

[http://www.jyu.fi/library/tutkielmat/ 174/](http://www.jyu.fi/library/tutkielmat/174/)

Tomi Pienimäki

**TOIMINTOLASKENTAA JA -JOHTAMISTA TUKEVAT  
OHJELMISTOT SEKÄ NIIDEN KÄYTTÖ  
SUOMALAISSA YRITYKSISSÄ**

Tietojärjestelmätieteen  
Pro gradu -tutkielma  
5.5.1997

Jyväskylän yliopisto  
Tietojenkäsittelytieteiden laitos  
Jyväskylä

## TIIVISTELMÄ

Pienimäki, Tomi Petteri

Toimintolaskentaa ja -johtamista tukevat ohjelmistot sekä niiden käyttö suomalaisissa yrityksissä / Tomi Pienimäki.

Jyväskylä: Jyväskylän yliopisto, 1997.

99 s.

Tutkielma

Tässä tutkimuksessa on analysoitu toimintolaskentaa ja -johtamista tukevien ohjelmistojen ominaisuuksia. Tämän ohella tavoitteena on selvittää toimintoperusteista ajattelua tukevien ohjelmistojen käyttäjäkokemuksia.

Tutkimusmenetelmänä käytetään case-tutkimusta. Case-tutkimusmenetelmä on valittu, koska toimintoperusteista ajattelua tukevia ohjelmistoja on tutkittu vasta vähän ja näin voidaan analysoida ohjelmistoja yksityiskohtaisesti. Tutkimuksen tekemiseksi on muodostettu arviointikehikko, jonka avulla ohjelmistojen ominaisuuksia on arvioitu. Kehikko on muodostettu kahdesta pääosasta: johtamisen tietojärjestelmien yleisistä ominaisuuksista ja toimintolaskentaa sekä -johtamista tukevista ominaisuuksista.

Muodostettuun arviointikehikkoon on sijoitettu kolme toimintoperusteista ajattelua tukevaa ohjelmistoa. Nämä ohjelmistot ovat CostControl, EasyABCPlus ja ProDaCapo. Tutkimuksen tuloksena voidaan todeta, että ohjelmistot tukevat hyvin toimintolaskentaa, mutta puutteita löytyy toimintojohtamisen ja erityisesti prosessiajattelun tuesta.

AVAINSANAT: johdon tietojärjestelmä, EIS, toimintolaskenta, toimintojohtaminen, yritys

## SISÄLLYSLUETTELO

1 JOHDANTO.....	3
1.1 Tutkimuksen taustaa .....	3
1.2 Tutkimuksen tavoitteet ja rajaukset .....	3
1.3 Tutkimusmenetelmät ja tutkimuksen rakenne.....	4
2 TOIMINTOLASKENNAN TEORIAA.....	6
2.1 Toimintolaskennan taustaa .....	6
2.2 Syitä toimintolaskennan soveltamiseen .....	7
2.3 Toimintolaskennan käsitteitä ja teoriaa .....	9
2.3.1 Toimintolaskennan kustannusnäkökulma.....	12
2.3.2 Toimintolaskennan prosessinäkökulma.....	15
2.3.3 Kapasiteetin käsittely .....	16
2.4 Toimintojohtaminen .....	16
2.4.1 Toimintoanalyysi.....	18
2.4.2 Kustannusajurianalyysi.....	19
2.4.3 Suorituksen mittaaminen.....	20
2.5 Toimintojohtaminen yrityksen tietojärjestelmän osana.....	21
3 JOHTAMISEN TIETOJÄRJESTELMÄT .....	25
3.1 Yritysjohdon päätöksenteko .....	25
3.2 Perinteinen johdon tietojärjestelmä (MIS).....	28
3.3 Päätöksenteon tukijärjestelmä (DSS).....	29
3.4 Ylimmän johdon tietojärjestelmä (EIS).....	30
3.5 Data Warehouse.....	32
4 TOIMINTOJOHTAMISJÄRJESTELMIEN ARVIOINTIKEHIKON MUODOSTAMINEN.....	34
4.1 Kehikon muodostamisen taustaa.....	34
4.2 Toimintojohtamisjärjestelmä johtamisen tietojärjestelmänä.....	35
4.3 Arviointikehikko .....	36

4.4 Toimintojohtamisominaisuudet.....	39
4.4.1 Toimintolaskennan perusajatusten tukeminen .....	39
4.4.2 Toimintojohtamisen perusajatusten tukeminen.....	45
4.4.3 Prosessimallinnuksen tukeminen.....	47
4.5 Yleiset ominaisuudet .....	50
4.5.1 Raportointiominaisuudet .....	50
4.5.2 Järjestelmien tiedon analysointiominaisuudet .....	52
4.5.3 Järjestelmän tiedon käsittelyominaisuudet.....	53
4.5.4 Toimintojohtamisjärjestelmien käyttäjäominaisuudet.....	55
4.5.5 Järjestelmän kustannukset .....	57
4.5.6 Tekniset ominaisuudet .....	59
5 TOIMINTOJOHTAMISOHJELMISTOJEN SIOITTAMINEN	
ARVIOINTIKEHIKKOON .....	62
5.1 Tiedonhankinta ja tutkimusmenetelmä .....	64
5.2 Toimintojohtamisominaisuuksien sijoittuminen arvointikehikossa .....	67
5.2.1 Toimintolaskennan perusajatukset.....	67
5.2.2 Toimintojohtamisen perusajatusten tukeminen.....	74
5.2.3 Prosessimallinnuksen tukeminen.....	78
5.3 Yleisten ominaisuuksien sijoittuminen arvointikehikossa .....	79
5.3.1 Raportointiominaisuudet .....	79
5.3.2 Järjestelmien tiedon analysointiominaisuudet .....	82
5.3.3 Järjestelmän tiedon käsittelyominaisuudet.....	83
5.3.4 Toimintojohtamisjärjestelmien käyttäjäominaisuudet.....	86
5.3.5 Järjestelmän kustannukset .....	90
5.3.6 Tekniset ominaisuudet .....	91
5.4 Yhteenvedo tutkimustuloksista .....	94
6 YHTEENVETO .....	97
LÄHDELUETTELO	
LIITTEET	

# 1 JOHDANTO

## 1.1 Tutkimuksen taustaa

Johtamisen tietojärjestelmät ovat olleet merkittävässä asemassa tietojärjestelmätieteen tutkimuksessa. Täten johtamisen tietojärjestelmien kehityksestä ja ominaisuuksista on tehty useita tutkimuksia. Erityisesti EIS-järjestelmiä ja -ohjelmistoja sekä niiden ominaisuuksia on tutkittu paljon. Vastaavasti taloustieteiden alueella on tutkittu yrityksen johtamista ja laskentatoimen antamaa tukea yrityksen johtamiselle. Eräänä uusimpana laskentatoimen toteutuksena on esitetty toimintolaskentaa ja siihen perustuvaa toimintojohtamista.

Nykyiset laskentatoimen järjestelmät ovat poikkeuksetta tietojärjestelmiä ja toimivat osana yrityksen kokonaistietojärjestelmää. Erityisesti toimintolaskenta asettaa vaatimuksia yrityksen tietojärjestelmille, koska toimintolaskennan tarvitsema tieto poikkeaa perinteisestä laskentainformaatiosta. Toimintolaskentaa ja -johtamista tukeva järjestelmä on myös johtamisen tietojärjestelmä, koska toimintolaskentainformaatiota käytetään päätöksenteon tukena. Toimintolaskentaa tukevia järjestelmiä ja ohjelmistoja on kuitenkin tutkittu hyvin vähän. Aiheesta on kirjoitettu vain joitakin artikkeleita (Albright 1995, Borden 1991, Borden 1994). Tämä tekee tutkimusalueen relevantiksi etenkin, kun toimintoperusteinen ajattelu on vahvasti lisääntymässä yritysmaailmassa.

## 1.2 Tutkimuksen tavoitteet ja rajaukset

Tämän tutkimuksen tavoitteena on analysoida toimintoperusteista ajattelua tukevien ohjelmistojen ominaisuuksia. Toisena tavoitteena on selvittää käyttäjäkokemuksia toimintoperusteista ajattelua tukevista ohjelmistoista. Näiden ohella tavoitteena

on selvittää, millaisissa organisaatioissa toimintoperusteista ajattelua tukevia ohjelmistoja käytetään ja miten ohjelmistojen käyttäjät organisaatioissa sijoittuvat.

Tutkimuksella pyritään hahmottamaan toimintoperusteista ajattelua tukevien ohjelmistojen suhdetta toimintoperusteisen ajattelun teoriaan. Samoin tutkimuksella pyritään asemoimaan johtamisen tietojärjestelmien ja toimintoperusteista ajattelua tukevien järjestelmien välistä suhdetta. Tämä yhteys on tärkeä, koska on selvää, että toimintoperusteista ajattelua sovelletaan päätöksenteon tukena.

Tutkimus rajataan käsittelemään toimintopohjaista ajattelua varten valmistettuja ohjelmistoja. Tämä rajaus jättää tutkimuksen ulkopuolelle ohjelmistotuotteet, jotka sisältävät toimintoperusteista ajattelua tukevia ominaisuuksia, mutta ovat pääasias-  
sa tarkoitettu muuhun käyttöön. Tällaisia tuotteita ovat esimerkiksi erityiset pro-  
sessinmallinnusohjelmistot ja simulointiohjelmistot. Toinen tutkittuihin ohjelmis-  
toihin liittyvä rajaus koskee ohjelmistojen saatavuutta. Tutkimusalueeksi rajataan  
vain vapaasti saatavilla olevat ohjelmistot. Tämä rajaa tutkimuksen ulkopuolelle eri  
konsultti- ja tilintarkastustoimistojen omat ohjelmistotuotteet, joita he käyttävät  
toimintoperusteisen kustannuslaskennan projekteissaan. Edelleen tutkittavia oh-  
jelmistoja rajataan niin, että tutkimus käsittää vain Suomessa käytössä olevat oh-  
jelmistot. Ohjelmistojen käyttäjäkokemuksia ja käyttäjien mielipiteitä kartoitettaes-  
sa case-yrityksiä rajataan niin, että tutkitaan yrityksiä, joissa toimintoperusteista  
ajattelua tukevia ohjelmistoja käytetään aktiivisesti.

### **1.3 Tutkimusmenetelmät ja tutkimuksen rakenne**

Tutkimusmenetelmän valinnalla on keskeinen merkitys tieteellisessä tutkimuksessa. Tässä tutkimuksessa tutkimusmenetelmäksi on valittu case- eli tapaustutkimus. Case-tutkimuksilla on vahvat traditiot tietojärjestelmätieteen tutkimuksessa (Bensabat et al.1987, Kraemer et al. 1987, Lee 1989). Case-tutkimuksessa kuvataan vaiku-  
tussuhteita ja perehdytään yksityiskohtaisesti valittuun kohteeseen. Case-

tutkimusmenetelmää voidaan soveltaa, jos aihetta on tutkittu vähän (Bensabat et al.1987, s. 370). Case-tutkimusten ongelmana on, että niiden pohjalta on vaikea tehdä yleistyksiä.

Case-tutkimusmenetelmä on valittu, koska toimintoperusteista ajattelua tukevia ohjelmistoja on tutkittu vasta vähän ja case-tutkimus mahdollistaa ohjelmistojen yksityiskohtaisen tarkastelun. Ohjelmistojen lisäksi tutkitaan niiden käyttäjiä ja käyttäjäorganisaatioita.

Tutkimuksen tekemistä varten muodostetaan arviointikehikko, johon toimintoperusteista ajattelua tukevia ohjelmistoja sijoitetaan. Tutkimus perustuu ohjelmistotoimittajilta kerättyyn aineistoon, haastatteluihin, kyselylomakkeisiin sekä tekijän omiin kokemuksiin.

Tutkimuksen raportointi rakentuu seuraavasti: Toisessa luvussa käsitellään toimintolaskennan ja -johtamisen peruskäsitteitä sekä teoriaa. Kolmannessa luvussa tutkitaan päätöksenteon teoriaa ja näihin pohjautuvia johtamisen tietojärjestelmiä. Neljännessä luvussa muodostetaan kahden edellisen luvun kirjallisuustarkastelun perusteella arviointikehikko toimintoperusteista ajattelua tukevien järjestelmien ja ohjelmistojen ominaisuuksille. Viidennessä luvussa tutkitaan ohjelmistoja rakennetun arviointikehikon perusteella ja esitetään tutkimustulokset. Viides luku on jäsennetty arviointikehikon ominaisuuksien mukaisesti. Kuudennessa luvussa esitetään yhteenveto tutkimuksesta ja sen tuloksista sekä tehdään johtopäätökset.

## 2 TOIMINTOLASKENNAN TEORIAA

Tässä luvussa käsitellään toimintolaskennan taustaa, käsitteitä ja teoriaa. Näiden lisäksi selvitetään toimintojohtamista ja sen tietojärjestelmiä. Toimintolaskentaa tarkastellaan tietojärjestelmien näkökulmasta.

### 2.1 Toimintolaskennan taustaa

Teknologian kehitys on johtanut muutoksiin myös yrityksissä ja niiden toimintaympäristössä. Toimintaympäristöt ovat globalisoituneet ja siten monimutkaistuneet. Tämä on asettanut uusia vaatimuksia johdon laskentatoimen tuottamalle informaatiolle. Yritysten sisällä tuotantomenetelmät ovat kehittyneet, mikä on muuttanut merkittävästi syntyneiden kustannusten rakennetta. Automaation lisääntyminen on vähentänyt välittömiä kustannuksia ja lisännyt välillisten kustannusten osuutta kokonaiskustannuksista. Laskentatoimi ei kuitenkaan ole muuttunut vastaavalla tavalla. Perinteinen laskentatoimi tuottaa informaatiota lähinnä välittömistä kustannuksista ja jakaa välilliset kustannukset niiden perusteella.

Toinen perinteisen laskennan tyypillinen piirre on painottuminen ulkoiseen laskentaan. Laskentatoimi tuottaa informaatiota pitkälti kirjanpitoa varten. Itse yrityksen kannalta laskentatoimesta voidaan hyötyä vain, jos tuotetaan päätöksentekoa tukevaa informaatiota. Sheridanin (1989) mukaan kustannuslaskennan tulisikin olla osa yrityksen liiketoimintaprosessia eikä vain pelkkä laskentajärjestelmä.

Toimintolaskenta (activity-based costing) pyrkii vastaamaan kustannuslaskennalle asetettuihin uusiin haasteisiin. Toimintolaskenta keskittyy välillisten kustannusten kohdistamiseen ja johdolle hyödyllisen informaation tuottamiseen. Sen avulla ei pyritä tuottamaan yrityksen ulkoista informaatiota, vaan tavoitteena on ymmärtää



yrityksessä muodostuvien kustannusten syntyminen syy-seuraus suhteita. Luomalla kustannusten muodostumisesta selkeä kuva voidaan rakentaa perusta päätöksenteolle ja ohjausjärjestelmille.

## 2.2 Syitä toimintolaskennan soveltamiseen

Jotta yrityksen on järkevää vaihtaa olemassa olevaa kustannuslaskentajärjestelmäänsä, täytyy uudella järjestelmällä saavuttaa selviä etuja entiseen nähden. Toisaalta järjestelmän vaihtamiseen syynä voi olla myös tyytymättömyys nykyiseen järjestelmään. Cooperin (1989) mukaan on syytä harkita uutta kustannuslaskentajärjestelmää, kun:

- linjapäälliköt haluavat lopettaa tuottavilta näyttäviä tuotteita,
- vaikeasti valmistettavat tuotteet näyttävät tuottavan suuria voittoja,
- osastoilla on omia laskentajärjestelmiä,
- yritys on yksin erittäin kannattavilla markkinoilla,
- kilpailijoiden hinnat ovat epärealistisen edullisia ja
- asiakkaat eivät valita hinnankorotuksista.

Toisaalta kustannuslaskentajärjestelmän muutostarvetta ennakoivia merkkejä saattaa ilmetä myös yrityksestä itsestään aiheutuvina. Yrityksen perusrakenteissa ja toimintatavoissa tapahtuvat muutokset vaikuttavat suoraan kustannusrakenteiden muodostumiseen. Kustannusrakenteiden muuttuessa on myös laskentajärjestelmää muutettava. Tällaisina syinä Cooper (1989) mainitsee

- automaation kasvun,
- muutokset tuotestrategiassa,
- tuotanprosessin yksinkertaistuminen tai monimutkaistuminen ja
- teknologian muutokset.

Toimintolaskenta ei kuitenkaan ole välttämättä kaikille yrityksille ainoa oikea ratkaisu. Vaikka toimintolaskennan periaatteet soveltuisivatkin yrityksen toimintatapaan, tulee kuitenkin huomioida myös yrityksestä itsestään aiheutuvia seikkoja, jotka vaikuttavat muutoksen hyödyllisyyteen. Toimintolaskennan käyttöönoton hyödyllisyyteen vaikuttaa Cooperin (1991) mukaan kolme tekijää:

1. Yrityksen tietojärjestelmän taso
2. Virheellisten päätösten aiheuttamat kustannukset
3. Yrityksen tuotesortimentin monipuolisuus, kuten myös yrityksen koko ja toiminnan monipuolisuus

Tietojärjestelmän taso vaikuttaa toimintolaskennan rakentamisen helppouteen ja valmiin järjestelmän ylläpitoon. Perinteiset tietojärjestelmät on rakennettu organisaatorakennetta seuraten funktionaalisesti. Tyypillisesti yrityksissä on esimerkiksi erilliset tuotannonohjauksen ja taloushallinnon tietojärjestelmät. Tämä rajoittaa usein raportoinnin osastojen rajoja noudattavaksi ja kokonaiskuvan saaminen liiketoiminnasta vaikeutuu. Toimintolaskenta edellyttää prosessien mukaista ajattelua myös tietojärjestelmien osalta. Tietojärjestelmän tulisi pystyä tarjoamaan toimintolaskennalle monipuolista tietoa yrityksen liiketoiminnasta ja prosesseista.

Virheellisistä päätöksistä aiheutuu kustannuksia, joita päätöksentekoa tukevalla mahdollisimman oikealla informaatiolla pystytään vähentämään. Virheellisiä päätöksiä voidaan tehdä kaikkialla yrityksessä investointipäätöksistä tuotesuunnitteluun. Koska virheellisiä päätöksiä on vaikea todentaa, on myös niistä aiheutuvien kustannusten osoittaminen ongelmallista. Jotta toimintolaskennalla voidaan vähentää näitä kustannuksia, täytyy olettaa, että sillä pystytään tuottamaan perinteistä laskentaa oikeampaa informaatiota. Tämä oletus perustuu siihen, että toimintolaskennalla kohdistetaan välillisiä kustannuksia, mitä perinteiseen laskentaan perustuvassa päätöksenteossa ei ole tehty.

Tuotesortimentin monipuolisuus johtaa muun muassa pieniin eräkokoihin ja siihen että tuotteet kuluttavat vaihtelevasti yrityksen resursseja. Perinteisessä laskennassa ei huomioida eräkohtaisia kustannuksia, joten myös pienten erien lisäkustannukset jäävät huomioimatta ja kohdistamatta. Joissakin tapauksissa niillä on kuitenkin vaikutusta esimerkiksi tuotteiden lopettamispäätöksiin (Pienimäki 1995, s. 82). Virheellisyydet tuotekustannuksissa heijastuvat myös niiden perusteella kohdistettaviin välillisiin kustannuksiin, mikä aiheuttaa kokonaisvääristymän yrityksen kustannusrakenteessa. Tästä seuraa johtamisongelmia, koska päätöksentekijä ei voi luottaa saamaansa informaatioon.

### **2.3 Toimintolaskennan käsitteitä ja teoriaa**

Toimintolaskennan ja termin activity-based costing esittelivät Cooper ja Kaplan (1988). Toimintolaskentaa voidaan soveltaa sekä tuotekohtaiseen kustannuslaskentaan että yrityksen prosessien analysoimiseen. Sovellettaessa toimintolaskentaa yrityksen tuotekustannusten laskemiseen, on perusajatus hyvin yksinkertainen: toiminnot kuluttavat yrityksen resursseja ja tuotteiden valmistamiseen tai palvelujen tarjoamiseen tarvitaan näitä toimintoja. Käytettäessä toimintolaskentaa prosessien analysoimiseen, perusajatuksena on, että prosessit muodostuvat toiminnoista. Brimson (1992, s. 76) määrittelee toimintolaskennan seuraavasti: Toimintolaskenta on menetelmä, jolla kerätään, käsitellään ja raportoidaan yrityksen toimintoja koskevia taloudellisia ja toiminnallisia tietoja. Toisen määritelmän mukaan toimintolaskenta on taloudellisen ja ei- taloudellisen tiedon keräämistä organisaation merkittävistä toiminnoista ja tämän tiedon yhdistämistä organisaation tavaroihin ja palveluihin (Benke 1992). Molemmissa määritelmässä tuodaan esiin toimintolaskennan yhteys myös muiden kuin taloudellisten arvojen mittaamiseen. Tämä piirre erottaa toimintolaskennan perinteisistä kustannuslaskentatavoista, joiden tarkoitus on yleensä vain pyrkiä kohdistamaan yrityksessä syntyviä kustannuksia eri periaattein.

Toimintolaskennalla saadaan informaatiota myös yrityksen prosesseista, mikä auttaa ohjaamaan yrityksen toimintaa.

*Resursseilla* tarkoitetaan kaikkia tuotannontekijöitä, joista aiheutuvia kustannuksia toimintolaskennan avulla aiotaan kohdistaa. Resursseja ovat esimerkiksi ihmiset, koneet, kiinteistöt jne.. Resurssit kohdistetaan toiminnoille *resurssiajureilla* (resource driver) ja toiminnot kohdistetaan edelleen laskentakohteille *toimintoajureilla* (activity driver). Toimintoajurit ovat tekijöitä, jotka aiheuttavat toiminnon tekemisen eli toimintoajurit ovat tekemisen syy ja toiminnot niiden seuraus (Sharman 1993). Toimintolaskennan yksi keskeinen ajatus on toimintoajureiden käyttäminen toimintokustannusten kohdistamisessa. Näiden ajureiden valinta onkin kriittinen tekijä kohdistettaessa kustannuksia toimintoperusteisesti.

*Toiminto* (activity) kuvaa yrityksessä tehtävää työtä. Toiminnot muodostavat prosesseja ja ne kuluttavat resursseja. Toiminnoilla tuotetaan suoritteita. Suoritteet voivat olla tuotteita tai palveluita ja näille suoritteille toimintojen kustannuksia kohdistetaan. Toiminnot voidaan jakaa makro- ja mikrotason toimintoihin (Turney 1991, s. 129). Mikrotason toimintoja tarkastellaan tutkittaessa yrityksen prosesseja ja makrotoiminnot taas muodostuvat mikrotoiminnoista ja niitä käytetään tuotekohtaisessa kustannuslaskennassa.

Toiminnot jaetaan neljään tai viiteen ryhmään sen mukaan, mihin toiminnon tekeminen kohdistuu. Toimintojen kohdistustasoja ovat: yksikkö-, erä-, tuote- ja tehdastaso (Cooper ja Kaplan 1991). Näiden lisäksi joissakin lähteissä esitetään viidentenä tasona prosessitason toimintoja (Drury 1994, s. 281).

*Yksikötason* toiminnot ovat riippuvaisia valmistettujen tuoteyksiköiden määrästä. Näitä toimintoja tarvitaan, kun valmistetaan yksi tuoteyksikkö. Tällaisia toimintoja ovat esimerkiksi sorvaaminen, hiominen ja kokoonpano.

*Erätason* toimintoja tarvitaan jokaista erää kohden. Eriä voivat olla esimerkiksi valmistus-, osto-, tai varastointierät. Erätoiminnoilla tarkoitetaan erän aiheuttamia toimintoja, jotka synnyttävät vakiokustannuksen kutakin erää kohti. Tällaisia ovat esimerkiksi asetukset ja ostotilaustoiminnot.

*Tuotetason* toiminnot voidaan kohdistaa tietylle tuotteelle, mutta ei erillisille tuotteille tai tuoteyksiköille. Nämä toiminnot eivät siis riipu siitä, kuinka paljon tai monta erää jotain tuotetta valmistetaan. Tuotetason toimintoja ovat esimerkiksi tuotesuunnittelu ja tuoterakenteen muodostaminen.

*Tehdastason* toiminnot aiheutuvat itse yrityksen olemassaolosta. Tehdastason toimintojen kustannusten kohdistamisessa ei toimintolaskennasta ole sanottavaa hyötyä. Nämä kustannukset voidaan kohdistaa esimerkiksi muiden tasojen toimintojen kustannusten suhteessa. Toinen vaihtoehto on kohdistaa kaikki tehdastason kustannukset samalla toimintoajurilla. Tällöin toimintoajurin valinnalla voidaan oikean kohdistamisen sijasta tavoitella johtamista tukevia motivointitekijöitä. Näin esimerkiksi kohdistamalla tehdastason kustannukset ajurilla 'läpäisy aika' kannustetaan henkilöstöä tavoittelemaan lyhyempiä läpäisy aikoja. Tehdastason kustannuksia aiheuttavat esimerkiksi vartiointi ja ruokala.

*Prosessitason* toiminnot liittyvät yrityksen prosesseihin ja niiden ohjaamiseen. Tämän tason toimintoja ovat esimerkiksi laadunvalvonta ja työnjohto.

*Toimintokokonaisuus* (activity center) on joukko toimintoja, joilla on sama tavoite. Tällaisia ovat esimerkiksi tilausten tekeminen, laatu ja johtaminen.

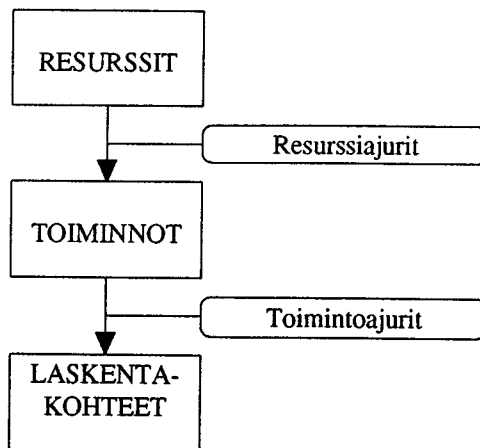
*Kustannusallas* (cost pool) taas tarkoittaa toimintojen kustannuksia (Lumijärvi 1993, s. 24).

### 2.3.1 Toimintolaskennan kustannusnäkökulma

Kustannuksia kohdistetaan joko aiheuttamisperiaatteen mukaan jäljittämällä tai jakamalla. Aiheuttamisperiaatteen mukainen kustannusten kohdistaminen tarkoittaa, että on löydettävissä tekijä, joka on aiheuttanut jäljitettävän kustannuksen. Kustannuksille pyritään löytämään aiheuttaja. Jos kustannusten aiheuttajaa ei löydetä, joudutaan kustannukset jakamaan. Tällöin kustannukset jaetaan esimerkiksi aiheuttamisperiaatteen mukaisten kustannusten suhteessa.

Toimintolaskennassa kustannuksia kohdistetaan toiminnoilta *laskentakohteille* (cost object). Laskentakohde on syy, miksi yrityksessä tehdään työtä ja josta halutaan erillistä kustannusten tarkastelua (Turney 1992b, s. 56). Laskentakohteita voivat olla esimerkiksi tuote, erä, asiakas tai yritys. Laskentakohteiden valinta auttaa erilaisten kannattavuusanalyysien tekemisessä. Jos esimerkiksi laskentakohteena käytetään asiakasta, voidaan tarkastella asiakkaiden välisiä kannattavuuksia eikä vain tuotekohtaisia kannattavuuksia. Usein perinteisessä laskennassa keskitytään pelkkien tuotekustannusten laskemiseen, vaikka yrityksen toiminnan kehittämisen kannalta olisi järkevämpää kiinnittää huomiota suurempiin kokonaisuuksiin, kuten markkina-alueisiin, asiakkaisiin tai myyntieriin.

Toimintolaskennassa resurssien kustannukset kohdistetaan toiminnoille resurssiajureilla. Vastaavasti toimintojen kustannukset kohdistetaan laskentakohteille toimintoajureilla. Toimintoajurilla mitataan sitä, kuinka usein toimintoja suoritetaan. Toimintoajurilla saadaan kohdistettua laskentakohteen kustannukset täsmällisesti, jos toimintoajuri korreloi toimintojen kulutusta (Turney 1991, s. 108). Toimintolaskennan kustannusten kohdistamisen periaate on esitetty kuviossa 1.



*Kuvio 1. Toimintolaskennan kustannusnäkökulman periaate*

Toimintoajureiden määrällä vaikutetaan yrityksestä saatavan ja päätöksenteossa käytettävän tiedon tarkkuuteen. Jos käytetään lukumääräisesti hyvin vähän toimintoajureita, saattaa tämä vääristää saatavan informaation oikeellisuutta. Toisaalta taas monen toimintoajurin käyttö vaikeuttaa jatkuvaksi tarkoitetun järjestelmän ylläpitoa. Samalla tietojen pohjalta suoritettava laskenta hidastuu päivitettävien ajureiden määrän suhteessa. Toimintoajureiden avulla on tarkoitus johtaa yrityksen toimintoja, joten myös johdettavuus kärsii, jos yrityksen johdon täytyy keskittyä liian moneen ajuriin.

Yrityksen on vaikea löytää kerralla oikeita toimintoajureita. Yleensä toimintoajureiden valinta etenee kierroksittain. Ensin valitaan esimerkiksi 40 ajuria, joista jatkossa käytetään enää 30:tä. Baileyn (1991) tekemän tutkimuksen mukaan Englannissa toimintolaskentaa käyttöön ottaneissa yrityksissä käytettiin keskimäärin 15 toimintoajuria.

Jatkuvuuden ja vertailtavuuden säilyttämiseksi pitkällä aikavälillä on toimintoajureiden muutoksissa kuitenkin noudatettava pitkäjänteisyyttä. Toimintoajureissa tapahtuviin muutoksiin vaikuttaa omalta osaltaan yrityksen toimintatavoissa tapahtuvat muutokset. Jos esimerkiksi yrityksen tuotantoprosessi

automatisoituu, on järkevää vaihtaa toimintoajuri 'työntekijöiden tekemät tunnit' ajuriksi 'konetunnit'. Jos taas yrityksen johto haluaa kiinnittää huomiota tuotannon läpäisy aikaan, tulee ajuri 'konetunnit' korvata ajurilla 'tuotannon läpäisy aika'. Lyhyellä aikavälillä muuttuvat kustannukset tulisi kohdistaa tuotteille volyyymiin pohjautuvilla toimintoajureilla (Drury 1989). Taulukossa 1 on esitetty esimerkkitoimintoja ja niiden toimintoajureita.

<i>Toiminto</i>	<i>Toimintoajuri</i>
Valmistus	Konetunnit
Laadunvalvonta	Tarkastuskerrat
Ostotilausten käsittely	Ostotilauskerrat
Yhteydenpito asiakkaisiin	Asiakaskontaktikerrat
Suunnittelu	Erilaisten tuotteiden lukumäärä

*Taulukko 1. Esimerkkitoimintoja ja niiden toimintoajureita*

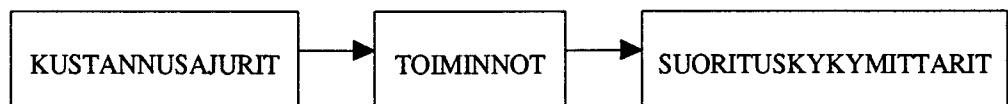
Toimintoajureiden valintaa rajoittavina käytännön tekijöinä ovat tietojen saatavuus ja päivityksen helppous. Vaikka periaatteessa tiedettäisiin joillekin toiminnoille oikea toimintoajuri, mutta tämän ajuritiedon kerääminen vaatii huomattavaa voimavarojen käyttöä suhteessa saatavaan hyötyyn, ei ajuria kannata valita. Valituilla toimintoajureilla tulisi olla myös selkeää käytännön merkitystä toimintolaskentajärjestelmän käyttäjille (Pemberton et al. 1996). Ajureiden lukumäärää karsittaessa yhtenä tekijänä tulee olla ajurilla kohdistettavien toimintokustannusten suuruus. Jos jollakin ajurilla kohdistetaan vain vähäinen kustannusmäärä, ei tällaista toimintoajuria kannata ylläpitää. Tällöin joudutaan helposti tilanteeseen, jossa kustannusten kohdistamiseen käytettävän ajurin selvittäminen maksaa enemmän kuin sillä kohdistettavat toiminnot.



### 2.3.2 Toimintolaskennan prosessinäkökulma

Toimintoajuriin hyvin läheisesti liittyvä käsite on *kustannusajuri* (cost driver). Toimintoajurien ja kustannusajurien erona on tarkastelunäkökulma: toimintoajurilla kohdistetaan kustannuksia laskentakohteille ja kustannusajureilla tarkastellaan toimintojen tehokkuutta ja kustannuksia. Kustannusajurilla tarkoitetaan niitä tekijöitä, jotka vaikuttavat yksittäisen toiminnon kustannuksiin ja siihen, miten toiminto suoritetaan (Lumijärvi 1993, s. 26). Esimerkiksi tilausten käsittelyn toimintoajurina voi olla 'tilausten lukumäärä' ja kustannusajurina tilausten virheettömyys. Tilausten virheettömyydellä kuvataan sitä, kuinka hyvin toiminto suoritetaan.

Kustannusajureita tarkastelemalla voidaan analysoida suoritettavien toimintojen tehokkuutta (Lumijärvi 1993, s. 26). Yhdistämällä näitä toimintoja ketjuksi saadaan muodostettua yrityksen prosessit. Kustannusajureiden avulla voidaankin tarkastella yrityksen prosesseja sekä niiden tehokkuutta. Kuviossa 2 on esitetty toimintolaskennan prosessinäkökulman periaate. Prosessinäkökulmaa käsitellään tarkemmin kappaleessa 4.4.3.



**Kuvio 2.** Toimintolaskennan prosessinäkökulman periaate

Tärkein tekijä kustannusajureita valittaessa on tieto siitä, mitä halutaan seurata. Tämä vaikuttaa suoraan siihen mitä halutaan johtaa, koska ei voida johtaa sellaisia asioita, joita ei seurata. Kustannusajureita voidaan myös muuttaa sen mukaan, mihin kulloinkin halutaan kiinnittää organisaation huomiota. Kuten toimintoajureissa myös kustannusajureissa tapahtuviin muutoksiin vaikuttaa omalta

osaltaan yrityksen toimintatavoissa tapahtuvat muutokset. Erilaisten kustannusajureiden käyttöä rajoittavat käytännössä tietojen saatavuus ja päivityksen helppous. Tiettyä yksittäistä toimintoa ei kannata seurata kustannusajurilla, jota ei voida käyttää muualla yrityksessä.

### **2.3.3 Kapasiteetin käsittely**

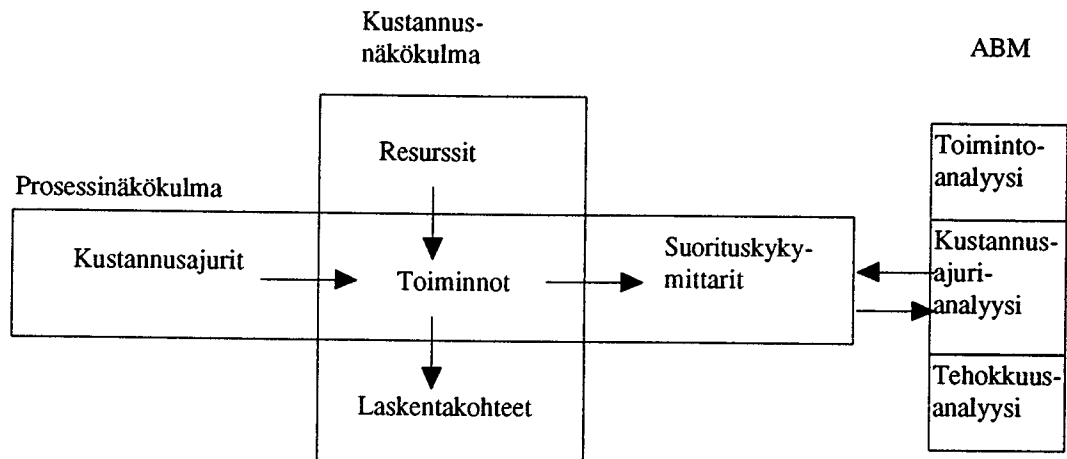
Toimintolaskenta poikkeaa perinteisistä kustannuslaskentamenetelmistä myös siten, että se on resurssikulutusmalli (Vehmanen 1994). Tämä tarkoittaa sitä, että laskeaan käytettyjen resurssien aiheuttamia kustannuksia eikä olemassa olevien resurssien kustannuksia. Toimintolaskennan periaatteiden mukaisesti laskentakohteille tulisi kohdistaa kustannuksia vain niiden käyttämistä resursseista (Kee 1995). Ylikapasiteetin kustannuksilla ei tulisi rasittaa tuotteita tai muita laskentakohteita. Tarkasteltaessa yrityksen kustannusten muodostumista, on selvää, että ylikapasiteetistakin aiheutuu kustannuksia, jotka on katettava. Käytettäessä laskentatoimen tuottamaa informaatiota päätöksenteossa, tuotteiden rasittaminen ylikapasiteetin kustannuksilla kuitenkin vääristää lopputulosta.

Olellainen ero toimintolaskennan ja perinteisen laskennan välillä on, että tunnistetaan ylikapasiteetin aiheuttamat kustannukset. Toimintolaskenta ei anna vastausta siihen, mitä ylikapasiteetin kustannuksille tulisi tehdä, mutta niiden tiedostaminen tukee jo sinällään päätöksentekijää. Resurssikulutuksen käsittelytapa omalta osaltaan puoltaa sitä, että toimintolaskennan tulisi olla ensisijaisesti päätöksentekoa tukeva laskentatapa eikä kirjanpitojärjestelmä.

## **2.4 Toimintojohtaminen**

Toimintojohtamisella tarkoitetaan toimintolaskentaan perustuvaa johtamistapaa. Toimintojohtaminen (Activity Based Management; ABM) on järjestelmä, joka

keskittyy toimintojen johtamiseen, jotta voidaan parantaa asiakkaan kokemaa arvoa ja saada lisävoittoa tämän arvon tuottamisesta (Turney 1992a, Ray ja Sclic 1993, Raffish ja Turney 1991). ABM:n perusajatus on johtaa toimintojen kautta yrityksen toimintaa kokonaisvaltaisesti. Tässä toimintojohtaminen käyttää toimintolaskentaa pää tietolähteenään (Ray ja Slic 1993). Toimintojohtamisen voidaan yksinkertaisesti sanoa olevan liiketoiminnan parantamista toimintolaskennan avulla (Turney 1992a, s. 20). Toimintojohtamisen ja -laskennan suhdetta voidaan tarkastella kuviossa 3 esitetyllä tavalla. Kuviossa esitetyn mukaisesti toimintojohtaminen sisältää toimintanalyysin, kustannusajurianalyysin ja tehokkuuden mittaamisen.



**Kuvio 3.** Toimintolaskennan ja toimintojohtamisen ero (Lumijärvi 1993, s. 37)

Toimintolaskentaa tarkastellaan kuviossa sekä prosessi- että kustannusnäkökulmasta. Kuviossa on yhdistetty aiemmin esitetyt toimintolaskennan kustannusnäkökulma ja prosessinäkökulma. Kustannusnäkökulma muodostuu resurssien kustannusten kohdistamisesta toimintojen avulla laskentakohteille (esimerkiksi tuote, asiakas, projekti). Prosessinäkökulman muodostavat toiminnot, joita ketjuttamalla syntyvät yrityksen prosessit. Tämä voidaan esittää myös työnkulun (workflow) avulla, sillä työnkulku voidaan määritellä kokoelmaksi yksittäisiä askeleita tai tehtäviä, jotka on järjestetty niin, että ne toteuttavat jonkin liiketoimintaprosessin (WfMC 1996, s. 28). Näistä prosesseista muodostuu yrityksen koko toiminta. Toimintojohtamisen

prosessinäkökulma liittyy läheisesti myös organisaation materiaali- ja liiketoimintaprosessien hallintaan (Greenwood ja Reeve 1992). Heidän mukaansa toimintojen analysointi ja prosessien muodostus tulisikin suorittaa ristiriidattomasti liiketoimintaprosessien kanssa.

#### **2.4.1 Toimintoanalyysi**

Toimintoanalyysillä pyritään kartoittamaan yrityksen toiminnot ja muodostamaan niistä prosesseja. Johtamisen kannalta on tärkeää tunnistaa yrityksen toiminnot ja prosessit, jotta niistä osataan muodostaa selkeitä kokonaisuuksia ja eliminoida ylimääräiset toiminnot. Yksi toimintojohtamisen tavoitteista onkin pystyä eliminomaan kaikki turhat toiminnot ja sitä kautta keskittyä vain lisäarvoa tuottaviin toimintoihin. Toisaalta toimintoja eliminoitaessa ei yksittäisiä toimintoja pidä käsitellä erillisinä, sillä jokin prosessin toiminto saattaa olla koko prosessin kannalta kriittinen, vaikka se erillisenä vaikuttaisi turhalta. Kunkin yksittäisen toiminnon tulisi kuulua johonkin prosessiin. Toiminnoista muodostetuille prosesseille on tunnusomaista, että:

- kullakin prosessilla on asiakas eli prosessilla on tuotos ja tällä tuotoksella vastaanottaja,
- prosessit ylittävät organisaation osastojen väliset rajat (Davenport ja Short 1990),
- prosessit ovat yrityksen organisaatorakenteesta riippumattomia ja
- kullakin prosessilla on omistaja, joka vastaa prosessista.

On ilmeistä, että perinteinen organisaatorakenne ei sovellu edellisten kaltaisten prosessien toteuttamiseen. Tästä seuraa johtamisongelma, sillä jos organisaatiota johdetaan hierarkisesti ja toiminnot etenevät organisaation poikki prosessien mukaisesti, aiheuttaa tämä tilanteen, että itse prosesseja ei johda kukaan. Näin ollen

jokaisella prosessilla tulisikin olla vastuuhenkilö. Pelkkä prosessivastuiden määrittäminen päällekkäin perinteisen organisaation kanssa taas johtaa eräänlaiseen matriisiorganisaatioon, joten myös johtamistoiminta tulisi määritellä yrityksessä uudelleen johtamisprosessiksi. Toimintojohtaminen on tällainen tapa hallita yritystä prosessien mukaisesti ja niiden avulla.

Yrityksen toimintojen ja prosessien tarkastelu voi tapahtua kolmella tasolla:

1. Yksittäisten toimintojen tehokkuuden tarkastelu
2. Toiminnoista muodostuvan prosessin tarkastelu
3. Prosessien kytkeytyminen toisiinsa (Lumijärvi 1993, s. 64)

Toimintojohtamisessa tulee analysoida kaikkia näitä tasoja, sillä ne ovat kytkeytyneet toisiinsa. Yrityksellä voi olla vaikeuksia sopeutua prosessien mukaiseen ajatteluun, koska se poikkeaa perinteisestä hierarkisesta organisaatioajattelusta. Varsinkin siirtymävaiheessa perinteisestä organisaatiosta prosessiorganisaatioon syntyy ongelmia, kun osa henkilöistä noudattaa perinteisiä toimintatapoja ja osa prosesseihin perustuvia menettelytapoja.

#### **2.4.2 Kustannusajurianalyysi**

Kustannusajurianalyysiä tehtäessä on huomioitava, että analyysin tarkoitus ei ole tutkia kustannusajureita tai niiden suoritusmääriä sinänsä, vaan toimintoja ajureiden avulla (Johnson et al. 1991). Pelkkien ajureiden analysointi aiheuttaa helposti vääriä johtopäätöksiä. Esimerkiksi kustannusajurin 'materiaalilaukset' suureen määrään puuttuminen voi lisätä tilattavien erien suuruutta ja kasvattaa samalla varastoarvoja.

Kustannusajurianalyysin avulla tulisi löytää kullekin toiminnolle oikea ajuri, jolla kuvataan sitä, kuinka hyvin toiminto suoritetaan. Tämän lisäksi kustannusajuriana-

lyysillä tulisi keskittyä ajureiden kulutukseen yksikkötasolla eli esimerkiksi, miksi yhtä suorituskertaa kohden vaaditaan ilmoitettu määrä tiettyä toimintoa. Kuten aiemmin on todettu kustannusajurit ovat toimintojohtamisen kannalta kriittinen tekijä, sillä yritystä johdetaan toimintojen avulla juuri kustannusajureiden kautta.

#### **2.4.3 Suorituksen mittaaminen**

Suorituksen mittaaminen on kaiken toiminnan johtamisen edellytys. Jos jotain ei voi mitata, ei sitä voi myöskään johtaa. Jotta toimintaa voitaisiin parantaa, on ensin selvitettävä kyseisen toiminnan nykytila. Tämän jälkeen on määritettävä mittari, jolla mitataan toiminnan parantumista. Yritystoiminnassa käytetyt mittarit ovat tyypillisesti olleet rahamääräisiä. Prosessiajattelu tuo mukanaan vaatimuksen myös muunlaisten mittareiden käyttämisestä. Muita kuin rahamääräisesti mitattavia asioita voivat olla esimerkiksi tuotannon tehokkuus, ympäristöystävällisyys, laatu tai asiakastyytyväisyys. Mitattavien asioiden selvittämisen jälkeen on selvitettävä, millä kyseisiä asioita voidaan mitata. Mittareina voivat olla esimerkiksi läpäisy aika, päästöjen määrä tai reklamaatioiden lukumäärä. Suorituksen mittaamisessa käytettävien mittareiden tulisi olla ymmärrettäviä ja yhteisesti hyväksytyjä. Suorituksen mittausta voidaan käyttää myös henkilöstön motivoimistekijänä (Lyne 1991). Mittaamisen käyttäminen motivoimistekijänä edellyttää, että käytettäviä mittareita on riittävän vähän. Toimintojohtamisen näkökulmasta tulisi mitata yrityksen toimintoja ja niiden tehokkuutta. Yksittäisten toimintojen seuranta yhdistelemällä voidaan analysoida prosessien tehokkuuden kehitystä.

Toimintojohtamisen avulla pyritään edistämään jatkuvaa kehitystä yrityksessä, mikä tapahtuu lisäämällä ymmärrystä avainprosesseista ja toiminnoista (Gwynne ja Ashworth 1993, Turney ja Reeve 1990). Toimintolaskentaa puolestaan voidaan käyttää jatkuvan kehityksen aikaan saamien tulosten arviointiin (Turney 1991). Toimintojohtaminen tarjoaa yritysjohdolle välineen tarkastella yrityksen resurssien kulutusta, uudelleen allokointia sekä eri prosessien kustannuksia.

Toimintojohtamisen hyödyiksi voidaan mainita muun muassa seuraavat seikat:

- monimutkaisuuden aiheuttamat kustannukset tunnistetaan,
- yleiskustannusten sisältö, luonne ja vaikutusmahdollisuudet ymmärretään,
- kannattavuus ja kannattamattomuus sekä niiden syyt tiedetään,
- tuottavuus tunnetaan ja siihen osataan vaikuttaa,
- laskennan tuloksia ryhdytään käyttämään hyväksi päätöksenteossa ja
- laskennasta tulee entistä ymmärrettävämpää (Lumijärvi 1993, s. 64).

## **2.5 Toimintojohtaminen yrityksen tietojärjestelmän osana**

Tietojärjestelmän kehittyneisyys vaikuttaa toimintolaskennan rakentamisen ja valmiin järjestelmän ylläpidon aiheuttamiin kustannuksiin. Toimintolaskentajärjestelmän vaatiman tiedon käsittelyn kustannukset muodostuvat pääosin kolmesta tekijästä: tiedon keräämisestä, siirtämisestä laskentajärjestelmään ja tämän tiedon muokkaamisesta kustannusmuotoon. Jos yrityksen tietojärjestelmä ei tue näitä tavoitteita, vaikeutuu toimintojohtamisen toteuttaminen.

Toimintolaskenta vaatii usein yritykseltä perinteistä laskentaa tarkempaa tiedon keräämistä, mikä nostaa mittauksen kustannuksia. Toimintolaskentaan tarvittava tieto on tyypillisesti myös erilaista kuin perinteisen laskennan tarvitsema tieto. Tämä asettaa lisävaatimuksia yrityksen tietojärjestelmälle. Jos tietojärjestelmä on rakennettu tukemaan perinteistä laskentatapaa, on siitä yleensä vaikea saada suoraan toimintolaskennan vaatimia tietoja. Tällaisessa tapauksessa tietojärjestelmän kehittyneisyys vaikuttaa järjestelmän kykyyn mukautua tarjoamaan uudenlaista päätöksentekoa palvelevaa informaatiota. Esimerkiksi kustannusajuritiedot ovat periaatteessa yksinkertaisia, mutta jos niiden tarve on jätetty huomioimatta tietojärjestelmän vaatimusmäärittelyssä, voi tiedon saaminen jälkikäteen olla vaikeaa. Joka ta-

pauksessa osa tarvittavasta tiedosta joudutaan keräämään manuaalisesti, mutta tämä osuus tulisi pyrkiä minimoimaan. Tiedon keräämisen automatisoinnilla on merkitystä varsinkin toimintolaskentajärjestelmää ylläpidettäessä. Toimintolaskentajärjestelmän rakentamisvaiheessa ei tiedon keruun automatisoinnilla ole niin suurta merkitystä, koska lopulliset tietotarpeet eivät vielä ole muodostuneet selviksi.

Tiedon siirtäminen laskentajärjestelmään tulisi pystyä hoitamaan suoraan tietojärjestelmästä. Käytännössä tämä tapahtuu muodostamalla linkkejä muun muassa yrityksen tuotannonohjaus- ja taloushallinnon järjestelmistä. Optimitilanteessa tiedon kerääminen ja siirtäminen saadaan integroitua niin, että tarvittava tieto pystytään keräämään tietojärjestelmillä ja siirtämään samassa muodossa suoraan toimintolaskentajärjestelmään. Yleensä muista tietojärjestelmistä laskentajärjestelmään siirrettävä tieto täytyy ainakin konvertoida laskentajärjestelmän hyväksymään muotoon.

Toimintolaskennassa käytetään muutakin kuin markkamääräistä informaatiota. Tämä johtaa siihen, että laskentajärjestelmään siirretty tieto täytyy muuttaa markkamääräiseen muotoon. Esimerkiksi ihmisresurssien kulutusta kuvaavat ajankäyttötiedot muutetaan tunneista palkkatietojen avulla kustannuksiksi. Käytännössä tämä voidaan tehdä eri toimintolaskentaohjelmistojen avulla tai rakentamalla oma taulukkolaskentapohjainen toimintolaskentasovellus.

Yrityksen harkitessa toimintolaskennan ja toimintojohtamisen käyttämistä toimintansa ohjaamisessa, sillä on kaksi vaihtoehtoa: Ensinnäkin on mahdollista laskea kustannukset kerran toimintoperusteisesti ja tehdä päätöksiä tämän laskelman perusteella. Toinen vaihtoehto on rakentaa päätöksenteon tueksi päivitettävä toimintolaskentajärjestelmä. Kertaluonteinen laskelma on yksinkertaisempi toteuttaa ja vaatii vähemmän resursseja, mutta se ei ole yhtä informatiivinen kuin päivitettävä järjestelmä. Yleensä kertaluonteista laskelmaa ei ole järkevää tehdä yhtä tarkalla tasolla kuin päivitettävää järjestelmää, mikä vähentää sen luotettavuutta. Toisaalta



taas päivitettävän järjestelmänkin tarkkuustasoa on syytä harkita päivittämisen yhteydessä lisääntyvän työmäärän vuoksi. Kertaluonteisen järjestelmän päätöksenteon tueksi voi rakentaa käyttäen esimerkiksi pelkää taulukkolaskentaohjelmaa. Tällöin laskentaan saadaan tietoa seuraavista lähteistä (Roth ja Borthick 1989):

- olemassa olevasta laskentajärjestelmästä,
- muista yrityksen raporteista ja
- manuaalisesti kerätyistä toimintomittareista.

Toinen valinta, jonka yritys joutuu tekemään on, integroidaanko toimintojohtamisjärjestelmä yrityksen muihin tietojärjestelmiin vai pidetäänkö se muista järjestelmistä erillisenä järjestelmänä.

Integroidulla järjestelmällä saavutettavista eduista merkittävin on, että se tukee toimintojohtamista jatkuvana toimintana eikä kertaluonteisena laskelmana. Samoin integroidussa järjestelmässä voidaan hyödyntää olemassaolevia liittymiä. Saatavan tiedon luotettavuutta taas lisää se, että kaikissa järjestelmissä käytetään samoja perustietoja. Integroidun järjestelmän suurin ongelma on, että nykyiset yritysten tietojärjestelmät eivät tue toimintolaskentaa. Tämä vaatii tyypillisesti toimintolaskennan ja muiden tietojärjestelmien integroinnin yhteydessä huomattavien muutosten tekemistä tietojärjestelmän perustoimintoihin. Muutosohjelmointi voi osoittautua monimutkaiseksi tehtäväksi, varsinkin jos yrityksellä on käytössä vanha, esimerkiksi DOS-pohjainen, tietojärjestelmä. Myös vanhaan tietojärjestelmään, sen elinkaaren eri vaiheissa, tehdyt räätälöidyt muutokset kostautuvat rakennettaessa integroitua toimintolaskentajärjestelmää. Räätälöinneistä johtuen ei koko järjestelmää voida päivittää kerralla valmistajan uudempaan tuotteeseen, vaan kaikki muutokset on tehtävä erikseen.

Erillisen järjestelmän etuna on, että se ei vaadi muutoksia olemassaoleviin järjestelmiin. Myös kustannuksiltaan erillisjärjestelmä on edullisempi toteuttaa kuin in-

tegroitu järjestelmä. Itsenäisen järjestelmän haittana on sen vaatima erillinen tietojen päivitys. Muutenkin tiedon kerääminen muista järjestelmistä vaatii erilaisten linkkien rakentamista tai tiedon tallentamista järjestelmään manuaalisesti, mikä hidastaa päivitystä. On myös olemassa riski, että erillisjärjestelmästä tulee ainoastaan laskentaosaston työkalu, joka jää muulle yritykselle vieraaksi. Tämä vähentää muun yrityksen sitoutumista toimintolaskentaan ja alentaa järjestelmän hyödyntämisen astetta.

Käytettäessä toimintolaskentaa ja toimintojohtamista yrityksen päätöksenteossa pitkällä aikavälillä, tulisi tätä varten rakentaa pysyvä toimintolaskentajärjestelmä, joka on integroitu yrityksen muihin tietojärjestelmiin. Kertaluonteisen laskelman perusteella ei voida seurata toiminnan kehittymistä, mikä on edellytys toimintojen johtamiselle.

### 3 JOHTAMISEN TIETOJÄRJESTELMÄT

Johtamisen tietojärjestelmien taustalla ovat johtamiseen ja päätöksentekoon liittyvät teoriat. Tässä luvussa käsitellään ensin päätöksenteon teoriaa, jonka jälkeen tarkastellaan erilaisia johtamisen tietojärjestelmiä. Johtamisen tietojärjestelmiä käsitellään niiden kehitysvaiheiden mukaisessa järjestyksessä.

#### 3.1 Yritysjohdon päätöksenteko

Johtamista ja johtajia on tutkittu paljon. Tämä johtuu pitkälti johtamisen ja johtajien kriittisestä asemasta yrityksen menestymisen kannalta. Mintzbergin (1975) mukaan Henri Fayolin vuonna 1916 johtajasta esittämä näkemys on edelleen varsin kuvaava. Sen mukaan johtaja suunnittelee, organisoii, antaa määräyksiä, koordinoi ja valvoo.

Kirjallisuudessa esiintyvän yleisen käsityksen mukaan johtajat toimivat erilaisissa rooleissa. Mintzberg (1975) esittää johtajan työn jakautuvan kymmeneen eri rooliin. Nämä roolit kuuluvat kolmeen pääluokkaan, jotka ovat yhdyshenkilö-, informaatiota käsittelevä- ja päätöksentekijärooli. Ensinnäkin johtajan auktoriteetista ja statuksesta johtuen hänen tehtäviinsä kuuluu toimia johtajana ja hoitaa organisaation ihmissuhteita. Informaatiota käsittelevässä roolissa johtaja levittää tietoa ja valvoo organisaatiota hänelle jaettavan informaation perusteella. Yksi johtajan tärkeimpiä tehtäviä on myös analysoida hänelle välitettävää informaatiota. Yhtenä informaatioon liittyvänä roolina johtaja toimii organisaationsa edustaja ulospäin. Päätöksentekijän roolissa johtaja allokoii resursseja, ohjaa organisaatiota ulkoisissa muutoksissa ja toimii neuvottelijana. Ajallisesti johtajat käyttävät suuren osan ajastaan neuvotteluissa (Mintzberg 1975).

Organisaation talouden johtamisessa tulevat johtajan rooleista esiin erityisesti informatiivinen rooli sekä päätöksentekijärooli. Johtajan tehtävänä on viestiä laskentatoimen yrityksen taloudellisesta tilasta tuottamaa informaatiota sidosryhmille. Saamansa informaation perusteella johtaja myös tekee yritystä koskevia päätöksiä. Näillä päätöksillä ohjataan yritystä, joten niitä voidaan pitää johtajan tärkeimpänä tehtävänä.

Vaikka laskentatoimessa joudutaan tuottamaan informaatiota, joka ei ole johtamisen kannalta relevanttia, tämä informaatio tulisi pyrkiä karsimaan päätöksentekijälle toimitettavasta informaatiosta. Päätöksenteon kannalta epärelevanttia informaatiota on usein ulkoisten sidosryhmien vaatima informaatio. Sen sijaan päätöksentekijän tarvitsemaa informaatiota ovat esimerkiksi erilaiset vaihtoehto- ja kannattavuuslaskelmat.

Päätöksentekotilanteissa on aina kysymys erilaisten valintojen tekemisestä. Yksinkertaisimmillaan tehdään valinta jonkun toimenpiteen suorittamisesta tai suorittamatta jättämisestä. Analysoitaessa organisatorista päätöksentekoa on loogista ottaa tarkastelun lähtökohdaksi näiden vaihtoehtojen välillä valintoja suorittava yksilö eli päätöksentekijä. Päätöksenteon tulisi olla rationaalista toimintaa, vaikka näin ei käytännössä aina ole. Keen ja Scott Morton (1978) esittävät rationaalisen päätöksentekijän ominaisuuksiksi seuraavia:

- Yksilöllä on pysyvä preferenssijärjestelmä, jonka avulla hän kykenee asettamaan eri toimintavaihtoehtoja paremmuusjärjestykseen.
- Yksilön preferenssijärjestelmä on luonteeltaan transitiivinen: yksilön preferoidessa vaihtoehtoa A vaihtoehtoon B nähden ja vaihtoehtoa B vaihtoehtoon C verrattuna, hän asettaa A:n etusijalle myös C:hen verrattuna.

- Yksilö tavoittelee toiminnassaan mahdollisimman täydellistä tarpeiden tyydytys-astetta omaksumansa preferenssijärjestyksen mukaisesti.
- Yksilö toimii johdonmukaisesti arvioimalla aina käytettävissä olevia toimintavaihtoehtoja etukäteen asetettujen preferenssien nojalla. Päätöksentekijän käyttäytyminen on siten kuvattavissa loogisena prosessina.

Päätöksentekoprosessia on tutkittu hyvin paljon. Tutkimuksissa esitetään useita malleja päätöksentekoprosessin etenemiselle. Simonin mukaan päätöksentekoprosessi on kolmetasoinen (Ahituv ja Neumann, 1987):

1. Tiedustelu (intelligence): ongelman tunnistaminen ja tiedon kerääminen
2. Suunnittelu (design): vaihtoehtoisten ratkaisujen suunnittelu
3. Valinta (choice): toteutettavan ratkaisun valinta ja sen soveltamisen tarkastelu

Cooke ja Slack (1984, s. 5) taas esittävät päätöksenteon kehänä, jossa Simonin malliin on lisätty päätöksenteko-ongelman tarkempi määrittely, tavoitteiden asettaminen sekä päätöksen toteuttaminen.

Yleiset päätöksenteon ongelmat voidaan jaotella jäsentyneisiin (structured), puoli-jäsentyneisiin (semi structured) ja jäsentymättömiin ongelmiin (nonstructured) (Keen ja Scott Morton 1978, Silver 1991). Samaa jaottelua voidaan käyttää myös strategiseen päätöksentekoon liittyvissä ongelmissa (Thierauf 1993).

Jäsentynyttä päätöksentekotilannetta voidaan kuvata seuraavasti (Keen ja Scott Morton 1978). Toiminnalle on asetettu yksiselitteisiä tavoitteita ja käytettävissä olevat toimintavaihtoehdot ovat tunnistettavissa tai kukin vaihtoehto on jollain tavalla etukäteen täsmennetty. Näiden lisäksi kunkin toimintavaihtoehdon seuraukset ovat määritettävissä, koska toimintaa koskevat syy-seuraus -suhteet tunnetaan.

Näin ollen päätöskriteereinä on mahdollista käyttää parhaan mahdollisen vaihtoehdon valintaa.

Puoliksi jäsentynyttä päätöstilannetta voidaan hahmotella seuraavasti (Silver 1991): Päätöstilanteesta voidaan rakentaa malli, mutta malli ei anna yksiselitteistä ratkaisua ongelmaan. Päätöksentekijän on valittava ratkaisu eri vaihtoehdoista.

Jäsentymätöntä päätöksentekotilannetta voidaan kuvata seuraavasti (Cooke ja Slack 1984, s. 32): Toiminnalle asetetut tavoitteet ovat epämääräisiä ja epäjohdonmukaisia. Päätöstilanteessa valintavaihtoehdot eivät ole selkeitä eikä päätöksentekijällä ole yksiselitteistä kuvaa siitä, miten päätös tulisi tehdä.

### **3.2 Perinteinen johdon tietojärjestelmä (MIS)**

Ensimmäistä tietojärjestelmätyyppiä, joka on tarkoitettu ensisijaisesti ylimmän johdon käyttöön kutsutaan nimellä (Management Information System, MIS). Tämä tietojärjestelmätyyppi oli esillä erityisesti 1960-luvulla ja 1970-luvun alkupuolella. MIS:lle on tunnusomaista kaikki yrityksen johdon tietotarpeet kattava lähestymistapa. MIS:n tavoitteena on tuottaa johdolle informaatiota oikea-aikaisesti, oikealla tarkkuudella ja oikeassa muodossa. MIS:n ongelma onkin johtamisessa tarvittavan olennaisen tiedon esille saaminen koko tietomassasta.

MIS:lle ei ole esitetty laajasti hyväksyttyä määrittelyä. Se voidaan kuitenkin ymmärtää tietojärjestelmäksi, jonka tarkoituksena on kerätä, rekisteröidä ja käsitellä organisaation sisäistä tietoa ja tuottaa yrityksen johdon päätöksenteossa ja suunnittelussa tarvitsemaa informaatiota.

Tyypillisesti MIS:llä pyritään varastoimaan tietoa ja automatisoimaan tuon tiedon käsittelyä (Davis ja Olson 1985, s. 11). MIS on tarkoitettu usein rutiinien tukemiseen on siten myös teknisesti orientoitunut järjestelmä.

### **3.3 Päätöksenteon tukijärjestelmä (DSS)**

Päätöksenteon tukijärjestelmän (Decision Support System, DSS) periaatteet esittelivät Gorry ja Scott Morton 1971. DSS:n syntyyn vaikuttivat MIS:ssä havaitut puutteet. MIS ei tuottanut johdolle sen tarvitsemaa informaatiota, vaan tuotetun tiedon hyödyt näkyivät enemmänkin rutiininomaisissa toiminnoissa, joita tehdään tyypillisesti alemmilla organisaatiotasoilla. Sprague (1980) esittää MIS:n ja DSS:n peruseroksi sitä, että MIS on painottunut tiedon käsittelyyn ja päätöksenteon tukijärjestelmä on keskittynyt päätöksentekoprosessiin ja sen tukemiseen. Toisena selkeänä peruserona järjestelmien välillä on, että MIS tukee lähinnä jäseneltyjä päätöksenteko-ongelmia kun taas DSS tarjoaa tukea myös puoliksi jäseneltyihin ja jäsentymättömiin päätöksenteko-ongelmiin (Sprague ja Watson 1993, Hogue 1987). Tätä seikkaa voidaan pitää merkittävimpänä kehitysaskeleena järjestelmien välillä.

DSS:n määrittely on jäänyt tieteessä varsin yleiselle tasolle. Sprague (1980) luonnehtii DSS:ää interaktiiviseksi tietokonepohjaiseksi järjestelmäksi, joka auttaa päätöksentekijää hyödyntämään tietoa ja malleja ratkaistessaan jäsentymättömiä ongelmia. Yksikäsitteisen määritelmän sijasta tutkijat ovat pyrkineet esittämään DSS:lle ominaisia piirteitä, mitkä erottavat ne muista tietojärjestelmistä. Tällaisina esitetään muun muassa seuraavia (Keen 1981):

- tukevat ei-rutiininomaista päätöksentekoa sekä ad hoc -analyysyjä,
- mahdollistavat nopean pääsyn tietoon ja yksilölliset raportit,
- tukevat usein mitä-jos tilanteiden selvittelyä,

- eivät tarjoa valmiita vastauksia päätöksenteko-ongelmiin, vaan tukevat niiden ratkaisua ja
- tukeutuvat tyypillisesti matemaattisiin malleihin.

Näiden lisäksi Silver (1991) esittää DSS:n ominaisuuksiksi tiedon esittämistä eri muodoissa ja simulointia. Simulointia on esimerkiksi mitä-jos analyysien tekeminen. Rakenteellisesti DSS muodostuu kolmesta pääkomponentista tietokannasta (database), päätöksentekoa tukevista malleista (model base) ja käyttöliittymästä (user interface) (Sprague ja Watson 1993). Verrattaessa päätöstukijärjestelmää rakenteellisesti MIS:ään, merkittävimpänä lisänä on malleista muodostettu kanta ja erona kehittyneempi käyttöliittymä.

### 3.4 Ylimmän johdon tietojärjestelmä (EIS)

Ylimmän johdon tietojärjestelmä (Executive Information System) on Rockartin ja Treacyn 1982 esittelemä käsite, jolla tarkoitetaan ylimmän johdon käyttöön tarkoitettu tietojärjestelmää (Rockart ja Treacy 1982). Kuten DSS:lle, myöskään EIS-järjestelmille ei ole olemassa yksikäsitteistä määritelmää. Watson, Rainer ja Koh (1991) määrittelevät EIS:n tietokonejärjestelmäksi, joka tarjoaa johtajalle helpon pääsyn sekä yrityksen sisäiseen että ulkoiseen tietoon, joka on relevanttia heidän työssään. Kirjallisuudessa on esitetty useita muitakin erilaisia määritelmiä EIS:lle, esimerkiksi Turban ja Watson 1989 sekä Paller ja Laska 1990. Yksikäsitteisen määritelmän puutteen vuoksi monet tutkijat tyytyvät esittämään EIS:n määritelmän sijasta EIS:lle tyypillisiä piirteitä.

Houdeshelin ja Watsonin (1987) mukaan EIS:n erottaa DSS:stä ja MIS:stä seuraavat piirteet:

- johtajat käyttävät EIS:ää suoraan ilman välikäsiä,



- EIS tarjoaa helpon ja reaaliaikaisen pääsyn organisaation tilasta kertoviin tietoihin,
- EIS-järjestelmät on suunniteltu ottaen huomioon johtamisen kriittiset menestystekijät ja
- ylimmän johdon tietojärjestelmä tarjoaa korkealaatuista grafiikkaa, kommunikointivälineen sekä mahdollisuuden tiedon varastointiin ja hakuihin.

EIS ei myöskään tarjoa yhtä laajaa päätöksenteon analyysien tukea kuin DSS (Houdeshel ja Watson1987). EIS-järjestelmille ominaisena piirteenä mainitaan useissa lähteissä erityisesti helppokäyttöisyys, jotta johtaja voi keskittyä olennaiseen teknologian asiaa vaikeuttamatta (Houdeshel ja Watson1987, Rockart ja Treacy 1982, Ball 1992, Barrow 1990). Helppokäyttöisyys on edellytys sille, että johtajat voivat käyttää EIS:ää itse ilman välikäsiä. Helppokäyttöisyys myös alentaa uuden käyttäjän kynnystä siirtyä käyttämään EIS-järjestelmää. EIS:n tärkeänä ominaisuutena pidetään useissa tutkimuksissa ulkoisista tietolähteistä saatavan tiedon käyttöä (Watson et al. 1992, Carlisle ja Alameddine 1990). Tosin Milletin ja Mawhinneyn (1990) mukaan EIS:ssä käytettävästä tiedosta 90 prosenttia on sisäistä tietoa. Tämä havainto vähentää ulkoisen tiedon käytön merkittävyyttä EIS-ominaisuutena.

EIS-järjestelmille on tyypillistä, että niillä tarkkaillaan menneisyyden tapahtumia (Millet et al. 1992). Menneisyyden tapahtumia tarkkailemalla johtaja voi tehdä päätelmiä tulevasta. Päätöksentekoa tukevassa laskentatoimessa tarkkailulla ja valvonnalla on merkittävä rooli, mikä yhdistää EIS-järjestelmät yrityksen talouden kehityksen seurantaan. Millet et al. (1992) tuovat esiin EIS:n piirteenä myös mahdollisuuden hakea johdon tarvitsemaa tietoa yrityksen tietokannoista ja muista tietojärjestelmistä. Muilla tietojärjestelmillä he tarkoittavat MIS-järjestelmiä. EIS:ää voidaan pitää yhtenä kehitysaskeleena johdon tietojärjestelmien kehittämisessä.

### 3.5 Data Warehouse

Termin Data Warehouse esitteli Inmon vuonna 1990. Hän luonnehti Data Warehousen (DW) niin, että sen tarkoitus on yhdistää eritasoisissa ja eri-ikäisissä tietojärjestelmissä oleva epäyhtenäinen ja eriaikainen tieto tiettyjen aihealueiden kannalta mielekkäällä tavalla (Hackathorn 1995).

Hackathornin (1995) mukaan Data Warehouse muodostuu viidestä keskeisestä tietovirrasta:

1. **Inflow.** Kerätään tietoa olemassaolevista järjestelmistä ja muista tietolähteistä sekä uudelleen strukturoidaan tätä tietoa.
2. **Upflow.** Lisätään edellä kerätyn tiedon arvoa. Tiedon arvoa lisätään tiivistämällä (packaging), kokoamalla yhteen (summarizing) ja järjestämällä tiedon jakelu (distributing).
3. **Downflow.** Poistetaan vanhentunut tai muuten epärelevantti tieto.
4. **Outflow.** Järjestetään käyttäjille vaivaton pääsy heille soveliaimman jalostusasteen mukaisiin tietoihin.
5. **Metaflow.** Muodostetaan kuvaus Data Warehousen sisällöstä, rakenteesta ja alkuperästä tietokannan käytön ja täydentämisen helpottamiseksi.

Data Warehouse sisältää hyvin paljon erilaista tietoa. Tälle tiedolle on tyypillistä ajallinen toistuvuus (time-variant) ja moniulotteisuus. Moniulotteisuudella tarkoitetaan, että samoja tietoja halutaan esimerkiksi usealla eri tavalla jaoteltuina.

Data Warehousen tietokantaratkaisut perustuvat Coddin (Codd et al. 1993) esittämään OLAP-konseptiin (On-Line Analytical Processing). OLAP-konsepti muodostuu 12:sta eri säännöstä, joista keskeisimmät ovat:

- *moniulotteinen käsitteellinen näkymä tietoon,*

- *läpinäkyvyys*; OLAP-tietokantaan tulee olla samanlainen käyttöliittymä kuin muihinkin sovelluksiin,
- *tietojen vapaa saatavuus*,
- *raportoinnin riippumattomuus*,
- *asiakas/palvelin -arkkitehtuuri*,
- *tasaveroiset ulottuvuudet* eli rakenne, toiminnot ja laskentasäännöt ovat samat kaikille ulottuvuuksille,
- *monen käyttäjän tuki*,
- *vuorovaikutteinen tietojen käsittely*: taulukon mistä tahansa solusta on voitava porautua yksityiskohtiin suoraan osoittamalla solua ja
- *joustava raportointi*: käyttäjän on voitava muokata vapaasti raporteissa näkyviä rivejä ja sarakkeita, järjestystä ja ulkoasua.

Data Warehousen etuina on, että sen avulla yritysjohto voi vertailla nykyisiä ja aiemmin tehtyjä ratkaisuja sekä hahmottaa koko liiketoiminnan kuvan yhdellä kertaa. Samoin tulevaisuuden skenaariot voidaan tehdä rationaalisesti käytettävissä olevien historiatietojen pohjalta.

## 4 TOIMINTOJOHTAMISJÄRJESTELMIEN ARVIOINTIKEHIKON MUODOSTAMINEN

Tässä luvussa muodostetaan kehikko toimintojohtamisjärjestelmien arvioimisen tueksi. Ensin käsitellään kehikon muodostamisen taustaa. Sitten tarkastellaan toimintojohtamisjärjestelmien ominaisuuksia suhteessa muihin johtamisen tietojärjestelmiin. Tämän jälkeen muodostetaan toimintojohtamisjärjestelmien arviointikehikko.

### 4.1 Kehikon muodostamisen taustaa

Tämän tutkimuksen tarkoitus on tarkastella toimintolaskentaa ja -johtamista tukevia ohjelmistoja. Jotta ohjelmistoja voitaisiin tarkastella yhteismitallisesti, on järkevää muodostaa arviointikehikko, jonka ominaisuuksien perusteella kaikkia ohjelmistoja analysoidaan samalla tavalla. Tällainen menettely on laajasti käytetty esimerkiksi EIS-järjestelmien tutkimuksessa (Watson et al. 1992, Carlisle ja Alameddine 1990, Bergeron ja Raymond 1992). Koska toimintoperusteista ajattelua tukevia ohjelmistoja käsittelevissä artikkeleissa ei tällaista kehikkoa ole, koettiin tarpeelliseksi muodostaa arviointikehikko. Arviointikehikossa esitetään toimintoperusteista ajattelua tukevissa järjestelmissä tarvittavat ominaisuudet. Arviointikehikon rakentamisen perustaksi määritellään ensin käsitteet toimintojohtamisjärjestelmä ja toimintojohtamisohjelmisto.

**Toimintojohtamisjärjestelmä** on johtamisen tietojärjestelmä, joka tukee toimintolaskentaa ja toimintojohtamista, ja on integroitu osaksi yrityksen muita tietojärjestelmiä. Vaikka järjestelmästä käytetäänkin termiä *toimintojohtamisjärjestelmä*, niin määritelmän mukaan siihen katsotaan sisältyvän myös toimintolaskennan tukemisen ominaisuudet.

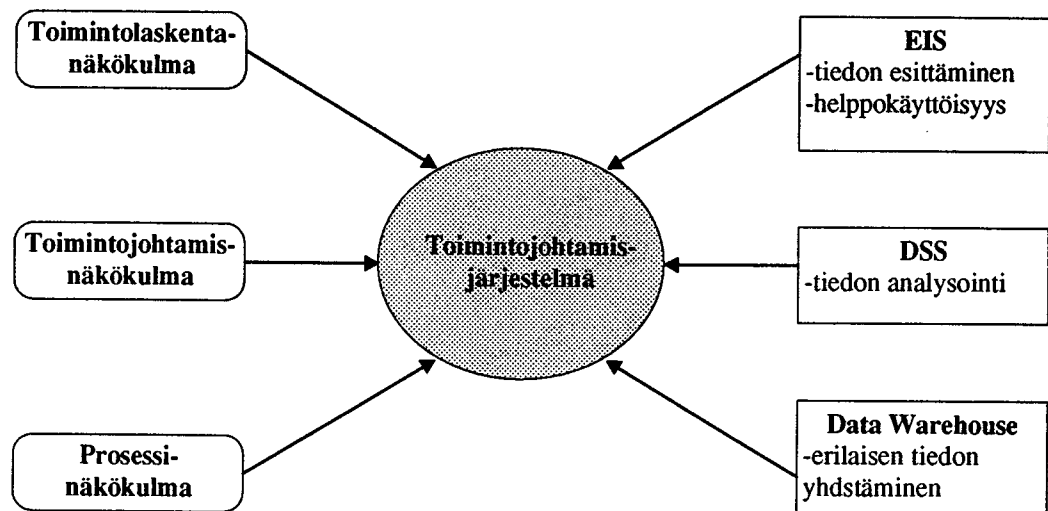
**Toimintojohtamisohjelmisto** on yksittäinen tai moduuleista koostuva ohjelmisto, joka tukee toimintolaskentaa ja toimintojohtamista. Toimintojohtamisohjelmisto voi siis olla erillään yrityksen tietojärjestelmistä.

#### 4.2 Toimintojohtamisjärjestelmä johtamisen tietojärjestelmänä

Toimintojohtamisen tueksi tarkoitetut järjestelmät poikkeavat erityyppisistä johtamisen tueksi tarkoitetuista tietojärjestelmistä. Ne sisältävät ominaisuuksia sekä päätöstukijärjestelmistä että EIS-ohjelmistoista. Keenin (1981) esittämistä päätöstukijärjestelmien ominaisuuksista toimintojohtamisjärjestelmien tulisi sisältää ainakin nopea pääsy tietoon. Silverin (1991) esittämistä DSS:n ominaisuuksista toimintojohtamisjärjestelmään tulisi kuulua tiedon esittäminen eri muodoissa ja mitä-jos analyysit. Houdeshelin ja Watsonin (1987) esittämistä EIS-järjestelmien piirteistä toimintojohtamisjärjestelmiin tulisi sisältyä järjestelmän käyttäminen ilman välisiä, korkealaatuinen grafiikka sekä muidenkin tutkijoiden esittämä helppokäyttöisyys (Houdeshel ja Watson 1987, Rockart ja Treacy 1982, Ball 1992). Toimintojohtamista tukevissa järjestelmissä on piirteitä jopa Data Warehouse -ratkaisusta. Data Warehouse -ominaisuuksiin toimintojohtamisessa tukevissa järjestelmissä liittyy erityisesti epäyhtenäisen ja eriaikaisen tiedon hallinta sekä tiedon ajallinen toistuvuus. Fahey (1996) esittääkin toimintolaskennan ja -johtamisen liittämistä Data Warehouse -ratkaisuun.

Kuitenkaan toimintojohtamista tukevia järjestelmiä ei voida sijoittaa selkeästi mihinkään edellisistä kategorioista. Vaikka johtamisen eri tietojärjestelmille ei ole löydettävissä yksiselitteisiä määritelmiä, niin määritelmien ja luonnehdintojen poikkeavuuksista ja vaihtelusta huolimatta on vaikea sijoittaa toimintojohtamista tukevia järjestelmiä mihinkään kategoriaan. Tämä johtuu paljolti itse toimintojohtamisen vaatimuksista, sillä toimintojohtaminen edellyttää järjestelmältä toimintolaskennan ja toimintojohtamisen perusajatusten sekä prosessin havainnollistamisominais-

suuksien tukemista. Näitä johtamisen tietojärjestelmät eivät tarjoa. Kuviossa 4 on esitetty toimintojohtamisjärjestelmän sijoittuminen johtamisen tietojärjestelmien kentässä. Kuviossa 4 on eri johtamisen tietojärjestelmien osalta esitetty ne pääpiirteet, joita myös toimintojohtamisjärjestelmät sisältävät. Toimintojohtamisjärjestelmien toisena osa-alueena on kuvattu niitä eri sisällöllisiä näkökulmia, joita järjestelmät sisältävät.



*Kuvio 4. Toimintojohtamisjärjestelmä johtamisen tietojärjestelmänä.*

### 4.3 Arviointikehikko

Arviointikehikko muodostetaan kahdesta pääosasta: toimintojohtamisen asettamisesta erityisvaatimuksista ja johtamisen tietojärjestelmien yleisistä ominaisuuksista. Toimintojohtaminen tarkoittaa yrityksen liiketoiminnan kehittämistä toimintolaskennan tarjoaman informaation perusteella (Turney 1992a, s.20). Tästä syystä arviointikehikkoon on sisällytetty myös toimintolaskennan vaatimia ominaisuuksia. Toimintoperusteinen ajattelu muodostuu kahdesta pääaspektistä kustannus- ja prosessinäkökulmasta. Tämän vuoksi on kehikkoon otettu mukaan myös prosessinäkökulma.

Johtamisen tietojärjestelmien ominaisuudet muodostetaan arviointikehikossa EIS-järjestelmien ominaisuuksien pohjalta. Johtamisen tietojärjestelmien ominaisuuksien osalta arviointikehikon muodostamisessa käytetään perustana Carlislen ja Alameddinen (1990) rakentamaa EIS-ohjelmistojen arviointikehikkoa. Tämän lisäksi pohjana on käytetty myös Bergeronin ja Raymondin (1992) luokitusta EIS-järjestelmien ominaisuuksista sekä eri lähteistä kerättyä tietoa EIS-järjestelmien ominaisuuksista (Watson et al. 1992, Carlson et al. 1992, Laska ja Paller 1992). Kehikon muodostamisessa näitä EIS luokitteluja on käytetty soveltuvin osin. EIS-ominaisuuksien lisäksi kehikon rakentamisessa on apuna ollut Coddin (Codd et al. 1993) esittämä OLAP (On-Line Analytical Processing) -konsepti, joka muodostaa Data Warehousen periaatteet.

Vaikka kuviossa 4 on esitetty toimintojohtamisjärjestelmien sisältävän ominaisuuksia myös päätöstukijärjestelmistä ja Data Warehousesta, perustuu arviointikehikon johtamisen tietojärjestelmien osuus pitkälti EIS-järjestelmien tutkimuskehysissä esitettyihin ominaisuuksiin. Tämä johtuu siitä, että näissä tutkimuskehysissä EIS-järjestelmien ominaisuuksina on tutkittu ominaisuuksia, jotka sisältyvät myös DSS:n (Keen 1981, Silver 1991) ja Data Warehousen (Hackathorn 1995) piirteisiin. Toinen syy EIS-järjestelmien tutkimuskehysten käyttämiseen on, että toimintojohtamisjärjestelmissä DSS:n, EIS:n ja Data Warehousen ominaisuuksia käsitellään pintapuolisemmin kuin esimerkiksi erityisesti DSS:n tutkimuksessa. Koska aiemmat EIS-tutkimuskehukset kartoittivat myös muissa johtamisen tietojärjestelmissä esiintyviä ominaisuuksia tässä tutkimuksessa riittäväksi katsotulla tarkkuudella, valittiin arviointikehikon perustaksi EIS-järjestelmien tutkimuskehukset.

Carlisle ja Alameddine (1990, s. 160) määrittelivät EIS:n: EIS on tietokonepohjainen järjestelmä, jolla voidaan tarjota johdon käyttöön analysointia, kommunikointia ja suunnittelutarpeita tukevaa informaatiota. Tämän määrittelyn pohjalta he muodostivat arviointikehikon sille, miten EIS-järjestelmät täyttävät määrittelyn mukaiset ominaisuudet.

Eri tutkimuksissa esitettyjen EIS-järjestelmien ominaisuuksien perusteella muodostetaan rakennettavan toimintojohtamisjärjestelmien arviointikehikon yleisiä ominaisuuksia käsittelevä osa. Arviointikehikkoa varten luokitellaan toimintojohtamisjärjestelmien ominaisuudet Taulukon 2 mukaisesti.

---

### **TOIMINTOJOHTAMISOMINAISUUDET**

#### **Toimintolaskennan perusajatusten tukeminen**

- Resurssien tukiominaisuudet
- Resurssikulutusmallin tuki
- Toiminnot-osan tukiominaisuudet
- Toiminnon tukitoiminto ominaisuus
- Laskentakohde-ominaisuudet
- Ajuri-ominaisuudet
- Toimintomallien yhdistämismahdollisuudet
- Laskentaominaisuudet
- Tiedon keräämisen tuki

#### **Toimintojohtamisen perusajatusten tukeminen**

- Toimintoanalyysin tuki
- Kustannusajurianalyysin tuki
- Tehokkuuden mittauksen tuki
- Toimintoperusteinen budjetoinnin tuki
- Asiakaskannattavuusanalyysin tuki

#### **Prosessimallinnuksen tukeminen**

- Erityyppisten työntekijöiden mallinnusominaisuudet
- Työntekijöiden ja prosessien kuvausominaisuudet

### **YLEISET OMINAISUUDET**

#### **Raportointiominaisuudet**

- Raporttien monipuolisuus
- Raporttien esitystapojen monipuolisuus
- Raporttien tuki analyyseille, budjetoinnille ja ennusteille
- Toimintomallien yhteenveto- ja vertailuraportit
- Raporttien räätälöintimahdollisuudet ja räätälöinnin tuki
- Poikkeamaraportit

#### **Toimintojohtamisjärjestelmien käyttäjäominaisuudet**

- Käyttöliittymä
- Grafiikka
- Käytön oppimisen helppous
- Käytön helppous

#### **Järjestelmän kustannukset**

- Perusohjelmistolisenssin hinta
- Monen käyttäjän lisenssien hinnat
- Ohjelmistotoimittajan tuen kustannukset
- Koulutuksen kustannukset

#### **Järjestelmien tiedon analysointiominaisuudet**

- Tietoon porautuminen
- Ennakoimattomat kyselyt
- Mitä-jos -analyysit

#### **Järjestelmän tiedon käsittelyominaisuudet**

- Järjestelmien välisen tiedonsiirtojen tuki
- Haut tietokantoihin
- Tiedon päivitysten tuki
- Ulkoisen tiedon käyttöominaisuudet

#### **Tekniset ominaisuudet**

- Avoin tai suljettu arkkitehtuuri
  - Yksittäiskoneessa toimiva ohjelmisto
  - Verkossa toimiva ohjelmisto
  - Verkko-ohjelmisto
  - Järjestelmän suojausominaisuudet
  - Näkymien- ja funktioiden suojausominaisuudet
  - Laitteisto- ja käyttöjärjestelmävaatimukset
- 

*Taulukko 2. Toimintojohtamisjärjestelmien arviointikehikko*



#### **4.4 Toimintojohtamisominaisuudet**

Toimintojohtamisjärjestelmien arviointikehikon ensimmäisen pääkohdan muodostavat toimintolaskenta- ja toimintojohtamisominaisuudet. Kehikon tässä osassa arvioidaan toimintojohtamisjärjestelmän sisällöllisiä ominaisuuksia ja toimintoperusteisen ajattelun tukea. Kehikon ensimmäisen osan muodostavat toimintolaskennan ja toimintojohtamisen perusajatusten tukeminen sekä prosessimallinnuksen tukeminen.

##### **4.4.1 Toimintolaskennan perusajatusten tukeminen**

Toimintolaskennan perusajatuksilla tarkoitetaan tässä kustannusten kohdistamista resursseilta toimintojen kautta laskentakohteille. Varsinaisen kustannusten kohdistamisen lisäksi kehikon tässä osassa on käsitelty sellaisia toimintolaskentamallin rakentamista ja käyttöä helpottavia tekijöitä, jotka liittyvät toimintolaskentaan eivätkä yleisiin ohjelmistojen käytettävyysominaisuuksiin. Tällaisia ominaisuuksia ovat esimerkiksi tiedon keräämisen tuki ja toimintomallien yhdistämismahdollisuudet.

*Resurssien tukiominaisuuksilla* tarkoitetaan niitä järjestelmän ominaisuuksia, joilla tuetaan toimintolaskentaketjun resurssiosan rakentamista ja hallintaa. Kohdassa tarkastellaan resurssien luomiseen liittyviä toimenpiteitä ja niiden helppoutta asiantuntemattomalle käyttäjälle.

Perusominaisuutena resurssiosassa tulisi voida määrittää yrityksen tilit, joita käytetään resursseina. Nämä tilit tulisi voida järjestää vastaavaan hierarkiaan kuin pääkirjan tilit tai olla ainakin muuten jaoteltavissa järkeviin kokonaisuuksiin hallittavuuden parantamiseksi. Jos yrityksessä on esimerkiksi satoja eri tilejä tai koneita, on niiden hallitseminen yhtenä järjestymättömänä listana vaikeaa ja hidasta. Resurssit tulisi voida yksilöidä tunnuksen avulla. Arviointikehikon tässä kohdassa arvioidaan

myös mahdollisuutta liittää resurssin yhteyteen nimen ja tunnuksen ohella resurssista kertovia lisätietoja. Tällaisia voivat olla esimerkiksi resurssin tarkempi kuvaus, sijaintipaikkatieto tai vastuuhenkilö.

Resursseina on ihmisten lisäksi käytännössä aina koneita ja kiinteistöjä. Koneet ja kiinteistöt sitovat pääomaa, joten ne aiheuttavat tileillä näkymättömänä kustannuseränä sisäisiä korkokustannuksia. Pääomakustannusten huomioimismahdollisuus tulisikin sisältyä resurssiosan ominaisuuksiin. Sisäisessä laskentatoimessa ei aina ole järkevää käyttää ulkoisen laskennan edellyttämiä poistoaikoja tai -arvoja, joten käyttäjän olisi voitava halutessaan itse antaa toimintojohtamisjärjestelmään muitakin tileistä poikkeavia lähtöarvoja.

Yksi toimintojen mukaisen kustannuslaskennan perusteita on *resurssikulutusmalli*. Näin ollen toimintojohtamisjärjestelmän tulisi tukea resurssien kulutukseen pohjautuvaa ajattelua. Tämä tarkoittaa sitä, että toimintojohtamisjärjestelmän resurssiosan tulisi sisältää kunkin resurssin kapasiteetti, jota toiminnot kuluttavat. Järjestelmästä tulisi käydä selkeästi ilmi kunkin yksittäisen resurssin käytetty kapasiteetti ja jäljellä oleva vapaa kapasiteetti tai kapasiteetin ylitys. Tässä kohdassa tutkitaan myös, voidaananko nämä tiedot saada koottuina käyttäjän valitsemista kokonaisuuksista graafeina ja raporteina.

*Toiminnot-osan tukiominaisuuksilla* tarkoitetaan toimintolaskentaketjun toiminnot-osan rakentamista ja hallintaa tukevia ominaisuuksia. Toiminnot-osan perusominaisuuksia ovat mahdollisuus luoda toimintoja ja kiinnittää niihin resursseja, joita toiminnot kuluttavat. Lähtöoletuksena on, että toiminto voidaan yksilöidä nimen ja tunnuksen avulla. Pelkkää nimen avulla yksilöintiä ei katsota riittäväksi, koska yrityksessä saattaa olla erillisiä yksiköitä, joissa tehdään samannimistä toimintoa. Toiminnot osassa kiinnitetään huomiota mahdollisuuteen merkitä toiminto lisäarvoa tuottavaksi tai tuottamattomaksi toiminnoksi (value added tai non value added), mikä on myös toimintolaskennan perusajatuksia. Esimerkiksi tuotannon toi-

minnot ovat lisäarvoa tuottavia ja hallinnon toiminnot ovat lisäarvoa tuottamattomia toimintoja.

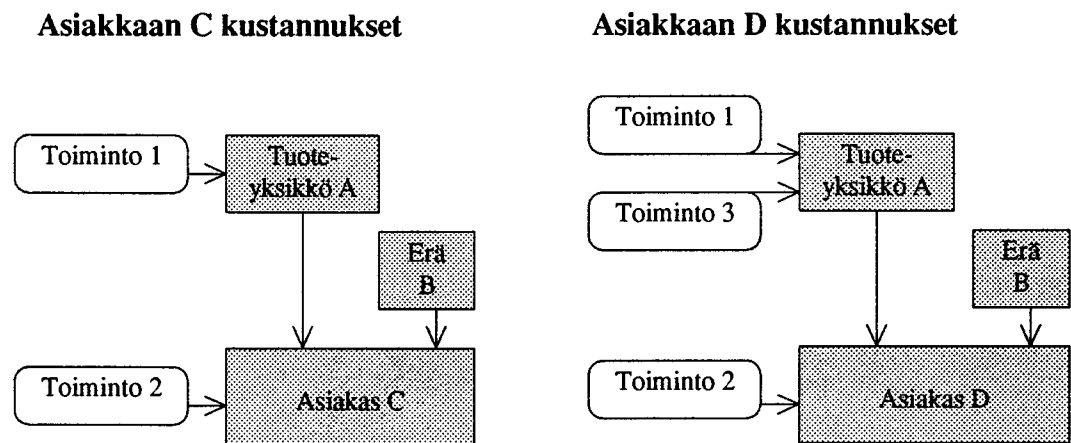
Näiden lisäksi arvioidaan järjestelmän mahdollisuuksia muodostaa toiminnoista toimintohierarkioita ja toimintokokonaisuuksia. Nämä ominaisuudet helpottavat järjestelmän käyttöä. Toimintohierarkian osa voisi olla esimerkiksi seuraavanlainen: yritys-toiminto muodostuu myynti- ja tuotanto-toiminnoista, joka edelleen koostuu sorvauksesta ja hionnasta. Toimintokokonaisuus tarkoittaa esimerkiksi kaikkia ostoon liittyviä toimintoja ilman hierarkista rakennetta.

Toiminnot on voitava jaotella laskentakohteen mukaisesti eri tasoille. Aiemmin esitetyn mukaisesti laskentakohdetasoja ovat yksikkö-, tuote-, erä-, prosessi- ja tehdastasot. Järjestelmän ei tarvitse tukea kaikkia näistä tasoista, mutta ainakin suurinta osaa. Jaottelemalla toiminnot laskentakohteen mukaisesti helpottuu toimintojen kohdistaminen laskentakohteille.

*Toiminnon tukitoiminto* -ominaisuus tarkoittaa mahdollisuutta määrittää toiminto toisen toiminnon tukitoiminnoksi sen sijaan, että kohdistettaisiin se suoraan laskentakohteelle. Yleensä toimintolaskentamalliin sisältyy tukitoimintoja (sustaining activity) (Turney 1991, s. 120). Käytännössä tällaisia ovat esimerkiksi hallintoon liittyvät toiminnot. Tässä kohdassa tutkitaan, onko järjestelmässä mahdollista kohdistaa toimintoa toiselle toiminnolle vai ainoastaan laskentakohteille.

*Laskentakohde-ominaisuudet* ovat toimintolaskentaketjun viimeisen osan ominaisuuksia. Ensinnäkin myös laskentakohdeosan olisi tuettava toimintojen jaottelua eri laskentakohdetasojen tasolle. Tämän lisäksi laskentakohteista tulisi voida muodostaa oma hierarkiansa. Esimerkiksi tietty asiakastoimitus koostuu tuotantoeristä, jotka muodostuvat tuoteyksiköistä. Kustannuksia tulisi voida kohdistaa erikseen tietylle laskentatasolle ja alempien tasojen kustannusten tulisi kumuloitua ylemmille tasoille. Tämä on esitetty kuviossa 5. Esimerkiksi tietylle asiakkaalle (Asiakas C tai D)

myydään tiettyjä tuotteita (tuoteyksikkö A), jolloin kyseisen asiakkaan kustannuksiksi kumuloituvat tuotteiden ja niiden tuotantoerien (Erä B) kustannukset. Näiden lisäksi asiakkaalle voidaan tehdä markkinointityötä (Toiminto 2), joka ei liity mihinkään yksittäiseen tuotteeseen. Tämä kustannus tulisi voida kohdistaa suoraan toiminnolta asiakkaalle ilman että tuotteita rasietaan näillä kustannuksilla. Toisaalta järjestelmän tulisi mahdollistaa myös asiakaskohtaiset eroavaisuudet tuoteyksikkötason (Toiminto 3) kustannuksissa. Erot voivat johtua esimerkiksi erilaisesta pakkauksesta tai laatutarkastusvaatimuksista.



*Kuvio 5. Kustannusten kohdistaminen eri laskentakohteille*

Laskentakohdeosan peruominaisuuksina arviointikehikossa pidetään laskentakohteen nimeämistä ja tunnuksen anto mahdollisuutta, hierarkian muodostusta, hintakenttää ja laskentakohteen lisätietokenttää. Hintakentän avulla voidaan laskea kate kullekin laskentakohteelle. Lisätietokenttää pidetään myös perusominaisuutena, koska käytännössä laskentakohteita on määrällisesti paljon, joten pelkkä nimi ei riitä kertomaan niiden välisiä eroavaisuuksia riittävän hyvin.

Laskentakohteille tulisi voida kohdistaa kustannuksia suoraan resursseilta toimintojen ohi. Tällainen tarve esiintyy esimerkiksi materiaalikustannuksilla, jotka voidaan kohdistaa suoraan tuotteelle.

Hinnoittelu-ominaisuus tarkoittaa mahdollisuutta hinnoitella jokin laskentakohde perustietona annettavasta vakiohinnasta poikkeavasti. Tähän voi olla syynä esimerkiksi erilainen tuotantotapa tai aggressiivinen markkinoiden valtaus.

*Ajuri-ominaisuudet* jakautuvat resurssiajuri- ja toimintoajuri-ominaisuuksiin. Ajurit ovat toimintoperusteisen johtamisen kriittisin kohta, koska niillä kohdistetaan syntyneet kustannukset. Resurssiajureilla kohdistetaan resurssien kustannuksia toiminoille ja toimintoajureilla toimintojen kustannuksia laskentakohteille. Ajuriominaisuuksien osalta tarkastellaan ajurien toteutusta järjestelmässä. Lähtöoletuksena pidetään resurssiajurin sijoittamista kunkin resurssin ja toiminnon yhteyteen sekä toimintoajurin sijoittamista kunkin toiminnon ja laskentakohteen yhteyteen. Ajureista tulisi myös ilmetä, kuinka paljon mitään ajuria on käytetty ja paljonko on vielä käyttämättä sekä yksittäisen ajurin kustannus. Esimerkiksi toimintoajurin 'erien lukumäärä' kohdalla tulisi näkyä yhden erän kustannus ja laskentakohteosiossa kuinka paljon tietty laskentakohde kyseistä ajuria käyttää.

*Toimintomallien yhdistämismahdollisuudet* ovat tarpeellisia, jos yritys koostuu esimerkiksi kahdesta eri tehtaasta ja yrityksen toimintaa halutaan tarkastella yhtenä kokonaisuutena. Kohdassa arvioidaan ohjelmiston antamaa tukea mallien yhdistämiselle. Yhdistämisessä on pidetty tärkeänä helppoa toteutusta ja että eri mallien mahdollisista päällekkäisyyksistä varoitetaan käyttäjää.

*Laskentaominaisuuksilla* tarkoitetaan toimintomalleissa suoritettavien laskutoimitusten tarkoituksenmukaista toteutusta. Vaihtoehtoisina tapoina ovat laskutoimitusten suoritus käyttäjän niin halutessa tai automaattisesti jatkuvasti suoritettava laskenta. Automaattisesti suoritettava laskenta helpottaa, mutta samalla hidastaa

ohjelmiston käyttöä ja toisaalta se myös vaikeuttaa mitä-jos analyysien tekemistä ainakin samassa laskentamallissa. Tämä siksi, että käyttäjästä riippumatta ohjelma päivittää mallia eikä käyttäjä voi tallentamalla kokeilla eri vaihtoehtoja.

*Tiedon keräämisellä* luodaan perusta toimintolaskennalle ja siten myös toimintojohtamiselle. Suuri osa tarvittavasta tiedosta on periaatteessa yleensä yrityksessä olemassa, mutta sen esille saaminen ja kerääminen tietojärjestelmistä tai ihmisiltä voi olla vaikeaa. Toimintojohtamisjärjestelmän perusedellytys on, että laskennassa ja johtamisessa tarvittavaa tietoa saadaan kerättyä riittävän tehokkaasti. Yhdysvalloissa 200 johtajan mielipiteitä kartoittaneessa kyselyssä kaksi kolmannesta piti tiedon keräämisen parantamista kriittisenä edellytyksenä täyden hyödyn saamiselle toimintolaskennasta (Geishecker 1996).

Kuten aiemmin kappaleessa 2.5 esitettiin, voidaan toimintolaskenta toteuttaa kertaluonteisena laskelmana tai päivitettävänä järjestelmänä. Käytettäessä toimintolaskentaa pysyvästi johtamisvälineenä, tulee laskennan olla päivitettävä järjestelmä. Tämä edellyttää mahdollisimman automatisoitua tiedon keräämistä laskennan perustaksi. Pääasiassa tietoa kerätään tietojärjestelmistä tai ihmisiltä. Käytännössä tietojärjestelmistä kerättävän tiedon osalta kysymykset liittyvät enemmänkin tiedon siirtoihin kuin tiedon keräämiseen. Näitä käsitellään toimintojohtamisjärjestelmien yleisten ominaisuuksien osuudessa kappaleessa 4.5.3. Kehikon tässä kohdassa arvioidaan järjestelmän tukea ihmisiltä tapahtuvaan tiedon keräämiseen. Käytännössä tämä kerättävä tieto tarkoittaa ihmisten ajankäyttökyselyitä (Miller 1990, s. 10). Henkilöiden ajankäytön perusteella toimintolaskennassa kohdistetaan heidän palkkakustannuksensa oikeille toiminnoille. Manuaalisesti kerättyä tämän tiedon saaminen koko henkilöstöltä on työlästä. Järjestelmä voi helpottaa tiedon hankkimista esimerkiksi tuetulla toimintotietojen muokkaamisella kyselykaavakkeen muotoon tai tukemalla taulukkolaskentaohjelmistoa niin, että tällä voidaan järjestelmän toimintotietojen pohjalta muokata kyselykaavake ja siirtää se täytettynä

suoraan toimintojohtamisohjelmistoon. Kaavakkeen jakelu tulisi voida toteuttaa yrityksen sisäisessä tietoverkossa.

#### **4.4.2 Toimintojohtamisen perusajatusten tukeminen**

Toimintojohtamisen perusajatuksia ovat toimintoanalyysi, kustannusajurianalyysi sekä tehokkuuden mittaus (Lumijärvi 1993, s. 37). Näiden lisäksi arviointikehiköön on otettu mukaan toimintoperusteinen budjetointi ja asiakaskannattavuusanalyysi. Toimintoperusteinen budjetointi on otettu mukaan, koska se liitetään toimintolaskentaan ja -johtamiseen useissa lähteissä (Morrow ja Conolly 1991, Cooper ja Kaplan 1992). Asiakaskannattavuusanalyysi taas koettiin tärkeäksi, koska tyypillisessä käytännön toimintolaskentaprojektissa halutaan tulokseksi saada juuri asiakaskohtaisia kannattavuuksia.

*Toimintoanalyysissä* tarkastellaan yrityksen toimintoja ja muodostetaan niistä prosesseja. Analyysillä tuotetaan tarkkaa informaatiota yrityksen toiminnoista. Toimintojohtamisjärjestelmä voi tukea toimintoanalyysiä tuottamalla mahdollisimman monipuolista toimintoja koskevaa tietoa. Tämä edellyttää vaihtoehtoisia tapoja ryhmitellä toimintoja erilaisiin ryhmiin ja luokkiin sekä mahdollisuutta laskea näiden toimintojen kustannuksista käyttäjän haluamia välisummia. Ryhmiteltäessä toimintoja, ryhmien tulisi olla toisistaan riippumattomia, jolloin sama toiminto voi kuulua useampaan ryhmään. Luokittelu tarkoittaa esimerkiksi toimintojen jakoa lisäarvoa tuottaviin ja tuottamattomiin toimintoihin tai sitä, että laskemalla yhteen laatutoimintojen kustannukset, voidaan tarkastella yrityksen laatukustannuksia. Rakennetussa arviointikehikössä toimintoanalyysin tukea arvioidaan toimintojen käsittelyn monipuolisuuden perusteella.

*Kustannusajurianalyysissä* tarkastellaan yrityksen kustannusajureita ja niiden valintaa. Kuten jo aiemmin kappaleessa 2.4.2 todettiin, kustannusajurit ovat toimintolaskennan kriittisin osa-alue. Kustannusajureilla analysoidaan toimintoja ja niiden

suorittamista. Toimintojohtamisjärjestelmän kustannusajuriominaisuudet tarkoittavat arviointikehikossa järjestelmän kykyä tukea kustannusajureiden valintaa ja mahdollisuuksia käsitellä valittuihin ajureihin tai ajurimääriin tehtäviä muutoksia. Tämän lisäksi tarkastellaan järjestelmän ominaisuuksia analysoida vaihtoehtoisten ajurien ja ajurimäärien vaikutuksia tuloksiin.

*Tehokkuuden mittauksella* tarkoitetaan yksittäisen toiminnon tai prosessin suoritusten arviointia. Arviointikehikossa tarkastellaan, kuinka hyvin järjestelmä tukee tehokkuuden mittaamista. Kehikossa tehokkuuden mittausta arvioidaan sen monipuolisuuden perusteella. Tehokkuutta voidaan mitata vertaamalla suoritusta aikaisempaan tai suunniteltuun suoritustasoon. Samoin vertailuja voidaan tehdä yrityksen eri yksiköiden kesken. Mittaukseen tuo monipuolisuutta myös mahdollisuus käyttää vaihtoehtoisia mittareita ja vaihtoehtoiset tavat esittää mittauksen tuloksia. Erilaisia mittareita voivat olla esimerkiksi luokitellut tai numeeriset asteikot.

*Toimintoperusteinen budjetointi* tulee tarpeelliseksi, kun yritys siirtyy käyttämään toimintolaskentaa kustannusten laskemisessa. Toimintolaskennan kokonaisvaltainen käyttö edellyttää toimintoperusteista budjetointia. Toimintoperusteisessa budjetoinnissa tulisi markkamäärien lisäksi voida budjetoida myös resurssien tarve. Arviointikehikossa tarkastellaan toimintojohtamisjärjestelmän tukea toimintoperusteiselle budjetoinnille. Arvioitavina ominaisuuksina ovat mahdollisuus budjetoida toiminnoille ja prosesseille kustannuksia sekä mahdollisuus budjetoida resurssien tarve. Tämän lisäksi tutkitaan järjestelmän tuki perinteisessäkin budjetoinnissa edellytettäville laskentakohteiden tuottojen budjetoinnille.

*Asiakaskannattavuusanalyysillä* selvitetään eri asiakkaiden kannattavuuksia. Toimintolaskennan ja -johtamisen avulla pyritään tyypillisesti selvittämään juuri asiakaskannattavuuksia (Cooper et al. 1992). Tästä syystä asiakaskannattavuusanalyysi otettiin mukaan arviointikehikkoon. Asiakaskannattavuusanalyysi edellyttää toimintojohtamisjärjestelmältä kykyä kohdistaa kullekin asiakkaalle tämän aiheuttamat



kustannukset. Järjestelmän tulisi myös tarjota mahdollisuuksia vertailla eri asiakkaiden kannattavuuksia esimerkiksi graafein ja raportein.

#### 4.4.3 Prosessimallinnuksen tukeminen

Toimintojohtamisessa on kaksi näkökulmaa: kustannusnäkökulma- ja prosessinäkökulma. Edellä on käsitelty kustannusnäkökulman mukaista toimintolaskentaketjua, jolla lasketaan tuotteiden ja muiden laskentakohteiden kustannukset. Toimintojohtamisen toinen osa-alue koostuu prosessien muodostamisesta toimintojen pohjalta ja näiden prosessien johtamisesta sekä kustannusten laskemisesta. Prosessinäkökulma on myös tärkeä, sillä esimerkiksi Pohjois-Amerikassa haastatelluista 324:stä linja- ja talousjohtajasta 92% sovelsi toimintojohtamista prosessien parantamiseen (Swenson ja Flesher 1996). Toimintojohtamisen ja prosessiajattelun yhteys on tuotu esiin useissa lähteissä esimerkiksi (Greenwood ja Reeve 1992, Campi 1992)

Prosessimallinnusominaisuuksia tarkastellaan tässä työkulun avulla, koska toimintojohtamisessa prosessit muodostetaan yksittäisistä toiminnoista, mikä on työkulujen rakentamisen kanssa yhtenevä mallinnustapa. Työkulun hallinnassa toiminnoista vain käytetään nimitystä tehtävä tai askel. Arviointikehikon prosessiosan perustana käytetään Workflow Management Coalitionin (WfMC) työkulun viitemallia (Reference model) ja sen standardiliittymiä (WfMC 1994). WfMC on otettu arvioinnin pohjaksi, koska se on tunnustettu ja puolueeton työkulunhallinnan standardointielin.

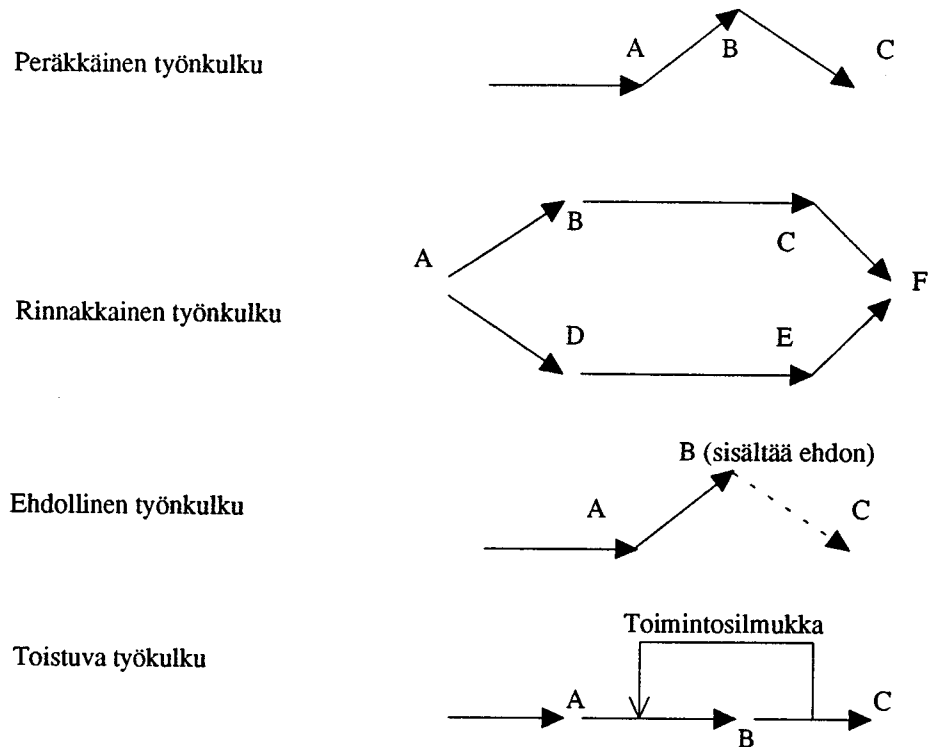
WfMC määrittelee viisi standardiliittymää työkuluille. WfMC:n (WfMC 1994) liittymät ovat prosessin kuvaustyökalut -liittymä (Process Definition Tools Interface 1), työkulun asiakassovellus -liittymä (Workflow Client Application Interface 2) ja käynnistetyn sovelluksen -liittymä (Invoked Application Interface 3). Näiden lisäksi ovat vielä muut työkulun hallinnan ohjeistukset -liittymä (Other Workflow

Enactment Services Interoperability Interface 4) sekä hallinto- ja valvontatyökalut -liittymä (Administration & Monitoring Tools Interface 5). Arviointikehikossa on sovellettu vain prosessin kuvaustyökalut -liittymän sisältöä. Muiden liittymien ominaisuudet on jätetty pois, koska niitä ei ole koettu tarkoituksenmukaisiksi kehikon kannalta. Prosessin kuvaustyökalut -liittymän ominaisuuksiakin sovelletaan huomattavasti karsittuina. Tämä siksi, että arvioitavia järjestelmiä ei ole varsinaisesti tarkoitettu työnkulun hallintaan, vaan työnkulun käsittely on otettu yhdeksi osaksi toimintojohtamisjärjestelmien ominaisuuksia. Arviointikehikossa keskitytään prosessien ja työnkulun osalta vain tarkastelemaan erityyppisten työkulkujen ja prosessien muodostamismahdollisuuksia määritellyistä toiminnoista. Arvioinnissa kiinnitetään huomiota myös työkulkujen ja prosessien kuvausominaisuuksiin.

*Erityyppisten työkulkujen mallinnusominaisuudet* tarkoittavat mahdollisuuksia mallintaa peräkkäisiä, rinnakkaisia, ehdollisia tai silmukkana toistuvia työkulkuja. Nämä vaihtoehdot on esitetty kuviossa 6. Mallinnuksella tarkoitetaan tässä kykyä esittää prosessi graafisesti, mitä on pidetty toimintojohtamisjärjestelmien kohdalla riittävänä mallinnustapana. Prosessin graafinen kuvaaminen on tärkeää kokonaisuuden hahmottamisen kannalta ja edellytys prosessin parantamiselle (Swenson et al. 1994).

*Työkulkujen ja prosessien kuvausominaisuudet* ovat prosesseihin liitettäviä lisätietoja ja tarkennuksia. Tässä prosessien kuvauksilta edellytetään, että prosessien eri vaiheisiin voidaan liittää kustannus- ja kapasiteettitietoja. Näiden lisäksi prosessiin tulisi voida liittää sen sanallinen kuvaus, tieto prosessin vastuuhenkilöstä (prosessin omistajasta) sekä prosessin alku- ja loppupisteestä.

Toimintojohtamisjärjestelmät perustuvat kustannusvirtojen ohjaamiseen, joten prosessien vaiheiden ja työkulkujen kustannusten mallinnus on koettu tärkeäksi ominaisuudeksi. Kapasiteettitietojen mallinnusominaisuudella pyritään tukemaan toi-



**Kuvio 6.** Vaihtoehtoiset työnkulut. Kirjaimilla on kuvattu yksittäisiä toimintoja.

mintolaskennan mukaista resurssikulutusmalliajattelua. Kapasiteettitiedoilla ja etenkin tämän kapasiteetin kulutustiedoilla saadaan selville yrityksen prosessien pullonkaulakohdat. Prosessin sanallisella kuvausominaisuudella voidaan kuvata prosessia pelkkää työnkulkukarttaa monipuolisemmin. Sanalliseen kuvaukseen voidaan esimerkiksi liittää tieto siitä, kuuluuko kyseinen prosessi yrityksen ydinprosesseihin. Prosessin alku- ja loppupistetiedoilla hahmotetaan koko prosessin kulku ja suhde muihin prosesseihin. Nämä ominaisuudet eivät ole kovinkaan monipuolisia prosessin mallintamista ajatellen, mutta toimintojohtamisjärjestelmille ne on katsottu riittäviksi, koska uusien prosessien muodostamisominaisuuksia tärkeämpiä toimintojohtamisjärjestelmissä ovat ominaisuudet, joilla pystytään hallitsemaan prosessien kustannustietoja.

## 4.5 Yleiset ominaisuudet

### 4.5.1 Raportointiominaisuudet

Toimintojohtamisjärjestelmien yksi pääajatus on tuottaa päätöksentekoa tukevaa informaatiota. Informaation tuottamisen kannalta keskeisiä ovat järjestelmän raportointiominaisuudet. Myös aiemmissa EIS-ominaisuuksia käsitelleissä tutkimuksissa tuodaan esiin järjestelmän raportointiominaisuudet (Carlisle ja Alameddine 1990, Carlson et al. 1992, Watson et al. 1992). Raportointiominaisuudet kohdassa tarkastellaan raporttien ja niiden esitystapojen monipuolisuutta, raporttien antamaa tukea analyysien tekemiselle, budjetoinnille ja ennusteille. Lisäksi tutkitaan ohjelmiston mahdollisuuksia tuottaa eri toimintomalleja koskevia yhteenveto- ja vertailuraportteja sekä ohjelmiston tarjoamia raporttien räätälöintimahdollisuuksia ja räätälöinnin tukea. Räätälöintimahdollisuuksia pidetään yleisesti tärkeänä EIS-järjestelmien ominaisuutena (Carlisle ja Alameddine 1990, Carlson et al. 1992, Watson et al. 1992). Arviointikehikon raportointiominaisuuksissa tarkastellaan myös ohjelmiston kykyä generoida poikkeamaraportteja.

EIS-järjestelmien osalta käsitellyissä lähteissä raportointia tuo laajasti esiin vain Watson et al. (1992). Toimintojohtamisjärjestelmät kuitenkin poikkeavat raportoinnin osalta luonteeltaan ja taustaltaan EIS-järjestelmistä. Toimintojohtamisen perustana oleva toimintolaskenta kuuluu johdon laskentatoimeen, jonka yksi pää tarkoitus on tuottaa informaatiota päätöksenteon tueksi ja kommunikoinnin avuksi. Tämä informaatio tuotetaan tyypillisesti raporttien muodossa (Drury 1994, s. 16). Näin ollen muodostetussa arviointikehikossa raportointiominaisuuksille annetaan suurempi painoarvo kuin aiemmissa EIS-ominaisuuksien luokitteluissa.

*Raporttien monipuolisuudella* tarkoitetaan järjestelmän tuottamia erilaisia valmisraportteja. Monipuolisuuden mittarina käytetään erilaisten raporttien lukumäärää. *Raporttien esitystapojen monipuolisuudella* arvioidaan vaihtoehtoisia tapoja il-

maista asioita raporteissa. Tässä kohdassa tarkastellaan järjestelmän kykyä käsitellä raporteissa taulukoita, tekstiä ja graafeja erikseen tai samanaikaisesti.

*Raporttien tuki analyyseille, budjetoinnille ja ennusteille* -kohdassa kuvataan subjektiivisesti raporttien informatiivisuutta ja tarkoituksenmukaisuutta. Ohjelmistosta saatavien raporttien tulisi tukea toimintojohtamisen edellyttämiä analyyseja. Samoin raporttien tulisi avustaa toimintopohjaisen budjetin laadinnassa sekä ennusteiden tekemisessä.

*Toimintomallien yhteenveto- ja vertailuraportit* tarkoittavat esimerkiksi konsernin eri osista laadittujen erillisten toimintomallien yhteenveto- ja vertailumahdollisuuksia. Johtamisen kannalta on tärkeää, että eri yksiköistä saadaan vaivattomasti yhteenvetotietoja ja että yksiköitä voidaan vertailla keskenään.

*Raporttien räätälöintimahdollisuudet ja räätälöinnin tuki* -kohdassa arvioidaan mahdollisuuksia räätälöidä ohjelmiston tuottamia raportteja. Räätälöinti tulisi voida suorittaa käyttäjän-, aihealueen- tai aikajänteen suhteen. Räätälöinnin tarkoituksena on tuottaa kaikille tahoille juuri oikeanlaista informaatiota ilman puutteellisuksia tai ylimäärää. Raporttien räätälöinnin tuella tarkoitetaan ohjelmiston antamaa opastusta räätälöityjä raportteja laadittaessa. Tämä ominaisuus on otettu tarkasteluun mukaan, koska saadakseen räätälöidyistä raporteista täyden hyödyn, on johtajan pystyttävä generoimaan nopeasti ja vaivattomasti haluamiaan raportteja. Jos raporttien generoiminen on kovin monimutkaista tai ohjelmisto ei anna siinä opastusta, jää koko generointiominaisuus todennäköisesti käyttämättä.

*Poikkeamaraportit* tarkoittavat raportteja, jotka ilmoittavat päätöksentekijälle, jos jokin seurattu asia ei ole ennalta määrätyissä rajoissa. Poikkeamaraportteja voidaan tuottaa esimerkiksi tietyn tuotteen valmistuskustannuksista.

#### 4.5.2 Järjestelmien tiedon analysointiominaisuudet

Johtamisen näkökulmasta tarkasteltuna tiedon analysointiominaisuudet ovat tärkeä osa toimintojohtamisjärjestelmiä. Aiemmissa EIS-järjestelmien tutkimuksissa tiedon analysointiominaisuudet on myös koettu tärkeiksi, sillä mukaanotetut analysointiominaisuudet esiintyvät myös niissä (Bergeron ja Raymond 1992, Watson et al. 1992, Carlson et al. 1992, Laska ja Paller 1992, Carlisle ja Alameddine 1990). Arviointikehikossa tarkastellaan seuraavia tiedon analysointiominaisuuksia: tietoon porautuminen, ennakoimattomat kyselyt sekä mitä-jos -analyysit.

EIS-tutkimuksissa esitetään muitakin kuin tässä käsiteltyjä tiedon analysointiominaisuuksia, kuten esimerkiksi kriittisten tekijöiden seuranta ja kuvioiden sanalliset selitykset (Carlisle ja Alameddine 1990), mutta näitä ei koettu toimintojohtamisjärjestelmissä tärkeiksi, joten ne on jätetty arviointikehikon ulkopuolelle.

*Tietoon porautuminen* eli drill-down -ominaisuus tarkoittaa mahdollisuutta päästä tutkimaan tiettyä tietoalkiota tarkemmin suoraan tietoalkion valitsemalla. Tämä edellyttää, että tieto on esitetty järjestelmässä hierarkisena rakenteena, jossa käyttäjä voi halutessaan porautua syvemmälle yksityiskohtaisempiin tietoihin. Toimintojohtamisjärjestelmissä esimerkkinä ovat toimia prosessitiedot, joista voidaan valitua prosessin osaa näpäyttämällä tarkastella tarkemmin jotain yksittäistä toimintoa.

*Ennakoimattomat kyselyt* eli ad hoc -kyselyt tukevat spontaanien ongelmien ratkaisua. Tässä tutkimuksessa ennakoimattomilla kyselyillä tarkoitetaan kyselyitä, joille ohjelmistossa ei ole valmista pohjaa, vaan käyttäjä voi tehdä niitä spontaanisti niin halutessaan.

*Mitä-jos -analyysilla* käyttäjä voi tarkastella vaihtoehtoisia toimintatapoja ja niiden seurauksia. Mitä-jos -ominaisuutta pidetään EIS-järjestelmien perusominaisuutena. Ominaisuus on otettu mukaan myös toimintojohtamisjärjestelmien arviointi-

kehikkoon, koska se on erityisen hyödyllinen yrityksen taloudellisia tietoja koske-  
vissa tarkasteluissa. Tämä johtuu siitä, että taloudelliset arvot ilmaistaan lukumää-  
räisillä mittareilla, jolloin mitä-jos -analyyseista saadaan hallittavuudeltaan riittävän  
yksinkertaisia. Esimerkiksi yrityksen strategiaa koskevat mitä-jos -analyysit ovat  
huomattavasti vaikeampia toteuttaa, koska syy-seuraussuhteita ei voida ilmaista  
yhtä selkeästi ja yksikäsitteisesti kuin esimerkiksi yrityksen myynnin lisääntymisen  
vaikutusta tulokseen. Mitä-jos -analyysit soveltuvat myös budjetointiin ja erilaisten  
tuotantovaihtoehtojen tarkasteluun. Arviointikehikon tässä kohdassa tutkitaan jär-  
jestelmien mahdollisuuksia erilaisten mitä-jos -analyysien tekemiseen.

#### **4.5.3 Järjestelmän tiedon käsittelyominaisuudet**

Toimintojohtamisjärjestelmät sisältävät hyvin suuren osan kaikesta yrityksessä ole-  
vasta tiedosta. Järjestelmän on pystyttävä hallitsemaan jollain tarkkuudella kaikkea  
yrityksen kustannustietoa ja suurta osaa myös muusta tiedosta. Tämä asettaa vaa-  
timuksia toimintojohtamisjärjestelmän tiedon käsittelyominaisuuksille. Tässä suh-  
teessa toimintojohtamisjärjestelmät ovat yhteneviä EIS-järjestelmien kanssa, joilla  
tiedon käsittely on myös useissa tutkimuksissa esitetty arvioitavaksi ominaisuu-  
deksi (Bergeron ja Raymond 1992, Watson et al. 1992, Carlson et al. 1992).

Kuten EIS-järjestelmät, toimintojohtamisjärjestelmätkin tulisi rakentaa osaksi yri-  
tyksen kokonaistietojärjestelmää. Tästä syystä yksi tärkeä järjestelmän ominaisuus  
on kyky suorittaa tiedonsiirrot eri tietojärjestelmän osien välillä (Bergeron ja Ray-  
mond 1992, Watson et al. 1992, Carlson et al. 1992, Laska ja Paller 1992). Erityi-  
sesti toimintojohtamisjärjestelmissä siirretään kustannustietoja edestakaisin järjes-  
telmien välillä. Ensin kustannustietoja kerätään muista järjestelmistä ja siirretään  
toimintojohtamisjärjestelmään, jossa esimerkiksi lasketaan toimintolaskennan mu-  
kaisia tuotteiden hintatietoja. Tämän jälkeen hintatiedot siirretään myynnin järjes-  
telmään markkinoinnin käytettäväksi.

Tiedon käsittely ominaisuuksista arviointikehikossa tarkastellaan siirtoja järjestelmien välillä ja yrityksen ulkoisen tiedon käyttöä. Näiden lisäksi arvioidaan vielä hakuja tietokantoihin ja tiedon päivityksiä.

*Tiedonsiirrot järjestelmien välillä* tarkoittavat ohjelmiston tukemia järjestelmien välisiä tiedonsiirtoja. Kohdassa tarkastellaan vaihtoehtoisia tiedonsiirtoformaatteja sekä siirtojen räätälöintejä ja yleistä toimivuutta. Tiedonsiirtoja tarkastellaan sekä muista järjestelmistä toimintojohtamisjärjestelmään että toimintojohtamisjärjestelmästä ulospäin. Räätälöidyillä siirroilla tarkoitetaan käyttäjän mahdollisuuksia vaikuttaa toimintojohtamisjärjestelmään siirrettävän tiedon muotoon.

Yleistä toimivuutta kuvaavana seikkana kiinnitetään huomiota epäonnistuneista siirroista ilmoittavien virheilmoitusten informatiivisuuteen. Tämä on otettu erillisenä esiin, koska tietoa siirretään tyypillisesti formaalisuudeltaan vaihtelevan tasoista lähteistä, jolloin virhetodennäköisyys on kohtuullisen suuri ja virheen paikantaminen on vaikeaa ilman selkeää ilmoitusta. Myös toimintojohtamisjärjestelmään saakka päässyt virheellinen tieto saattaa vääränä lähtötietona heikentää merkittävästi toimintojohtamisjärjestelmän tuottaman informaation luotettavuutta.

*Haut tietokantoihin* tarkoittavat toimintojohtamisjärjestelmän kykyä tehdä hakuja järjestelmän ulkopuolisiin tietokantoihin. Kohdassa arvioidaan haulle annettavaa tukea eli esimerkiksi standardi SQL:n (Standard Query Language) lisäksi olevia hakujen tekemistä helpottavia työkaluja. Huomiota kiinnitetään tietokantoja tuntemattoman henkilön mahdollisuuksiin käyttää hakuja itsenäisesti. Samoin tarkastellaan lukumääräisesti, mistä eri tietokannoista hakuja voi tehdä.

*Tiedon päivitykset* tarkoittavat toimintojohtamisjärjestelmän mahdollisuuksia päivittää järjestelmän sisältämiä tietoja. Päivitykset tapahtuvat pääasiassa toimintojohtamisjärjestelmän ulkopuolisista tiedonlähteistä. Toimintojohtamisen kannalta tietojen vaivaton päivittäminen on edellytys jatkuvan järjestelmän ylläpidolle. Esimer-



kiksi resurssien kulutustietoja joudutaan joissain tapauksissa muuttamaan kerran kuukaudessa, joten päivitysten on sujuttava turvallisesti ja automatisoituina.

*Ulkoisen tiedon käytöllä* ei toimintojohtamisjärjestelmissä ole yhtä merkittävää roolia kuin EIS-järjestelmissä, joissa ulkoisen tiedon käyttö sisältyy usein jo EIS:n määritelmään (Watson et al. 1992, Watson, Rainer ja Koh 1991). Toimintojohtamisjärjestelmissä ulkoisen tiedon käyttö rajoittuu esimerkiksi markkinahintojen hyväksikäyttöön.

#### **4.5.4 Toimintojohtamisjärjestelmien käyttäjäominaisuudet**

Käyttäjäominaisuuksilla tarkoitetaan ominaisuuksia, jotka vaikuttavat järjestelmän käyttömukavuuteen ja käytön helppouteen. Näiden ominaisuuksien tarkoitus on tehdä järjestelmän käyttämisestä mahdollisimman vaivatonta, jotta käyttäjä voi keskittyä toimintojen hallitsemisen sisällöllisiin kysymyksiin. Arviointikehikon käyttäjäominaisuudet-osa muodostuu seuraavista osista käyttöliittymä, grafiikka-ominaisuudet, käytön oppimisen helppous, käytön helppous. Näitä ominaisuuksia pidetään tärkeänä osana johtamisen tietojärjestelmiä (Carlisle ja Alameddine 1990, Laska ja Paller 1992, Bergeron ja Raymond 1992, Watson et al. 1992):

*Käyttöliittymällä* tarkoitetaan niitä järjestelmän ominaisuuksia, joilla määritetään käyttäjän ja järjestelmän välinen vuorovaikutus (Dos Santos et al. 1989). Useiden lähteiden mukaan DSS:n ja EIS:n menestyksekkäälle käytölle on hyvin olennaista, että sillä on helppokäyttöinen käyttöliittymä (Watson et al. 1992, Dos Santos et al. 1989, Carlson et al. 1992) Käyttöliittymällä on merkittävä vaikutus ohjelmiston käytön helppouteen. Käyttöliittymä kohdassa arvioidaan ohjelmiston tarjoamia näkymiä käsiteltävään tietoon sekä järjestelmän ohjauslaitteita. Ohjauslaitteilla tarkoitetaan fyysisiä välineitä, joilla käyttäjä ohjaa järjestelmää. Käytännössä ohjauslaitteina nykyisissä järjestelmissä toimivat näppäimistö ja hiiri. Näppäimistön käyttötarpeen tulisi olla EIS-ohjelmistoissa mahdollisimman pieni (Goldstein 1988). Tä-

män lisäksi tutkitaan valikkojen monipuolisuus, mahdollisuus käyttää ohjelmistoa käyttäjän valitsemilla pikanäppäinyhdistelmillä sekä työkalurivin tarkoituksenmukaisuus. Edellä mainittujen ominaisuuksien kohdalla kiinnitetään huomiota myös käyttäjän mahdollisuuksiin räätälöidä haluamansa kaltaisia valikoita, pikanäppäinyhdistelmiä tai työkalurivejä.

*Grafiikalla* tarkoitetaan järjestelmän kykyä esittää informaatiota erilaisina graafeina. Tarkastelussa kiinnitetään huomiota järjestelmän graafien määrään ja laatuun. Määrällä tarkoitetaan kaikkea erilaista informaatiota, joka voidaan esittää graafisesti. Laadulla taas tarkoitetaan graafien havainnollisuutta ja informatiivisuutta eli esimerkiksi otsikoiden sisältöä, graafin skaalautuvuutta ja akseleiden otsikointia. Näiden lisäksi arvioidaan graafien monipuolisuutta eli esimerkiksi asian esittämistä pylväs-, sektori- tai kolmiulotteisina graafeina. Graafien hyödyntämisen kannalta olennaista on myös käyttäjän mahdollisuudet luoda kokonaan uusia graafeja tai muokata haluamiaan graafeja esimerkiksi värejä ja otsikointia vaihtamalla. Itse grafiikkaominaisuuksien ohella kohdassa tarkastellaan järjestelmän tarjoamia mahdollisuuksia tulostaa graafeja tai siirtää niitä muiden ohjelmistojen käyttöön.

*Käytön oppimisen helppoudella* tarkoitetaan sitä, kuinka helposti ja nopeasti käyttäjä oppii itsenäisesti käyttämään toimintojohtamisjärjestelmää. Oppimisen helppoutta arvioidaan ohjelmiston manuaaleja ja On-Line help -toimintoa analysoimalla, edellyttäen, että ohjelmisto sisältää On-Line helpin. Oppimisen nopeuden kriteerinä käytetään järjestelmän käyttäjien omakohtaisia arvioita. Järjestelmän käyttö katsotaan opituksi, kun käyttäjä osaa itsenäisesti käyttää järjestelmän avulla toimintojohtamiseen liittyviä perusasioita (resursseja, toimintoja ja laskentakohteita) sekä hallitsee käyttöliittymän perustoiminnot.

*Käytön helppoudella* tarkoitetaan sitä, kuinka vaivaton järjestelmää on käyttää sen jälkeen, kun käyttö on kertaalleen opittu. Käytön helppoutta arvioidaan käyttäjien kokemusten perusteella. Vertailukohdaksi asetetaan kullekin käyttäjälle ennestään

tuttu tekstinkäsittelyohjelmisto. Käyttäjäkokemusten lisäksi kiinnitetään huomiota järjestelmän käyttökielen. Kielellä on merkitystä toimintojohtamisen terminologian hallitsemisen kannalta ja järjestelmän käyttöönottokynnyksen alentamisessa koko organisaatiota ajatellen.

#### **4.5.5 Järjestelmän kustannukset**

Toimintojohtamisjärjestelmän tarkoitus on ensin määrittää ja sen jälkeen pyrkiä alentamaan yrityksessä syntyviä kustannuksia. Tästä syystä on syytä tarkastella erikseen myös itse järjestelmästä aiheutuvia kustannuksia. Tämän työn puitteissa ei oteta kantaa toimintojohtamisjärjestelmän kokonaiskustannuksiin, vaan keskitytään pelkästään toimintojohtamisohjelmiston kustannuksiin.

Toimintojohtamisjärjestelmästä aiheutuu ohjelmistohankinnan lisäksi kustannuksia mahdollisen ulkopuolisen konsultoinnin käytöstä, oman henkilöstön projektiin käyttämästä ajasta sekä mahdollisista laitteisto- ja käyttöjärjestelmähankinnoista. Toimintojohtamisen käyttöönotto tapahtuu tyypillisesti yrityksen sisäisenä projektina, joten projektin palkkakustannukset saattavat nousta kohtuullisen suuriksi. Vaihtoehtoisesti yritys voi käyttää ulkopuolista konsulttia, jolloin tästä aiheutuu kuluja. Lopputuloksen pysyvyyden kannalta olisi järkevää sitouttaa omaa henkilöstöä projektiin, jotta toimintojohtamisaaminen ei häviä yrityksestä konsultin poistuttua.

Laitteisto- ja käyttöjärjestelmähankinnoista voidaan sanoa, että toimintojohtamisjärjestelmien asettamat vaatimukset tulevat yleensä täytetyiksi jo olemassaolevilla koneilla. Jos yritykseen ollaan hankkimassa verkossa toimivaa toimintojohtamisjärjestelmää saattaa kustannuksia syntyä mahdollisista verkon parantamistoimenpiteistä, mutta tämäkin on hyvin epätodennäköistä. Yrityksen tulisi järjestelmää hankkiessaan selvittää sen asettamat vaatimukset, jotta turhilta lisäinvestoinneilta vältyttäisiin.

Toimintojohtamisjärjestelmän kustannuksia arvioivassa osassa kustannuksiin on sisällytetty perusohjelmistolisenssin hinta ja ohjelmistotoimittajan tuen kustannukset (Carlisle ja Alameddine 1990, Carlson et al. 1992). Toimintojohtamisjärjestelmää käyttävät yrityksessä normaalisti useat ihmiset, joten edellisten ominaisuuksien lisäksi arviointikehikkoon on otettu mukaan monen käyttäjän lisenssit. Vertailtaviksi määräksi valittiin viisi ja kymmenen lisenssiä.

Koska normaalisti ohjelmistotoimittaja antaa koulutusta ohjelmiston käytössä, on mukaan otettu myös koulutuksen erilliset kustannukset. Myös EIS-järjestelmien kohdalla on käsitelty järjestelmäkoulutusta (Watson et al. 1992). Kustannuksia aiheuttavana yksikkönä on käytetty yhden koulutuspäivän hintaa, koska koulutustarve vaihtelee tapauskohtaisesti. Joissakin tapauksissa ohjelmistotoimittaja toimii tuotteeseensa liittyvän koulutuksen ohella myös toimintolaskentaprojektin konsulttina, mikä vaikuttaa merkittävästi ohjelmistotoimittajalta ostettavien palvelujen määrään. Toisaalta koulutustarve saattaa vaihdella myös ohjelmistosta riippuvasti, mistä ei saada käsitystä yhden koulutuspäivän kustannuksia vertaamalla. Tästä syystä on kunkin ohjelmiston osalta pyritty subjektiivisesti arvioimaan ohjelmiston vaatima koulutustarve.

*Perusohjelmistolisenssin hinnalla* määritellään toimintojohtamisohjelmiston yhden käyttäjälisenssin hinta. Nämä kustannukset sisältävät käyttöohjeet ja muun vastaavanlaisen dokumentaation kustannukset, mutta eivät järjestelmän implementointiin liittyviä kustannuksia.

*Monen käyttäjän lisenssien* kustannuksilla tarkoitetaan ensimmäisen käyttäjälisenssin ohella hankittujen lisenssien kustannuksia. Tyypillisesti näiden lisenssien hinnat laskevat määrän kasvaessa. Tästä syystä monen käyttäjän lisenssien hintoja vertaillaan viidellä ja kymmenellä lisenssillä. Tarkastelussa oletetaan, että kaikki lisenssit ostetaan samalla kertaa.

*Ohjelmistotoimittajan tuen* kustannukset tarkoittavat ohjelmiston vuosittaista ylläpito- ja käyttömaksua. Tyypillisesti ylläpitomaksun suuruus määritellään prosentteina ostetun ohjelmistopakettin hinnasta. Ylläpitomaksua vastaan toimittaja sitoutuu esimerkiksi järjestämään puhelintukea ohjelmiston käyttäjille. Normaalisti ylläpitomaksu sisältää myös oikeuden ostetun ohjelmiston päivitysversioihin.

*Koulutuksen kustannuksilla* ilmaistaan ohjelmistotoimittajan antamasta koulutuksesta aiheutuvat kustannukset. Edellä selostetun mukaisesti käytetään vertailukohdaksi yhden koulutuspäivän hintaa. Tämän lisäksi arvioidaan kustakin ohjelmistosta riippuva koulutustarve.

#### **4.5.6 Tekniset ominaisuudet**

Johtamisen tietojärjestelmien teknisillä ominaisuuksilla ei ole suurta merkitystä järjestelmän hyödyllisyyden tai käyttäjän kannalta. EIS:ää käsittelevissä tutkimuksissa Teknisillä ominaisuuksilla on enemmänkin merkitystä järjestelmän yritykseen implementointivaiheessa.

*Avoin tai suljettu arkkitehtuuri* vaikuttaa tiedon siirtoihin toimintojohtamisjärjestelmän ja muiden tietojärjestelmien välillä. Suljettu arkkitehtuuri kontrolloi järjestelmään sen ulkopuolelta siirrettävää tietoa. Tämä parantaa tietokantojen turvaamista ja eheyttä, mutta voi joissakin tapauksissa vaikeuttaa erilaisissa tietokannoissa käsiteltävän tiedon siirtoa toimintojohtamisjärjestelmään. Avoin arkkitehtuuri ei kontrolloi hajautettujen tietokantojen tiedon eheyttä, mutta mahdollistaa vapaammat tiedon haut erilaisista laitteisto- ja käyttöjärjestelmäympäristöistä.

*Yksittäiskoneessa toimiva ohjelmisto* tarkoittaa ohjelmistoa, joka on asennettava jokaiseen koneeseen erikseen. Kukin tällainen ohjelmistolisenssi toimii omana erillisenä ohjelmanaan eikä sitä voi käyttää verkon yli. Yksittäiskoneissa toimivilla ohjelmistoilla ei voi käsitellä samanaikaisesti samaa lähdeaineistoa.

*Verkossa toimiva ohjelmisto* tarkoittaa ohjelmistoa, joka toimii verkossa ja hakee tarvitsemansa tiedot verkosta, mutta jossa itse tiedon prosessointi tapahtuu kullakin koneella erikseen. Verkossa toimivalla ohjelmistolisensseillä voidaan käsitellä yhtäaikaaisesti samaa lähdeaineistoa.

*Verkko-ohjelmisto* tarkoittaa ohjelmistoa, joka perustuu Asiakas/Palvelin (Client/Server)-arkkitehtuuriin. Näissä ohjelmistoissa tiedon käsittely tapahtuu verkkopalvelimella, johon käyttäjän pääte toimii käyttöliittymänä.

Johdon tietojärjestelmissä suojausominaisuudet ovat olennaisia. Tämä johtuu pitkälti siitä, että johtajat käsittelevät yrityksen kannalta kriittistä informaatiota, jonka on pysyttävä sivullisilta salassa. Suojausominaisuudet otetaan esiin useissa lähteissä (Carlisle ja Alameddine 1990, Bergeron ja Raymond 1992, Laska ja Paller 1992, Carlson et al. 1992) ja erityisesti niitä painottavat Watson et al. (1992). Suojausominaisuudet on jaoteltu järjestelmän- sekä näkymien- ja funktioiden suojausominaisuuksiin.

*Järjestelmän suojausominaisuuksilla* tarkoitetaan järjestelmän sisältämien yritystä koskevien tietojen suojaamista järjestelmän ulkopuolisilta tahoilta. Tieto on voitu suojata esimerkiksi kultakin käyttäjältä vaadittavalla käyttäjätunnuksella ja salasanalla sekä verkon yli tapahtuvan tiedonsiirron koodauksella.

*Näkymien- ja funktioiden suojausominaisuuksilla* tarkoitetaan järjestelmän käyttöön oikeutettujen käyttäjien kontrollointimahdollisuuksia. Käyttäjille voidaan antaa erilaisia oikeuksia käsitellä järjestelmän tarjoamaa tietoa. Tietyllä turvallisuustasolla olevat käyttäjät eivät esimerkiksi voi katsella henkilöstön palkkatietoja esittäviä näyttöjä tai graafeja. Samoin voidaan antaa joillekin henkilöille luku-, mutta ei kirjoitusoikeuksia ohjelmiston tietoihin. Luku- ja kirjoitusoikeuksia voidaan rajata vielä niin, että käyttäjällä on oikeus käyttää vain joitakin ohjelmiston sisältämiä

funktioita. Tässä kohdassa arvioidaan, kuinka monipuolisia mahdollisuuksia ohjelmisto tarjoaa edellä esitetyn kaltaisten suojausten toteuttamiseen.

*Laitteisto- ja käyttöjärjestelmävaatimukset* määrittelevät järjestelmän asettamat vaatimukset tietokonelaitteistolle ja käyttöympäristölle. Laitteistovaatimuksissa esitetään keskusmuistin, kovalevytilan ja prosessorin tarve. Normaalisti laitteistovaatimukset eivät ole ongelma järjestelmää valittaessa, koska yleensä toimintojohtamisjärjestelmät toimivat tavanomaisissa, yrityksessä olemassa olevissa laitteissa. Tästä syystä laitteistovaatimuksille ei anneta arviointikehikossa suurta painoarvoa.

Käyttöjärjestelmänä ohjelmistoilla on nykyisin yleensä Windows 3.xx, Windows -95 tai Windows NT. Käyttöjärjestelmänä saattaa olla myös Macintosh-ympäristö. Tässä kohdassa tarkastellaan ohjelmistojen toimivuutta eri ympäristöissä.

## 5 TOIMINTOJOHTAMISOHJELMISTOJEN SIJOITTAMINEN ARVIOINTIKEHIKKOON

Tässä luvussa analysoidaan toimintojohtamiseen käytettäviä ohjelmistoja ja niiden ominaisuuksia. Analysoinnin perustana käytetään muodostettua arviointikehikkoa. Ominaisuuksia on käsitelty arviointikehikon mukaisessa järjestyksessä. Markkinoilla on useita toimintolaskentaan ja toimintojohtamiseen tarkoitettuja ohjelmistoja. Jos toimintolaskentaa ja -johtamista aiotaan toteuttaa yrityksessä pysyvästi eikä pelkkänä kertaluonteisena laskelmana, on järkevää hankkia tarkoitukseen suunniteltu työkalu. Yleensä toimintolaskentaa toteutetaan taulukkolaskentapohjaisena, räätälöidyllä pienohjelmistoilla tai varsinaisilla toimintojohtamisen työkaluilla.

Toimintojohtamisohjelmistot voidaan jakaa vapaasti saataviin ja konsultti- tai tilintarkastustoimistojen myymiin ohjelmistoihin (Albright 1995). Näiden olennainen ero on, että konsulttitoimistojen ohjelmistoja ei voi vapaasti ostaa, vaan ne toimitetaan konsultoinnin yhteydessä, jolloin ostaja sitoutuu ohjelmiston ja konsulttiyrityksen käyttöön. Tällaisia konsulttiyritysten toimittamia ohjelmistoja ovat esimerkiksi Price Waterhousen Activa, KPMG:n Profit Manager ja Deloitte & Touchen TR/ACM (Albright 1995). Tämän tutkimuksen ulkopuolelle on rajattu konsulttiyritysten toimittamat ohjelmistot.

Tutkimuksessa on rajauduttu tarkastelemaan Suomessa käytettäviä ohjelmistoja: oululaista Quality Production & Researchin (Q.P.R.) tuotetta CostControlia, amerikkalaista ABC Technologies yrityksen tuotetta EasyABCPlus:aa sekä ruotsalaista ProDaCapoa. ABC Technologies yhtiöllä on markkinoilla kolme versiota EasyABC:stä: EasyABCQuick, EasyABCPlus sekä EasyABC Oros. EasyABCQuick on tarkoitettu lähinnä opiskelukäyttöön. Se sisältää tiedonsiirto-ominaisuuksia lukuunottamatta samat ominaisuudet kuin EasyABCPlus. Quick versiossa on myös rajoitettu hallittavaa tietomäärää. EasyABCPlus on yksittäiskäyttäjän hyötykäyt-



töön tarkoitettu ohjelmisto. EasyABC Oros on Client/Server-verkoissa toimiva toimintojohtamista tukeva järjestelmä. EasyABC Oroksen mallinnusmoduuli on vastaava kuin EasyABCPlus. Mallinnusmoduulin lisäksi Oros sisältää seuraavat moduulit:

*Links Engine*, jolla liitetään ABCPlus ODBC (Open DataBase Connectivity) yhteensopiviin tietolähteisiin,

*Oros Reports*, jolla generoidaan raportteja,

*Oros EIS Designer*, jolla siirretään tietoa toimintomallista OLAP-tietokantaan,

*Oros Connect*, jolla linkitetään ABCPlus Excelliin,

*Oros Surveys*, joka tukee tiedon keräämistä toimintolaskentaa varten,

*Oros Budgets*, jolla rakennetaan toimintoperusteinen budjetti ja

*Oros Yield*, joka tukee toimintoperusteisen tiedon analysointia.

Tässä tutkimuksessa käsitellään ABC Technologies yhtiön tuotteista EasyABCPlus:aa, koska ensinnäkin se muodostaa myös Oroksen keskeisimmän osan ja toisaalta sen hintaluokka sijoittuu muiden ohjelmistotalojen tuotteiden kanssa samalle tasolle. Joidenkin käsiteltyjen ominaisuuksien kohdalla viitataan Orokseen, jos siitä löytyy vastaava ominaisuus ja se puuttuu tutkituista ohjelmistoista. Jatkossa EasyABCPlus:sta käytetään nimitystä EasyABC.

Tutkituista ohjelmistoista CostControl ja EasyABC ovat saatavilla Suomessa ja ProDaCapo on käytössä joillakin merkittävillä referenssiyrityksillä Suomessa, vaikka suomalaista maahantuojaa ei olekaan. Suomessa ei tekijän tietojen mukaan ole käytössä muita vapaasti hankittavia toimintoperusteista ajattelua tukevia ohjelmistoja kuin tutkimuksessa käsitellyt. Tosin esimerkiksi SIMPROCESS prosessien simulointiohjelmistolla voidaan simuloida toimintolaskentasovelluksia, mutta sitä ei voida pitää toimintojohtamisohjelmistona (Simprocess 1996, s. 175-190). Näiden lisäksi joillakin yrityksillä, joihin tutkimuksen puitteissa oltiin yhteydessä, on käy-

tössä itse kehitetty toimintolaskentasovellus tai ohjelmisto, jota Suomessa on myyty yksittäiskappaleita.

### 5.1 Tiedonhankinta ja tutkimusmenetelmä

Tutkimus perustuu tekijän kokemuksiin tutkituista ohjelmistoista sekä käyttäjien keskuudessa tehtyyn tapaustutkimukseen. Tämän ohella arviointikehikkoon sijoitetuista ohjelmistoista on hankittu tietoa ohjelmistotoimittajilta. Tutkimuksen tietolähteenä on ollut käytettävissä valmistajan haastattelut, ohjelmistolisenssit, ohjelmistomanuaalit sekä esittelymateriaalia CostControlista ja EasyABC:stä. ProDaCapon osalta tutkimus perustuu käyttäjäorganisaation henkilöiden haastatteluihin, ohjelmistotoimittajan haastatteluihin, ohjelmistoesitteisiin, ohjelmiston esittelyversioon sekä valmistajan internet-sivuihin. Vaikka ProDaCaposta ei ollut käytettävissä varsinaista ohjelmistolisenssiä, voidaan esittelyversiota pitää hyvin informatiivisena. Esittelyversiota täydensivät käyttäjien sekä valmistajan haastattelut ja valmistajan monipuoliset internet-sivut, joten tuloksia voidaan pitää vertailukelpoisina muiden ohjelmistojen kanssa.

Tehty tapaustutkimus perustui muodostettuun arviointikehikkoon. Tapaustutkimuksella on pyritty selvittämään ensinnäkin mitä ohjelmistoja toimintolaskennassa ja -johtamisessa käytetään. Tämän ohella kysyttiin käyttäjien mielipiteitä heidän käyttämistään ohjelmistoista ja niiden ominaisuuksista. Tutkimuksen perustana ovat olleet arviointikehikossa käsitellyt ominaisuudet. Ominaisuuksista on selvitetty myös, mitä ominaisuuksia ohjelmistoista käytettiin. Tämän lisäksi selvitettiin toimintojohtamisohjelmistojen käyttäjien sijoittumista organisaatioissa sekä käyttäjäorganisaatioiden kokoa ja toimialaa.

Tapaustutkimuksen tekemistä varten oltiin yhteydessä noin 30:een organisaatioon. Yritysten nimet saatiin ohjelmistotoimittajien referenssilistoilta tai asiantuntijoilta. Organisaatioista, joihin oltiin yhteydessä, kymmenellä oli toimintojohtamisohjelmis-

to ja sitä käytettiin aktiivisesti. Näistä organisaatioista seitsemän suostui osallistumaan tutkimukseen. Tyypillisesti organisaatioissa, joihin toimintojohtamisohjelmisto oli installoitu, mutta sitä ei käytetty aktiivisesti, oli ohjelmisto ollut pilottiprojektin tukena. Kaikissa näissä organisaatioissa toimintolaskentaa aiottiin jatkossa soveltaa ja toimintojohtamisohjelmistoa käyttää, mutta ongelmaksi koettiin ajan riittämättömyys kehitysprojekteihin. Toimintolaskentaprojekti oli koettu vaikeaksi toteuttaa oman toimen ohella.

Tämän tutkimuksen kohteiksi valittiin organisaatiot, joissa toimintojohtamisohjelmistoa käytettiin aktiivisesti ja jotka suostuivat yhteistyöhön. Näistä organisaatioista toimintojohtamisohjelmistojen käyttäjinä otettiin mukaan sellaiset, jotka olivat käyttäneet ohjelmistoa aktiivisesti. Näillä valintaperusteilla oli tarkoitus kompensoida tutkimuksen pientä otoskokoja. Valintakriteerit täyttäviä organisaatioita tutkimuksessa käsitellään mainittua seitsemää.

Tapaustutkimus suoritettiin haastattelemalla kaikkia käyttäjiä joko puhelimitse tai henkilökohtaisesti. Ohjelmistojen käyttöön tutustuttiin myös vieraillemalla yhdessä kutakin ohjelmistoa käyttäneessä organisaatiossa. Yhdessä organisaatiossa vierailtiin hyvinkin tiiviisti, jotta saatiin taustatietoa toimintolaskennan käytännön toteutuksesta ohjelmiston avulla.

Näiden tutkimustapojen lisäksi käyttäjille postitettiin kyselylomake joko perinteisenä postina tai sähköpostina. Kyselylomakkeen toimittamisen joko sähköisesti tai paperilla ei katsottu vaikuttavan vastaajiin, koska vastaajiin oltiin puhelimitse henkilökohtaisesti yhteydessä ennen lomakkeen lähettämistä. Tällöin tiedusteltiin, missä muodossa he halusivat saada kyselyn. Tämän lisäksi sähköpostitse lähetettyihin kyselyihin sisällytettiin mahdollisuus vastata paperille ja palauttaa vastaus faxilla tai perinteisenä postina. Tässä yhteydessä voidaan mainita, että kaikki vastaajat, joilla oli sähköposti käytettävissä, halusivat saada kyselyn sähköisessä muodossa.

Ohjelmistojen käyttäjäorganisaatioita etsittäessä ilmeni, että ohjelmistoja ei käytetä pienissä yrityksissä. Samoin selvisi, että ohjelmistoja käytettiin yleisimmin metalliteollisuudessa. Tämä ei sinänsä ole yllätys, sillä suomalainen metalliteollisuus on perinteisesti ollut edistyksellistä ja uusia ajatuksia toteuttavaa. Tämä on nähtävissä esimerkiksi Metalliteollisuuden kustannuksen teknisistä tiedotteista. Yrityksessä käytettävällä ohjelmistolla ei tämän tutkimuksen perusteella ole yhteyttä yrityksen kokoon, tosin tutkimuksen otoskoko on hyvin pieni. Taulukossa 3 on esitetty tutkimukseen osallistuneiden case-yritysten taustatiedot. Taustatietojen jaotteluna on käytetty liitteessä 1 esitetyn kyselylomakkeen jaottelua.

<i>Yritys</i>	<i>Toimiala</i>	<i>Liikevaihto (Mmk)</i>	<i>Henkilöstö</i>	<i>Käytetty ohjelmisto</i>
A	metalliteollisuus	0-100	0-200	CostControl
B	metalliteollisuus	101-300	0-200	CostControl
C	palveluala	801-1400	1001-3000	CostControl
D	muu teollisuus	301-800	501 - 1000	EasyABC
E	palveluala	801-1400	1001- 3000	EasyABC
F	palveluala	101-300	501-1000	EasyABC
G	metalliteollisuus	yli 1400	yli 3000	ProDaCapo

*Taulukko 3. Case-yritysten taustatiedot*

Käyttäjien toimenkuvat keskittyivät taloushallintoon. Tyypillisin tehtävänimike oli business controller tai controller. Muidenkin käyttäjien tehtävänimikkeet viittasivat organisaation tai taloushallinnon kehitystehtäviin. Vain yhden käyttäjän toimenkuva oli talouspäällikkö, mitä voidaan pitää johtavana asema. Kyseinen käyttäjäkin oli tosin käyttänyt ohjelmistoa pääasiassa aiemmassa toimessaan taloushallinnon kehittäjänä. Tutkimuksen perusteella voidaan havaita, että vaikka toimintojohtamisjärjestelmät on tarkoitettu tukemaan yrityksen päätöksentekoa, käyttäjät eivät kuulu yrityksen johtoon. Tähän voidaan pitää syynä sitä, että toimintoperusteinen ajattelu on yrityksissä niin uusi asia, että se on vielä enemmänkin kehitysprojekti kuin käytännön johtamisen työkalu.

Henkilöiden mielipiteitä eri ominaisuuksista kysyttiin asteikolla 1 - 5, jossa yksi tarkoitti erittäin huonoa ja viisi erittäin hyvää arvosanaa.

## **5.2 Toimintojohtamisominaisuuksien sijoittuminen arvointikehikossa**

Tässä kappaleessa sijoitetaan tutkitut ohjelmistot toimintojohtamisjärjestelmien arvointikehikon toimintojohtamisominaisuuksia käsittelevään osaan, joka muodostettiin luvussa 4. Aiemmin kappaleessa 4.1 määriteltiin toimintojohtamisjärjestelmä ja toimintojohtamisohjelmisto. Määritelmien mukaisesti jaoteltuna on tutkittu toimintojohtamisohjelmistoja, jotka muodostavat keskeisen osan toimintojohtamisjärjestelmää. Toimintojohtamisohjelmistoja on tutkittu, koska varsinainen toimintojohtamisjärjestelmä on aina kunkin yrityksen yksilöllinen ratkaisu, johon vaikuttavat kaikki yrityksessä käytössä olevat tietojärjestelmät. Jatkossa käytetään termiä toimintojohtamisohjelmisto myös toimintojohtamisohjelmiston ja -järjestelmän välistä rajapintaa käsittelevien ominaisuuksien yhteydessä.

### **5.2.1 Toimintolaskennan perusajatukset**

#### *Resurssien tukiominaisuudet*

CostControlissa resurssit voidaan jaotella ihmis-, kone-, varasto-, kiinteistö- ja muihin resursseihin. ProDaCapossa resurssit voidaan vastaavasti jaotella ihmis-, kone- ja muihin resursseihin. Tämä helpottaa suuren resurssimäärän hallintaa. Toinen resurssien hallitsemista helpottava seikka molemmissa ohjelmistoissa on mahdollisuus luokitella resurssit eri osastoihin. Tällöin saadaan esimerkiksi eri tehtaissa olevat koneet omiksi osastoikseen. Toisaalta CostControlista ja ProDaCaposta puuttuu mahdollisuus muodostaa resursseista hierarkinen rakenne. Tämän ohella puutteena on, että resurssin kustannuksia ei voida kohdistaa toiselle resurssille. EasyABC:ssä on mahdollista muodostaa resursseista hierarkinen rakenne. Siinä hierarkia luodaan käyttäen kolmea tasoa: center, account ja cost element. Näistä kah-

della ensimmäisellä voi olla vielä alatasoja. EasyABC sisältää myös mahdollisuuden kohdistaa kustannuksia resurssilta toiselle.

Resurssien osalta CostControl ja ProDaCapo poikkeavat toimintolaskenta-ajattelusta, jossa toimintolaskentaketju muodostuu kolmesta osasta: resurssit, toiminnot ja laskentakohteet. Ohjelmissa on näiden lisäksi vielä ennen resursseja yksi moduuli. CostControlissa tämä on nimetty 'kustannustekijöiksi' ja ProDaCapossa moduuli on 'department costs'. Tyypillisesti tuloslaskelman tilit toimivat tällaisina kustannustekijöinä tai osaston kustannuksina. Kustannustilien erottaminen resursseista selkeyttää kustannusten muodostamista, mutta toisaalta tuo laskentaan yhden vaiheen lisää.

CostControlissa ja ProDaCapossa voidaan resurssiin liittää perustietoina pidettävien nimen ja tunnuksen lisäksi sanallinen kuvaus ja tieto vastuuhenkilöstä. EasyABC:ssä resursseihin voidaan perustietojen lisäksi liittää hyvin monipuolisia ominaisuustietoja (properties).

CostControlissa resursseille voidaan laskea pääomakustannuksia. Pääomakustannusten laskentaa varten ohjelmaan syötetään korkoprosentti, hankinta-aika, poisto-aika ja resurssin hankintahinta. Pääomakustannusten huomioimismahdollisuus puuttuu EasyABC:stä ja ProDaCaposta.

CostControlissa resurssien luominen onnistuu vaivattomasti ja niiden hallintakin on selkeää. Resurssien käyttöä vaikeuttaa se, että käyttäjä ei saa samanaikaisesti näytölle eri kategorioihin luokittelemiaan resursseja, kuten esimerkiksi koneet ja ihmiset, vaan kerralla voi tarkastella vain yhtä resurssityyppiä. EasyABC:ssä ja ProDaCapossa resurssit osiota ei ole toteutettu yhtä visuaalisesti kuin CostControlissa, mutta kylläkin hyvin selkeästi. Käyttäjien mielestä kaikki ohjelmistot tukivat hyvin resurssien muodostamista ja hallintaa, sillä ohjelmistosta riippumatta arvosanaksi annettiin neljä.

### *Resurssikulutusmalli*

Kaikki ohjelmistot tukevat resurssikulutusajatusta. Resurssikulutusmallin tukena ohjelmistot ilmaisevat resurssien käytetyn ja vapaan kapasiteetin. CostControlissa tätä voidaan tarkastella erillisenä raporttina tai graafina. Käyttäjien mielestä resurssikulutusmalli-ominaisuudet olivat ohjelmistoissa heikolla tasolla. ProDaCapo sai arvosanan huono (2) ja muutkin ohjelmistot vain neutraalin arvosanan (3). Jotkut käyttäjät eivät olleet käyttäneet resurssikulutusmalliominaisuuksia.

### *Toiminnot-osan tukiominaisuudet*

Kaikissa ohjelmissa toiminnot-osa toteutettu samoin periaattein kuin resurssiosakin. Kukin toiminto voidaan yksilöidä tunnukseksi ja nimellä. CostControlissa ja ProDaCapossa kuhunkin toimintoon voidaan liittää sanallinen kuvaus, mikä puuttuu CostControlin resurssiosasta. EasyABC:ssä ei varsinaista kuvausta voi antaa, mutta property-osioon voi valita monipuolisesti lisätietoja kuten resurssiosassakin. Kaikissa ohjelmissa toiminnoista voidaan muodostaa hierarkkinen rakenne, jossa käyttäjä voi valita ne toiminnot, joiden kustannukset kumuloituvat toimintohierarkiassa. CostControlissa hierarkiassa toimii drag-and-drop ominaisuus eli käyttäjä voi vaihtaa toiminnon paikkaa hierarkiassa tarttumalla siihen ja siirtämällä sitä.

CostControlissa ei ole mahdollisuutta valita, onko toiminto lisäarvoa tuottava vai tuottamaton. ProDaCapossa valinta on mahdollinen ja EasyABC:ssä valinta on toteutettu attribuuttien (attributes) avulla. EasyABC:ssä attribuutteja voi liittää mihin tahansa resurssiin, toimintoon tai laskentakohteeseen. ProDaCapossa attribuutteja voi liittää toimintoihin ja laskentakohteisiin. Merkitsemällä näitä objekteja samalla attribuutilla voi käyttäjän muodostaa mielivaltaisen määrän haluamiaan kokonaisuuksia. EasyABC:ssä on neljänlaisia attribuutteja: hierarkia (center), teksti (text), numeerisia (numeric) ja laskennallisia (calculated). Hierarkia-attribuuteilla muodostetaan muista attribuuteista hierarkia. Teksti-attribuuteilla ilmaistaan tekstitietoa, kuten esimerkiksi juuri lisäarvoa tuottavat toiminnot. Numeerisia attribuutteja käytetään esimerkiksi suorituksen mittaamisessa ja laskennallisia attribuutteja las-

kennallisten ajureiden yhteydessä. Myös ProDaCapossa toimintoihin voi liittää attribuutteja.

CostControlista attribuuttien kaltainen ominaisuus puuttuu. Osan attribuuttien ominaisuuksista siinä voi korvata muodostamalla toiminnoista toimintokokonaisuuksia, joista käytetään termiä 'osamalli'. Osamalli on toimintohierarkiasta riippumaton ja käyttäjä voi valita siihen haluamiaan toimintoja. Sama toiminto voi myös kuulua useaan osamalliin. Osamallin avulla käyttäjä voi keinotekoisesti määrittellä lisäarvoa tuottavat ja lisäarvoa tuottamattomat toiminnot vastaavien osamallien avulla. Osamallia vastaava toiminto puuttuu EasyABC:stä, jossa toiminnot voi järjestää vain hierarkian mukaisesti. Myös ProDaCapossa on CostControlin osamallia vastaava ominaisuus, sillä siinä toimintoja voi vapaasti ryhmitellä hierarkiasta riippumatta. ProDaCapossa toimintoja muodostettaessa käyttäjä voi valita toiminnon erillisestä toimintokirjastosta (activity dictionary) tai perustaa kokonaan uuden toiminnon. Toimintolaskennan teorian mukaisesti CostControlissa toiminnot voidaan jaotella laskentakohdetasojen mukaan. Laskentakohdetasoa on tosin käytävissä vain neljä eikä viisi, kuten muun muassa Drury (1994, s. 281) esittää. EasyABC:ssä ja ProDaCapossa tällaista mahdollisuutta ei ole. Kaikissa ohjelmistoissa toiminto voidaan kohdistaa tukitoiminnoksi toiselle toiminnolle.

Käyttäjien mielipiteet ohjelmistojen tuesta toiminnoille vaihtelivat. CostControlin antamaa tukea arvioitiin arvosanalla kolme. EasyABC sai käyttäjiltään arvosanoja hyvästä erittäin hyvään (4-5). Myös ProDaCapo sai arvosanan hyvä (4).

#### *Laskentakohde-ominaisuudet*

Perustietoina annettavien tunnuksen ja nimen lisäksi CostControlin laskentakohdeosa sisältää kentät laskentakohteen sanalliselle kuvaukselle ja hinnalle. Näiden lisäksi ProDaCapo ilmaisee erikseen laskentakohteen jäljitetyt (traced) ja jaetut (allocated) kustannukset. Kuten toimintolaskentaketjun muissakin osioissa EasyABC:ssä sanallinen kuvaus on korvattu property-osiolla. Vakiohinnan lisäksi käyt-

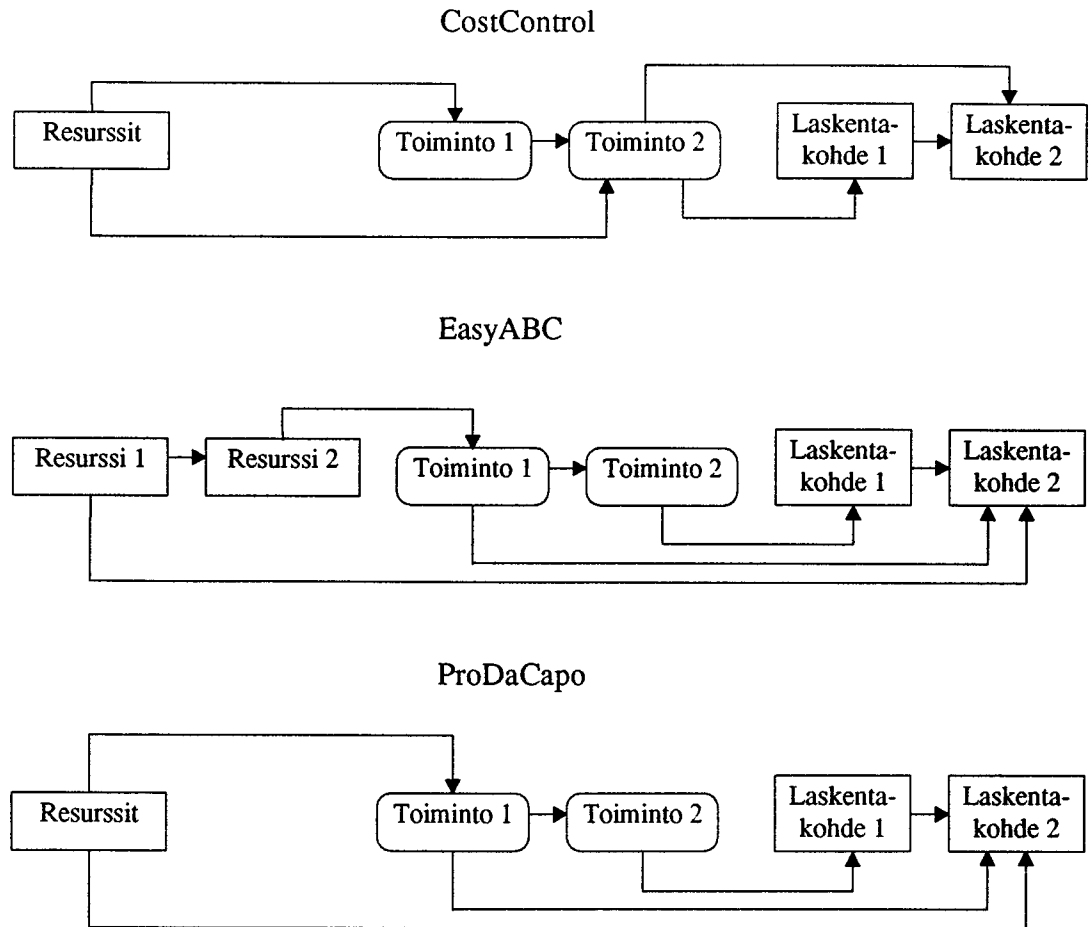


täjä voi CostControlissa määritellä laskentakohteelle tarjouskohtaisen hinnan. Päinvastoin kuin EasyABC:ssä ja ProDaCapossa CostControlissa ei ole mahdollista kohdistaa resurssien kustannuksia suoraan laskentakohteelle. CostControlissa ainoastaan materiaalikustannuksen voi kohdistaa laskentakohteelle ilman toimintoa. EasyABC:ssä laskentakohteille kohdistettavat materiaali- tai komponenttikustannukset voi vielä jakaa ulkopuolelta ostettuihin ja omassa yrityksessä tehtyihin. CostControlissa myös ohjelmiston laskentakohde-osa tukee jo toiminnot-osassa esiintynyttä toimintojen jaottelua neljään eri laskentakohdetasoon.

CostControlin laskentakohteista voi muodostaa myös toimintojen laskentakohdetasoista riippumattoman hierarkian. Tämä on tärkeää muun muassa komponenteista muodostuvien tuotteiden kustannuksia laskettaessa. Kaikissa ohjelmistoissa mikä tahansa laskentakohde voi olla toisen laskentakohteen komponentti. Hierarkian lisäksi laskentakohteet voidaan jaotella vielä käyttäjän haluamiin ryhmiin.

Toimintolaskennan perusteiden mukaisesti toimintoja voidaan kaikissa ohjelmistoissa kohdistaa mille tahansa laskentakohdetasolle. CostControlissa on varsinaisen laskentakohde-osan lisäksi vielä markkina-alue -osa, jolla voidaan kohdistaa haluttuja toimintoja tietyille asiakkaille tai markkina-alueille. Markkina-alue-osan avulla voidaan myös esimerkiksi samalle tuoteyksikölle kohdistaa eri toimintoja asiakkaasta riippuen. ProDaCapo sisältää CostControlin markkina-alue osiota vastaavan asiakas-osion, jolla asiakkaille voidaan kohdistaa heidän aiheuttamiaan kustannuksia. Kustannusten kohdistamisen vaihtoehtoiset kohdistamistavat resursseilta laskentakohteille eri ohjelmistoissa on esitetty kuviossa 7.

Ohjelmistojen laskentakohdeosalle antamaa tukea käyttäjät arvioivat EasyABC:ssä hyvästä erittäin hyväksi ja ProDaCapossa hyväksi. CostControl arvioitiin neutraalisti arvosanalla kolme. Eryityisesti EasyABC:n käyttäjät olivat tyytyväisiä ohjelmiston selkeään ilmaisutapaan, jossa käyttäjä näkee samanaikaisesti koko toimintolaskentaketjun resursseilta laskentakohteille.



**Kuvio 7.** Kustannusten kohdistaminen eri ohjelmistoilla

#### Ajuri-ominaisuudet

Kustannusten kohdistaminen resursseilta toiminnoille tapahtuu resurssiajureilla ja kustannusten kohdistaminen toiminnoilta laskentakohteille toimintoajureilla. Kaikissa ohjelmistoissa kunkin resurssin yhteydessä esitetään sen kohdistamisessa käytettävä resurssiajuri. Samoin toimintojen ja laskentakohteiden yhteydessä esitetään kyseinen toimintoajuri. Molemmista ajurityypeistä ilmaistaan kohdentamatta oleva ajurien lukumäärä, kohdistettu ajurimäärä sekä yhden ajuriyksikön kustannus. Yksittäisten ajurien tiedot on kaikissa ohjelmistoissa esitetty hyvin, mutta CostControlin puutteena voidaan pitää ajureista saatavien yhteenvetotietojen puuttumista.

EasyABC:ssä ja ProDaCapossa kokonaisuuden hahmottaminen on selkeämpää, koska käyttäjä näkee samassa näytössä koko toimintolaskentaketjun resursseilta laskentakohteille. Molemmissa myös ajurien kohdistaminen on selkeästi toteutettu, sillä käyttäjä voi kohdistaa ajureita suoraan hiirellä osoittamalla.

Käyttäjien mielipiteet ohjelmistojen resurssi- ja toimintoajurien tuen osalta olivat kauttaaltaan positiivisia. CostControl ja ProDaCapo arvioitiin hyviksi sekä EasyABC:n arviot vaihtelivat hyvän ja erittäin hyvän välillä.

#### *Toimintomallien yhdistämismahdollisuudet*

Erityisesti eri yksiköistä koostuvassa yrityksessä joudutaan yhdistämään rakennettuja toimintomalleja toisiinsa. CostControl tukee toimintomallien yhdistämistä Liitä-toiminnolla. Ohjelmistolla voidaan yhdistää erillisissä tietokannoissa olevia toimintomalleja. Malleja yhdistettäessä ohjelmisto varoittaa käyttäjää mallien mahdollisista päällekkäisyyksistä. Puutteena CostControlissa on, että se ei tue laskentakohte tai markkina-alue tietojen yhdistämistä. EasyABC:ssä toimintomalleja voidaan yhdistää siirtämällä koko mallin tiedot toiseen toimintomalliin. Siirrossa on mahdollista valita siirretäänkö tiedot budjettitietoina vai toteutuneina tietoina. ProDaCapossa toimintomallien yhdistäminen on mahdollista vastaavalla tavalla kuin EasyABC:ssä.

Käyttäjien näkemysten mukaan toimintomallien yhdistäminen oli kaikissa ohjelmistoissa joko vaikeaa tai erittäin vaikeaa. Tämä oli myös ominaisuus, jota käyttäjät eivät tyypillisesti olleet käyttäneet.

#### *Laskentaominaisuudet*

Toimintomalleissa laskutoimitusten suoritus voidaan toteuttaa joko automaattisesti jatkuvasti suoritettavana tai vain käyttäjän niin halutessa. CostControlissa käyttäjä ei voi valita, haluaako hän itse vaikuttaa laskennan suorittamiseen vai suorittaako ohjelma sen automaattisesti. CostControl päivittää toimintomallia automaattisesti

jokaisen tehdyn muutoksen jälkeen. EasyABC:ssä ja ProDaCapossa laskenta tapahtuu Calculate-toiminnolla vain käyttäjän halutessa päivittää malliaan. Käyttäjän päätettävissä oleva laskenta on parempi vaihtoehto, koska se nopeuttaa ohjelman käyttöä ja mahdollistaa muutosten kokeilemisen ilman, että ne tallentuvat malliin pysyvästi. Sekä ProDaCapossa että EasyABC:ssä käyttäjä voi valita, haluaako hän laskea kaikkien moduulien (resurssit, toiminnot ja laskentakohteet) tiedot, vai ainoastaan jonkin yksittäisen moduulin tiedot.

### *Tiedon kerääminen*

Tiedon keräämisen tuella helpotetaan ajankäyttötietojen keräämistä. CostControl ja ProDaCapo eivät sisällä ominaisuuksia, jotka tukisivat tiedon keräämistä. EasyABC:aan ei näitä ominaisuuksia sisällä, mutta EasyABC:n verkkoversio EasyABC Oros on varustettu Oros Survey -moduulilla, joka on tarkoitettu tiedon keräämiseen. Survey-moduulilla voi muokata toimintotiedoista suoraan kyselykaavakkeen ja jakaa sen verkon kautta henkilöstölle. Kaikilla ohjelmistoilla tiedon keruulomakkeiden laatimista voi helpottaa sillä, että siirtää toimintotiedot niistä taulukkolaskentaohjelmistoon, jossa voi muokata kyselylomakkeen ja palauttaa sen jälkeen täytettyjen lomakkeiden tiedot takaisin toimintojohtamisohjelmistoon.

Kuten jo tiedon keräämisen tuen puuttumisesta voi päätellä, käyttäjät pitivät tukea huonona tai erittäin huonona. ProDaCapon tukea pidettiin huonona ja muiden ohjelmistojen tukea erittäin huonona.

## **5.2.2 Toimintojohtamisen perusajatusten tukeminen**

### *Toimintoanalyysi*

Toimintoanalyysin tarkoitus on tuottaa informaatiota yrityksen toiminnoista päätöksenteon tueksi. CostControlissa toimintojen analysoimista tukee mahdollisuus ryhmitellä niitä osamalli-ominaisuuden avulla. ProDaCaposta löytyy vastaava ominaisuus. Molemmissa ohjelmistoissa toimintoja voidaan ryhmitellä vapaasti niin,

että sama toiminto voi kuulua rajattomaan määrään osamalleja. Osamalleista ei kuitenkaan voi muodostaa hierarkisia kokonaisuuksia. Myös toimintojen kustannuksia käyttäjä voi CostControlissa ja ProDaCapossa tarkastella haluaminaan kokonaisuuksina. EasyABC:ssä olevilla attribuuteilla toimintoja voi analysoida CostControlia ja ProDaCapoa monipuolisemmin. Attribuuteilla toiminnoista voi koota vaihtoehtoisia hierarkioita. Toimintojen analysoimista tukee EasyABC:ssä ja ProDaCapossa myös mahdollisuus tarkastella samalla näytöllä sekä resursseja että toimintoja tai sekä toimintoja että laskentakohteita. Attribuuttien avulla EasyABC:n toimintojen kustannuksia voi tutkia hyvin monipuolisesti.

### *Kustannusajurianalyysi*

Kustannusajurianalyysin tarkoitus on analysoida kustannusajureita ja niiden tarkoituksenmukaisuutta. EasyABC ja ProDaCapo tukevat kustannusajureiden valintaa CostControlia paremmin, sillä EasyABC:ssä ja ProDaCapossa käyttäjä voi valita ajureita aiemmin määrittelemästään ajurilistasta. CostControlissa kukin ajuri on määritettävä erikseen. EasyABC:n kustannusajurien analysointiominaisuudet ovat muita ohjelmistoja monipuolisemmat, sillä ohjelmistossa voidaan valita eri tyyppisiä ajureita. Ensinnäkin jokainen ajuri on joko yhteinen (shared) tai erillinen (unique). Yhteistä ajuria käytetään, kun sekä ajuri että ajurimäärät ovat samat. Erillisiä ajureita käytetään, kun ajuri on sama, mutta ajurimäärät poikkeavat toisistaan. Tämän lisäksi EasyABC:ssä kukin ajuri voi olla perusajuri (basic driver), painotettu ajuri (weighted driver) tai laskettu ajuri (calculated driver). Painotettu ajuri kohdistaa kustannuksia halutulla painotuksella esimerkiksi resurssikulutuksen suhteessa. Laskettu ajuri taas kohdistaa kustannuksia annetun laskentakaavan perusteella. EasyABC:n vaihtoehtoiset ajurit tukevat myös CostControlin ja ProDaCapon ajureita paremmin mitä-jos -analyysseja ja muutosten tekemisen vaikutuksia laskentatuloksiin.

### *Tehokkuuden mittaus*

Tehokkuutta mittaamalla selvitetään, kuinka hyvin toiminnot suoritetaan. Kaikissa ohjelmistoissa voidaan käyttää sekä rahamääräisiä että ei- rahamääräisiä tehokkuuden mittareita. EasyABC:ssä tehokkuuden mittaaminen on toteutettu attribuuttien avulla. Kullekin toiminnolle voidaan määrittää numeerinen attribuutti, jolla seurataan toiminnon tehokkuutta. Liittämällä toimintoja yhteen ja määrittämällä niille yhteinen attribuutti voidaan seurata prosessin tehokkuutta. CostControlissa on tehokkuuden mittaamiseen oma moduuli, jolla määritetään mittarit ja seurataan tehokkuutta. Mittariksi voi valita joko numeerisen tai luokitellun mittarin ja mittauksen kohteeksi jo haluamansa prosessin tai yksittäisen toiminnon. CostControlissa on myös havainnollinen Toimintojen tehokkuus -näyttö, jolla voidaan tarkastella kussakin ajanjaksossa käytettyä osuutta kunkin toiminnon kokonaiskapasiteetista. ProDaCapossa voidaan kullekin toiminnolle määrittää 'best practise' taso, johon toiminnon tai prosessin suoritusta verrataan. Vastaavasti vertaillaan eri osastojen samojen toimintojen suoritusta. Suoritusten mittaaminen tapahtuu toiminnon kustannusten perusteella. Muunlainen suorituksen tai tehokkuuden mittaus ei ProDaCapossa ole mahdollista. Kokonaisuutena tehokkuuden mittaus on toteutettu CostControlissa EasyABC:tä ja ProDaCapoa paremmin.

### *Toimintoperusteinen budjetointi*

Toimintojohtamisen kokonaisvaltaisen hyödyntämisen kannalta on tärkeää että ohjelmistot tukevat toimintoperusteista budjetointiä. CostControlissa budjetointimalli luodaan vastaavalla tavalla kuin toimintomallikin. Ainoa ero on, että kertaluonteisia kustannustekijöitä ei oteta huomioon (CostControl 1995). Itse toimintomallissa voidaan budjettiluvuista esittää vain yhden kustannusajuryksikön budjetoitu kustannus. CostControlin toimintomallin budjetointiominaisuuksia voidaan pitää hyvin puutteellisina. EasyABC:ssä toimintomallin kaikkiin esitettäviin tietoihin voidaan liittää budjetoidut määrät sekä budjetoidut kustannukset. Ainoa budjetointiin liittyvä puute EasyABC:ssä on, että myyntituottoja ei ole mahdollista budjetoida.

ProDaCapo tukee budjetointia vain yksittäisillä mahdollisuuksilla budjetoida toimintojen ja resurssien kulutusta sekä niiden kustannuksia.

Käyttäjien kokemukset ohjelmistojen toimintoperusteisen budjetoinnin tuesta vaihtelivat huomattavasti. EasyABC:n käyttäjistä jotkut pitivät tukea huonona ja jotkut hyvänä. CostControlin käyttäjät kokivat toimintoperusteisen budjetoinnin tuen hyväksi. Jotkut käyttäjät pitivät CostControlin puutteena mahdollisuutta käsitellä samanaikaisesti toteutuneita ja budjettitietoja. ProDaCapon budjetoinnin tukea pidettiin huonona.

#### *Asiakaskannattavuusanalyysi*

Usein toimintolaskennan ja -johtamisen soveltamisen taustalla yrityksessä on halu selvittää eri asiakkaiden todellisia kannattavuuksia. Tämä edellyttää mahdollisuutta kohdistaa asiakkaalle tämän aiheuttamat kustannukset. Kaikissa tutkituissa ohjelmistoissa on mahdollista kohdistaa asiakkaille kaikki näiden aiheuttamat kustannukset. CostControlissa asiakkaiden välisiä kustannuksia on vaivaton tarkastella Markkina-alue -näytöllä, jossa voidaan määrittää kunkin asiakkaan tarvitsemat erityistoiminnot tuotteittain tai asiakkaittain. Tämän lisäksi tuotteet voidaan hinnoitella erilailla kullekin asiakkaalle. CostControl tukee myös asiakkaiden välisten kannattavuuksien vertailua.

ProDaCapossa asiakaskannattavuuksia voidaan tarkastella Customer-näytöllä, jossa kullekin asiakkaalle voidaan kohdistaa hänen aiheuttamansa kustannukset. ProDaCapossa on mahdollista laskea erikseen tuotteiden kannattavuuksia, tuoteryhmien kannattavuuksia, asiakkaiden kannattavuuksia tai tuotekannattavuuksia asiakkaittain. Näiden lisäksi käyttäjä voi attribuuttien avulla määritellä haluamansa kustannuserät ja niitä vastaavat laskentakohteet, joiden kannattavuuksia halutaan tarkastella. Näin ProDaCapossa voidaan tarkastella asiakaskannattavuuksia monipuolisesti ja suorittaa myös asiakkaiden välisiä vertailuja.

EasyABC:ssä on mahdollista analysoida tuottoja ja kannattavuuksia käyttäjän haluamista näkökulmista (dimensions). Käyttäjä voi esimerkiksi tarkastella tietyn alueen, asiakkaan tai tuotteen myyntituottoja ja kannattavuutta. Näkökulmia muodostetaan attribuuteilla merkitsemällä halutun näkökulman mukaiset kustannukset samalla attribuutilla.

Käyttäjien mielipiteet asiakaskannattavuuksien tuesta olivat CostControlin ja EasyABC:n kohdalla neutraaleja. ProDaCapon asiakaskannattavuustarkastelujen tuen käyttäjät arvioivat erittäin hyväksi, mikä todettiin jo ohjelmiston ominaisuuksiin tutustuttaessa.

### 5.2.3 Prosessimallinnuksen tukeminen

#### *Erityyppisten työnkulkujen mallinnusominaisuudet*

Tutkituista ohjelmistoista ainoastaan ProDaCapossa on mahdollista mallintaa työnkulkuja tai prosesseja. Siinä käyttäjä voi mallintaa graafisesti peräkkäisiä ja rinnakkaisia työnkulkuja. Työnkulun eri vaiheisiin voi myös liittää kustannustietoja ja työnkulun kukin vaihe voi olla aliprosessi. CostControlissa voi periaatteessa kuhunkin Osamalliin kuuluvista toiminnoista muodostaa työnkulkuja. Käytännössä tämä kuitenkin tarkoittaa vain sitä, että ohjelmisto sijoittaa toiminnot käyttäjän haluamaan järjestykseen. Tällaista ominaisuutta ei voida pitää prosessin mallintamisena. EasyABC:ssä prosessin mallinnus on samalla tasolla kuin CostControlissakin.

Kuten ohjelmistoja tarkasteltaessakin ilmeni, niin myös käyttäjien mielipiteet tukevat näkemystä, että ohjelmistojen prosessimallinnus- ja kustannuslaskentaominaisuudet ovat puutteellisia. Ainoastaan ProDaCapo sai käyttäjiltä arvion erittäin hyvä. Muiden ohjelmistojen käyttäjät pitivät ominaisuuksia huonoina tai ilmoittivat etteivät olleet käyttäneet kyseistä ominaisuutta.



### *Työnkulkujen ja prosessien kuvausominaisuudet*

EasyABC:ssä ja CostControlissa prosesseja ei voida kuvata eikä niihin voida liittää lisätietoja. Prosessien eri vaiheisiin (toimintoihin) voidaan molemmissa ohjelmistoissa liittää kustannus- ja kapasiteettitietoja. Vaikka ProDaCapossa onkin mahdollista mallintaa työnkulkuja graafisesti, puuttuvat siitäkin prosessien muut kuvausominaisuudet. Ainoastaan kustannus- ja kapasiteettitietoja voidaan liittää prosessin vaiheisiin. Kokonaisuutena voidaan sanoa, että prosessinmallinnusominaisuudet ovat kaikissa tutkituissa ohjelmistoissa heikot. Tosin ProDaCapo erottuu positiivisesti joukosta.

## **5.3 Yleisten ominaisuuksien sijoittuminen arvointikehikossa**

Tässä kappaleessa sijoitetaan tutkitut ohjelmistot toimintojohtamisjärjestelmien arvointikehikon yleisiä ominaisuuksia käsittelevään osaan. Tarkasteltavina ominaisuuksina ovat raportointi, tiedon analysointi- ja käsittelyominaisuudet sekä järjestelmän kustannukset ja tekniset ominaisuudet.

### **5.3.1 Raportointiominaisuudet**

#### *Raporttien monipuolisuus*

Raporttien monipuolisuutta arvioidaan ohjelmiston tuottaminen raporttien lukumäärällä. CostControlissa raportit on toteutettu niin, että jokaisella näytöllä on omat raporttinsa erikseen. Yhteensä CostControl tarjoaa 34 erilaista raporttia. Näitä raportteja on mahdollista saada esimerkiksi käyttäjän haluamista tuotteista tai toiminnoista. ProDaCapo sisältää n. 100 erilaista raporttia ja EasyABC:ssä niitä on 15 erilaista, joita käyttäjä voi attribuuttien avulla tuottaa haluamistaan objekteista. Vaikka toteutustapa on EasyABC:ssä on erilainen kuin CostControlissa ja ProDaCapossa, on lopputulos käytännössä hyvin vastaavanlainen. Erona on CostControlin ja ProDaCapon erilaisten raporttien suurempi määrä, mikä on tietysti niiden

etu. Erilaisia raportteja tulisi voida tuottaa ainakin enemmän kuin EasyABC:ssä ja CostControlinkaan raporttimäärä ei välttämättä ole riittävä. Ohjelmistojen erilaiset raportit on esitetty liitteessä 2.

#### *Raporttien esitystapojen monipuolisuus*

CostControlilla ja ProDaCapolla voi raporttien tueksi tuottaa myös graafeja. Graafeja voi tuottaa vastaavalla tavalla kuin raporttejakin. CostControl tarjoaa 21 erilaista graafia (liite 3), mutta tietoa ei voi esittää taulukkomuodossa. EasyABC:stä graafien, kuten myös taulukoiden, tuottamisominaisuudet puuttuvat kokonaan, mikä on huomattava puute. Tosin EasyABC:n monipuolisemmassa tuotteessa Oroksessa on Reports-moduuli, jolla taulukoiden tuottaminen on mahdollista. Vertailtujen ohjelmistojen raporttien ulkoasut ovat lähes yhtenevät. Kaikkien ohjelmistojen raportit sisältävät perustiedot, kuten raportin nimen, päiväyksen ja aikajakson. Kaikissa ohjelmistoissa raportteja voidaan tulostaa paperille, näytölle tai tiedostoon. CostControl tarjoaa mahdollisuuden tulostaa graafeja myös värillisinä. Vaikka CostControlissa ja ProDaCapossa onkin mahdollista tuottaa graafeja, ei graafeja ja raportteja voi yhdistää itse ohjelmassa. Esitystavaltaan monipuolisempia raporteista saa, jos tulostaa graafin ja raportin tiedostoon ja yhdistää ne esimerkiksi tekstinkäsittelyohjelmassa.

#### *Raporttien tuki analyyseille, budjetoinnille ja ennusteille*

CostControlissa budjettien muodostaminen tapahtuu vastaavalla tavalla kuin toimintomallienkin muodostus. Tästä johtuen budjeteista on mahdollista saada samat raportit ja graafit kuin toimintomalleistakin. Ongelmana tämänkaltaisessa raportoinnissa on, että toteutuneita ja budjetoituja kustannuksia sekä ei-rahamääräisiä arvoja on vaikea vertailla. Kuitenkin yksi tärkeä vaihe budjetoinnin hyödyntämisessä on toteutuneiden ja budjetoitujen kustannusten vertailu (Riistama ja Jyrkkiö 1994, s. 350). EasyABC:ssä budjetoidut tiedot on ilmaistu samassa raportissa kuin toteutuneetkin tiedot. Käyttäjä voi valita tulostetaanko raporttiin sekä toteutuneet että budjetoidut tiedot vai pelkästään toiset näistä. Tällainen toteutustapa tukee

vertailujen tekemistä. Toisaalta EasyABC:ssä raportointimahdollisuudet eivät ole yhtä laajat kuin CostControlissa ja ProDaCapossa, joten raporttien antama tuki budjetoinnillekaan ei voi olla yhtä monipuolista.

#### *Toimintomallien yhteenveto- ja vertailuraportit*

Ohjelmistosta vain ProDaCapo tukee yhteenvetoraportteja eri toimintomalleista. ProDaCapossa käyttäjä voi valita toimintomalleja ja tuottaa niistä yhteenvetoraportteja. Muissa ohjelmistoissa ainoa tapa saada yhteenvetoraportteja, on ensin yhdistää toimintomalleja ja sitten tulostaa raportteja tästä yhdistetystä mallista. EasyABC:n Oros -version Reports moduulilla on mahdollista yhdistellä eri toimintomallien tietoa samassa raportissa.

#### *Raporttien räätälöintimahdollisuudet ja räätälöinnin tuki*

CostControlissa on Raporttipohja-ominaisuus raporttien räätälöintiä varten. Raporttipohjan avulla käyttäjä voi muodostaa haluamiaan raportteja. Ajan suhteen on valittavana raportoitavaksi mikä tahansa ajanjakso. Raporteista räätälöitäviä ovat kuitenkin vain erilaiset tuloslaskelmaraportit. Niissä käyttäjä voi yhdistellä eri tuloslaskelman osia tai ottaa tuloslaskelmaan mukaan vain haluamiaan komponentteja. CostControl tukee raporttien räätälöintiä opastus-ikkunalla. EasyABC:ssä raportteja räätälöidään attribuuttien avulla. Merkitsemällä halutut objektit samalla attribuutilla, voidaan näistä tulostaa vaihtoehtoisia raportteja. Kuitenkin valittavana ovat vain ohjelmiston 15 valmisraporttia, joita voi attribuuteilla muokata, mutta uudenlaisia raportteja ei voi muodostaa. EasyABC:n monipuolisempi versio EasyABC Oros sisältää Reports-moduulin, jossa raportteja voi muokata hyvin monipuolisesti. ProDaCapossa raportteihin voi vaikuttaa rajallisesti valitsemalla raportoinnin kohteen.

Verrattaessa tuloksia EIS-ohjelmistoilla tehtyihin tutkimuksiin, voidaan todeta, että Carlislen ja Alameddinen (1990) tutkimissa 21:ssä EIS-ohjelmistossa vain kolmeen sisältyi räätälöidyt raportit. Näin ollen tämä tukee ajatusta, että raportointi on toi-

mintojohtamisjärjestelmän keskeinen ominaisuus, sillä toimintojohtamisjärjestelmissä raporttien räätälöintiominaisuudet ovat monipuolisempia kuin EIS-ohjelmistoissa.

#### *Poikkeamaraportit*

Ohjelmistot eivät varsinaisesti tue poikkeamaraportointia. CostControl tarjoaa joitain graafeja, joista voidaan visuaalisesti nähdä esimerkiksi käytetyt resurssit versus kapasiteetti, mutta ei suoranaisia poikkeamaraportteja. EasyABC:ssä voidaan raportoida erikseen esimerkiksi resurssit, jotka sisältävät ylikapasiteettia tai toiminnot, joihin vaikuttamalla saavutettaisiin suurimmat kustannussäästöt. Ominaisuuksia ei voida pitää poikkeamaraportointina, mutta sinänsä hyödyllisinä ohjelmistojen piirteinä. EIS-järjestelmissä poikkeamaraportointi esiintyy yleisenä ominaisuutena, sillä Partasen (1992, s. 82) tutkimuksessa tutkituista 13 EIS-ohjelmistosta 11 sisälsi poikkeamaraportoinnin. Tässä suhteessa toimintojohtamisjärjestelmät poikkeavat EIS-järjestelmistä. Koska toimintojohtamisjärjestelmissä raportointi on keskeinen ominaisuus, voidaan tämän tutkimuksen perusteella todeta, että poikkeamaraportointia tulisi toimintojohtamisjärjestelmissä kehittää.

Käyttäjät suhtautuivat vaihtelevasti ohjelmistojen raportoinnin tukeen. Ainoastaan CostControl arvioitiin hyväksi. Muiden ohjelmistojen raportoinnin tukea pidettiin neutraalina tai huonona. Ohjelmistojen raportointiominaisuuksia tulisi selkeästi kehittää, sillä päätöksenteon tukivälineenä ohjelmistojen tulisi tuottaa monipuolisia ja käyttäjiä tyydyttäviä raportteja.

### **5.3.2 Järjestelmien tiedon analysointiominaisuudet**

#### *Tietoon porautuminen ja ennakoimattomat kyselyt*

Kaikki tutkitut ohjelmistot tukevat tietoon porautumista. Kaikissa ohjelmistoissa tärkeimmät ominaisuudet ja ohjelmiston osat tukevat tietoon porautumista. Toi-

saalta mikään tutkituista ohjelmistoista ei tue ennakoimattomia kyselyjä. Vaikka tulos sinänsä on yllättävä ovat Bergeron ja Raymond (1992) tulleet EIS-järjestelmiä koskeneessa tutkimuksessaan myös tulokseen, että kovinkaan suuri osa EIS-järjestelmistä ei sisällä ennakoimattomia kyselyitä. Heidän tutkimuksessaan 13:n yrityksen järjestelmä 28:sta tuki ennakoimattomia kyselyitä. Toimintojohtamisjärjestelmissä ennakoimattomista kyselyiden tuki olisi hyödyllinen ominaisuus ainakin ylimmälle johdolle, jonka tietotarpeet ovat tyypillisesti nopeasti muuttuvia ja ne eivät ole ennakoitavissa (Hackathorn 1995, s. 38).

#### *Mitä-jos -analyysit*

Tutkituista ohjelmistoista EasyABC ja ProDaCapo tukevat parhaiten mitä-jos-analyyseja. Niissä analyysien tekemisen mahdollistaa Calculate-funktio, jolla toimintomallin laskenta suoritetaan vasta käyttäjän niin halutessa. Varsinaista muuta tukea mitä-jos-analyyseille EasyABC tai ProDaCapo ei anna, mutta sen antamaa tukea voitaneen pitää riittävänä. CostControlissa ohjelma suorittaa automaattisesti mallin laskentaa, joten mitä-jos-analyyseja ei voi tehdä muuten kuin tallentamalla kaikki kokeillut muutokset. Tätä ei voida pitää mitä-jos-analyyseja tukevana ominaisuutena. Verrattaessa toimintojohtamisjärjestelmiä EIS-järjestelmiin voidaan todeta, että Carlisle ja Alameddinen (1990) tutkimuksessa 12:sta EIS-ohjelmistosta 11 tuki mitä-jos-kyselyitä. Näin ollen tämän tutkimuksen perusteella EIS-järjestelmissä mitä-jos-analyysien tuki on tyypillisempi ominaisuus kuin toimintojohtamisjärjestelmissä.

### **5.3.3 Järjestelmän tiedon käsittelyominaisuudet**

#### *Tiedonsiirrot järjestelmien välillä*

Normaalisti toimintolaskentamallia ei rakenneta kokonaan itse ohjelmistolla, vaan ohjelmistoon tuodaan tietoa muualta organisaatiosta. Tästä syystä on tärkeää, että toimintojohtamisohjelmisto tukee tiedonsiirtoja muista järjestelmistä. Tiedonsiirrot on toteutettu ohjelmistoissa eri tavoin. CostControlissa on Siirtotiedostot-ikkuna,

jossa käyttäjä voi valita 18:sta eri tiedostotyypistä (lueteltu liitteessä 4), millaista tietoa ohjelmistoon halutaan siirtää. Tämän jälkeen valitaan tiedosto, joka halutaan siirtää. Käyttäjä voi esimerkiksi valita siirtotiedostotyypin resurssit, jolloin valitun tiedoston tiedot siirretään ohjelmistossa kohtaan resurssit. Vastaavasti siirretään tietoa ohjelmistosta ulospäin. CostControl lukee ja kirjoittaa määrämuotoisia ASCII (\*.txt)-tiedostoja, joissa sarake erottimena on tabulaattori (CostControl 1995).

EasyABC:ssä tiedostojen siirto on toteutettu niin, että kuhunkin siirtotiedostoon sisällytetään seuraavat otsikkokentät: ohjelmiston nimi, version numero, kohde moduuli (resurssit, toiminnot, laskentakohteet tai kaikki moduulit) ja siirtotiedoston tyyppi (lueteltu liitteessä 4). Otsikkokenttä voi näyttää esimerkiksi seuraavalta:

```
EASY ABC
VERSION      3.0
RESOURCE
COSTS_ACTUAL
```

EasyABC tunnistaa 17 eri siirtotiedostotyyppiä. Näistä osa on sellaisia, että niillä voi siirtää tarkastelujaksoja koskevaa tietoa joko todellisina tai budjetoituina tietoina. Tällaisia siirtotiedostoja on kuutta eri tyyppiä (liite 4). Perustietojen ja tarkastelujaksokohtaisten tietojen ohella EasyABC:hen voi siirtää kokonaisia toimintomalleja. Tätä on käsitelty kohdassa Toimintomallien yhdistämismahdollisuudet. Kuten CostControlissa myös EasyABC:ssä tietoa voidaan siirtää vastaavilla tavoilla ohjelmistosta ulospäin. EasyABC lukee ja kirjoittaa joko tabulaattorilla tai pilkulla erotettua tekstiä sekä DIF (Data Interchange Format)-formaatin mukaista tietoa. Kaikissa ohjelmistoissa on selkeät virheilmoitukset tiedonsiirroissa tapahtuvista virheistä.

ProDaCapoon voi siirtää tekstitiedostoja, Excel-tiedostoja sekä AS400 formaatin mukaista tietoa. ProDaCapoon voi siirtää muuta kuin toimintoja koskevaa tietoa.

tävä tekijä positiiviseen arviointiin saattaa olla se, että vielä nykyisin organisaatioissa on käytössä joitakin merkkipohjaisia käyttöliittymiä esimerkiksi tuotannonohjauksessa. Merkkipohjaisten ohjelmistojen käyttöön tottunut käyttäjä pitää todennäköisesti kaikkia Windows-käyttöliittymiä selkeinä.

### *Grafiikka*

Grafiikalla voidaan ilmaista sanallisesti vaikeasti selitettävissä olevia asioita nopeasti ja havainnollisesti. Tutkituista ohjelmistoista EasyABC:llä ei voi tuottaa grafiikkaa. Tätä voidaan pitää suorastaan yllättävänä tuloksena. Tulosta omalta osaltaan selittää se, että EasyABC:n Oros versio toimii integroidusti PowerPlay - grafiikkaohjelmiston kanssa, jolla on tarkoitus tuottaa tarvittava grafiikka. Tämä edellyttää kuitenkin erillisten ohjelmistojen hankkimista.

CostControlista on mahdollista saada 21 erilaista graafia (liite 3). Kaikki graafit voi tulostaa paperille joko mustavalkoisina tai värillisinä. CostControlin graafeja käyttäjä voi muokata melko rajallisesti. Graafeina käytettävissä ovat pylväsdiagrammit. Graafien räätälöinti rajoittuu pylväsdiagrammissa pylväiden esitysjärjestyksen valintaan. Tämän lisäksi käyttäjä voi skaalata graafeja, mutta niiden otsikoihin ei voi vaikuttaa. Sinänsä CostControlin graafit ovat havainnollisia.

ProDaCapolla voi tuottaa noin 30 erilaista graafia. Graafit käsittelevät esimerkiksi tuote- tai asiakaskohtaista kannattavuutta sekä resurssien ja toimintojen toteutuneita kustannuksia verrattuna budjetoituihin. ProDaCapon graafien muokkausmahdollisuudet ovat melko rajalliset.

Grafiikan osalta tulokset poikkeavat EIS-ohjelmistoilla saaduista tuloksista, sillä aiempien tutkimusten perusteella grafiikkaominaisuudet sisältyvät lähes jokaiseen EIS-ohjelmistoon (Carlson et al. 1992, Carlisle ja Alameddine 1990).

Ohjelmistojen välillä syntyi eroja käyttäjien arvioidessa ohjelmistojen graafeja. CostControlin graafeja pidettiin hyvinä. Ymmärrettävästi EasyABC:n graafeja pidettiin erittäin huonoina, koska ohjelmisto ei sisällä graafeja. ProDaCapon graafit arvioitiin neutraalisti, tosin käyttäjät kiinnittivät huomiota lukumääräisesti vähäiseen erilaisten graafien määrään.

#### *Käytön oppimisen helppous*

Käytön oppimisen helppoutta on arvioitu käyttäjien kokemusten perusteella. Tämän lisäksi arvioidaan ohjelmiston manuaaleja sekä On-Line help -toimintoa. Kaikissa ohjelmistoissa manuaali on selkeä ja monipuolinen sekä sisältää esimerkkejä ohjelmiston käytöstä. CostControlissa manuaalin saa myös suomenkielisenä. On-Line help löytyy kaikista ohjelmistoista. Muissa ohjelmistoissa toiminto on englanninkielinen, mutta CostControlissa käyttäjä voi valita myös suomenkielisen On-Line helpin. EasyABC:n On-Line help on muiden ohjelmistojen vastaavaa toimintoa laajempi.

Ohjelmiston käytön oppimisessa on yksilöllisiä eroja. Tästä syystä käytön oppimista arvioitiin saman ohjelmiston kohdalla neutraalista erittäin helppoon. Kuitenkaan kukaan käyttäjä ei pitänyt minkään ohjelmiston käytön oppimista vaikeana. Käytön oppimista sanottiin helpottaneen toimittajan järjestämä koulutus. Manuaaleja ei voida pitää käytön oppimista helpottaneena tekijänä, sillä kaikki käyttäjät arvioivat ohjelmistojen manuaalien olevan huonoja tai enintäänkin neutraaleja. Tutkimuksen perusteella manuaaleissa olisikin selvästi tehtävä kehitystyötä.

#### *Käytön helppous*

Myös käytön helppoutta on arvioitu käyttäjien kokemusten perusteella. Tämän lisäksi kiinnitetään huomiota ohjelmiston käyttökieleen. Kaikkia ohjelmistoja voi käyttää englannin kielellä. CostControlia voi englannin lisäksi käyttää suomeksi, ranskaksi, ruotsiksi tai hollannin kielellä. Vaikka CostControlissa käyttäjä voi valita



valikoiden kielen viidestä mahdollisesta, on helpin sisältö kuitenkin suomeksi tai englanniksi.

Kaikki käyttäjät arvioivat ohjelmistonsa käyttöä vastaavalla arviolla kuin ohjelmiston käytön oppimista. Näin ollen arviot vaihtelivat neutraalista erittäin helppoon.

### **5.3.5 Järjestelmän kustannukset**

#### *Perusohjelmistolisenssin hinta*

EasyABC:n yksi lisenssi maksaa noin 32 000 markkaa ja ProDaCapon yksi lisenssi maksaa noin 50 000 markkaa. EasyABC:n verkkoversion EasyABC Oroksen yksi lisenssi, joka sisältää Administrator-moduulin, maksaa noin 38 000 markkaa. ProDaCapon verkkoversion yhden lisenssin hinta on 63 000 markkaa. Koulutuskäyttöön myytävä EasyABCQuick maksaa noin 3 200 markkaa ja ProDaCapon vastaava rajoitettu versio maksaa noin 26 000 markkaa. Normaalisti CostControlia ei myydä pelkästään yhtä lisenssiä. Tämä johtuu siitä, että tavallisesti CostControl toimitetaan vain verkossa toimivana ohjelmistona. Q.P.R. ei halua ilmoittaa ohjelmistonsa CostControlin yhden lisenssin hintaa.

#### *Monen käyttäjän lisenssien hinnat*

EasyABC:n viisi Administrator-moduulin sisältävää lisenssiä maksavat noin 160 tuhatta markkaa ja kymmenen lisenssiä maksavat noin 300 000 markkaa. ProDaCapon viisi lisenssiä maksavat noin 115 000 markkaa ja kymmenen lisenssin hinta on noin 181 000 markkaa. Q.P.R:n ilmoituksen mukaan CostControlin hinta riippuu asiakkaasta ja lisenssien määrästä. Tyypillinen CostControlin hankinnan kustannus on noin 150 - 200 tuhatta markkaa.

#### *Ohjelmistotoimittajan tuen kustannukset*

EasyABC:n ylläpitosopimus, joka sisältää puhelintuen ja ohjelmistopäivitykset, maksaa asiakkaalle vuosittain 15 prosenttia ohjelmiston hankintahinnasta.

CostControlin vastaava ylläpitosopimus maksaa 20 prosenttia hankintahinnasta. ProDaCapossa vastaavan ylläpitosopimuksen maksuprosentti vaihtelee niin, että yksittäiskoneeseen asennetussa ohjelmistossa se on 20 prosenttia ja verkkoversiossa 15 prosenttia ohjelmiston listahinnasta.

#### *Koulutuksen kustannukset*

EasyABC:n koulutuspäivä maksaa noin 3100 markkaa. Q.P.R ei halua ilmoittaa koulutuspäivänsä hintaa. Heidän mukaansa tyypillinen CostControlin käyttöönottoprojekti sisältää 5-30 päivää konsultaatiota, joka veloitetaan erikseen. Myös ProDaCapolta koulutuksen saaminen on ongelmallista, koska ohjelmistolla ei ole edustajaa Suomessa.

Kokonaisuutena voidaan sanoa, että ohjelmistotoimittajien hinnoittelupolitiikoissa on eroja. Kunkin ohjelmiston hinta toisiinsa verrattuna vaihtelee lisenssimäärien ja koulutustarpeen mukaan. Kuitenkaan tällaisia ohjelmistoja hankittaessa hinta ei ole tärkein kriteeri ja ohjelmistojen välisillä eroilla ei tässä suhteessa ole suurta painoarvoa.

### **5.3.6 Tekniset ominaisuudet**

#### *Avoin tai suljettu arkkitehtuuri*

Tutkituista ohjelmistoista ProDaCapo ja CostControl tukevat avointa arkkitehtuuria. Ne tukevat ODBC (Open DataBase Connectivity)-standardia eli tietoa voidaan hakea suoraan halutusta tietokannasta. EasyABC ei tue ODBC-standardia, mutta EasyABC Oroksen Links Engine tukee ODBC:tä.

#### *Yksittäiskoneessa toimiva ohjelmisto*

EasyABC on yksittäiskoneessa toimiva ohjelmisto. EasyABC:tä ei voida asentaa verkossa käytettäväksi. Ainoastaan EasyABC Oros on verkossa toimiva ohjelmisto. EasyABC:n toimimattomuus verkossa on ohjelmiston käyttötarkoituksen huo-

mioiden merkittävä puute. Normaalisti toimintojohtamisohjelmistoa käyttävät yrityksessä useat ihmiset ja ohjelmiston toimimattomuus verkon yli rajoittaa käytettävyyttä.

#### *Verkossa toimiva ohjelmisto*

CostControl ja ProDaCapo ovat verkossa toimivia ohjelmistoja. Tämä tarkoittaa sitä, että ohjelmistoja voi käyttää verkon yli ja toimintomallilla voi olla useita samanaikaisia käyttäjiä. Samanaikaista käyttöä kontrolloidaan niin, että ensimmäisenä toimintomalliin sisäänkirjoittautuva käyttäjä voi tehdä malliin muutoksia ja muut voivat käyttää mallia ilman kirjoitusmahdollisuutta. Tästä myös informoidaan käyttäjää mallia avattaessa. Sekä ProDaCapoa että CostControlia voi käyttää myös yksittäiskoneessa toimivana ohjelmistona.

#### *Verkko-ohjelmisto*

Mikään tutkituista ohjelmistoista ei ole määritellyn mukainen verkko-ohjelmisto. Tämä ei sinänsä ole mikään puute, koska ohjelmistojen luonteen huomioiden niille riittää toimivuus verkossa. EasyABC:n Oros on client/server verkon vaativa verkko-ohjelmisto.

#### *Järjestelmän suojausominaisuudet*

Toimintojohtamisjärjestelmä sisältää yrityksen kannalta kriittistä tietoa, joten järjestelmän tulisi olla tehokkaasti suojattu. CostControlissa on erillinen Käyttäjaoikeuksien hallinta -moduuli, jolla voidaan määritellä kullekin käyttäjälle käyttäjätunnus ja käyttäjä voi määritellä itselleen salasanan. Ohjelmistolla voi olla yksi tai useampia pääkäyttäjiä, joilla on oikeus muuttaa muiden käyttäjien käyttöoikeuksia. Myös ProDaCapossa käyttäjaoikeuksien hallinta on pääkäyttäjillä. EasyABC:ssä on mahdollista määritellä vastaavasti käyttäjätunnuksella ja salasanalla oikeutetut käyttäjät kullekin toimintomallille. EasyABC:ssä ei ole erillisiä pääkäyttäjiä, vaan kaikilla mallin käyttöön oikeutetuilla on mahdollisuus muuttaa käyttäjätunnuksen salanaa.

### *Näkymien- ja funktioiden suojausominaisuudet*

Tietoturvan takaamiseksi tulisi pystyä valvomaan myös niitä henkilöitä, jotka on oikeutettu järjestelmän käyttöön. CostControlissa voidaan käyttäjätunnusten perusteella määritellä näyttökohtaiset käyttöoikeudet seuraavasti (CostControl 1995):

- käyttäjällä on kaikki oikeudet näytön toimintoihin ja tietoihin,
- käyttäjällä on oikeus ainoastaan katsella ja selata näytön tietoja tai
- käyttäjällä ei ole oikeutta avata näyttöä.

EasyABC:ssä ja ProDaCapossa toimintomallin eri näkymiä ei voi erikseen suojata, vaan mallin oikeutetulla käyttäjällä on pääsyoikeus koko malliin. Tosin ProDaCapon pääkäyttäjä voi antaa yksittäiselle käyttäjälle kaikki oikeudet, nimittäin luku- tai kirjoitusoikeudet tai rajoitetut kirjoitusoikeudet, tiettyyn toimintomalliin. EasyABC:ssä käyttäjälle voidaan antaa toimintomalliin joko luku- tai kirjoitusoikeudet.

### *Laitteisto- ja käyttöjärjestelmävaatimukset*

Laitteisto- ja käyttöjärjestelmä ominaisuuksia ei arviointikehikossa painoteta. Niiden osalta tyydytään esittämään taulukossa 4 kunkin ohjelmiston laitteisto- ja käyttöjärjestelmävaatimukset.

	<b>CostControl</b>	<b>EasyABCPlus</b>	<b>ProDaCapo</b>
<b>Proessorivaatimus</b>	386	486	386 (suositellaan 486)
<b>Käyttöjärjestelmä</b>	Windows 3.1 tai uudempi	Windows 3.1 tai uudempi (suositellaan Windows 95)	Windows 3.1 tai uudempi
<b>Keskusmuisti</b>	4 MB	8 MB (suositellaan 16 MB)	8 MB
<b>Kovalevytila</b>	2,5 MB	14 MB	25 MB

**Taulukko 4.** Kunkin ohjelmiston laitteisto- ja käyttöjärjestelmävaatimukset

#### 5.4 Yhteenveto tutkimustuloksista

Muodostettuun arviointikehikkoon perustuen tutkittiin kolmen toimintoperusteista ajattelua tukevan ohjelmiston ominaisuuksia. Nämä ohjelmistot sijoitettiin kehikkoon ja analysoitiin ominaisuus kerrallaan. Tutkimuksen tuloksena voidaan todeta, että ohjelmistot tukivat toimintolaskentaa kohtuullisen hyvin. Puutteita löytyi resurssikulutusmallin tukemisesta ja toimintojen jaottelusta toimintolaskennan perusteiden mukaisesti. Näiden lisäksi toimintolaskentaketju oli erityisesti CostControlissa esitetty epäselvästi.

Yksi selkeimmistä tietojenkäsittelyn tuomista eduista toimintolaskennassa ovat tiedon hankinnan ja tiedon siirtojen helpottaminen. Yllättävänä lopputuloksena tutkimuksessa ilmeni, että ohjelmistot eivät käytännössä tue toimintolaskentaa varten tarvittavan tiedon hankintaa. Sen sijaan tiedon siirtämistä muista järjestelmistä toimintojohtamisohjelmistoon kaikki ohjelmistot tukivat monipuolisesti.

Toimintojohtamisen perusajatusten tuesta toiminto- ja asiakaskannattavuusanalyysi olivat parhaiten toteutetut ominaisuudet. Kustannusajurianalyysiominaisuuksista puuttuivat kaikista ohjelmista monipuoliset yhteenvetotiedot. Tehokkuuden mittaaminen oli CostControlissa monipuolisin, mutta muutkin ohjelmistot tukivat tehokkuuden mittaamista riittävästi. Suurimmat puutteet toimintojohtamisen perusajatusten tukemisesta löytyivät toimintoperusteisesta budjetoinnista.

Prosessinäkökulma on kustannusnäkökulman ohella toimintoperusteisen ajattelun perusasia. Tästä syystä ohjelmistojen tulisi tukea prosessiajattelua. Kuitenkin tutkituissa ohjelmistoissa oli huomattavia puutteita juuri prosessien mallintamisessa.

Johtamisen tietojärjestelmien yleisistä ominaisuuksista raportointi- ja tiedon analysointiominaisuudet olisivat tutkimuksen perusteella potentiaalinen kehittämiskohde. Tätä mieltä olivat myös ohjelmistojen käyttäjät. Sen sijaan tiedon käsittelyominais-

suudet ja käyttäjäominaisuudet oli pääosin toteutettu hyvin. Järjestelmien kustannukset eivät, kaikki tekijät huomioiden, merkittävästi poikenneet toisistaan. Myöskään teknisten ominaisuuksien osalta ei tutkimuksessa ilmennyt mitään yllättävää.

Kokonaisuutena voidaan sanoa, että käyttäjät olivat pääosin tyytyväisiä käyttämänsä ohjelmistoon. Tähän on syynä paljolti myös se, että käyttäjät ovat käyttäneet vain omaa ohjelmistoaan eikä heillä välttämättä ole tietoa muiden ohjelmistojen ominaisuuksista.

Taulukossa 5 on esitetty yhteenveto arviointikehikossa tutkituista ominaisuuksista. Jos ohjelmistossa on tietty ominaisuus, on se ilmaistu yhdellä '+'-merkillä. Tämän sijasta käytetty '+ +' -merkintä tarkoittaa, että ominaisuus on toteutettu ohjelmistossa hyvin. Edelleen '+ + +' -merkintä tarkoittaa erittäin hyvää ominaisuuden toteutusta. Arviot perustuvat aiemmissa kappaleissa raportoituihin tekijän ja käyttäjien kokemuksiin.

	CostControl	EasyABCPlus	ProDaCapo
<b>Toimintolaskennan tuki</b>			
Resurssien tukiominaisuudet	++	++	++
Resurssikulutus-mallin tuki	+	+	+
Toiminnot-osan tukiominaisuudet	+	++	+
Toiminnon tukitoiminto ominaisuus	+	+	+
Laskentakohde-ominaisuudet	+	++	++
Ajuri-ominaisuudet	++	++	++
Toimintomallien yhd. mahd.	+	+	+
Laskentaominaisuudet	autom.	Calculate-painike	Calculate-painike
Tiedon keräämisen tuki	ei	ei	ei
<b>Toimintojohtamisen tuki</b>			
Toimintoanalyysin tuki	+	++	+
Kustannusajurianalyysin tuki	ei	++	+
Tehokkuuden mittauksen tuki	++	+	+
Toimintoperusteinen budjetoinin tuki	eri malli	++	ei
Asiakaskannattavuusanalyysin tuki	+	+	++
<b>Prosessimallinnuksen tuki</b>			
Erityyppisten työkulkujen mallinnus-ominaisuudet	ei	ei	+
Työkulkujen ja prosessien kuvausominaisuudet	ei	ei	ei

(jatkuu)

Taulukko 5. (jatkuu)

	CostControl	EasyABCPlus	ProDaCapo
<b>Raportointiominaisuudet</b>			
Raporttien monipuolisuus ( lkm)	34 erilaista	15 erilaista	n. 100 erilaista
Raporttien esitystavat	myös graafeja		myös graafeja
Raporttien tuki analyyseille, budjetoinnille ja ennusteille	+	++	ei
Toimintomallien yhteenveto- ja vertailuraportit	ei	ei	+
Raporttien räätälöinti	+	++	+
Poikkeamaraportit	ei	ei	ei
<b>Tiedon analysointiominaisuudet</b>			
Tietoon porautuminen	+	++	+
Ennakoimattomat kyselyt	ei	ei	ei
Mitä-jos -analyysit	ei	++	++
<b>Tiedon käsittelyominaisuudet</b>			
Järjestelmien väliset tiedonsiirrot	++	++	++
Haut tietokantoihin	++	ei	+
Tiedon päivitysten tuki	++	++	++
Ulkoisen tiedon käyttöominaisuudet	+	+	+
<b>Käyttäjäominaisuudet</b>			
Käyttöliittymä	+	++	++
Grafiikka	++	ei	+
Käytön oppimisen helppous	+	+	+
Käytön helppous	+	+	+
<b>Järjestelmän kustannukset</b>			
Perusohjelmistolisenssin hinta	*)	32 000 mk	50 000 mk
Viiden käyttäjän lisenssien hinta	*)	160 000 mk	115 000 mk
Kymmenen käyttäjän lisenssien hinta	*)	300 000 mk	181 000 mk
Ohjelmistotoimittajan tuen kust. **)	20 %	15 %	20 %
<b>Koulutuksen kustannukset</b>			
<b>Tekniset ominaisuudet</b>			
Avoin tai suljettu arkkitehtuuri	avoin	avoin	avoin
Käyttötapa	verkko	stand-alone	stand-alone/verkko
Järjestelmän suojausominaisuudet	salasana	salasana	salasana
Näkymien- ja funktioiden suojausominaisuudet	++	ei	ei

\*) Q.P.R ei halua ilmoittaa ohjelmistolisenssiensä hintoja.

\*\*) Ohjelmistotoimittajan tuen kustannukset on ilmaistu prosentteina hankintahinnasta.

*Taulukko 5. Yhteenveto ohjelmistojen ominaisuuksista*

## 6 YHTEENVETO

Tässä tutkimuksessa selvitettiin toimintojohtamiseen tarkoitettujen ohjelmistojen ominaisuuksia. Tätä varten muodostettiin arviointikehikko, johon toimintojohtamisohjelmistoja sijoitettiin. Toimintojohtamisohjelmisto määriteltiin yksittäiseksi tai moduuleista koostuvaksi ohjelmistoksi, joka tukee toimintolaskentaa ja toimintojohtamista. Tämän lisäksi määriteltiin toimintojohtamisjärjestelmä johtamisen tietojärjestelmäksi, joka tukee toimintolaskentaa ja toimintojohtamista, ja on integroitu osaksi yrityksen muita tietojärjestelmiä.

Arviointikehikko muodostui kahdesta pääosasta: toimintojohtamisominaisuuksista ja yleisistä ominaisuuksista. Toimintojohtamisominaisuudet jakautuivat toimintolaskentaa, toimintojohtamista- ja prosessiajattelua tukeviin ominaisuuksiin. Tämän osan perustana käytettiin toimintolaskennan ja -johtamisen teoriassa esiintyviä ominaisuustarpeita. Kehikon johtamisen tietojärjestelmien yleisiä ominaisuuksia käsittelevä osa pohjautui soveltuvien osin aiempiin EIS-järjestelmien arviointikehikoihin. Kehikon yleisessä osassa käsiteltiin kaikilta johtamisen tietojärjestelmiltä vaadittavia ominaisuuksia ja toimintojohtamisosassa toimintoperusteisen ajattelun asettamia erityisvaatimuksia. Tutkimuksen tärkeimpänä kontribuutiona on arviointikehikon muodostaminen toimintojohtamisjärjestelmille. Aiemmissä tutkimuksissa on käsitelty useiden eri tutkijoiden esittämiä EIS-järjestelmien tutkimuskehityksiä, mutta toimintoperusteista ajattelua tukeville järjestelmille vastaavaa kehikkoa ei aiemmin ole esitetty. Sekundäärisenä kontribuutiona toimintojohtamisjärjestelmiä on asemoitu suhteessa muihin johtamisen tietojärjestelmiin.

Tutkimuksessa kehikkoon sijoitettiin kolme toimintolaskentaa ja -johtamiseen tarkoitettua ohjelmistoa. Tutkitut ohjelmistot olivat CostControl, EasyABC ja ProDaCapo. Tutkimus rajattiin käsittelemään vain vapaasti saatavilla olevia ohjelmisto-



ja. Tästä syystä tutkimuksesta rajattiin pois eri konsultti- ja tilintarkastustoimistojen omia ohjelmistoja, joita he käyttävät toimintolaskentaprojekteissaan.

Tutkimus perustui tekijän omiin kokemuksiin sekä tapaustutkimukseen ohjelmistoja käyttävien organisaatioiden kokemuksista. Käyttäjäkokeimuksilla selvitettiin käyttäjien näkemyksiä käyttämistään ohjelmistoista ja niiden yksittäisistä ominaisuuksista. Ominaisuuksista selvitettiin myös, mitä ominaisuuksia ohjelmistoista käytettiin. Tämän ohella selvitettiin toimintojohtamisohjelmistojen käyttäjien sijoittumista organisaatioissa sekä käyttäjäorganisaatioiden kokoa ja toimialaa.

Tutkimuksessa muodostetun arviointikehyksen mukaisesti toimintojohtamisjärjestelmien ominaisuudet jaettiin kahteen pääosaan toimintojohtamista tukeviin ja yleisiin ominaisuuksiin. Toimintojohtamisominaisuuksista toimintolaskennan edellyttämiä ominaisuuksia ohjelmistot tukivat kohtuullisen hyvin. Resurssikulutusmallin tukeminen ja toimintojen jaottelu toimintolaskennan perusteiden mukaisesti olivat tosin puutteellisesti toteutettuja ominaisuuksia. Näiden lisäksi ohjelmistot tukivat huonosti toimintolaskennassa välttämättömän tiedon hankintaa. Tiedon hankinnan tuen kehittäminen on tutkimuksen perusteella yksi ohjelmistojen kriittisimmistä kehityskohteista. Nykyiset yritysten tietojärjestelmät eivät useinkaan kerää toimintolaskennan vaatimaa tietoa, kuten esimerkiksi henkilöiden ajankäytön seurantaa. Tästä syystä toimintojohtamisohjelmistojen tulisi itsessään tukea tämänkaltaisen tiedon keräämistä.

Toimintojohtamisen perusajatuksista toiminto- ja asiakaskannattavuusanalyysejä ohjelmistot tukivat parhaiten. Vastaavasti puutteita löytyi toimintoperusteisen budjetoinnin tuesta sekä prosessien mallintamisesta.

Johtamisen tietojärjestelmien yleisistä ominaisuuksista tiedon käsittelyominaisuudet ja käyttäjäominaisuudet oli pääosin toteutettu hyvin. Puutteita löytyi raportointi- ja tiedon analysointiominaisuuksista.

Eri käyttäjien mielipiteet eivät merkittävästi poikenneet toisistaan. Jopa eri ohjelmistojen kohdalla vaihtelu oli melko vähäistä. Käyttäjät olivat suurimmaksi osaksi tyytyväisiä oman ohjelmistonsa ominaisuuksiin.

Toimintolaskenta ei ole uusi asia suomalaisissa yrityksissä. Toimintolaskennan hyödyntämisessä on sitä tukevan ohjelmiston käyttäminen askel kohti jatkuvatoimista ja muihin tietojärjestelmiin integroitua toimintolaskentajärjestelmää. Tutkimuksen perusteella oltiin yhteydessä noin 30:een toimintolaskentaa hyödyntävään yritykseen. Näistä kuitenkin vain pieni osa käytti toimintojohtamisohjelmistoaan aktiivisesti. Tästä voidaan tehdä johtopäätös, että Suomessa ei olla vielä siinä vaiheessa, että hyödynnettäisiin integroituja toimintojohtamisjärjestelmiä. Tätä osoittaa myös saatu tutkimustulos, jonka mukaan toimintojohtamisohjelmistojen hyödyntäjät olivat pääasiassa kustannuslaskennan kehittäjiä eivätkä ylimpään johtoon kuuluvia henkilöitä.

Tulevaisuudessa toimintojohtamisohjelmistojen käyttäjäryhmien profiili tulee muuttumaan niin, että käyttäjinä on nykyistä enemmän päätöksentekijöitä. Toinen tulevaisuuden kehityssuunta lienee on, että toimintojohtamisohjelmistojen pohjalta tullaan rakentamaan yrityksiin toimintojohtamisjärjestelmiä, jotka on integroitu yrityksen muihin tietojärjestelmiin. Tämä tuo mukanaan uusia tutkimushaasteita, koska toimintopohjaisen ja prosessiajattelun mukaisen informaation kerääminen nykyisistä funktionaalisisista tietojärjestelmistä ei ole ongelmaton. Vielä tätä rakentavampia ja eteenpäin suuntautuneempia tutkimusnäkökulmia ovat prosessi- ja toimintoperusteisen ajattelun mukanaan tuomat vaatimukset tulevaisuuden tietojärjestelmiä kehitettäessä.

**LÄHDELUETTELO**

- Ahituv N., Neumann S., Decision making and value of information. Tel-Aviv University, 1987
- Albright T.L., Software for Activity-Based Management. *Journal of Cost Management*, Spring 1995, s. 6 - 22
- Bailey J., Implementation of ABC systems by UK companies: Management accounting, February 1991, s. 30 - 32
- Ball G. L., The User Expectations - The CEO's Perspective. (in Watson H.J., Rainer R.K., Houdeshel G. ed.) *Executive Information Systems: emergence development impact*. John Wiley & Sons, 1992, 353 s
- Barrow C., Implementing an Executive Information System: Seven Steps for Success. *Journal of Information Systems Management*, Vol. 7, No. 2, Spring 1990
- Benke R. L., Teaching activity-based costing. *Management accounting*, August 1992, s. 61 - 62
- Bensabat I., Goldstein D., Mead M., The Case Research Strategy in Studies of Information Systems, *MIS Quarterly*, Vol. 11, No. 3, September 1987, s. 369-386
- Bergeron F., Raymond L., Evaluation of EIS from a Managerial Perspective. *Journal of Information Systems*, No. 2, 1992, s. 45 - 60
- Borden J., Cost Management Software. *Journal of Cost Management*, Fall 1991, s. 7 - 37
- Borden J., Cost Management Software. *Journal of Cost Management*, Winter 1994, s. 39 - 49
- Brimson J. A. (suom. Riistama V. ja Lydman K.), Toimintolaskenta - activity-based accounting. Weiling+Göös, *Ekonomia-sarja*, Jyväskylä, 1992, 288 s
- Campi J.P., It's Not as Easy as ABC. *Journal of Cost Management*, Summer 1992, s. 4 - 10
- Carlisle J. P., Alameddine K.D., A Study Evaluating Existing Executive Information Systems Products. *Proceedings of the Twenty-third Annual Hawaii International Conference on System Science* (ed. Nunamaker J.F.), Kailua-Kona, Hawaii, January 2.-5., 1990

- Carlson E., Sullivan-Trainor M.E., Pizzano R., Product Spotlight on EIS. (in Watson H.J., Rainer R.K., Houdeshel G. ed.) Executive Information Systems: emergence development impact. John Wiley & Sons, 1992, 353 s
- Codd E.F., Codd S.B., Salley C.T., Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate, 1993, (White paper)
- Cooke S., Slack N., Making management decisions. Prentice Hall, 1984, 422 s
- Cooper R., ABC: The Right Approach for You? Accountancy, January - February 1991, s. 70 - 72
- Cooper R., Kaplan R. S., Activity-Based Systems: Measuring the Costs of Resource Usage. Accounting Horizons, Vol. 6, No. 3, September 1992, s. 1 - 13
- Cooper R., Kaplan R. S., From ABC to ABM. Management Accounting, November 1992, s. 54 - 57
- Cooper R., Kaplan R. S., Measure Costs Right: Make the Right Decisions. Harvard Business Review, September - October 1988, s. 96 - 103
- Cooper R., Kaplan R. S., Profit priorities from Activity-based costing. Harvard Business Review, May - June 1991, s. 130 - 135
- Cooper R., You Need a New Cost System When... Harvard Business Review, January - February 1989, s. 77 - 82
- CostControl, Net Management Tool 1.5, Quality Production & Research, Oulu, 1995, Ohjelmistomanuaali
- Davenport T.H., Short J.E., The New Industrial Engineering: Information Technology and Business Process Redesign. Sloan Management Review, Summer 1990, Vol. 31, No. 4, s. 11 - 27
- Davis G. B., Olson M. H., Management information systems: conceptual foundations, structure and development. McGraw-Hill, New York, 1985, 681 s
- Dos Santos B.L., Holsapple C.W., A Framework for Designing Adaptive DSS Interfaces. Decision Support Systems Vol. 5., No. 1., 1989, s. 1 - 11
- Drury C., Activity-based costing. Management Accounting, September 1989, s. 60-66
- Drury C., Management and Cost Accounting. Chapman & Hall, London, UK, 1994, 874 s

Easy ABC, Training material for Easy ABC, ABC Technologies Ltd, 1996, Beaverton, USA

Fahey B.G., Building an ABC Datawarehouse. *Management Accounting*, March 1996, s. 33 - 36

Geishecker M.L., New Technologies Support ABC, 1996, *Management Accounting*, March 1996, s. 42 - 48

Goldstein M.H., Executive Information Systems (part I), *National Productivity Review*, No. 7, 1988, s. 353 - 355

Gorry G.A., Scott Morton M.S., A Framework for Management Information Systems. *Sloan Management Review*, Vol. 13, No. 1, 1971, s. 55 - 70

Greenwood T.G., Reeve J.M., Activity-based Cost Management for continuous Improvement: A Process Design Framework. *Journal of Cost Management*, Winter 1992, s. 22 - 40

Gwynne R., Ashworth G., Implementing activity-based management at Mercury Communications. *Management accounting*, December 1993, s. 34 - 36

Hackathorn R., Data Warehousing Energizes Your Enterprise. *Datamation*, Vol. 41, No. 2, February 1995, s. 38 - 42

Hogue J. T., A framework for the examination of management involment in Decision Support Systems. *Journal of Management Information Systems*, Vol. 4, No. 1, 1987, s. 96 - 110

Houdeshel G., Watson H.J., The Management Information and Decision Support (MIDS) System at Lockheed-Georgia. *MIS Quarterly*, Vol. 11, No. 1, March 1987

Johnson H. T., Vance T. P., Player S. R., Pitfalls in Using ABC Cost-Driver Information to Manage Operating Costs. *Corporate Controller*, January - February 1991, s. 26 - 32

Kee R., Integrating Activity-based costing with the theory of constraints to enhance production-related decision-making. *Accounting Horizons*, December 1995, Vol. 9, No. 4, s. 48 - 61

Keen P. G. W., Value analysis: Justifying Decision Support Systems. *MIS Quarterly*, Vol. 5, No. 1, March 1981

Keen P. G. W., Scott Morton M., *Decision Support Systems: An Organization Perspective*, Addison Wesley, Reading, 1978

- Kraemer K., Dickhoven S., Tierney S., King J., *Datawars: The Politics of Modeling in Federal Policymaking*. Columbia University Press, New York, 1987
- Laska R., Paller A., *What Users Want Today*, (in Watson H.J., Rainer R.K., Houdeshel G. ed.) *Executive Information Systems: emergence development impact*. John Wiley & Sons, 1992, 353 s
- Lee A.S., *A Scientific Methodology for MIS Case Studies*. *MIS Quarterly*, March 1989, s. 33 - 50
- Lumijärvi O.-P.(toim.), *Toimintojohtaminen - activity based managementin suomalaisia sovelluksia*. Weiling+Göös, *Ekonomia-sarja*, Jyväskylä, 1993, 199 s
- Lyne S., *Accounting Measures, Motivation and Performance Appraisal*.(in Ashton D., Hopper T., Scapens R.W. ed.) *Issues in Management Accounting*, Prentice Hall, New York, 1991, 321 s
- Miller J.A., *The Best Way to Implement an Activity-Based Cost Management System*. *Corporate Controller*, September - October 1990, s. 8 - 13
- Millet I., Mawhinney C.H., *EIS versus MIS: A Choice Perspective*. *Proceedings of HICCS-23, 23rd Hawaii International Conference on System Sciences*, Kailua-Kona, Hawaii, January 3. - 6., 1990, s. 202 - 209
- Millet I., Mawhinney C.H., Kallman E.A., *A Path Framework for Executive Information Systems*. (in Watson H.J., Rainer R.K., Houdeshel G. ed.) *Executive Information Systems: emergence development impact*. John Wiley & Sons, 1992, 353 s
- Minzberg H., *The Manager's Job: Folklore and Fact*. *Harvard Business Review*, July - August, Vol. 53, No. 4, 1975
- Morrow M., Connolly T., *Activity-Based Budgeting. The Emergence of Management Accounting*, February 1991
- O'Guin M. C., *The complete guide to activity-based costing*. Englewood Cliffs, Prentice Hall, New York, 1991
- Paller A., Laska R., *The EIS Book*. Dow Jones-Irwin, Homewood, IL, 1990
- Partanen K., *EIS-ohjelmistot ja -järjestelmät sekä niiden hyväksikäyttö suuryrityksissä*. Pro gradu -tutkielma, Tietojenkäsittelyopin laitos, Jyväskylän yliopisto, 1992
- Pemberton N.R., Arumugam L., Hassan N., *From Obstacles to opportunities*. *Management Accounting*, March 1996, s. 20 - 32

- Pienimäki T., Eräkoon vaikutus kannattavuuteen makeisten valmistuksessa. Diplomityö, Lappeenrannan teknillinen korkeakoulu, Lappeenranta, 1995
- ProDaCapo, Programmet som analyserar ditt företag, Rubicon Produktivitet-utveckling AB, Örnsköldsvik, 1996
- Raffish N., Turney P. B. B., Glossary of activity-based management. Journal of Cost Management, Fall 1991, s. 53 - 63
- Ray M. R., Sclic T. W., Activity-based management of innovation and R&D operations. Journal of Cost Management, Vol. 6, No. 4, Winter 1993, s. 16 - 22
- Riistama V., Jyrkkiö E., Operatiivinen laskentatoimi, Weilin+Göös, Gummerus Kirjapaino Oy, Jyväskylä, 1994
- Rockart J.F., Treacy M.E., The CEO Goes On-Line. Harvard Business Review, January - February 1982
- Roth H.P., Borthick A.F., Getting closer to real product costs. Management Accounting, May 1989, s. 28 - 33
- Sheridan T., Don't count your costs, manage them. Management Accounting, February 1989, s. 20 - 24
- Silver M.S., Systems That Support Decision Makers Description and Analysis, John Wiley & Sons, West Sussex, 1991
- Simprocess User's Manual, 1996
- Sprague R. H., A Framework for the development of Decision Support Systems. MIS Quarterly, Vol. 4, No. 4, December 1980
- Sprague R. H., Watson H. J., Decision Support Systems: Putting theory into practice. Englewood Cliffs, New Jersey, 1993, 435 s
- Swenson D.W., Flesher D.L., Are You Satisfied With Your Cost Management ? Management Accounting, March 1996, s. 49 - 53
- Swenson K.D., Maxwell R.J., Matsumoto T., Saghari B., Irwin K., A Business Process Environment Supporting Collaborative Planning. The Journal Of Collaborative Computing, Vol. 1, No. 1, Spring 1994, s. 1 - 13
- Thierauf R.J., Creative Computer Software for Strategic Thinking and Decision making, Quorum books, Wesport, 1993

Turban E., Watson H.J., Integrating Expert Systems, Executive Information Systems and Decision Support Systems. DSS '89 Transactions, The Institute of Management Sciences, s. 37 - 45

Turney P. B. B., Common cents: the ABC performance breakthrough. Cost Technology, Hillsboro (Or.), 1991

Turney P. B. B., Activity-Based Management. Management Accounting, January 1992a, s. 20 - 25

Turney P. B. B., What an activity-based cost model looks like. Journal of Cost Management, Winter 1992b, s. 54 - 60

Turney P. B. B., Reeve J. M., The impact of continuous improvement on the design of activity-based cost systems. Journal of Cost Management, Summer 1990, s. 43 - 50

Ward T., Patel K., ABC - a framework for improving shareholder value. Management Accounting, July - August 1990, s. 34 - 36

Watson H.J., Rainer R.K., Houdeshel G. (ed.), Executive Information Systems: emergence development impact. John Wiley & Sons., 1992, 353 s

Watson H.J., Rainer R.K., Koh C.E., Executive Information Systems: A framework for Development and a Survey of Current Practices. MIS Quarterly, Vol. 15, No.1, March 1991

Vehmanen P. Toimintolaskenta yrityksen johtamisessa. Liiketaloustieteellinen aikakauskirja, No. 3, 1994, s. 329 - 337

WfMC (Workflow Management Coalition), Interface 1: Process Definition Interchange. Document Number WfMC TC-1016, Issued on May 29, 1996

WfMC (Workflow Management Coalition), The Workflow Reference Model. Document Number WfMC TC00-10003, Issued on November 29, 1994



## TOIMINTOLASKENTA- JA TOIMINTOJOHTAMISOHJELMISTOJEN ARVIOINTI

**Vastaaajan tiedot: (Näitä tietoja käytetään vain mahdolliseen vastaaajan myöhempään tavoittamiseen)**

Nimi: \_\_\_\_\_

Asema: \_\_\_\_\_

Yritys: \_\_\_\_\_

---

### Taustatiedot:

#### Yrityksen liikevaihto

- 0-100 Mmk  
 101-300 Mmk  
 301-800 Mmk  
 801- 1 400 Mmk  
 yli 1 400 Mmk

#### Henkilöstö

- 0-200 hlöä  
 201-500 hlöä  
 501-1 000 hlöä  
 1 001- 3 000 hlöä  
 yli 3 000 hlöä

#### Toimiala

- metalliteollisuus  
 puunjalostusteollisuus  
 muu teollisuus  
 pankki ja rahoitus  
 palveluala

#### Arvioitava ohjelmisto:

- Cost Control       Easy ABC Plus       Easy ABC Oros   
 Pro DaCapo       Muu, mikä \_\_\_\_\_

## 1. Toimintolaskennan perusajatusten tukeminen

1.1. Miten hyvin ohjelmisto mielestäsi tukee toimintolaskentaketjun (resurssit, toiminnot, tuotteet) resurssit-osan muodostamista ja hallintaa?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

1.2. Miten hyvin ohjelmisto mielestäsi tukee toimintolaskentaan liittyvää resurssikulutusmalli-ajatusta eli ylikapasiteetin käsittelyä?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

1.3. Miten hyvin ohjelmisto mielestäsi tukee toimintolaskentaketjun (resurssit, toiminnot, tuotteet) toiminnot-osan muodostamista ja hallintaa?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

1.4. Miten hyvin ohjelmisto mielestäsi tukee toimintolaskentaketjun (resurssit, toiminnot, tuotteet) tuotteet-osan muodostamista ja hallintaa?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

1.5. Miten ohjelmisto mielestäsi tukee resurssi- ja toimintoajureiden muodostamista sekä hallintaa?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

1.6. Miten ohjelmisto mielestäsi tukee ajankäyttötietojen keräämistä ihmisiltä (esimerkiksi valmis kyselylomakepohja tai mahdollisuus poimia toimintoja listasta ja muodostaa niistä kyselylomake)?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

## 1.7. Onko eri toimintomallien yhdistäminen ohjelmiston avulla mielestäsi

Erittäin vaikeaa					Erittäin helppoa	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

## 2. Toimintojohtamisen perusajatusten tukeminen

## 2.1. Ohjelmiston antama tuki asiakaskannattavuuksien tarkasteluille?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

## 2.2. Miten hyvin ohjelmisto mielestäsi tukee toimintoperusteista budjetointia?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

## 2.3. Miten hyvin ohjelmisto mielestäsi tukee prosessiajattelua ja prosessien mallintamista?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

## 2.4. Miten hyvin ohjelmisto mielestäsi tukee prosessien kustannusten ja pullonkaulakohtien määrittämistä?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

## 3. Yleiset ominaisuudet

## 3.1. Miten hyvin ohjelmisto mielestäsi tukee siihen syötettyjen tietojen päivittämistä(mm. tilien saldot, ajankäyttötiedot jne.)?

Erittäin huonosti					Erittäin hyvin	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

3.2. Ohjelmiston raportointiominaisuudet (mm. raporttien monipuolisuus, takoituksenmukaisuus, räätälöintimahdollisuudet) ovat mielestäni

Erittäin huonot					Erittäin hyvät	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

3.3 Tiedon siirto-ominaisuudet ohjelmistoon ja ohjelmistosta ulos ovat mielestäni

Erittäin huonot					Erittäin hyvät	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

3.4. Ohjelmiston graafit ovat mielestäni

Erittäin huonot					Erittäin hyvät	En ole käyttänyt ko. ominaisuutta
1	2	3	4	5		<input type="checkbox"/>

3.5. Ohjelmiston käytön oppiminen oli mielestäni

Erittäin vaikeaa					Erittäin helppoa
1	2	3	4	5	

3.6. Ohjelmiston käyttäminen on mielestäni

Erittäin vaikeaa					Erittäin helppoa
1	2	3	4	5	

3.7. Mitä mieltä olet ohjelmiston käyttöliittymän ja näyttöjen selkeydestä?

Erittäin epäselkeitä					Erittäin selkeitä
1	2	3	4	5	

3.8. Ohjelmiston käyttöohjeet ovat mielestäni

Erittäin huonot					Erittäin hyvät
1	2	3	4	5	

**KIITOKSET VAIVANÄÖSTÄSI !!!**

## OHJELMISTOJEN TUOTTAMAT RAPORTIT

## Easy ABC raportit

Raportti	Kuvaus
Assignments	Kohdistusraportti
Attribute costs	Attribuuttikustannusraportti
Attributes	Raportti attribuuttien määrällisistä tiedoista
Bill of costs	Laskentakohteiden kustannusraportti
Contributions	Raportti tileistä ja niillä olevista kustannustiedoista
Drivers	Ajuriraportti
Idle capacity	Ylijäämäkapasiteettiraportti
Module Hierarchy	Hierarkiaraportti resursseista, toiminnoista ja laskentakohteista
Notes	Raportti ajanjaksotietojen sanallisista selityksistä
Performance measures	Tehokkuuden mittausraportti
Resource contributions	Raportti resurssien kulutuksesta
Unit costs	Yksikkökustannusraportti
Unit usage	Raportti yhden yksikön vaatimista panoksista suhteessa kaikkiin vaadittuihin panoksiin
Unassigned Costs	Raportti kohdistamattomista kustannuksista

## CostControl raportit

<p><b>Päänäyttö</b></p> <p>Toimintojen yksikkökustannukset</p> <p>Tukitoimintojen käyttö</p> <p>Toiminnon kustannukset</p> <p>Toiminnon perustiedot</p> <p>Toimintojen investointikustannukset</p> <p>Tuotteiden käyttämät toiminnot</p> <p>Toiminnon käyttö</p>	<p><b>Toiminnon tiedot</b></p> <p>Kustannusrakenne</p> <p>Kumulatiiviset kustannukset</p>
<p><b>Resurssinäyttö</b></p> <p>Resurssit</p> <p>Resurssien kustannustekijät</p> <p>Resurssien käyttö</p> <p>Kustannustekijät</p> <p>Valitut kustannustekijät</p>	<p><b>Markkina-alueet</b></p> <p>Markkina-alue luettelo</p> <p>Rakenne</p> <p>Tuotekustannukset markkina-alueittain</p> <p>Tuloslaskelma</p> <p>Tuote tuloslaskelma</p>
<p><b>Tuotenäyttö</b></p> <p>Rakenne</p> <p>ABC-kate</p> <p>Tuotteen tuloslaskelma</p> <p>Tuotteen kumulatiiviset kustannukset</p> <p>Tuotehierarkia</p>	<p><b>Tehokkuus</b></p> <p>Toimintojen tehokkuus</p> <p>Tuloslaskelma</p> <p>Tuloslaskelma toiminnoittain</p> <p>Tuotemixin käyttämät toiminnot</p> <p>Tuloslaskelma kustannustekijöittäin</p>
<p><b>Tarjouskohtainen hinnoittelu</b></p> <p>Tilauskohtainen hinnoittelu -raportti</p>	<p><b>Mittarit näyttö</b></p> <p>Perustiedot</p> <p>Mittarin yksikkökustannukset</p> <p>Luokitellut mittarit</p>
<p><b>Hierarkianäyttö</b></p> <p>Hierarkiaraportti</p>	

## OHJELMISTOJEN GRAAFIT

LIITE 3

## CostControl

<b>Päänäyttö</b> Resurssikustannukset Kokonaiskustannukset Tukitoimintokustannukset Toimintojen investointikustannukset	<b>Tuotenäyttö</b> Toimintokustannukset Kumulatiiviset kustannukset Kustannukset jaksoittain ABC-katteet
<b>Resurssinäyttö</b> Resurssien käyttö	<b>Toiminnon tiedot -näyttö</b> Yksikkökustannukset Kumulatiiviset kustannukset Resurssikustannukset
<b>Markkina-alueet -näyttö</b> ABC tulokset Tulosvertailu Kustannuskertymä Ryhmän kustannuskertymä	<b>Mittarit näyttö</b> Mittarin yksikkökustannukset Luokitellut mittarit Mittarin kapasiteetit

## OHJELMISTOJEN SIIRTOTIEDOSTOT

## EasyABC

Siirtotiedosto	Kuvaus	Jakso- tietoja	Myös budjetti- tietoja
ATTRIBUTE_ HIERARCHY	Attribuuttien hierarkiatietoa		
ATTRIBUTE_QUANTI TIES_ACTUAL	Numeerista attribuuttitietoa	x	x
ATTRIBUTES_FOR_ EACH_ITEM	Kunkin resurssien, toiminnon tai laskentakohteen attribuutit		
BILL_OF_COSTS	Laskentakohteen kustannukset		
BUNDLES COSTS_ACTUAL	Kunkin resurssin kustannukset	x	x
DESTINATION_ SOURCE_ ASSIGNMENTS	Ajurien jäljitystiedot		
DRIVER_QUANTITIES _ACTUAL	Ajurimäärät	x	x
DRIVERS	Toimintomallin ajurit		
EXTERNAL_UNIT_ COSTS_ACTUAL	Ulkoisen tiedon yksikkökustannukset	x	x
ITEMS_FOR_EACH_ ATTRIBUTE	Kuhunkin attribuuttiin liittyvät tiedot		
MODEL_NOTES	Toimintomallin sanalliset kuvaukset		
OUTPUT_QUANTI- TIES_ACTUAL	Tilien yksikkömäärät	x	x
PERIOD_NOTES	Jaksokohtaiset tiedot	x	
SOURCE_DESTINATI ON_ASSIGNMENTS	Ajurien kohdistustiedot		
STRUCTURES_BY_LE VEL	Mallin tiedot tason mukaan jaoteltuna		
STRUCTURES_BY_ REFERENCE	Mallin tiedot tunnuksen mukaan jaoteltuna		



**CostControl**

<b>Siirtotiedosto</b>	<b>Kuvaus</b>
Kustannustekijät	Resurssien kustannukset
Resurssit	Resurssien suoritelmäärät
Toimintotiedot	Toiminnon tunnus, suoritelmäärä, käyttötunnit/jakso
Toimintomalli	Toiminnon nimi, vastuuhenkilö, suoriteyksikkö ja -määrä, käyttötunnit/jakso, kuvaus
Toiminto ja isätoiminto	Toiminnon paikka toimintohierarkiassa
Tuotetiedot	Tuotteen tunnus, nimi, kuvaus, materiaalikustannukset, eräko, markkinahinta
Tuotteen käyttämät toiminnot	Toiminnot, joita tuote käyttää
Tuotteen komponentit	Komponentit, joita tuotteeseen kuuluu
Tuotemix	Tuotteen myyntimäärä, tuotantoerien lukumäärä, myyntihinta
Toiminnon käyttämät resurssit	Resurssit, joita toiminto käyttää
Markkina-alueet	Markkina-alueen nimi, kuvaus, myynnit ja ostot
Markkina-alueen toiminnot	Toiminnot, joita markkina-alue käyttää
Markkina-alueen tuotteet	Tuotteet, joita markkina-alueella myydään
Markkina-alueen tuotteen toiminnot	Markkina-alueen tuotteiden toiminnot
Tuoteryhmät	Tuoteryhmät, joihin tuotteet kuuluvat
Markkina-alueryhmät	Markkina-alueryhmät, joihin markkina-alueet kuuluvat
Osamallit	Osamallit, joihin toiminnot kuuluvat
Tuotteen materiaalikustannukset	Tuotteeseen kuuluvien materiaalien kustannukset

**ProDaCapo**

<b>Siirtotiedosto</b>	<b>Kuvaus</b>
Accounts	Käytettävät tilit
Accountgrouping	Tilien ryhmittely
Resources	Resurssien tunnus, nimi ja suoritelmäärät
Cost	Tileillä olevat kustannukset
Trace information	Ajuritiedot resurssilta toiminnoille ja toiminnoilta tuotteille
Products	Tuotteen tunnus, nimi, kuvaus, materiaalikustannukset, eräko, markkinahinta
Productstructure	Komponentit, joita tuotteeseen kuuluu
Customer	Asiakkaan tunnus, nimi ja kuvaus
Customerstructure	Markkina-alueryhmät, joihin asiakkaat kuuluvat
Customersales	Asiakkaan myynnit ja ostot