät on koettu otettavan puutteellisina käyttöön, koska käytön aikana niistä on löydetty puutteita ja vikoja, joita ei aina ole onnistuttu ennakoimaan tai löytämään mittavissakaan testeissä. Tästä saattaa syntyä pelkoja uuden järjestelmän käyttöä kohtaan. Näitä olettamuksia pitää pystyä poistamaan, jos ne muodostavat esteitä muutokselle.

Projektin henkilöstö teki runsaasti työtä järjestämällä esittelyjä ja niin sanottuja tutustumiskoulutuksia, jossa henkilö pääsi käyttämään järjestelmää ilman, että kyseessä olisi ollut varsinainen koulutustilaisuus. Muutosvastarintaa oli havaittavissa käytön jälkeen tietyn toimipisteen osalta, joka katsoi menettäneensä toimivaltaa muutoksen myötä. Kriittinen käyttöympäristö vaikuttaa myös osaltaan vastustamisen tapaan ja suuruuteen. Näkemykseni mukaan henkilöstön suurin pelko oli joutua työskentelemään epäluotettavalla ja vaikeasti opittavalla järjestelmällä. On olemassa pelko, että onnettomuuden sattuessa syyllistyy järjestelmän käyttäjä eikä virheellisesti toiminut järjestelmä. Työympäristön kulttuuri on sellainen, että henkilöstö pyrkii osaltaan vaikuttamaan siihen, että lentoliikenne voidaan hoitaa turvallisesti.

Arvojen ohjaamana suhtautuminen muutokseen on merkittävää kriittisissä ympäristöissä. Turvallisuusnäkökohdat ovat erittäin voimakkaita lennonvarmistuksen kulttuurissa ja siten niiden huomioiminen muutoksen suunnittelussa on tärkeää.

### 8.7. Arvio muutoksen hallinnasta

Muutoksen hallinnan toteutumista voidaan arvioida käyttäen erilaisia aikajaksoja. Käytän tässä arviossa neljää tarkasteltavaa ajanjaksoa, jotka ovat yliheittoa edeltäneet toimet, varsinainen yliheitto (noin vuorokausi käyttöä), toiminnan vakiintuminen (kaksi viikkoa käyttöä) ja kokonaisuunnistumisen arvio (noin vuosi käyttöä).

Edeltäneet toimet sujuivat hyvin, kun ottaa huomioon olosuhteet, jotka silloin vallitsivat. Valmistautuminen oli vaikeaa, koska ohjelmisto oli tuolloin puutteellinen. Siitä syystä työmenetelmät vaativat jatkuvaa muutosta aivan viimehetkille asti. Näistä puutteista johtuen koulutuksen toteutus oli vaikeaa, ja sitä jouduttiinkin paikkaamaan käyttöönnoton lähestyessä.

Varsinainen yliheitto sujui rauhallisesti ja ehkä jopa hieman odottavissa tunnelmissa. Käyttöönotto oli ilmatilavarausten osalta melko hiljaista aikaa. Tästä johtuen työtehtävät voitiin suorittaa aivan kaikessa rauhassa.

Toiminta ilmatilavarausten käsittelyssä vakiintui melko nopeasti. Vaikka ohjeisiin joutuikin turvautumaan melko usein, toiminta oli sujuvaa ja henkilöstö kykeni työskentelemään itsenäisesti. Satunnaisesti varmuutta haettiin tehdyille toimille kysymällä tukihenkilöiltä apua tai varmistusta.

Vaikka muutoksen hallinta kokonaisuutena oli suuri ja haastava projekti, sujui se operatiivisen toiminnan kannalta hyvin. Turvallisuutta vaarantaneita tilanteita ei päässyt syntymään. Henkilöstö sopeutui muutokseen ja työmenetelmät vakiintuivat yhtenäiseksi kokonaisuudeksi.

## 9. Tulokset

Tässä tutkielmassa käsiteltiin kriittisten tietojärjestelmien muutoksen hallintaa. Kriittiset järjestelmät eroavat muista tietojärjestelmistä erityisesti siinä, että yksikin järjestelmän virheellinen toiminta voi aiheuttaa menetyksen, jota ei voida sallia. Päällimmäisenä oli kysymys turvallisuuden varmistamisesta muutoksen hallintaan liittyvissä tilanteissa. Tällaisia tilanteita ovat järjestelmien käyttöönotot, ohjelmaversioiden vaihdokset tai työskentelymenetelmien muutokset. Tutkimuksen avulla on löydettävissä useita näkökulmia ja keinoja, joiden avulla järjestelmien turvallista käyttöä voidaan edistää ja varmistua niiden oikeasta, ennakoitavasta toiminnasta.

Tutkimuksessa tietoja kerättiin sekä kirjallisuudesta että toteutuneesta kriittisen tietojärjestelmän käyttöönotosta. Niitä tarkastelemalla on selvästi havaittavissa, että kirjallisuuden tiedot keskittyvät suurelta osin ohjelmistojen suunnitteluprosessiin ja muutoksen valmisteluun. Muutoksen toteuttamisesta ja sen vaikutusten seuraamisesta ei kirjallista tutkimustietoa ollut juuri lainkaan. Tältä osin case:n tutkiminen ja siinä käytettyjen menetelmien kuvaaminen täydentää tutkimuksen tuloksia.

Vaikka varsinaisia ristiriitaisuuksia ei kirjallisuuden ja case:n välillä ollut löydettävissä, asioiden painottamisessa ja toimintatavoissa oli eroja. Kirjallisuuden tutkimustiedot painottavat suunnittelun merkitystä, eikä turhaan, sillä suurin osa vioista on nimenomaan suunnitteluvirheitä. Toisin kuin kirjallisuudessa, käytännön case:ssa pyritään turvallisuus varmistamaan ohjelmiston toimintavarmuuden parantamisen sijaan etsimällä keinoja havaita järjestelmän virheellinen toiminta ja luomalla varamenetelmiä, joilla toimintaa voidaan jatkaa virheestä huolimatta. Yhteistä molemmille oli, että turvallisuutta vaarantavia tilanteita ei sallita, vaikka niiden esiintymisen todennäköisyys olisikin pieni.

Case:ssa vaaratilanteiden ennakointi keskittyi miltei kokonaan järjestelmän sisäisten ongelmien ennakointiin. Kirjallisten lähteiden avulla löytyi keinoja, joilla ennakoinnissa voidaan ottaa myös ympäristön vaikutukset huomioon. Ongelmat voivat tutkimustiedon perusteella olla seurausta ympäristötekijöistä tai organisaatiotason asioista, jotka mahdollistavat ongelmatilanteet mahdollistavat ympäristötekijät.

Käyttöturvallisuus voidaan saavuttaa käyttämällä sopivassa suhteessa erityyppisiä menetelmiä ja tarkastelemalla niitä eri näkökulmista. Tässä tutkimuksessa ei vertailtu menetelmiä keskenään. Vertaamalla kirjallisuuden keinoja case:n avulla tehtyihin havaintoihin oli mahdollista tehdä toisiaan

täydentävä lista toimista, joilla turvallisesta kriittisten järjestelmien muutoksesta voidaan varmistua. Menetelmien soveltuvuus tulee arvioida erikseen tapauskohtaisesti. Soveltuvien menetelmien valinta ja niiden käyttö oikeassa laajuudessa on avainasemassa turvallisuuden varmistamisessa. Tutkimustiedon mukaan yli puolet onnettomuuksia aiheuttavista juurisivistä löytyy suunnitteluvaiheesta. Tällä alueella parannusta voitaisiin saavuttaa poistamalla tai minimoimalla väärinymmärrykset operaattorien ja suunnittelijoiden välillä sekä tunnistamalla suunnittelussa aiemmin tapahtuneiden virheiden perustyyppit.

Vikoja tutkittaessa niiden aiheuttajaksi usein selviää käyttäjän virheellinen toiminta. Vaikka käyttäjä olisikin toiminut virheellisesti, varsinainen syy voi löytyä poikkeavista olosuhteista käyttöympäristössä tai niin sanotuista organisaatiotason asioista, jotka ovat mahdollistaneet poikkeavien olosuhteiden syntymisen.

Toisin kuin yleisesti saatetaan luulla, järjestelmien oikean toiminnan varmistamisessa ei ole kyse vain ohjelmistojen testauksesta. Huolellisella testauksella voidaan löytää koodausvirheet eli ne tilanteet, jolloin ohjelmisto toimii eri tavalla kuin on määritelty. Turvallisuuden varmistamisen kannalta tämä on kuitenkin riittämätöntä, koska ohjelmistojen virheellinen toiminta johtuu usein virheistä määrittelyssä.

Arvioitaessa ohjelmistojen toimintaa on mahdollista määrittellä kriittiset tilat, jotka voivat aiheuttaa vaarallisen tilanteen. Ohjelmistolta voidaan vaatia, että tällaiseen tilan meneminen tulee estää tai se tulee kiertää.

Tutkimustiedon perusteella onnettomuudet ovat usein syntyneet yksittäisten tai rinnakkaisten tapahtumaketjujen seurauksena. Mikäli tällaiset ketjut on löydettävissä ennalta, tarjoavat ne monia tapoja tunnistaa ja siten estää niiden syntymisen.

Käytännön tilanteissa ongelmien tunnistaminen on merkittävin asia. Kun ongelma voidaan tunnistaa, voidaan järjestelmän käyttöä jatkaa joissain tilanteissa virheet huomioiden. Vikojen tunnistaminen mahdollistaa turvalliseen toimintamalliin siirtymisen, silloin kun menetelmät on etukäteen suunniteltu, ohjeistettu ja koulutettu. Samalla tavoin voidaan rajoitetusta toimintamallista taas siirtyä takaisin normaalitoimintaan, kun viat on rajattu ja korjattu.

Seuraavassa on tutkimuksen tuloksena saatu lista niistä käytännön toimista ja näkökulmista, jotka tulisi ottaa huomioon suunniteltaessa kriittisiä tietojärjestelmiä, niiden muutoksia ja niiden saattamista operatiiviseen käyttöön.

1. Käyttövarmuuden vaatimukset tulee asettaa seuraaville osa-alueille:
  - a. luotettavuus,
  - b. saatavuus,
  - c. käyttöturvallisuus,
  - d. luottamuksellisuus,
  - e. eheys ja
  - f. ylläpidettävyys.
2. Vaatimusten täytyminen tulee todentaa ja niiden jäljitettävyydestä on huolehdittava. Keinoja ovat
  - a. testaaminen ja koekäyttö oikealla datalla oikeassa ympäristössä,
  - b. aiempien luotettavuus- tai vikahistoriatietojen tarkastelu,
  - c. kokonaisuuksien mittaaminen,
  - d. muodollisten tapojen käyttö oikean toiminnan osoittamisessa ja
  - e. kaikkien mahdollisten ongelmatilanteiden tarkastelu ennalta ja korjaavien toimintamallien suunnittelu.
3. Kriittisyys ja muutos tulee tarkastella perusteellisesti. Niihin liittyvät yleiset ongelmakohdat ja vaatimukset tulee ottaa huomioon.
  - a. turvallisuuskriittiset, tehtäväkriittiset, talouskriittiset
  - b. jatkuva tai yksittäinen muutos
  - c. muutoskohteet: ohjelmisto ja organisaatiotaso
  - d. liiketoimintaprosessien uudelleenjärjestäminen, parantaminen, automatisointi
  - e. muutos yksilön näkökulmasta
  - f. muutos organisaation näkökulmasta
  - g. muutoksen vaiheet
  - h. määräysten noudattaminen ja vaatimusten täyttäminen
4. Ongelmiin tulee varautua ennakolta. Tehtäviä toimia ovat
  - a. menetysten suuruuden arviointi,
  - b. järjestelmävikaan mahdollisesti johtavien vikojen tunnistaminen ja varautuminen,
  - c. estämiseksi tulee tunnistaa vikojen aiheuttajat,
  - d. ongelmien luokittelu,
  - e. ongelmien havaitsemisen varmistaminen,
  - f. korjaavien toimien suunnittelu,
  - g. toimien suunnittelu normaalitoimintaan palaamiseksi,

- h. vaaran poistaminen, vähentäminen ja hallitseminen,
  - i. ongelmista syntyvien vahinkojen vähentäminen ja
  - j. operatiivisen ja teknisen riskianalyysin laadinta.
5. Ongelmien aiheuttajien kartoittaminen tulee tehdä eri osa-alueilta. Niitä ovat
- a. suunnitteluvirheet, kulumiset,
  - b. tapahtumaketjut ja tapahtumaketjujen suhteet,
  - c. virhetoiminto (johtuu esim. käyttäjästä tai mittavirhe) ja
  - d. organisaatiotason ja ympäristön syyt ja myötävaikuttajat.
6. Tietoturvallisuus tulee huomioida kaikilla sektoreilla. Osa-alueita ovat:
- a. tekninen turvallisuus,
  - b. fyysinen turvallisuus ja
  - c. toiminnallinen turvallisuus.
7. Muutoksen toteuttamisen tehtävät tulee suunnitella ennalta. Tehtäviä voivat olla esimerkiksi:
- a. tarvittavien lupien hankkiminen ja tiedotus,
  - b. järjestelmän tarkastaminen ennen käyttöönottoa,
  - c. osallistujien osaamistason varmistaminen,
  - d. käyttöönottosuunnitelma ja toteutuksen dokumentointi,
  - e. vastuuhenkilöiden määrittäminen ja johtaminen,
  - f. varamenetelmät ja muutoksen palauttaminen,
  - g. tuotantomäärien laskeminen yliheitossa ja
  - h. tukihenkilöt (operatiiviset ja tekniset).
8. Muutoksen jälkeiset toimenpiteet tulee suunnitella ennalta. Tehtäviä voivat olla esimerkiksi:
- a. järjestelmän oikeasta toiminnasta varmistuminen ja toiminnan seuranta,
  - b. tehtyjen toimien riittävyden tarkastelu,
  - c. tuotantomäärien hallittu nostaminen ja
  - d. jatkotoimien tarpeen arviointi.

## 10. Yhteenveto ja suositukset

Tutkimus antoi kuvauksen kriittisen järjestelmän muutoksen hallinnasta. Tutkielmassa käsiteltiin ongelmien aiheutumista ja niihin reagoitua. Pääkysymyksenä pohdittiin miten ohjelmiston turvallisesta käytöstä voidaan varmistua. Toiminnallista prosessia arvioitiin kuvauksen lisäksi kirjallisten lähteiden avulla löydetyn tutkimustiedon perusteella. Tuloksena saatiin joukko näkökulmia ja toimia, jotka tulee ottaa huomioon kriittisen järjestelmän muutosprosessin aikana.

Yleiset periaatteet ovat käytettävissä minkä tahansa kriittisen järjestelmän muutoksen suunnittelussa ja toteutuksessa. Kuvatut seikat eivät yksistään kuitenkaan riitä, vaan todellinen käyttöympäristö erityisvaatimuksineen tulee aina huomioida erityistä huolellisuutta noudattaen. Kriittisissä käyttöympäristöissä turvallisuuden mittaaminen on hankalaa, koska jo yksi vika voi aiheuttaa sellaisia menetyksiä, joita ei voida sallia. Jatkotutkimuksena olisikin haasteellista löytää luotettavia mittareita, joilla voitaisiin varmistua kriittisten järjestelmien turvallisesta käytöstä.

## Viiteluettelo

- [Abbot, 1990] Russel J. Abbot, Resourceful Systems for Fault Tolerance, Reliability, and Safety. *ACM Computing Surveys* **22**, 1 (1990), 35-68.
- [Adams et al. ,1976] J. Adams, J. Hayes, B. Hopson, Transitions: Understanding and managing personal change. Martin Robertson, London, 1976.
- [AMC-Tool Turvallisuudenvarmistusasiakirja, 2004] Turvallisuudenvarmistusasiakirja, AMC-Tool Safety Case versio 1.2, Etelä-Suomen lennonvarmistuskeskuksen asiakirja, 2004.
- [AMC-varaohje, 2005] Etelä-Suomen lennonvarmistuskeskuksen AMC-varaohje, Osa: amcvara, versio 2, 2.5.2005.
- [Boddy et al., 2001] D Boddy, A Boonstra, G Kennedy, Managing the Information Revolution An Organisational Perspective, Financial Times Prentice Hall, Harlow, Uk, 2001.
- [Butler & Finelli, 1991] Ricky W. Butler, George B. Finelli, The Infeasibility of Quantification of Life-Critical Software Reliability. Technical Report: Nasa-91-acm-rwb, 1991.
- [Chaffey et. al., 2005] Dave Chaffey & Steve Wood, Business information management, Improving performance using information systems, 2005, s. 385-400.
- [Doherty et. al. 1999] N. Doherty, C. Marples, A. Suhaimi, The relative success of alternative approaches to strategic information systems planing: an empirical analysis, *International Journal of Information Management*, s. 263, 1999.
- [ESARR 6, 2003] Eurocontrol safety regulatory requirement, ESARR 6, Software in ATM systems, edition 1.0, 6 Novenber 2003. <http://www.eurocontrol.int/src>.
- [Hallinnon kehittäminen, tietoturvasanasto] Hallinnon kehittäminen, tietoturvasanasto, <http://www.vm.fi/tietoturvasanasto/sisallys.htm> (27.2.2005)
- [Haikala ja Märajärvi, 2002] Ilkka Haikala ja Jukka Märajärvi, Ohjelmistotuotanto, Talentum Media Oy, 8. painos, 2002.
- [Hammer, 1972] Willie Hammer, Handbook of System and Product Safety. Prentice-Hall, Englewoods Cliffs, N.J., 1972.
- [Hammer et al, 1993] Michael Hammer and James Champy, Reengineering the Corporation, Harper Business, New York, 1993.
- [Ilmailumääräys, ANS M1-3, 2004] Ilmailukenteen hallintapalvelun tekninen henkilöstö, Ilmailumääräys ANS M1-3 17, Lentoturvallisuushallinto, 17.12.2004.

- [Johnson, 1973] W. G. Johnson, Management oversight and risk tree, 1973.
- [Järvinen ja Järvinen, 2000] Pertti Järvinen ja Annikki Järvinen, Tutkimustyön metodeista. Opinpajan kirja, 2000.
- [Knight, 2002] John C. Knight, Safety Critical Systems: Challenges and Directions. Proceedings of the 24<sup>th</sup> International Conference on Software Engineering, State-of-the art presentations (2002), 547-550.
- [Knight, 2003] John C. Knight, Computing Systems Dependability. Proceedings of the 25<sup>th</sup> International Conference on Software Engineering, One and two-day tutorials (2003), 742-743.
- [Koulutusraportti, 2004] Lennonjohtoapulaisten FATMI-koulutuksen kooste 26.5.2004, Etelä-Suomen lennonvarmistuskeskuksen asiakirja, 2004.
- [Käyttöönottohyväksyntä GEN130.01, 2001] Käyttöönottohyväksyntä GEN130.01, Lennonvarmistuslaitteiden tekninen ohjeisto, Ilmailulaitoksen asiakirja, 2001.
- [Leveson, 1990] Nancy G. Leveson, Evaluation of Software Safety, 1990.
- [Leveson, 1995] Nancy G. Leveson, Safeware, System Safety and Computers. Addison-Wesley, 1995.
- [Leveson, 1986] Nancy G. Leveson, Why, What, and How. ACM Computing Surveys **18**, 2 (1986), 125-163.
- [Lewycky, 1987] Peter Lewycky, Notes toward an understanding of accident causes, Hazard prevention (March/April): 6-8, 1987.
- [Lutz, 2000] Robyn R. Lutz, Software Engineering for Safety: A Roadmap. Proceedings of the Conference on The future of Software Engineering, (2000) 213-226.
- [Malasky, 1982] S. W. Malasky, System Safety Technology and Application. Garland STPM Press, New York, 1982.
- [Nadler et al., 1995] D. Nadler, R. Shaw, E. Walton, Discontinuous Change, Jossey-Bass, San Francisco, 1995, 24.
- [Parnas et al., 1990] David L. Parnas, A. John van Schouwen, and Shu Po Kwan, Evaluation of safety-critical Software. Communications of the ACM **33**, 6 (June 1990) 636-648.
- [Petersen, 1971] D. Petersen, Techniques of Safety Management, McGraw-Hill, New York, 1971.
- [Ridley, 1983] John Ridley, Safety at Work. Butterworths, London, 1983.
- [Roelen et al., 2004] Alfred Roelen, Steve Kinnersly, Fabrice Drogoul, Review of root causes of accidents due to design, Eurocontrol Experimental Centre, EEC Note No. 14/04, 2004.

- [Safety in air navigation, Eurocontrol] Safety in air navigation, Eurocontrol, [http://www.eurocontrol.int/corporate/public/standard\\_page/cb\\_safety.html](http://www.eurocontrol.int/corporate/public/standard_page/cb_safety.html) (24.11.2005).
- [Safety regulation commission, Eurocontrol] Safety regulation commission, Eurocontrol, [http://www.eurocontrol.int/src/public/subsite\\_homepage/homepage.html](http://www.eurocontrol.int/src/public/subsite_homepage/homepage.html) (24.11.2005).
- [Schein, 1992] Edgar Schein, *Organizational Culture and Leadership*. Jossey-Bass, San Francisco, 1992.
- [Sommerville, 2000] Ian Sommerville, *Software Engineering* (6th Edition). Addison-Wesley, 2000, 6. painos.
- [Uusitalo, 2001] Hannu Uusitalo, *Tiede, tutkimus ja tutkielma, Johdatus tutkielman maailmaan*, WSOY, 2000, 7. painos.
- [Waterman et. al., 1980] R. J. Waterman, T. J. Peters, J. R. Phillips, *Structure is not organization*, McKinsey Quarterly. In-house journal. McKinsey & Co., New York, 1980.
- [Yin , 1989] R. K. Yin, *Case study research: Design and methods*, Sage Publ., Beverly Hills Ca, 1989.

Risk Assessment and Mitigation in AMT- ESARR 4

Safety Regulation Commission  
 Safety Regulatory Requirement - ESARR 4  
 Risk assessment and mitigation in ATM

Severity Class	1 [Most Severe]	2	3	4	5 No safety effect [Least Severe]
Effect on Operations*)	Accidents	Serious incidents	Major incidents	Significant incidents	No immediate effect on safety
Examples of effects on operations include *):	<ul style="list-style-type: none"> <li><input type="checkbox"/> one or more catastrophic accidents,</li> <li><input type="checkbox"/> one or more mid-air collisions</li> <li><input type="checkbox"/> one or more collisions on the ground between two aircraft</li> <li><input type="checkbox"/> one or more Controlled Flight into Terrain</li> <li><input type="checkbox"/> total loss of flight control.</li> </ul> <p>No independent source of recovery mechanism, such as surveillance or ATC and/or flight crew procedures can reasonably be expected to prevent the accident(s).</p>	<ul style="list-style-type: none"> <li><input type="checkbox"/> large reduction in separation (e.g., a separation of less than half the separation minima), without crew or ATC fully controlling the situation or able to recover from the situation.</li> <li><input type="checkbox"/> one or more aircraft deviating from their intended clearance, so that abrupt manoeuvre is required to avoid collision with another aircraft or with terrain (or when an avoidance action would be appropriate).</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> large reduction (e.g., a separation of less than half the separation minima) in separation with crew or ATC controlling the situation and able to recover from the situation.</li> <li><input type="checkbox"/> minor reduction (e.g., a separation of more than half the separation minima) in separation without crew or ATC fully controlling the situation, hence jeopardising the ability to recover from the situation (without the use of collision or terrain avoidance manoeuvres).</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> increasing workload of the air traffic controller or aircraft flight crew, or slightly degrading the functional capability of the enabling CNS system.</li> <li><input type="checkbox"/> minor reduction (e.g., a separation of more than half the separation minima) in separation with crew or ATC controlling the situation and fully able to recover from the situation.</li> </ul>	<p>No hazardous condition i.e. no immediate direct or indirect impact on the operations.</p>

FIG. A-1: Severity Classification Scheme in ATM

Note: The worst credible effect in the environment of operations determines the severity class.  
 \*:- The Severity Classification of effects is common to that in ESARR 2 but the examples chosen relate to a priori assessment. This list is by no means exhaustive.