

TAMPEREEN AMMATTIKORKEAKOULU  
Kone- ja tuotantotekniikka  
Lentokonetekniikka

Tutkintotyö

Miikka Sohlman

**MUUTOSHALLINAN SEURATTAVUUDEN KEHITTÄMINEN**

Työn ohjaaja  
Työn teettäjä  
Tampere 2010

lehtori Simo Marjamäki  
Patria Aerostructures Oy,  
valvojana konfiguraatiopäällikkö Minna Innala

## TAMPEREEN AMMATTIKORKEAKOULU

Kone- ja tuotantotekniikka

Lentokonetekniikka

Sohlman, Miikka

Tutkintotyö

Työn ohjaaja

Työn teettäjä

Muutoshallinnan seurattavuuden kehittäminen

31 sivua + 31 liitesivua

lehtori Simo Marjamäki

Patria Aerostructures Oy,

valvojana konfiguraatiopäällikkö Minna Innala

Joulukuu 2010

Hakusanat

tuotetiedonhallinta, muutoshallinta, konfiguraatio, seuranta

## TIIVISTELMÄ

Patria Aerostructures Oy:llä (AST) tuotteisiin tai toimintatapoihin liittyvät muutokset käsitellään muutoshallintaryhmissä eli CCB:ssä (*Change Control Board*). Muutosten läpiviemiseksi tehtäviä jaetaan vastuuhenkilöille. Vastuuhenkilöt kuittaavat tehdyt tehtävät Enovia-tuotetiedonhallintajärjestelmän avulla.

Menettelytapa, jolla muutoslomakkeiden tehtäviä kuitataan ja niitä seurataan Enovian kautta, on ollut monimutkainen ja aikaa vievä. Siksi ongelmaksi on muodostunut epävarmuus siitä, onko tehtäviä lainkaan suoritettu vai onko ne tehty, mutta jätetty kuittaamatta.

Tämän tutkintotyön tekijä on laatinut Kuittauksen seuranta -nimisen Excel-sovelluksen, joka toimii muutoshallintaryhmän (CCB), konfiguraatioinsinöörien, hankepäälliköiden ja kaikkien muutoksista vastaavien apuna, kun varmistetaan muutosten toteutuminen sovitulla tavalla.

Ohjelma toteutettiin Excel-taulukkolaskentaohjelmalla ja makroilla, jotka on kirjoitettu VBA (*Visual Basic for Applications*) -ohjelmointikielellä.

Kuittausten seuranta -ohjelma on otettu käyttöön AST:ssä. Sen avulla vastuuhenkilöt näkevät yhdelle silmäyksellä omat tehtävänsä sekä sen, onko ne hoidettu ajallaan. Kuittausten suorittaminen käy entiseen verrattuna nopeasti ja helposti. Lisäksi muutoshallintaryhmä näkee vaivattomasti muutosten tilan.

Ohjelmassa mahdollisesti esiintyvät puutteet tulevat esiin vasta käytön myötä. Jos niitä ilmenee, niin on sovittu, että tutkintotyön tekijä käy AST:n pyynnöstä muokkaamassa ohjelmaa näiden poistamiseksi.

TAMPERE POLYTECHNIC

Mechanical and Production Engineering

Aircraft engineering

Sohlman, Miikka

Developing the Follow-up of Change Management

Engineering Thesis

31 pages, 31 appendices

Thesis supervisor

lecturer Simo Marjamäki

Comissioning Company

Patria Aerostructures Oy.

Supervisor: konfiguration manager Minna Innala

December 2010

Keywords

product data management, change management, konfiguration, follow-up of changes

## ABSTRACT

In Patria Aerostructures (AST) changes to products or procedures are supervised by Change Control Boards (CCB). To make a change happen, CCB allocates tasks to a selected group of people. After completing their tasks, the people sign them to let the CCB know that their tasks are completed. The signing process is done via Enovia Product Data Management (PDM) software.

The follow up to who has signed and who hasn't has been difficult as Enovia is not a very elegant medium for that purpose.

In this study, an Excel application called "Kuittauksen seuranta" has been made that addresses the issues by providing a single excel sheet that summaries all the info about who has signed their tasks and who hasn't. The Kuittauksen seuranta contains many macros that have been coded with VBA (Visual Basic for Applications) programming language.

## SISÄLLYSLUETTELO

TIIVISTELMÄ .....	2
ABSTRACT.....	3
SISÄLLYSLUETTELO .....	4
1 JOHDANTO.....	5
2 PATRIA AEROSTRUCTURES OY.....	6
2.1 Patrian lentokoneteollisuuden historia .....	6
2.2 Patria Aerostructures Oy lyhyesti .....	7
3 TUOTETIEDONHALLINNAN KESKEISIÄ KÄSITTEITÄ.....	8
3.1 Tuotetieto .....	8
3.2 Tuotetiedonhallinta – PDM ( <i>Product Data Management</i> ) .....	8
3.3 Dokumentit.....	10
3.4 Tuotetiedonhallintajärjestelmä.....	10
3.5 Järjestelmäarkkitehtuuri .....	11
3.6 Tuoterakenne.....	13
4 MUUTOSHALLINTA PATRIA AEROSTRUCTURES (AST) OY:SSÄ.....	14
4.1 <i>Baseline</i> -käsite .....	14
4.2 Muutuskäsite .....	14
4.3 Konfiguraatiohallinta .....	15
4.4 Muutosten toteutumisen seuranta.....	15
4.5 Muutosprosessikaavio .....	16
5 MUUTOSHALLINNAN KÄYTÄNNÖN ONGELMA JA RATKAISU.....	17
5.1 Tehtävien etenemisen seuraamisen hankaluus Enovialla.....	17
5.2 Ongelman ratkaisu.....	18
6 KUINKA OHJELMA TOIMII.....	20
6.1 Muutoslomake.....	20
6.2 Kuittauksen Seuranta -taulukon rakentuminen .....	21
6.3 Infolaatikkomakro .....	28
7 LOPPUPÄÄTELMIÄ .....	29
LÄHTEET.....	30
LIITTEET .....	31

# 1 JOHDANTO

Nykymaailmassa tiedon määrä lisääntyy ja muutosten tahti kiihtyy. Työpaikoilla ja yrityksissä ihmiset yrittävät pysyä vauhdissa mukana ja ovat päivittäin yhä uudelleen saman haasteen edessä: miten hallita työhön liittyvää informaatiotulvaa.

Ongelman ratkaisemiseksi nykyaikainen tietotekniikka tarjoaa ratkaisuksi erilaisia tietokoneavusteisia järjestelmiä. Samalla, kun nämä järjestelmät ja ohjelmat tuovat helpotusta asioiden hallintaan, ne luovat uusia ongelmia. Niiden ymmärrettävyys ja käytettävyys eivät aina kohtaa käyttäjien tarpeita, jolloin ihmiset alkavat vieroksua niiden käyttöä ja mahdollisesti luovat omia menettelytapojaan selviytyäkseen työtehtävistään.

Tuotetiedonhallintajärjestelmät (PDM – Product Data Management) on kehitetty ratkaisemaan tuotteiden ja komponenttien lyheneviin elinkaariin liittyviä ongelmia. Patria Aerostructures Oy (AST) käyttää Enovia PDM-järjestelmää, ja on luonut tarkat menettelytavat muutoshallinnan varmistamiseksi.

Muutoshallinta on osa konfiguraatiohallintaa, eli suoraan tuotteeseen liittyvän tiedon hallintaa. Ilmailuteollisuudessa tuotteeseen liittyvän muutoshallinnan merkitys korostuu, koska ollaan tekemisissä lentoturvallisuuden kanssa. Siksi vähäisiltäkin tuntuvat muutokset on tunnistettava, suunniteltava ja toteutettava hallitusti muutoshallintaryhmän (CCB – *Change Control Board*) valvonnassa ja ohjauksessa.

AST:ssä jokaisesta muutoksesta avataan sen käsittelyvaiheessa muutoslomake, joka täydentyy muutosprosessin aikana siihen liittyvillä tehtävillä ja niiden vastuuhenkilöillä. Kun näiden tehtävien toteutumista seurataan Enovian avulla, törmätään käytännön ongelmaan. Laaja PDM-järjestelmä on hyvä ohjelmisto asiakokonaisuuksien hallintaan, muttei välttämättä helppokäyttöinen työkalu muutoslomakkeisiin merkittyjen tehtävien ja niiden kuittausten seurantaan.

Sain tutustua Patria Aerostructures Oy:n muutoshallintaan ollessani yrityksessä konfiguraatioinsinööriharjoittelijana v. 2008–2009. Samalla tuli tutuksi Enovian hankala käytettävyys, kuten kohdassa 5.1 on esitetty. Tämä ilmeni käytännössä yleisenä ongelmana, kun ihmiset eivät näyttäneet kuittaavan tehtäviään, mikä kasvatti CCB-statuslistalla avoinna olevien tehtävien luetteloa.

Syntyi ajatus luoda Microsoft Excel-taulukkolaskentaohjelmalla aivan oma Kuittausten seuranta -ohjelma, ja sen seurauksena muutoslomakkeet muutettiin Word-muodosta Excel-muotoon. Näin saatiin hyödynnettyä makroja ja tuloksena oli helppokäyttöinen ohjelma, josta näkee yhdellä silmäyksellä vastuuhenkilöt ja tehtävien tilan, eli ovatko vastuuhenkilöt kuitanneet tehtävänsä vai eivät. Ohjelmassa on myös hyperlinkit suoraan muutoslomakkeisiin, joten niissä olevien tehtävien kuittaminen käy helposti.

## 2 PATRIA AEROSTRUCTURES OY

### 2.1 Patrian lentokoneteollisuuden historia

Patrian lentokoneteollisuuden historia juontaa juurensa vuonna 1921 Suomenlinnaan perustettuun **Ilmavoimien lentokonetehtaaseen**. Vuonna 1928 sen ja Santahaminassa toimineen, lentokoneita ja lentokonemoottoreita korjanneen **ilmailutelakan** toiminta yhdistettiin. Näin syntyi **Valtion lentokonetehdas**, jonka toiminta siirtyi aluksi Tampereelle ja sieltä aikanaan Kuorevedelle.

Vuonna 1945 kaikki valtion asetehtaat yhdistettiin **Valtion Metallitehtaat** nimen alle. Vuonna 1952 siitä tuli **Valmet Oy**, jonka osana oli **Valmet Lentokoneteollisuus**. 1990-luvulla yhdistettiin Suomessa toimivat asetehtaat yhdeksi yhtiöksi, jonka nimeksi tuli **Patria Oy**. Patriassa lentokoneteollisuutta on **Patria Aviation** ja **Patria Aerostructures** yhtiöissä. Patria Aviation huolehtii pääasiassa Ilmavoimien lentokaluston huolloista, korjauksista ja modifikaatioista.

Vuonna 2009 koko Patria-konsernin liikevaihto oli 539,6 milj. EUR ja henkilöstö keskimäärin 3414 henkilöä.

## 2.2 Patria Aerostructures Oy lyhyesti

Patria Aerostructures valmistaa Jämsän Hallissa komposiittimateriaalista lentokoneiden osia, mm. spoileriteita Airbus A380-lentokoneisiin. Vuoden 2009 aikana yhtiö toimitti komposiittirakenteita useille asiakkaille, joista merkittävimpiä olivat Airbus UK, Airbus France, Airbus Deutschland, Agusta-Westland ja Stork Fokker.



**Kuva 1, Airbus A380 matkustajakone ja Patrian valmistama spoileri /8/**

### 3 TUOTETIEDONHALLINNAN KESKEISIÄ KÄSITTEITÄ

#### 3.1 Tuotetieto

Tuotetieto voidaan jakaa karkeasti kolmeen ryhmään.

**Tuotteen määrittelytiedot**, joita ovat fyysiset ja toiminnalliset ominaisuudet. Tähän ryhmään kuuluvat täsmällisten teknisten tietojen lisäksi abstraktit, käsitteelliset ja jopa mielikuviin liittyvät tiedot.

**Tuotteen elinkaaritiedot**, jotka alkavat teknologiatutkimuksesta ja tuotteen suunnittelusta ja päättyvät valmistamisen jälkeiseen käyttöön, huoltoon ja hävittämiseen.

**Metatieto**, tämä on informaatiota itse tiedosta: missä muodossa se on, mistä se löytyy, kuka sen on tallentanut ja milloin.

#### 3.2 Tuotetiedonhallinta – PDM (*Product Data Management*)

Kenneth McIntohin määritelmän mukaan (Engineering Data Management, McGraw-Hill Book Company, England, 1995): *Tuotetiedonhallinta on systemaattinen tapa suunnitella, hallita, ohjata ja valvoa kaikkea sitä tietoa, jota tarvitaan tuotteen dokumentoinnissa, tuotteen kehittämis-, suunnittelu-, valmistus-, testausprosessin ja käytön aikana, tuotteen koko elinkaaren ajan.*

Tuotetiedonhallinnalla ei tarkoiteta jotain yksittäistä tietokoneohjelmistoa, vaan se on ennemminkin yrityksen toimintatapa, joka toteutetaan käytännössä lähes poikkeuksetta erilaisten tietojärjestelmien avulla. Se on siis ennen kaikkea kokonaisuuden hallintaa. Sen ydin on yrityksen valmistamaan tuotteeseen ja sitä kautta yrityksen toimintaan liittyvän tiedon luominen, säilyttäminen ja tallentaminen siten, että päivittäisessä toiminnassa tarvittavan tiedon löytäminen, jalostaminen, jakelu ja uudelleenkäyttö on helppoa, nopeaa ja vaivatonta.

Tieto voi olla sähköisessä muodossa tai paperilla. Sitä ovat määrittelydokumenttien lisäksi esim. projektisuunnitelmat, sopimukset, muistiot, faksit jne.

Kirjassa *Tuotetiedonhallinta – PDM* (Sääksvuori, Antti – Immonen, Anselmi, Talentum Media Oy. Helsinki 2002. 201 s.) esitetään, että tyypilliset tuotetiedonhallinnan ongelmat yrityksissä ilmenevät kahdella eri alueella:

- Tiedon käyttö- ja tallennusmuodot vaihtelevat, mistä johtuen tiettyä tarkoitusta varten luotua tuotetietoa ei voida hyödyntää muissa tehtävissä.
- Eri yksiköissä, osastoilla ja yrityksissä tuotettavan tiedon eheyttä ja ristiriidattomuutta ei voida varmistaa.

Kun yrityksessä pyritään ratkaisemaan näitä ongelmia PDM-järjestelmällä, se edellyttää integroitumista yrityksen muihin järjestelmiin. Tällöin on varmistuttava siitä, kuka tiedon omistaa eli missä kunkin tiedon alkuperäinen lähde on ja mikä funktio siitä vastaa.

Integraatioiden taso voi vaihdella huomattavasti. Kaksi yleisintä tapaa integroida järjestelmiä ovat tiedon siirtäminen (siirtotiedosto) tai sen jakaminen (tietokanta-integraatio). Edellinen on helpompi toteuttaa, mutta sen heikkoutena on mm. kaikkien kopioitujen tiedostojen yhteneväisyyden varmistaminen. Jälkimmäinen puolestaan mahdollistaa yhteisten tietokantojen nopean käytön usealle sovellukselle. Toteutus on kuitenkin raskaampi ja muutosten teko hankalampaa.



Kuva 2, Tuotetiedon hallinta järjestelmäintegraation avulla /9/

### 3.3 Dokumentit

Dokumenttien hallinta on yksi PDM:n tärkeimmistä sovellusalueista. Keskitetyllä dokumenttien hallinnalla voidaan kehittää tehokkaasti tiedonhaku- ja arkistointimenetelmiä, mikä taas osaltaan lisää suunnittelutyön tehokkuutta. Esimerkiksi piirustukset ja niihin liittyvät osaluettelot voidaan koota yhdeksi dokumentiksi riippumatta siitä, millä ohjelmistolla kyseiset dokumentit on tehty ja sijaitsevatko ne eri paikoissa yrityksen tietojärjestelmässä. Internet-teknologiaa soveltavat PDM-sovellukset tarjoavat myös toimivan ratkaisun verkostoituneen alihankintaympäristön hallintaan.

### 3.4 Tuotetiedonhallintajärjestelmä

Tuotetiedonhallintajärjestelmä (PDM-järjestelmä) on koko yrityksen kattava tietojärjestelmä, joka yhdistää, integroi ja hallitsee yrityksen liiketoimintaprosesseja valmistettavien tuotteiden ja niihin kiinteästi liittyvien tietojen kautta.

Sääksvuori – Immonen määrittelevät kirjassaan järjestelmälle tyypillisiksi ominaisuuksiksi mm:

- a) **Nimikkeiden hallinta:** Järjestelmä hallitsee nimikkeiden tietoja ja elinkaarta sekä kontrolloi yhdessä käyttöoikeuksien kanssa nimikkeiden perustamiseen ja ylläpitämiseen liittyviä prosesseja.
- b) **Tuoterakenteen hallinta ja ylläpito:** Järjestelmä tunnistaa yksittäisen tiedon ja sen yhteydet toisiin tietoihin tuoterakenteen avulla, joka muodostuu hierarkkisesti yhteen liitetyistä nimikkeistä.
- c) **Käyttöoikeuksien hallinta:** Järjestelmä määrittelee henkilöt, jotka saavat luoda uutta tietoa, tehdä muutoksia, tarkastaa ja hyväksyä tehdyt muutokset sekä henkilöt, joilla on niihin vain katseluoikeus.
- d) **Dokumenttien ja nimikkeiden tilan, statuksen, ylläpito:** Järjestelmä ylläpitää tietoa kunkin dokumentin ja nimikkeen tilasta ja versiosta (esim. luonnos, tarkastettu, hyväksytty, luovutettu, jaettu) ja tilaan tehdyistä muutoksista, kuka teki, milloin, mitä?
- e) **Tiedonhaku:**
  - tiedon luomisvaiheessa, esimerkiksi suunnittelussa, on mahdollista hyödyntää olemassa olevaa, hyväksi ja laadukkaaksi havaittua tietoa

- saadaan helposti esille kaikki tiettyyn yksittäiseen asiaan, esimerkiksi tuotteeseen tai kokoonpanoon, liittyvät tiedot koskien suunnitelmia, dokumentteja, tuotteen osia, muita kokoonpanoja ja komponentteja
  - saadaan helposti ja nopeasti selville, miten tiedot liittyvät toisiin tietoihin.
- f) **Muutosten hallinta:** Työkalu, jolla viimeisin tieto tuotteisiin ja niiden osiin – dokumentteihin ja nimikkeisiin – tehdyistä muutoksista saadaan perille oikeaan aikaan ja oikeaan paikkaan.
- g) **Konfiguraation hallinta:** Samaan käyttötarkoitukseen tehdyn tuotteen fyysisten ominaisuuksien muuntelu ja vaihtokelpoisten kokoonpanojen tai komponenttien vaihtaminen, esimerkiksi asiakkaan toiveiden mukaan tehty tuoterakenteen variointi.
- h) **Viestien hallinta:** Viestintä toteutetaan joko sähköpostitse tai tietokantojen välityksellä.
- i) **Tiedostojen/dokumenttien hallinta:** Kyseessä on metatieto eli tieto siitä, mikä tieto sijaitsee missäkin (myös fyysiset tiedostot ovat järjestelmän hallinnassa).
- j) **Tiedon katoamisen esto:** PDM-ohjelmisto valvoo tiedostojen kopiointia ja varmistaa, että master-kopio säilyy niin kauan, kunnes tiedostot on onnistuneesti päivitetty.
- k) **Varmuuskopioiden hallinta:** Järjestelmä pitää automaattista lokia tehdyistä varmuuskopioista.
- l) **Lokikirjanpito:** Kaikki PDM:n hallinnan piirissä tehdyt toimenpiteet, kuten dokumenttien päivittäminen tai komponenttinimikkeen vaihtaminen, voidaan tarvittaessa päivittää.
- m) **Tietoholvi (*electronic vault, file vault*):** Paikka, johon tiedostot talletetaan, yleensä tiedostojen päivittäjän ja ylläpitäjän läheisyyteen, kuten saman lähiverkon tiedostopalvelimelle. Tietoholvissa tiedostot ovat PDM-ohjelmiston hallinnassa.

### 3.5 Järjestelmäarkkitehtuuri

Kaikki kaupalliset PDM-järjestelmät koostuvat tietyistä toiminnallisista kokonaisuuksista ja tekniikoista, jotka ovat toiminnallisesti samoja järjestelmästä riippumatta. Näitä ovat:

- tietoholvi
- metatietokanta
- ohjelmistosovellus.

PDM-järjestelmä vaatii toimiakseen yhden tai useampia tietokantoja. Riippuen järjestelmän toimittajasta, se voi vaatia toimiakseen juuri tietynlaisen tietokannan tai järjestelmätoimittajan oman tietokannan. Useimmat PDM-järjestelmät tukevat kuitenkin lähes aina yleisiä, laajassa käytössä olevia kaupallisia SQL-relaatio-tietokantoja, kuten Oraclea.

Yleensä järjestelmä rakentuu yhdestä tai useammasta fyysisestä palvelimesta, jotka toimivat alustoina käytössä oleville PDM-ohjelmistosovellutuksille, tietokannoille ja tiedostopalveluille. PDM-ohjelmistoille sopivia käyttöjärjestelmä-alustoja ovat tyypillisesti Microsoft Windows NT, Unix ja mahdollisesti Linux. Verkko-ratkaisuksi vaaditaan useimmiten TCP/IP-protokollalla varustettua Ethernet-verkkoa. PDM-järjestelmän käyttö Internetin kautta tapahtuu lähes poikkeuksetta normaalia Internet-selainta käyttämällä.

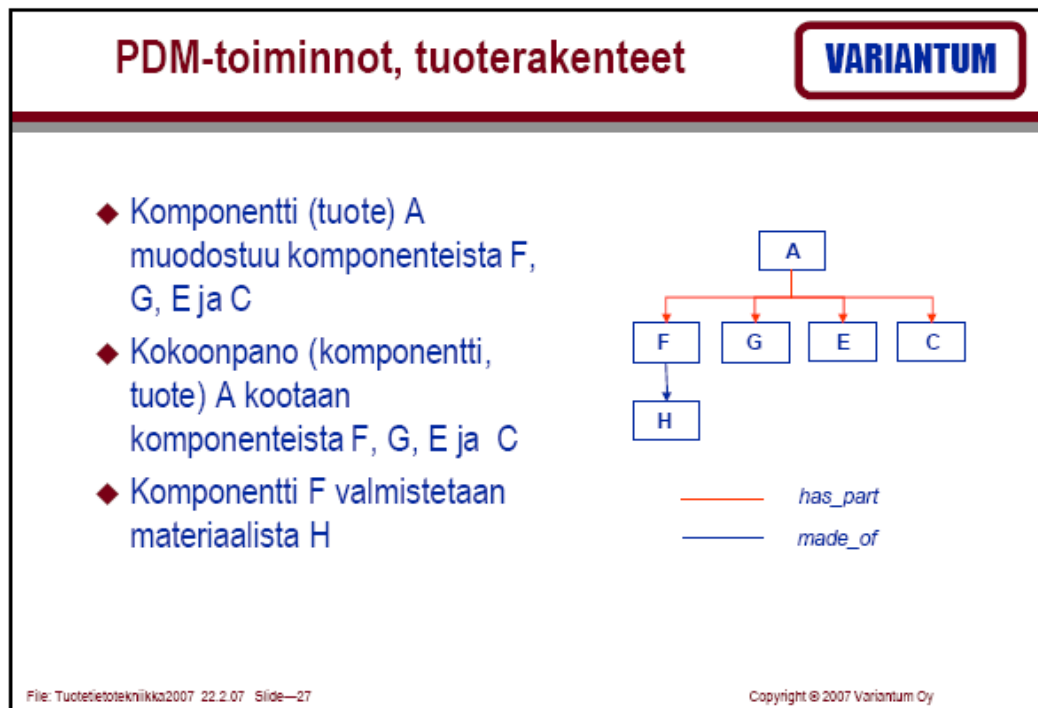
Tuotetietokuvauksen hallintaan ja tiedonsiirtoliittymien kehittämiseen sekä tietojärjestelmien integrointiin organisaatioiden sisällä ja organisaatioiden välillä on tarjolla useita kaupallisia ohjelmistosovellutuksia. Ne pohjautuvat lukuisiin standardeihin, kuten STEP (*Standard for the Exchange of Product Model Data*) ja XML (*Extended Markup Language*).

PDM-järjestelmä voi olla myös suoraan *integroitu* ohjelmistoihin, joilla dokumentit tuotetaan, esimerkiksi CAD- ja toimisto-ohjelmiin. Tällöin PDM-ohjelmiston komennot voidaan antaa suoraan vaikkapa CAD-ohjelman sisältä.

Suomessa yritysten eniten käyttämä PDM-järjestelmä oli vuonna 2009 Valokynä-lehden kyselyn mukaan Modultekin Aton (Aton PDM ja SolidPDM) [Valokynä 2/2009]. Muita tunnettuja PDM-järjestelmiä ovat muun muassa Vertex PDM (suomalainen), Enovia Smarteam, Enovia LCA, Enovia MatrixOne, Auric Evo, Windchill, Teamcenter engineering & enterprise ja PDMWorks. Näistä useat ovat ainakin laajennettavissa myös PLM- eli tuotteen elinkaaren hallintajärjestelmiksi./10/

## 3.6 Tuoterakenne

Suuri osa PDM-järjestelmän toiminnoista pohjautuu tuoterakenteen ja nimikkeiden käyttöön. Tuoterakenteen hallinta ja ylläpito on yksi tärkeimmistä osista koko järjestelmässä, se luo pohjan useimmille muille järjestelmän perustoiminnoista. Nimike on systemaattinen ja standardi tapa identifioida, koodata ja nimetä tuote, tuotteen osa tai komponentti, materiaali tai palvelu. Jokaisella objektilla on yksilöllinen koodi.



**Kuva 3, Tuoterakenteen periaate /9/**

PDM-järjestelmän piirissä olevat nimikkeet – eli osat tai komponentit, sekä kokoonpanot – kiinnitetään kuhunkin tuotteeseen ja toisiinsa tuoterakenteen kautta.

Tuoterakenteen tulee muodostaa tuotteesta kussakin tilanteessa tarkoituksenmukaisen tarkka kuvaus. Siihen voidaan luoda erityisiä näkymiä tietyn tarpeen mukaan. PDM-järjestelmissä tuoterakennetta voidaan tyypillisesti suodattaa siten, että tietyt rakenteen osat tulevat korostetusti esille ja tietyt osat eivät näy lainkaan.

Tuoterakenteeseen liitetään tyypillisesti myös attribuuttitietoa:

- tuoteyksilökohtaista – esim. ostokomponentin sarjanumero
- geneeristä (yleistä) – koskien tuotteita, tuotetta, kokoonpanoa, osia
- käyttäjäkohtaista – omia huomautuksia, muistiinpanoja

## 4 MUUTOSHALLINTA PATRIA AEROSTRUCTURES (AST) OY:SSÄ

Kuten edellä kohdassa 3.4 todettiin, muutoksia hallitaan yrityksen PDM-järjestelmällä. Muutoshallinta on myös riskienhallintaa. AST:ssä toteutetaan tämän osalta menettelytapaa, joka on kuvattu ohjeessa ”A-PR-041-012 - Muutoshallinta”, sekä ”EV5-0169 Enovia LCA V5 Muutoshallinta (pikaohje)”.

PDM-järjestelmän tehtävänä muutosprosessissa on:

- toimia tiedonkeruu- ja jakeluväylänä muutosprosessin aikana – mitä muutettiin ja minkä takia
- auttaa selvittämään muutoksen vaikutukset tuotteeseen (muutospyyntö)
- hallita muutoksen läpivientiä tehtävien avulla (muutosmääräys)
- muutosten jäljitettävyyden seuranta

### 4.1 *Baseline*-käsite

AST:ssä käytetään *baseline*-termiä kuvaamaan tarkasteluhetkellä (*current baseline*) voimassa olevaa hankkeen sisältöä. Se ei rajoitu pelkästään tuotemäärittelyyn, vaan kattaa kaiken, mitä hankkeessa on sovittu. Hankkeen aikana *baseline* kehittyy täydentymällä, tarkentumalla ja muutosten kautta.

*Baselinella* tarkoitetaan yleensä ulkoista *baselinea*, jolla on yhteinen rajapinta asiakkaan kanssa. Se voi olla myös sisäinen *baseline*, jolloin sillä tarkoitetaan AST:n sisäisiä sopimuksia ja suunnitelmia.

Ulkoinen *baseline* jäädytetään asiakkaan kanssa kirjallisesti sopimalla. Tähän liittyvät suunnittelu- ym. katselmukset. Tyypillisesti tärkeimmät sopimukselliset dokumentit, kuten Sopimus, State of Work, pääaikataulu jne. muodostavat baseliinin selkärangan.

### 4.2 Muutoskäsite

AST:ssa muutoksella tarkoitetaan poikkeamista aiemmin sovitusta asiasta, joko siihen jotain lisäämällä ja/tai siitä jotain poistamalla. Muutoksia ovat kaikki tuotepiirteet, toiminta, tapahtumat yms., jotka johtavat poikkeamaan määritellystä ja sovitusta *baselinesta*.

Ilmailuteollisuudessa kaikki tuotteeseen liittyvät muutokset on kyettävä tunnistamaan, suunnittelemaan ja toteuttamaan hallitusti, lentoturvallisuuden varmistamiseksi.

Muutoshallinta edellyttää, että kaikella hankkeisiin liittyvällä tiedolla tai asialla on omistajat, jotka vastaavat kyseisen tiedon seurannasta ja tietoon saattamisesta sitä tarvitseville tahoille.

### 4.3 Konfiguraatiohallinta

Konfiguraatiohallinnalla käsitetään suoraan tuotteeseen liittyvää tiedon hallintaa. Tuotteen konfiguraatio muodostuu sen määrittelydokumenttien perusteella: CAD-mallit, piirustukset, materiaalispesifikaatiot, prosessipesifikaatiot, työvälineet, valmistusohjeet jne. Hankkeissa konfiguraatiohallinnasta vastaa kyseiselle hankkeelle nimetty konfiguraatioinsinööri.

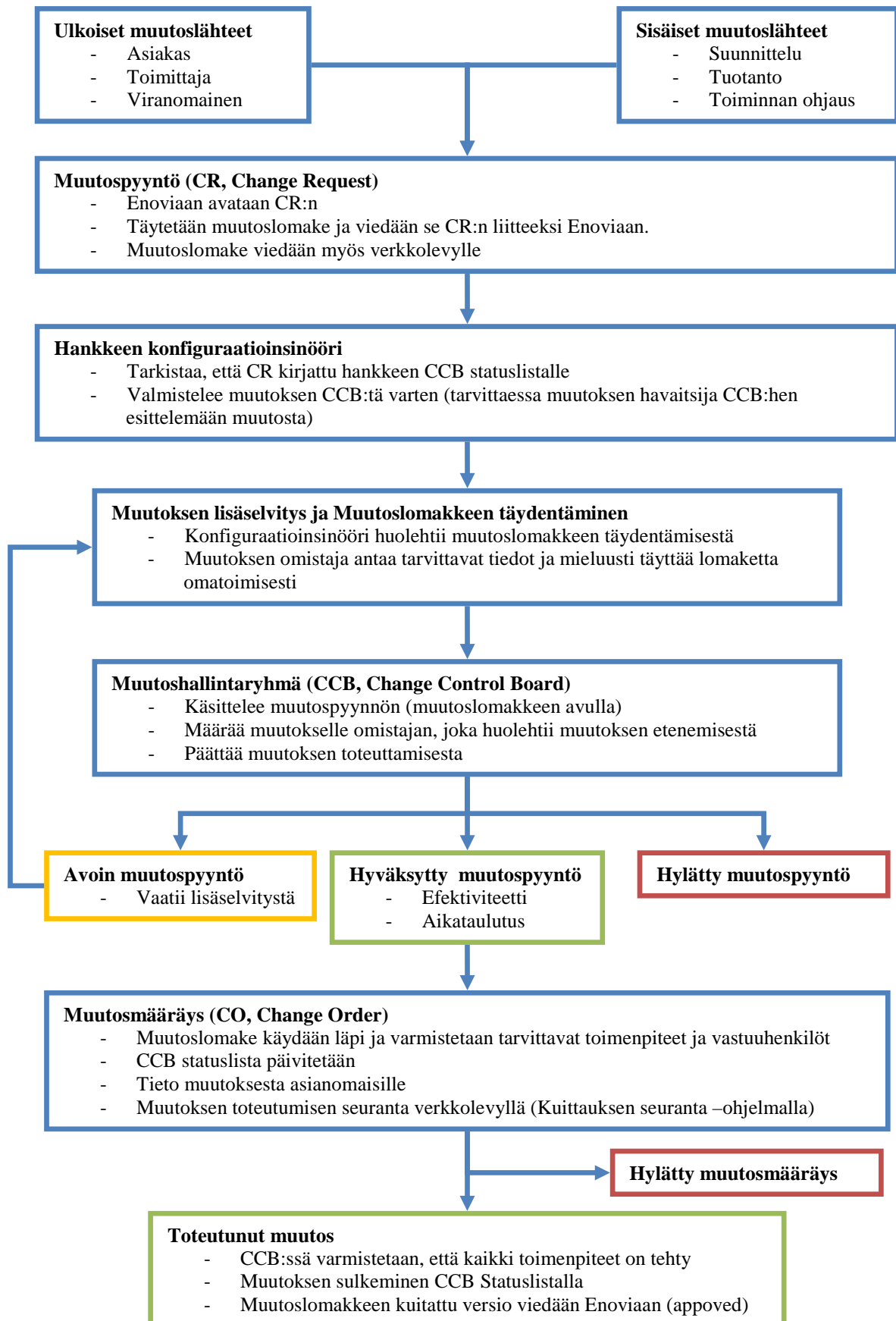
### 4.4 Muutosten toteutumisen seuranta

Seuraavassa muutosprosessikaaviossa näemme, että hyväksytystä muutospyynnöstä tehdään PDM-järjestelmään muutosmääräys (*Change Order – CO*). Muutosmääräyksellä kerättyjen lähtötietojen perusteella tehdään muutosmääräyksen alle tehtäviä, jotka suoritetaan muutoksen läpiviemiseksi.

Muutospyynnön ja muutosmääräyksen omistaa niiden tekijä, joka vastaa myös niiden läpiviemisestä. Muutospyynnöt voivat olla myös hierarkisia esim. muutosmääräys voi sisältää alimuutospyyntöjä. Muutosmääräys hyväksytään suoritetuksi, kun kaikki tehtävät on suoritettu hyväksytysti. Tieto tehdyistä muutoksista liitetään muutosmääräykseen.

## 4.5 Muutosprosessikaavio

Muutosprosessi käydään AST:llä nykyään läpi seuraavasti:



Kuva 4, Muutosprosessikaavio /6/

## 5 MUUTOSHALLINNAN KÄYTÄNNÖN ONGELMA JA RATKAISU

### 5.1 Tehtävien etenemisen seuraamisen hankaluus Enovialla

Muutospyyntöön (CR) ja muutosmääräykseen (CO) liittyvät tehtävät kirjataan muutoslomakkeeseen ja tehtäville määritetään vastuuhenkilöt. Kun tehtävät on hyväksytysti suoritettu, asianomaiset vastuuhenkilöt kuittaavat ne kyseiseen lomakkeeseen suoritetuiksi.

Koska samaa lomaketta käsittelee useampi henkilö, se talletetaan Unix-asemalle turvaan ja siihen voi päästä käsiksi vain Enovia-tuotetiedonhallintajärjestelmällä. Näin varmistetaan, että sitä voi muokata vain yksi ihminen kerrallaan ja vaaraa lomakkeen tuhoutumisesta, häviämisestä tai muuten muuttumisesta ei ole, koska Enovia säilyttää varmuuskopion lomakkeesta aina kun sen ottaa ulos muokattavaksi.

Ongelmana on se, että ihmiset eivät näytä kuittaavan tekemisiään. Tällöin syntyy epävarmuus siitä, onko tehtäviä suoritettu, vai onko tehty, mutta jätetty kuittaamatta. Vastuuhenkilöiden on siis kuitattava tehtävä suoritetuksi sekä muutoslomakkeeseen että statuslistaan. CCB-statuslistalla olevien muutosten määrä kasvaa, mutta niitä ei voi poistaa ennen kuin vaadittavat kuittaukset on saatu kaikilta vastuuhenkilöiltä.

Kuittauksen suoritus Enovian kautta tapahtuu näin:

- kirjaudu Enoviaan
- etsi muutosdokumentti ”abstraktin” ID:n perusteella (esim. NHSP-1423), tai jollain muulla hakukriteerillä
- jos et muista ID:tä, tarkista se CCB-statuslistalta, joka löytyy Enoviasta
- ”check out” dokumentti ulos järjestelmästä, jotta voit muokata sitä
- etsi muutosdokumentti omalta koneelta ja avaa se
- kuittaa ”OK” oman tehtävän kohdalle, tallenna ja sulje
- ”check in” dokumentti takaisin Enoviaan
- etsi CCB-statuslista Enoviasta
- ”check out”
- etsi ja avaa, poista nimesi tehtävältä, tallenna, sulje
- ”check in” Enoviaan
- tallenna Enovia

Kuten edellä olevasta ohjeesta käy ilmi, kuittausten päivittäminen on monimutkaista ja hidasta. Tämä korostuu silloin, jos kuittaamattomia muutoksia on kertynyt kymmeniä kappaleita, koska sama kuittausprosessi on käytävä yksitellen läpi jokaisen muutoksen osalta. Ongelmana on myös statusraportoinnin vaikeus, koska Enoviasta ei saada listausta tehtävien tilasta.

## 5.2 Ongelman ratkaisu

Ongelman ratkaisemiseksi tehtiin kattava Excel-tilukkolaskentaohjelma, jolle annettiin nimi Kuittausten seuranta (KS). Tällä voidaan seurata reaaliajassa muutoslomakkeisiin eri vastuuhenkilöille merkittyjä tehtäviä ja niiden tilaa, eli onko niitä tehty ja kuitattu vai ei.

Avoimna olevat muutoslomakkeet siirrettiin Enoviasta verkkolevylle hankkeittain ”Muutokset”-kansioiden alle. Enoviassa säilyy aina varmuuskopiot muutoslomakkeista, samoin kuin itse KS-ohjelmasta. Muutoslomakkeet muutettiin Word-muodosta Excel-muotoon.

Myös CCB-statuslista siirtyi Enoviasta verkkolevylle. Statuslistaa päivitetään viikottaisissa CCB:n kokouksissa, joissa käydään muutoksia läpi. Statuslista on siis lista sillä hetkellä avoimna olevista muutoksista. Joka kokouksen jälkeen dokumentti tallennetaan Enoviaan ja *approvataan* (hyväksytään), uusi revisio otetaan verkkolevylle. Statuslista dokumentissa on myös pöytäkirjasivu.

KS toimii siten, että se hakee ensin avoimna olevat muutokset CCB-statuslistalta, sitten tarkistaa ”Muutokset”-kansiossa olevat muutoslomakkeet ja linkittää niistä vastuuhenkilöiden alle yhä kuittaamatta olevat tehtävät sekä summaa ne.

Ohjelmasta on apua sekä vastuuhenkilöille, hankepääliköille että konfiguraatio-insinööreille. Vastuuhenkilöt näkevät yhdellä sivulla kaikki tehtävänsä ja niiden statuksen ja pääsevät yhdellä hiiren painalluksella muokkaamaan muutoslomaketta. Hankepäälliköt taas näkevät yleiskatsauksen koko hankkeen muutosten tilasta.

KS sisältää myös hyperlinkit muutoslomakkeisiin, joten ne on helppo aukaista itse ohjelmasta. Ohjelma päivittyy sitä mukaan, kun vastuuhenkilöt kuittaavat tehtäviään suoritetuiksi.

Uudistuksen seurauksena Enoviaa ei tarvitse enää käyttää pelkän ”OK”:n merkitsemiseen, eikä CCB:n statuslistaa kuittausten seurantaan.

Ohjelman on tehnyt tämän tutkintotyön laatija ollessaan konfiguraatioinsinööri-harjoittelijana Patria Aerostructures Oy:ssä. Se on otettu käyttöön osana AST:n muutoshallintaa, ja sen käytöstä on mainittu dokumentissa ”Uudistetut toimintatavat Doc ID: AST-0209”.

Ohjelman laatimisessa käytettiin VBA:ta (*Visual Basic for Applications*), joka on Microsoftin sovellus-ohjelmissa makrokielenä käytetty ohjelmointikieli. Ohjelmassa on merkittävä rooli makroilla. Ne löytyvät liitteinä tämän tutkintotyön lopussa.

	A	B	C	D	E	F	K	L	M	N	O	P	Q	R	S	T	U	V	V
1	Kuittauksen seuranta			kuittattu / Signed															
2	[Type Program name here]			työn alla / In work															
3	CCB Status List			myöhässä / Late															
4	Names List																		
5	Sum.	NHRF-	Subject	Effectivity	Owner	CO	DM	JaRe	KaRi	KiHa	LaVi	Miin	JoAa	JoSu	RiKa	VeMi	YrEI	Juls	Trl
33	3 / 8	1358		ASN65 GIT.CI-TREE		410		ok			1		ok				ok		0
34	3 / 11	1350		ASN65 GIT.CI-TREE		410		ok					ok				ok		0
35	1 / 7	1472		ASN65 GIT.CI-TREE		410	1						ok				ok		0
36	1 / 7	1473		ASN65 GIT.CI-TREE		410	1						ok				ok		0
37	1 / 10	1348		ASN65 GIT.CI-TREE		410		ok	ok				ok					ok	
38	1 / 10	1349		ASN65 GIT.CI-TREE		410		ok	ok				ok					ok	
39	2 / 8	1359		ASN65 GIT.CI-TREE		410		ok					ok				ok		0
40	2 / 4	1351		ASN65 GIT.CI-TREE		410		ok					ok						
41	2 / 4	1388		ASN65 GIT.CI-TREE		410		ok					ok					1	
42	1 / 5	1411		ASN65 GIT.CI-TREE		410		ok					ok				ok		
43	7 / 10	1412		ASN65 GIT.CI-TREE		410	2	ok					ok						
44	0 / 0	1422		ASN65 GIT.CI-TREE		410													
45	1 / 3	1426		ASN65 GIT.CI-TREE		410		ok					ok						
46	3 / 5	1474		ASN65 GIT.CI-TREE		410							ok						
47	1 / 7	1476		ASN65 GIT.CI-TREE		410							ok				ok		0
48	1 / 7	1477		ASN65 GIT.CI-TREE		410							ok				ok		0
49	5 / 8	1480		CI-TREE		410		ok					ok					1	
50	3 / 7	1481		CI-TREE		410		ok					ok						0
51	3 / 5	1494		ASN67 TFI		410							ok						
52	2 / 6	1366		ASN69 TGE CI-TREE		410		ok	ok				ok		1				
53	0 / 2	1306		ASN69 TGE CI-TREE		410		ok					ok						
54	6 / 7	1305		ASN69 TGE CI-TREE		410		ok					ok						1
55	5 / 7	1307		ASN69 TGE CI-TREE		410		ok					ok						1
56	9 / 10	1372		ASN69 TGE CI-TREE		410		ok					ok		1	2			1
57	3 / 8	1356		ASN69 TGE CI-TREE		410		ok					ok						0
58	0 / 3	1363		ASN69 TCCI-TREE		410		ok					ok						
59	1 / 9	1367		ASN69 TGE CI-TREE		410		ok	ok				ok						
60	2 / 7	1368		ASN69 TGE CI-TREE		410		ok	1				ok						
61	6 / 8	1373		ASN69 TGE CI-TREE		410		ok					ok						1
62	6 / 8	1374		ASN69 TGE CI-TREE		410		ok					ok						1
63	9 / 10	1376		ASN69 TGE CI-TREE		410		ok					ok		1	2			1
64	6 / 8	1377		ASN69 TGE CI-TREE		410		ok					ok		1				1
65	7 / 8	1375		ASN69 TGE CI-TREE		410		ok					ok		1				1
66	3 / 7	1482		ASN69 TGE CI-TREE		410							ok				ok		0
67	5 / 6	1483		ASN69 TGE CI-TREE		410							ok						0

Kuva 5, Kuittauksen seuranta. Subject-sarakkeen tiedot peitetty luottamuksellisina

## 6 KUINKA OHJELMA TOIMII

Seuraavassa on esimerkkien avulla esitetty pääkohdat siitä, miten Kuittauksen seuranta -ohjelma on toteutettu.

### 6.1 Muutoslomake

Kuittauksen seuranta hakee tietoja varsinaisista muutoslomakkeista, jotka ovat omien hankkeittensa ”Muutokset” -kansioissa verkkolevyllä.

Tyhjä muutoslomake on kokonaisuudessaan liitteenä.

Kuvassa 3 on muutoslomakkeen tehtävälistaosa. Kun muutos päätetään toteuttaa, jaetaan vastuuhenkilöille tehtäviä muutoksen läpiviemiseksi. Tehtävien kuvaus, vastuuhenkilöt, deadline ja kuittaus löytyvät muutoslomakkeen tehtävälistaosasta.

Esimerkiksi kuvassa 7 voidaan nähdä, että kaikki muut tehtävät on kuitattu, paitsi ”ERP rakenne/työpapereiden päivitys” rivillä 59, (vastuuhenkilönä RiKa) sekä ”Leikkuriohjelmien korjaus + vienti Enoviaan” rivillä 68 (vastuuhenkilönä JoSu).

	Effected area	EFFECTS and ACTIONS	Resp	Due Date	Used Hours	Signature
53						
54	<b>CONFIGURATION / DATA MANAGEMENT (Minna Innala)</b>					
55	CM actions	Ohjelmien nimeämis- ja tallennusohje, A-PR-041-010 issue 3 --> 4	Miln	05.06.2009		Ok RiYL 12.6.2009
56	DM actions	Ohjejakelut	KiHa	08.08.2009		ok 22.7.2009
57	Supplier ADL					
58	<b>MANUFACTURING ENGINEERING (Ritva Karinsalo)</b>					
59	Manufacturing instructions	ERP rakenne/työpapereiden päivitys	RiKa	05.06.2009		
60	Manufacturing instructions	A-TO ohjeiden päivitys	RiKa			ok 22.5.2009
61	Manufacturing Files					
62	- NC					
63	- Laser Data / Calibration					
64	- Plybook					
65	- apumitat	pystyseinän palojen paikoitusmitat	AnAM	02.06.2009		ok 2.6.2009
66	- Flatten geometry	FG:n Korjaus	AnAM	04.05.2009		ok 11.5.2009
	- muokatut palat					Ok
67		FC-50...FC-62 1:1 kuvat	RiKa	13.06.2009		13.06.2009
68	- Cutting	Leikkuriohjelmien korjaus + vienti Enoviaan	JoSu	05.09.2009		
69	- Other	Leikkuriohjelmien tarkastus	AnAM			ok 11.5.2009
70	Mfg. Dossier / Other?					

Kuva 6, Muutoslomakkeen etusivun tehtävät ja vastuuhenkilöt

## 6.2 Kuittauksen Seuranta -taulukon rakentuminen

Jos taulukkoon ei ole linkitetty yhtään mitään, se näyttää kuvan 8-laiselta. Sarakkeilla A-F ja riviltä 5 alaspäin on soluissa kaavoja. Kaavat näyttävät tällä hetkellä nollaa, ja nollat on maalattu valkoisiksi muodollisilla ehtolauseilla. Sarakkeilla K-L on pivot-taulukko odottamassa dataa.

Esimerkiksi kuvan 8 solussa C5 on seuraava kaava:

```
=IF(ISERROR(INDEX(Statuslista;ROW($A1);Program!$B$16));0;INDEX(Statuslista;ROW($A1);Program!$B$16))
```

Kaava tarkoittaa: ”Näytä Statuslistan solu, jonka rivi on 1 ja sarake Program-välilehden solun B16 arvo, mutta jos virheitä, näytä 0.”

Linkki muodostuu epäsuorasti johonkin CCB Statuslistan soluun. Näin kaava osoittaa aina tiettyyn soluosoitteeseen riippumatta siitä, liikkeuko lähdesolu CCB Statuslistalla esimerkiksi solujen lisäämisten ja poistamisten myötä.

“Statuslista” on nimetty alue, jota ei ole vielä määritetty, joten kaava on virheellinen ja näyttää nollaa.

	A	B	C	D	E	F	K	L	M
1	Kuittauksen seuranta								
2	[Type Program name here]		Kuitattu / Signed						
3	CCB Status List		Työn alla / In work						
4	Names List		Myöhässä / Late						
5	Sum.	-5	0	0	0	0	(blank)		
6									
7									
8									
9									
10									
11									
54									

Kuva 7, Tyhjä Kuittauksen Seuranta

Jotta taulukko saadaan käyntiin, täytyy mennä KS:n Program välilehdelle ja syöttää tarvittavia tietoja ja painaa ”Update Table” nappia. Kuittauksen seuranta vaatii mm. hankkeen kansion polun, sekä CCB Statuslistan ID:n, rivi- ja sarakenumeroita, joista muodostetaan nimetty alue ”Statuslista”.

Kun tiedot on kerran syötetty, niihin ei tarvitse myöhemmin koskea.

Kuvassa 9 on KS:n Program-välilehti ja valmiiksi syötetty tietoja.

	A	B	C	D
1	Ready	TRUE		
2				
3	Main Folder:	C:\Esimerkkikansio	Update Table	
4	Name list ID:	XYZ-1234		
5				
6	Program			
7	Program ID:	NHRF		
8	Program folder:	C:\Esimerkkikansio\Hanke		
9	Changes folder:	C:\Esimerkkikansio\Hanke\Muutokset		
10				
11	CCB Status List:			
12	ID	NHRF-1043		
13	Header Row	4		
14	ID Column	4		
15	CO Column	5		
16	Subject Column	8		
17	Extra 1 Column	9		
18	Extra 2 Column	10		
19	Right Edge	10		
20				
54				

Kuva 8, Program-välilehdellä on tarvittavat CCB Statuslistan rivi-, ja sarakenumerot, joilta linkitetään tietoja seurantaan.

Kun ”Update Table” –nappia painetaan, alkavat makrot toimia ja tapahtuu seuraavaa:

### 1. vaihe

Nimetty alue ”Statuslista” muodostuu Program-välilehden tietojen perusteella. Nyt etusivun kaavat toimivat ja CCB-statuslistan tiedot listautuvat Kuittauksen seurantaan (kuva 10).

	A	B	C	D	E	F	K	L
1	Kuittauksen seuranta							
2	[Type Program name here]		kuitattu / signed		Update			
3	CCB Status List		työn alla / In work		Table		Osa	Nar
4	Names List		myöhässä / Late				(bla	
5	Sum.	NHRF-	Subject	Effectivity	Owner	CO	(blank	
6	0 / 0	1061		TFIA20		191		
7	0 / 0	1107		ASN39		191		
8	0 / 0	1108		ASN39		191		
9	0 / 0	1076		ASN 031		194		
10	0 / 0	1102				202		
11	0 / 0	1135		ASN40		212		

Kuva 9, CCB Statuslista linkitetty.

## 2. vaihe

Data-välilehdelle muodostuu alustava lista avointen muutosten ID:istä etusivun sarakkeen B perusteella. Kuvassa 11, rivillä 14 on ID-arvo -17. Tämä johtuu siitä, että CCB Statuslistalla on tuossa kohtaa tyhjä solu. Myöhemmin pivot-taulukko kuitenkin vaatii myös tälle tyhjälle solulle ainutkertaisen ID:n, joten sille on annettu negatiivinen arvo ”-17”, koska se on etusivulla rivillä 17 (kuva 12).

	A	B	C	D	E	F
1		DataPivot	<-- -->	SeurantaPivot		
2	DataIDt	Total	ID	Name	Osasto	Pivot Data
3	1061	-1	1061			
4	1107	-1	1107			
5	1108	-1	1108			
6	1076	-1	1076			
7	1102	-1	1102			
8	1135	-1	1135			
9	1176	-1	1176			
10	1177	-1	1177			
11	1186	-1	1186			
12	1204	-1	1204			
13	1205	-1	1205			
14	-17	-1	-17			
15	1266	-1	1266			
16	1267	-1	1267			
54	1356	-1	1356			

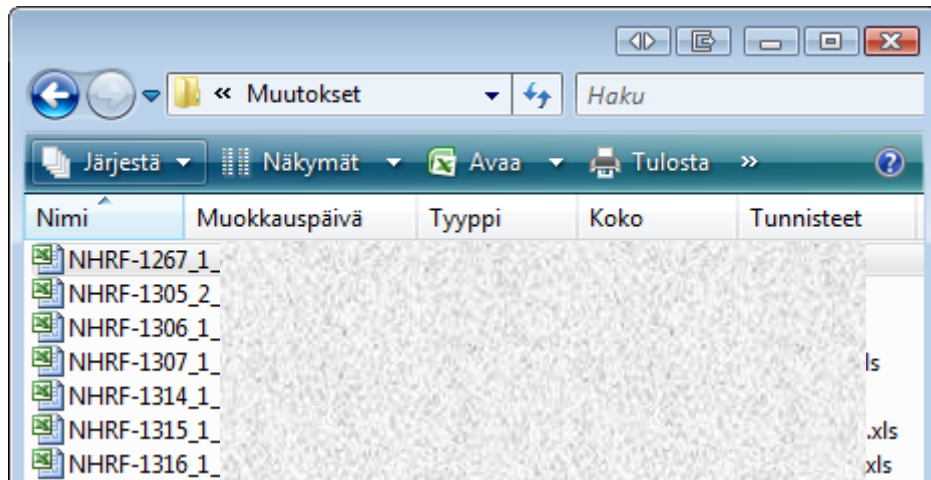
Kuva 10, Datasivulle kopioitiin kaikki Muutos-ID:t

16	No Link	1205		ASN57 TG
17	No Link	-17		
18	No Link	1266		
63	9 / 10	1376		ASN69 TG CI-TREE

Kuva 11, Tyhjä ID rivillä 17. Solu mustattu esimerkin vuoksi.

### 3. vaihe

Makro käy kaikki Data-välilehdellä olevat muutos-ID:t läpi yksitellen ja etsii ”Muutokset”-kansioista vastaavat Muutoslomakkeet (Kuva 13).



Kuva 12, Muutoslomakkeita kansiossa

Kun oikea lomake löytyy, makro avaa muutoslomakkeen (kuva 14) ja menee Piilo-välilehdelle, joka on normaalisti piilotettu (kuva 15).

	Effected area	EFFECTS and ACTIONS	Resp	Due Date	Used Hours	Signature
53						
54	<b>CONFIGURATION / DATA MANAGEMENT (Minna Innala)</b>					
55	CM actions	Ohjelmien nimeämis- ja tallennusohje, A-PR-041-010 issue 3 --> 4	Miln	05.06.2009		Ok RiYL 12.6.2009
56	DM actions	Ohjejakelut	KiHa	08.08.2009		ok 22.7.2009
57	Supplier ADL					
58	<b>MANUFACTURING ENGINEERING (Ritva Karinsalo)</b>					
59	Manufacturing instructions	ERP rakenne/työpapereiden päivitys	RiKa	05.06.2009		
60	Manufacturing instructions	A-TO ohjeiden päivitys	RiKa			ok 22.5.2009
61	Manufacturing Files					
62	- NC					
63	- Laser Data / Calibration					
64	- Plybook					
65	- apumitat	pystyseinän palojen paikoitusmitat	AnAM	02.06.2009		ok 2.6.2009
66	- Flatten geometry	FG:n Korjaus	AnAM	04.05.2009		ok 11.5.2009
	- muokatut palat					
67		FC-50...FC-62 1:1 kuvat	RiKa	13.06.2009		Ok 13.06.2009
68	- Cutting	Leikkuriohjelmien korjaus + vienti Enoviaan	JoSu	05.09.2009		
69	- Other	Leikkuriohjelmien tarkastus	AnAM			ok 11.5.2009
70	Mfg. Dossier / Other?					

Kuva 13, Muutoslomakkeen etusivun tehtävät ja vastuuhenkilöt

Muutoslomakkeen Piilo-välilehti on kerännyt tietoja etusivulta ja valmistellut ne Kuittauksen Seuranta varten. Kuittauksen Seuranta tarvitsee yhteenvetoa varten tiedon, mikä on kunkin vastuuhenkilön kuittaamattomien tehtävien varhaisin deadline. Näin voidaan määrittää onko vähintään yksi *Deadline* ylitetty. Tämä vaatii matriisikaavoja ja se on toteutettu seuraavasti.

Sarakkeen I soluilla on matriisikaavat. Esimerkiksi kuvan 15 solussa I7 on seuraava kaava:

$\{=MIN(IF(\$A\$2:\$A\$81=\$A7;IF(\$H\$2:\$H\$81<>0;\$F\$2:\$F\$81)))\}$

Kaava toimii näin:

Kaava etsii ensin A-sarakkeelta kaikki solut, joissa esiintyy solussa A7 oleva nimi, eli tässä tapauksessa "RiKa". Tässä esimerkissä löytyy kolme solua: A7, A8 ja A15.

Seuraavaksi kaava katsoo näiden löytyneiden solujen riveiltä sarakkeelta H, että onko arvo muu, kuin nolla (0 = kuitattu). Jäljelle jää rivit 7 ja 8. Lopuksi kaava valitsee näistä kahdesta pienimmän päivämääräarvon sarakkeelta F. Siis 39969, joka tarkoittaa 5.6.2009.

Sarake D käyttää sarakkeen I päivämääriä, paitsi jos on kuitattu, niin arvo on 0, tai jos mitään päivämäärää ei ole, arvo on 100000.

	A	B	C	D	E	F	G	H	I	J
1	Name	Osasto	Pivot Data	Date	E & A	Due Date	Signature	OK = 0	Oldest date / person	
2				100000	0	0	0	1	0	
3	Miln			0	Ohjelmien ni	39969	Ok RiYL 12.	0	0	
4	KiHa			0	Ohjejakelut	40033	ok 22.7.2009	0	0	
5				100000	0	0	0	1	0	
6				100000	0	0	0	1	0	
7	RiKa			39969	ERP rakenne	39969	0	1	39969	
8	RiKa			39969	A-TO ohjeide	40051	0	1	39969	
9				100000	0	0	0	1	0	
10				100000	0	0	0	1	0	
11				100000	0	0	0	1	0	
12				100000	0	0	0	1	0	
13	AnAM			0	pystyseinän	39966	ok 2.6.2009	0	0	
14	AnAM			0	FG:n Korjaus	39937	ok 11.5.2009	0	0	
15	RiKa			0	FC-50...FC-6	39977	Ok 13.06.20	0	39969	
16	JoSu			39969	Leikkuriohjel	39969	0	1	39969	
17	AnAM			0	Leikkuriohjel	0	ok 11.5.2009	0	0	
18				100000	0	0	0	1	0	
54				100000	0	0	0	1	0	

Kuva 14, Muutoslomakkeen piilotetun välilehden data, joka on poimittu etusivulta ja käsitelty valmiiksi KS:ää varten.

#### 4. vaihe

Makro linkittää Muutoslomakkeen piilotetun välilehden seitsemän ensimmäistä saraketta KS:n Data-välilehdelle oikean ID:n kohdalle sarakkeille D-J (Kuva 16).

Sarakkeelle E rakentuu kaava, joka katsoo, mihin osastoon vastuuhenkilö sarakkeella D kuuluu. Päivitettyä listaa vastuuhenkilöistä osastoittain pidetään toisessa Excel taulukossa.

Sarakkeella F on kaavat, jotka tarkistavat, onko vastuuhenkilön kuittaamatta olevien tehtävien deadline ylitetty tällä hetkellä, vai ei. Esimerkiksi solussa F5710 on seuraava kaava

=IF(\$G5710=0;0;IF(\$G5710<TODAY();-1;1))

Kaavan selitys:

Jos G sarakkeen arvo näyttää nollaa, niin se on kuitattu ja arvo on 0. Jos G:n arvo on pienempi, kuin tämä päivä, niin arvo on -1 (myöhässä). Jos suurempi, kuin tämä päivä, arvo on 1 (aikataulussa). Arvo 100000 on sitä varten, että jos deadline päivää ei ole määritelty, niin arvo on varmuudella suurempi, kuin tämä päivä, eli aikataulussa.

Koska jokaisessa muutoksessa katsotaan vain pienintä päivämäärää per henkilö, ovat kaikki etumerkit silloin samoja ja ynnättävissä yhteenvetoa varten.

	A	B	C	D	E	F	G	H	I	J
1		DataPivot	<-- -->	SeurantaPivot						
2	DataDt	Total	ID	Name	Osasto	Pivot Data	Date	E & A	Due Date	Signature
5703		0	1516		OTHER	1	100000	0	0	0
5704	1517	0	1517		OTHER	1	100000	0	0	0
5705		1	1517	MiIn	CONFIGUR	0	0	Ohjelmien r	39969	Ok RiYL 12.6.2
5706		1	1517	KiHa	CONFIGUR	0	0	Ohjejakelut	40033	ok 22.7.2009
5707		0	1517		OTHER	1	100000	0	0	0
5708		0	1517		OTHER	1	100000	0	0	0
5709		1	1517	RiKa	MANUFAC	-1	39969	ERP rakenr	39969	0
5710		1	1517	RiKa	MANUFAC	-1	39969	A-TO ohjeid	40051	0
5711		0	1517		OTHER	1	100000	0	0	0
5712		0	1517		OTHER	1	100000	0	0	0
5713		0	1517		OTHER	1	100000	0	0	0
5714		0	1517		OTHER	1	100000	0	0	0
5715		1	1517	AnAM	DESIGN	0	0	pystyseinär	39966	ok 2.6.2009
5716		1	1517	AnAM	DESIGN	0	0	FG:n Korjat	39937	ok 11.5.2009
5717		1	1517	RiKa	MANUFAC	0	0	FC-50...FC:	39977	Ok 13.06.2009
5718		1	1517	JoSu	MANUFAC	-1	39969	Leikkuriohje	39969	0
5719		1	1517	AnAM	DESIGN	0	0	Leikkuriohje	0	ok 11.5.2009
5720		0	1517		OTHER	1	100000	0	0	0
5754		0	1517		OTHER	1	100000	0	0	0

Kuva 15, KS:n Data-välilehdelle linkitetty muutoslomakkeen data

## 5. vaihe

Kun makro on käynyt kaikki Muutoslomakkeet läpi, se päivittää etusivulla olevan pivot-tilin. Pivot-tili kokoaa Datasivulta ID:t, vastuuhenkilöiden nimet, vastuuhenkilöiden osastot, sekä kuittausdatan (-1, 0, 1), ja järjestele ne siten, että nimet ovat ylhäällä järjestettynä osastoittain, ja MuutosID:t ovat vasemmalla ja kuittausdata summattuna niiden välissä (kuva 17).

CO	Yht	Pos.	ID	JoAa	JoSu	RiKa	VeMi	YrEI	IrLa	AnFa	HeJa	IvSa	JuPo	OuKo	PaOj	PaOl
410	8	28	1358	72	0			0	0						-1	
410	11	29	1350	69	0			0	0		1				1	
410	7	30	1472	73	0			0	0							0
410	7	31	1473	73	0			0	0							0
410	10	32	1348	70	0								-1			
410	10	33	1349	70	0								1			
410	8	34	1359	72	0						-1				-1	
410	4	35	1351	76	0						-1		-1			
410	4	36	1388	76	0						-1					
410	5	37	1411	75	0						-1					
410	10	38	1412	70	0										1	
410		39	1422	80												
410	3	40	1426	77	0						-1					
410	5	41	1474	75	0						1				1	
410	7	42	1476	73	0						1				0	
410	7	43	1477	73	0						1					
410	8	44	1480	72	0						-1				-1	
410	7	45	1481	73	0						-1				-1	
410	5	46	1494	75	0						1				1	
410	6	47	1366	74									1			
410	2	48	1306	78	0											
410	7	49	1305	73	-1						-1				-1	
410	7	50	1307	73	0						-1				-1	
410	10	51	1372	70							-1				-1	
410	8	52	1356	72	0				0		-1				-1	

Kuva 16, Pivot-tili ilman mitään muotoilua tai järjestelyä

Lopuksi makro muotoilee taulukon hienommaksi ja luo ehdolliset muotoilut. Taulukon arvoilla on itse tehty muotoilu (0;0;"ok"), joka näyttää negatiiviset numerot positiivisina, ja nollan ok:na. Näin -1 näyttää 1, mutta punaisella pohjalla, koska se on myöhässä (kuva 18).

CO	Yht	Pos.	ID	DM	JaRe	KaRi	KiHa	LaVi	MiIn	JuLs	JoAa	JoSu	RiKa	VeMi	YrEI	IrLa
410	8	28	1358		ok			1			ok				ok	ok
410	11	29	1350		ok						ok				ok	ok
410	7	30	1472	1							ok				ok	ok
410	7	31	1473	1							ok				ok	ok
410	10	32	1348		ok	ok				ok	ok					
410	10	33	1349		ok	ok				ok	ok					
410	8	34	1359		ok						ok				ok	ok
410	4	35	1351		ok						ok					
410	4	36	1388		ok					1	ok					
410	5	37	1411		ok						ok				ok	
410	10	38	1412	2	ok						ok					
410		39	1422													
410	3	40	1426		ok						ok					

Kuva 17, Pivot-tili muotoiltuna ja järjestettynä

### 6.3 Infolaatikkomakro

Kuittauksen Seurannassa on myös makro, joka aktivoituu, kun käyttäjä valitsee solun pivot-taulukon sisältä. Makro luo laatikon ja sijoittaa sen valitun solun viereen. Laatikon sisälle se poimii tietoja valitun ID:n ja vastuuhenkilön kohdalta, kuten tehtävien kuvauksen ja kuittauksen statuksen (kuva 19).

Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	
MANUFACTURING													P.	QUALITY
JoAa	JoSu	RiKa	VeMi	YrEl	IrLa	AnFa	HeJa	IvSa	JuPo	OuKo	PaOj	PaOl	Ri	
	1	ok									ok		1	
	ok	2									ok			
	1	1									ok			
	ok	2									1			
	ok	2									1			
	ok	2									1			

RiKa NHRF-1517  
S535M1036103\_Side\_Panel\_Rear\_RH\_leikkuriohjelman\_korjaus  
(Click to follow)

ERP rakenne/työpapereiden päivitys  
Due Date: 5.6.2009  
A-TO ohjeiden päivitys  
Signed: ok 22.5.2009  
FC-50...FC-621:1 kuvat  
Signed: Ok 13.06.2009

Kuva 18, Infolaatikko, joka tulee esiin, kun valitsee solun

## 7 LOPPUPÄÄTELMIÄ

Ohjelma otettiin käyttöön AST:ssä, mikä kertoo siitä, että siitä on hyötyä toimeksiantajalle. Kiitokset yritykselle ennakkoluulottomasta suhtautumisesta harjoittelijan esittämiin ideoihin.

Ohjelman käyttäjiltä on myös tullut paljon positiivista palautetta, sillä omien tehtävien seuraaminen ja kuittausprosessi on paljon helpompaa, kuin ennen. Sovellus mahdollistaa muutosten seuraamisen myös henkilöille, jotka eivät jatkuvasti käytä Enoviaa.

Ohjelmaa on nyt testattu lähes vuoden verran eikä suurempia ongelmia ole havaittu. Jos puutteita ilmenee, niin on sovittu, että tutkintotyön tekijä käy AST:n pyynnöstä muokkaamassa ohjelmaa näiden poistamiseksi.

Ohjelman laatimisessa on ollut suurena apuna ohjekirjat ”Excel VBA Programming for Dummies”, sekä ”Excel 2003 Power Programming with VBA”.

Liitteenä on ohjelmassa käytetyt makrot niitä varten, jotka ovat VBA-ohjelmoinnista kiinnostuneita.

# LÄHTEET

## Painetut lähteet

1. Sääksvuori, Antti – Immonen, Anselmi, Tuotehallinta PDM. Talentum Media Oy. Helsinki 2002. 201 s.
2. Patria: Vuosikatsaus 2008. Patria Oyj. 27 s.
3. Walkenbach, John, Excel VBA Programming for Dummies ,Wiley Publishing, Inc., Indianapolis, Indiana. 2004. 391 s.
4. Walkenbach, John, Excel 2003 Power Programming with VBA, Wiley Publishing, Inc., Indianapolis, Indiana. 2004. 1018 s.
5. Innala M., Kallioinen J., Saarinen T.: CAE/PLM-kyselyn tulokset, Valokynä-lehti 2/2009, CAD/CAM-yhdistys ry, 80 s.

## Painamattomat lähteet

6. Patria Aerostructures, Muutoshallinta. Procedure A-PR-016-006. 2004. 25 s.
7. Patria, AST Muutoshallinta. Uudistetut toimintatavat 26.8.2008. Doc ID: AST-0209. 5 s.

## Sähköiset lähteet

8. Patria Aviation – Wikipedia. [www.sivu]. [viitattu 20.8.2009]. Saatavissa: <http://fi.wikipedia.org/wiki/Patria-Aviation>
9. Patria. [www.sivu]. [viitattu 22.8.2009] Saatavissa: <http://www.patria.fi>
10. Tuotetiedon hallinta – Mitä se on? Variantum Oy. [www-sivu].[viitattu 1.10.2009]. Saatavissa: [www.automationit.hut.fi/file.php?id=704](http://www.automationit.hut.fi/file.php?id=704) –
11. Tuotetiedon hallinta – Wikipedia [www.sivu]. [viitattu 1.10.2009]. Saatavissa: [http://fi.wikipedia.org/wiki/Tuotetiedon\\_hallinta](http://fi.wikipedia.org/wiki/Tuotetiedon_hallinta)

## **LITTEET**

1. Kuittauksen Seurannan Pikakäyttöohje. 1 s.
2. Kuittauksen Seuranta instructions. 5 s.
3. Kuittauksen Seurannan makrojen VBA koodi. 25 s.

# Kuittauksen Seurannan Pikakäyttöohje

U:\Muutoshallinta\_ChangeManagement\[Hankekansio]\[ID]\_Kuittauksen\_Seuranta.xls

1. Avaa taulukko ja **salli makrot**. Voit avata sen myös lukutilassa, mikäli se on jollain toisella auki. Lukutila ei vaikuta taulukon toimivuuteen. Vasemmalla puoliskolla on listattu avonaiset muutokset ja niiden tietoja (ID, Subject, jne.). Oikealla on lista vastuuhenkilöistä ja heidän actionit.
2. Etsi nimilyhenteesi. Samalla sarakkeella on kaikki tehtäväsi. ”ok” tarkoittaa, että olet jo kuitannut tehtävän. Numero tarkoittaa, että niin monta tehtävää on vielä kuittaamatta.
3. Voit tarkastella tehtäviäsi valitsemalla numeroidun solun. Musta apulaatikko näyttää tarkempia tietoja ja apulaatikkoa klikkaamalla pääsee suoraan muutoslomakkeeseen.
4. Jos haluat kuitata, kuittaa tehdyt tehtävät **muutoslomakkeeseen** kirjoittamalla Signature -sarakkeeseen OK

**HUOM!** Seurantataulukko päivittyy automaattisesti jos kuittaa muutoslomakkeen. Seurantataulukon ei tarvitse itse täyttää mitään.

Lista avonaisista muutoksista.  
Hyperlinkistä pääsee suoraan muutokseen.

Nimi- ja osastolista. Jos nimeäsi ei näy listalla, sinulla ei ole tällä hetkellä kuittaamattomia tehtäviä kyseisessä hankkeessa!

Esim. MiSo:lla on kaksi kuittaamatonta tehtävää muutoksessa NHRF-1350, joista ainakin toinen on myöhässä

Numero = Kuittaamattomien tehtävien lukumäärä

Vihreä	kaikki tehtävät kuitattu
Keltainen	kuittaamattomia, mutta aikataulussa
Punainen	kuittaamattomia, ainakin yksi myöhässä

Kun solun valitsee, pongahtaa esiin apulaatikko, joka listaa kaikki kyseisen henkilön actionit ja niiden statukset kyseisessä muutoksessa.

Vihreä	kuitattu
Keltainen	kuittaamatta, aikataulussa
Punainen	kuittaamatta, myöhässä.

Laatikkoa klikkaamalla pääsee suoraan lomakkeeseen



On the left side is the list of names, abbreviations and departments which are assigned to names. On the right is the list of all the possible departments and the order in which they appear in Seuranta (from left to right, so in this case CONFIGURATION is the left-most and OTHER is the rightmost department in Seuranta.)

Scroll down the list and add a new name and assign a department from the possible choices. Here you can also change a department for an existing person if needed.

Nimi	Lyh.	Osasto	Järjestys	Osastolista
Name	Abr.	Department	Order	List of Departments
	AnAM	DESIGN	1	CONFIGURATION
	AnFa	QUALITY	2	MANUFACTURING
	ArRi	DESIGN	3	TOOLING
	ArTa	MANUFACTURING	4	P. CONTROL
	ArKa	MANUFACTURING	5	QUALITY
	ArDa	MANUFACTURING	6	PROCUREMENT
	Elhe	MANUFACTURING	7	DESIGN
	ErHa	DESIGN	8	PRODUCTION
	ErPe	PROCUREMENT	9	PROGRAM
	EsPa	DESIGN	10	
	HaOl	DESIGN	11	
	HeJa	QUALITY	12	
	HeNo	PROGRAM	13	
	HeVi	PROCUREMENT	14	
	IIMa	PROGRAM	15	
	IrLa	P. CONTROL	16	
	IvSa	QUALITY	17	
	JaRe	CONFIGURATION	18	
	JaKo	DESIGN	19	
	JaPe	DESIGN	20	OTHER
	JaKy	PRODUCTION		
	VeMi	MANUFACTURING		
	VeTo	DESIGN		
	VeHe	QUALITY		
	YrEI	MANUFACTURING		
	JoKu	PROGRAM		

When you go back to Seuranta, the name automatically goes under right department.

PROCUREMENT	DESI	PROGRAM		OTHE				
ErPe	MaTa	SoLa	AnAM	HeN	JoKu	kaSa	PaRo	Paula
	ok				1			

### Names in OTHER area that shouldn't be there

If a name is already in the Names List and it has been assigned a department other than OTHER, but it still seems to go into the OTHER area, then it might be the case that in the Change Template the name is incorrectly typed in the cell. Be sure that there is max one name per cell and that there are no alt-enters before or after the name.

## Adding a new department

If you want to add a new department to the list, open AST-0288 and type the new department in List of Departments. The order of the departments in Seuranta (from left to right) is the same as in this list from top to bottom, so arrange the new department where you want it to show in Seuranta. For instance, if you want the new department to be the leftmost Department in Seuranta, then put it first in this list.

	A	B	C	D	E	F	G
1	Nimi Name	Lyh. Abr.	Osasto Department		Järjestys Order	Osastolista List of Departments	
2		AnAM	DESIGN		1	CONFIGURATION	
3		AnFa	QUALITY		2	MANUFACTURING	
4		ArRi	DESIGN		3	TOOLING	
5		ArTa	MANUFACTURING		4	P. CONTROL	
6		ArKa	MANUFACTURING		5	QUALITY	
7		ArDa	MANUFACTURING		6	PROCUREMENT	
8		Elhe	MANUFACTURING		7	DESIGN	
9		ErHa	DESIGN		8	PRODUCTION	
10		ErPe	PROCUREMENT		9	PROGRAM	
11		EsPa	DESIGN		10	NEW DEPARTMENT	
12		HaOl	DESIGN		11		
13		HeJa	QUALITY		12		
14		HeNo	PROGRAM		13		
15		HeVi	PROCUREMENT		14		
16		IlMa	PROGRAM		15		
17		IrLa	P. CONTROL		16		
18		IvSa	QUALITY		17		
19		JaRe	CONFIGURATION		18		
20		JaKo	DESIGN		19		
21		JaPe	DESIGN		20	OTHER	
22		JaKy	PRODUCTION				

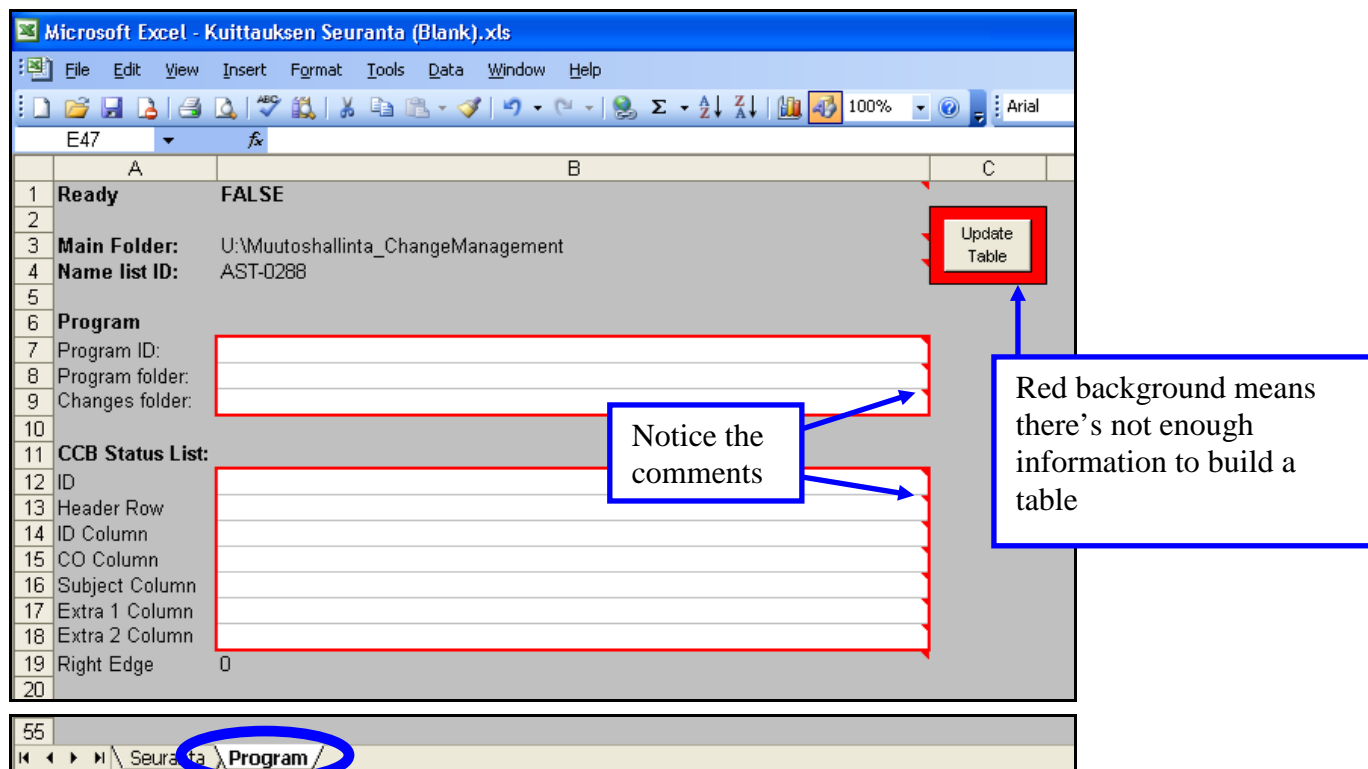
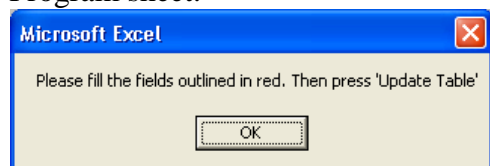
After you have added the new department, you can assign people to it.

88		VeTo	DESIGN	
89		VeHe	QUALITY	
90		YrEI	MANUFACTURING	
91		JoKu	NEW DEPARTMENT	
92			P. CONTROL	
93			QUALITY	
94			PROCUREMENT	
95			DESIGN	
96			PRODUCTION	
97			PROGRAM	
98			NEW DEPARTMENT	
99			OTHER	
100				

PROCUREMENT	DESI	PROGRAM	NEW	OTHE
ErPe	MaTa	SoLa	AnAM	HeNo
			KaSa	PaR
			JoKu	Paula
	ok			
			1	

## How to configure Kuittauksen Seuranta for a new program

Get **AST-0295** (Blank Seuranta) from Enovia and open it. It will give this warning message and open the Program sheet.



Fill the red bordered fields with correct data.

Program info:

- **Program ID** means the ID that Enovia Uses for documents.  
Example: *NHRF*
- **Program folder** is the folder path where the Seuranta and CCB-status lists are.  
Example: *U:\Muutoshallinta\_ChangeManagement\NH\_NH90\_Rear\_Fuselage*
- **Changes folder** is the folder path where all the Change Templates are.  
Example: *U:\Muutoshallinta\_ChangeManagement\NH\_NH90\_Rear\_Fuselage\Muutokset*

CCB Status List info:

- **ID** is the ID part of the CCB Status list. No revision or the full name.  
Example: *NHRF-1043*
- **Header row** is the number of the row in CCB Status list that contains the headers  
Example: *4*
- **ID Column**, **CO Column** and the **Subject Column** needs the relevant column numbers. The number is the order of the column from left to right, so for example Column C is 3.
- **Extra 1** and **Extra 2** columns are not important for table to function so they can be any columns. They can not be left blank however.

See example on the next page

Example:

Ready to build the table

For extra columns “Effectivity“ and “Owner” were chosen, but they could have been something other as well.

1	2	3	4	5	6	7	8	9	10	11	
			ID	CO			Subject	Extra 1	Extra 2		
1	NHRF-1043	Issue 30									
2	NH90RF Change Control Board Status										
3											
4	CR	CR opened	Status	NHRF-	CO	CO opened	CO status	Subject	Effectivity	Owner	Actions
5	175	28.11.2007	Approved	1061	191	28.11.2007	Proposed	3 Cover 2 LH	TFIA20		JoAa, MiSa,
6	175	28.11.2007	Approved	1107	191	28.11.2007	Proposed	2 Fairing	ASN39		JoAa, MiSa,
7	175	28.11.2007	Approved	1108	191	28.11.2007	Proposed	2 Fairing	ASN39		JoAa, Lere, f
8				1078	194	5.12.2007	Proposed	Perärungon mu	ASN 031		Miln, JaRe, J
9				1102	202	3.1.2008	Proposed	TN S535M0501	E01 General requirements		JaRe, JoJu
10				1135	212	22.2.2008	Proposed	S516M0103E01	ASN40		PaOj
11	175	28.11.2007	Approved	1135	191	28.11.2007	Proposed	S556M103051	COMPL CDF PANEL EP		JoAa, G-M

Header Row (4)

After the all the fields are filled, press the Update Table button to build a table.

### Editing the Program sheet for existing programs

The info in Program sheet can be edited if something changes. For example if a new row is added above the Header row in NHRF-1043, then the row number must be changed from 4 to 5 to work properly.

Also at any time Extra columns can be changed to something else if wanted.

Sheet1

Option Explicit

```
Private Sub Worksheet_Activate()  
    Call Seuranta_Activate  
End Sub
```

```
Private Sub Worksheet_SelectionChange(ByVal Target As Range)  
    Call Main  
    Call InfoBoksi(Target)  
End Sub
```

```
Private Sub Worksheet_PivotTableUpdate(ByVal Target As PivotTable)  
    Application.EnableEvents = False  
    Call Main
```

```
Application.ScreenUpdating = False  
Dim rSolu As Range  
Set rSolu = Selection
```

```
With shSeuranta.Columns("J:IV")  
    .FormatConditions.Delete  
End With  
Call DataAlue_Formaus  
Call OsastoRivi_formaus  
Call NamesRivi_Formaus  
Call HarmaaSolut
```

```
SPTable.TableRange1.BorderAround LineStyle:=xlContinuous, Weight:=xlThin  
rSolu.Select
```

```
Application.ScreenUpdating = True  
Application.EnableEvents = True  
End Sub
```

```
ThisWorkbook
```

```
Option Explicit
```

```
Private Sub Workbook_Activate()
```

```
    If ActiveSheet.Name = "Seuranta" Then Call Seuranta_Activate  
End Sub
```

```
Private Sub Workbook_BeforeClose(Cancel As Boolean)
```

```
    On Error Resume Next  
    shSeuranta.Shapes("boxi").Delete  
    On Error GoTo 0  
    Call Nimien_Formauksen_Palautus  
    Call Boldauksen_Poisto
```

```
    Application.DisplayAlerts = True  
    Application.EnableEvents = True  
    Application.ScreenUpdating = True
```

```
End Sub
```

```
Private Sub Workbook_Open()
```

```
    ThisWorkbook.UpdateLinks = xlUpdateLinksAlways  
    Application.DisplayAlerts = False  
    Application.EnableEvents = False  
    Application.ScreenUpdating = False  
    Call Main
```

```
    If Valmis = False Then Exit Sub
```

```
    Call Update_LinkRev
```

```
    If ActiveSheet.Name = "Seuranta" Then  
        Call Seuranta_Activate
```

```
    Else
```

```
        Worksheets("Seuranta").Activate  
        Call Seuranta_Activate
```

```
    End If
```

```
    Call Hyperlinkitys
```

```
    Application.DisplayAlerts = True
```

```
End Sub
```

Alku

Option Explicit

```
' Public muuttujat

' WorkBook ja sivut (osa sivuista on piilotettu)
  Public KS As Workbook
  Public shSeuranta As Worksheet
  Public shProgram As Worksheet
  Public shData As Worksheet
  Public shPiilo As Worksheet
  Public shDataPivot As Worksheet
  Public shOsastot As Worksheet

' Yleiset
  Public sKansio As String
  Public sNimiListaID As String
  Public bVirheitä As Boolean
  Public sVirheLista As String
  Public bTuplia As Boolean
  Public sTuplaLista As String
  Public bValmis As Boolean

' Tähän hankkeeseen liittyvät muuttujat
  Public sHanke As String
  Public sStatusID As String
  Public sHankeKansio As String
  Public sLomakeKansio As String

' Seurantasivun muuttujat, S = Seurantasivu
  Public Const iSSumSar As Integer = 1
  Public Const iSIdSar As Integer = 2
  Public Const iSSubjectSar As Integer = 3
  Public Const iSOikeaSar As Integer = 6
  Public Const iSTuplatSar As Integer = 7
  Public Const iSotsikkoRivi As Integer = 5

  Public rSeurantaIDt As Range

' Datasivun muuttujat, D = Datasivu
  Public Const iDDataIDtSar As Integer = 1
  Public Const iDTotalSar As Integer = 2
  Public Const iDIDSar As Integer = 3
  Public Const iDNameSar As Integer = 4
  Public Const iDOsastoSar As Integer = 5
  Public Const iDPivotDataSar As Integer = 6
  Public Const iDDateSar As Integer = 7
  Public Const iDEASar As Integer = 8
  Public Const iDDueDateSar As Integer = 9
  Public Const iDSignatureSar As Integer = 10

  Public rDataIDt As Range

' Piilosivun muuttujat, P = Piilosivu
  Public Const iPRivejä As Integer = 80 ' <-- noin monta riviä kopsataan dataa per
Muutoslomake
  Public Const iPSarakkeita As Integer = 7
  Public Const iPVikaSar As Integer = 9

' Seurannan Pivotin muuttujat
  Public SPTable As PivotTable
  Public SPFieldID As PivotField
  Public SPFieldNames As PivotField
  Public SpFieldOsasto As PivotField
```

```

' Datan Pivotin muuttujat
  Public DPTable As PivotTable
  Public DPFieldID As PivotField

  Public rPDataAlue As Range

Sub Main()

' Annetaan muuttujille arvot. Ajetaan ensimmäisenä

' Workbook ja sivut
  Set KS = ThisWorkbook
  Set shSeuranta = KS.Worksheets("Seuranta")
  Set shOsastot = KS.Worksheets("Osastot")
  Set shProgram = KS.Worksheets("Program")
  Set shData = KS.Worksheets("Data")
  Set shPiilo = KS.Worksheets("Piilo")
  Set shDataPivot = KS.Worksheets("DataPivot")

' Program-sivun tiedot
' If lauseet katoo päättyykö kansiopolut "\" merkkiin vaiko eivät
  With shProgram
    If Right(.Range("B3").Text, 1) = "\" Then
      sKansio = .Range("B3").Text
    Else
      sKansio = .Range("B3").Text & "\"
    End If
    If Right(.Range("B8").Text, 1) = "\" Then
      sHankeKansio = .Range("B8").Text
    Else
      sHankeKansio = .Range("B8").Text & "\"
    End If
    If Right(.Range("B9").Text, 1) = "\" Then
      sLomakeKansio = .Range("B9").Text
    Else
      sLomakeKansio = .Range("B9").Text & "\"
    End If
    sNimiListaID = .Range("B4").Text
    sHanke = .Range("B7").Text
    sStatusID = .Range("B12").Text
  End With

' Seurannan Pivot
  Set SPTable = shSeuranta.PivotTables("Pivot")
  Set SPFieldID = SPTable.PivotFields("ID")
  Set SPFieldNames = SPTable.PivotFields("Name")
  Set SpFieldOsasto = SPTable.PivotFields("Osasto")
' Datan Pivot
  Set DPTable = shDataPivot.PivotTables("DataPivot")
  Set DPFieldID = DPTable.PivotFields("ID")

End Sub

Sub Seuranta_Activate()

' Ajetaan aina, kun Seurantasivu aktivoidaan

  Application.ScreenUpdating = False
  Application.EnableEvents = False
  Call Main
  If Valmis = False Then Exit Sub
  Call Tarkista_Jarjestys
  Call Hyperlinkitys

```

```

    Call Formaus
Application.EnableEvents = True
    SPTable.RefreshTable
    DPTable.RefreshTable
    Call Formaus
Application.ScreenUpdating = True

End Sub

Sub Button()

' Kun "Update Table"-nappia painetaan
Application.EnableEvents = False
Application.ScreenUpdating = False

Call Main
If Valmis = False Then Exit Sub
Call Update_LinkRev
Call Data_Alustus
Call Data_Muutoslistoista
Call Data_Pivottiin
Call Korjaa_Jarjestys

DPTable.RefreshTable
Call Seuranta_Activate

If bVirheitä = False Then
    MsgBox "Table updated successfully"
Else
    MsgBox "The following documents were open in other computers and could not be updated:"
& sVirheLista
End If

End Sub

Sub Formaus()

Dim iAlinRivi As Integer

Call Nimien_Formauksen_Palautus
Call Boldauksen_Poisto

With shSeuranta
    With .Columns(iSOikeaSar).Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThick
        .ColorIndex = 56
    End With

    .Range(Cells(iSOtsikkoRivi, 1), Cells(65536,
iSOikeaSar)).Borders(xlInsideHorizontal).LineStyle = xlNone
    On Error Resume Next
        iAlinRivi = SPFieldID.DataRange.End(xlDown).Row
        .Range(.Cells(iAlinRivi, 1), .Cells(iAlinRivi,
iSOikeaSar)).Borders(xlEdgeBottom).LineStyle = xlContinuous
    On Error GoTo 0
        .Cells(iSOtsikkoRivi, iSOikeaSar).Borders(xlEdgeRight).ColorIndex = 2
    End With

End Sub

' Päivittää linkit, mikäli tiedostojen revisio on muuttunut
Dim vLinkit As Variant

Dim vNyk_StatusLinkki As Variant
Dim vNyk_NimiLinkki As Variant

```

```

Dim vUusi_StatusLinkki As Variant
Dim vUusi_NimiLinkki As Variant

Dim i As Integer
Dim sStatusLista As String
Dim sNimiLista As String
Dim sVanhaTiedosto As String
Dim sID As String
Dim sUusiLinkki As String
Dim sStatusAlue As String
Dim sNimiAlue As String
Dim sOsastoAlue As String

ChDir sHankeKansio

vLinkit = KS.LinkSources(xlExcelLinks)
If Not IsEmpty(vLinkit) Then
For i = 1 To UBound(vLinkit)

    If vLinkit(i) Like "*" & sStatusID & "*" Then
        vNyk_StatusLinkki = vLinkit(i)
    ElseIf vLinkit(i) Like "*" & sNimiListaID & "*" Then
        vNyk_NimiLinkki = vLinkit(i)
    Else
        sVanhaTiedosto = Mid(vLinkit(i), InStrRev(vLinkit(i), "\") + 1)
        sID = Left(sVanhaTiedosto, InStr(sVanhaTiedosto, "_"))
        sUusiLinkki = Dir(sLomakeKansio & sID & "*")
        If sUusiLinkki <> "" Then
            sUusiLinkki = sLomakeKansio & sUusiLinkki
            If vLinkit(i) <> sUusiLinkki Then
                'Debug.Print vLinkit(i)
                ThisWorkbook.ChangeLink Name:=vLinkit(i), NewName:=sUusiLinkki,
Type:=xlExcelLinks
                'Debug.Print sUusiLinkki
            End If
        End If
    End If
End If
Next i
End If

sStatusLista = Dir(sHankeKansio & sStatusID & "*")
vUusi_StatusLinkki = sHankeKansio & sStatusLista

sNimiLista = Dir(sKansio & sNimiListaID & "*")
vUusi_NimiLinkki = sKansio & sNimiLista

If Not IsEmpty(vNyk_StatusLinkki) And vNyk_StatusLinkki <> vUusi_StatusLinkki Then
    ThisWorkbook.ChangeLink Name:=vNyk_StatusLinkki, NewName:=vUusi_StatusLinkki,
Type:=xlExcelLinks
End If

If Not IsEmpty(vNyk_NimiLinkki) And vNyk_NimiLinkki <> vUusi_NimiLinkki Then
    ThisWorkbook.ChangeLink Name:=vNyk_NimiLinkki, NewName:=vUusi_NimiLinkki,
Type:=xlExcelLinks
End If

Dim iRivi As Integer
Dim iVasen As Integer
Dim iOikea As Integer

iRivi = shProgram.Range("B13").Value
iOikea = shProgram.Range("B19").Value
sStatusAlue = "=" & sHankeKansio & "[" & sStatusLista & "]"CR!R" & iRivi & "C1:R" & iRivi
+ 300 & "C" & iOikea
sNimiAlue = "=" & sKansio & "[" & sNimiLista & "]"Osastot!$A$1:$C$500"

```

```
sOsastoAlue = "=" & sKansio & "[" & sNimiLista & "]Osastot'!$E$1:$F$21"
```

```
KS.Names("StatusLista").RefersTo = sStatusAlue
```

```
KS.Names("NimiLista").RefersTo = sNimiAlue
```

```
KS.Names("OS_Order").RefersTo = sOsastoAlue
```

```
shSeuranta.Hyperlinks.Add anchor:=shSeuranta.Range("A3"), Address:=vUusi_StatusLinkki
```

```
shSeuranta.Hyperlinks.Add anchor:=shSeuranta.Range("A4"), Address:=vUusi_NimiLinkki
```

```
End Sub
```

Apulaatikko

Option Explicit  
Option Compare Text

Private rNimiSolu As Range  
Private rIDAlue As Range  
Private iRiviNum As Integer  
Private iSarNum As Integer

Sub InfoBoksi(Target As Range)

' Muuttujia

Dim rSolu As Range  
Dim snPosLeft As Single  
Dim snPosTop As Single  
Dim sTeksti() As String  
Dim sTeksti\_osissa() As String  
Dim sNimi As String  
Dim sID As String  
Dim sHyperLinkkiOs As String  
Dim i As Integer  
Dim j As Integer  
Dim iOsat As Integer  
Dim shapeBoxi As Shape  
Dim rSijoitusSolu As Range  
Dim rNäkyväAla As Range  
Dim snVaseniRaja As Single  
Dim snOikeaiRaja As Single  
Dim snAlaiRaja As Single

Set rSolu = Target  
Set rPDataAlue = SPTable.DataBodyRange

On Error Resume Next

' Vanhan Boxin poisto  
shSeuranta.Shapes("boxi").Delete

' Solujen palautus alkuperäisiksi, jos valitsee eri solun

If Intersect(Target, rPDataAlue) Is Nothing Or iSarNum <> rSolu.Column Then  
Call Nimien\_Formauksen\_Palautus  
End If

If Intersect(Target, rPDataAlue) Is Nothing Or iRiviNum <> rSolu.Row Then  
Call Boldauksen\_Poisto  
End If

On Error GoTo 0

' Onko valinta kohdallaan?

On Error GoTo Virheet:

If Target.Count = 1 And Not Intersect(Target, rPDataAlue) Is Nothing Then

On Error GoTo 0

' Tunnistussolujen muokkaus

Set rNimiSolu = Cells(iSotsikkoRivi, rSolu.Column)

With rNimiSolu

.Interior.ColorIndex = 2

.Font.ColorIndex = 0

End With

Set rIDAlue = Range(Cells(rSolu.Row, 1), Cells(rSolu.Row, 6))

rIDAlue.Font.Bold = True

' BOXI:

If Target.Value <> "" Then

snPosLeft = rSolu.Left + rSolu.Width + 5

```

snPosTop = rSolu.Top + rSolu.Height + 5
sNimi = rSolu.PivotCell.ColumnItems(1)

' Tekstin muodostus kutsuu tekstitys metodin
sTeksti() = Tekstitys(rSolu, sNimi)

' Boksiin pystyy pasteamaan vain 256 merkkiä kerralla. Siksi teksti on pilkottu
useampaan osaan.
' iOsat kertoo kuinka moneen
iOsat = Int(Len(sTeksti(0, 0, 0)) / 255) + 1

' Seuraava looppo Pilkkoo tekstin
ReDim sTeksti_osissa(1 To iOsat)
j = 1

For i = 1 To iOsat
    sTeksti_osissa(i) = Mid(sTeksti(0, 0, 0), j, 255)
    j = j + 255
Next

' Hyperlinkin osoite
On Error Resume Next
    sHyperLinkkiOs = Cells(rSolu.Row, iSidSar).Hyperlinks(1).Address
On Error GoTo 0

' Tekstiboksin lisäys
With shSeuranta.Shapes.AddShape(msoShapeRoundedRectangle, 0, 0, 0, 0)
    .Visible = False
    .Name = "boxi"
    .Fill.ForeColor.SchemeColor = 63
    .Fill.Transparency = 0.15
    .Line.Weight = 1.2
    With .TextFrame
        ' Seuraava looppo kokoaa tekstin osista
        j = 0
        For i = 1 To iOsat
            If i = 1 Then
                .Characters.Text = sTeksti_osissa(i)
            Else
                .Characters(j).Insert (sTeksti_osissa(i))
            End If
            j = j + 255
        Next
        ' Muotoilua
        .Characters.Font.ColorIndex = 2
        .Characters(1, InStr(sTeksti(0, 0, 0), Chr(10))).Font.Bold = True
        With .Characters(InStr(1,
shSeuranta.Shapes("boxi").TextFrame.Characters.Text, "(Click
to follow)"), 17)
            .Font.Size = 8
            .Font.ColorIndex = 37
        End With

        ' Muotoilee tekstiä. Matriisi kertoo miltä alueelta muotoillaan
        For i = 1 To sTeksti(1, 0, 1)
            Select Case sTeksti(i, 0, 0)
                Case 0
                    With .Characters(sTeksti(i, 1, 0), sTeksti(i, 1, 1)).Font
                        .ColorIndex = 24
                        .Bold = True
                    End With
                Case -1
                    With .Characters(sTeksti(i, 1, 0), sTeksti(i, 1, 1)).Font
                        .ColorIndex = 46
                        .Bold = True
                    End With
            End Select
        Next
    End With
End With

```

```

        End With
    Case 1
        With .Characters(sTeksti(i, 1, 0), sTeksti(i, 1, 1)).Font
            .ColorIndex = 6
            .Bold = True
        End With
    Case Else
        .Characters(sTeksti(i, 1, 0), sTeksti(i, 1, 1)).Font.Bold = True
    End Select
Next
' Muuttaa boksen koon tekstin koon mukaan, mutta aiheuttaa ylileveän boksen jos
paljon
tekstiä.
        .AutoSize = True
    End With

    Set shapeBoxi = shSeuranta.Shapes("boxi")
    Set rNäkyväAla = ActiveWindow.VisibleRange

    ' Boksen Sijoitus
    snVaseniRaja = ActiveWindow.VisibleRange.Columns(1).Left
    snOikeaiRaja =
ActiveWindow.VisibleRange.Columns(ActiveWindow.VisibleRange.Columns.Count).Left
    snAlaiRaja =
ActiveWindow.VisibleRange.Rows(ActiveWindow.VisibleRange.Rows.Count).Top
    shapeBoxi.Left = snPosLeft
    shapeBoxi.Top = snPosTop

    ' SijoitusSolu kertoo, onko boksi katoamassa ruudun ulkopuolelle ja korjaa
sijaintia sen mukaan

    Set rSijoitusSolu = shapeBoxi.BottomRightCell
    If rSijoitusSolu.Top >= snAlaiRaja Then
        If rSijoitusSolu.Left >= snOikeaiRaja Then
            shapeBoxi.Top = rSolu.Top - shapeBoxi.Height - 5
        Else
            shapeBoxi.Top = snAlaiRaja - shapeBoxi.Height - 5
        End If
    End If
    If rSijoitusSolu.Left >= snOikeaiRaja Then
        shapeBoxi.Left = snOikeaiRaja - shapeBoxi.Width - 5
    End If
    If shapeBoxi.Left < snVaseniRaja Then
        shapeBoxi.Left = snVaseniRaja + 5
    End If
    shapeBoxi.Visible = True
End With

    ' Hyperlinkitys
    shSeuranta.Hyperlinks.Add anchor:=shSeuranta.Shapes("boxi"),
Address:=sHyperLinkkiOs
    End If
    End If

    iRiviNum = rSolu.Row
    iSarNum = rSolu.Column
    Exit Sub

Virheet:
    Exit Sub
End Sub

Function Tekstitys(rSolu As Range, sNimi As String) As Variant

    ' Tämä matriisihässäkkä kokoaa boksiin kirjoitettavan tekstin. Anteeksi.

```

```

Dim rDataSolul As Range
Dim rDataSolu2 As Range
Dim sTekstiArray(100, 1, 1) As String
Dim sKokoID As String
Dim sID As String
Dim i As Integer
Dim iRaja As Integer

Set rDataIDt = shData.Columns(iDDataIDtSar).SpecialCells(xlCellTypeConstants)

i = 1
sNimi = rSolu.PivotCell.ColumnItems(2)
sTekstiArray(0, 0, 0) = sNimi

' (0, 0, 0) = itse teksti
' (0, x, x) = sijainteja ja ja tekstin pituuksia muotoilua varten

sID = rSolu.PivotCell.RowItems(1).Value
' Muodostaa boksiin kirjoitettavan ID:n riippuen onko CCB-statuslistalla pelkät numerot
vai hanke-numero
If IsNumeric(rSolu.PivotCell.RowItems(1).Value) Then
    sKokoID = sHanke & "-" & rSolu.PivotCell.RowItems(1).Value
Else
    sKokoID = rSolu.PivotCell.RowItems(1).Value
End If

sTekstiArray(0, 0, 0) = sTekstiArray(0, 0, 0) & "      " & sKokoID      'Itse sTeksti
sTekstiArray(0, 0, 1) = Len(sTekstiArray(0, 0, 0)) - 1                'Ekan rivin
pituus
sTekstiArray(0, 1, 0) = InStr(sTekstiArray(0, 0, 0), sHanke & "-" & sID)      'sID:n
sijainti
sTekstiArray(0, 1, 1) = Len(sTekstiArray(0, 0, 0)) - sTekstiArray(0, 1, 0) + 1 'sID:n
pituus

'Subject, sekä Click to Follow
sTekstiArray(0, 0, 0) = sTekstiArray(0, 0, 0) & Chr(10) & Cells(rSolu.Row(),
iSSubjectSar).Value & _
                Chr(10) & "(Click to follow)" & Chr(10)

' Alkaa käymään dataa läpi datasivulla ja kerää tietoja jos nimi ja id osuu kohdalle
For Each rDataSolul In rDataIDt
    If rDataSolul.Text = sID Then
        With shData
            For Each rDataSolu2 In .Range(.Cells(rDataSolul.Row, iDNameSar),
.Cells(rDataSolul.Row + iPRivejä - 1, iDNameSar))
                If rDataSolu2.Text = sNimi Then
                    sTekstiArray(0, 0, 0) = sTekstiArray(0, 0, 0) & Chr(10) &
.Cells(rDataSolu2.Row, iDEASar).Value & Chr(10)

                    If .Cells(rDataSolu2.Row, iDPivotDataSar).Value = 0 Then '0 = Kuitattu
                        sTekstiArray(i, 0, 0) = 0
                        sTekstiArray(0, 0, 0) = sTekstiArray(0, 0, 0) & "      Signed: " &
rDataSolu2.Offset(0, 6).Text
                        sTekstiArray(i, 1, 0) = InStrRev(sTekstiArray(0, 0, 0), "Signed:")
+ 7 'Chr(10))
                        'Pvm sijainti
                        sTekstiArray(i, 1, 1) = Len(sTekstiArray(0, 0, 0)) - sTekstiArray(i,
1, 0) + 1
                    Else
                        If .Cells(rDataSolu2.Row, iDDueDateSar).Value < Date Then '-1 =
Myöhässä ' If .Cells(rDataSolu2.Row, iDDueDateSar).Value < Date And .Cells(rDataSolu2.Row,
iDDueDateSar).Value <> 0 Then
                            sTekstiArray(i, 0, 0) = -1 '-1 = Myöhässä
                        Else
                            sTekstiArray(i, 0, 0) = 1 '1 = Aikataulussa

```

```

End If

' Antaa päivämäärän joko päivämääränä tai tekstinä, riippuen
lähtöarvosta
    If IsNumeric(.Cells(rDataSolu2.Row, iDDueDateSar).Value) And
.Cells(rDataSolu2.Row, iDDueDateSar).Value <> 0 Then
        sTekstiArray(0, 0, 0) = sTekstiArray(0, 0, 0) & " Due Date: " &
CDate(.Cells(rDataSolu2.Row, iDDueDateSar).Value)
    ElseIf .Cells(rDataSolu2.Row, iDDueDateSar).Value = 0 Then
        sTekstiArray(0, 0, 0) = sTekstiArray(0, 0, 0) & " Due Date: -"
    Else
        sTekstiArray(0, 0, 0) = sTekstiArray(0, 0, 0) & " Due Date: " &
.Cells(rDataSolu2.Row, iDDueDateSar).Text
    End If
    sTekstiArray(i, 1, 0) = InStrRev(sTekstiArray(0, 0, 0), "Due Date: ")
+ 9
    'Pvm sijainti
        sTekstiArray(i, 1, 1) = Len(sTekstiArray(0, 0, 0)) - sTekstiArray(i,
1, 0) + 1
    'Pvm pituus
        End If
        i = i + 1
    End If
    Next rDataSolu2
    Exit For
End With
End If
Next rDataSolul

sTekstiArray(1, 0, 1) = i - 1 'määrä
Tekstitys = sTekstiArray
End Function

Sub Nimien_Formauksen_Palautus()
    With SPFieldNames.DataRange
        .Interior.ColorIndex = 56
        .Font.ColorIndex = 2
    End With
End Sub

Sub Boldauksen_Poisto()
    shSeuranta.Range(shSeuranta.Cells(iSOtsikkoRivi + 1, 1), shSeuranta.Cells(65536,
iSOikeaSar)).Font.Bold = False
End Sub

```

Datasivu

Option Explicit

' Muutosdokkareihin liittyvät muuttujat

```
Private CT As Workbook
Private CTetuSivu As Worksheet, CTPiilo As Worksheet
```

```
Private bSulje As Boolean
Private bTallenna As Boolean
Private iRivi As Integer
Private sDataID As String
Private sTiedosto As String
```

Sub Data\_Delete()

```
Call Main
With shData.Range(Rows(3), Rows(65536))
    .Clear
    .NumberFormat = "@"
End With
shData.Columns(iDTotalSar).NumberFormat = "General"
shData.Columns(iDOSastoSar).NumberFormat = "General"
shData.Columns(iDPivotDataSar).NumberFormat = "General"
```

End Sub

Sub Data\_Alustus()

' Tämä proseduuri alustaa datasivun vastaanottamaan dataa muutoslomakkeista.  
' Toisinsanoen poistaa vanhat IDt ja lisää uusia riippuen CCB-statuslistasta

' Muuttujat

```
Dim bJatka As Boolean
Dim sID As String
Dim sNimiSolu As String
Dim rSSolu As Range
Dim rDSolu As Range
Dim rDSolu2 As Range
Dim rUusiID As Range
Dim rPoistoAlue As Range
```

```
Set rSeurantaIDt = shSeuranta.Columns(iSIdSar).SpecialCells(xlFormulas)
With shData
    Set rDataIDt = .Columns(iDDataIDtSar).SpecialCells(xlCellTypeConstants)
    Set rPoistoAlue = .Rows(65536) ' <-- tarvitaan, jotta voi käyttää Union-metodia
```

' POISTO

' Tarkistaa poiston tarpeen, kerää poistettavat rivit ja poistaa ne lopuksi

```
bJatka = True
For Each rDSolu In rDataIDt
    If rDSolu.Row > 2 Then
        For Each rSSolu In rSeurantaIDt
            If rDSolu.Text = rSSolu.Text And rSSolu.Text <> 0 And rSSolu.Row >
isOtsikkoRivi Then
                bJatka = False 'ei poiston tarvetta, koska vastaava löytyi
                Exit For
            End If
        Next
    If bJatka = True Then
        sID = rDSolu.Text
        Set rDSolu2 = .Cells(rDSolu.Row, iDIDSar)
        Do While rDSolu2 = sID
            Set rPoistoAlue = Union(rPoistoAlue, rDSolu2.EntireRow)
```

```

        Set rDSolu2 = rDSolu2.Offset(1, 0)
    Loop
End If
bJatka = True
End If
Next
' Varsinainen poisto
rPoistoAlue.Delete
Set rDataIDt = .Columns(iDDataIDtSar).SpecialCells(xlCellTypeConstants)

' LISÄYS
' Lisää ID:n, jos on CCB-statuslistalla, mutta ei vielä lisätty Data-sivulle
For Each rSSolu In rSeurantaIDt
    If rSSolu.Text <> 0 And rSSolu.Row > iOtsikkoRivi Then
        For Each rDSolu In rDataIDt
            If rDSolu.Text = rSSolu.Text And rDSolu.Row > 2 Then
                bJatka = False 'ei lisäyksen tarvetta, koska vastaava löytyi
                Exit For
            End If
        Next
        If bJatka = True Then
            Set rUusiID = .Range(.Cells(1, iDIDSar).End(xlDown).Offset(1, -2),
                .Cells(1, iDIDSar).End(xlDown).Offset(1, 0))
            sNimiSolu = .Cells(rUusiID.Row, iDNameSar).Address(False, False)
            With rUusiID
                .Value = rSSolu.Text
                .Next.Formula = "=IF(ISBLANK(" & sNimiSolu & "),-1,IF(OR(" & sNimiSolu
& "=0," & sNimiSolu & "="" ""),0,1))"
                ' näyttää tältä: =IF(ISBLANK(D3);-1;IF(OR(D3=0;D3="");0;1))
            End With
        End If
    End If
    bJatka = True
Next
End With
Application.Calculate
End Sub

Sub Data_Muutoslistoista()

' Aloittaa datan hakemisen muutoslistoista ja kutsuu muita proseduureja tilanteen mukaan

Dim rDSolu As Range
bSulje = False
bTallenna = False
Set rDataIDt = shData.Columns(iDDataIDtSar).SpecialCells(xlCellTypeConstants)

With shData
    For Each rDSolu In rDataIDt
        sDataID = rDSolu.Text
        iRivi = rDSolu.Row
        If iRivi > 2 And Not IsEmpty(rDSolu) Then
            If .Cells(iRivi, iDTotalSar).Value < 0 Or IsEmpty(.Cells(iRivi, iDTotalSar))
Then
                Call Lomakkeen_Avaus
                bSulje = False
                bTallenna = False
            End If
        End If
    Next rDSolu
End With
Seuraava:
Next rDSolu
End With
End Sub

Sub Lomakkeen_Avaus()

```

```

' Avaa lomakkeen

Dim bItsellaAuki As Boolean
Dim bMuillaAuki As Boolean
Dim sPolku As String
Dim sID As String

With shData
    sTiedosto = Dir(sLomakeKansio & "*.xls")
    Do While sTiedosto <> ""
        sPolku = sLomakeKansio & sTiedosto
        sID = Left(Mid(sTiedosto, InStr(sTiedosto, "-") + 1), 4)
        If Len(sID) >= 4 And IsNumeric(sID) And sDataID Like "*" & sID Then
            bItsellaAuki = Check_if_Open_Me(sTiedosto)
            If Not bItsellaAuki Then
                bMuillaAuki = Check_if_Open_Others(sPolku)
                If bMuillaAuki = True Then
                    Workbooks.Open Filename:=sPolku, ReadOnly:=True
                Else
                    Workbooks.Open Filename:=sPolku, ReadOnly:=False
                End If
                bSulje = True
            End If

            Call Lomakkeen_Tila

            If bSulje = True Then
                Application.DisplayAlerts = False
                CT.Close SaveChanges:=bTallenna
                Application.DisplayAlerts = True
            End If
            Exit Do
        End If
        sTiedosto = Dir
    Loop
End With
End Sub

Sub Lomakkeen_Tila()

' Katsoo avattiinko lukutilassa vai normaalista
Dim bLukuTila As Boolean
Dim sTunnisteOsoite As String

Set CT = Workbooks(sTiedosto)
Set CTetuSivu = CT.Worksheets(1)
Set CTPiilo = CT.Worksheets(2)
bLukuTila = CT.ReadOnly

sTunnisteOsoite = shPiilo.Range("Tunniste").Address
With CTPiilo
    If .Range(sTunnisteOsoite).Text <> shPiilo.Range("Tunniste").Text Then
        If bLukuTila Then
            bVirheita = True
            sVirheLista = sVirheLista & Chr(10) & sTiedosto
            Exit Sub
        Else
            Call Piilon_Muokkaus
            bTallenna = True
        End If
    End If
End With

Call Datasivulle_Pastetus

```

End Sub

Sub Datasivulle\_Pastetus()

```
' Pasteaa tiedot muutoslomakkeen Piilo-välilehdeltä Seurannan Data-välilehdelle

Dim sNimiSolu As String
Dim sVrtSolu As String

With shData
  ' Lisää rivejä
  .Activate
  .Range(.Rows(iRivi + 1), .Rows(iRivi + iPRivejä - 1)).Insert Shift:=xlDown
  .Range(.Cells(iRivi, iDIDSar), .Cells(iRivi + iPRivejä - 1, iDIDSar)).Value = sDataID

  ' Copy Paste
  CTPiilo.Range(CTPiilo.Cells(2, 1), CTPiilo.Cells(iPRivejä + 1, iPSarakkeita)).Copy
  .Activate
  .Cells(iRivi, iDNameSar).Activate
  .Paste link:=True

  ' Total-sarakkeen kaavat
  .Cells(iRivi, iDTotalSar).Select
  Selection.AutoFill Destination:=.Range(Selection, Selection.Offset(iPRivejä - 1,
0))

  ' Osasto-sarakkeen kaavat
  .Cells(iRivi, iDOsastoSar).Select
  sNimiSolu = .Cells(Selection.Row, iDNameSar).Address(RowAbsolute:=False)
  Selection.Formula = "=IF(ISNA(VLOOKUP(" & sNimiSolu &
",Osastot!$B$1:$C$100,2,FALSE)), "OTHER",VLOOKUP(" & sNimiSolu &
",Osastot!$B$1:$C$100,2,FALSE))"
  ' näyttää tältä
=IF(ISNA(VLOOKUP($D3;Osastot!$B$1:$C$100;2;FALSE));"OTHER";VLOOKUP($D3;Osastot!$B$1:$C$
100;2;FALSE))
  Selection.AutoFill Destination:=.Range(Selection, Selection.Offset(iPRivejä - 1,
0))

  ' Pivot Data-sarakkeen kaavat
  .Cells(iRivi, iDPivotDataSar).Select
  sVrtSolu = Selection.Offset(0, 1).Address(RowAbsolute:=False)

  Selection.Formula = "=IF(" & sVrtSolu & "=0,0,IF(" & sVrtSolu & "<TODAY(),-1,1))"
  ' näyttää tältä: =IF($G3=0;0;IF($G3<TODAY();-1;1))
  Selection.AutoFill Destination:=.Range(Selection, Selection.Offset(iPRivejä - 1,
0))
End With
```

End Sub

Sub Piilon\_Muokkaus()

```
' Muokkaa muutoslomakkeen piilotettua Piilo-välilehteä

Dim i As Integer
Dim rStatusLista As Range
Dim rActionLista As Range

' nimettyjen solujen tyhjäys
For i = CT.Names.Count To 1 Step -1
  If Not (CT.Names(i).Name Like "*Print_Area") Then
    CT.Names(i).Delete
  End If
Next
```

```
'Piilon muokkaus
With CTPiilo
    .Visible = True
    .Activate
    ' postaa edellisen KS:n jäljiltä olevan varoitusboksen
    On Error Resume Next
        .Shapes().SelectAll
        Selection.Delete
    On Error GoTo 0
    Application.DisplayAlerts = False
    ' Kopioi Seurannan piilosivun muutoslomakkeeseen
    shPiilo.Cells.Copy Destination:=CTPiilo.Cells
    Application.DisplayAlerts = False
    Application.CutCopyMode = False

    Set rStatusLista = .Cells.Find(what:="Statuslista", searchorder:=xlByRows,
MatchCase:=True)
    Set rStatusLista = Range(rStatusLista.Offset(1, 0), rStatusLista.End(xlDown))

End With
With CTetuSivu
    .Names.Add Name:="status", RefersTo:=rStatusLista
    ' Muodostaa alueen, jolla seurattavat tiedot ovat
    Set rActionLista = .Cells.Find(what:="EFFECTS and ACTIONS", searchorder:=xlByRows,
MatchCase:=True).End(xlToLeft)
    Set rActionLista = .Range(rActionLista.Offset(iPrivejä, 0), rActionLista.Offset(0,
100).End(xlToLeft))
End With
With CTPiilo
    .Names.Add Name:="Taulu", RefersTo:=rActionLista
    .Range(.Cells(2, 1), Cells(1000, iPvikaSar)).NumberFormat = "General"
    .Range("A1").Select
    .Visible = False
End With
End Sub
```

Hyperlinkit

Option Explicit

Sub Hyperlinkitys()

Dim rSolu As Range

Dim sTiedosto As String

Dim sPolku As String

Dim sID As String

Set rSeurantaIDt = shSeuranta.Columns(iSidsar).SpecialCells(xlFormulas)  
rSeurantaIDt.Hyperlinks.Delete

sTiedosto = Dir(sLomakeKansio & "\*.xls")

Do While sTiedosto <> ""

    sPolku = sLomakeKansio & sTiedosto

    For Each rSolu In rSeurantaIDt

        If rSolu.Row > iSotsikkoRivi Then

            sID = Left(Mid(sTiedosto, InStr(sTiedosto, "-") + 1), 4)

            If Len(sID) = 4 And rSolu.Text Like "\*" & sID Then

                shSeuranta.Hyperlinks.Add anchor:=rSolu, Address:=sPolku

                Exit For

            End If

        End If

    Next rSolu

    sTiedosto = Dir

Loop

End Sub

## Metodeita

## Option Explicit

```
Function Check_if_Open_Me(Book As String) As Boolean
    ' Tää funktio tarkistaa onko Workbook auki, siten, että avonainen KS ei aiheuta virheilmoitusta.
```

```
    ' Ei ItsellaAvattu KS aiheuttaa virheen, jos sille yrittää jotain tehdä
```

```
    Dim sWBNimi As String
```

```
    On Error GoTo EiAuki
```

```
        sWBNimi = Workbooks(Book).Name
```

```
        Check_if_Open_Me = True
```

```
    Exit Function
```

```
EiAuki:
```

```
    Check_if_Open_Me = False
```

```
End Function
```

```
Function Check_if_Open_Others(Book As String)
```

```
    Dim iTiedostoNum As Integer
```

```
    Dim iErrNum As Integer
```

```
    On Error Resume Next          ' Turn error checking off.
```

```
    iTiedostoNum = FreeFile()    ' Get a free file number.
```

```
    ' Attempt to open the file and lock it.
```

```
    Open Book For Input Lock Read As #iTiedostoNum
```

```
    Close iTiedostoNum          ' Close the file.
```

```
    iErrNum = Err               ' Save the error number that occurred.
```

```
    On Error GoTo 0            ' Turn error checking back on.
```

```
    ' Check to see which error occurred.
```

```
    Select Case iErrNum
```

```
        ' No error occurred.
```

```
        ' File is NOT already open by another user.
```

```
    Case 0
```

```
        Check_if_Open_Others = False
```

```
        ' Error number for "Permission Denied."
```

```
        ' File is already opened by another user.
```

```
    Case 70
```

```
        Check_if_Open_Others = True
```

```
        ' Another error occurred.
```

```
    Case Else
```

```
        Error iErrNum
```

```
    End Select
```

```
End Function
```

```
Function Valmis()
```

```
    Dim rSolu As Range
```

```
    Valmis = True
```

```
    ' Tarkistaa onko tarpeeksi tietoja Seurantalomakkeen generoimiseen
```

```
    If shProgram.Range("B1").Value = False Then
```

```
        shProgram.Activate
```

```
        ' Jos ei tarpeeksi tietoja, muuttaa nimetyt alueet osoittamaan harmittomaan paikkaan
```

```
        KS.Names("StatusLista").RefersTo = "=Seuranta!$G$1"
```

```
        KS.Names("NimiLista").RefersTo = "=Seuranta!$G$1"
```

```
        KS.Names("OS_Order").RefersTo = "=Seuranta!$G$1"
```

```
    Application.ScreenUpdating = True
```

```
    MsgBox "Please fill the fields outlined in red. Then press 'Update Table'"
    Valmis = False
Else
    If Range("Tuplia").Value > 1 Then
        sTuplaLista = "There are multiple instances of the same ID in CCB Status list."
& vbNewLine & _
            "Fix the following ID's so that there are only one"

        On Error Resume Next
        For Each rSolu In shSeuranta.Columns(iSTuplatSar).SpecialCells(xlFormulas)
            If rSolu.Row > 1 And rSolu.Value > 1 Then
                sTuplaLista = sTuplaLista & Chr(10) & shSeuranta.Cells(rSolu.Row, iSIIdSar)
                Valmis = False
            End If
        Next

        If Valmis = False Then
            MsgBox sTuplaLista
        End If
        On Error GoTo 0
    End If
End If
End Function

Sub Tuplat()

End Sub
```

```
SeurantaPivot
```

```
Option Explicit
```

```
Sub Data_Pivottiin()
```

```
Dim pCache As PivotCache
Dim rVikaSolu As Range
Dim sSTaulu As String
Dim sDTaulu As String
```

```
' Cachen päivitys
```

```
shSeuranta.Activate
```

```
SPTable.PivotCache.MissingItemsLimit = xlMissingItemsNone
```

```
For Each pCache In KS.PivotCaches
```

```
    On Error Resume Next
```

```
        pCache.Refresh
```

```
    On Error GoTo 0
```

```
Next
```

```
Set rVikaSolu = shData.Cells(2, 2).End(xlDown)
```

```
sSTaulu = "Data!R2C" & iDIDSar & ":R" & rVikaSolu.Row & "C" & iDPivotDataSar & "
```

```
sDTaulu = "Data!R2C" & iDTotalSar & ":R" & rVikaSolu.Row & "C" & iDIDSar & "
```

```
SPTable.PivotTableWizard SourceType:=xlDatabase, SourceData:=sSTaulu
```

```
DPTable.PivotTableWizard SourceType:=xlDatabase, SourceData:=sDTaulu
```

```
Dim pField As PivotField
```

```
For Each pField In SPTable.PivotFields
```

```
    On Error Resume Next
```

```
        pField.PivotItems("(blank)").Visible = False
```

```
        pField.PivotItems("0").Visible = False
```

```
    On Error GoTo 0
```

```
Next pField
```

```
SPTable.PivotCache.MissingItemsLimit = xlMissingItemsNone
```

```
For Each pCache In KS.PivotCaches
```

```
    On Error Resume Next
```

```
        pCache.Refresh
```

```
    On Error GoTo 0
```

```
Next
```

```
KS.ShowPivotTableFieldList = False
```

```
Application.Calculate
```

```
SPTable.RefreshTable
```

```
DPTable.RefreshTable
```

```
End Sub
```

```
Sub Tarkista_Jarjestys()
```

```
Dim i As Integer
```

```
Dim bPoikkeamia As Boolean
```

```
Dim rSolu As Range
```

```
Dim pItem As PivotItem
```

```
bPoikkeamia = False
```

```
For Each pItem In SPFieldID.PivotItems
```

```
    On Error Resume Next
```

```
        If pItem.Position <> shSeuranta.Cells(pItem.LabelRange.Row,  
pItem.LabelRange.Column - 1).Value Then
```

```
            On Error GoTo 0
```

```
                bPoikkeamia = True
```

```
                Exit For
```

```

        End If
    Next

    If SPTable.RowRange.End(xlDown).Offset(1, -2).Value <> 0 Then
        bPoikkeamia = True
    End If

    If bPoikkeamia = True Then
        Call Data_Alustus
        Call Data_Pivottiin
        Call Korjaa_Jarjestys
    End If

    On Error Resume Next
    ' Piilottaa turhan datan
    With SPFieldNames
        .PivotItems("0").Visible = False
        .PivotItems("(blank)").Visible = False
        .PivotItems("").Visible = False
    End With
    SPFieldID.PivotItems("(blank)").Visible = False
    On Error GoTo 0
    '
    SPTable.DataBodyRange.Interior.ColorIndex = 0
    '
    SPTable.DataBodyRange.Borders(xlInsideHorizontal).LineStyle = xlNone
    '
    SPTable.DataBodyRange.Borders(xlInsideVertical).LineStyle = xlNone
    Exit Sub
End Sub

Sub Korjaa_Jarjestys()

    Dim rSolu As Range
    Dim pItem As PivotItem
    Dim pCache As PivotCache
    Dim i As Integer
    Dim j As Integer
    Dim jarjestys(20) As String

    SPTable.PivotCache.MissingItemsLimit = xlMissingItemsNone
    For Each pCache In KS.PivotCaches
        On Error Resume Next
        pCache.Refresh
        On Error GoTo 0
    Next
    For Each pItem In SPFieldID.PivotItems
        On Error Resume Next
        If pItem.Position <> Cells(pItem.DataRange.Row, pItem.LabelRange.Column -
1).Value Then
            pItem.Position = Cells(pItem.DataRange.Row, pItem.LabelRange.Column -
1).Value
        End If
        On Error GoTo 0
    Next

    ' Kerää osastot yhteen matriisiin Osastot-välilehdellä olevaan järjestykseen
    shOsastot.Unprotect
    For Each rSolu In shOsastot.Columns(6).SpecialCells(xlFormulas)
        For Each pItem In SpFieldOsasto.PivotItems
            If rSolu.Text = pItem.Name Then
                i = i + 1
                jarjestys(i) = rSolu.Text
            Exit For
        End If
    Next
Next
shOsastot.Protect

```

```

For Each pItem In SpFieldOsasto.PivotItems
    For j = 1 To i
        If pItem.Name = jarjestys(j) Then
            pItem.Position = j
        End If
    Next
Next
End Sub

Sub DataAlue_Formaus()

    Dim sOsoite As String
    Dim rTempSolu As Range

    Set rPDataAlue = SPTable.DataBodyRange

    ' Data-alueen Sisäset viivotukset
    With rPDataAlue
        On Error Resume Next
        If .Rows.Count > 1 Then
            With .Borders(xlInsideHorizontal)
                .LineStyle = xlContinuous
                .Weight = xlThin
                .ColorIndex = 15
            End With
        End If
        If .Columns.Count > 1 Then
            With .Borders(xlInsideVertical)
                .LineStyle = xlContinuous
                .Weight = xlThin
                .ColorIndex = 15
            End With
        End If
        On Error GoTo 0
    End With

    On Error GoTo Virhe
    SPTable.DataBodyRange.Select
    On Error GoTo 0
    With Selection
        .HorizontalAlignment = xlCenter
        ' Eka konditio
        .FormatConditions.Add Type:=xlCellValue, Operator:=xlLess, Formula:="0"
        .FormatConditions(1).Interior.ColorIndex = 22
        ' Toka konditio
        .FormatConditions.Add Type:=xlCellValue, Operator:=xlGreater, Formula:="0"
        .FormatConditions(2).Interior.ColorIndex = 23
        ' Kolmas konditio
        sOsoite = .PivotCell.Range.Address(False, False)

        Set rTempSolu = shSeuranta.Cells(1, iStuplatSar + 1)
        rTempSolu.Formula = "=ISNUMBER(" & sOsoite & ")"
        ' Formulalocal-säätö värjää laittaa oikean kaavan riippumatta kieliasetuksista
        .FormatConditions.Add Type:=xlExpression, Formula:=rTempSolu.FormulaLocal
        .FormatConditions(3).Interior.ColorIndex = 24
        ' Borderit joka solulle
    End With
    On Error Resume Next
    With .SpecialCells(xlCellTypeConstants).Borders
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    On Error GoTo 0
End With

```

```
Virhe:
End Sub
```

```
Sub OsastoRivi_formaus()

    Dim pItem As PivotItem

    With SpFieldOsasto
        On Error Resume Next
        .DataRange.HorizontalAlignment = xlGeneral
        .DataRange.Borders(xlInsideVertical).Weight = xlMedium
    On Error GoTo 0
    For Each pItem In .PivotItems
        If pItem.Visible = True Then
            On Error Resume Next
            pItem.DataRange.Borders(xlEdgeRight).Weight = xlMedium
            pItem.DataRange.Borders(xlEdgeRight).ColorIndex = 1
            pItem.DataRange.Borders(xlEdgeLeft).Weight = xlMedium
            pItem.DataRange.Borders(xlEdgeLeft).ColorIndex = 1
            On Error GoTo 0
        End If
    Next
    End With
End Sub
```

```
Sub NamesRivi_Formaus()

    Dim pItem As PivotItem

    With SPFieldNames.DataRange
        With .Font
            .Bold = True
            .ColorIndex = 2
        End With
        With .Borders(xlInsideVertical)
            .LineStyle = xlContinuous
            .ColorIndex = 2
        End With

        For Each pItem In SPFieldNames.PivotItems
            If pItem.Visible = True Then
                On Error Resume Next
                If pItem.LabelRange.Offset(1, 0).Borders(xlEdgeRight).Weight = xlMedium Then
                    pItem.LabelRange.Borders(xlEdgeRight).Weight = xlMedium
                Else
                    pItem.LabelRange.Borders(xlEdgeRight).Weight = xlThin
                End If
                On Error GoTo 0
            End If
        Next
        With .Borders(xlTop)
            .LineStyle = xlContinuous
            .ColorIndex = 0
        End With
        With .Borders(xlBottom)
            .LineStyle = xlContinuous
            .ColorIndex = 0
        End With
        .Interior.ColorIndex = 56
    End With
End Sub
```

```
Sub HarmaaSolut()  
  
    Dim iSarVasen As Integer  
    Dim iSarOikea As Integer  
    Dim rSolu As Range  
  
    With shSeuranta  
        ' Rajoita alue  
        With SPTable.DataBodyRange  
            iSarVasen = .Columns(1).Column  
            iSarOikea = .Columns(.Columns.Count).Column  
        End With  
        ' poistaa harmaan värin  
        .Range(.Cells(iSOTSikkoRivi + 1, iSarVasen), Range("IV65536")).Interior.ColorIndex  
= 0  
  
        For Each rSolu In .Columns(iSSumSar).SpecialCells(xlFormulas)  
            If rSolu.Value = "No Link" Then  
                .Range(shSeuranta.Cells(rSolu.Row, iSarVasen), shSeuranta.Cells(rSolu.Row,  
iSarOikea)).Interior.ColorIndex = 20  
            End If  
        Next rSolu  
    End With  
End Sub
```