

RAM-ANALYYSI

käytännön näkökulmia

RAM- eli luotettavuusanalyysi on tehokas työkalu esimerkiksi kun suunnitteluvaiheessa tutkitaan vaihtoehtoisia järjestelmärakenteita ja niiden vaikutusta käyttövarmuuteen. Se sopii erinomaisesti myös kunnossapito-ohjelman räätälöintiin järjestelmän kriittisten osien luotettavuuden mukaan, järjestelmän omistamisesta aiheutuvien kustannusten etukäteen arviointiin ja kun suunnitellaan tai arvioidaan huolto- ja ylläpitosopimuksia ja sidotaan ne saavutettavaan käyttövarmuuteen. Lyhenne RAM tulee sanoista Reliability (toimintavarmuus), Availability (käyttövarmuus) ja Maintainability (kunnossapidettävyyden). RAM-malli on aina perustana myös RCM- ja LCC-analyysille.

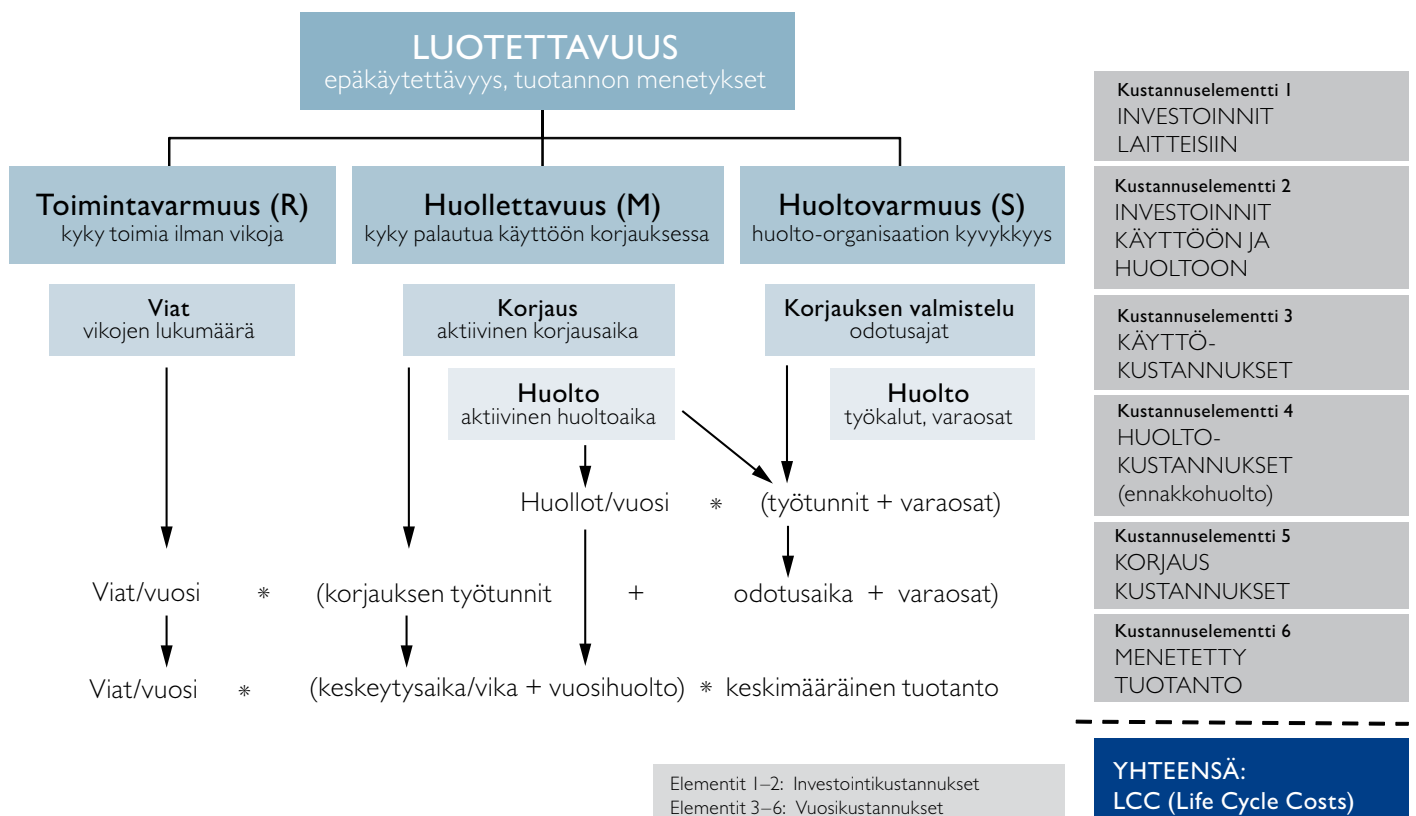
ANTTI LYTTIKÄINEN
dipl.ins.
Suomen
luotettavuusakatemia
info@
reliabilityacademy.fi



Yksinkertaisimmillaan RAM-mallinnusprojekti on seuraavanlainen:

1. Tehdään lohkokaavioesitys järjestelmästä.
2. Laaditaan lohkokaavioesityksen avulla luotettavuusmalli esimerkiksi Exceliin tai mallinnusohjelmaan.

3. Hankitaan vikadataa (RAM-dataa) ja dokumentoidaan se.
4. Identifioidaan mallin avulla vaihtoehtoisia tapoja parantaa luotettavuutta
5. Lasketaan RAM-mallilla parannusehdotusten vaikutus tuotantoon ja



KUVA 1. Luotettavuus ja kustannukset.

kunnossapitokustannuksiin.

6. Tehdään luotettavuuden parannussuunnitelma.

Luotettavuusmalli

RAM-malli havainnollistuu parhaiten lohkokaaavion muodossa, jonka osiin eli lohkoihin tai alilohkoihin liitetään vastaava luotettavuusdata. Luotettavuusdatan voi merkitä näkyviin jo itse

graafiseen lohkokaavioon, jolloin mallin oikeellisuus on helppo muidenkin kuin luotettavuusinsinöörin nähtävillä, todettavissa ja päivitettävissä.

Lohkokaavio(t) on syytä hallita dokumentointijärjestelmässä, jolla varmistetaan se, että on aina selvää, mihin kohdejärjestelmän versioon kukin dokumenttiversio liittyy. Itse lohkokaaviossa RAM-mallin kukin lohko ja osa-

lohko on identifioitava selkeällä tunnuksella, joka koostuu kirjainosasta ja numero-osasta. Tähän voi soveltaa esimerkiksi prosessikaaviosta löytyvä tunnuksia.

Luotettavuusmallin voi laatia yksinkertaisimmillaan esimerkiksi Exceliin, jos ei halua käyttää kalliita ja aikaavieviä mallinusohejelmia. Tällöin Exceliin laitetaan samalle riville

- kunkin lohkon tunnus
- vikataajuus (vikaa per aikayksikkö)
- korjausaika
- korjauksen odotusaika
- vikadatan lähde ja päivämäärä.

Vikadata

Usein vika- eli luotettavuusdatan puute on kynnyskysymys mallia laadittaessa. Vikadataa voi saada mallinnusta varten erilaisista lähteistä:

Yksi mahdollisuus on analysoida saatavilla olevan vastaavan järjestelmän vikaistoria ja muodostamaa sen avulla estimaatti kunkin komponentin vikataajuudelle ja korjausajalle. Tämä on kuitenkin työläs tapa, koska aiemmin kerätty vikaistoria on yleensä hyödyttömässä muodossa. Samoin vähäinen vikadata on usein hyvin vaikea kohdistaa oikeille komponenteille

Vikatietopankkeja ja geneeristä vikadataa on saatavilla datakirjojen muodossa mm. voimalaitoksista (esimerkiksi EPRI) ja off-shore sovelluksista (esimerkiksi OREDA) sekä elektronika. Vikadata on näissä yleensä hyvin dokumentoitu ja perusteltu ja niitä kannattaa käyttää jos voi vedota komponenttien samankaltaisuuteen.

Vikadatan keräys haastatelmalla on tehokas ja luottamusta herättävä metodi varsinkin mekaanisten komponenttien ja prosessilaitteiden osalta. Vikadatan keräykseen on tosin varattava aikaa kunnolla. Varsinkin korjausaikaan ja korjauksen odo-

tukseen liittyvät parametrit ovat luotettavia jos haastateltavana on kokeita kunnossapitäjiä. Luotettavuusmatemaattisesti tällöin epäkäytettävyyden toinen termi, eli MTTR, on hallinnassa.

Myös vikataajuus z saadaan kohtalaisen luotettavasti, kun tiedetään laitteiden populaatiojärjestelmässä. Mallin tulokset on helppo hyväksyä, jos on itse ollut arvioimassa vikadataa.

Vakiovikataajuusoletus on usein riittävä ja myös matemaattisesti perustellu vaihtoehto. Se on lisäksi helppo arvioida kun vastataan kysymykseen: Kuinka monta vikaa laitteessa on ollut esimerkiksi periodilla 1 vuosi, 5 vuotta tai 10 vuotta?

Muun kuin vakiovikataajuuden käyttö vaatisi monimutkaista simulointiohjelmistoa laskennassa. Näiden ohjelmien heikkous on siinä, että uskottava vikatiheysfunktio vaatisi minimissään noin viisi datapistettä, joita voi olla hyvin vaikea saada mistään aineistosta.

Lisäksi järjestelmien summa- vikataajuus lähestyy nopeasti vakiovikataajuutta, olipa komponenttien vikatiheysfunktio mikä tahansa. Ohjelmistojen hankkiminen ja opettelu on myös kallista, joten niistä saatava lisäarvo voi olla vähäinen. Ohjelmistoissakin pätee vanha insinöörin viisaus: ”garbage in – garbage out”, vaikka kaikki näyttää hienolta.

Luotettavuuden parantaminen

Kun taulukkolaskentamalli kaavoineen on valmis, voidaan sen avulla tarkasteluihin erilaisia asioita, esimerkiksi

- millä komponenteilla on suurin epäkäytettävyyden (kriittisimmät komponentit)?
- millä komponenttiryhmillä on suurin epäkäytettävyyden?
- millä osajärjestelmillä on suurin epäkäytettävyyden?

RAM-laskennan perusteita

RAM-laskenta noudattaa luotettavuustekniikan peruskaavoja.

$$U = z \times \text{MTTR} \quad (1)$$

missä U on epäkäytettävyys (Unavailability), z on vikataajuus (esim. vikaa per tunti) ja MTTR on keskimääräinen korjausaika (tuntia). Korjausajassa on mukana sekä varsinainen korjausaika että odotusaika.

Sarjajärjestelmät ovat yleisin luotettavuusrakenteen muoto, ja usein järjestelmä koostuukin jo kustannussyistä pelkästään sarjassa olevista laitteista tai komponenteista. Sarjajärjestelmän käytettävyyden A (Availability) ja epäkäytettävyys U noudattavat seuraavia yksinkertaisia kaavoja.

$$A = 1 - U \quad (\%) \quad (2)$$

$$A_{\text{sarja}} = A_1 \times A_2 \times \dots \times A_n \quad (3)$$

$$U_{\text{sarja}} = 1 - [(1-U_1) \times (1-U_2) \times \dots \times (1-U_n)] \quad (4)$$

Tämä redusoituu pienillä arvoilla muotoon:

$$U_{\text{sarja}} = U_1 + U_2 + \dots + U_n \quad (5)$$

Tähän voidaan sijoittaa kaava 1, ja sitten siirtää laskutoimitukset taulukkolaskentaohjelmaan. Kunkin osajärjestelmän epäkäytettävyyden summataan sen lohkojen summana erikseen omalle riville ja koko järjestelmän epäkäytettävyyden taulukon viimeiselle riville.

Rinnakkaisrakenteet parantavat järjestelmien käytettävyyttä, koska vian sattuessa toinen järjestelmä hoitaa tarvittavat tehtävät. Yhtäaikainen vikaantuminen on jo matemaattisesti erittäin epätodennäköistä ja siksi rinnakkaisrakenteen laskenta ei välttämättä ole konaisuuden kannalta tärkeä mallinnuskohde, ellei sitten haluta muuttaa sitä sarjarakenteeksi, jolloin säästetään tuplajärjestelmän kustannuksia. Yleisimpään tupla-rinnakkaisrakenteeseen liittyvät seuraavat kaavat:

$$A_{\text{rinnan}} = 1 - [(1-A_1) \times (1-A_2) \times \dots \times (1-A_n)] \quad (6)$$

joka redusoituu muotoon

$$U_{\text{rinnan}} = U_1 \times U_2 \times \dots \times U_n \quad (7)$$

vyys (=summaa osajärjestelmien lohkot)?

Mallissa kannattaa myös vaihdella kriittisten osajärjestelmien ja komponenttien vikataajuutta z ja korjausaikaa MTTR, ja katsoa, onko niillä merkittävää vaikutusta lopputuloksiin. Kannattaa myös tutkia, mistä kriittisimpien kohteiden epäluotettavuus johtuu, eli onko vikataajuus liian suuri vai korjausaika liian pitkä.

Luotettavuus ja kustannukset

Luotettavuusmalliin voidaan myös lisätä kustannuselementtejä, kuten korjaustyöajan kustannus, varaosat ja tarvikkeet sekä menetetyt tuotannon kustannus tai korvaavan tuotannon kustannus.

KUVASSA 1 on esitetty laajemman, ns. LCC- eli elinjaksokustannusanalyysin periaate. Se yhdistää luotettavuuden ja kustannuslaskennan. LCC:tä käytetään erityisesti suurten järjestelmien hankintavaihtoehtojen vertailussa ja sopimusten teossa.

Parannussuunnitelma

RAM-parannussuunnitelmassa kannattaa keskittyä mm seuraaviin asioihin:

- osajärjestelmät, jotka eivät täytä luotettavuusvaatimuksia.
- kriittisimmät komponentit: hanki parempilaatuksia tai pienennä korjausaikaa
- harkitse redundanssin eli varmennusten lisäämistä ja laske vaikutus. Tuplajärjestelmä tai -komponentti voi olla liian kallis.

Huomaa, että olemassa olevan järjestelmän RAM-mallintaminen ei ole välttämättä hyödyllistä, koska järjestelmämuutokset ovat kalliita. Suunnitteilla olevan järjestelmän muutokset voidaan sen sijaan tehdä paperilla.

Yhteenveto

RAM-mallinnus oikein käytettynä ja oikeaan aikaan tehtynä tuo paljon hyödyllistä lisätietoa järjestelmän kustannusrakenteesta ja tulevista tuotannonmentyksistä. Yksinkertaisellakin mallilla

ja luotettavuuden peruskaavoilla voidaan vertailla järjestelmiä ja järjestelmärakenteita keskenään. Mallinnuksessa on tärkeää dokumentoida kunnolla sekä RAM-malli että siinä käytetyt datalähteet.

Yksinkertainen malli voidaan tarvittaessa myöhemmin syöttää vaativampaan simulointi- tai mallinnuohjelmaan, jos katsotaan että sillä saadaan lisäarvoa.

RAM-analyysin vakavia puutteita ovat mm. seuraavat:

- Analyysi perustuu pelkästään todennäköisyyspohjaiseen vikojen esiintymiseen (satunnaisviat).
- Yhteisvikojen huomiointi ja arviointi on vaikeaa koska dataa ei ole saatavilla. Yhteisvikoja on kuitenkin olemassa, sellaisia ovat esimerkiksi kunnossapitovirheet, valmistusvirheet jne, jotka kohdistuvat samanlaisiin laitteisiin tai komponentteihin
- Systemaattiset viat kuten suunnitteluvirheet aiheuttavat todellisuudessa pal-

jon epäkäytettävyyttä varsinkin uudessa teknologiassa. Usein väitetäänkin että ”se ei ole vika vaan ominaisuus”.

- Järjestelmissä ohjelmistojen osuus lisääntyy kovaa vauhtia. Ohjelmistovirheet ja -viat eivät ole todennäköisyyspohjaisia.

Järjestelmien RAM-mallinuksessa on tullut joskus esiin seikkoja, jotka saattavat yllättävästi vähentää RAM-mallin merkitystä. Aikoinaan esimerkiksi paperikoneen luotettavuusprojektissa selvitettiin myös, että noin 50 % kaikista paperiradan katkoksisista johtui paperin laatuviirheistä, joiden alkuperää ei voitu kohdistaa mihinkään RAM-mallin lohkoon.

RAM-mallin täydentäminen esimerkiksi Hazop-analyysillä voi kuitenkin tuoda uusia parannusehdotuksia erilaisista prosessihäiriöistä johtuvaan epäkäytettävyyteen. Siksi RAM-insinöörin tulisi hallita sekä luotettavuustekniikkaa että riskianalyysyjä. ■