

Opinnäytetyö (AMK)

Tietojenkäsittelyn tradenomi

2020

Jani Laamanen

RAHOITUSYHTIÖN  
TIETOJÄRJESTELMÄN  
INTEGRAATIO  
YHTEISTYÖKUMPPANIN  
KANSSA

OPINNÄYTETYÖ (AMK) | TIIVISTELMÄ

TURUN AMMATTIKORKEAKOULU

Tietojenkäsittelyn tradenomi

2020 | 24 sivua

Jani Laamanen

# RAHOITUSYHTIÖN TIETOJÄRJESTELMÄN INTEGRAATIO YHTEISTYÖKUMPPANIN KANSSA

Opinnäytetyön tehtävänä oli integroida uusi maksuvalvoja Lähirahoitus Oy:n rahoitus järjestelmään. Integraatiossa tietojen välitys tapahtuu XML tiedostojen kautta, joita välitetään osapuolille SFTP protokollalla. Tietojärjestelmien välillä liikkuvat lasku-, update- sekä status-aineistot sekä laskujen kuvat.

Koska opinnäytetyössä tehdään integraatio, käydään sen toteuttamista varten läpi erilaisia metodeja. Toteutus suoritettiin point-to-point-metodilla.

Tuloksena yrityksen järjestelmä eritteli laskut eri maksuvalvojille lähetettäväksi ja vastaanotti maksuvalvojan lähettämiä aineistoja. Vastaanotettujen aineistojen sisällöt kirjataan järjestelmään tarpeiden mukaan ja niiden vaatimalla tavalla.

ASIASANAT:

SFTP, Finvoice, Coldfusion, XML

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Business Information Technology

2020 | 24 pages

Jani Laamanen

# INTEGRATION OF NEW COLLABORATOR INTO FINANCE COMPANY INFORMATION SYSTEM

The goal of the thesis was to integrate a new payment controller into Lähirahoitus Inc's information system. Communication between the systems was managed with XML files that were sent via SFTP protocol. Information shared between the systems includes invoice, update and status messages.

A proper method to perform the integration was needed and a few different methods are presented. The method used in the project ended up being the point-to-point method.

As a result of the thesis, the system could sort invoices to be sent to the correct payment controller and receive messages from the new payment controller. The content of these messages is saved according to the system's requirements.

KEYWORDS:

SFTP, Finvoice, Coldfusion, XML

# SISÄLTÖ

<b>KÄYTETTY SANASTO</b>	<b>5</b>
<b>1 JOHDANTO</b>	<b>6</b>
<b>2 ERILAISIA INTEGRAATIOITA</b>	<b>7</b>
2.1 Point-to-point integraatio	7
2.2 Pystysuuntainen integrointi (Vertical Integration)	7
2.3 Vaakasuora integraatio (Horizontal Integration)	8
<b>3 AINEISTOINTEGRAATIOT</b>	<b>9</b>
3.1 Käytettävä metodi integraation toteutuksessa	9
3.2 Laskuaineistojen määrittäminen ja alkutestaukset	10
3.3 Laskuaineistojen muodostaminen	11
3.4 Päivitysaineistossa lähetettävien tietojen seuraaminen ja kerääminen	11
3.4.1 Lähetettävän aineiston muodostaminen	13
3.5 Statusaineisto	14
3.5.1 Statustietojen seuraaminen ja kirjaus	14
3.6 Suoritusaineisto	16
3.6.1 Tarkoitus	16
3.6.2 Aineiston lukeminen ja suorituksien kirjaus	16
<b>4 PÄIVITYKSET JÄRJESTELMÄÄN</b>	<b>17</b>
4.1 Maksuvalvontaan lähettäminen, seuranta ja vaihtaminen	17
4.2 Suorituksien tilittäminen	19
4.3 Suorituksien välittäminen väärään maksuvalvontaan	20
4.4 Laskujen lajittelu maksuvalvonnan mukaan	20
4.5 Laskun katselmointijärjestelmästä	21
<b>5 YHTEENVETO</b>	<b>23</b>
<b>LÄHTEET</b>	<b>24</b>

## KÄYTETTY SANASTO

Finvoice	Suomessa käytössä oleva suomalaisten pankkien määrittelemä julkinen laskupohjastandardi; Finvoice 3.0 on tästä pohjasta kolmas versio (Finanssi ry 2020)
ColdFusion	Adoben ohjelmointikieli, joka soveltuu verkkosovelluksien kehittämiseen; tunnetaan myös nimellä ColdFusion Markup Language ja lyhenteellä CFML (Adobe 2017)
velallinen	laskun saaja / toimeksiantajan asiakas
toimeksiantaja	yritys, joka on Lähirahoituksen asiakas
SQL-injektio	tietojärjestelmään kohdistuva hyökkäys, jossa hyökkääjä syöttää kantaan haitallista sisältöä hyödyntäen SQL-kyselyjä (OWASP 2020)
SSH File Transfer Protocol	SSH-salattu tiedonsiirtoprotokolla, tunnetaan myös lyhenteellä SFTP (SSH Communications Security 2020)

# 1 JOHDANTO

Yksinkertaistettuna tietojärjestelmien integraatiolla tarkoitetaan erilaisten järjestelmien tai komponenttien yhdistämistä yhdeksi isommaksi kokonaisuudeksi (Lehtonen 2018). Toteutusmetodista riippumatta integraatioiden tavoite on saada komponentit kommunikoimaan keskenään mahdollistaen tiedon siirtämisen paikasta toiseen. Integraatioita voi myös tehdä yrityksen sisäisesti tai kolmansien osapuolien kanssa, kuten tässä tapauksessa.

Integraation toteutusta mietittäessä on oleellista valita oikea toteutusmetodi (FOLIO3 2019). On tärkeää ymmärtää prosessien kulku sekä komponenttien toimivuus keskenään metodia valitessa, sillä sen tulisi mukaila komponenttien välisiä suhteita. Luvussa 2 käydään läpi erilaisia integraatiometodeista ja niiden toimivuutta.

Tässä opinnäytetyössä toteutetaan Lähirahoitus Oy:lle uuden maksuvalvontajärjestelmän integraatio. Työssä valitaan integraatioon soveltuva integraatiometodi ja integraatio toteutetaan valitun metodin mukaisesti. Kyseessä on toiminallinen opinnäytetyö, jonka toimeksiannon on antanut Lähirahoitus Oy.

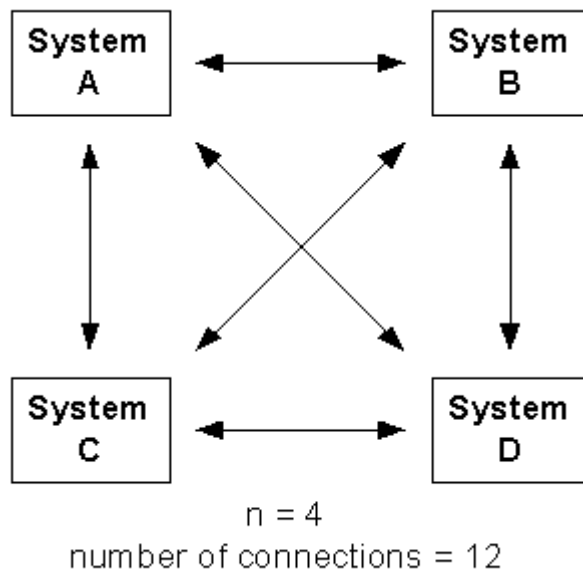
Lähirahoitus on turkulainen rahoituspalveluja tarjoava yritys. Se aloittaa yhteistyötoiminnan Intrum Oy:n kanssa. Intrum on pohjoismainen yritys, joka tarjoaa perintä-, laskutus- ja rahoituspalveluita kaiken kokoisille yrityksille. Intrum tulee hoitamaan Lähirahoitukselle laskutusta, maksuvalvontaa ja reskontraa. Tätä varten on luotava tapoja välittää laskujen aineistoja maksuvalvontaa varten sekä mahdollistaa kommunikaatio molempien yritysten tietojärjestelmien välille. Työn tavoitteena on jatkokehittää Lähirahoituksen tietojärjestelmää mahdollistamaan nämä kommunikaatiotarpeet sekä käsittelemään siinä kulkeva data.

Opinnäytetyön rakenne mukailee projektin etenemisvaiheita. Projekti alkoi aineistojen integraatioilla, jotka suoritettiin tämän opinnäytetyön aineistointegraatiot luvun mukaisessa järjestyksessä. Näiden jälkeen hoidettiin pienempiä päivityksiä Lähirahoituksen tietojärjestelmään.

## 2 ERILAISIA INTEGRAATIOITA

### 2.1 Point-to-point integraatio

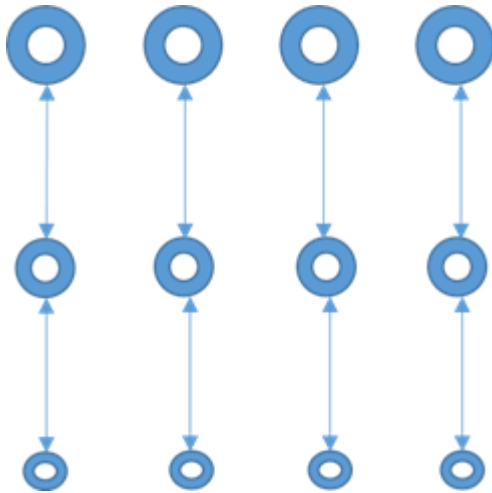
Point-to-point-integraatio sopii silloin, kun tiedoilla on yksi lähettäjä ja yksi vastaanottaja. Järjestelmien tai komponenttien määrän kasvaessa toteutuksen hallittavuus karkaa kärsistä seuraten sääntöä  $n(n - 1)$  tarvittavien yhteyksien määrässä (kuva 1). Opinnäytetyön käytännön osuudessa tehtävä integraatio toteutetaan point-to-point-integraationa, koska integraatio koskee vain kahta tietojärjestelmää.



Kuva 1. Esimerkki point-to-point-integraation yhteyksien kasvamisesta 4 järjestelmällä (Nghiem 2002).

### 2.2 Pystysuuntainen integrointi (Vertical Integration)

Pystysuuntaisessa integraatiossa komponentit linkitetään "siilomaiseen" rakenteeseen, jossa toiminnot muuttuvat sitä monimutkaisemmiksi, mitä ylemmäs rakennetta mennään (kuva 2). Vaikka tämä metodi on yksinkertainen, se on ajoittain joustamaton esim. uutta toimintoa lisättäessä.



Kuva 2. Esimerkki pystysuuntaisesta integraatiosta (FOLIO3 2019).

### 2.3 Vaakasuora integraatio (Horizontal Integration)

Vaakasuorassa integraatiossa jokainen komponentti käyttää samaa rajapintaa linkityksessä. Tämä vähentää tarvittavien yhteyksien määrää, koska komponentit eivät ole suoraan yhteydessä toisiinsa vaan jakavat yhteisen pääkomponentin. Tästä yhdistävästä komponentista käytetään nimitystä Enterprise Service Bus (ESB). ESB-integraatiossa isona etuna on uusien ja vanhojen komponenttien liittäminen keskenään, kun tietojen käsittely ja muuttaminen tapahtuu vain yhdessä paikassa.

Vaikka opinnäytetyössä käytetään point-to-point-metodia, voi Lähirahoituksen kaavoittaa ESB:ksi, sillä se kerää sen eri maksuvalvonnoilta laskun tietoja ja suorituksia sekä välittää niitä vastaanottajan haluamassa muodossa (kuva 3). Lähirahoituksen järjestelmä myös kerää ja käsittelee toimeksiantajien laskuaineistoja samalla tavalla.



Kuva 3. Esimerkki vaakasuorisesta integraatiosta (FOLIO3 2019).

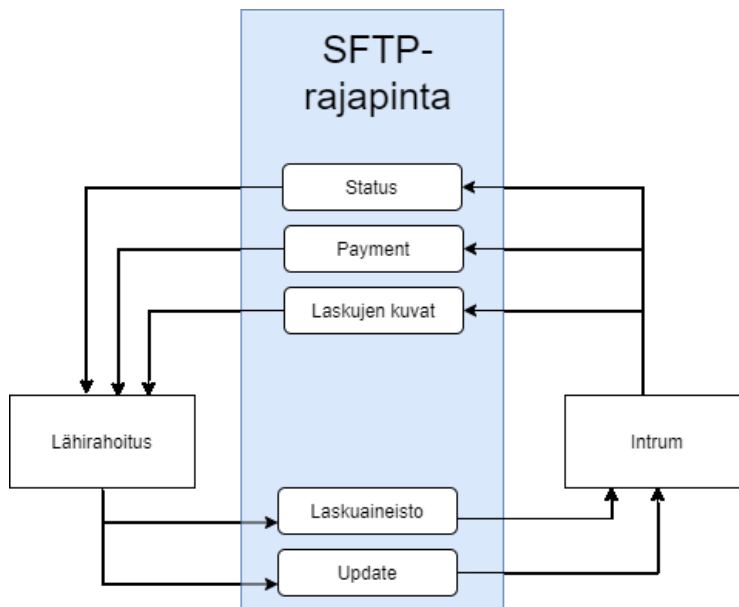


### 3 AINEISTOINTEGRAATIOT

Työn integraatiossa suurin osa on aineistojen välittämistä järjestelmien välillä. Tässä osiossa esitellään jokaisen aineiston tarkoitus sekä sen käsittely. Kaikki aineistojen siirrot suoritetaan PowerShell-skriptien avulla käyttäen SFTP-yhteyksiä.

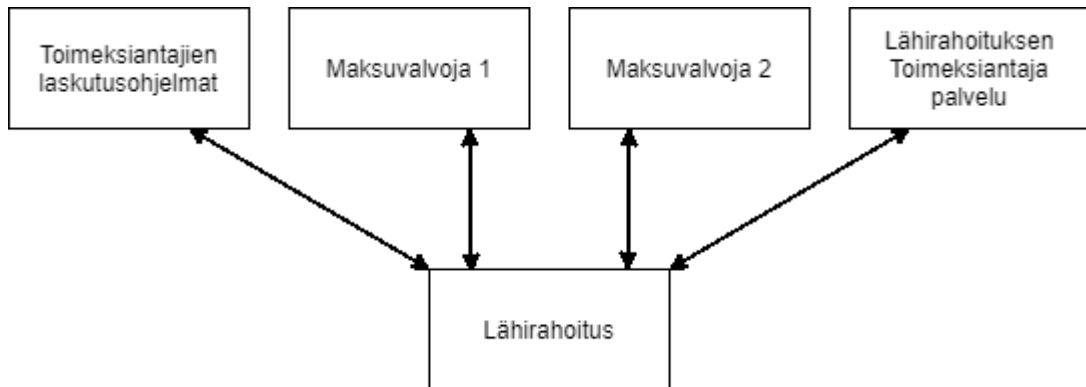
#### 3.1 Käytettävä metodi integraation toteutuksessa

Kuten jo edellä mainittiin, integraatio toteutetaan point-to-point-metodilla. Työssä integroitavia järjestelmiä on kaksi, eikä niiden välillä ole muuta kuin tietojensiirtorajapinta (kuva 4).



Kuva 4. Aineistojen liikkuvuus integraatiossa

Lähirahoituksen toiminnallisuutta katsellen voidaan todeta, että se itse toimii ESB:nä (kuva 5). Se vastaanottaa tietoja useista lähteistä sekä käsittelee niistä tulleita tietoja toisia kokonaisuuden järjestelmiä varten.



Kuva 5. ESB:tä havainnollistava kuva.

### 3.2 Laskuaineistojen määrittäminen ja alkutestaukset

Laskuaineisto muodostetaan XML-tiedostoon, mikä mukailee Finanssiala Ry:n Finvoice 3.0 -pohjaa. Standardisoitu aineistopohja helpottaa laskutietojen välitystä järjestelmien välillä. Vaikka tämä pohja on alalla standardi, on oleellista varmistaa, että aineiston vastaanottajalla sekä lähettäjällä on yhteinen ymmärrys kenttien sisällöstä sekä mahdollisista poikkeusjärjestelyistä. Intrumin järjestelmä vaatii osana sen toimintaa jokaiselle velalliselle oman asiakasnumeron, mikä pidetään yhteydessä toimeksiantajan reskontra-tunnukseen. Kaikki Lähirahoituksen asiakkaat eivät käytä järjestelmissään asiakasnumeroita, joten näissä poikkeustilanteissa asiakasnumeroksi asetetaan Lähirahoituksen kannasta velalliset yksilöivä id-tunniste, joka saa laskuaineistossa etuliitteen, jolla todennetaan asiakasnumeron alkuperä. Etuliitteen avulla tämä asiakasnumero voidaan siten esim. jättää se pois laskulta toimeksiantajan niin halutessa.

Intrum suoritti laskupohjan testausta käytössä olevalla aineistonmuodostusohjelmalla, jonka ohessa ilmeni, että skandinaaviset merkit menivät joissain kohdissa rikki. Uutta aineiston muodostus ohjelmaa tehdessä ilmeni, että edeltäjä tallensi lähetettävään XML-tiedostoon merkkijonon, joka mukaili XML-objektin rakennetta. Vaikka uuden aineiston muodostus toimii samalla tavalla, hajonneita merkkejä ei ilmentynyt enää, koska samainen merkkijono muutetaan ennen tiedostoon tallentamista XML-objektiksi hyödyntäen ColdFusionin xmlParse -funktioita.

### 3.3 Laskuaineistojen muodostaminen

Uusi laskuaineiston muodostusohjelma on rakennettu vanhan pohjalta poistaen mittavia määriä tarpeetonta koodia. Tähän lukeutui mm. asiakkuutensa päättäneiden yksilölliset muutokset. Turvallisuutta on lisätty tekemällä laskutietoja ja laskurivejä hakevista kyselyistä parametrisoituja kyselyitä, suojaten kantaa mahdolliselta SQL-injektioilta (OWASP).

Vanha laskuaineiston muodostusohjelma käytti omatekoista XML:ssä olevien kiellettyjen merkkien korvausfunktiota. Tämä toiminto suoritti samalla rivien pituuden tarkistuksen, mutta koska aineistoa tullaan vielä käsittelemään, on rivien pituuksia turha säädellä tai mitata. W3C:n on määritellyt joukon kiellettyjä merkkejä XML-tietorakenteissa (W3C 2008). Näiden kiellettyjen merkkien korvaukseen käytetään ohjelmointikielen omia funktioita näiden merkkien käsittelyyn kuulumattomissa paikoissa. Kenttien arvoissa syötettävä data ajetaan EncodeForXML-funktion kautta ja XML-attribuuteissa vastaavasti EncodeForXMLAttribute-funktion kautta. Ennen lopullisen aineiston tallentamista siirtoa varten aineisto ajetaan validoinnin läpi, jotta saadaan validi Finvoice 3.0 -aineisto.

### 3.4 Päivitysaineistossa lähetettävien tietojen seuraaminen ja kerääminen

Intrum vastaanottaa päivitysaineistoja, joilla ilmoitetaan muutoksista velallisen tietoihin, laskun tietoihin sekä mahdolliset ohisuoritukset tai hyvityslaskujen kohdistukset. Päivitysaineisto on myös XML-aineisto, jonka sisällön muodon Intrum on itse määrittänyt.

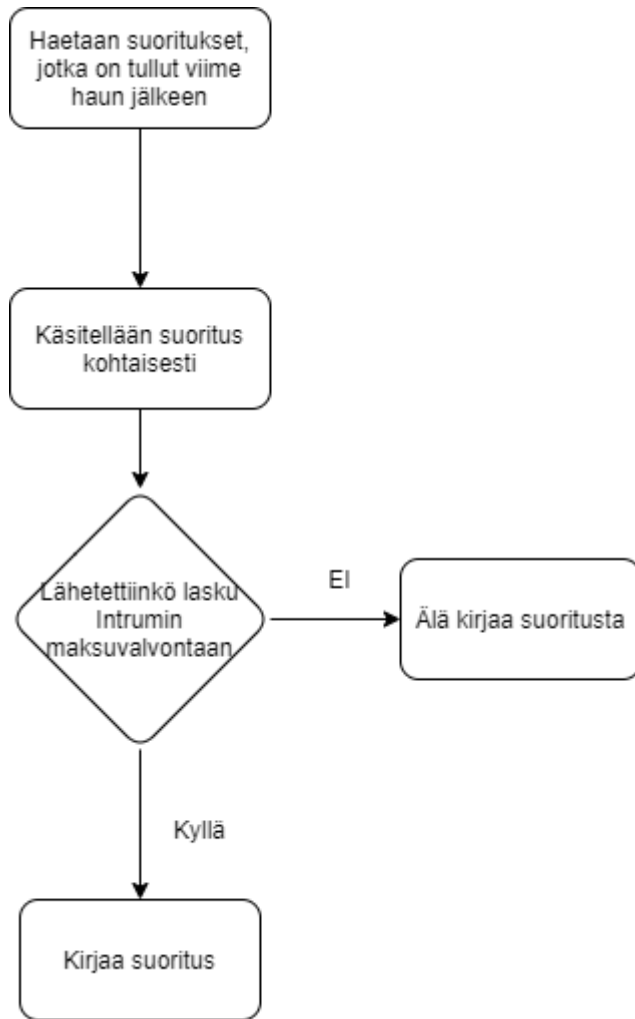
Päivitysaineistossa voidaan päivittää velallisen tietoja mutta tärkeimmät päivitysaineistossa lähetettävät tiedot koskevat ohisuorituksia sekä hyvityssuorituksia.

Intrumiin liittyvää dataa varten luotiin oma tietokanta sekä taulut päivitysaineistossa päivittyvien seuraamista varten. Tietojen keräämisessä oli pidettävä mielessä Intrumin tarpeet, jotta tiedot saadaan oikeille laskuille tai oikeille velallisille, sekä tarvittava määrä Lähirahoituksen omia tunnisteita, että vastaavat asiat voidaan todentaa omasta kannasta omien tarkoituksien mukaisesti. Velallisen tietojen päivitykset saadaan tallennettua velallistietoja seuraavaan tauluun, samalla kun ne päivitetään Lähirahoituksen omaan järjestelmään. Samalla periaatteella saadaan tallennettua laskulle asetettu tai siltä poistettu perinnän esto.

Hyvitys- ja ohisuoritukset haetaan suorituksia seuraavasta SQL-taulusta tähän tarkoitukseen tehdyllä ohjelmalla, joka hyödyntää parametrisoituja kyselyitä. Toimintaperiaatteena tällä ohjelmalla on aina hakea edellisen hakukerran jälkeen tulleet suoritukset, jotka eivät ole tulleet Intrumilta. Kun suoritukset on haettu, niitä tarkistetaan yksi kerrallaan, kenen maksuvalvontaan suoritusta koskeva lasku on lähetetty. Jos se on lähetetty Intrumille, kirjataan suoritukset update-aineiston suoritustietoja seuraavaan tauluun.

Haku kertoja seuraamaan tehtiin SQL taulu, josta haetaan aluksi edellinen ajokerta ja lopuksi kirjataan tämän kertaisen ajon ajankohta. Ohjelma on kaksiosainen tämän jälkeen. Ensin haetaan kirjaustilin sekä suorituksen tyyppin mukaan kaikki hyvityssuoritukset, jotka on tehty edellisen haun jälkeen. Tämän lisäksi kerätään päivitysaineiston vaatimat välttämättömät tiedot, kuten asiakasnumero, laskun numero ja Intrumin määrittämä ID toimeksiantajalle. Kaikkia suorituksia ei kuitenkaan kirjata, sillä tarkoitus on lähettää vain laskusuorituksia niiden toimeksiantajien laskuille, joiden maksuvalvontaa Intrum hoitaa (kuva 6). Näitä asiakkuuksia seurataan taas omassa taulussaan, josta etsitään suoritusrivin omistava toimeksiantaja. Asiakkuuden löydyttyä suoritus kirjataan suorituksia seuraavaan tauluun H-tyypin suorituksena. Tyyppi auttaa määrittämään tilin, jonne suoritus kirjataan.

Ohisuoritusten kirjausprosessi on lähestulkoon identtinen, paitsi rivit tunnistetaan suorituskannasta omalla kenttätunnuksella. Nämä suoritukset kirjataan seurantatauluun O-tyypillä.

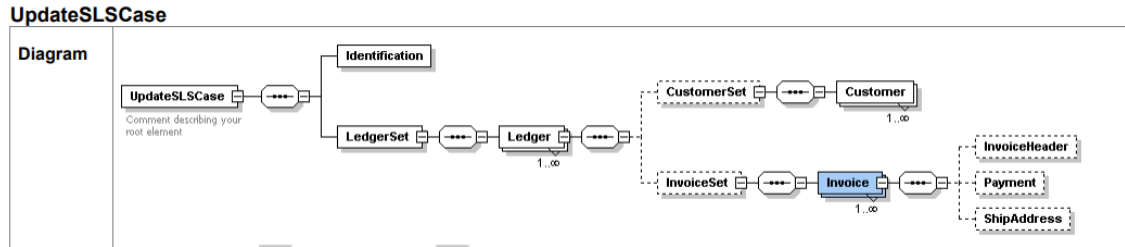


Kuva 6. Suoritusten hakuprosessi update-aineistoa varten.

#### 3.4.1 Lähetettävän aineiston muodostaminen

Päivitysaineistossa viestitään velallis- ja laskumuutokset sekä laskuille tulleet ohi- sekä hyvityssuoritukset toimeksiantajakohtaisesti. Jokaista toimeksiantajaa kohden luodaan Ledger-elementti, johon luodaan CustomerSet- ja InvoiceSet-elementit tarpeiden mukaisesti. CustomerSet sisältää velallisten tietoihin kohdistuvia muutoksia, kuten osoitemuutokset. Nämä muutokset tulevat CustomerSetin alle Customer-alaelementeissä. Jokaiselle velalliselle, jonka tiedot ovat muuttuneet, luodaan oma Customer-elementti tietojen välittämistä varten (kuva 7).

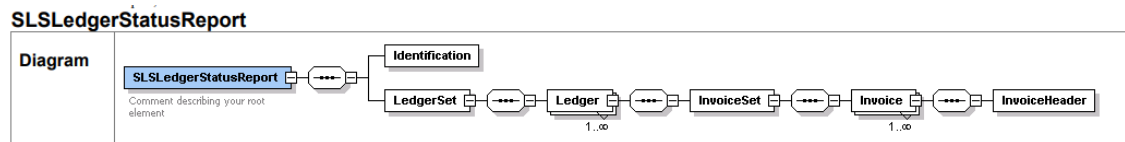
InvoiceSet-elementin Invoice-alaelementit kertovat laskukohtaisista muutoksista sekä laskun suorituksista. Näistä kahdesta huomattavampi kuitenkin on laskulle tulleet suoritukset.



Kuva 7. Päivitysaineiston rakenne (Intrum).

### 3.5 Statusaineisto

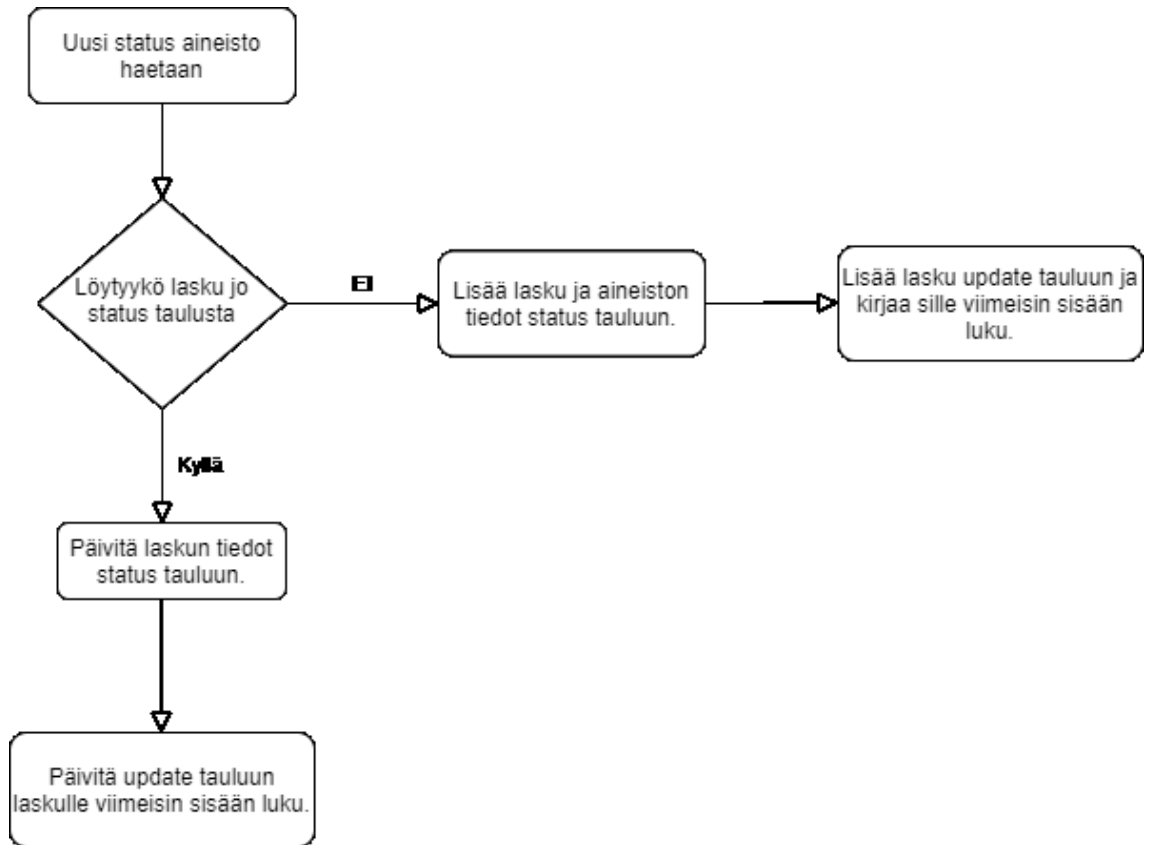
Status-aineisto on Intrumilta sisään luettavaa tietoa, jolla viestitään laskun tilaa Intrumin puolella. Tästä tiedosta Lähirahoitus saa tietää mm. milloin ja kuinka monta muistutusta tai perintäkirjettä laskusta on lähetetty tai paljonko laskulla on avointa pääomaa Intrumin järjestelmän mukaan (kuva 8). Aineistoa käytetään maksuvalvonnan seuraamiseen Lähirahoituksen järjestelmässä, ja sen avulla luodaan relevanteista tiedoista laskulle tapahtumia.



Kuva 8. Statusaineiston rakenne (Intrum).

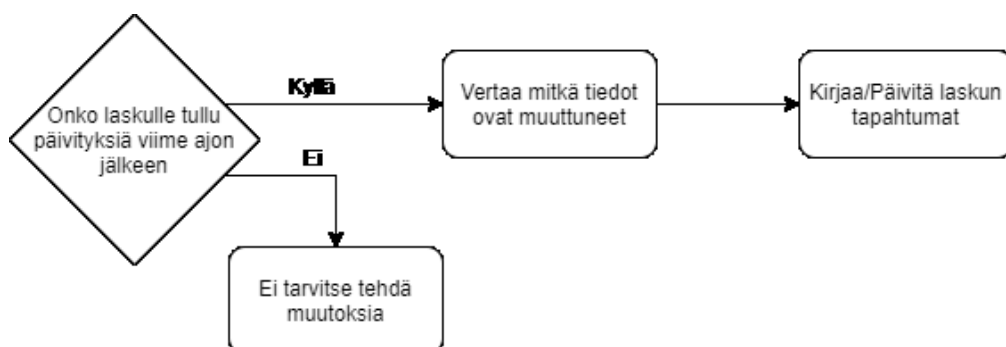
#### 3.5.1 Statustietojen seuraaminen ja kirjaus

Aineiston tietoja seurataan kahdessa taulussa. Ensimmäinen "statustaulu" pitää kirjata viimeisimmistä muutoksista, joista statusaineistossa on ilmoitettu. Nämä tiedot tallennetaan laskukohtaisesti ja päivitetään aina uuden saapuessa. Kun aineisto on luettu, kirjaetaan toiseen "updatetauluun" laskulle viimeisin sisään luettu päivityserä (kuva 9). Statustaulun tietojen avulla luodaan laskulle tapahtumia, mutta taulun tiedot ovat myös näkyvillä laskun näkymässä sen tilan seuraamisen vuoksi.



Kuva 9. Statusaineiston lukemisprosessi.

Updatetaulu seuraa, mitkä Intrumin järjestelmästä luetuista tiedoista on kirjattu Lähira-hoituksen järjestelmään. Taulussa on viimeisin sisään luetun erän tunniste sekä viimeisimmän järjestelmään päivitetyn erän tunniste. Näiden kahden erotessa tiedetään, että laskun tila on muuttunut ja aloitetaan muutosten selvittely ja kirjaus järjestelmään (kuva 10).



Kuva 10. Lasku tapahtumien kirjaus update taulusta.

## 3.6 Suoritusaineisto

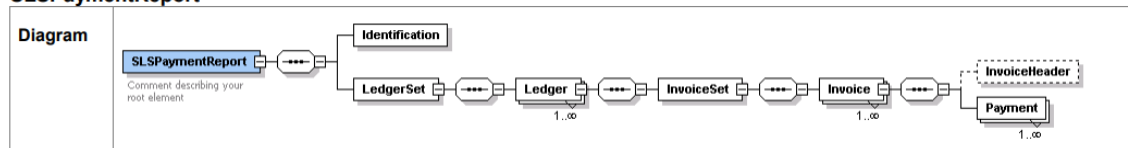
### 3.6.1 Tarkoitus

Suoritusaineistolla Intrum välittää Lähirahoitukselle laskuihin tulleita maksusuorituksia. Näitä ovat tavallisten maksusuoritusten lisäksi mm. hyvitys suoritukset ja ohisuoritukset. Suoritukset kirjataan niiden tyyppin mukaan toimeksiantajakohtaisesti.

### 3.6.2 Aineiston lukeminen ja suorituksen kirjaus

Kun suoritusaineisto luetaan järjestelmään sisään, se käydään läpi toimeksiantajan laskujen mukaisesti. Kukin Payment-elementti sisältää suorituksen tyyppin, mikä määrittää sen, miten se kirjataan laskulle (kuva 11). Esimerkkejä suoritustyypeistä ovat hyvitysuoritus, ohisuoritus sekä tavallinen maksusuoritus.

#### SLSPaymentReport



Kuva 11. Suoritusaineiston rakenne (Intrum).



## 4 PÄIVITYKSET JÄRJESTELMÄÄN

Tässä osiossa tarkastellaan Lahirahoituksen tietojärjestelmään tehtyjä pienempiä muutoksia, jotka olivat integraation kannalta oleellisia.

### 4.1 Maksuvalvontaan lähettäminen, seuranta ja vaihtaminen

Jokaisella laskulla näkyy laskun maksuvalvonnan tila (kuva 12). Tätä osiota piti muokata huomioimaan Intrumin uutena maksuvalvojana. Lasku näkymä tarkistaa päivitetynä, mille maksuvalvonnalle lasku on lähetetty ja näyttää sen tiedot (kuva 13). Tämä vaatii myös toimeksiantajilta tulevan laskuaineiston sisään lukuun muutoksen, jossa laskulle merkitään toimeksiantajan maksuvalvontaa suorittava taho. Tallentamalla tieto maksuvalvojasta laskukohtaisesti mahdollistaa sen, ettei maksuvalvonnan vaihtaminen väärin vanhojen laskujen maksuvalvonta tietoja.



Kuva 12. Laskun maksuvalvontatiedot, kun status aineiston tietoja ei ole vielä saatu laskulle.

Maksuvalvontatiedot			
<b>Maskuvalvonnassa Intrumilla</b>			
Viimeisin tila luettu : 16.07.2020			
Tiedostosta:			
Batch:			
Viite:			
<b>Laskulla avointa</b>	<b>Laskua maksettu</b>	<b>Kertyneet korot</b>	<b>Kertyneet kulut</b>
23.20	0.00	0.00	0.00
<b>Perintä esto</b>	<b>Perintä esto päättyy</b>	<b>Muistutus esto</b>	<b>Laskun siirretty eräpäivä</b>
Ei	-	Ei	07.08.2020
<b>Muistutuksia lähetetty</b>	<b>1. muistutus lähetetty</b>	<b>2. muistutus lähetetty</b>	<b>3. muistutus lähetetty</b>
2	16.06.2020	26.06.2020	-
<b>Perintä kirjeitä lähetetty</b>	<b>1. kirje lähetetty</b>	<b>2. kirje lähetetty</b>	
1	10.07.2020	-	

Kuva 13. Laskun maksuvalvontatiedot, kun status aineisto on vastaanotettu.

Vanhassa järjestelmässä toimeksiantajille pysyi jo vaihtamaan maksuvalvontaa toimeksiantajanäkymässä. Tähän näkymään oli kuitenkin lisättävä Intrum ID-kenttä, kun maksuvalvojana on Intrum (kuva 14).

Kuva 14. Maksuvalvonnan vaihtaminen Intrumiin.

Laskun lähettäminen maksuvalvontaan tapahtui vanhalla järjestelmällä siihen tarkoitettulla ohjelmalla. Vanha ohjelma oli suunniteltu toimimaan vain olemassa olevan maksuvalvojan kanssa, joten sitä ei voinut tähän soveltaa. Intrum myös lisää laskut maksuvalvontaan laskuaineiston sisään lukemisen yhteydessä, joten erilliselle maksuvalvontaan lähettämishjelmalle ei ollut enää tarvetta. Täten laskut voidaan merkitä lähetetyiksi maksuvalvontaan laskuaineiston luonnin yhteydessä.

Uusi ratkaisu ottaa huomioon edellä mainitun maksuvalvontaan lähettämishjelman, joka listaa lähettämiskelpoiset laskut, joita ei ole lähetetty. Maksuvalvonnassa olevien laskujen seurantaohjelmaa oli kuitenkin muokattava. Listauksen on otettava huomioon toimeksiantajan voimassa oleva maksuvalvoja. Lisäksi on olennaista ottaa laskuissa huomioon, mihin maksuvalvontaan ne on lähetetty. Täten listaus ei hae vääriä tietoja toimeksiantajista tai laskuista.

#### 4.2 Suorituksen tilittäminen

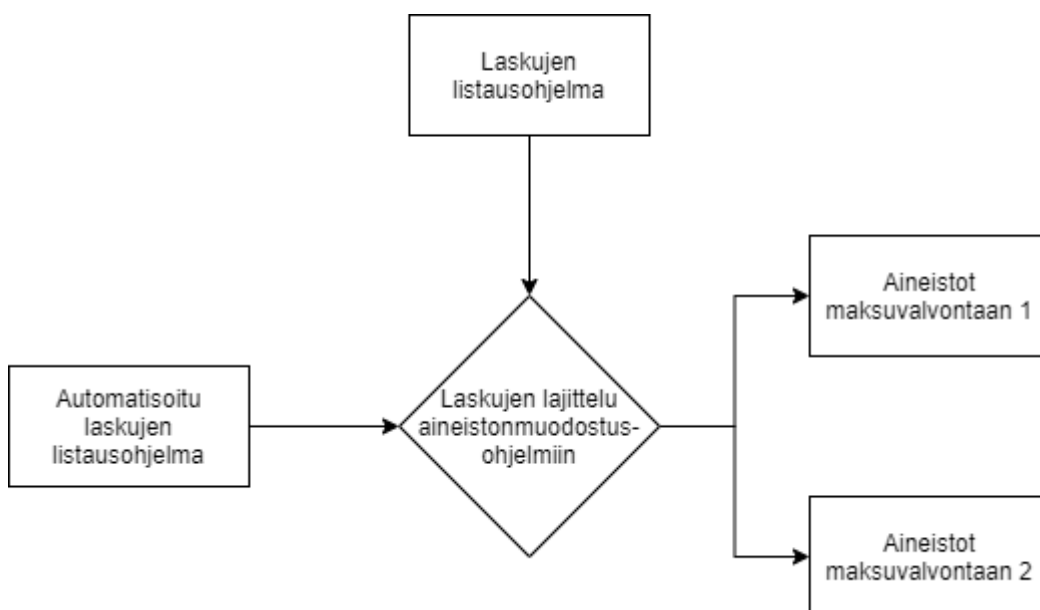
Suorituksen tilittämisellä tarkoitetaan sisään luettujen maksusuorituksen kirjausta ja kohdistamista laskulle. Sama ohjelma myös merkitsee laskut päättyneiksi, mikäli suoritukset kattavat kaiken avoimen pääoman sekä mahdolliset korot ja kulut. Ohjelma muutettiin huomioimaan Intrumille lähetetyissä laskuissa avoimet kulut ja korot eri tavalla. Näille laskuille kirjataan suorituksesta kaikki mahdollinen ylijäämä ja mikäli se ei kata kaikkea, lasku kuitenkin päätetään, jos se ei ole edennyt perintään.

#### 4.3 Suorituksien välittäminen väärään maksuvalvontaan

Olemassa olevalle maksuvalvontaa suorittavalle yritykselle on tehty rajapinta, jolla lähetetään ohi menneitä suorituksia. Näihin lähetettäviin suorituksiin voi lukeutua laskujen suorituksia, jotka eivät kuulu tälle maksuvalvonnalle. Suoritusten lähetystä rajaa kaksi oleellista asiaa. Ensimmäisenä rajapinnan kautta lähetetään laskuja, joiden alkuperä on Lähirahoituksen järjestelmä. Näitä ovat mm. toimeksiantajien ilmoittamat ohi suoritukset tai rahoitusjärjestelmästä tehdyt suoritusmerkinnät. Intrumilta saadut suoritus tiedot eivät lukeudu tähän listaukseen, sillä niiden lähteeksi on merkitty muu kuin Lähirahoituksen oma järjestelmä. Toinen rajaava tekijä on, kun laskua mihin suoritukset kohdistuvat ei löydy vastaanottavan osapuolen järjestelmästä. Kuten luvussa 4.1 mainittiin, Intrum laskuja ei lähetetä toisen maksuvalvojan järjestelmään. Tämä tarkoittaa että vaikka suoritus läpäisisi jonkin tarkistuksen ja suoritus tieto olisi lähtemässä väärään järjestelmään, laskun puuttumisen takia järjestelmä ei vastaanota suoritusta.

#### 4.4 Laskujen lajittelu maksuvalvonnan mukaan

Nyt kun laskuihin merkitään, kenen maksuvalvontaan ne kuuluu lähettää, ne voidaan eritellä oikeisiin aineistonmuodostusohjelmiin. Aiemmin järjestelmä ohjasi laskujen listauksessa valitut laskut suoraan aineiston muodostus ohjelmaan. Tähän oli lisättävä väli ohjelma, joka suorittaa lajittelun (kuva 15).



Kuva 15. Laskujen lajittelu.

#### 4.5 Laskun katselmointijärjestelmästä

Vanhassa järjestelmässä laskuja pystyi esikatsелеmaan Lähirahoituksen omalla laskupohjalla lasku näkymästä. Intrumin asiakkuuksien kanssa tämä metodi ei toimi, koska Intrum lähettää laskut. Kun laskuaineistot on luettu sisään Intrumin järjestelmän puolella, näistä luodut laskujen kuvat välitetään takaisin samaa SFTP-yhteyttä käyttäen kuin muidenkin aineistojen haku. Haetut laskujen kuvat lajitellaan toimeksiantaja kohtaisiin kansioihin, joista ne löydetään nimeämisperusteen avulla ja saadaan avattua samasta näkymästä kuin aikaisemmin. Käyttäjälle ilmoitetaan lasku kuvan saatavilla olemisesta muuttamalla näytä painikkeen ulkoasua (kuvat 16 ja 17).

Laskun kuvaa ei vielä saatavilla

Kuva 16 Laskupainike kuvan puuttuessa

Näytä laskun kuva

Kuva 17 Laskupainike, kun kuva on vastaanotettu

## 5 YHTEENVETO

Lähirahoituksessa oltiin tyytyväisiä työn lopputulokseen. Vielä ennen integraation lisäämistä tuotantoon Lähirahoituksessa suoritetaan testejä ja tehdään hienosäätöä oman järjestelmän toiminallisuuksiin sekä ominaisuuksiin, jotta järjestelmän toiminnot huomioisivat integraation paremmin. Työssä tehdyt suuret linjaukset, kuten aineistojen muodostukset ja aineistojen lukemiset sisään järjestelmään, todettiin toimiviksi sekä molempien osapuolien tarpeet kattaviksi.

Hankaluuksia työssä aiheutti tarvittavan dokumentaation puute ja vajavaisuus. Tämä otettiin integraatiota toteuttaessa huomioon, ja integraatiota dokumentoitiin mahdollisimman paljon aineistoiksi, joita ei voi yrityksen ulkopuolelle jakaa. Dokumentaatio kuvaa integraation osioiden toimivuutta, tarkoitusta sekä vaikutusta kokonaisiin prosesseihin esim. laskujen kulku järjestelmässä.

Projektin aikana selveni entuudestaan, miten tärkeää dokumentaatio on. Integroitavan tietojärjestelmän aineistoista oli selvät sisältö kuvaukset ja tarkoitukset, mikä nopeutti uusien toimintojen kehittämistä. Näiden toimintojen liittäminen järjestelmään hankaloitti kuitenkin dokumentaation puutteellisuus. Asioiden selvittäminen ja uuden tiedon nopea omaksunta osottaituivat tämän takia arvokkaiksi taidoiksi. Myös kommunikointi projektin osallisten kanssa auttoi pahemmissa tilanteissa.

## LÄHTEET

Adobe 2017. CFML Basics. Viitattu: 02.6.2020 <https://helpx.adobe.com/coldfusion/developing-applications/the-cfml-programming-language/elements-of-cfml/cfml-basics.html>.

Finanssiala ry 2020. Finvoice. Viitattu 18.5.2020 <https://www.finanssiala.fi/finvoice/Sivut/default.aspx>.

Finanssiala ry. 2018. FINVOICE 3.0 SOVELTAMISOHJE. Viitattu 18.05.2020 [https://www.finanssiala.fi/finvoice/dokumentit/Finvoice\\_3\\_0\\_soveltamisohje.pdf](https://www.finanssiala.fi/finvoice/dokumentit/Finvoice_3_0_soveltamisohje.pdf).

FOLIO3. 2019. What is System Integration and how do you do it?. Viitattu 27.7.2020. <https://dynamics.folio3.com/blog/system-integration/>.

Lehtonen, Karri 2018. What is system integration?. Viitattu 27.7.2020. <https://www.youredi.com/blog/what-is-system-integration>.

Nghiem, Alex 2002. Web Services Part 6: Models of Integration. Viitattu 27.7.2020. <https://www.informit.com/articles/article.aspx?p=28713>.

OWASP 2020. SQL Injection. Viitattu 12.6.2020 [https://owasp.org/www-community/attacks/SQL\\_Injection](https://owasp.org/www-community/attacks/SQL_Injection).

OWASP 2020. SQL Injection Prevention Cheat Sheet. Viitattu 12.6.2020 [https://cheatsheetseries.owasp.org/cheatsheets/SQL\\_Injection\\_Prevention\\_Cheat\\_Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html).

SSH Communications Security 2020. SFTP – SSH Secure File Transfer Protocol. Viitattu 12.9.2020 <https://www.ssh.com/ssh/sftp/>.

W3C 2008. Extensible Markup Language (XML) 1.0 (Fifth Edition). Viitattu 20.6.2020 <https://www.w3.org/TR/xml/>.