

Tampereen ammattikorkeakoulu
Kone- ja tuotantotekniikan koulutusohjelma
Modernit tuotantojärjestelmät ja tuotantotalous
Seppo Maljanen

Opinnäytetyö

Yrityksen toiminnanohjausjärjestelmä
Case:Lean System/Locos

Työn ohjaaja lehtori, Arto Jokihara
Työn teettäjä TAMK, Koulutuspäällikkö, yliopettaja Olavi Kopponen

Tampere 11/2010

Tampereen ammattikorkeakoulu
Kone- ja tuotantotekniikan koulutusohjelma
Modernit tuotantojärjestelmät ja tuotantotalous

Tekijä	Maljanen, Seppo Juhani
Työn nimi	Yrityksen toiminnanohjausjärjestelmä Esimerkkinä: Lean System
Sivumäärä	73 + 23 liitesivua
Valmistumisaika	Marraskuu 2010
Työn ohjaaja	lehtori, Arto Jokihaara
Työn teettäjä	TAMK, Koulutuspäällikkö, yliopettaja Olavi Kopponen

TIIVISTELMÄ

Tämän opinnäytetyön tarkoituksena on antaa tietämystä toiminnanohjausjärjestelmän yleisistä käytännöistä ja luoda yleiskuva Lean System ohjelmistojärjestelmästä. Lean System järjestelmä on joukko erillisiä sovelluksia, jotka sisältävät ja toteuttavat teknisesti joukon tiedonhallinnollisia toimintoja. Järjestelmä tarjoaa selkeän rajapinnan näiden toimintojen käyttämiseen ja hallitun tiedon selailuun.

Tarkoituksena on antaa tietämystä yritykseen hankittavaan toiminnanohjausjärjestelmään ja huomioida vaatimusmäärittelyn mukaiset muutoksien tuomat etuudet ja haitat valmiin tuotteen ja räätälöidyn ohjelmistojärjestelmän välillä.

Työssä käytettiin myös V-mallia (V-Modell), jonka avulla määritellään ohjelmistoprojektin päämäärät ja tavat, joilla näihin tavoitteisiin päästään. Tavoitteena ja tarkoituksena oli myös muutospyynnön Locos/CR1346 V- mallin mukainen käsittely.

Tutkittiin yritykseen hankittavaa toiminnanohjausjärjestelmää C-CEI –menetelmällä järjestelmähankkeen tueksi.

Luottamuksellista syistä Lean System ja Locos liitetiedostot, jotka viittaavat opinnäytetyöhön on jätetty pois.

Avainsanat	opinnäytetyö, toiminnanohjausjärjestelmä, ERP, Lean System, ohjelmistojärjestelmä, räätälöity sovellus
------------	---

TAMK University of Applied Sciences
Mechanical and Production Engineering
Modern Production Systems & Industrial Engineering

Author	Maljanen, Seppo Juhani
Thesis	ERP Enterprise Resource Planning Case: Lean System
Pages	73 pages + 23 appendices
Graduation time	November 2010
Thesis supervisor	lecturer, Arto Jokihäärä
Commissioned by	principal lecturer, Olavi Kopponen, TAMK

ABSTRACT

The purpose of this thesis is to provide knowledge of the ERP (Enterprise Resource Planning) system general practices and create a picture from Lean System software, which is a series of separate applications that include and implement technically a number of information functions and provide a clear interface of these functions and controlled information browsing.

The purpose is to provide helpful knowledge for acquiring an ERP system for a company and take into account the requirement in accordance with the definition of the changes brought about by the benefits and drawbacks of the product and custom-made software in the range.

The V-Modell was used in this work. It is used to define a software project goals and methods, which these goals are reached. The goal and purpose was also in the treatment of the change request Locos/CR1346 according the V-Modell.

The purchased ERP system of the company was studied by the C-CEI method, which supports the system project.

Lean System and Locos's appendices in this work are confidential.

Keywords	thesis, enterprise resource planning, ERP, Lean System, programming system software, custom-made software
----------	---

Esipuhe

Tämä opinnäytetyö on tehty Tampereen ammattikorkeakoulun toimeksiannosta ja opinnäytetyöksi Tampereen ammattikorkeakoulun Kone- ja tuotantotekniikan koulutusohjelman aikuisopiskelijan opintolinjalle.

Kiitän perhettäni, vaimoani Maijaa, lapsiani Mariaa ja Tuomasta isälle antamasta tuesta ja kannustuksesta ei vain tämän opinnäytetyön tekemisessä vaan koko kolme ja puoli-vuotisen insinööri opiskelun ajalta.

Kiitos myös tutkintotyönvalvojalle Arto Jokihaaralle ohjauksesta, neuvoista ja tuesta.

Ylöjärvellä marraskuussa 2010

Seppo Maljanen

Sisällysluettelo

1	Johdanto.....	11
2	Tavoitteet ja rajaukset.....	12
3	Yrityksen toiminnanohjausjärjestelmän ongelmat.....	15
4	Ohjelmisto – Case: Lean System	17
4.1	<i>Lean System -järjestelmä.....</i>	17
4.2	<i>Järjestelmän moduuli rakenne</i>	18
4.3	<i>Pääohjelma.....</i>	19
4.4	<i>Ylläpito.....</i>	19
4.5	<i>Perustiedot.....</i>	20
4.6	<i>Nimiketiedot.....</i>	20
4.7	<i>Myynti.....</i>	21
4.8	<i>Toimitus</i>	22
4.9	<i>Laskutus.....</i>	23
4.10	<i>Valmistusrakenne</i>	23
4.11	<i>Tuotannonsuunnittelu.....</i>	24
4.12	<i>Valmistus</i>	25
4.13	<i>Osto.....</i>	25
4.14	<i>Vastaanotto.....</i>	26
4.15	<i>Ostolaskuntarkastus</i>	26
4.16	<i>Varasto.....</i>	27
4.17	<i>Raportointi.....</i>	28
4.18	<i>Tiedonsiirto.....</i>	28
4.19	<i>Taustaprosessi</i>	28
4.20	<i>Projektinohjaus.....</i>	28
5	Lean System – Sovellusten peruskäsitteet	29
5.1	<i>Lean System työpöytä</i>	29
5.2	<i>Lean System sovellukset</i>	30
6	Lean System - Lomakkeet ja tulostus.....	31
7	Tilaus- ja toimitusketjun hallinta.....	33

8	Ohjelmoinnin työvälineet	37
8.1	<i>SQL Windows</i>	37
8.2	<i>PL/SQL</i>	38
8.3	<i>Tietokantaskriptit</i>	40
8.4	<i>Perl</i>	41
9	Valmisohjelma vs. räätälöityohjelma ratkaisu	42
9.1	<i>Toimitus hierarkkia</i>	42
9.2	<i>Asiakkaan vastuut ohjelmistoprojektissa</i>	44
9.3	<i>Toimittajan vastuut ohjelmistoprojektissa</i>	44
10	V Malli	45
10.1	<i>V-malli tavoitteet</i>	45
10.2	<i>V-malli rakenne</i>	46
10.3	<i>Hyödyt ja rajoitukset</i>	46
11	Case: Lean System / Locos CR1346	47
11.1	<i>Liiketoiminnan vaatimukset</i>	47
11.2	<i>Järjestelmän vaatimukset</i>	48
11.3	<i>Locos järjestelmän toteutus</i>	49
11.4	<i>Locos järjestelmän dokumentointi</i>	53
11.5	<i>Locos järjestelmän komponenttitestaus</i>	53
12	Räätälöidyn ERP-järjestelmän kustannukset	55
12.1	<i>Operatiiviset tietotekniikkainvestoinnit</i>	55
12.2	<i>Strategiset tietotekniikkainvestoinnit</i>	55
12.3	<i>Tietotekniikkainvestoinnin hyödyt</i>	55
12.4	<i>Tietojärjestelmän kustannukset – Nykyarvomenetelmä</i>	56
12.5	<i>C-CEI menetelmä</i>	59
13	Tuotantotesti	60
13.1	<i>Testaukseen valmistautuminen</i>	61
13.2	<i>Komponentti- ja moduulitesti</i>	62
13.3	<i>Virheiden luokittelu</i>	62
13.4	<i>Testauksen ympäristö</i>	63

13.5 Ohjelmiston toimitus ja asennus.....	64
14 Tietokanta.....	65
14.1 Fyysinen taso.....	65
14.1.1 Ohjaustiedosto.....	65
14.1.2 Online redo –lokit	66
14.1.3 Archive log –tiedostot	66
14.1.4 Tietokantatiedostot.....	66
14.2 Looginen taso	67
14.3 Instanssi.....	67
14.4 Muistirakenne	68
14.5 Tietokantataulut.....	68
14.6 Palautus.....	68
14.7 Ohjaustiedot	68
15 ERP – toimittajat Suomessa	69
16 Yhteenveto	70
Lähdeluettelo	72
Liitteet	

Käytetyt lyhenteet

Accounting	Kirjanpito (Taloushallintotoiminnot).
Batch Runs	Tausta-ajot (Järjestelmätoiminnot).
BI	<i>Business Intelligence</i> . Johdonliiketoiminnot.
Cost Accounting	Kustannuslaskenta (Taloushallintotoiminnot).
COTS	<i>Commercial-Off-The-Shelf –type ERP system</i> . “Suoraan hyllyltä” ostettava valmis ERP järjestelmä.
C-CEI	<i>Customer-Centered ERP Implementation</i> . Systemaattinen menetelmätutkimus.
CR	<i>Change Request</i> . Muutospyyntö.
CT	<i>Component test</i> . Komponenttitesti.
CRM	<i>Customer Relationship Management</i> . Asiakassuhteidenhallinta.
Customer	Asiakas.
DD	<i>Detail Design</i> . Yksityskohtainen suunnittelu.
Delivery	Toimitus.
ERP	<i>Enterprise Resource Planning</i> . Toiminnanohjausjärjestelmä.
FD	<i>Functional Design</i> . Toiminnallinen suunnittelu.
Finance System	Taloushallintojärjestelmä.
HR	<i>Human Recource</i> . Henkilöstöhallinto.
Inventory	Varastointi.
Invoicing	Laskutus.
LEAN System[®]	<i>Product name for the operations management system</i> . Ohjelmatuotteen nimi.

LOCOS	<i>Application name for the NET BMS.</i> Räätälöydyn ohjelmatuotteen nimi.
Logical model	Looginen malli.
MLS	<i>Modular Logistics System.</i> Modulaarinen logistinen järjestelmä.
MSS	<i>Message Switching System.</i> Viestinvälitysjärjestelmä.
MT	<i>Module test.</i> Moduulitesti.
Pk-yritys	Pieni - keskisuuri yritys.
PP	<i>Project Plan.</i> Projektisuunnitelma.
Production	Tuotanto.
Production Planning	Tuotannosuunnittelu.
Production System	Tuotantojärjestelmä.
PT	<i>Product test.</i> Tuotantotesti.
Purchase	Ostotoiminto.
Purchase Invoicing	Ostolaskutus.
Receive	Vastaanotto.
Sales	Myyntitoiminto.
SAP	<i>Systems Analysis And Program Development.</i> Ohjelmatuotteen nimi.
SCM	<i>Supply Chain Management.</i> Toimitusketjun hallinta.
Stock	Varasto.
Structure	(Tuote)rakenne.
Supplier	Toimittaja.
TOM	<i>Team Object Manager.</i> Työkaluohjelmisto.
UAT	<i>User Acceptance Test.</i> Tilaaajan hyväksymä testi.
V-Modell	<i>The V-model of the Systems Engineering Process.</i> V-Malli.
WP	<i>Maintenance Work Package.</i> Toimituksen sisältö.

Käytetyt käsitteet ja termit

Kustomointi	Eriytetään ohjelmistojärjestelmä tuotteesta asiakkaan toiveiden mukaiseksi.
Release	Tiettyyn ohjelmatoimitukseen menevät revisiot.
Revisio	Yhden ohjelmistojärjestelmään kuuluvan komponentin tietynä ajanhetkenä jäädytetty tila.
Räätälöity sovellus	Sovellus, joka on ohjelmistoprojektissa räätälöity asiakasta varten.
Sovellus	Yksi ajettava ohjelmakokonaisuus.
Tietokantaskripti	Tietokannan muuttamiseksi tarvittava komentotiedosto.
Versio	Ohjelmistojärjestelmän yhtenä ajanhetkenä jäädytetty kokonaisuus, jota voidaan erikseen ylläpitää, ja joka siis elää version perustamisen jälkeen omaa elämäänsä. Versio on kokonaisuus kaikista ohjelmistojärjestelmän sovelluksista ja osista, joista kustakin on otettu mukaan tietty revisio.

1 Johdanto

Nykyisin markkinoilla on useita saatavilla olevia toiminnanohjausjärjestelmiä, joita ei ole erikseen räätälöity yhden yrityksen tarpeita varten. Tämä asettaa yritykselle omat haasteet ja suuntauksset toiminnanohjausjärjestelmää hankittaessa. Tässä opinnäytetyössäni käsittelen Lean System/Locos järjestelmän käyttöönottoa ja toteutusta ohjelman toimittajan ja asiakasyrityksen näkökulmasta ollessani tekemisissä näiden asioiden kanssa.

Lean System on Tieto Oyj (ent. TietoEnator Oyj) kehittämä toiminnanohjausjärjestelmätuote, joka on suunniteltu joustavuutta ja nopeutta vaativiin ympäristöihin. Lean Systemiä käyttää päivittäin yli 100 pääasiallisesti verkottuneen teollisuuden, palveluliiketoiminnan ja projektiohjauksen yritystä eri puolilla maailmaa. /11/

Monissa ammattikorkeakouluissa ja yliopistoissa käytetty Lean System on saavuttanut myönteisen vastaanoton opettajien ja oppilaiden keskuudessa. Opinnäytetyön perusidea pohjautuu juuri opitun ja käytännössä tehtävien asioiden ymmärtämiseen.

Lean System on käytössä Tampereen ammattikorkeakoulun Kone- ja tuotantotekniikan koulutusohjelmassa.

2 Tavoitteet ja rajaukset

Tämän opinnäytetyön tavoite antaa tietämystä toiminnanohjausjärjestelmän yleisistä käytännöistä ja luoda yleiskuva Lean System/Locos järjestelmästä. Esimerkiksi tilaus ja toimitusketjun ohjauksesta. Tarkempaan ja syvällisempään tarkasteluun Lean System/Locos ohjelmistoon ei tämän opinnäytetyön rajallisuuden puitteissa ole mahdollista syventyä.

Tavoite on antaa tietämystä yritykseen hankittavaan ERP - toiminnanohjausjärjestelmään ja huomioida vaatimusmäärittelyn mukaiset muutoksien tuomat etuudet ja haitat tuotteen ja räätälöidyn ohjelmiston välillä.

Tässä opinnäytetyön tarkastelussa ovat valmisohjelma tuote Lean System versio 5.3 ja räätälöity ohjelmisto Locos versio 5.3. Lean System / Locos - toiminnanohjausjärjestelmät on toteutettu asiakas - palvelin (client – server technology) teknologialla Windows NT käyttöjärjestelmä ympäristössä. Palvelin tietokanta ohjelmistona on käytetty Oracle 7 ja siihen tarvittavat Oraclen komponentit esimerkiksi Oracle SQL*Plus ja Oracle7 Server.

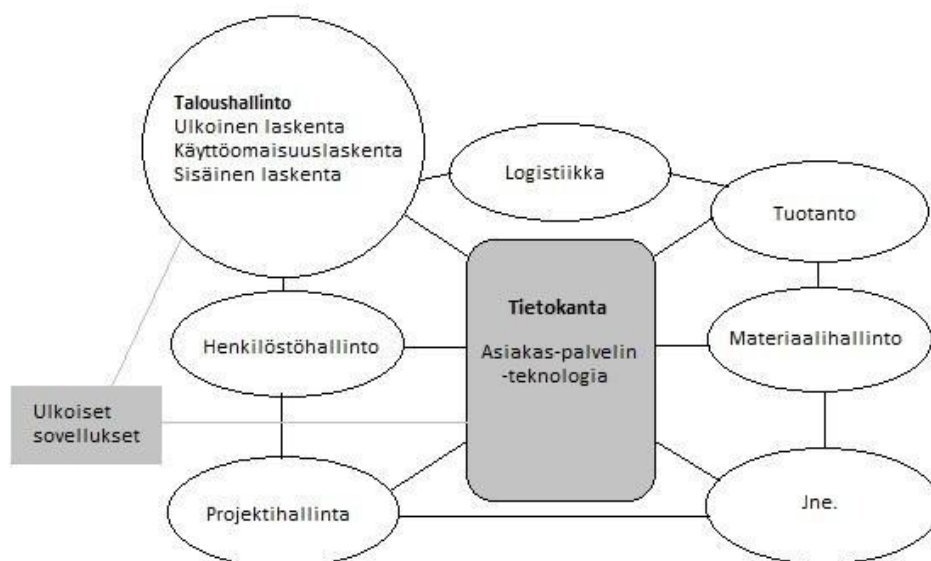
Opinnäytetyössä on rajattu pois Lean System / Locos ohjelmistojen käyttöliittymän käsittely, Lean Systemin konfiguraattori, ohjelmiston asennus- ja ylläpito sekä sovellusalueet, kuten valuutta-, hinta-, tuote-, asiakasrekisterisovellukset. Sähköinen tiedon siirto-ohjelmisto ja laitteistot eivät kuulu tämän opinnäytetyön alueeseen. Ohjelmiston testausta käsitellään tuotantotestin osalta. Yleistä ERP-järjestelmästä

ERP-järjestelmistä (Enterprise Resource Planning) käytetään suomenkielisessä kirjallisuudessa yleisesti nimitystä toiminnanohjausjärjestelmä, jonka tarkoitus yleisesti on yhdistää kaikki yrityksen toiminnot. Sen avulla pystytään ohjaamaan ja/tai hallinnoimaan tehokkaasti ja reaaliaikaisesti tietojenkäsittelyä ja tiedonvälitystä. Toiminnanohjausjärjestelmällä ei tarkoiteta pelkästään ERP-ohjelmistoa. Sillä tarkoitetaan myös periaatteita, menetelmiä ja niitä tukevia tietotekniikoita sekä yrityksen prosesseja. ERP-järjestelmä koostuu toisiaan täydentävistä moduuleista, joilla voidaan korvata toimintokohtaisia erillisiä tietojärjestelmiä.

Toiminnanohjausjärjestelmiä kehitetään yhä enenevässä määrin pienten ja keskisuurten yritysten (pk-yritys) tarpeisiin. Organisaatioiden lisääntyvä hajautettutoiminta ja alueellinen sekä globaali sijoittuminen vaatii yrityksiltä liiketoimintojen yhteensovittamista ja pieni- keskisuuri yritystenkin täytyy kytkeytyä isompiensa järjestelmiin. Ohjelmistokehityksessä ja työnteossa tapahtuneet merkittävät muutokset viimeisien vuosikymmenien aikana sekä teknologioiden kehittyminen on automatisoinut yritysten toimintoja, ja tietotyöläisten päivittäiseen työhön liittyvät toiminnalliset ulottuvuudet ovat kasvaneet huomattavasti. /1, s.6/

Yleinen määritelmä ERP-järjestelmälle on integroitu ja kaikki yrityksen tietovirrat kattava ohjelmisto. Kuvioista 1 nähdään, kuinka sen ytimen muodostaa yksi kokonaisvaltainen tietokanta, johon tieto syötetään vain kerran, ja tämä tieto on heti kaikkien toiminnanohjausjärjestelmän sisältämien moduulien käytettävissä. Tiedonsyöttäminen vain kerran lisää sen luotettavuutta sekä vähentää viivästyksiä, mutta korostaa myös samalla vastuuta syötettävän tiedon oikeellisuudesta. Järjestelmä rakentuu tietokannan ympärille erillisistä, toisiinsa nivoutuvista ohjelmistomoduuleista, joista voidaan käyttää tilanteen mukaan vain tarvittavia toimintoja. Tarvittaessa nämä itsenäiset ohjelmisto-osat tukevat yrityksen toimintoja globaalisti yli funktio- ja yksikkörajojen.

/2, s.32/



Kuvio 1: ERP-järjestelmän perusrakenne /2, s.33/

Yleensä ERP:llä hallitaan yrityksen sisäisiä perustoimintoja kuten logistiikkaa, tuotantoa ja taloushallintoa. ERP-järjestelmien modulaarisuus mahdollistaa eri ominaisuuksien lisäämisen järjestelmään jälkikäteen. Perustoimintojen lisäksi nykyiset ERP-toimittajat ovat laajentaneet järjestelmiensä moduuleja jo niin paljon, että joillain ERP-järjestelmillä voidaan hallita jo lähestulkoon kaikki toiminnot yrityksessä. ERP:n ja muiden järjestelmien (kuvio 2), kuten CRM:n, (Customer Relationship Management), ja SCM:n, (Supply Chain Management), keskinäiset roolit ovat muuttumassa ERP-järjestelmien pystyessä hallitsemaan yhä useampia toimintoja. /1, s.16/



Kuvio 2: ERP:n osa-alueet /1, s.16/

3 Yrityksen toiminnanohjausjärjestelmän ongelmat

Toiminnanohjausjärjestelmien toteutuksien ongelmana on, ettei järjestelmä vastaa asiakkaan liiketoimintaprosesseja tai siihen, ettei järjestelmä tue liiketoiminnassa syntyvää tietoa dataa. Perimmäinen syy ongelmiin on se, ettei kukaan osapuolista, ei liiketoiminta- eikä toimittajapuoli, ota kokonaisvastuuta järjestelmien laadusta ja sen määrittelystä./8/

Alan järjestelmätoimittaja Lawson Softwaren Suomessa toteuttaman tutkimuksen mukaan toiminnanohjausjärjestelmän tuomien hyötyjen mittaaminen on yrityksillä varsin hataraa. Lawsonin tutkimuksen mukaan peräti 76 prosenttia yrityksistä joko ei mittaa toiminnanohjausjärjestelmän tuomia hyötyjä mitenkään (29 % vastaajista) tai käyttää niin sanottua mutu-tuntumaa (47 % vastaajista). Tutkimuksessa kysyttiin lisäksi toiminnanohjausjärjestelmän valintakriteerejä. Kolme kriteeriä nousi ylitse muiden toiminnanohjausjärjestelmää valittaessa tai uusittaessa: sen on oltava helppokäyttöinen (95 % vastaajista koki tärkeäksi tai erittäin tärkeäksi), säästettävä työaika (94 %), ja sen on tuettava liiketoiminnan kehittämistä (94 %). Vastaajat toivoivat lisäksi, että järjestelmä tukisi mahdollisimman hyvin vaativiakin liiketoimintaprosesseja (85 %). Järjestelmän tekniset ominaisuudet jäivät näistä kriteereistä selvästi vähemmälle painoarvolle. Tutkimus toteutettiin sähköisenä suurten ja keskisuurten yritysten tietohallinto- ja liiketoimintajohdolle syksyllä 2008. Tutkimukseen vastasi 100 henkilöä. /7/

Pk-yritysten toiminnan kannalta kriittiset kysymykset, jotka koskevat nykyisten ERP -järjestelmien soveltuvuutta pk-yrityksen toiminnanohjaukseen, voidaan jakaa neljään pääryhmään: /6, s.49-52/

1. Toiminnanohjausjärjestelmien joustamattomuus.
2. Toiminnanohjausjärjestelmän pitkä käyttöönottoprosessi.
3. Toiminnanohjausjärjestelmän hierarkkisuus.
4. Organisaation osaaminen ja suhtautuminen tietojärjestelmähankeisiin.

ERP –järjestelmän joustamattomuus on keskeinen ongelma pk-yrityksen näkökulmasta. Toiminnanohjausjärjestelmät on tavallisesti rakennettu yritysten tyypillisiä toimintaprosesseja kuvaaviin prosessimalleihin perustuen. Joustamattomuuden seurauksena yritys

joutuu usein mukautumaan tietojärjestelmän logiikkaan, eikä moin ollen voi toimia optimaalisella, oman ansaintalogiikkansa vaatimalla tavalla.

Pitkä käyttöönottoprosessi voidaan nähdä toiminnanohjausta ja uuteen järjestelmään mukautumista vaikeuttavana tekijänä. Tyypillisesti pk-yritysympäristössä järjestelmän ja suunnittelu- ja käyttöönottovaihe kestää noin vuoden (vaihteluväli on kuitenkin suuri aina muutamasta kuukaudesta useampaan vuoteen). Tämän seurauksen uusi järjestelmä voi olla ”vanha” jo tuotantokäytön alkaessa.

Toiminnanohjausjärjestelmien mukanaan tuoma tiedon keskitetty hallinta ja tämän mahdollistama ”kontrollointi” voidaan nähdä yritystoimintaa rasittavina tekijöinä. Tarkka kontrollointi ja raportointi voivat tuntua esimerkiksi työntekijöistä liialliselta valvonnalta, joka herkästi tulkitaan epäluottamuksen osoitukseksi. Toiminnanohjausjärjestelmän hierarkkisuus taas ”sotii” nykyaikana korostettuja organisaation avoimuuden ja vapauden periaatteita vastaan.

Organisaation osaaminen ja suhtautuminen tietojärjestelmähankkeisiin on kriittinen tekijä erityisesti järjestelmän hankinta- ja käyttöönottovaiheessa. Tämä vaikeuttaa järjestelmien määrittelyä ja lisää epäonnistuneiden valintojen riskiä. Toisaalta järjestelmähankkeet mielletään usein korostetun tietoteknisiksi hankkeiksi, jolloin liiketoiminnan kehittäminen jää helposti vähemmälle. Lisäksi monissa yrityksissä järjestelmien mukanaan tuomiin mahdollisuuksiin suhtaudutaan liiankin optimisesti.

Toiminnanohjaus- ja tietojärjestelmien soveltuvuutta voidaan lähestyä toiminnallisuuden ja organisatorisen näkökulman lisäksi myös järjestelmä- ja teknologialähtöisesti. Teknologinen kehitys voidaan nähdä sekä heikkoutena että mahdollisuutena muuttuvassa toimintaympäristössä. Uusien teknologioiden soveltaminen ja hyötykäyttö vaativat sekä tietämystä että resursseja (aikaa ja rahaa), jotka usein ovat kummatkin rajallisia pk-yrityksessä.

Järjestelmien integrointivaikeudet ja tietojärjestelmäarkkitehtuuri on yksi merkittävä tekijä yritystoiminnan muuttuvassa kentässä. Näiden merkitys korostuu erityisesti tilanteissa, jossa yritysten verkostoitumista halutaan tukea tietoteknisin välinein. Nykyisten ERP-järjestelmien ongelmana onkin toiminnallisen joustamattomuuden lisäksi tekninen joustamattomuus ja järjestelmien integrointihaasteet.

4 Ohjelmisto – Case: Lean System

4.1 Lean System -järjestelmä

Lean System on tarkoitettu nykyaikaisia, joustavia ohjausperiaatteita toteuttaville yrityksille, joille Lean Production -periaatteet ovat tärkeitä /21, s.3/:

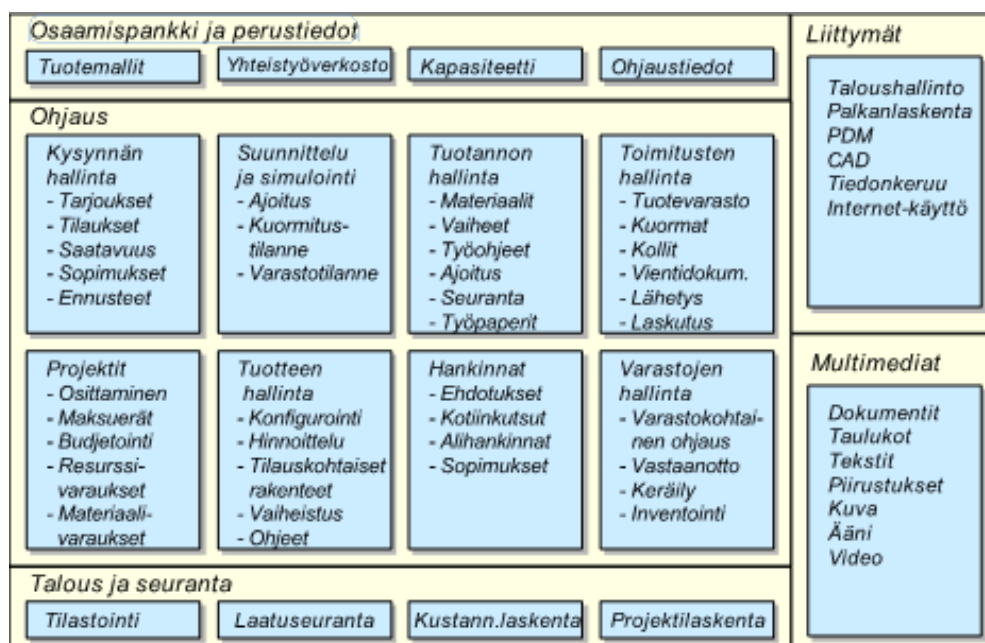
- asiakasohjautuvuus, nopea reagointi muutoksiin
- lyhyet ja hallitut läpäisyajat, ehdoton toimitusvarmuus
- itseohjautuva toimintatapa, monitaitoiset ja vastuulliset työryhmät
- jatkuva kehittäminen ja kokonaisvaltainen laatu
- avoimuuden ja informaation korostaminen
- kustannustehokkuus
- toiminnan tason jatkuva mittaaminen ja oppiminen
- terveet ja joukkuepeliä korostavat johtamisperiaatteet

Lean Systemissä käyttäjän päätöksenteko ja tilannearviot on nostettu ratkaiseviksi ohjaustekijöiksi tehokkaiden ja havainnollisten työasemaohjelmistojen avulla. Saman ongelmatilanteen ratkaisemiseksi voi lisäksi olla useita erilaisia vaihtoehtoja perustietoratkaisun pysyessä samana. Merkittävä osa tietotekniikan kannalta hankalista "sumeista" ongelmista ja ennakoimattomista poikkeustilanteista annetaan käyttäjien ratkaistavaksi -ikkunoivien, graafisten työasemaohjelmistojen avulla. Näin varmistetaan toiminnan sujuminen, johdonmukainen päätöksenteko ja huippuluokan asiakaspalvelu nopeasti muuttuvissa ja ennalta arvaamattomissakin tilanteissa.

Lean System kattaa teollisen toiminnanohjauksen kaikki alueet tarjouslaskennasta tuotteiden toimittamiseen ja laskuttamiseen asti (kuvio 3): /21,s.1-2/

- ennusteiden, tarjousten ja tilausten hallinta
- vakiorakenteiden ja tilauskohtaisten tuoterakenteiden hallinta
- vaiheistusten ja työohjeiden ylläpito vakiotuotteille ja tilauskohtaisille tuotteille
- projektien ja töiden suunnittelu ja seuranta
- tuotannon aikataulut ja kuormitus
- materiaalitarpeiden laskenta ja jatkuva päivitys
- materiaalien ja komponenttien hankintatilanteen hallinta ja ostorutiinien hoito
- alihankintojen suunnittelu ja seuranta

- eräpohjainen, jäljittävä varastointijärjestelmä materiaaleille, puolivalmisteille ja tuotteille
- projekti-, tilaus-, työ-, tuote- ja organisaatioyksikkökohtaiset kustannuslaskennat (ennakkolaskenta ja vertailu toteutuneeseen)
- materiaali- ja palikatapahtumien keruu viivakoodien avulla
- lähetysten ja kuormien hallinta
- laskutus
- laatusuranta ja tehokkuusseuranta
- yhteydet OVT-siirtoihin, ostoreskontraan, myyntireskontraan, kirjanpitoon ja palkanlaskentaan
- tehokkaat mekanismit epäformaalin tiedon hallintaan ja välitykseen

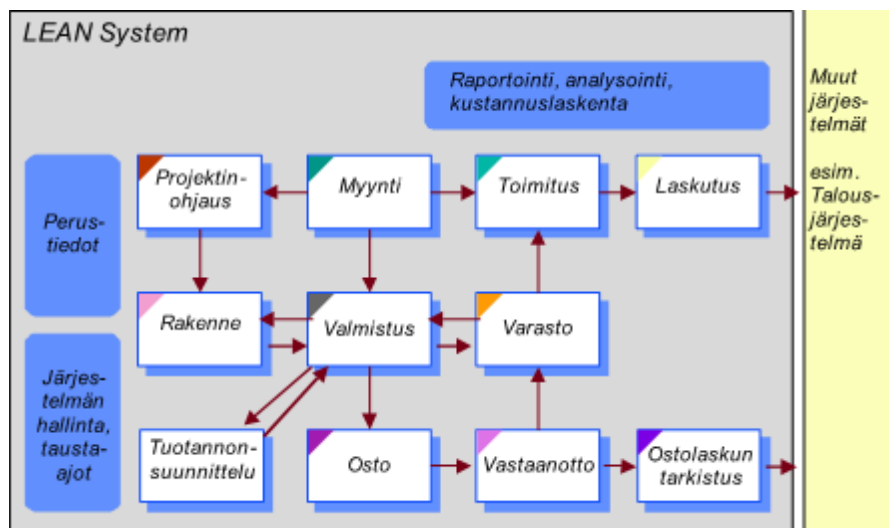


Kuvio 3: LEAN System osat /21, s.2/

Lean System sisältää valmiit liittämät myös multimedia-tyyppiseen työskentely-ympäristöön. Tietokannassa valmiina oleviin tietoihin voidaan liittää taulukoita, tekstidokumentteja, kuvia, työohjeita, ääniviestejä tai vaikka videokuvaa.

4.2 Järjestelmän moduulirakenne

Lean System - järjestelmä jakautuu useaan erilliseen sovellukseen (kuvio 4), jotka koostuvat erilaisista näytöistä ja toiminnoista. Sovellukset sisältävät useita moduuleita. Järjestelmän toiminta edellyttää, että perustana olevat tiedot ovat ajan tasalla.



Kuvio 4: LEAN System ohjelmisto – Sovelluksen suhteet toisiinsa /21, s.9/

Näiden staattisten tietojen ylläpitoon käytetään seuraavia exe-moduleita.

4.3 Pääohjelma

Lean System pääohjelma (lean.exe), josta jokainen moduuli aloitetaan (kuvio 5).



Kuvio 5: LEAN System pääohjelma /14,s.7/

4.4 Ylläpito

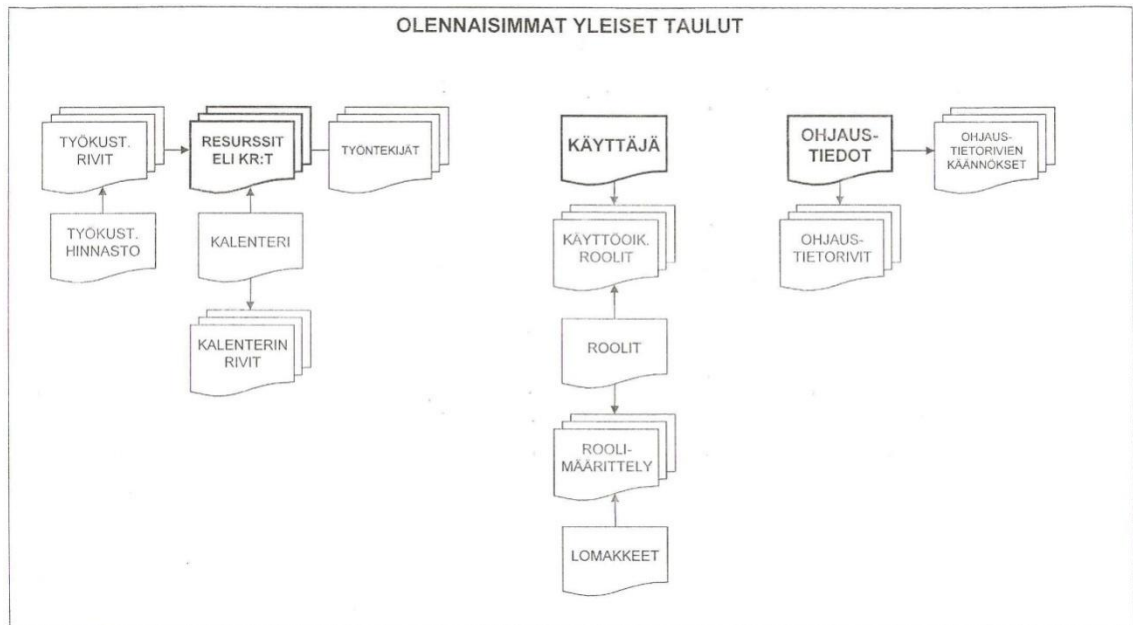
Lean System (lean_sys.exe) Ylläpito sovellus on ensisijaisesti järjestelmän pääkäyttäjän työkalu. Sovellus sisältää toiminnot mm. ohjaustietojen ylläpitoon sekä vastuualueiden, käyttöäoikeuksien ja tunnussarjojen hallintaan. Suurin osa näistä tiedoista määritellään Lean System järjestelmän käyttöönoton yhteydessä.

Ylläpito sovellukseen liittyvät tiedot ohjaavat pitkälle järjestelmän toimintaa ja vaativat siten myös suurta huolellisuutta muutettaessa aiempia asetuksia. /21, s. 9-10/

4.5 Perustiedot

Lean Perustiedot (lean_dlg.exe) - sovelluksessa ylläpidetään kaikkia järjestelmään liittyviä perustietoja, kuten yritys-, resurssi-, kalenteri-, hinnasto- ja kustannustietoja. Perustietojen ylläpito tapahtuu pääsääntöisesti ns. avainkäyttäjien toimesta. Muut käyttäjät pääsevät hyödyntämään näitä perustietoja operatiivisissa sovelluksissa (kuvio 6).

/21, s.10/

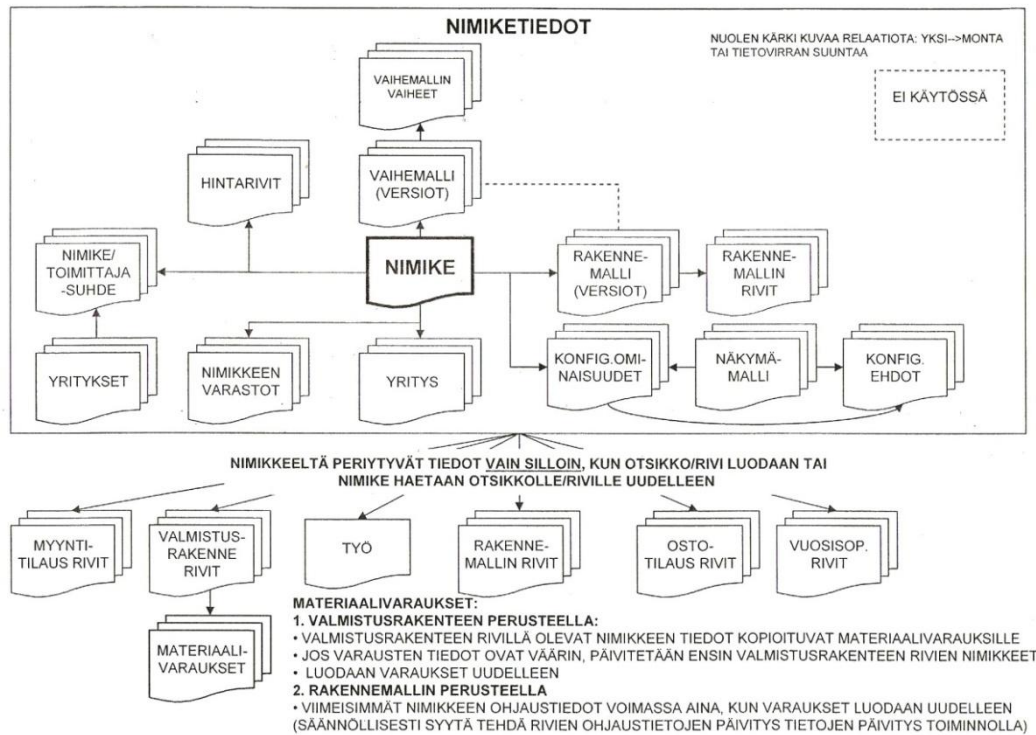


Kuvio 6: Lean System – Yleiset taulut /15, s.7/

4.6 Nimiketiedot

Lean Nimiketiedot (lean_bas.exe) – sovelluksessa ylläpidetään järjestelmän kannalta keskeisimmät tiedot, nimiketiedot, johon kuuluvat niin hinta-, rakenne- ja vaihemalli-tiedot kuin nimike/toimittaja- ja nimike/varastosuhteet. Nimiketiedot muodostavat perustan lähes kaikelle operatiiviselle toiminnalle järjestelmässä (kuvio 7).

Nimiketietojen ylläpitovastuu kuuluu usein nimetyille avainkäyttäjille, muut käyttäjät hyödyntävät nämä tiedot operatiivisessa toiminnassa. /21, s.10/



Kuvio 7: Lean System – Nimiketiedot /15, s.2/

4.7 Myynti

Lean Myynti (lean_sal.exe) – sovellus kattaa toiminnot, jolla hoidetaan myyntisuunnitelmien ja ennusteiden hallinta, tarjousten ja tilausten käsittely sekä toimitussopimukset ja niihin kohdistuvat kotiinkutsut. Myyntitilaukset perustetaan järjestelmään käyttäjän toimesta. Tilauksen tiedot pohjautuvat asiakas- ja nimiketietoihin, joita voidaan tarvittaessa muuttaa tilausta syötettäessä. Tilauksen hinnoittelussa voidaan käyttää nimikekohtaisia myyntihintoja tai hyödyntää erityyppisiä hinnastoja. /21, s.10/

Loogisen toimintoketjun eteneminen myynnistä tuotannonsuunniteluun voi yrityksen toimintatavoista riippuen tapahtua seuraavasti:

- Kysynnän ja tarjonnan tasapainottamiseksi käytetään Lean Tuotannonsuunnittelu – sovellusta (PTA), jolla tilausten/ennusteiden perusteella luodaan tuotantohjelma.
- Yksittäisestä myyntitilauksista luodaan suoraan työ tuotantoon.
- Tilaukseen liitetään valmistusrakenne. Valmistusrakenne kuvaa tilauksella myytyä asiakaskohtaista laitetta. Työt tuotantoon avataan valmistusrakenteen pohjalta.

Myyntiketju sisältää myös työkalut toimitusten muodostamiseksi tilausten perusteella sekä laskujen muodostamiseksi tilausten tai toimitusten perusteella. Toimitusten muodostamiseksi tilausten perusteella on olemassa kolme vaihtoehtoista tapaa /21, s.10-11/;

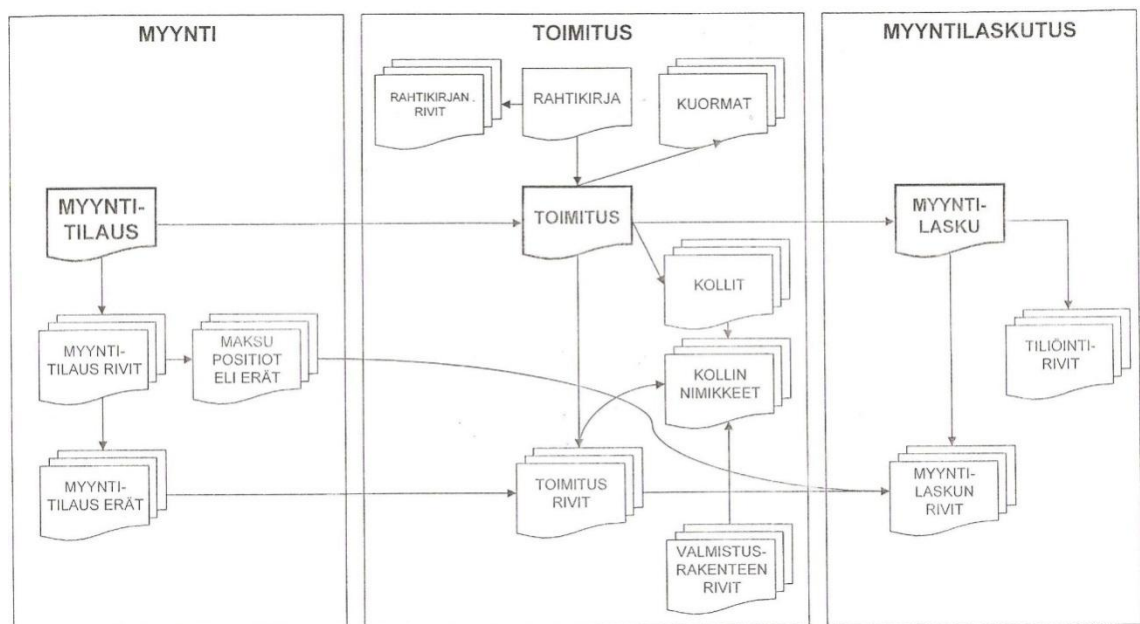
- Valituista myyntitilauksista tehdään toimitusten luontipyyntö, jonka tuloksena syntyy toimitusotsikoita ja toimitusrivejä suoraan tilaustietojen pohjalta.
- Valittujen myyntitilausrivien/tilauserien perusteella luodaan toimitustiedot. Tässä tapauksessa muodostetaan toimitusotsikko ja toimitusrivejä suoraan tilaustietojen pohjalta.
- Valituista myyntitilauseristä luodaan toimitusta varten keräilyrivejä. Samassa luodaan myös toimitusotsikko, jota vastaan keräilyrivit kuitataan.

4.8 Toimitus

Lean Toimitus (lean_del.exe) – sovellus sisältää toimitusten käsittelyn, näiden perusteella tapahtuvien kollien ja kuormien muodostamisen sekä proforma-laskujen, rahtikirjojen ja muiden vientiasiapapereiden tulostamisen (kuvio 8). Toimituksella tarkoitetaan listaa yhdelle asiakkaalle kerralla toimitettavista tuotteista. /21, s.11/

Toimitusten luonti voi tapahtua seuraavilla tavoilla:

- Suoraan myyntitilaukselta tai keräilylistan kautta.
- Toimitus-sovelluksen kautta valitsemalla halutut tilaukset.
- Käsin syöttämällä tai kopioimalla toimitus uudeksi.



Kuvio 8: Lean System – Tilaus ja toimitusketjun hallinta /15, s.3/

Toimituksen tietojen perusteella tulostetaan kaikki lähetykseen liittyvät dokumentit kuten, rahtikirjat, lähetysluettelot sekä muut huolinta- ja vientidokumentit.

4.9 Laskutus

Lean Laskutus (lean_inv.exe) – sovellus sisältää työkalut laskujen muodostamiseksi ja tulostamiseksi. Laskut luodaan joko manuaalisesti tai automaattisesti tilaus- tai toimitustietojen perusteella. Tilauksen perusteella muodostettu lasku voi perustua joko tilausrivien/erien tietoihin tai tilaukseen liitettyihin maksueriin. /21, s.11/

Laskujen muodostus voi myös tapahtua tausta-ajojen avulla. Automaattisesti luodun laskun tietoja voidaan tarvittaessa muuttaa ennen lopullista hyväksymistä. Hyväksytyjä laskuja voidaan edelleen tiliöidä ennen niiden siirtämistä taloushallinnon järjestelmiin. Laskutustietojen perusteella tulostetaan kuukausittain viranomaisten määräysten mukaan Intrastat-raportti.

4.10 Valmistusrakenne

Lean Valmistusrakenne (lean_con.exe) – sovelluksessa ylläpidetään asiakaskohtaisia toimitusrakenteita (kuvio 9). Jokainen toimitettava laite voidaan tarvittaessa kuvata omana rakenteena. Valmistusrakenteen perusteella luodaan työt sekä näille materiaali-varauksia ja työvaiheistuksia. Tarvittaessa voidaan myös rakenteen perusteella luoda materiaaleille ennakkovarauksia. /21, s.12/

Rivi	Nim.tunnus	Nim.nimi	O	i	t	d	h	Määrä	Yks.	Tyyppi	Luol
1	50001	Moottori / 4						1	kpl	Ostetaan	L501
2	20001	Kehikko M01						1	kpl	Rakennemalli	L201
3	50016	Hammaspyörä						1	kpl	Ostetaan	L511
4	50012	Hammaspyörä						1	kpl	Ostetaan	L511
5	21001	Akseli, M01-M02						1	kpl	Rakennemalli	L202
6	50107	Laakeri SKF49335						3	kpl	Ostetaan	L505
7	20101	Kela M01-02						1	kpl	Rakennemalli	L203
8	52103	Pultti 8x60						1	kpl	Ostetaan	L302
9	52108	Pultti 12x50						8	kpl	Ostetaan	L302
10	50013	Hammaspyörä						3	kpl	Ostetaan	L511
30	59009	Kaapeli F103913						1	kpl	Ostetaan	L581
40	54002	Ohjauksyksikkö CB45						1	kpl	Ostetaan	L541

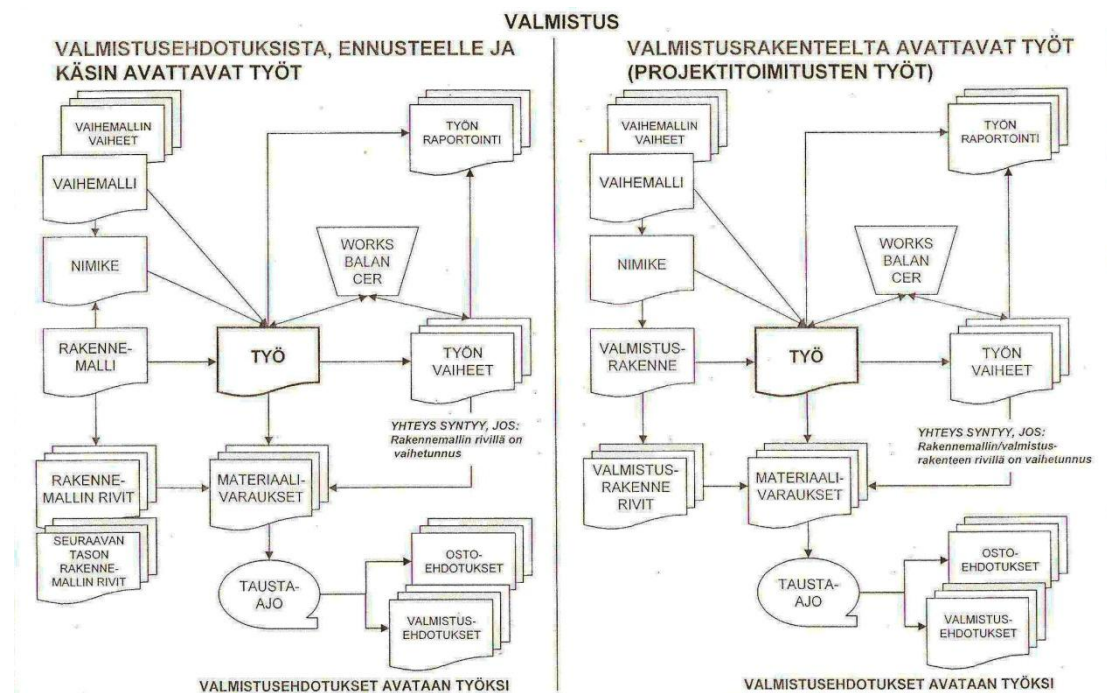
Kuvio 9: Lean System – Tuoterakenne /16, s.9/

4.11 Tuotannosuunnittelu

Lean Tuotannosuunnittelu muodostuu kolmesta osakokonaisuudesta, joita voidaan yrityksen toimintatavasta riippuen hyödyntää joko erikseen tai yhdessä. /21, s.12/

Vakiotuotantoympäristössä yksinkertaisin tapa on käyttää materiaaltarpeiden laskentaan tarkoitettua tausta-ajoa (tarpeiden laskenta), joka tuotannosuunnitelman perusteella generoi tarpeita ostettaville nimikkeille sekä valmistusehdotuksia valmistettaville osille.

Vaiheistustaulu esittää työt tai projektit ja niihin liittyvät rakenteet. Kuormitustaulu sisältää resurssien kuormitustilanteet. Aikataulumuutosten vaikutukset kuormitustilanteeseen näkyvät käyttäjälle välittömästi (kuvio 10).



Kuvio 10: Lean System – Tuotannon suunnittelu ja ohjaus /15, s.5/

Lean Palvelutyöasema (PTA) on vakiotuoteympäristöön soveltuva graafinen suunnittelumoduuli. Palvelutyöaseman avulla suunnitellaan tuotanto-ohjelmat. Tilaukset voidaan kohdistaa tuotantoerille tai palvelu varastosta. Varastoprofiilit nähdään todellisen tuotanto-ohjelman mukaisina. Palvelutyöaseman pääikkuna on suunnitteluikkuna, jonka avulla saadaan helposti ja nopeasti kokonaiskuva tuotannon tilanteesta.

Lean Ohjaustyöasema (OTA) on projektien, töiden ja resurssien ohjaukseen tarkoitettu työasemaohjelmisto. Ohjaustyöaseman runkona on kaksi suunnittelutaulua.

4.12 Valmistus

Lean Valmistus (lean_wor.exe) – sovellus sisältää tuotannon töiden ohjauksen, raportoinnin ja seurannan. Valmistus – sovelluksessa voidaan käsitellä myyntitilausriveiltä, valmistusrakenteelta, tarvelaskennan tai PTA:n avulla sekä manuaalisesti luotuja töitä, niihin liittyviä materiaalivarauksia ja ajoitettuja työvaiheita. Töiden tietojen perusteella tulostetaan työkortit, vaihekohtaisine materiaaleineen./21, s.12-13/

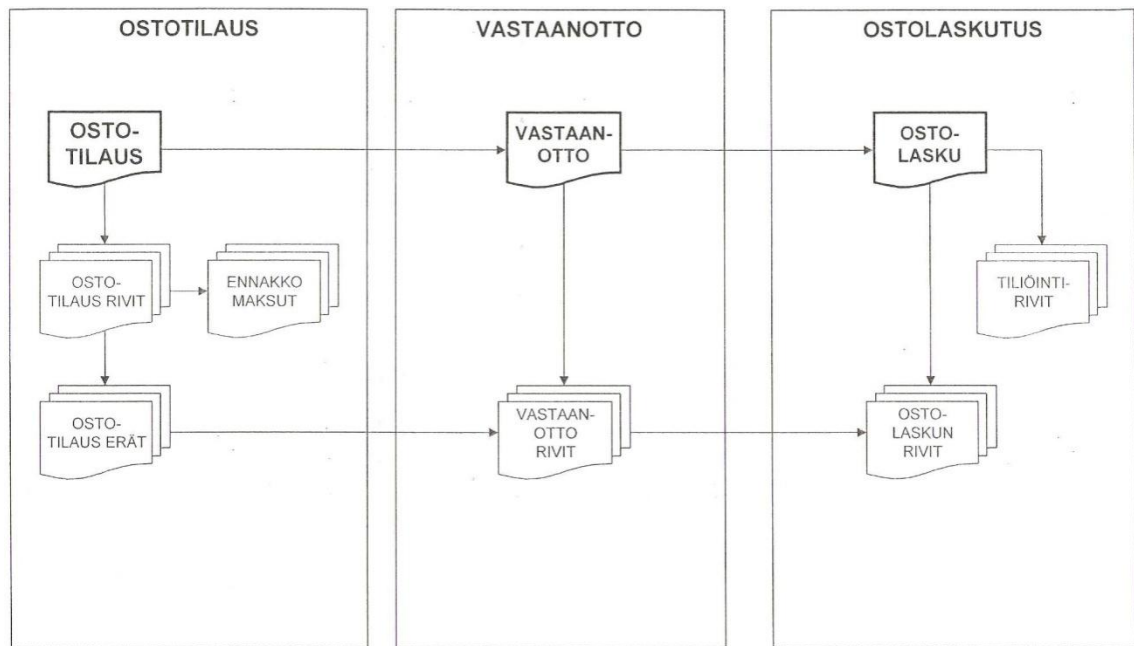
Töille kohdistuvat materiaalikustannustapahtumat syntyvät joko vastaanottaessa materiaalia suoraan työlle, keräilyn kuittauksen yhteydessä tai viimeistään ns. jälkipoistona työn valmistumisilmoituksen yhteydessä. Työkustannukset saadaan raportoimalla tehdyt tunnit tai käyttäen hyväksi vaiheistuksen suunniteltuja tunteja.

Työ päätetään ilmoittamalla työ valmistuneeksi valmistumisilmoituksella (kuvio 10).

4.13 Osto

Ostotilaus (lean_pur.exe) voi luonteeltaan olla varsinainen tilaus, vuosisopimus, kotiinkutsu, tarjouspyyntö tai ostoennuste (kuvio 11). Impulssi ostoon saadaan automaattisesti materiaalivarausten, hälytysrajojen ja myyntierien perusteella muodostettavista ostoehdotuksista. Yhteistyö vakiotoimittajien kanssa tapahtuu usein vuosisopimusten ja niihin kohdistuvien kotiinkutsujen perusteella. /21, s.13/

Ostotilaukset tehdään joko automaattisesti ostoehdotusten perusteella, kotiinkutsujen muodossa tai suoraan käsin syöttämällä.



Kuvio 11: Lean System – Ostoketjun hallinta /15, s.4/

Tilauksen muodostuksessa käytetään oletusarvoina toimittajan ja nimikkeen perustietoja. Tilausten maksuehtojen ja arvioitujen toimitusaikojen perusteella saadaan pohja vaLuuttakohtaisille rahantarve-ennusteille tulevaisuudessa.

4.14 Vastaanotto

Lean (lean_rec.exe) – järjestelmä käsittelee vastaanotot kaksitasoisesti (kuvio 11). Vastaanotto-otsikko edustaa yhtä fyysisen vastaanottoa ja liittyy yleensä samaan rahtikirjaan. Vastaanottoriveillä määritellään vastaanotettavat nimikkeet ja niihin liittyviä tietoja. Vastaanotot muodostetaan järjestelmään joko tilausten pohjalta tai käsin syöttämällä. Yhteen vastaanottoon voi liittyä useita saman toimittajan tilauksia. /21, s.14/

Varastotapahtumat, ostotilauksen toteumatiedot ja toimittajaan liittyvät tilastotiedot ylläpidetään päivitettyjen vastaanottotapahtumien perusteella.

4.15 Ostolaskuntarkastus

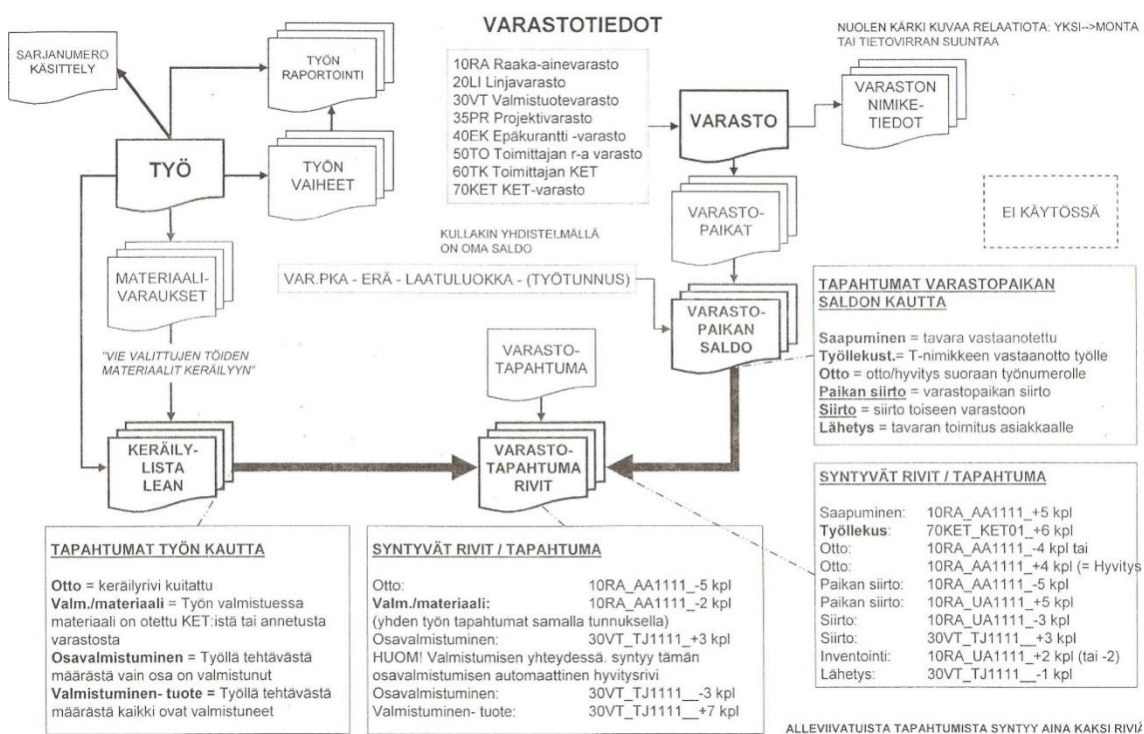
Oletustiedot ostolaskun tarkastusta (lean_pin.exe) varten muodostetaan pääsääntöisesti joko ostotilauksen tai vastaanoton tietojen perusteella. Ostolasku voi myös liittyä ostotilaukselle perustettuihin maksueriin. Laskun tietoja voidaan päivittää ja muuttaa ennen

laskun hyväksymistä. Hyväksytyt ostolaskut tiliöidään, jonka jälkeen ne ovat siirrettävissä reskontraan. /21, s.14/

4.16 Varasto

Lean Varasto (lean_sto.exe) – sovellus sisältää tarvittavat toiminnot varastojen ja materiaalien saldojen hallintaan. Samaa nimikettä voidaan varastoida eri varastoissa ja varastopaikoissa vielä siten, että tarvittaessa jokainen saapumiserä kirjataan omaksi saldoeräksi, jäljitystunnuksella varustettuna. Myös töille tai myyntitilauksille varatut saldoerät näkyvät erillään varastosaldoissa (kuvio 12).

Järjestelmän sisältämän monivarastokäsittely ominaisuuden avulla hallitaan nimikkeiden ohjaus varastokohtaisesti sekä luodaan automaattisesti varastojen välisiä täydennyssimpulsseja. On myös mahdollista, että varsinaiset varastotapahtumat tehdään muissa sovelluksissa kuten tavaran vastaanotto, tavaran toimittaminen ja materiaalien keräilyyn kuittaaminen työlle. /21, s.13/



Kuvio 12: Lean System – Varaston hallinta /15, s.6/

Varasto – sovelluksen varsinaiset operatiiviset tehtävät liittyvät varaston sisällä tapahtuviin toimintoihin. Tällaisia toimintoja ovat varastojen tai varastopaikkojen väliset siirrot, inventointi, käsitapahtumat ja varastosta otot ilman erillisiä varauksia.

4.17 Raportointi

Raportointi (lean_rep.exe)– sovellus käytetään järjestelmän vakioraporttien käynnistykseen. Lisäksi sovellus sisältää toteumatietoja eri toimintojen tapahtumista.

/21, s.15/

4.18 Tiedonsiirto

Leanin tiedonsiirrosta huolehtii Lean transaktio ja MSS viestinkäsittely (Lean System/Locos). (lean_mss.exe, lean_tra.exe).

4.19 Taustaprosessi

Lisäksi järjestelmässä on joukko taustaprosesseja, joista huolehtii Lean tausta-ajot (lean_bap.exe). /21, s.15/

4.20 Projektinohjaus

Lean Projektit on moniyritys- ja moniprojektiympäristöön tarkoitettu projektien suunnittelun ja seurannan kokonaisjärjestelmä. Järjestelmä sisältää sekä aikataulu- ja kuormitusohjauksen että taloussuunnittelun ja -seurannan toiminnot. /21, s.14/

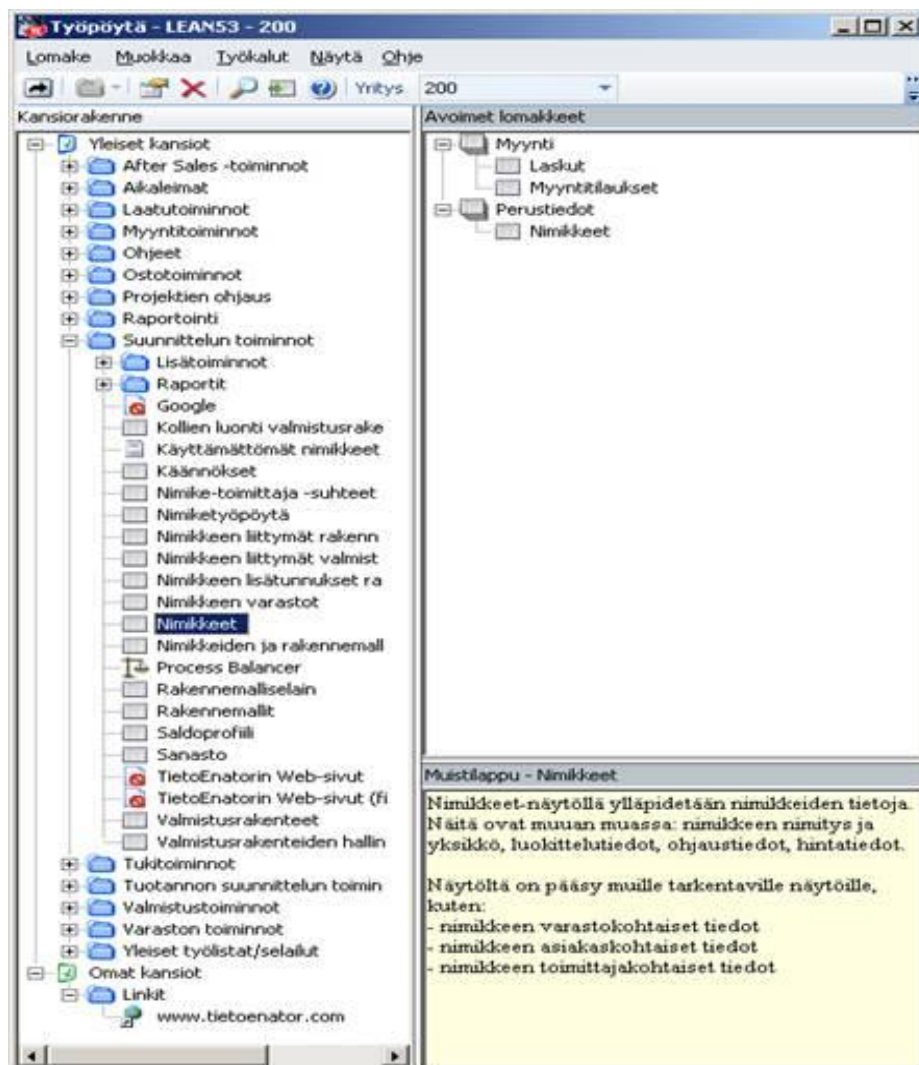
Liitettynä Lean System -järjestelmäkokonaisuuden muihin sovelluksiin Lean Projektit -järjestelmästä muodostuu projektitoimintaa harjoittavan yrityksen tai yksikön toiminnanohjausjärjestelmä, jossa voi olla mukana mm. myynnin, hankinnan, varaston, tuotteenhallinnan ja valmistuksen ja laskutuksen osakokonaisuudet.

Projekti koostuu pääprojektista sekä sen kanssa hierarkkisessa suhteessa olevista alaprojekteista/aktiviteeteistä, jotka yhdessä muodostavat projektihierarkian.

5 Lean System – Sovellusten peruskäsitteet

5.1 Lean System työpöytä

Sovellukset käynnistetään LEAN Työpöydästä. Erikseen haluttaessa voidaan yksittäinen sovellus käynnistää myös suoraan Windowsin resurssienhallinnasta käsin. LEAN-tietokantaan kirjaudutaan joko käynnistettäessä LEAN Työpöytä tai käynnistettäessä yksittäinen sovellus. Käynnistettäessä sovelluksia LEAN Työpöydästä, ei tietokantaan enää kirjauduta erikseen (kuvio 12). /14, s.4-5/



Kuvio 12: Lean System työpöytä /14, s.5/

LEAN Työpöytä on Lean Systemin eri osien sekä muiden ohjelmien yhteiskäyttöä varten suunniteltu integroiva sovellus (kuvio 13). Se helpottaa, yksinkertaistaa ja jäsentää järjestelmän käyttöä eri käyttäjäryhmissä ja tilanteissa. LEAN Työpöytä mahdollistaa erilaisten näkymien luonnin eri käyttäjäryhmille tai kullekin käyttäjälle erikseen.

5.2 Lean System sovellukset

LEAN System asiakasohjelmisto (client) koostuu seuraavista sovelluksista: /14, s.5-6/

- LEAN Työpöytä
- LEAN Ennusteet
- LEAN Jälkimyynti, Service
- LEAN Kalenteri
- LEAN Kysynnänhallinta
- LEAN Laatumiedot
- LEAN Myynti
- LEAN Osto
- LEAN Perustiedot
- LEAN Projektinohjaus
- LEAN Raportointi
- LEAN Tausta-ajot
- LEAN Tehtäväsuunnittelu
- LEAN Tiedonsiirrot
- LEAN Valmistus
- LEAN Varasto
- LEAN Ylläpito

Järjestelmään on mahdollisuus liittää seuraavat suunnittelutyökalut:

- LEAN Process Balancer
- LEAN Project Balancer
- LEAN Purchase Balancer
- LEAN Works Balancer

LEAN System -järjestelmään kuuluvat myös seuraavat palvelinohjelmat, joiden käyttöä ei ole kuvattu tässä ohjeessa:

- LEAN Tausta-ajopalvelin
- LEAN Kuormituspalvelin
- Lisäksi järjestelmään on mahdollisuus liittää web-selaimessa toimivia Lean Portal –toimintoja.



Kuvio 13: LEAN System –sovellukset /14, s.6/

6 Lean System - Lomakkeet ja tulostus

Tietoja hallitaan ensisijaisesti taulukko-tyyppisillä selailulomakkeilla (kuvio 14). Näiden lisäksi järjestelmään kuuluu vuorovaikutteiseen grafiikkaan perustuvia suunniteluohjelmia. Lomakkeet koostuvat vakio-osista ja -toiminnoista, joiden avulla tietojen hallinta on helppoa ja yhtenäistä koko järjestelmässä. Tässä kohdassa on kuvattu kaikille lomakkeille yhteisiä piirteitä. Lomaketyyppejä on järjestelmässä muutama: otsikkotiedot, niihin liittyvät rivitiedot ja tietojoukkoihin liittyvät kaikki tiedot. /14, s.20-21/

The screenshot shows a software window titled 'Tilaukset - Kaikki - DEMI LEANSZ'. The window contains a table with columns: Tunnus, i, t, d, h, Asiakas, Asiakkaan nimi, Tyyppi, Tila, Til.pvm., Toim.pvm., and Lähtöpvm. The table lists 15 orders. Numbered callouts (1-14) point to specific UI elements: 1 (Title bar), 2 (Menu bar), 3 (Title field), 4 (Toolbar), 5 (Filter dropdown), 6 (Summary dropdown), 7 (Table border), 8 (Grouping button), 9 (Refresh button), 10 (Table header), 11 (Table scroll bar), 12 (Table content), 13 (Type filter), and 14 (Status filter).

Tunnus	i	t	d	h	Asiakas	Asiakkaan nimi	Tyyppi	Tila	Til.pvm.	Toim.pvm.	Lähtöpvm.
MT00001					1000	Bike Planet	Tilaus	Vahvistettu	06.07.06	15.09.06	15.09.06
MT00002					1002	Gamebike	Tilaus	Vahvistettu	30.06.06	01.08.06	01.08.06
MT00003					1014	Cykel Specialisten	Tilaus	Vahvistettu	26.06.06	20.07.06	21.07.06
MT00004					1000	Bike Planet	Tilaus	Vahvistettu	23.08.06	16.10.06	16.10.06
MT00006					1002	Gamebike	Tilaus	Kesken	24.08.06		29.08.06
MT00007					1005	Prisma Sello	Tilaus	Vahvistettu	28.08.06	05.09.06	05.09.06
MT00008					1005	Prisma Sello	Tilaus	Osittain toim.	28.08.06	05.09.06	05.09.06
MT00010					1004	Prisma Olari	Tilaus	Toimituksessa	29.08.06	30.08.06	
MT00011					1004	Prisma Olari	Tilaus	Vahvistettu	26.09.06	29.09.06	
MT00012					1004	Prisma Olari	Tilaus	Kesken	26.09.06		
MT00013					1004	Prisma Olari	Tilaus	Vahvistettu	26.09.06	15.10.06	
MT00014					1003	Prisma Iso Omena	Tilaus	Kesken	26.09.06	14.10.06	14.10.06
MT00015					1004	Prisma Olari	Tilaus	Kesken	26.09.06	06.11.06	06.11.06

Kuvio 14: Lean System taulukkolomake /14, s.21/

Taulukkolomakkeet koostuvat seuraavista alueista:

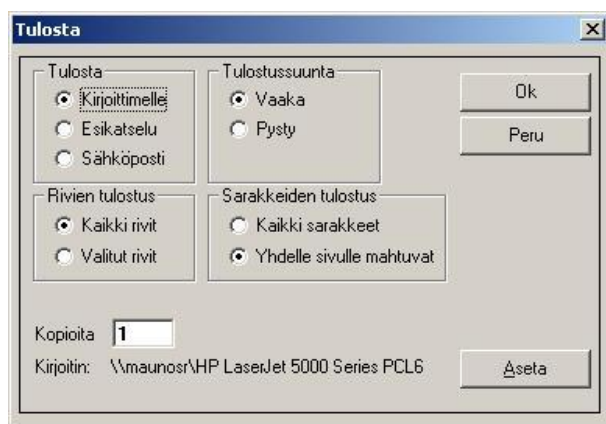
- 1) otsikko
- 2) valikko
- 3) lomakkeen tila (oikeassa yläkulmassa)
- 4) työkalupalkki, jossa painikkeet
- 5) synkronoinnin valintakenttä
- 6) summauksen valintakenttä
- 7) rajausten näyttöpalkki
- 8) ryhmittely-painike
- 9) pyyhi-painike
- 10) rajauskentät
- 11) tallennettujen rajausten pikavalinta
- 12) taulukko
- 13) tyyppi-tilakohtaiset toimintopainikkeet
- 14) tilarivi

LEAN-järjestelmästä saatavat tulosteet voidaan jakaa kolmeen ryhmään:

- 1) listaukset
- 2) asiakirjat
- 3) raportit

Listaukset ovat yksinkertaisia listamuotoisia tulosteita. Listaukset saa tulostettua kaikista taululomakkeiden sisällöistä, jolloin niihin tulostuu lähes samat tiedot kuin lomakkeen taulussa näkyy. /14, s. 55-57/

Listaukset tulostetaan taululomakkeiden Lomake | Tulosta listaus –valinnalla (kuvio 15).



Kuvio 15: Lean System – Tulosta listaus /14, s.56/

Asiakirjat ovat järjestelmästä saatavia ns. virallisia asiakirjoja, joita ovat esim. lasku, ostotilaus ja tarjous. Asiakirjat tulostetaan lomakkeen Lomake | Tulosta asiakirja –valinnalla.

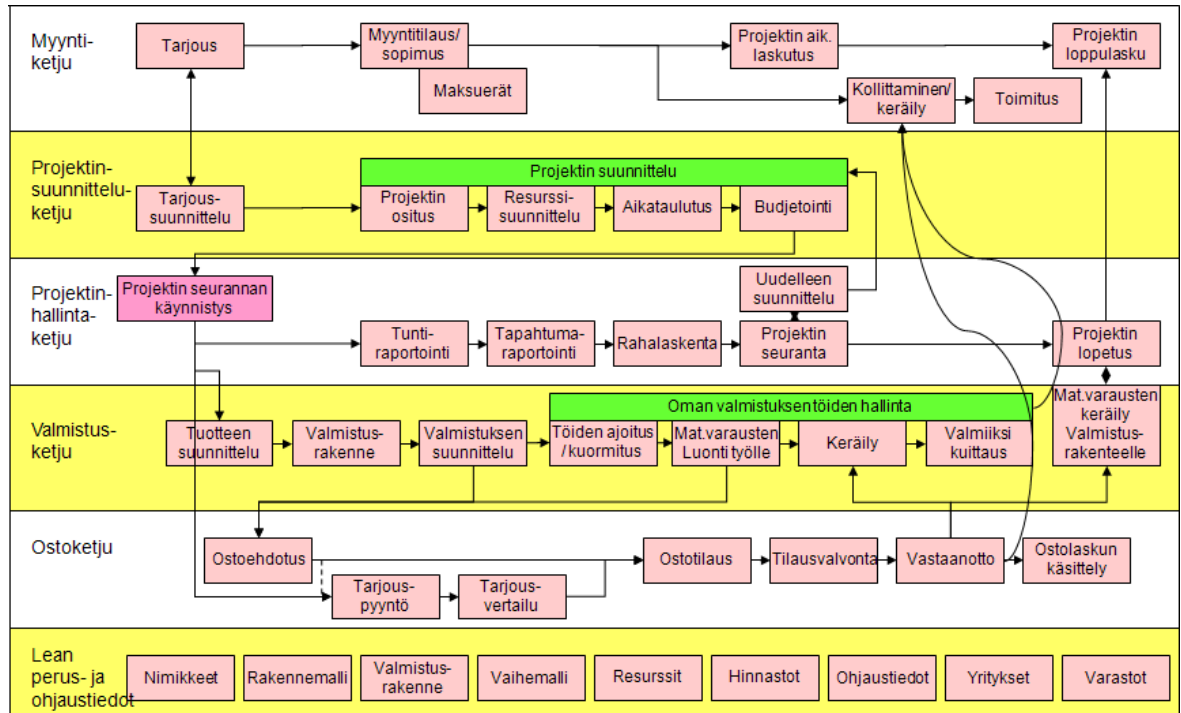
Asiakirjan tulostusikkunassa on Päivitä tulostetuksi –valintaruutu. Lisäksi ikkunassa voi olla muita tietojoukkokohtaisia kenttiä. Tietojoukoilla, joihin liittyy asiakirjoja, on kannassa tulostuskentät, joista nähdään onko asiakirjan ns. virallinen versio tulostettu. Päivitä tulostetuksi -valinta päivittää kannan tulostuskentät käyttäjän niin halutessa.

Varsinaiset raportit käsittävät kaikki muut tulosteet. Raportit käynnistetään ja raporttivalintoja selailaan Lean Työpöydästä tai LEAN Raportointi -sovelluksen raporttien selailulomakkeelta käsin.

Raporttien käyttöoikeudet hallitaan vastaavasti kuin järjestelmän lomakekohtaiset käyttöoikeudet. Käyttöoikeusroolien roolimäärittelyssä kerrotaan mihin raporteihin ko. roolilla on tulostusoikeudet. /14, s.57/

7 Tilaus- ja toimitusketjun hallinta

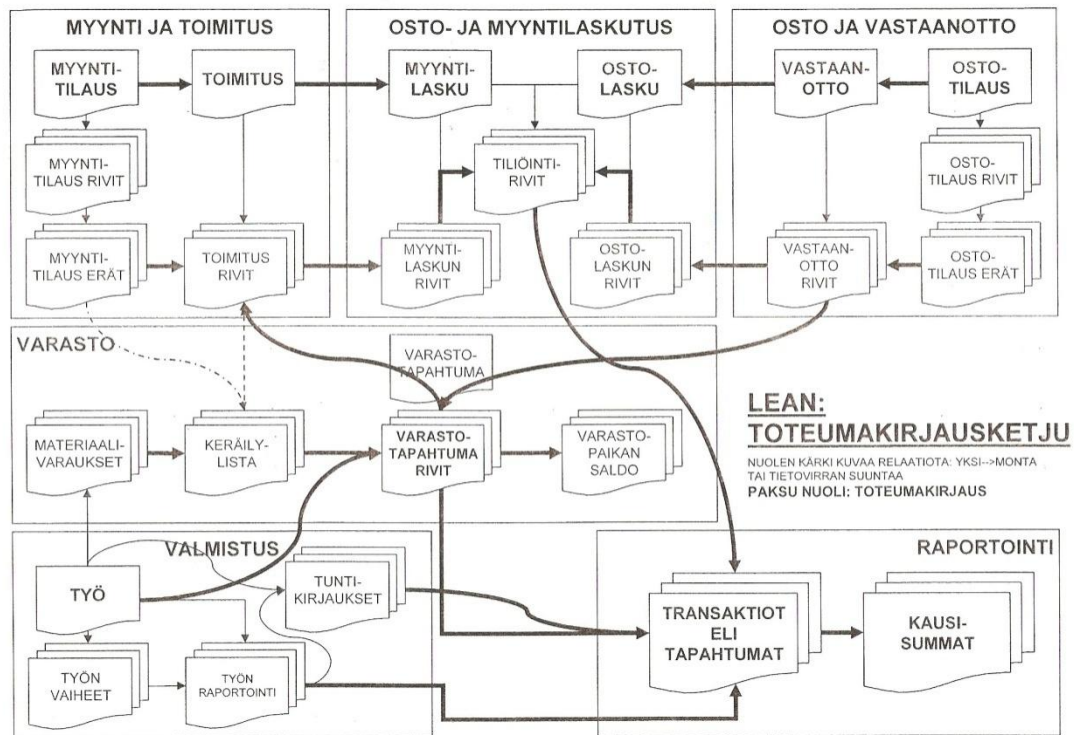
Tilaus- ja toimitusprosessi kattaa kaikki vaiheet asiakkaan tilauksesta toimituksen vastaanottoon. ERP:n tilaus- ja toimitusketjun hallintaa kuvataan kuviossa 16.



Kuvio 16: ERP sisältö kaaviona /16, s.6/

Lean System toteutumakirjausketju jakaantuu kuuteen eri lohkoon. Käyttöliittymä tasolla toiminnot ovat melkein samankaltaisia, mutta tilauksen kirjausketjusta riippuen käsitteistö ja tapahtumakirjaus nimikkeistö eroavat toisistaan. Kaikkien ketjussa olevien toimintojen on toimittava saumattomasti yhteen. Tiedon virheettömyys ja oikeudellisuus vahvistaa oikean tiedon perille menoa ketjussa.

Lean System jokainen toiminto on jaettu erilaisiin ketjuihin, jotka on kuvattu toteutumakirjausketjussa (kuvio 17).



Kuvio 17: Lean System: Toteumakirjausketju /15, s.8/

Kuten kuviosta 17 selviää, kaikki staattinen tieto tarkistetaan ja käytetään oletuksena, kun luodaan uutta tilausta ja edetään toteumakirjausketjussa eteenpäin.

Seuraavaa tietojen hierarkkia käytetään, kun tietoa käsitellään eri Lean System sovelluksissa:

- a) ylemmän tason hierarkkian otsikkotiedot (header)
- b) alemman tason hierarkkian rivitiedot (row).

Mallikaavio (Model Diagram) on paljolti käytetty yksityiskohtainen kuvaus toteumakirjausketjun eri osa-alueista sovelluksen käyttöliittymätasolla. Tällä pyritään ohjeistamaan sovelluksen käyttäjää toteumakirjausketjun oikeaan hallintaan ja käyttöön. /21/

Esimerkkinä: Mallikaaviossa on kuvattu myyntitilauksen ja toimituksen ja ostotilauksen vahvistusilmoituksen -käsittelyt. Myös prosesseihin liittyvät tietokantataulut on kuvattuna, joihin tieto päivittyy (Taulukot 4 ja 5).

Taulukko 4: Myyntitilauksen ja toimituksen käsittely ja tietokantataulut

<p>Tehdään uusi myyntitilaus</p> <ul style="list-style-type: none"> • avataan myyntisovellus Lean Systemin päävalikosta • valitaan Windows/MSS sanomaviesti tai kirjataan käsin vastaanotettu fax/puhelinsoitto/e-mail/ tilaus myyntitilaussovellukseen • tarkistetaan ja asetetaan pakolliset järjestelmän vaatimat myyntitilaustiedot 	←	<p>XMS_SOH_45 XMS_SNOTICE_45 XMS_SOR_45 XMS_SOL_45</p>
<p>Käsitellään myyntitilaus</p> <ul style="list-style-type: none"> • avataan Myyntitilaus (kaikki tiedot) ja Rivitiedot –lomakkeet ja tarkistetaan pakolliset tiedot • valitaan Työkalut/Käsittele myyntitilausviesti • valitaan Rakenne tarkistus –laatikko ja lasketaan hinnat myyntitilaukselle • valitaan Window/Tilaukset 	→	<p>SAL_ORDER SAL_ORDERROW CON_STRUCT CON_STRUCTROW</p>
<p>Tarkistetaan myyntitilaus</p> <ul style="list-style-type: none"> • valitaan tilaus ja klikataan Myyntitilaus (kaikki tiedot) -lomaketta ja tarkistetaan ja päivitetään pakolliset pääotsikko tiedot • klikataan Rivitiedot –lomaketta ja tarkistetaan myyntitilaus rivit (1:n tason nimikkeet) ja pyydetty toimitus päivämäärä 	↔	<p>SAL_ORDER SAL_ORDERROW (CON_STRUCT) (CON_STRUCTROW)</p>
<p>Tarkistetaan ja konfiguroidaan rakenteet</p> <ul style="list-style-type: none"> • valitaan myyntitilaus ja klikataan Rakenne -lomaketta • klikataan Rakenne (kaikki tiedot) –lomaketta ja tarkistetaan rakenne tiedot • klikataan Rakennerivitiedot -lomaketta ja tarkistetaan rakenteen rivitiedot 	↔	<p>CON_STRUCT CON_STRUCTROW</p>
<p>Tarkistetaan kapasiteetti</p>		
<p>Muokataan myyntitilausta jos on tarvetta</p> <ul style="list-style-type: none"> • vaihdetaan vahvistettu toimituspäivämäärä Myyntitilaus rivi- ja Rakennerivi –lomakkeella • jaetaan (splittaus) tilausrivit jos on tarvetta 	↔	<p>SAL_ORDER SAL_ORDERROW CON_STRUCT CON_STRUCTROW</p>
<p>Luodaan ostoehdotus (back-to-back)</p> <ul style="list-style-type: none"> • valitaan Työkalut/Luodaan ostoehdotus myyntitilaus ja Rakenne riveille • asetetaan eräkoko jos on tarvetta • odotetaan kunnes systeemi on luonut tilauksen selailulomake ikkunaan 	↔	<p>SAL_ORDER SAL_ORDERROW CON_STRUCT CON_STRUCTROW</p>

<p>Luodaan ostotilaus</p> <ul style="list-style-type: none"> vahvistetaan että kaikki rivit, jotka halutaan ostotilaukseen on valittu (tummaväri) valitaan Työkalut/Luodaan uusi ostotilaus vahvistetaan että tilaus tyyppi on oikea (tilaus tai varaus) 	<p>↔</p> <p>→</p>	<p>PUR_ORDERBATCH</p> <p>PUR_ORDER</p> <p>PUR_ORDERROW</p>
<p>Lähetään ostotilaus (tai tulostetaan)</p> <ul style="list-style-type: none"> klakataan tilaus Ostotilaus –lomake ikkunassa valitaan Työkalut/Luodaan MSS Tilausviesti Ostotilaus – lomake ikkunassa 	<p>→</p>	<p>XMS_POH_45</p> <p>XMS_PNOTICE_45</p> <p>XMS_POR_45</p>
<p>Päivitetään ostotilaus varaukset</p> <ul style="list-style-type: none"> jos koko varattu määrä on ”syöty”, luodaan ja lähetään uusi ostotilaus varaus luodaan MSS viesti (Toiminto:Keskeytys) - Ilmoitus: ”Syöty varaus”. (kuten lähetettäisiin uusi ostotilaus) 	<p>→</p>	<p>SAL_ORDER</p> <p>SAL_ORDERROW</p> <p>CON_STRUCT</p> <p>CON_STRUCTROW</p>
<p>Vahvistetaan myyntitilaus (tavaran toimittajalle)</p> <ul style="list-style-type: none"> palataan takaisin Myyntitilaus –lomakeelle valitaan tekemäsi Myyntitilaus ja Työkalu/Luodaan MSS Vahvistettu viesti 	<p>→</p>	<p>XMS_SOH_45</p> <p>XMS_SNOTICE_45</p> <p>XMS_SOR_45</p> <p>XMS_SOL_45</p>

Taulukko 5: Ostotilauksen vahvistus ja tietokantataulut

<p>Tehdään uusi ostotilauksen vahvistusilmoitus</p> <ul style="list-style-type: none"> avataan Ostosovellus valitaan Windows/MSS sanomaviesti Ostosovelluksen Lean Systemin päävalikosta valitaan Näytä/Käsitellään ja Hyväksytään toiminto – haku kriteerinä esim. vastuuhenkilön käyttäjätunnus 	<p>←</p>	<p>BMS_POH_45</p> <p>BMS_POR_45</p>
<p>Käsitellään vahvistusilmoitus</p> <ul style="list-style-type: none"> valitaan kaikki ostotilauksen vahvistusilmoitukset jotka haluat käsitellä valitaan Työkalu/Käsittele vahvistusilmoitus 	<p>↔</p>	<p>PUR_ORDERROW</p> <p>CON_STRUCT</p> <p>CON_STRUCTROW</p> <p>SAL_ORDER</p> <p>SAL_ORDERROW</p>

8 Ohjelmoinnin työvälineet

Lean System / Locos ohjelmiston lähdekoodissa on käytetty seuraavia ohjelmoinnin työvälineitä.

8.1 SQL Windows

SQLWindows on sovelluskehitin, jonka ohjelmointikielenä on olio-ohjelmointisuuntautunut SAL (SQLWindows Application Language). Se soveltuu tietokantapohjaisten Client-Server -sovellusten tekemiseen MS-Windows- ja Linux-ympäristöön. Tämä graafisen sovelluskehittimen ja ohjelmointikielen yhdistelmä tunnetaan myös nimillä Centura sekä TD (Team Developer).^{9/}

SQLWindows tukee ohjelmistokehityksessä ryhmätyöskentelyä. Lähdekoodi voidaan pilkkoa kirjastoiksi, joita samanaikaisesti muokataan eri tahoilla, ja kun yhden kirjaston muutokset ovat valmiit ja kirjasto siirretään työryhmälle jakoon, uusi versio lähdekoodikirjastosta on heti kaikkien muiden nähtävissä ja käytettävissä.

SQLWindows käyttää samantapaisia rakenteita (If, Select Case, While ...) kuin mitä esiintyy esimerkiksi C-kielessä, mutta SQLWindowsin perusajatuksena on toteuttaa ohjelmointi lähettämällä ja vastaanottamalla viestejä graafisten komponenttien välillä. Viestit voivat olla sekä Windows-käyttöjärjestelmän lähettämiä (esimerkiksi WM_LBUTTONDOWN), ohjelmointikielen omia (esimerkiksi SAM_Click), tai ohjelmoijan itse määrittelemiä.

Viestien käsittely tarkoittaa sitä, että sovellusikkuna tai sovellusikkunassa oleva elementti (esimerkiksi painike) saa monenlaisia viestejä, mutta sovelluksessa erikseen kerrotaan, mihin viesteihin halutaan reagoida ja millä tavalla. Esimerkiksi kun käyttäjä napsauttaa sovellusikkunassa olevaa painiketta, painike saa viestin SAM_Click. Painikkeen alle koodissa voidaan sitten laittaa Message Actions -lohkoon rivin ”On SAM_Click” sekä sen alle tehtävät toimenpiteet. (Listaus 1).

```

On SAM_Click
    !
    !   Open dialog for selecting scarp type.

If GetEnumByNabbr ( hWndForm, 'SCRAPTYPE', TRUE, FALSE,
    TXT_GETSCRAPTYPE_HDR, ", ", 'AND VALUE12 = 1',
    strSelectId, strSelectName )
    !
    !   Set the selected value to TableWindow column and clear the
    !   transfer variable.
    !
    Set strScrapType           = strSelectId
    Set strScrapTypeName      = strSelectName
    Set strSelectId           = "
    Set strSelectName         = "
    Call ChangeColor(dlgCancelShipment.strScrapTypeName)

```

Listaus 1: Lean System/Locos – SQL Windows – SAM_Click koodia

8.2 PL/SQL

PL/SQL on Oraclen ohjelmointikieli, jolla voi tehdä tietokantaläheisiä ohjelmia. Ohjelmat kirjoitetaan jollain editorilla, ladataan tietokantaan esimerkiksi SQL-työkalulla (SQLPlus) ja käännetään tietokannassa. Ohjelmia voi ajaa tietokannassa, jos ne on sekä ladattu että käännetty tietokantaan. PL/SQL ohjelmia toteutetaan sovittujen menetelmien mukaisesti. (Listaus 2a ja 2b)./17/

```

PROCEDURE CheckSORChanges(
    astrCompany          SAL_ORDERROW.COMPANY%TYPE,
    astrDep              SAL_ORDERROW.DEP%TYPE,
    astrUser             SAL_ORDERROW.UPDATED_BY%TYPE,
    anSohMsgId          BMS_SOH_45.RECID%TYPE,
    -- RECID of the SAL_ORDERROW table row that is to be updated
    anRowRecId          SAL_ORDERROW.RECID%TYPE,
    -- new SO row contents
    arSalOrderRow       SAL_ORDERROW%ROWTYPE,
    nStockChange        NUMBER,
    abChanged           OUT BOOLEAN,
    nOptions            NUMBER)

```

Listaus 2a: Lean System/Locos – SQL Windows – PL/SQL koodia

```

CURSOR cOldSORow IS
    SELECT      ITEMID,
               ITEMNAME,
               ...
               QUALITYSPEC
    FROM SAL_ORDERROW
    WHERE RECID = anRowRecId;
    rSORowOld   SAL_ORDERROW%ROWTYPE;
    strDiff     VARCHAR2(32768);
-- bChanged is used to record whether there are any differences between the new and old values.
-- In case of SAL_ORDERROW this information is stored in USR_STATUS3 field to be later used in
the
-- PO handling phase.
    bChanged    BOOLEAN := FALSE;

BEGIN
    OPEN cOldSORow;
-- fetch the old values of the SAL_ORDERROW table row that is going to be updated
    FETCH cOldSORow INTO rSORowOld;
    IF cOldSORow%NOTFOUND THEN
        anError := OLDSO_NOT_FOUND;
        CLOSE cOldSORow;
        RETURN;
    END IF;
    CLOSE cOldSORow;
-- initialize the difference text buffer
    strDiff := 'User: ' || astrUser || ' handled at: ' ||
        TO_CHAR(SYSDATE, 'dd.mm.yyyy hh24:mi:ss') || ' message ID = ' ||
        TO_CHAR(anSohMsgId) || ' ';
-- compare the old and new field values one by one and record the difference
    IF rSORowOld.ITEMID <> arSalOrderRow.ITEMID OR
    OR      ( rSORowOld.ITEMID IS NOT NULL AND arSalOrderRow.ITEMID IS NULL )
    OR      ( rSORowOld.ITEMID IS NULL AND arSalOrderRow.ITEMID IS NOT NULL )
    THEN
        bChanged := TRUE;
        strDiff := strDiff || ' Item ID old = ' || rSORowOld.ITEMID ||
            ' new = ' || arSalOrderRow.ITEMID ;
    END IF;

    ..... repeat the check for all relevant Sales Order Row fields

-- when all the fields have been compared, attach the difference text to the
-- SO row in question
    IF bAddTextLinkOption THEN
        LIB_TEXT_002.AddText(strDiff, astrCompany, 'SAL_ORDERROW',
            anRowRecId,
            arSalOrderRow.ORDERID, arSalOrderRow.ORDER_ROWNUM,
            'Order Changes',
            NULL, 'EN',
            'MSSREC','OK',NULL,NULL,'NEVER',astrUser,7,anTextID, anOK);
        IF anOK > 0 THEN
            -- adding text link to SO row failed, do something...
        END IF;
    END IF;
-- tell the calling procedure if any differences were found
    abChanged := bChanged
END;

```

8.3 Tietokantaskriptit

Lean System sovelluskehitysympäristössä käytetään tietokannan muuttamiseksi tarvittava komentotiedostoja eli skriptejä. (Listaus 3) /17/

```

REM New Dep-parameter MANDATORYSCRAPTYPE;
REM Locos5.3.013 PDS CRI269 new dep-parameter MANDATORYSCRAPTYPE SEM 25.07.2001;

CREATE VIEW TMP_DEP_VIEW_KLUDGE AS
SELECT      COMPANY, ATTRIB6 DEPPARAM
FROM        DLG_DEPARTMENT
WHERE       RECTYPE = 0
AND         ATTRIB6 IS NOT NULL
GROUP BY   COMPANY, ATTRIB6;

INSERT INTO DLG_ENUMITEM (
    ENUMITEMID,
    VALUE10,
    NABBR,
    INFO,
    NAME,
    CREATE_STAMP,
    CREATED_BY,
    POS,
    RECID,
    COMPANY,
    ENUMID
)
SELECT
    'MANDATORYSCRAPTYPE',
    0,
    'Scrap type',
    'Scrap type',
    'Scrap type is mandatory information in the form',
    SYSDATE,
    'SYS',
    1,
    DLG_ENUMITEM_S.NEXTVAL,
    COMPANY,
    DEPPARAM
FROM      TMP_DEP_VIEW_KLUDGE;

DROP VIEW TMP_DEP_VIEW_KLUDGE;
INSERT INTO DLG_DBVERSION ( VERSIONID, INFO1, CREATED_BY, CREATE_STAMP )
VALUES ( 'L53_dep44.SQL', 'MANDATORYSCRAPTYPE Dep-param ', 'SEM', SYSDATE );
COMMIT;

```

Listaus 3: Lean System/Locos – Tietokantaskripti koodia

Skriptit ovat ohjelmia, jotka ajetaan jollakin työkalulla, jolla saa yhteyden ts. Oracle kantaan. Oraclen SQLPlus ja Servermanager (Oracle 7.3) ohjelmat käyvät Oraclen kantaan ajettaessa.

8.4 Perl

Perl on tulkettava ohjelmointikieli, joka muistuttaa C-kieltä on UNIX-järjestelmän työkaluja. Perl-ohjelmia tehdään tekstieditorilla, ja ohjelmia ajetaan perl-tulkilla, ohjelmia ei siis tarvitse kääntää. Perl-kieli on erittäin joustava ja sujuva ohjelmoitava. Perl kirjoitetaan yleisten C-standardien mukaista koodia. (Listaus 4). /17/

```

sub
ProcessTable
{
    chop($sql);
    $sql .= "\nWHERE\n\tCOMPANY = '$company'\n"
    if $company;
    printf "$sql\n";

    &OraDo($lda, $sql);

    if($OraErrno != 0) {
        printf "Oracle error: $OraError.\n";
        return 0;
    } else {
        printf "Konvertointi suoritettu taululle:
        $old_fields[0]\n\n";
    }
    &OraCommit($lda);
    return 1;
}

```

Listaus 4: Lean System/Locos – Perl –ohjelma koodia

9 Valmisohjelma vs. räätälöityohjelma ratkaisu

Case: Lean System / Locos

Valmiin ERP -järjestelmän ja räätälöidyn ohjelmiston välillä on suuriakin eroavaisuuksia toiminnallisuuksissa. Vaikkakin valmiiden ohjelmistojenkin toteutus vaatii toimittajalta työtä ohjelmiston asentamisessa jopa ylimääräistä räätälöintiä, jotta asiakkaan toimimaan toiminnallisuuteen päästään. Lomakkeet ja raportit ovat hyvin yleisiä räätälöintikohteita ERP -järjestelmässä.

Nykyisin räätälöity toiminnanohjausjärjestelmä pohjautuu yhä enemmän valmiiden ohjelmistokomponenttien käyttöön. Tuotteen omien komponenttien avulla voidaan rakentaa asiakkaan tarpeiden mukaan räätälöity ERP- järjestelmää. Tässä tapauksessa kaikkea ei kuitenkaan tarvitse rakentaa itse alusta asti, vaan perustana toimivat valmiit tuotteen ohjelmistokomponentit.

Valmiin ERP -järjestelmän ja räätälöidyn järjestelmän välinen valinta riippuu paljolti kustannuksista, projektin läpivientiajasta, koulutuksesta, räätälöitävyydestä asiakkaan tarpeet ottaen huomioon, omista asiakkaista ja ajankohdasta.

Kokonaan asiakkaan omista lähtökohdista lähtevä ERP -järjestelmä kannattaa valita silloin kun se liikkeenjohdollisesti on järkevää toteuttaa, odotettu räätälöinti ja sen tuomat uudet toiminnallisuudet palvelevat tehokkaasti omaa yritystä että asiakasta.

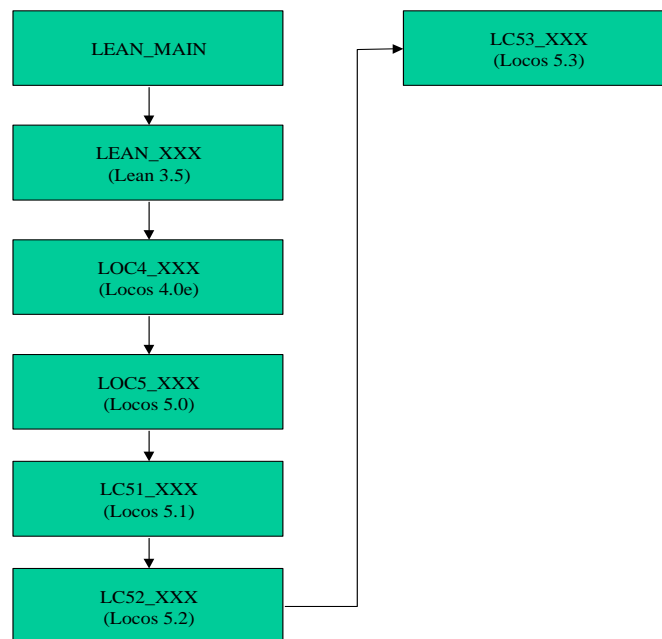
9.1 Toimitushierarkkia

Asiakaskohtaiset räätälöinnit tullee toteuttaa siten, etteivät ne näy tuotteessa tai muissa projekteissa, jotta räätälöinti ei näkyisi muille, se toteutetaan omiin tiedostoihinsa (taulukko 6). Asiakasprojekti Locos 4.0 on toteutettu Lean System 3.5 versiosta (kuvio 18).

Taulukko 6:

Locos sovellusten lähdekoodien paikat Team Object Manager :ssa (TOM) /20,s.5/

<i>Module</i>	<i>TOM module</i>	<i>Locos 5.3 source code</i>	<i>Locos 5.3 report templates</i>
<i>Batch Runs</i>	<i>LEAN_BAP</i>	<i>-none</i>	<i>-none</i>
<i>Item Data</i>	<i>LC53_BAS</i>	<i>\LC53</i>	<i>-none</i>
<i>Structure</i>	<i>LC53_CON</i>	<i>\LC53</i>	<i>.\reports\LC53</i>
<i>Deliveries</i>	<i>LC53_DEL</i>	<i>\LC53</i>	<i>.\reports\LC53</i>
<i>Resources</i>	<i>LC53_DLG</i>	<i>\LC53</i>	<i>-none</i>
<i>Invoicing</i>	<i>LC53_INV</i>	<i>\LC53</i>	<i>.\reports\LC53</i>
<i>Logi-Tool</i>	<i>LEAN_LOG</i>	<i>-none</i>	<i>-none</i>
<i>Main</i>	<i>LC53LEAN</i>	<i>\LC53</i>	<i>-none</i>
<i>MSS</i>	<i>LC53_MSS</i>	<i>-all</i>	<i>.\reports\LC53</i>
<i>Purchase</i>	<i>LC53_PUR</i>	<i>\LC53</i>	<i>.\reports\LC53</i>
<i>Purchase invoicing</i>	<i>LC53_PIN</i>	<i>\LC53</i>	<i>-none</i>
<i>Receiving</i>	<i>LC53_REC</i>	<i>\LC53</i>	<i>-none</i>
<i>Reporting</i>	<i>LC53_REP</i>	<i>\LC53</i>	<i>.\reports\LC53</i>
<i>Sales</i>	<i>LC53_SAL</i>	<i>\LC53</i>	<i>.\reports\LC53</i>
<i>Stock</i>	<i>LC53_STO</i>	<i>\LC53</i>	<i>-none</i>
<i>Administration</i>	<i>LEAN_SYS</i>	<i>None</i>	<i>-none</i>
<i>Data Transfers</i>	<i>LC53_TRA</i>	<i>\LC53</i>	<i>-none</i>



Kuvio 18: Locos 5.3 kustomointi /20, s.5/

9.2 Asiakkaan vastuut ohjelmistoprojektissa

Ohjelmisto projektin tilaus ja projektin toimitus asiakkaalle.

Asiakkaan vastuulla ovat esimerkiksi: /13/

- Uuteen ohjelmisto versioon kuuluvat kaikki muutospyynnöt (CR).
- Projektin johto.
- Koordinoida versionhallinta ja priodisoida muutospyynnöt.
- Organisoita ja johtaa projektikokoukset johtotasolla.
- Vahvistaa ja hyväksyä toiminnalliset määrittelyt (FD ja DD).
- Tehdä testiympäristöt.
- Tehdä koulutusmateriaali ja ohjeistus loppukäyttäjille.
- Kouluttaa tarvittava henkilöstö.
- Ohjelmistovirheet on kirjattu muutostyökalulle (CR-Tool) riittävällä tarkkuudella niin että virheet on mahdollista korjata.
- Tehdä tarvittavat testimateriaalit ja testisuunnitelmat.

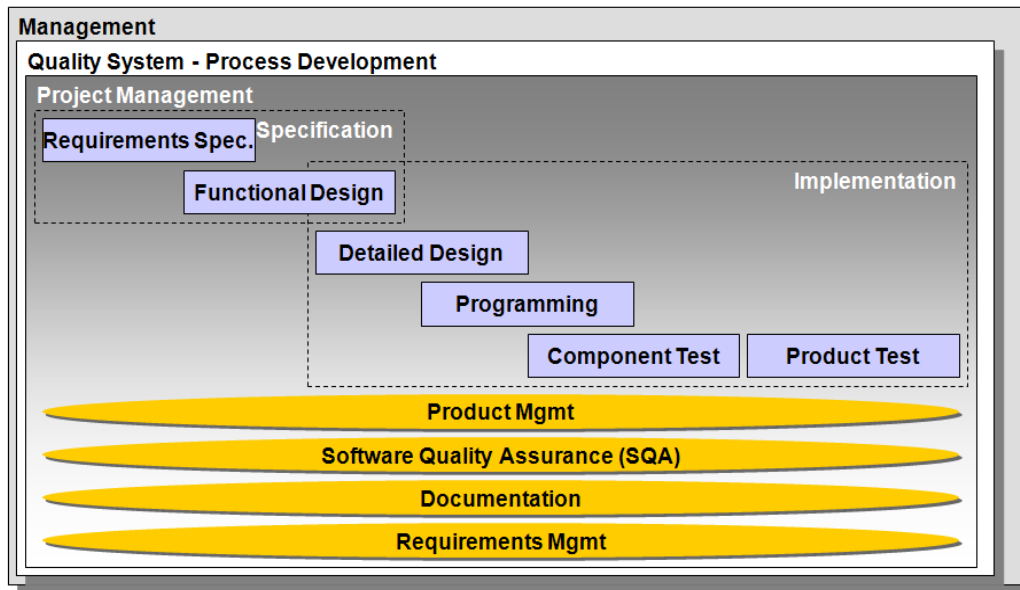
9.3 Toimittajan vastuut ohjelmistoprojektissa

Toimittajan vastuulla ovat esimerkiksi: /13/

- Projektisuunnitelman teko.
- Määrittelyt ja testausohjeitten teko.
 - Toiminnallinen määrittely (Functional Design - FD) .
 - Yksityiskohtainen määrittely (Detail Design – DD).
 - Ohjelmointi (Programming).
 - Tuotantotesti (Product Testing –PT).
 - Komponenttitesti (Component Testing – CT).
 - Moduulitesti (Module Testing – MT).
- Tuotantotesti ympäristön tekeminen.
- Toimituksen tekeminen (Roll-out package).
- Testiohjeitten luovutus ennen asiakkaan testauksien alkua.
- Johtaa ja valvoa projektia ja henkilöstöä.
- Vaihtaa projektin henkilöstöä.
- Vastata ja johtaa alihankkijoita jos projektissa käytetään niitä.

10 V Malli

V Malli (V-Modell) on otettu käyttöön monissa yrityksissä ohjelmistoprojektien suunnittelussa ja toteutuksessa. V-mallin avulla voidaan määrittellä projektin päämäärät ja tavat, joilla näihin tavoitteisiin päästään (kuviot 19 ja 20). Se määrittelee myös kaikkien ohjelmistoprojektiin osallistujien roolit: kuka tekee mitä ja milloin. /12/



Kuvio 19: Räätelöidyn ohjelmiston prosessivaiheet /18, s.1/

10.1 V-malli tavoitteet

- 1 Projektin riskien minimointi
 - V-malli parantaa projektinhallintaa määrittelemällä yhtenäiset käytännöt ja roolit, jotka edesauttavat projektin riskien tunnistamisen ja siten niiden ehkäisemisen.
- 2 Laadun parantaminen ja takaaminen
 - Standardina kehitysmallina V-malli takaa tulosten laadukkuuden. Laatu voidaan varmistaa jo projektin varhaisessa vaiheessa.
- 3 Kustannusten pienentäminen
 - Standardin kehitysmallin kulut on helppo laskea jo projektin aikana, jolloin niissä pysyminen on myös helpompaa.
- 4 Kommunikoinnin parantaminen tilaajan ja toimittajan välillä

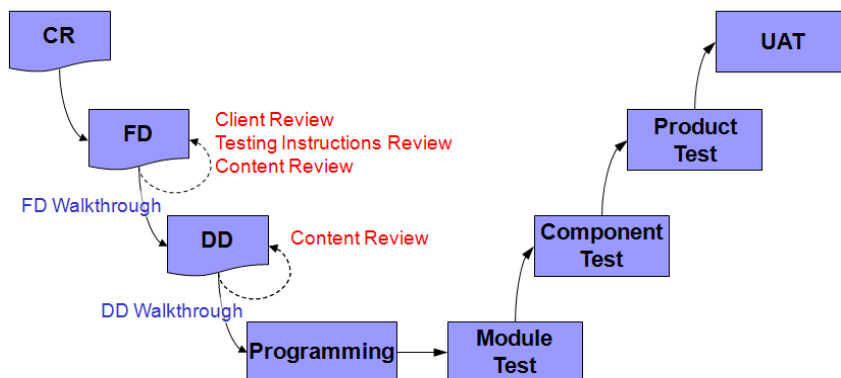
- V-mallissa on määritelty tarkkaan jokaisen rooli projektissa. Tämä tarkoittaa myös tilaajalta, joilta vaaditaan oman osansa projektissa. Heidän tulee olla yhteisymmärryksessä projektin päämääristä ja tavoitteista.

10.2 V-malli rakenne

- 1 V-malli määrittelee kuka tekee mitä ja milloin. V-mallia voidaan myös muokata projektin tarpeiden mukaisesti.
- 2 Testausvaiheessa verrataan koko ajan vastaako tuote samalla tasolla olevaan suunnitteluvaihetta.

10.3 Hyödyt ja rajoitukset

- 1 V-mallia voidaan muokata jokaista projektia ja organisaatiota varten juuri sellaiseksi kuin on tarpeellista.
- 2 V-mallin avulla on helppo määrittellä mitä kukin tekee ja missä vaiheessa ohjelmistoprojektia.
- 3 V-malli keskittyy ainoastaan projektiin, ei koko organisaatioon ja sitä käytetään vain kerran projektin aikana.
- 4 V-malli on joissain asioissa liian muokkautuva, jolloin mallin käyttäjillä jää liikaa liikkumavaraa, joka saattaa aiheuttaa sen, että projektin aikana ei tiedetä mitä asioita malli ottaa mukaan ja mitä jättää pois. /12/



Kuvio 20: V Mallin (V-Modell) mukaiset määrittelyt ja testivaiheet /18, s.2/

11 Case: Lean System / Locos CR1346

Esimerkkinä: V Mallin (V-Modell) mukainen muutospyyntö CR1346. (Liitteet 2-8)

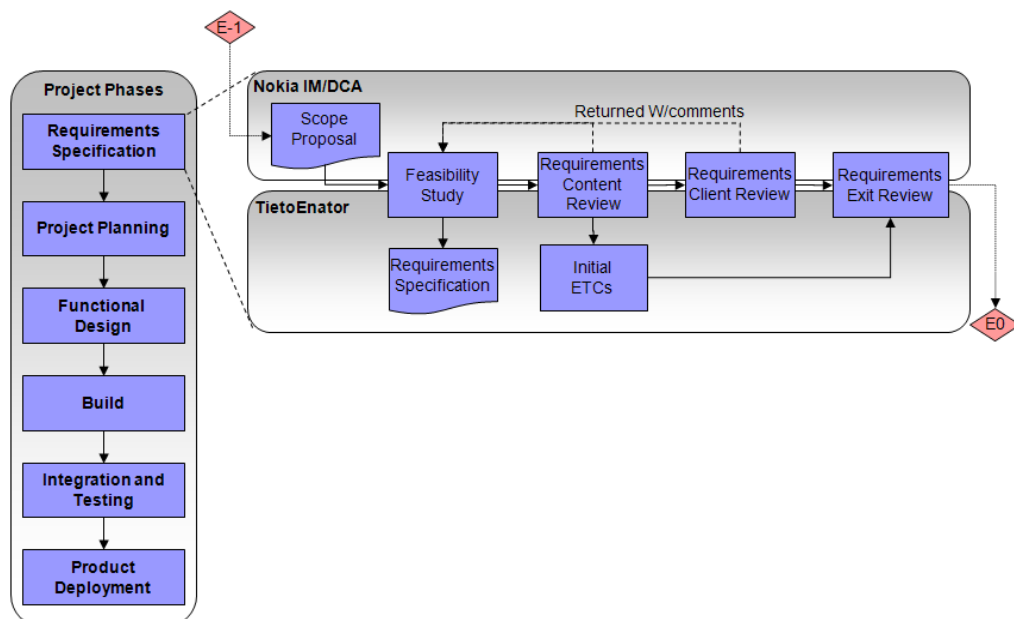
11.1 Liiketoiminnan vaatimukset

(Specification - Business Requirements – FD1346)

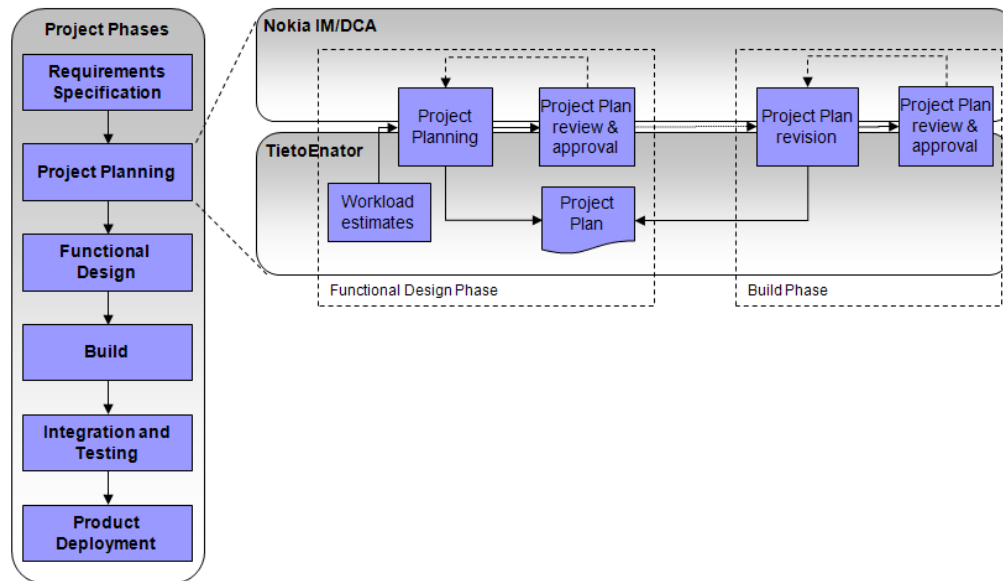
Kuvioissa 21 ja 22 nähdään ne projektin osa-alueet, joihin liiketoiminnan vaatimukset vaikuttavat.

Muutospyynnön FD1346 liiketoiminnan vaatimukset.

- Kun vastaanotto on tehty, järjestelmän täytyy ilmoittaa käyttäjälle normaalin varaston sijasta tavarankäytön puskurivarasto ns. buffer -varasto. Jos myyntitilaus on toimitettu puskurivarastosta, järjestelmän täytyy päivittää varasto/ varaston sijaintitiedot vastaavasti myynti- ja ostotilauksiin.
- Käyttäjät, jotka eivät tee tavarankäytön vastaanottoja eivät helposti erota toimintatapoja toisistaan. Koska suuri määrä tavarankäytön vastaanottoja tehdään on hyvin vaikeaa ja aikaa vievää jäljittää myyntitilauksia. Varastojen riskit virheiden osalta on oletettava.



Kuvio 21: Vaatimusmäärittelyt (Requirements Specification) /18, s.3/



Kuvio 22: Projektisuunnitelma (Project Planning) /18, s.4/

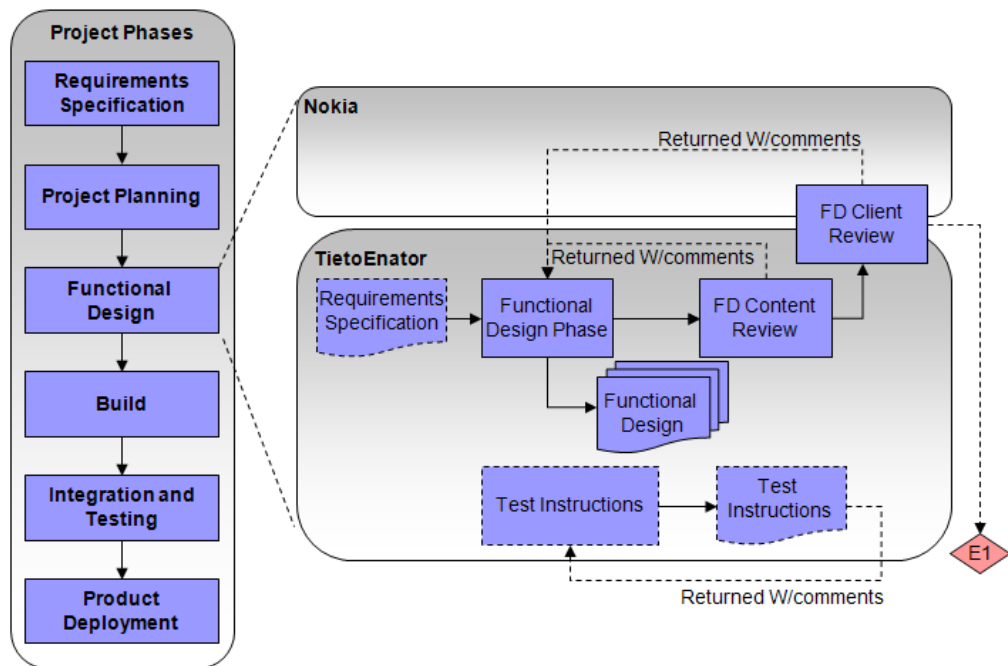
11.2 Järjestelmän vaatimukset

(Specification - System Requirements) (Kuvio 23)

Liitteessä 6 kuvataan myös muutospyynnön toiminnalliset vaatimukset ja testiohjeet.

(Liite 6)

- Järjestelmään täytyy luoda hälytys joka näytetään viestilaatikossa. Jos tavaran vastaanottava tilausrivin varasto /varaston sijainti on eri kuin tavaran toimitusrivi(t).
- Järjestelmän täytyy tarkistaa tai päivittää varasto/varastonsijainti tiedot, jotka liittyvät myynti- ja ostotilauksiin, kun toimitus on annettu.
- Nämä uudet toiminnot on parametrisoitava erikseen.

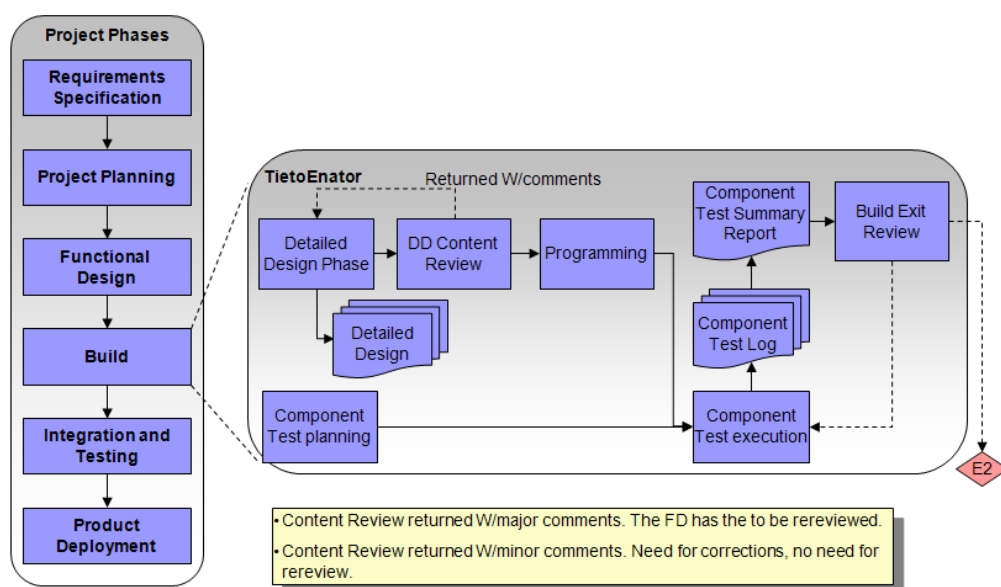


Kuvio 23: Toiminnallinen määrittely ja testiohjeet (Functional Design) /18, s.5/

11.3 Locos järjestelmän toteutus

(Implementation - Programming impact – DD1346)

Kuviossa 24 nähdään ne projektin osa-alueet, joihin toteutuksen vaatimukset vaikuttavat. Muutospyynnön Locos/DD1346 ohjelmoinnin vaatimukset. (Liite 3)



Kuvio 24: Toiminnallinen toteutus, ohjelmointi, testiohjeet ja testaus (Build) /18, s. 6/

- Hälytysilmoitus (varoitus) ehdottaa käyttäjälle vastaanotosta ja näyttää tämän käyttäjälle viestilaatikossa.
- Seuraavat muutokset ovat välttämättömiä, jotka koskevat toimituksen vastaanotto- ja vastaanottorivilomaketta.

Ohjelmoinnin toteutus kuvattu alla. (Listaus 5a, 5b, 5c, 5d) (Liite 3)

Following changes are needed to implement on the Receive and Receive Rows forms.

Following changes are needed to implement on SO and PO Rows.

- *New dep-parameter STOCKHANDLING is needed to control default value for Sales Order and Purchase Order Rows.*

If the dep-parameter STOCKHANDLING has Value11 = 1, system must update delivery Stock/Stock locations from which delivery items are delivered to the corresponding Sales Order and Purchase Order rows.

- *In On PAM_Reserved22 message actions of the Delivery form is called by a new function e.g. 'CustomUpdateStockDifferent'. This new function 'CustomUpdateStockDifferent' is called for each row returned from the select query. This function update delivery row Stock/Stock locations to the corresponding values in the Sales Order and Purchase Order Rows. Followings codes can be used for this:*

(new code added with bold, please don't copy / paste)

!

! Save RECIDs of HOLDed-rows for further processing.

!

If(strRowStatus != DELSTATUS_HOLD)

Else

Listaus 5a : Locos/CR1346 ohjelma koodia

```
If Not CustomUpdateStockDifferent (strTmpStockId, strTmpStockLoc, strOrderId,
nOrderRowNum, strRowItemId)
```

```
!
! Rollback & return
```

```
Return SalSendMsg (hWndForm, PAM_Reserved15, DISPLAY_ROWNUM, 0)
```

```
!
! All delivery rows handled succesfully
!
```

Function: CustomUpdateStockDifferent

Description:

Returns

Boolean: bOk

Parameters

String: strStockId

String: strStockLoc

String: strOrderId

String: nOrderRowNum

String: strRowItemId

Static Variables

Local variables

Actions

```
If Locos4NField ( strDEP, 'STOCKHANDLING', 11, 0 ) > 0
```

```
!
! Update StockId and StockLocation to Sales Orders / Purchase Or-
der Rows and batches
```

```
!
Set SqlBuf = '
```

```
UPDATE SAL_ORDERROW SOR
```

```
SET STOCKID = :strStockId,
```

```
STOCKLOC = :strStockLoc
```

```
WHERE COMPANY= :strCOMP
```

```
AND RECTYPE = 0
```

```
AND ORDERID=: strOrderId
```

```
AND ORDER_ROWNUM =: nOrderRowNum
```

```
AND ITEMID = :strRowItemId
```

```
AND STOCKID != :strStockId AND STOCKLOC !=
```

```
:strStockLoc
```

```
;
```

Listaus 5b: Locos/CR1346 ohjelma koodia

```

If Not SqlPrepareAndExecute ( hSql, SqlBuf )
    Return FALSE
!
!
Set SqlBuf = '
        UPDATE PUR_ORDERROW
        SET STOCKID = :strStockId,
            STOCKLOC = :strStockLoc
        WHERE COMPANY= :strCOMP
        AND RECTYPE = 0
        AND SAL_ORDERID=: strOrderId
        AND SAL_ROWNUM = :nOrderRowNum
        AND ITEMID = :strRowItemId
        AND STOCKID != :strStockId
        AND STOCKLOC != :strStockLoc
        '

If Not SqlPrepareAndExecute ( hSql, SqlBuf )
    Return FALSE
!
!
Set SqlBuf = '
        UPDATE SAL_ORDERBATCH
        SET STOCKID = :strStockId,
            STOCKLOC = strStockLoc
        WHERE COMPANY= :strCOMP
        AND RECTYPE = 0
        AND ORDERID = :strOrderId
        AND ORDER_ROWNUM = :nOrderRowNum
        AND ITEMID = :strRowItemId
        AND STOCKID != :strStockId
        AND STOCKLOC != :strStockLoc
        '

If Not SqlPrepareAndExecute ( hSql, SqlBuf )
    Return FALSE
!
!
Set SqlBuf = '
        UPDATE PUR_ORDERBATCH
        SET STOCKID = :strStockId,
            STOCKLOC = :strStockLoc
        WHERE COMPANY= :strCOMP
        AND RECTYPE = 0
        AND SAL_ORDERID= :strOrderId
        AND SAL_ROWNUM = :nOrderRowNum
        AND ITEMID = :strRowItemId
        AND STOCKID != :strStockId
        AND STOCKLOC != :strStockLoc
        '

If Not SqlPrepareAndExecute ( hSql, SqlBuf )
    Return FALSE
!
!
Return TRUE

```

Listaus 5c: Locos/CR1346 ohjelma koodia

New dep-parameter STOCKHANDLING is added.

<i>REF.</i>	<i>PARAM. GROUP</i>	<i>PARAM. ROW</i>	<i>FIELD</i>	<i>DESCRIPTION</i>	<i>SCRIPT NAME</i>
	<i>DEP%</i>	<i>STOCKHANDLING</i>	<i>Value10</i>	<i>1= the alarm is created and the message box is shown to users</i>	<i>L53_depxxx.sql</i>
	<i>DEP%</i>	<i>STOCKHANDLING</i>	<i>Value11</i>	<i>1= update stock/stock location into the related SORs and PORs</i>	

Listaus 5d: Locos/DD1346 parametrien lisäys (Liite 3)

11.4 Locos järjestelmän dokumentointi

Taulukoissa 9 ja 10 nähdään muutospynnön Locos/CR1346 vaikutukset ohjelmiin ja dokumentteihin. (Liite 3)

Taulukko 9: Muutospynnön Locos/CR1346 vaikutukset seuraaviin ohjelmiin

<i>REF.</i>	<i>MODULE</i>	<i>FILE</i>	<i>CUSTOMIZATION</i>
	<i>REC</i>	<i>..\Leanv3_5\rec\Lc53\custom.apl</i>	<i>Locos5.3 specific</i>
	<i>REC</i>	<i>..\Leanv3_5\rec\Lc53\Newrec.apl</i>	
	<i>DEL</i>	<i>..\Leanv3_5\del\Lc53\custom.apl</i>	<i>Locos5.3 specific</i>
	<i>DEL</i>	<i>..\Leanv3_5\del\Lc53\Delivery.apl</i>	<i>Locos5.3 specific</i>

Taulukko 10: Muutospynnön Locos/CR 1346 vaikutukset seuraaviin dokumentteihin

<i>REF.</i>	<i>AFFECTED DOCUMENTATION</i>	<i>FILE</i>
	<i>New dep-parameter STOCKHANDLING</i>	<i>Parameter.doc</i>
	<i>Lean Receive User Manual</i>	<i>Lean_rec.doc</i>
	<i>Lean Receive Help File</i>	<i>Lean_rec.hlp</i>
	<i>Lean Delivery User Manual</i>	<i>Lean_del.doc</i>
	<i>Lean Delivery Help File</i>	<i>Lean_del.hlp</i>

11.5 Locos järjestelmän komponenttitestaus

(Implementation - Component test)

Taulukoissa 7 ja 8 nähdään muutospynnön Locos/CR1346 komponenttitestin testiohjeet ja ohjaustietojen eli parametrien asetukset. (Liite3)

Taulukko 7: Parametrien asetukset (Liite 3)

#	PARAM. GROUP	PARAM. ROW	FIELD	VALUE
1	DEP	STOCKHANDLING	Value10	1
2	DEP	STOCKHANDLING	Value10	0
3	DEP	STOCKHANDLING	Value11	1
4	DEP	STOCKHANDLING	Value11	0

Taulukko 8: Komponenttitestin testiohjeet (Liite 3)

REF.	PARAM#	CASE	TEST STEP
2.1.1.	1	'Receive' -form	Go to Receive (Id = 272) form and create receiving from Purchase Order which has differences in stock/stock locations than compared to the delivery related to the same Sales Order.
		Alarm created and the message box will be shown to user	Check that the alarm(s) are created and the message box will be opened.
	2	'Receive' -form	Repeat step → Create Receiving from PO... Check that the alarm(s) are NOT created and the message box will NOT be shown.
2.1.2.	3	'Deliveries' form	Go to Deliveries (Id 291) form and change stock locations in delivery rows and select 'Deliver selected' command.
			Check that stock locations are changed in related Sales Order and Purchase Orders.
	4		Repeat step → Change stock locations in delivery rows... Check that stock locations are NOT changed in related Sales Order and Purchase Orders rows.
2.4.2		Documentation changes	Verify that all the documentation changes have been made.

Testiohjeita ja parametrien asetuksia tulee noudattaa kirjaimellisesti, jotta testaus onnistuisi odotetulla tavalla. Komponenttitestauksella etsitään mahdolliset toiminnalliset ja/tai ohjelmointivirheet.

12 Räätelöidyn ERP-järjestelmän kustannukset

Perusajatus räätelöidyllä ohjelmalla on, että lisätään liiketoiminnallista tehokkuutta ja tuottavuutta tekemällä asiat toisin ja automatisoimalla aikaisemmin manuaalisesti tehtyjä työvaiheita. Räätelöidyn ohjelman muutospyynnön hyödyt ovat kuitenkin monesti mutkikkaita, sekä vaikeasti hahmotettavissa ja laskettavissa. Tietotekniikkainvestoinnit on nähtävä koko yrityksen kannalta, joten hyötyjä pitäisi pystyä arvioimaan myös muilta osin kuin rahallisesti.

Tietotekniikkainvestointilajit voidaan jakaa karkeasti kahteen päälajiin operatiiviseen ja strategiseen. /5/

12.1 Operatiiviset tietotekniikkainvestoinnit

Operatiiviset investoinnit liittyvät välttämättömyysinvestointeihin, joita ovat esimerkiksi sidosryhmien painostuksesta johtuvat liiketoiminnan ratkaisut, laajennusinvestoinnit, rationalisointi eli kustannusten alentaminen, korvausinvestoinnit eli uudet investoinnit kuluneiden tai elinkaaren loppuvaiheessa olevien tuotteiden tai palveluiden tilalle.

12.2 Strategiset tietotekniikkainvestoinnit

Strategiset investoinnit taas voivat liittyä esimerkiksi markkina-aseman tai asiakassuhteiden turvaamiseen, kilpailuedun saavuttamiseen tai uusien liiketoimintamallien kehittämiseen.

12.3 Tietotekniikkainvestoinnin hyödyt

Tietotekniikkainvestointien hyödyt voidaan jakaa kahteen kategoriaan. Suoraan rahassa mitattavat, eli tangible benefits ja vaikeasti mitattavat hyödyt intangible benefits. Rahassa mitattavia hyötyjä ovat tuotot, kustannussäästöt, henkilöstösäästöt, huoltokustannusten väheneminen, pääoman kiertonopeuden kasvaminen ja markkinaosuuksien muutos. Vaikeasti mitattavia puolestaan ovat luotettavan tiedon saaminen, päätöksenteon nopeutuminen, parempi työmoraali, virheiden väheneminen, parempi palvelutaso ja joustavuuden lisääntyminen. Tietotekniikka saattaa muuttaa ansaintalogiikkaa ja luoda aivan uudenlaisia liiketoimintamalleja. /5/

12.4 Tietojärjestelmän kustannukset – Nykyarvomenetelmä

Kuinka kustannuksia voi sitten oikeasti mitata? Kustannuslaskenta riippuu luonnollisesti aivan tietotekniikkainvestoinnin laadusta. Jos investointi kohdistuu järjestelmään, jonka avulla voidaan automatisoida aikaisemmin manuaalisesti tehtyjä toimintoja, voidaan hyötyjä mitata helpommin, kuin esimerkiksi päätöksentekoa tukevan järjestelmän hyötyjä pohdittaessa.

Kun tietojärjestelmän räätälöintiä suunnitellaan, on hyvä mallintaa ne liiketoimintaprosessit, joita järjestelmässä tulisi muuttaa. Mallinnuksessa tulisi määrittää nykytila ja räätälöinnillä aikaansaama tavoitetila. Sen lisäksi, että nykytilan mallinnuksella nähdään mahdolliset pullonkaulat ja kehitystarpeiden kohteet, voidaan sen avulla havainnoida prosessin sitomat kustannukset. Kun prosessi mallinnetaan ja eri toimintojen roolit ja tekijät saadaan selville, voidaan haastattelututkimuksin selvittää kuhunkin vaiheeseen ja toimintoon kulunut aika. Ajanmäärittelyn avulla voidaan laskea toimintoon käytetyt henkilöstökustannukset./5/

Eräs tapa selvittää järjestelmän aiheutuneita kustannuksia on käyttäen nykyarvomenetelmää (Net Present Value, NPV). Kaikki investoinnista aiheutuvat tuotot ja kustannukset diskontataan valitulla laskentakorkokannalla nykyhetkeen. Kaikki tuotot ja kustannukset tulee siis muuttaa rahaksi, jotta projektin nykyarvo voidaan laskea (Kaavat 1 ja 2). /22, s.30-31/

tarkkailujakso nettonykyarvolla

$$NPV = \sum_{t=0}^n \frac{B_t}{(1+i)^t} - \sum_{t=0}^n \frac{C_t}{(1+i)^t} \quad (1)$$

jossa NPV on projektin nettonykyarvo ja

i = diskonttokorko, B_t = hyöty vuonna t, C_t = kustannukset vuonna t,

n = projektin kesto vuosina ja $\frac{1}{(1+i)^t}$ = diskonttotekijä, t = aika

tai vastaavasti

NB = net benefit = nettohyöty

$$NPV = \sum_{t=0}^n \frac{NB_t}{(1+i)^t} \quad (2)$$

$$\text{jossa } NB_t = B_t - C_t$$

Mikäli projektin nettonykyarvo eli nykyhetkeen diskontatut hyödyt miinus nykyhetkeen diskontatut kustannukset on positiivinen luku, investointihanke on kannattava.

Esitän seuraavassa kuvitteellisen investointilaskelman tilaajan näkökulmasta. Investointilaskelmaa varten ainakin viisi eri tekijää tulisi ottaa huomioon: investoinnin vaikutusaika, käytettävä korkokanta, hankintahinta, vuotuiset nettokassavirrat vaikutusajalle ja mahdollinen jäännösarvo. Laskennassa käytettävä laskentakorkokanta muodostuu yhdistelmästä, jossa on mukana inflaatio, rahoituksen hinta, investoinnin tuottovaatimus ja riskipreemio.

Esimerkkinä muutospyyntö Locos/CR1346 (Global Siso Buffer Stock Handling) toteutuneet kustannukset. Toimittajan puolelta toteutuneet kustannukset muodostuvat V-mallin mukaisesti eri projektivaiheista. Lisäksi tulevat muut kustannukset, jotka ovat projektin yhteisiä kustannuksia (Taulukko 11).

Taulukko 11. Toimittajan Locos/ CR1346 määrittelyjen, ohjelmoinnin ja testauksen kustannukset

<i>CR</i>	<i>Header</i>	<i>Phase</i>	<i>Work/day</i>	<i>Budget</i>	<i>Actual</i>
1346	<i>Global Siso buffer stock handling</i>	<i>FD</i>	<i>4</i>	<i>3668</i>	<i>2293</i>
1346		<i>DD</i>	<i>4</i>	<i>3668</i>	<i>2843</i>
1346		<i>PG</i>	<i>4</i>	<i>3668</i>	<i>3210</i>
1346		<i>CT</i>	<i>1</i>	<i>917</i>	<i>825</i>
1346		<i>TI</i>	<i>0,5</i>	<i>0</i>	<i>275</i>
		<i>Total</i>	<i>13,5</i>	<i>11921</i>	<i>9445</i>
	<i>Management</i>				
	<i>Misc-FD, Misc-DD, Misc-PG, Misc-CT</i>				
	<i>Meetings (project meeting(TE), project meeting(IM+TE)</i>				
	<i>Documents</i>				
	<i>Test planning, execution and environment</i>				
	<i>Training material and execution</i>				
	<i>Delivery/installation</i>				
	<i>Average price xxx EUR</i>				

Esimerkki: $C_t =$ kustannukset vuonna t, (Hankintahinta) = **30696 €**

- toimittajan laskuttama muutospyyntö 1.5 (kerroin) x 9445 = 23612 €

- tilaajalle (yritys) investoinnista aiheuttamat muut kulut + 30 % = 7084 € (esim., asennus-, testaus-, koulutus-, hallintokulut ovat kertaluonteisia)

B_t = hyöty vuonna t, 22500 x 5 v. (investoinnin vaikutusaika) = **112500 €**

Laskukaava: $(RG/(m*d*h^*)) * s * t$, jossa

- RG = toimituksen vastaanottokäsittelyjä kappaletta per vuosi
- m = kuukaudet (vuodessa)
- d = päivät (kuukaudessa)
- h = tunnit (järjestelmä käytettävissä)
- s = arvioitu käsittelyajan pituus tunteina (esim., arvio 0,10 tuntia, 6 min)
- t = työntekijä (ohjelmiston käyttäjä) lukumäärä
- nykytila,
ennen investointia $(10000/(12*30*24))*0,50 * 3 = 1,736$
- tavoitetila,
jälkeen investoinnin $(50000/(12*30*24))*0,10 * 3 = 1,736$
- tavaranvastaanotto - käsittelyajansäästö 80 % (30 min → 6 min)
- tästä 80 % ajansäästöstä vaikutus palkkakustannuksiin noin 25 %
- vastaanottokäsittelyn lisäys 5 kertainen (10000 kpl → 50000 kpl)
- ts. arvioidut työntekijöiden (3) palkkasäästöt yhteensä 22500 €/v
(3 * 30000 = 90000 ./. 25 % = 22500 €/v työntekijöiden ajansäästö järjestelmän vastaanottokäsittelyssä)

Edellä määritellyt investointilaskelmassa käytettävät luvut on koottu seuraavaan laatiikkoon:

Hankintameno	30696 €
Nettotuotot 1.vuosi	22500€
2-5	90000€
Investointiajanjakso	5 vuotta
Jäännösarvo	0
Laskentakorkokanta	30 %

Esimerkki voidaan laskea sijoittamalla luvut kaavaan 2,). /22, s.30-31/

$$-30696 + \frac{22500}{(1+0,30)^1} + \frac{22500}{(1+0,30)^2} + \frac{22500}{(1+0,30)^3} + \frac{22500}{(1+0,30)^4} + \frac{22500}{(1+0,30)^5} = 24104 \text{ €}$$

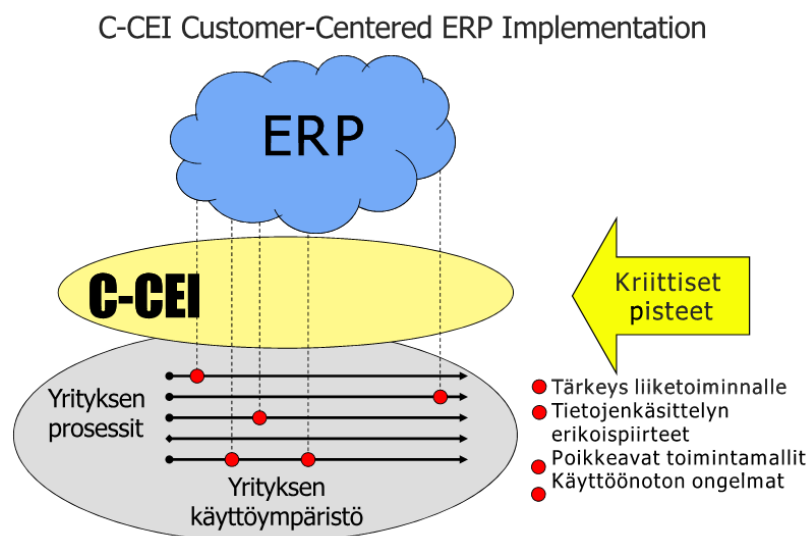
Laskutoimituksen jälkeen NVP on 24104 euroa, joten investointihanke tilaajalle (muutospyyntö Locos/CR1346) on kannattava jopa 30 prosentin laskentakorolla (investoinnin tuottovaatimuksella). Kun räätälöidyssä ohjelmaprojektissa saattaa olla useita kymmeniä jopa satoja muutospyyntöjä valmiiseen ohjelmatuotteeseen nähden, niin voidaan päätellä, että räätälöity ohjelma tulee huomattavan kalliiksi. Toisaalta, jos asetettuun investoinnin tuottovaatimukseen päästään investoinnin ajanjaksolla ja säästöjä syntyy työntekijöiden palkassa että ajankäytössä sekä tuotannon tehokkuus kasvaa, niin investointihanke on kannattava.

12.5 C-CEI menetelmä

Systemaattinen menetelmä (Customer-Centred ERP Implementation, C-CEI) on kehitetty toimintatutkimukseen suuntaan (kuvio 25). C-CEI menetelmä pyrkii kehittämään ERP kolmella analyysillä:

1. Toimintoanalyysi.
2. Toimintaympäristöanalyysi.
3. Riskianalyysi.

Toimintoanalyysillä tuotetaan ERP järjestelmän vaatimuksia, ja riskianalyysi ottaa huomioon yritykselle ERP toteutuksesta tulevat riskit. Toimintaympäristöanalyysillä pyritään kokonaisvaltaisempaan ja monialaisen ERP-prosessin hallintaan. Toimintaympäristöanalyysin tulokset paljastavat tarvittavat muutokset organisaatiossa ja työntekijöiden prosessissa. Lisäksi toimintaympäristöanalyysi on toimittajasta riippumaton, siten se tarkentaa asiakkaan tarpeet ilman erityisiä COTS –tyyppisen ERP-järjestelmän rajoituksia. /10/



Kuvio 25: C-CEI menetelmä /10, s.15/

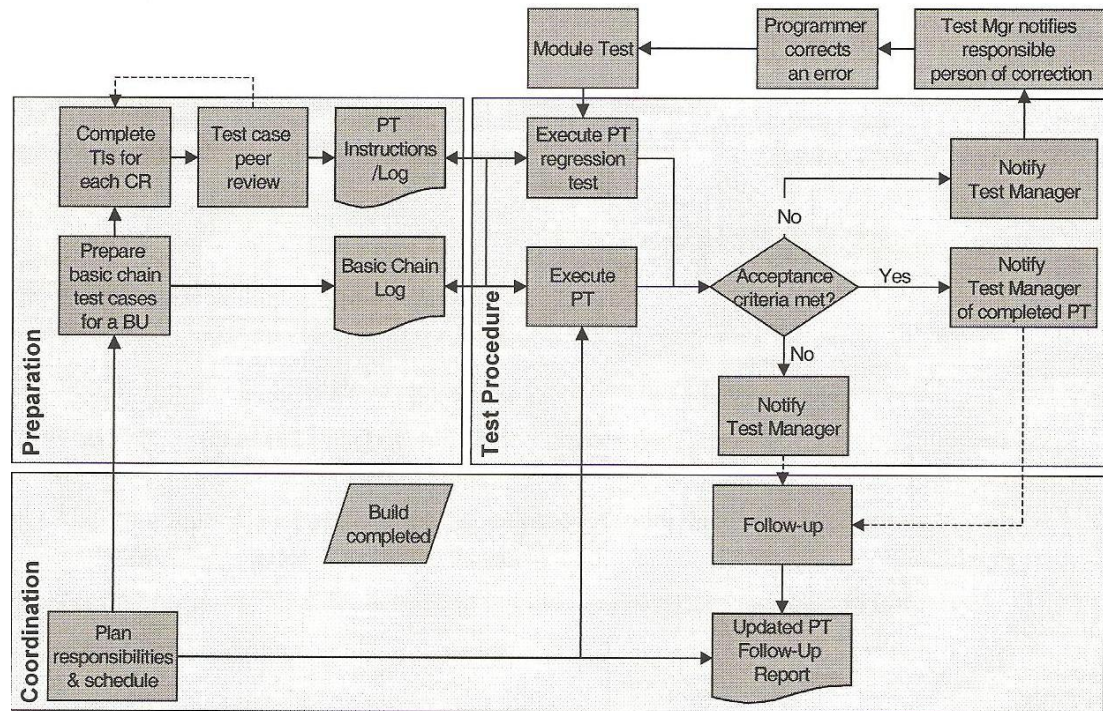
On olemassa maailmanlaajuisia ERP-järjestelmän hankintaan erikoistuneita yrityksiä, jotka auttavat eri toimialojen yrityksiä toiminnanohjausjärjestelmän hankinnassa. /4/

Toiminnanohjausjärjestelmän tutkimus C-CEI menetelmällä. Liite 9.

C-CEI menetelmän riskianalyysi. Liite 10.

13 Tuotantotesti

Tuotantotesti vaiheessa (Product Test - PT) testataan koko toiminnallinen ketju kattaen sekä uudet ja olemassa olevat toiminnallisuudet. Testi tehdään FD määrittelyä vasten ja ohjelma versiossa joka on läpäissyt komponenttitestin. (Kuvio 26)



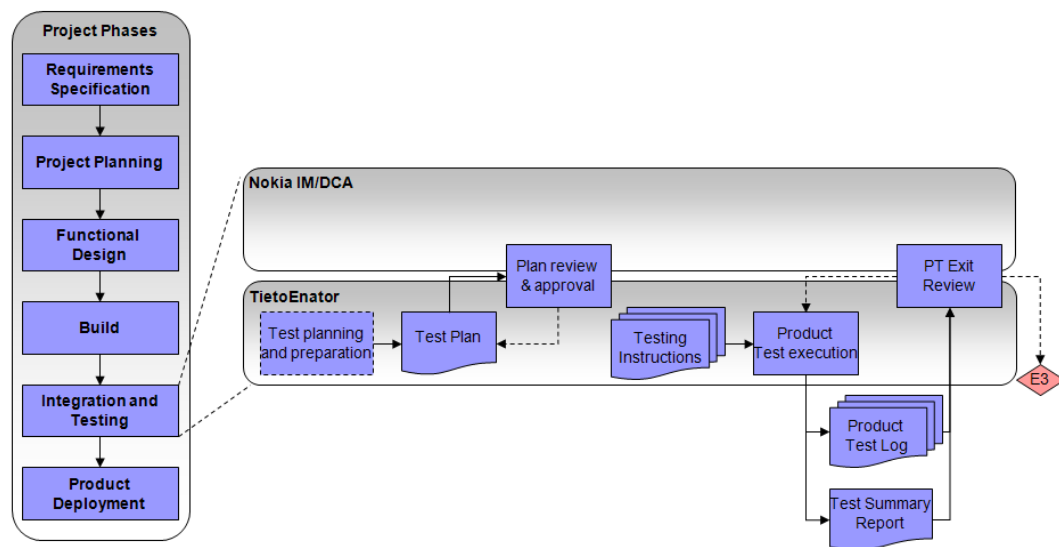
Kuvio 26: Tuotantotestin prosessikaavio /13, s.13/

Tuotantotestivaihe sisältää tuotanto-, moduuli-, ja regressiotestit. Jos virheitä löytyy tuotannon testauksen aikana havaitut virheet kirjataan testipöytäkirjaan ja testaus aloitetaan uudelleen moduuli- ja regressiotesteillä. Tuotantotestiä suoritettaessa tulee mukaan kaikki ohjelman uudet ja vanhat muutospyyntöt, jotka kuuluvat toimitettavaan ohjelma versioon. Ohjelman täytyy myös olla ennen testin alkua ns. ”frozen” eli jäädytetyssä lähtökohta tilanteessa. (Kuvio 27)

Testaajat varmistavat hyvissä ajoin ennen testauksen alkua ohjelman ympäristön toiminnallisuus asettamalla oikeat parametrit ja tarvittavat ohjaustiedot paikalleen.

Tuotantotestin vastuu aikataulusta ja johtamisesta kuuluu testauksen johtajalle (Test Manager). Projektijohtajan ja testijohtajan yhteisillä päätöksillä jaetaan testattavat tapaukset testihenkilöille niin, että testaukseen ei liitetä ulkopuolisia testaukseen kuulumat-

tomia resursseja. Testijohtajan vastuu on valvoa tuotantotestin prosessia, testin laatua ja testiajia. /13/



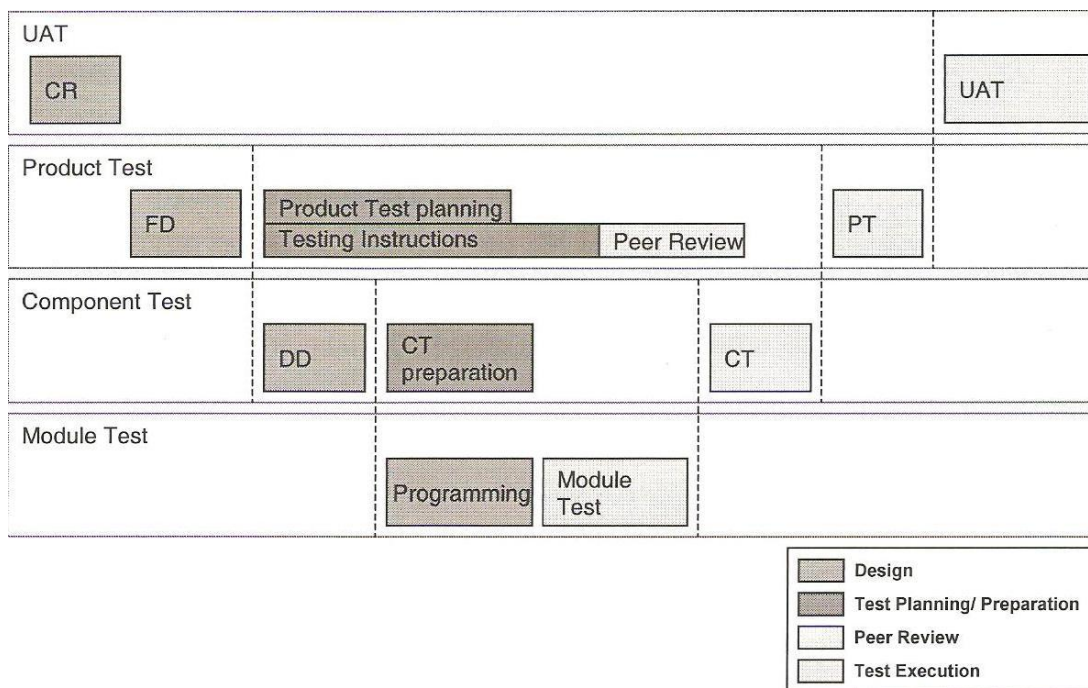
Kuvio 27:

Tuotantotestivaihe - testisuunnitelma, testaus ja raportointi (Product Testing) /18, s.7/

13.1 Testaukseen valmistautuminen

Suunnittelijat, jotka määrittelevät uuden toiminnallisuuden, valmistavat testiohjeet etukäteen. Testijohtaja katselmoi ryhmänsä kanssa tuotantotesti ohjeet ja määrää samalla testihenkilön suorittamaan tuotantotestiä. Testaaja valmistautuu testitapauksiin seuraavalla tavalla (kuvio 28): /13/

- Tutustumalla FD:n ja tuotantotesti ohjeisiin.
- Tutustumalla ohjelman toiminnallisuuden ketjuun ja niihin ohjelma kohtiin joilla FD:llä on vaikutusta.
- Täydentää tuotannon testiohjeita, jos on tarvetta.
- Päivittää tai lisää tarvittavat parametri(t) ja ohjaustiedot ennen testin aloittamista.
- Jos virheitä löytyy, merkitä havaitut virheet testipöytäkirjaan ja ilmoittaa testijohtajalle havaitut virheet.



Kuvio 28: V Mallin (V Modell) mukaiset määrittelyt ja testivaiheet /13, s. 6/

13.2 Komponentti- ja moduulitesti

Komponenttitestit (Component Test – CT) vaiheessa testataan yksi ohjelman toiminnallisuus. Testi tehdään DD määrittelyä vasten ja ohjelma versiossa joka on läpäissyt moduulitestin (MT).

Moduulitesti (Module Test – MT) vaiheessa ohjelmoija testaa oman ohjelman. Kohteena löytää mahdolliset ohjelmointi virheet.

13.3 Virheiden luokittelu

Virheet voidaan luokitella neljään luokkaan riippuen vakavuusasteesta ja haitasta: /13/

- 1 Show stopper.
- 2 Has to be fixed before release.
- 3 Has to be fixed.
- 4 Nice to have.

”Show stopper” ovat vakavia virheitä. Ne estävät testauksen jatkumisen.

”Has to be fixed before release” virheet ovat myös vakavia virheitä, mutta ne eivät estä testauksen jatkumista. Nämä virheet tulee korjata ennen seuraavaa testivaihetta.

”Has to be fixed” virheet eivät ole vakavia virheitä. Virheet tulee korjata ennen ohjelma version toimitusta.

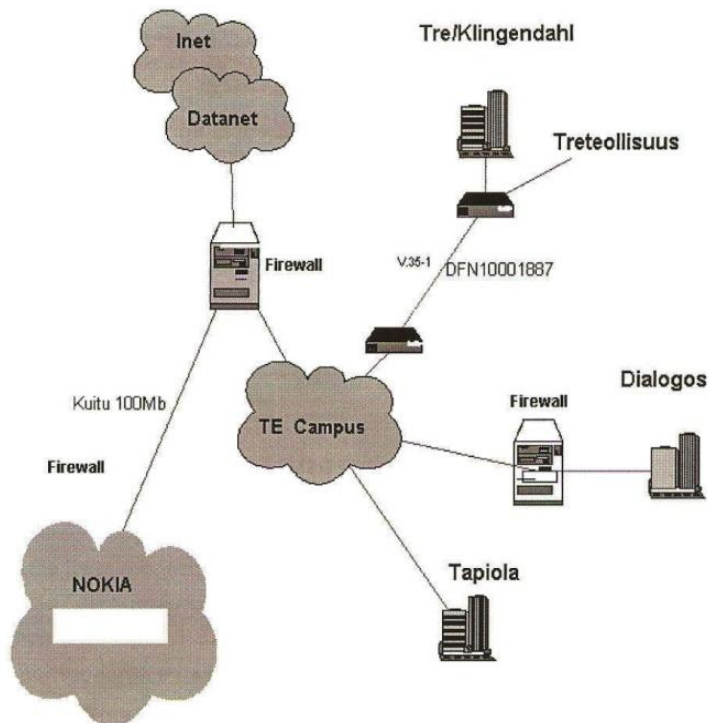
”Nice to have” virheillä ei ole vaikutusta tuotanto käytössä olevaan ohjelmaan.

Kaikki virheet tulee merkitä testi logeihin. Jos virhettä ei pystytä korjaamaan toimitettuun ohjelma versioon tulee se merkitä ”Feed Back Tool” rekisteriin.

13.4 Testauksen ympäristö

Testaus suoritetaan yleensä toimittajan tiloissa ja asiakkaan saatavalla olevalla testikannalla. Testikanta pyritään saamaan mahdollisimman hyvin vastaamaan tuotantokantaa. Tällöin paras tapa on ottaa tuotantokannasta kopio ja jättää pois testikannasta kaikki tilaajan liikesalassa pidettävä tieto esimerkiksi hintatiedot.

Kuviossa 29 on kuvattu toimittajan ja asiakkaan tietoliikenne yhteydet testauksen aikana.



Kuvio 29: Toimittaja/Asiakas - Tietoliikenne yhteydet testauksessa /13, s.16/

Tuotanto-, (PT) ja tilaajan hyväksymä (UAT) testaukset suoritetaan tilaajan tiloissa ja tilaajan tekemällä testikannalla. Tuotantotestaus suoritetaan toiminnallisen vaatimusmäärittelyn mukaisesti ja tilaajan muutospyyntöä vasten tarkistaen onko haluttu toiminto tullut tehtyä. /13/

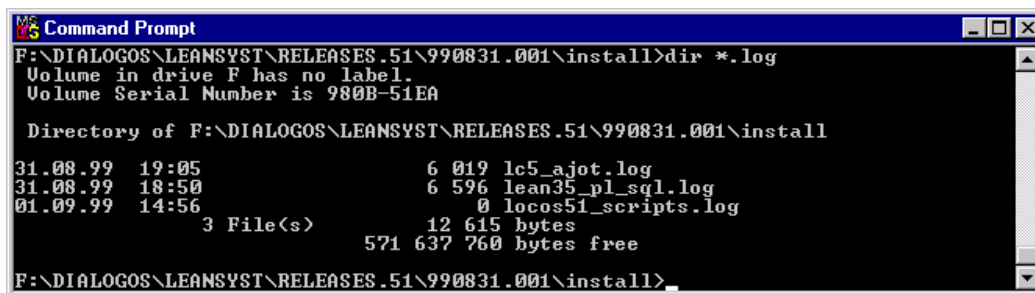
13.5 Ohjelmiston toimitus ja asennus

Locos 5.3 ohjelmiston toimitus asiakkaalle sisältää useita erilaisia dokumentteja ja asiakirjoja (Taulukko 12). Ohjelmiston toimitus tapahtuu yleensä fyysisesti sähköisesti tai muulla sovitulla tavalla.

Taulukko 12: Locos dokumenttien pääaiheet

<i>Functional designs</i>	<i>Clocal Clearance documents</i>
<i>Functional designs review documents</i>	<i>R/3 interface documents</i>
<i>Logical model</i>	<i>Product Test Logs</i>
<i>Technical Description of Database (also General Features if changes apply)</i>	<i>Product Test summary report</i>
<i>System architecture (if changes apply)</i>	<i>Parameter documents</i>
<i>Detailed Designs</i>	<i>Technical Release Note</i>
<i>Detailed Designs review documents</i>	<i>Installation Instructions for Server and Client</i>
<i>Testing Instructions for Product Test</i>	<i>Periodic project reports</i>
<i>Component Test summary report</i>	<i>Release Training material for key users and UAT</i>
<i>User Manuals:</i>	<i>Installation instructions for NT, both for upgrade from 5.2 to 5.3 and for a new installation</i>
<i>Print documents</i>	<i>Database scripts from release xxx to release xxx</i>
<i>Intrastat documents</i>	<i>Executables</i>

Ohjelmiston asennuksen aikana luodaan kolme eri ns. logi -tiedostoa (kuvio 30) , joihin tallentuu asennuksen ajonaikainen data.



```

Command Prompt
F:\DIALOGOS\LEANSYST\RELEASES.51\990831.001\install>dir *.log
Volume in drive F has no label.
Volume Serial Number is 980B-51EA

Directory of F:\DIALOGOS\LEANSYST\RELEASES.51\990831.001\install

31.08.99  19:05                6 019 lc5_ajot.log
31.08.99  18:50                6 596 lean35_pl_sql.log
01.09.99  14:56                0 locos51_scripts.log
          3 File(s)              12 615 bytes
          571 637 760 bytes free

F:\DIALOGOS\LEANSYST\RELEASES.51\990831.001\install>

```

Kuvio 30: Locos ohjelmiston asennuksen logi -tiedostot /13, s.16/

14 Tietokanta

Lean System / Locos järjestelmässä käytetään Oracle-tietokantaa ja siihen kuuluvia komponenttia.

Oracle voidaan käsittää kahtena eri asiana. Se on relaatiotietokanta, mutta samalla myös tietokannanhallintajärjestelmä. Tietokannanhallintajärjestelmän (Database Management System, DBMS) on pystyttävä hallitsemaan suurta määrää informaatiota usean käyttäjän järjestelmässä. Käyttäjien on vielä päästävä samanaikaisesti käsiksi samaan dataan. Lisäksi järjestelmän vasteaika informaation hakemiseen, lisäämiseen, muokkaamiseen ja poistamiseen ei saa olla liian pitkä. DBMS-järjestelmän tulee olla tietoturvallinen siinä mielessä, että jokainen kantaan kirjautuva käyttäjä todennetaan esimerkiksi käyttäjätunnuksen ja salasanan avulla. Näin estetään järjestelmän luvaton käyttö.

Oracle-tietokanta koostuu fyysisistä ja loogisista tasoista. Fyysiset rakenteet ovat nähtävissä käyttöjärjestelmässä tiedostoina, ja niitä voidaan myös hallita tätä kautta. Loogiset rakenteet näkyvät vain Oraclen omilla hallintaohjelmistoilla ja käyttöjärjestelmä ei ole näistä tietoinen. /3, s.20/

14.1 Fyysinen taso

Tietokannan fyysinen taso koostuu tiedostoista Oracle-tietokantapalvelimella. Tiedostoja ovat mm. tietokanta-tiedostot (database file, dbf), joihin varsinkin tietokannan sisältämä informaatio on tallennettu ja ohjaustiedostoja (control file, ctl). /3 s.21/

14.1.1 Ohjaustiedosto

Ohjaustiedosto on erittäin kriittinen komponentti tietokannan toiminnan kannalta, koska se sisältää tietoa tietokannan fyysisestä rakenteesta. Tiedosto kuuluu mm. seuraavia asioita:

- Arkistolokihistoria.

- Informaatiota taulualueista sekä tietokantatiedostoista.
- Tieto käytössä olevasta redo-lokista.
- Tietokannan nimi.

Ohjaustiedoston kriittisyyden vuoksi tulee siitä olla aina useampi kopio. Esimerkiksi voitaisiin pitää kolmea identtistä ohjaustiedostoa, joista vain yhtä kanta käytäisi ja loput kaksi olisivat varalla. Ohjaustiedostojen sijainti määritetään init-tiedostossa, josta edempänä lisää. Kannan varmistuksen yhteydessä on myös hyvä ottaa ohjaustiedostosta kopio trace-tiedostoon käskyllä "ALTER DATABASE BACKUP CONTROLFILE TO TRACE;" /3, s.21/

14.1.2 Online redo –lokitt

Oracle kirjoittaa muutostietoa online redo -tiedostoihin, kun kannan sisältö muuttuu. Tiedostoja on yleensä kaksi tai useampia. Kanta kirjoittaa muutostietoa vuorotellen eri tiedostoihin. Lokitietoja voidaan käyttää hyväksi, jos kanta joudutaan palauttamaan. /3, s.21/

14.1.3 Archive log –tiedostot

Oraclessa on mahdollista arkistoida redo-lokeja sen sijaan, että lokin kirjoittaja ylikirjoittaisi muutostiedot seuraavalla kiertokerralla. Tässä kohtaa sanotaan kannan olevan arkistointimoodissa (archive log mode). Oraclen taustaprosessi ARC0 (numero voi olla muukin kuin nolla) suorittaa redo-lokien arkistointia. Näin ollen kanta voidaan korruptiotilanteessa palauttaa kauemmas ajassa. /3, s.21/

14.1.4 Tietokantatiedostot

Tietokantatiedostot sisältävät varsinaisen kannan informaatioisällön. Ne ovat fyysisiä tiedostoja käyttöjärjestelmässä, jotka voidaan nähdä, mutta niiden sisältöä ei varsinaisesti pääse näkemään. Tietokantatiedostot voidaan jakaa kolmeen osaan. Segmentti (segment) sisältää tietynlaisia tietokantaobjekteja. Esimerkiksi tietokantataulut kuuluvat taulusegmenttiin ja indeksit indeksisegmenttiin. Extentit koostuvat useasta datalohkosta (data block). Datalohko taas on pienin yksikkö tietokantatiedostossa. /3, s.23/

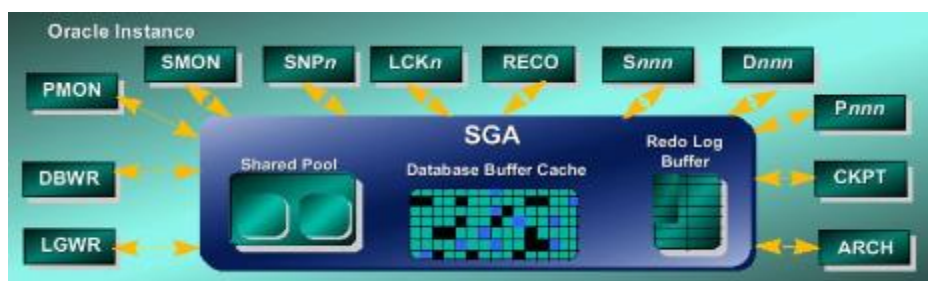
14.2 Looginen taso

Oraclen looginen taso koostuu taulualueista (tablespace). Niitä voi olla kannassa yksi tai useampi. Taulualue puolestaan koostuu yhdestä tai useammasta tietokantatiedostosta. Taulualueita on kolmea eri tyyppiä. Pysyvä taulualue (permanent), kumoustaulualue (undo) sekä väliaikainen taulualue (temporary). Pysyvä taulualue nimensä mukaisesti säilyttää tiedon pysyvästi. Kumoustaulualueita tarvitaan, kun halutaan kumota kantaan tehdyt muutokset ROLLBACK-komennolla. Kumoustaulualueita voidaan käyttää myös palautuksessa. Väliaikainen taulualue on nimensä mukaisesti tarkoitettu väliaikaisen tiedon tallentamiseen. Jos esimerkiksi SQL-lauseessa halutaan tehdä lajittelu, niin tämä suoritettaisiin väliaikaisessa taulualueessa.

Toinen loogisen tason elementti on skeema (schema), joita on yleensä kannassa useampia. Näihin kuuluvat mm. taulut, indeksit, herättimet sekä proseduurit. Toisin sanoen siis skeema on kokoelma erilaisia tietokantaobjekteja. Taulualueet ja skeemat eivät ole yhteydessä toisiinsa, vaan yhden skeeman objekteja voi olla tallennettu useaan eri taulualueeseen. /3, s.24/

14.3 Instanssi

Oracle vaatii toimiakseen taustaprosesseja, jotta käyttäjät pääsevät käsiksi tietokantaan. Nämä prosessit varaavat resursseja tietokantapalvelimen keskusmuistista. Taustaprosesseja ja varattua keskusmuistia yhdessä kutsutaan Oracle-instanssiksi (kuvio 31). /3, s.24/



Kuvio 31: Oracle-instanssi /3, s.25/

14.4 Muistirakenne

Oracle-instanssin muistirakenne rakentuu karkeasti kolmesta osa-alueesta. SGA (System Global Area), PGA (Program Global Area) sekä UGA (User Global Area). Nämä sijaitsevat tietokantapalvelimen keskusmuistissa. /3, s.26/

14.5 Tietokantataulut

Lean Systemin tietokantana on, että kaikki järjestelmän eri versiot käyttävät samanlaista (identtistä) tietokantaa. Jos kantaan tarvitsee tallettaa asiakaskohtaista tai toimialakohtaista tietoa, niin tiedot pyritään tallettamaan olemassa olevien taulujen `USR_`-kenttiin tai erillisiin tauluihin. Lisäksi on myös olemassa asiakaskohtaisia tauluja. . /3, s.35/

14.6 Palautus

Tietokannan korruptoitua voidaan palautus siis suorittaa joko fyysisestä tai loogisesta varmistuksesta. Loogisesta varmistuksesta palautus vaatii tietokannan uudelleen luonnin, ja kantaa ei voida palauttaa kuin varmistushetkeen kuten aikaisemmin on mainittu. Palautus on parasta näin ollen tehdä fyysisestä kuuma-varmistuksesta. Palautus voidaan tehdä tiettyyn aikaan sekunnin tarkkuudella tai vaihtoehtoisesti palauttaa jonkin arkistoidun redo-lokin perusteella. Myös ei-arkistoidun redo-lokin kautta palautus onnistuu. RMAN-apuohjelmaa käyttäen palautusrutiini on hyvin samantyyppinen kuin, jos se tehtäisiin komentokielellä. /3, s.35/

14.7 Ohjaustiedot

Locos toiminnallisuutta ohjataan ohjaustietojen eli parametrien avulla. Seuraava taulu näyttää esimerkkinä Locos järjestelmän varastonkäsittelyn riippuvuuden `STOCKHANDLING` -parametrilla. Taulukko 12.

Taulukko 12. Ohjaustiedot eli parametrien käyttö /Liite 3/

<i>New dep-parameter STOCKHANDLING (CRI346)</i>		
<i>DELIVERYCHECK</i>	<i>Value10</i>	<i>When defined, reservation tools in SO rows, Structure rows and Delivery rows forms are disabled. New tools Delivery Check and Release reservations and cancel deliveries are used instead in delivery module.</i>
<i>STOCKHANDLING</i>	<i>Value10</i>	<i>When defined (Value10 =1), the alarm is created and the message box is shown to user.</i>
<i>STOCKHANDLING</i>	<i>Value11</i>	<i>When defined (Value11 =1), update StockId and Stock locations into the related SO and PO Rows.</i>

15 ERP – toimittajat Suomessa

Alla viitteellinen taulukko (Taulukko 13) ERP-järjestelmä toimittajista Suomessa.

Taulukko 13. ERP-järjestelmä toimittajat Suomessa

<i>Nimi</i>	<i>Tuote</i>	<i>Kotisivu</i>
<i>Tieto Oyj</i>	<i>Lean System</i>	<i>http://tieto.fi/</i>
<i>Logica Finland</i>	<i>V8, V10</i>	<i>http://www.logica.fi/</i>
<i>Capgemini Finland</i>	<i>SAP, mySAP, Oracle</i>	<i>http://www.fi.capgemini.com/</i>
<i>System Finland</i>	<i>Dynamics NAV, AX</i>	<i>http://www.mitecgroup.se/fi/fi/Etusivu/</i>
<i>IFS</i>	<i>IFS</i>	<i>http://www.ifsworld.com/fi-FI</i>
<i>Oracle</i>	<i>Oracle</i>	<i>http://www.oracle.com/global/fi/index.html</i>
<i>Microsoft</i>	<i>Dynamics ERP</i>	<i>http://www.microsoft.com/dynamics/fi/fi/default.aspx</i>
<i>WiseTime Oy</i>	<i>Wise ERP</i>	<i>http://www.wisetime.fi/jarjestema.html</i>
<i>IBS</i>	<i>IBS</i>	<i>http://www.ibs.net/en/industries/</i>
<i>Visma Software</i>	<i>VismaL7, Visma Nova</i>	<i>http://www.visma.fi/</i>
<i>SysOpen</i>	<i>MS Dynamics AX</i>	<i>http://www.digia.com/</i>
<i>Mech-Soft Oy</i>	<i>MatFox</i>	<i>http://www.mech-soft.fi/</i>
<i>SAS</i>	<i>SAS</i>	<i>http://www.sas.com/offices/europe/finland/</i>

ERP toimittajat maailmanlaajuisesti.[www-sivu].[viitattu 3.11.2010].Saatavissa:
http://erp.softwareresearchtools.com/mrp/ERP_MRP_SoftwareVendors.asp

16 Yhteenveto

Nykyään on saatavilla yritykselle monia ERP-toiminnanohjausjärjestelmiä. Ongelma tulee siinä, ettei kukaan osapuolista, ei liiketoiminta- eikä toimittajapuoli, ota kokonaisvastuuta järjestelmien laadusta ja sen määrittelystä. Monet asiat tukevat yrityksen hankittavaa ERP- järjestelmää, esimerkiksi järjestelmän helppokäyttöisyys, järjestelmän nopeus, toimitettu ERP –järjestelmä, joka tukee ns. ”out-of-the-box ” ajatusmallia, tuki kolmannen osapuolen ratkaisulle, johdon raportoinnin ja tulosteiden reaaliaikainen saatavuus ja näiden helppokäyttöisyys.

Yrityksen liiketoimintaprosesseja ohjaava ja tukeva ERP-järjestelmä luo paremmat edellytykset selviytyä alati kovenevassa globaalissa kilpailussa. Tietoteknisesti ERP-järjestelmissä on suuriakin eroja. Ongelmat tulevat esille järjestelmää käytettäessä, jolloin joudutaan miettimään asioita uudella tavalla. Yleisin syy järjestelmää hankittaessa on se, että järjestelmä on suuresti ylimitoitettu yrityksen tarpeisiin nähden tai toimintamalli ei vastaa yrityksen liiketoimintoja.

Esimerkiksi toimialoista valmistavan teollisuuden ja kaupan toiminnanohjausjärjestelmät eroavat toimintatavoiltaan. Molemmilla toimialoilla on kuitenkin samat liiketoiminnalliset tavoitteet kuten yrityksen liiketoiminnan tehostaminen, kustannusten karsiminen, palvelella asiakkaita, tehdä tulosta, luoda uusia asiakassuhteita, tukea yritysjohtoa ja helpottaa asiointien käsittelyä ns. lattiatasolla.

Kunnallisella ja julkisella sektorilla ERP- järjestelmät tekevät tuloaan. Tampereen kaupungilla on SAP- toiminnanohjausjärjestelmä ollut käytössä muutamia vuosia. Järjestelmä ei kaikilta osiltaan ole vastannut kunnallishallinnon vaatimuksia ja järjestelmä onkin saanut kovaa kritiikkiä järjestelmän käyttäjien ja asiakkaiden taholta. Kun Tampereen sähkölaitos otti SAP järjestelmän käyttöön. ERP järjestelmän taloushallinnon sovelluksissa havaittiin huomattavia puutteita, jotka aiheuttivat huomattavat lisäkustannukset sen käyttöönotolle. Tampereen kaupunki on nyt ulkoistanut tietotekniikkapalvelunsa samoin laitteistonsa viisivuotiskaudeksi.

ERP- järjestelmän kustomoinnista ja räätälöinnistä on moni yritys luopunut, koska nämä toimenpiteet ovat liian kalliita toteuttaa ja hidastavat uuden ERP- järjestelmän käyt-

töönottoa. Ostettu järjestelmä halutaan hyvin nopealla aikataululla tuotantokäyttöön, jossa vain loppukäyttäjät koulutetaan.

Investointipäätöksen tukena yrityksen johdolla ei ole kuin toimittajan ”sana” tai luottamus omaan henkilöstöön, jotta järjestelmän hankinta olisi kannattava ja tuloksellinen. On olemassa erilaisia malleja kuten C-CEI –tutkimusmenetelmä, herkkyysanalyysi, kustannus-hyöty-analyysi, Monte Carlo herkkyysanalyysi jne. joiden käyttö yrityksen päätöksenteon apuvälineenä on epävarmaa ja kuvatut tilanteet saattavat sopia heikosti kaavailun toiminnanohjausjärjestelmän toimintoihin.

Olen käsitellyt tässä opinnäytetyössä Lean System järjestelmää toimittajan näkökulmasta, mikä lähtökohta ei ole paras mahdollinen tehtäessä ERP- järjestelmä valintaa yritykselle.

Paras lähtökohta olisi uutta järjestelmää hankittaessa saada toimittajalta jo toimitettujen ERP- järjestelmien referenssilista. Olla yhteydessä yrityksiin, joissa ERP- järjestelmää käytetään ja myös testata ennen lopullista päätöksen tekoa järjestelmää omassa liiketoimintaympäristössä. Yrityksen järjestelmän pääkäyttäjä(t) tulisi ainakin olla mukana kaikissa määrittely-, testaus- ja koulutusvaiheissa. Tällä tavoin saadaan tulevan järjestelmän investoinnille mahdollisimman hyvä kustannus-hyöty suhde, järjestelmän valinta ja käyttöönotto helpottuvat, käytön riskit minimoidaan ja saadaan yrityksen koko henkilöstön tuki uudelle järjestelmälle.

Olen myös käsitellyt räätälöityä Locos toiminnanohjausjärjestelmää V-mallin (V-Modell) avulla ja sen tärkeimmät tavat, joilla ohjelmistoprojektin päämääriin ja tavoitteisiin päästään.

Toivon, että tekemäni opinnäytetyöni on antanut tietämystä valmiin ja räätälöidyn toiminnanohjausjärjestelmän eroavaisuuksista. Lean System ohjelmistojärjestelmän perusrutiineista, joiden avulla ylläpidetään myynti- ja ostotilaukset, toimitukset, varaston-, tuotannon- ja valmistuksenhallintaa.

17 Lähdeluettelo

Painetut lähteet

- 1 *From, Maarit. ERP-toiminnanohjausjärjestelmän käyttö metropolialueen pk-yritysten liiketoiminnan ja työskentelyn tukena. Dokumentti .[pdf-tiedosto]. Opinnäytetyö 2009, Tietojenkäsittelyn Koulutusohjelma, Laurea-ammattikorkeakoulu, Leppävaara. Espoo.*
- 2 *Granlund, Markus – Malmi, Teemu. Tietotekniikan mahdollisuudet taloushallinnon kehittämisessä. ISBN 951-0-27703-7. Jyväskylä: WSOY 2003.*
- 3 *Huotari, Pasi. Oracle-pohjainen tietojärjestelmä globaalissa ympäristössä. Dokumentti .[pdf-tiedosto]. Opinnäytetyö 2008. Tietotekniikka. Tietoliikennetekniikka. Tampereen ammattikorkeakoulu. Tampere.*

Sähköiset lähteet

- 4 *ERP – Toiminnanohjausjärjestelmän hankinta.[www-sivu].[viitattu 3.10.2010].Saataavissa: <http://www.technologyevaluation.com/evaluation-centers/>*
- 5 *Kaskela, Lauri.Tietotekniikka investointina. TIEKE Tietoyhteiskunnan kehittämiskeskus ry. [www-sivu].[viitattu 5.11.2010] Saataavissa: http://www.tieke.fi/verkkokaveri/teemat/tietotekniikkahankinnat/tietotekniikan_hankinta/hankintaprosessi/2_tietotekniikka_investointina/*
- 6 *Kettunen, Jari – Simons, Magnus – Mikkola, Markku. Toiminnanohjausjärjestelmän käyttöönotto pk-yrityksessä. Teknologialähtöisestä ajattelusta kohti tiedon ja osaamisen hallintaa. [www-sivu].[viitattu 3.11.2010] Saataavissa: <http://www.vtt.fi/inf/pdf/julkaisut/2001/J854.pdf>*
- 7 *Lahti, Jarmo.IT-Viikko 2.12.2008.[www-sivu].[viitattu 3.11.2010].Saataavissa: <http://www.itviikko.fi/ratkaisut/2008/12/02/toiminnanohjausjarjestelmien-hyotyjen-mittaaminen-olematonta/200831134/7>*
- 8 *Siltala, Tiina. IT-Viikko 1.2009. [www-sivu].[viitattu 22.8.2010] Saataavissa: <http://www.tietoviikko.fi/taustat/article208530.ece>*
- 9 *SQLWindows – DBpedia . About:SQLWindows.[www-sivu].[viitattu 3.11.2010] Saataavissa: http://dbpedia.org/page/Gupta_Technologies*
- 10 *Tampereen teknillinen yliopisto.Teollisuustalous.Ihmiskeskeinen teknologia. C-CEI Customer-Centered ERP Implementation. ERP. C-CEI. Kriittiset pisteet.[pdf-tiedosto].[www-sivu].[viitattu 5.10.2010]. Saataavissa: http://akseli.tekes.fi/opencms/opencms/OhjelmaPortaali/ohjelmat/SISU/fi/Dokumenttiarkisto/Viestinta_ja_aktivointi/Seminaarit/Loppuseminaari/Kouri.pdf*
- 11 *Tieto – Lean System toiminnanohjaus. Ketterämpi ERP.Lean System Go Lean.[www-sivu].[viitattu 3.11.2010]. Saataavissa: <http://www.leansystem.fi/>*

- 12 *V-Modell* [www-sivu]. [viitattu 3.11.2010] Saatavissa:
<http://v-modell.iabg.de/v-modell-xt-html-english/693fba5c2bd73.html#toc4>

Muut lähteet

- 13 *Huhdanperä, Leena – Sundell, Kati. TestPlan. Dokumentti. [doc-tiedosto]. 8.2.2001. TietoEnator Oyj, Process&Manufacturing. Espoo.*
- 14 *Huttunen, Kare. Luentomoniste. Lean System käyttöohje 5.3. Lean System. Dokumentti. [doc-tiedosto]. TAMK Tampereen ammattikorkeakoulu, Kone- ja tuotantotekniikka. Tampere.*
- 15 *Huttunen, Kare. Luentomoniste. Lean_toimintoketju. Dokumentti. [paperiversio - skannattu]. KKH 5.10.2009. TAMK Tampereen ammattikorkeakoulu, Kone- ja tuotantotekniikka. Tampere.*
- 16 *Huttunen, Kare. Luentomoniste. Toiminnanohjaus. Toiminnanohjauksen tietojärjestelmät ja projekti. Dokumentti. [ppt-tiedosto]. KKH 22.11.2005. TAMK Tampereen ammattikorkeakoulu, Kone- ja tuotantotekniikka. Tampere.*
- 17 *Lean toteuttajat. Dialogos-Team. Lean Standardit. Dokumentti. [doc-tiedosto]. TietoEnator Oyj, Process&Manufacturing. Espoo.*
- 18 *Nurminen, Mikko. Locos PDS Development Process. Process flowcharts. Dokumentti. [ppt-tiedosto]. 30.8.2000. TietoEnator Oyj, Process&Manufacturing. Espoo.*
- 19 *Pennanen, Heidi. Logical Model. Dokumentti. [doc-tiedosto]. 14.3.2000. TietoEnator Oyj, Process&Manufacturing. Espoo.*
- 20 *Saarinen, Risto – Intala, Petri. Configuration Management Plan. PP_App4_ConfManPlan. Dokumentti. [doc-tiedosto]. 2001 TietoEnator Oyj, Process&Manufacturing. Espoo.*
- 21 *TietoEnator Oyj. Lean System yleiskuvaus. Dokumentti. [pdf-tiedosto]. Yleisesite. 14.3.2000. TietoEnator Oyj, Process&Manufacturing. Espoo.*
- 22 *YE11. Kustannus-hyötyanalyysi. Kevät 2007. Luku 5 Diskonttaaminen. Helsingin yliopisto. Maatalous ja metsätieteellinen tiedekunta. Helsinki. Dokumentti. [doc-tiedosto]. [www-sivu]. [viitattu 30.11.2010]. Saatavissa:
http://www.mm.helsinki.fi/mmtal/ye/YE11/YE11lu5_diskonttaaminen.doc*

18 Liitteet

- 1 *Lean System – Käyttöohje versio 5.3*
- 2 *Locos – Component Test Log CR1346ctlog (Luottamuksellinen)*
- 3 *Locos – Detailed Design DD1346 (Luottamuksellinen)*
- 4 *Locos – FD Acceptance Document FD1346 Acceptance (Luottamuksellinen)*
- 5 *Locos – FD Content Review Checklist FD1346 Review (Luottamuksellinen)*
- 6 *Locos – Functional Design FD1346 (Luottamuksellinen)*
- 7 *Locos – Module Test Log CR1346mtlog (Luottamuksellinen)*
- 8 *Locos – Product Test Instructions/Log TIFDI1346 (Luottamuksellinen)*
- 9 *Toiminnanohjausjärjestelmän tutkimus C-CEI menetelmällä*
- 10 *C-CEI menetelmän riskianalyysi*

LIITE 3

CR1346 GLOBAL SISO BUFFER STOCK HANDLING

Owner: Zhou Xingming / BNT
Designer: Seppo Maljanen
Project: Locos 5.3.014 PDS
Location: <\\akseli\customer\Nokia\Locos5.3\Pds\Designs\DetDes\DD1346.doc>

DATE	VERSION	PERSON	COMMENT	STATUS
14.09.01	0.1	Seppo Maljanen	For comments	1 st Draft
20.09.01	1.0	Seppo Maljanen		TE Accepted

Revision history

1 Introduction

1.1 General

This detailed design document is based on the corresponding function design FD1346 “Global SiSo buffer stock handling”.

The business reasons for implementing this CR are described in closer detail in the chapter 1.1.1. of the functional design

1.2 Current Functionality

See FD1346 chapter 1.1.1.

1.3 To-Be Functionality

See FD1346 chapter 2.1.1/ 2.1.2

1.4 Limitations

N/A

2 Description

2.1 Application Modifications

2.1.1.1. Alarm to suggest user to receive goods to another stock and the message box must also be shown for the user.

Following changes are needed to implement on the Receive and Receive Rows forms.

- New dep-parameter STOCKHANDLING is needed to control default value for the Receive form.

Alarm and message box are created/shown to users if the dep-parameter STOCKHANDLING has Value10 = 1, when receive rows are created from PO and which had be found differences in stock/stock locations to the delivery related to the same SO.

(new code added with bold, please don't copy / paste)

! Create the Receive Header –

!! Create a RecId

!

Set nReceiveRecId = GenerateNewRecId ('PUR_DELIVERY', HDR_Receive, FALSE)

!

Set bStockDifferent = FALSE

Set nStockHandling = Locos4(strDEP, 'STOCKHANDLING', 0)

!

If nReceiveRecId < 0

Return DO_RETURN_FALSE

- When all batched selected in the function 'CreateFromOrder' to be copied the receive rows from PUR_ORDERBATCH. To this select –clause should get Stock-Id and StockLoc values. For this must be added two local variables e.g. String: strBatchStockId and String: strBatchStockLoc .

Following select-clause is used when the receiving is created from all the PO batch.

(new code added with bold, please don't copy / paste)

!!!

Set SqlBuf =

```

,
SELECT      POB.RECID,
            POB.ITEMID,
            POB.COMPID,
            POB.ORDER_ROWNUM,
            ....
            POB.STOCKID,
            POB.STOCKLOC
FROM        PUR_ORDERBATCH POB, PUR_ORDERROW POR
INTO        :nBatchRecId,
            :strBatchItemId,
            :strBatchSuppId,
            :nOrderRowNum,
            ....
            :strBatchStockId,
            :strBatchStockLoc
WHERE       POB.COMPANY          = :strCOMP
AND         POB.RECTYPE          = 0
AND         POR.COMPANY          = :strCOMP
AND         POR.RECTYPE          = 0
AND         POB.ORDERID          = :strOrderId
AND         POR.ORDERID          = :strOrderId
AND         POB.ORDER_ROWNUM = POR.ORDER_ROWNUM
,

```

- New function 'CustomStockDifferent' is called for each row returned from the select query. This function compares the receive row stockId/stockLoc to the corresponding values in the deliver row. If they are different the alarm is created to the receiving row. Following codes can be used for this:

(new code added with bold, please don't copy / paste)

Function: CustomStockDifferent

Description: Stock/Stock locations handling. System will be created alarm(s) if receiving row's stock/stock location differ from Sales Order Rows.

Returns

Boolean: bOk

Parameters

String: strBatchStockId

String: strBatchStockLoc

String: strBatchItemId

String: strOrderId

Number: nOrderRowNum

Number: nRowRecId

String: strReceiveId

Number: nReceiveRecId

Number: nNextRowNum

Receive String: strDeliveryId

Receive Boolean: bStockDiffers

String: strUser

Static Variables

Local variables

String: strAlarmText

Boolean: bStockDifferent

String: strStockId

String: strStockLoc

Number: nAlumRecid

Actions

!

Set SqlBuf = '

```

SELECT    SR.DELIVERYID,
          SR.STOCKID,

```

```

SR.STOCKLOC
INTO      :strDeliveryId, :strStockId, :strStockLoc
FROM      SAL_DELIVERYROW SR, PUR_ORDERROW PR
WHERE      SR.COMPANY = :strCOMP
AND       SR.RECTYPE = 0
AND       SR.ITEMID = :strBatchItemId
AND       SR.ORDERID = PR.SAL_ORDERID
AND       SR.ORDER_ROWNUM = PR.SAL_ROWNUM
AND       PR.COMPANY = :strCOMP
AND       PR.RECTYPE = 0
AND       PR.ORDERID = :strOrderId
AND       PR.ORDER_ROWNUM = :nOrderRowNum '

If SqlPrepareAndExecute (hSql3, SqlBuf)
    While SqlFetchNext (hSql3, nFetch)
        If ((strStockId != strBatchStockId) OR (strStockLoc != strBatchStockLoc))
            Set bStockDiffers =TRUE
            Set strAlarmText = strAlarmText || 'Different Stock and Stock location is given for item ' || strBatchItemId || ' in delivery ' || strDeliveryId || ' - ' || ' Check Stocks! '
            Call GenerateAlarm (strDEP, 'DATA', 'AUTO', 'Receive', strAlarmText, strUser, nRowRecId, 'PUR_DELIVERYROW', strReceiveId, SalNumberToStrX (nNextRowNum,0), 0, nAlumRecid )

Return TRUE
!

!   Insert the row
If Not SqlExecute( hSqlTmp1)
    ! Roll back the inserted receive header and inserted rows
Call SqlRollback( hSql )
Return DO_RETURN_FALSE
!
If nStockHandling > 0
    If (strBatchStockId = STRING_Null AND strBatchStockLoc = STRING_Null)

```



```

    Call CustomStockDifferent (strSelItemStockId, strSelItemStockLoc, strBatchItemId, strOrderId, nOrderRowNum, nRowRecId,
    strNewDeliveryId, nReceiveRecId, nNextRowNum, strStockDeliveryId, bStockDifferent, strUser)

```

```

Else

```

```

    Call CustomStockDifferent (strBatchStockId, strBatchStockLoc, strBatchItemId, strOrderId, nOrderRowNum, nRowRecId,
    strNewDeliveryId, nReceiveRecId, nNextRowNum, strStockDeliveryId, bStockDifferent, strUser)

```

```

!

```

```

Set nNextRowNum = nNextRowNum + 1

```

- The other checking must be done when the new receiving is created from a selected number of batches. The new 'CustomStockDifferent' function is called for each selected frmSelectedBatches.twSelectBatches row. Following code can be used for this:

(new code added with bold, please don't copy / paste)

```

!

```

```

! Insert the row

```

```

If Not SqlExecute( hSqlTmp1)

```

```

    ! Roll back the inserted receive header and inserted rows

```

```

    Call SqlRollback( hSql )

```

```

    Return DO_RETURN_FALSE

```

```

!

```

```

If nStockHandling > 0

```

```

    If (twSelectBatches.strStockId != STRING_Null AND twSelectBatches.strStockLoc != STRING_Null)

```

```

        Call CustomStockDifferent (twSelectBatches.strStockId, twSelectBatches.strStockLoc, twSelectBatches.strItemId, twSelectBatches.strOrderId,
twSelectBatches.nOrderRowNum, nRowRecId, strNewDeliveryId, nReceiveRecId, nNextRowNum, strStockDeliveryId, bStockDifferent, strUser)

```

```

    Else

```

```

        Call CustomStockDifferent (strSelItemStockId, strSelItemStockLoc, strBatchItemId, twSelectBatches.strOrderId, twSelect-
Batches.nOrderRowNum, nRowRecId, strNewDeliveryId, nReceiveRecId, nNextRowNum, strStockDeliveryId, bStockDifferent, strUser)

```

```

!

```

```

! Increment row counter.

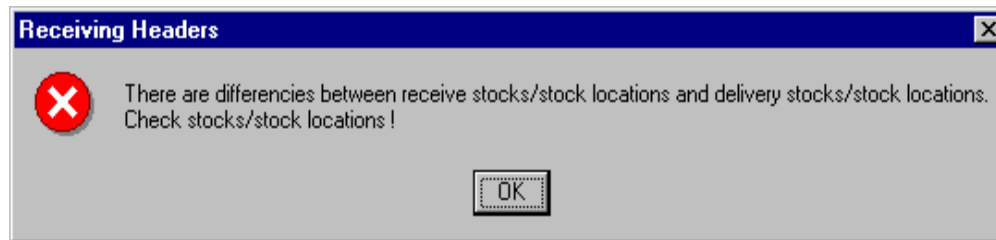
```

```

Set nNextRowNum = nNextRowNum + 1

```

- SalMessageBox is called by the function 'CreateFromOrder' when all the receive rows have been handled, the following message is shown, if there were difference in the StockIds/Stock locations in same receive row.



If the Stocks/Stock locations are same this message is not showed.

A new constant string variable e.g. string: ERR_StockDifferent has be added to the file: rec_eng.apl.

New alarm must be added to the receive header if they were Stock/Stock Locations differences in same receive row.

(new code added with bold, please don't copy / paste)

!

If bStockDifferent = TRUE And nStockHandling > 0

Call SalMessageBox(ERR_StockDifferent, HDR_Receive, MB_IconHand)

Set strAlarmText = strAlarmText || 'Different Stock and Stock location is given for item ' || strBatchItemId || ' in delivery ' || strStockDeliveryId || '- ' || ' Check Stocks !'

Call GenerateAlarm (strDEP, 'DATA', 'AUTO', 'Receive', strAlarmText, strUser, nReceiveRecId, 'PUR_DELIVERY', strNewDeliveryId, STRING_Null, 0, nAlumRecid)

!

Set nTempRow = SalTblInsertRow (twReceive, TBL_MaxRow)

!

! If insert succeded fill the row.

2.1.1.2. Update stocks/stock locations into related SO and PO.

Following changes are needed to implement on SO and PO Rows.

- New dep-parameter STOCKHANDLING is needed to control default value for Sales Order and Purchase Order Rows.

If the dep-parameter STOCKHANDLING has Value1 1 = 1, system must update delivery Stock/Stock locations from which delivery items are delivered to the corresponding Sales Order and Purchase Order rows.

- In On PAM_Reserved22 message actions of the Delivery form is called by a new function e.g. 'CustomUpdateStockDifferent' . This new function 'CustomUpdateStockDifferent' is called for each row returned from the select query. This function update delivery row StockI/Stock locations to the corresponding values in the Sales Order and Purchase Order Rows. Followings codes can be used for this:

(new code added with bold, please don't copy / paste)

!

! Save RECIDs of HOLDed-rows for further processing.

!

If(strRowStatus != DELSTATUS_HOLD)

Else

If Not CustomUpdateStockDifferent (strTmpStockId, strTmpStockLoc,strOrderId, nOrderRowNum,strRowItemId)

!

! Rollback & return

Return SalSendMsg (hWndForm, PAM_Reserved15, DISPLAY_ROWNUM,0)

!

! All delivery rows handled succesfully

!

Function: CustomUpdateStockDifferent

Description:

Returns

Boolean: bOk

Parameters

String: strStockId

String: strStockLoc

String: strOrderId
String: nOrderRowNum
String: strRowItemId

Static Variables

Local variables

Actions

If Locos4NField (strDEP, 'STOCKHANDLING',11, 0) > 0

```

!
! Update StockId and StockLocation to Sales Orders / Purchase Order Rows and batches
!
Set SqlBuf = '
    UPDATE SAL_ORDERROW SOR
    SET STOCKID          = :strStockId,
        STOCKLOC = :strStockLoc
    WHERE COMPANY= :strCOMP
    AND RECTYPE = 0
    AND ORDERID          =: strOrderId
    AND ORDER_ROWNUM =: nOrderRowNum
    AND ITEMID = :strRowItemId
    AND STOCKID != :strStockId AND STOCKLOC != :strStockLoc
    ,
If Not SqlPrepareAndExecute ( hSql, SqlBuf )
    Return FALSE
!
!
Set SqlBuf = '
    UPDATE PUR_ORDERROW
    SET STOCKID          = :strStockId,
        STOCKLOC = :strStockLoc

```

```

WHERE COMPANY= :strCOMP
AND RECTYPE = 0
AND SAL_ORDERID=: strOrderId
AND SAL_ROWNUM = :nOrderRowNum
AND ITEMID = :strRowItemId
AND STOCKID != :strStockId AND STOCKLOC != :strStockLoc

```

```

If Not SqlPrepareAndExecute ( hSql, SqlBuf )

```

```

    Return FALSE

```

```

!
```

```

Set SqlBuf = '

```

```

UPDATE SAL_ORDERBATCH
SET STOCKID          = :strStockId,
    STOCKLOC = strStockLoc
WHERE COMPANY= :strCOMP
AND RECTYPE = 0
AND ORDERID          = :strOrderId
AND ORDER_ROWNUM = :nOrderRowNum
AND ITEMID = :strRowItemId
AND STOCKID != :strStockId AND STOCKLOC != :strStockLoc

```

```

If Not SqlPrepareAndExecute ( hSql, SqlBuf )

```

```

    Return FALSE

```

```

!
```

```

!
```

```

Set SqlBuf = '

```

```

UPDATE PUR_ORDERBATCH
SET STOCKID          = :strStockId,
    STOCKLOC = :strStockLoc
WHERE COMPANY= :strCOMP

```

```

AND RECTYPE = 0
AND SAL_ORDERID= :strOrderId
AND SAL_ROWNUM = :nOrderRowNum
AND ITEMID = :strRowItemId
AND STOCKID != :strStockId AND STOCKLOC != :strStockLoc

```

If Not SqlPrepareAndExecute (hSql, SqlBuf)

Return FALSE

!

Return TRUE

2.1.1 Parameter Modifications

New dep-parameter STOCKHANDLING is added.

REF.	PA-RAM. GROUP	PARAM. ROW	FIELD	DESCRIPTION	SCRIPT NAME
	DEP%	STOCKHANDLING	Value10	1= the alarm is created and the message box is shown to users	L53_depxxx.sql
	DEP%	STOCKHANDLING	Value11	1= update stock/stock location into the related SORs and PORs	

2.2 Database Modifications

N/A

2.3 Interface Modifications

N/A

2.4 Influence

2.4.1 Locos programming.

These changes will affect the following programs.

REF.	MODULE	FILE	CUSTOMIZATION
	REC	..\Leanv3_5\rec\Lc53\custom.apl	Locos5.3 specific
	REC	..\Leanv3_5\rec\Lc53\Newrec.apl	
	DEL	..\Leanv3_5\del\Lc53\custom.apl	Locos5.3 specific
	DEL	..\Leanv3_5\del\Lc53\Delivery.apl	Locos5.3 specific

2.5

2.5.1 Documentation

REF.	AFFECTED DOCUMENTATION	FILE
	New dep-parameter STOCKHANDLING	Parameter.doc
	Lean Receive User Manual	Lean_rec.doc
	Lean Receive Help File	Lean_rec.hlp
	Lean Delivery User Manual	Lean_del.doc
	Lean Delivery Help File	Lean_del.hlp

3 Testing Instructions

3.1 Component Test Description

Testing this new functionality requires that with dep-parameter STOCKHANDLING Value10 and Value11 are set on.

3.2 Testing Instructions

Parameter settings. See FD1346 chapter 2.1.5.

#	PARAM. GROUP	PARAM. ROW	FIELD	VALUE
1	DEP	STOCKHANDLING	Value10	1
2	DEP	STOCKHANDLING	Value10	0
3	DEP	STOCKHANDLING	Value11	1
4	DEP	STOCKHANDLING	Value11	0

REF.	PARAM#	CASE	TEST STEP
2.1.1.	1	'Receive' -form	Go to Receive (Id = 272) form and create receiving from Purchase Order which has differences in stock/stock locations than compared to the delivery related to the same Sales Order.
		Alarm created and the message box will be shown to user	Check that the alarm(s) are created and the message box will be opened.
	2	Receive' -form	Repeat step → Create Receiving from PO... Check that the alarm(s) aren NOT created and the message box will NOT be shown.
2.1.2.	3	'Deliveries' form	Go to Deliveries (Id 291) form and change stock locations in delivery rows and select 'Deliver selected' command.
			Check that stock locations are changed in related Sales Order and Purchase Orders.
	4		Repeat step → Change stock locations in delivery rows... Check that stock locations are NOT changed in related Sales Order and Purchase Orders rows.
2.4.2		Documentation changes	Verify that all the documentation changes have been made.

4

Open Issues

n/a

LIITE 9

Toiminnanohjausjärjestelmän vaatimusten tutkimus
Enterprise Resource Planning , ERP Requirements Research

Taulukko 1: Luokitus

Luokitus			
Vastuu	Selitykset	Explanation	
1	Toimitettu ERP tukee "out-of-the-box" ajattelua	1= SUP	Supported as delivered "out-of-the box"
2	Muutoksia tuetaan (GUI räätälöinti, raportit, lomakkeiden ja näyttöjen konfiguraatiot)	2= MOD	Supported via modifications (screen configurations, reports, GUI tailoring)
3	Tuki kolmannen osapuolen ratkaisulle	3= 3RD	Supported via an existing third party solution
4	Tuki kustomoinnille (muutokset lähdekoodiin)	4= CST	Supported via customization (changes to source code)
5	Tulevien version tuki	4= FUT	Will be supported in a future release
6	Ei toimitettu	6= NS	Not supported
7	Kriteerien kuvaus	7= Explanation	Criterion description
Prioriteetti	0...10, missä 10 on tärkein	Priority	0 to 10 , where 10 is most important
Painoarvo %	0...20, missä 20 on suurin	Weight value	0 to 20 , where 20 is biggest
Pakollisuus	Kyllä/Ei ("must-have") tekijät	Mandatory	Yes, only for "must-have" factors

Taulukko 2: Priotiteetti

Prioriteetti	Selitykset	
10	täytyy olla	must have
8	erittäin tärkeä	very important
6	tärkeä	important
4	kosmeettinen	nice to have
2	ei tärkeä	not important
0	ei tarvita	no need

Taulukko 3: Painoarvo

Painoarvo %	Selitykset	
20	suurin	biggest
18		
16		
14		
12		
10		
8		
6		
4		
2	alhaisin	lowest
0	ei arvoa	not value

4.5	Tuoterakenteiden muokkaus
4.6	Nimikkeiden kohdistaminen työvaiheisiin
5	Valmistusprosessit
5.1	Valmistusvaiheet
5.2	Kustannusrakenne
5.3	Pullonkaulat
5.4	Kuormitusryhmät
5.5	Vaihtoehtoiset valmistusreitit
6	Tarjouslaskenta ja hinnoittelu
6.1	Tarjousten laadintaperiaatteet
6.2	Hinnoitteluperiaatteet
6.3	Hinnastot
6.4	Alennukset
7	Tilaus
7.1	Tuotteen määrittely tilaukseen
7.2	Tarjouksen siirto tilaukseen
7.3	Sähköiset tilaukset
7.4	Tilausvahvistus
7.5	Tilaukseen liitettävät lisätiedot
8	Ennustaminen
8.1	Ennusteiden käyttö
8.2	Ennusteiden laadintaperiaatteet
9	Tuotannon karkeasuunnittelu
9.1	Suunnittelun aikajänne
9.2	Tuotantosuunnitelman suunnittelun periaatteet
9.3	Entä – jos vaihtoehtoistarkastelujen käyttö
9.4	Toimitusaikojen määrittelyn perusteet
10	Projektinohjaus
10.1	Projektien suunnittelu
10.2	Projektiresurssien hallinta
10.3	Projektien kustannushallinta

11	Tuotannon hienosuunnittelu
11.1	Tuotantoerien muodostamisen perusteet
11.2	Tuotantoerien ajoitus, tuotantosuunnitelman teko
11.3	Muutosten hallinta
11.4	Työjonojen muokkaus
11.5	Hienosuunnittelun apuvälineet
12	Lattiatason ohjaus
12.1	Valmistuksen dokumentaatio
12.2	Töiden jakaminen
12.3	Lattiapäätteiden käyttö
13	Takaisin raportointi
13.1	Raportoitavat asiat
13.2	kapasiteetin ja materiaalikäytön kirjaukset
13.3	Raportointi tiheys
13.4	Raportointi tekniikka
14	Ostot
14.1	Ostoehdotusten synty
14.2	Automaattisen varastovalvonnan käyttö
14.3	Nimikkeistön hankintatarpeiden tarkistamisen periaatteet
14.4	Ostoerän määräytymisperiaatteet
14.5	Ostotilausten yhdistely
14.6	Tilausrutiinit
14.7	Sähköiset tilaukset
14.8	Vastaanottokirjaukset
14.9	Hyväksymiskirjaukset
15	Varastot
15.1	Materiaalivarausten käyttö
15.2	Eräkohtaisten seurantojen tarve
15.3	Varastomateriaalien hinnoittelu
15.4	Monien varastojen hallintatarve

15.5	Varastopaikkojen määrittely
15.6	Varastotasojen suunnittelu
15.7	Varastokirjausten teko
16	Lähetys
16.1	Kuljetus erien muodostus
16.2	Pakkauserien ja kollojen muodostus
16.3	Keräilylistat
16.4	Keräilyn ja lähetyksen kirjaus
16.5	Kuljetusasiakirjojen luonti
16.6	Vientiasiakirjojen tarve
17	Taloushallinta
17.1	Kustannus seurannan periaatteet
17.2	Laskujen muodostaminen
17.3	Maksupostien hallinta
17.4	Palkanlaskenta periaatteet
17.5	Parake, TES-tulkkaus
17.6	Kassavirran suunnittelu
17.7	Käyttöomaisuuskirjanpito
17.8	Sähköinen laskujen tarkistus
17.9	Takaisinraportoinnin ja kulunvalvonnan liittäminen järjestelmään

Lähde:

Teknologiainfo Teknova Oy

[www-sivu].[viitattu 2.10.2010] Saatavissa:

<http://www.teknologiainfo.net>

LIITE 10

C-CEI mentelmän riskianalyysi

ERP-järjestelmän valinnan, käyttöönoton ja käytön riskit, arviointilomake

Vaikutus (Vaik.):	Todennäköisyys (TN):
1 Hyvin pieni (Vaikutus hankkeelle on hyvin pieni)	1 Hyvin pieni (Tuskin tapahtuu)
2 Pieni (Saattaa hieman vaikeuttaa ERP-järjestelmän hankintaa)	2 Pieni (Voi tapahtua, mutta ei todennäköinen)
3 Kohtalainen (Vaikeuttaa ERP-järjestelmän käyttöönottoa)	3 Kohtalainen (Ihan mahdollinen)
4 Suuri (Haittaa merkittävästi hankkeen onnistumista)	4 Suuri (Voi hyvinkin tapahtua)
5 Hyvin suuri (Vaikutus katastrofaalinen)	5 Hyvin suuri (Melko varmaan tapahtuu)

Riski	Vaik.	TN	Hallinta
Valinnan riskit			
Valitaan epäsopiva ERP-järjestelmä			
ERP-hankinnan perusteet epäselvät			
Yrityksen erityistarpeita ei ole määritelty			
Sopimusehtoja muutosten tekemisen osalta ei ole sovittu			
Ei ole riittävä osaamista sopimuksen tekoon			
Valitaan väärä järjestelmätoimittaja			
Väärinymmärrykset toimittajan ja asiakkaan välillä			
Toimittajan projektipäällikköä/kouluttajia ei ole tavattu ennen projektin alkamista			
Valitaan järjestelmä, joka ei sopeudu mahdollisiin tulevaisuuden toiminnan muutoksiin			

Järjestelmä edellyttää suurta panostusta tietotekniikan hankintaan (uusia koneita, uusia printtereitä...)			
Järjestelmän tehokas käyttö ei ole helpolla teknisesti toteutettavissa			
Ylimmän johdon tuki hankkeelle on riittämätön			
Valintavaiheessa ei ole mukana riittävää edustusta kaikista keskeisistä organisaatioista			
Projektipäällikön ja projektiryhmän valinta epäonnistuu			
Yhteistyöverkoston paine ja vaikutus järjestelmävalintaan			
Mahdollinen konsernitason vaikutus järjestelmävalintaan			
Muita havaittuja riskitekijöitä			

Käyttöönoton riskit	Vaik.	TN	Hallinta
Henkilöstö ei sitoudu käyttöönotettavaan toimintamalliin			
Ihmiset eivät näe järjestelmän mahdollistamia hyötyjä työssään			
Henkilöstön motivoitumista järjestelmän käyttäjäksi ei tueta riittävästi			
Toimintamallin muutoksen hallintaa ja sen edellyttämää johtajuutta ei löydy			
Ajan puute koulutukseen ja käyttöönottoon osallistumiseksi			
Oman yrityksen projektipäällikkö/projektiryhmä ei ole sopiva			
Yrityksen projektipäällikkö ei ole päätoiminen			
Järjestelmän käyttöönotto jää puolitiehen			
ERP-hanke haittaa ns. normaalia liiketoimintaa			

Ihmisillä ei ole riittäviä tietoteknisiä perusvalmiuksia järjestelmän laajempaan käyttöön			
Muiden tehtävien tekeminen vie aikaa ERP-projektilta eikä siihen kyetä halutusti panostamaan			
Kustannukset nousevat verrattuna alkuperäisiin arvioihin			
Käyttöönottoprojektilla ei ole ohjausryhmää tai se ei toimi kunnolla			
Ylin johto ei anna tarvittavaa tukea/resursseja projektin läpivientiin			
Järjestelmän käyttöönotosta ei ole tiedotettu tasapuolisesti läpi organisaation			
Aikataulu venyy			
Toimittaja ei ole sitoutunut riittävästi järjestelmän käyttöönottoon			
Ei olla tarpeeksi vaativia asiakkaita järjestelmätoimittajalle			
Asiakkuus ei ole riittävän tärkeä järjestelmän toimittajalle ja se ei pistä kaikkea peliin			
Toimittajan projektipäällikkö ei ole sopiva			
Järjestelmä ei käytännössä kuitenkaan toimi määritellyllä tavalla			
Ohjelmistomuutosten toteutus ja testaus ei toimi juohevasti			
ERP-projektin hallintaan liittyvä dokumentointi jää tekemättä tai se on raskas			
Koulutusta ei ole riittävästi tarjolla			
Koulutus ei ole sopivaa tai sitä ei viedä loppuun asti			
Vaadittavien tietojen kurinalainen käyttöönotto järjestelmässä ei onnistu			
Tietojen siirrossa syntyy virheitä (uusi vs. vanha järjestelmä)			
Tietojen siirtäminen vanhasta järjestelmästä uuteen ei tapahdu helpolla			
Tietojen kirjaaminen tietokantaan (nimikkeet, rakenteet, asiakkaat ym.) ei tapahdu sujuvasti			
Järjestelmän linkittäminen muihin ohjelmistoihin			
ERP-järjestelmää viritellään (omia virityksiä, hallitsemattomasti)			

Muita havaittuja riskitekijöitä			
---------------------------------	--	--	--

Käytön riskit	Vaik.	TN	Hallinta
Järjestelmää ei käytetä kurinalaisesti			
Vain osaa järjestelmästä käytetään, eikä se edesauta hyötyjen syntymistä			
Järjestelmää ei koeta toimintaa helpottavaksi			
Järjestelmä ei tue toimintatapoja ja muutoksia liiketoiminnassa			
Järjestelmän laajempaa käyttöä ei aloiteta heti järjestelmän käyttöönoton jälkeen			
Yrityksen toiminta on niin riippuvainen järjestelmästä, että järjestelmän kaatuminen haittaa liiketoimintaa			
Järjestelmän käytön aiheuttamat tietoturvariskit			
Järjestelmä ei ole helposti muokattavissa			
Järjestelmä ei tarjoa päätöksentekoa helpottavia dokumentteja			
Järjestelmän toimittaja ei kehitä järjestelmää tulevaisuudessa			
Epäformaali kommunikaatio pysyy rinnakkaisena (ja tärkeänä) toimintamallina			
Järjestelmän tukeminen lopetetaan toimittajan taholta			
Kaikkia haluttuja organisaatiotasoja ei saada tavoitteen mukaisesti järjestelmän käytön piiriin			
Muita havaittuja riskitekijöitä			

Lähde:

Teknologiainfo Teknova Oy

[www-sivu].[viitattu 2.10.2010] Saatavissa:

<http://www.teknologiainfo.net>