

**Ketterä järjestelmäintegraatio monitoimijaympäristössä,
tapaustutkimus teollisuuden palveluyrityksestä**

Antti Varjoinen

Tampereen yliopisto
Informaatiotieteiden yksikkö
Tietojenkäsittelyoppi
Pro gradu -tutkielma
Ohjaaja: Mikko Ruohonen
Toukokuu 2014

Tampereen yliopisto
Informaatiotieteiden yksikkö
Tietojenkäsittelyoppi
Tekijän Nimi: Antti Varjoinen
Pro gradu -tutkielma, 53 sivua
Toukokuu 2014

Tutkimuksessa seurattiin järjestelmäintegraatioprojektia, jossa tavanomaisista toimintatavoista poikkeavasti integraatiotoimittaja sovelsi ketteriä menetelmiä. Projektissa rakennettiin asiakkaan liiketoiminnan tukemiseksi aiemmat yksittäiset integraatiot korvaavaa palveluväylää sekä sen kanssa samaan aikaan tietovarastoa sisältämään asiakkaan organisaation *perustietoa* (Masterdata).

Tutkielman tavoitteena oli selvittää minkälaisia haasteita ketterien menetelmien käyttö integraatioprojektissa aiheuttaa ja kirjallisuuden avulla pohtia näitä ongelmia. Tutkimuksessa käsiteltäväksi nostettiin aiheita, jotka selvästi liittyvät ketterien menetelmien soveltamiseen integraatioprojektiin.

Avainsanat ja -sanonnat: järjestelmäintegraatio, ketterät menetelmät, agile, monitoimittajaympäristö, tapaustutkimus, projektinhallinta.

Sisällys

| | |
|--|----|
| 1. Johdanto | 1 |
| 2. Tutkimusmetodi ja -kysymys..... | 3 |
| 3. Keskeiset käsitteet ja teoreettinen tausta..... | 5 |
| 3.1. Ketterät menetelmien käyttö projekteissa | 5 |
| 3.1.1. Erot lineaarisiin menetelmiin | 7 |
| 3.2. Liiketoimintaprosessit | 7 |
| 3.3. Palvelukeskeinen arkkitehtuuri | 8 |
| 3.4. Järjestelmäintegraatio | 8 |
| 3.4.1. Palveluväylä..... | 9 |
| 3.4.2. Järjestelmien integroiminen..... | 11 |
| 3.5. Monitoimijaympäristö..... | 12 |
| 3.6. Perustieto..... | 12 |
| 3.7. Tietämyksenhallinta | 13 |
| 4. Tapaustutkimuksen kuvaus..... | 16 |
| 4.1. Projektin vaiheet..... | 16 |
| 4.2. Kehitystiimi ja sen työskentely | 18 |
| 4.2.1. Henkilöresurssit | 18 |
| 4.2.2. Käytetyt prosessit ja menetelmät | 18 |
| 4.3. Työympäristö..... | 19 |
| 5. Soveltamisen haasteita..... | 22 |
| 5.1. Kommunikaatio..... | 22 |
| 5.1.1. Tiedon kerääminen eri osapuolilta tehokkaasti..... | 23 |
| 5.1.2. Jatkuvan kommunikaation ylläpitäminen..... | 24 |
| 5.1.3. Oikeiden asioiden kommunikointi | 25 |
| 5.1.4. Kehitystiimin sisäinen kommunikaatio | 26 |
| 5.1.5. Asiakkaan näkemys toimintatapoihin | 27 |
| 5.1.6. Kommunikoinnin käytännöt | 28 |
| 5.2. Epätietoisuuden hallinta | 29 |
| 5.2.1. Toimintatavat epätietoisuuden hallitsemiseksi | 30 |
| 5.2.2. Tulevaisuuteen varautuminen | 31 |
| 5.2.3. Muuttuvat vaatimukset..... | 32 |
| 5.2.4. Integraatioiden skaalautuminen | 33 |
| 5.2.5. Muutosten seuranta | 33 |

| | | |
|--------|--|----|
| 5.3. | Projektin tietämyksenhallinta | 34 |
| 5.3.1. | Tiedon jakaminen kehitystiimin sisällä | 35 |
| 5.3.2. | Oleellisen tiedon löytäminen..... | 36 |
| 5.4. | Projektisuunnittelu ja ohjaus..... | 36 |
| 5.4.1. | Työn resursointi..... | 37 |
| 5.4.2. | Projektin edistäminen niukoilla resursseilla | 38 |
| 5.4.3. | Kestävien ratkaisuiden tekeminen | 39 |
| 5.4.4. | Kokonaisuuksien hallinta..... | 41 |
| 6. | Suosituksia..... | 42 |
| 6.1. | Kommunikaatio..... | 42 |
| 6.2. | Epätietoisuuden hallinta | 43 |
| 6.3. | Tietämyksenhallinta | 45 |
| 6.4. | Projektisuunnittelu ja ohjaus..... | 45 |
| 7. | Tutkimukseen liittyvää pohdintaa..... | 47 |
| 8. | Yhteenveto..... | 49 |
| | Viiteluettelo..... | 51 |

1. Johdanto

Järjestelmäintegraatio (Systems Integration) eroaa luonteeltaan perinteisestä ohjelmistokehityksestä. Siinä missä Cockburn [2000] ja Schwaber [2004] keskittyvät enemmän pohtimaan kehitystiimin toimintaa ja suhdetta yhteen asiakkaaseen, integraatioprojekteissa määrittelyn osuus, osapuolten ja integraatioihin liittyvien järjestelmien määrä sekä projektin kulku voivat olla erilaisia verrattuna ohjelmistokehitysprojekteihin [Hobday et al. 2005].

Ketterät menetelmät (Agile methods) on alun perin suunniteltu ohjelmistokehitysprojektien tehostamiseksi, missä projektiin liittyvien osapuolten määrä on verrattain pieni ja projektin kohtaamat haasteet liittyvät usein projektitiimin toimintaan tai vuorovaikutukseen asiakkaan kanssa [Cockburn, 2000; Schwaber 2004]. Tästä syystä ketterien menetelmien soveltaminen suoraan integraatioprojekteihin ei tuota samanlaisia tuloksia kuin ohjelmistokehitysprojekteissa.

Viime aikoina esiin on noussut keskustelunavauksia ketterien menetelmien käytöstä muun muassa Aidanen [2010a, 2010b, 2010c, 2012,] kirjoituksissa, mutta asiaa on tutkittu tieteellisesti melko vähän. Järjestelmäintegraatioon ja ketteriin menetelmiin liittyvät tutkimukset ja artikkelit lähinnä keskittyvät isojen *toiminnanohjausjärjestelmien* (ERP, Enterprise resource planning) käyttöönoton nopeuttamiseen [Alleman, 2002].

Siksi tässä tutkimuksessa keskityttiinkin havainnoimaan järjestelmäintegraatioprojektia, jossa käytetään ketteriä menetelmiä projektin läpiviemiseen jokapäiväisessä toiminnassa. Tutkimuksen aikana seuratussa

järjestelmäintegraatioprojektissa rakennettiin palveluväylää teollisuuden palveluyritykselle osana yrityksen strategian täytäntöönpanoa. Palveluväylän tavoitteena oli korvata yrityksen yksittäiset integraatiot sekä tukea liiketoimintaa ja sen kasvua. Järjestelmäintegraatioprojektin rinnalla asiakasyritys uudisti myös liiketoimintaprosessejaan (Business Process) ja järjestelmiään sekä otti käyttöön uusia järjestelmiä, mikä heijastui muutoksina integraatioympäristöön sen kehityksenkin aikana.

Palveluväylän yhteydessä rakennettiin myös integraatiotoimittajan toimesta tietovarasto asiakkaan organisaation perustiedolle. Tämän tarkoituksena oli tukea palveluväylän toimintaa tarjoamalla yksi tiedonlähde asiakkaan organisaation tietojärjestelmille.

Muuttuva IT-ympäristö, asiakkaan resurssit sekä integraatiotiimille uusi liiketoimintaympäristö vaativat tavallisesta poikkeavaa lähestymistapaa järjestelmien integroimisessa. Tästä sekä aikatauluista johtuvista syistä projektia päädyttiin lähteä viemään eteenpäin ketterästi.

Tutkielma kuvaa ketterien menetelmien käyttöä järjestelmäintegraatioprojektissa ja nostaa esiin niiden käytöstä seuraavia haasteita. Lisäksi kirjallisuuskatsauksen avulla käsitellään esiin tulleita ongelmia.

2. Tutkimusmetodi ja –kysymys

Tutkimusmetodinä käytin *reflektiivaa käytäntöä* (Reflective practice) [Schön, 1983], jossa itse osallistuin järjestelmäintegraatioprojektin toteutukseen ja samalla havainnoin projektia sen toimintaympäristön muodostamassa kontekstissa reflektoiden havaintojani käytäntöön. Lähtökohtana tutkimukselle oli halu saada tietoa uudenlaisen kehitysmenetelmän käyttämisessä integraatioprojektissa ja havaintojen perusteella parantaa integraatiopalveluja tarjoavan yrityksen käytäntöjä. Iivarin [2007] mukaan ongelmalähtöisen tutkimuksen sijaan uuden ideat ja artifaktit voivat parantaa käytäntöjä jo paljon ennen niiden ratkaisemien ongelmien esiintymistä.

Schönin [2004] mukaan reflektiivassa käytännössä formaalin opetuksen sijaan tietoa hankitaan oppimalla käytännössä sekä peilaamalla kokemuksia tuleviin tilanteisiin. Tällöin eteen tulevat ongelmat pystytään ratkaisemaan ensimmäisellä kerralla ja kehittämään samalla ihmisten tai organisaatioiden toimintaa aikaisempaa paremmaksi.

Menetelmänä reflektiivaa käytäntöä sopi hyvin dynaamiseen projektiympäristöön, sillä se tukee uusien asioiden omaksumista, ymmärrystä sekä niistä oppimista yksilön tasolla ilman rajoittavampia tutkimusasetelmien määrittelyä [Schön, 2004].

Käytännössä keräsin havaintoja tutkielmaa varten yli kuuden kuukauden tarkastelujakson ajan projektin etenemisestä ja menetelmistä itsenäisesti sekä reflektiivalla kokemuksiani käytännöiksi ja havainnoimalla seurauksia. Lisäksi kävin keskusteluita kehitystiimin jäsen kanssa aiheeseen liittyen.

Tässä tutkielmassa tarkoituksena on käsitellä niitä käytäntöjä ja menetelmiä, joita tutkittavassa projektissa käytettiin. Havaintojen perusteella laaditaan lista suosituksista, joita tämän projektin perusteella kannattaa ottaa huomioon ketteriä menetelmiä integraatioprojektiin sovellettaessa. Tämä tutkimustapa noudattelee yleistä tietojärjestelmien tutkimuksen tavoitetta luoda teoriaa, jota voidaan käyttää todellisten ongelmien ratkaisemiseen [Piirainen and Gonzalez, 2013].

Kantavana tutkimuskysymyksenä tässä tutkielmassa on:

Mitä haasteita ketterien menetelmien soveltamisesta integraatioprojektissa aiheutuu ja minkälaisia hyviä käytäntöjä ketterän järjestelmäintegraation toteuttamiseksi projektin perusteella voidaan esittää?

3. Keskeiset käsitteet ja teoreettinen tausta

Tutkielman keskeisiin käsitteisiin kuuluu järjestelmäintegraatio, monitoimijaympäristö, palvelukeskeinen arkkitehtuuri sekä ketterät menetelmät. Tässä luvussa kuvataan näitä keskeisimpiä käsitteitä. Järjestelmäintegraatiota kuvataan muita tarkemmin sillä se on tutkielman kannalta oleellinen käsite.

3.1. Ketterät menetelmien käyttö projekteissa

Ketterien menetelmien johtava ajatus on tehdä monimutkaisista ja laajoista ohjelmistokehitysprojekteista helpommin hallittavia jakamalla ne pienempiin osiin sekä vähentämällä projektinhallintaan liittyvää byrokratiaa antamalla kehitystiimille enemmän vapauksia ja vastuuta [Cockburn, 2000; Schwaber 2004, Little 2005].

Laaja projekti jaetaan ketteriä menetelmiä käytettäessä pienempiin osiin, iteraatioihin, joiden tavoitteet ovat helpommin hallittavissa. Schwaberin [2004] mukaan on tärkeää saada asiakkaalta palautetta jatkuvasti tehdystä työstä, jotta tarvittaessa kehitystiimi pystyy muuttamaan projektin suuntaa.

Antamalla kehitystiimille enemmän vapauksia ratkoa eteen tulevia haasteita vähennetään projektin ohjaamiseen tarvittavia resursseja. Kehitystiimi toimiikin ketterässä projektissa yleensä autonomisesti ja on tarvittaessa suoraan yhteydessä asiakkaan edustajiin. Tämä kuitenkin vaatii tiimiltä omistautumista ja vastuun ottamista projektista [Augustine et al. 2005].

Kun projekti jaetaan pienempiin osiin, joista tarvitaan palautetta, täytyy myös asiakkaan sitoutua projektiin aiempaa tiukemmin. Allemanin [2002] mukaan ketterän projektin onnistuminen edellyttää asiakkaan osallistamista niin vaatimusten priorisointiin kuin palautteen antamiseen. Projektiin osallistuva asiakas on myös paremmin ajan tasalla projektin tilasta

Integraatioprojekteissa kokonaisuuden kannalta vaadittava tietämys on jakautunut niin usealle eri osapuolelle, että asiakkaan osallistamisen lisäksi myös muut toimijat täytyy saada osallistumaan projektiin sen onnistumiseksi.

Yksi tärkeimmistä ketterien menetelmien eduista muihin menetelmiin verrattuna on niiden tapa suhtautua vaatimusmäärittelyyn ja muuttuviin vaatimuksiin. Ennen ketteriä menetelmiä esimerkiksi vesiputousmallin mukaisesti toteutetuissa projekteissa vaatimusten määrittelyyn käytettiin projektin alussa huomattavasti aikaa ja tarkoituksena oli listata projektin toteuttamisen kannalta tarpeelliset vaatimukset. Muutokset vaatimuksissa kesken projektin toteutuksen saattoivat aiheuttaa suuria haasteita projektille, sillä toteutus pohjautui vahvasti ennalta määritellyille vaatimuksille.

Ketterien menetelmien yksi kulmakivistä on muuttuvien vaatimusten hyväksyminen ja huomioon ottaminen [Cockburn, 2000; Schwaber 2004]. Schwaberin [2004] mukaan tämä johtaa myös siihen, että tiukkojen aikataulujen sijaan voidaan korkeintaan keskustella arvioiduista työmääristä tietyn vaatimuksen toteuttamiseksi.

3.1.1. Erot lineaarisiin menetelmiin

Projektinhallinnallisesta näkökulmasta ketterät menetelmät eroavat huomattavasti lineaarisemmista menetelmistä. Suurempien ja monimutkaisempien projektien arvioiminen ajankäytön ja vaatimusten suhteen ennen projektin toteuttamista voi olla haasteellista ja tämä voi näkyä aikataulujen venymisenä sekä asiakkaan kannalta vääränlaisina ratkaisuina, kun toteutettu ohjelmisto ei vastaakaan ennalta tiukkaan määriteltyjä vaatimuksia. Ketterissä menetelmissä kehitystiimin ymmärrys projektista ja vaatimuksista tarkentuu projektin edetessä, jolloin tiimi pystyy arvioimaan työmääriä sekä määrittelemään vaatimuksia jo toteutettujen ominaisuuksien ja kerätyn toimialatuntemuksen pohjalta [Schwaber, 2004; Augustine et al. 2005].

3.2. Liiketoimintaprosessit

Organisaatioissa erilaiset toiminnallisuudet on järjestetty kokonaisuuksiksi, jotka tuottavat organisaation toiminnan kannalta oleellisia palveluita tai tuotteita. Näitä toiminnallisuuksien järjestettyjä kokonaisuuksia nimitetään liiketoimintaprosesseiksi [Salmela et al. 2010].

Järjestelmäintegraation kannalta liiketoimintaprosessit kuvaavat miten eri tietojärjestelmät toimivat yhdessä. Tämä ei välttämättä riitä määrikykseksi integraatioiden toteuttamista varten, mutta antaa kokonaiskuvan organisaation tietojärjestelmien yhteyksistä toisiinsa.

Liiketoimintaprosessit antavat myös liiketoimintalähtöisen perustan integraatioarkkitehtuurin suunnittelemiselle sekä toteuttamiselle. Tällöin

voidaan välttyä teknologiaalähtöiseltä ratkaisumallilta, joka ei välttämättä ota huomioon liiketoiminnan tarpeita.

3.3. Palvelukeskeinen arkkitehtuuri

Palvelukeskeinen arkkitehtuuri (SOA, Service Oriented Architecture) määrittää järjestelmien toiminnot *rajapintojen* (Interface) kautta käytettäväksi erillisiksi palveluiksi, joista voidaan kokoamalla luoda uusia palveluita tukemaan liiketoimintaprosesseja [Josuttis, 2007].

Ajatuksena on luoda erillisiä toisistaan riippumattomia komponentteja, jotka ovat uudelleenkäytettäviä IT-ympäristössään ja joille on määritelty tarkasti rajapinta kommunikointiin muiden järjestelmien kanssa.

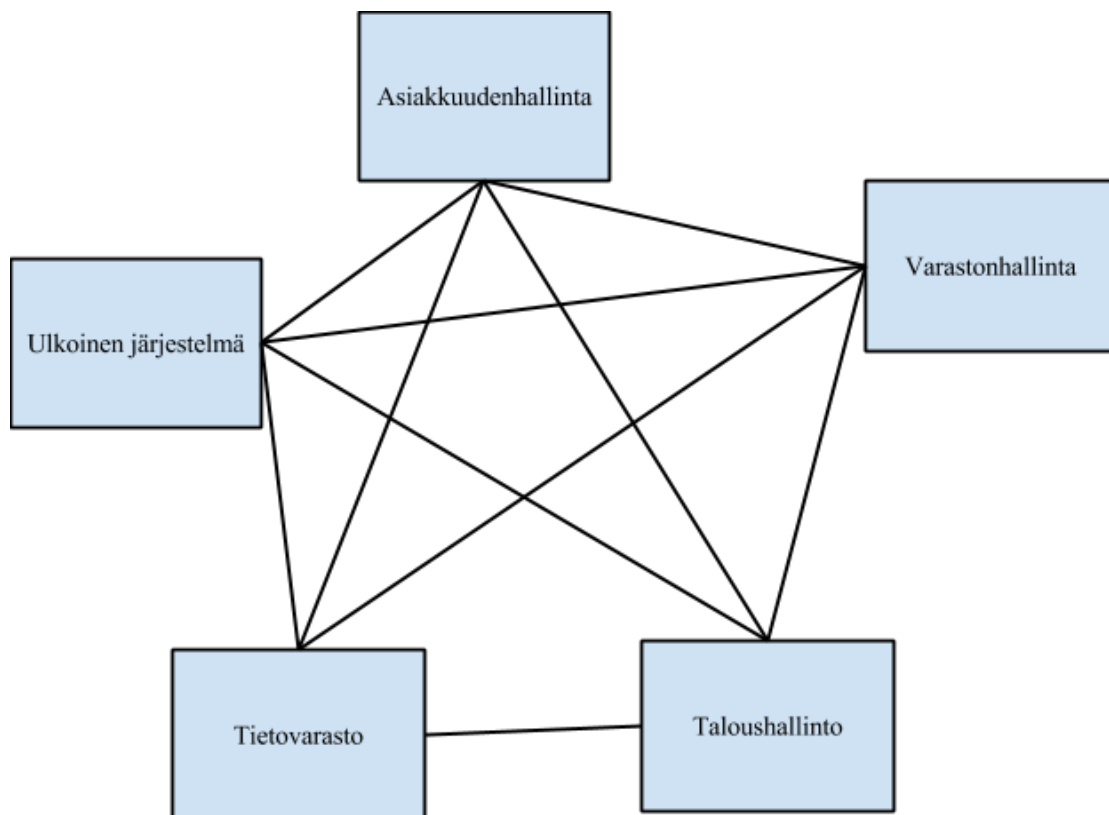
Integraatiotyön kannalta palvelukeskeisen arkkitehtuurin mukaisesti toteutetut järjestelmät tukevat tiedon siirtämistä niihin ja niistä ulos, jolloin integroitumisen kustannukset laskevat.

3.4. Järjestelmäintegraatio

Järjestelmäintegraatio käsitteenä tarkoittaa erillisten järjestelmien yhdistämistä siten, että ne toimivat yhdessä yhtenä kokonaisuena järjestelmänä [Hobday et al. 2005]. Tämä saavutetaan yhdistämällä järjestelmät toisiinsa siten, että ne jakavat toisten järjestelmien tarvitsemaa informaatiota ja vastaavasti voivat hakea tarvitsemaan informaatiota muilta järjestelmiltä. Tutkielman kontekstissa järjestelmäintegraatio tarkoittaa yksittäisten järjestelmien yhdistämistä toisiinsa mallinnettujen liiketoimintaprosessien kuvaamalla tavalla palveluväylää hyväksi käyttäen.

3.4.1. Palveluväylä

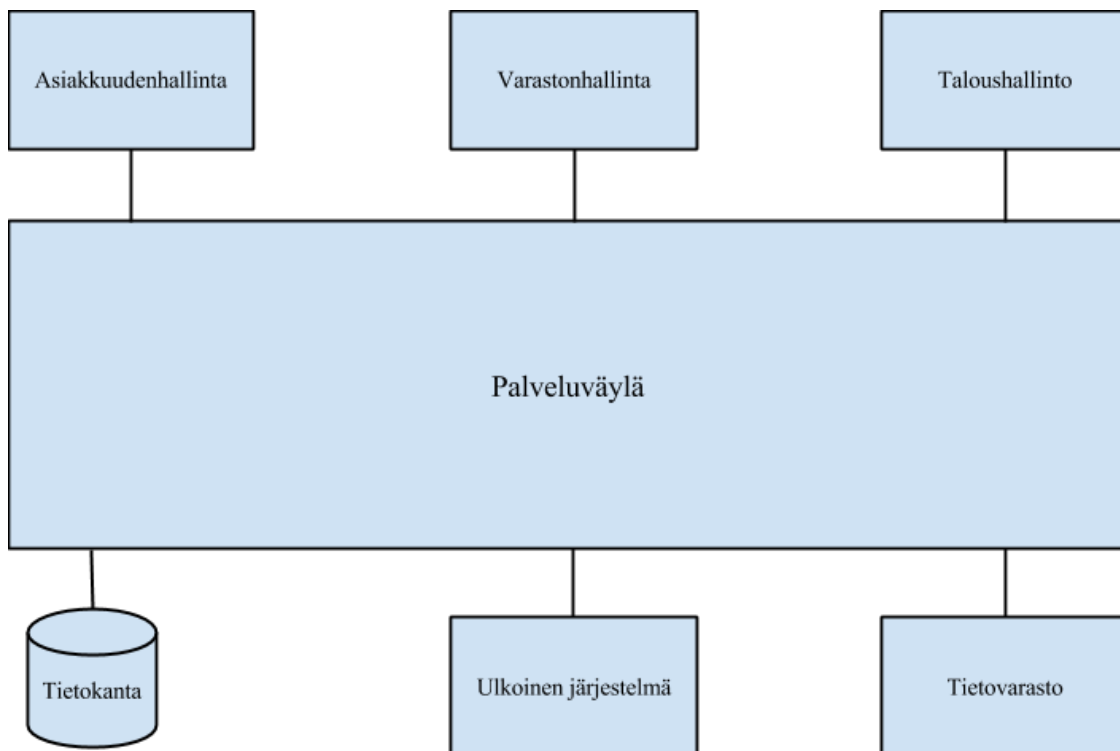
Tarkastelun kohteena olleessa projektissa integraatiot toteutettiin *palveluväylän* (ESB, Enterprise Service Bus) avulla. Palveluväylän on tarkoitus toimia eri järjestelmien välissä muuntamassa, reitittämässä sekä ohjaamassa sanomia liiketoimintaprosessien määrittelemillä tavoilla [Hohpe, 2012].



Kuva 1. Esimerkki suoraan toisiinsa integroiduista järjestelmistä

Yksinkertainen tapa integroida järjestelmiä toisten kanssa on suoraan rakentaa järjestelmien välille tarvittavien tietojen siirto, kuten kuvassa 1. Hohpen [2012] mukaan, kun järjestelmiä on paljon ja ne kaikki on integroitu toisiinsa,

tarvittavien järjestelmäkohtaisten integraatioiden määrä voidaan laskea kaavalla $n*(n-1)$. Jos yhden järjestelmän toiminta muuttuu tai se korvataan toisella, joudutaan kaikki aikaisempaan järjestelmään tehdyt integraatiot korvaamaan uusilla.



Kuva 2. Yksinkertainen kuva palveluväylästä ja sen liittymistä eri järjestelmiin.

Palveluväylän (kuva 2) ideana on integroida järjestelmät vain palveluväylään, jolloin se piilottaa eri järjestelmille näkyvät palvelut omien rajapintojensa taakse. Palveluväylällä eri järjestelmien sanomat muutetaan väylän omaan formaattiin, *kanoniseen malliin* (Canonical model), joka määrittellään mahdollistamaan kaikki eri järjestelmien vaatimat tietosisällöt kyseiselle mallille. Tällöin yhden järjestelmän muutos vaikuttaa vain yhteen integraatioon muuttuneen järjestelmän ja palveluväylän välillä.

3.4.2. Järjestelmien integroiminen

Järjestelmien integroimiseen on monia tapoja. Joitakin järjestelmiä ei välttämättä ole suunniteltu integroitaviksi tai jakamaan informaatiota, joten niiden liittäminen yhteen toimivaksi kokonaisuudeksi voi olla haastavaa. [Hohpe, 2012] Yleisimpiä tapoja integroida järjestelmiä ovat siirtotiedostot, tietokantayhteydet, *etäkutsut* (RPC, Remote Procedure Call) sekä sanomat eri järjestelmien välillä.

Tutkittavassa projektissa näistä käytettiin lähinnä sanomia, tietokantayhteyksiä ja siirtotiedostoja. Sanomien ideana on muun muassa serialisoida jonkin tietojärjestelmän käsitteen tila ja lähettää se sanomana toiselle järjestelmälle.

Rajapinnat ovat eri tietojärjestelmien sopimuksia tavasta kommunikoida. Ne määrittävät miten järjestelmiin voidaan integroitua ja mitä informaatiota järjestelmien välillä voi kulkea. Rajapinnat ovat luonteeltaan myös dynaamisia ajan suhteen. Ne voivat muuttua liiketoiminnan vaatimusten muuttuessa, mikä vaatii rajapintojen elinkaareen ja versioiden hallintaa. [Josuttis, 2007]

Järjestelmäintegraatoratkaisu kokonaisuudessaan voi koostua vallitsevan tilanteen tai arkkitehtuurin kartoittamisesta sekä sen pohjalta sopivan ratkaisun ehdottamisesta ja toteuttamisesta aina ylläpitoon asti. Ylläpidon näkökulmasta integraatioiden suoritusta ja tilaa täytyy pystyä valvomaan ja virhetilanteisiin puuttumaan, mikä asettaa erityisiä vaatimuksia integraatioiden toteuttamiselle.

3.5. Monitoimijaympäristö

Integraatioprojekteilte ominaista on monen toimijan ympäristö, jossa asiakkaan ja integraatiotoimittajan lisäksi voi olla useita järjestelmätoimittajia sekä muita tahoja, kuten viranomaisia. Työn koordinointi usean toimijan kesken voi vaatia huolellista projektisuunnittelua ja heijastuu yleensä integraatiotoimittajan työhön. Mitä useampi taho integraatioprojektissa on mukana, sitä suurempi on todennäköisyys erilaisten toimintatapojen ja prosessien aiheuttamille ongelmille projektin kokonaisuuden kannalta.

Organisaatioiden järjestelmähankinnat ovat vaihtumassa keskitetyistä ratkaisuksista parhaiten käyttötarkoituksiin sopiviin ohjelmistotuotteisiin, jotka parhaiten tukevat organisaatioiden toimintaa [Salmela et al. 2010]. Tästä syystä järjestelmätoimittajien kirjo voi kasvaa suuremman organisaation tapauksessa isoksi ja maantieteellisesti hajautuneeksi.

Monen eri järjestelmän saattaminen toimimaan keskenään vaatii laajaa tietämystä niiden toiminnasta sekä sanomanvälitykseen tarkoitetuista rajapinnoista tai muista integroitumistavoista. Useiden eri toimijoiden tapauksessa tietämys järjestelmästä on hajautunut laajalle asiakasorganisaatioon sekä järjestelmätoimittajille [Hobday et al. 2005].

3.6. Perustieto

Perustieto (Master data) on organisaation useiden järjestelmien käyttämää, hallittua informaatiota, jonka määrän voi katsoa kasvavan ennemminkin liiketoiminnan kuin ajan funktiona. Wolterin ja Haselden [2006] määrittelevät

perustiedon organisaation kannalta tärkeiksi entiteeteiksi, joiden hallinta on tärkeää organisaation liiketoiminnan kannalta.

Perustietoa voi kategorisoida ja sen sisältämät tiedot määräytyvät organisaation tarpeiden mukaan. Esimerkiksi suurella kansainvälisellä sekä pienellä paikallisella pankilla on todennäköisesti eriävät tarpeet asiakkaisiinsa liittyvissä tiedoissa.

Tutkittavassa projektissa perustiedon hallintaan toteutettiin tietovarasto, jonka tietojen päivityksestä lähdejärjestelmistä ja tietojen jakamisesta sitä kuluttaviin järjestelmiin palveluväylä vastasi. Tällä tavalla kaikilla järjestelmillä oli *yksi ja sama ajantasalla oleva tietolähde* (SOR, System of Record).

Integraatiotyön kannalta perustieto voi mahdollistaa kustannustehokkaan tavan dataintegraatioiden toteuttamiseksi. Yksi tietolähde mahdollistaa integraatiototeutusten uudelleenkäyttämisen, mikä näkyy etenkin uusien tietolähdettä käyttävien integraatioiden toteutusajoissa.

Perustieto voi myös määrittää yhteiset käsitteet koko organisaation järjestelmien laajudella, jolloin väärinymmärrysten määrä eri kontekstien välillä vähenee.

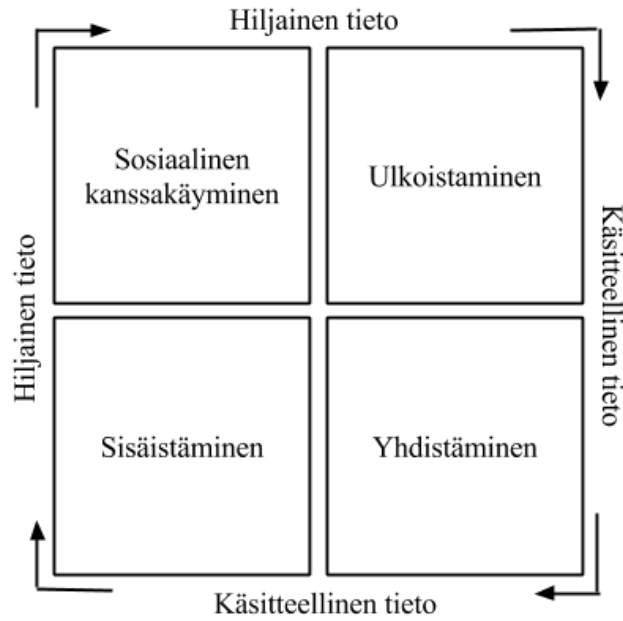
3.7. Tietämyksenhallinta

Tutkittavan projektin analysoinnin yhteydessä *Tietämyksenhallintaan* (Knowledge management) liittyvää tutkimusta käytetään nostamaan esiin tiettyjä ketterien menetelmien soveltamiseen liittyviä haasteita.

Tietämyksenhallinta organisaation kontekstissa on prosessi, jossa yksittäisten ihmisten tietämystä ja osaamista yritetään koko organisaation laajuudessa hyödyntää ja kasvattaa [Nonaka ja Krogh, 2009]. Keskeisiä projektiin liittyviä käsitteitä Nonakan ja Kroghin [2009] sekä myös Alavin ja Leidnerin [1999] mukaan ovat *hiljainen tieto* (Tacit knowledge) ja *käsitteellinen tieto* (Explicit knowledge). Hiljainen tieto on henkilön osaamisen, toimintatapojen sekä tietämyksen yhdistelmä, jota on haastavaa siirtää toiselle. Käsitteellinen tieto on jollekin medialle kuvattua tietämystä, jonka siirtäminen on hiljaista tietoa helpompaa.

Esimerkki hiljaisesta tiedosta on työntekijän tapa tehdä asioita vuosien kokemuksella tehokkaasti. Työntekijän toimintatapojen siirtäminen uudelle työntekijälle vaatii yleensä konkreettista työskentelyä vanhan työntekijän kanssa. Käsitteellinen tieto on helpommin kuvattavaa ja siitä esimerkkinä voisi olla vaikka tietyn laitteen ohjekirja, jossa kerrotaan miten laitetta käytetään.

Nonaka ja Krogh [2009] ovat käsitelleet hiljaisen ja käsitteellisen tiedon suhdetta sekä tiedon konvertoimista tällä akselilla näiden kahden ääripään välillä, Nonaka ja Toyama [2003] ovat kuvanneet tarkemmin kyseistä prosessia sekä sen luonnetta.



Kuva 3. Nonakan ja Toyaman [2003] kuvaama SECI-malli

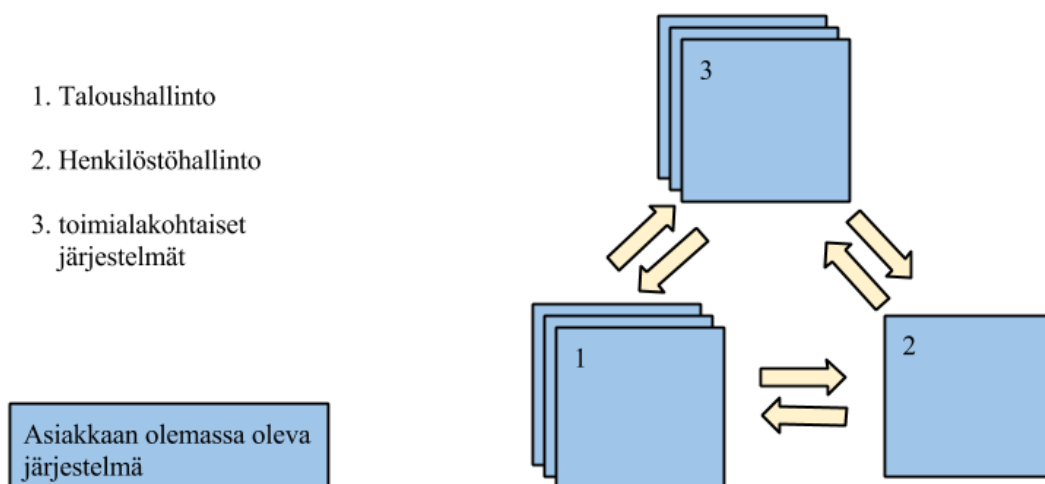
Nonaka ja Toyama [2003] kuvaavat hiljaisen ja käsitteellisen tiedon suhdetta *SECI-mallin* (Socialization, Externalization, Combination, Internalization) avulla. Kuvassa 3 olevassa mallissa kuvataan miten tietämys kasvaa organisaation sisällä ja miten se vaihtelee hiljaisen ja käsitteellisen tiedon välillä.

4. Tapaustutkimuksen kuvaus

Tarkastelun kohteena olevassa järjestelmäintegraatioprojektissa lähdettiin liikkeelle asiakasyrityksen tarpeesta luoda yhtenäinen integraatioarkkitehtuuri, joka tukee liiketoimintaprosesseja, yhdistää olemassa olevat ja uudet kehitteillä olevat järjestelmät sekä tukee liiketoiminnan kasvua. Osana rakennettavaa palveluväylää toteutettiin myös perustietojen hallintaratkaisu, jota varten määriteltiin tietojen lähde- ja kohdejärjestelmät sekä toteutettiin dataintegraatiot järjestelmien välille hyödyntämällä keskitettyä tietovarastoa.

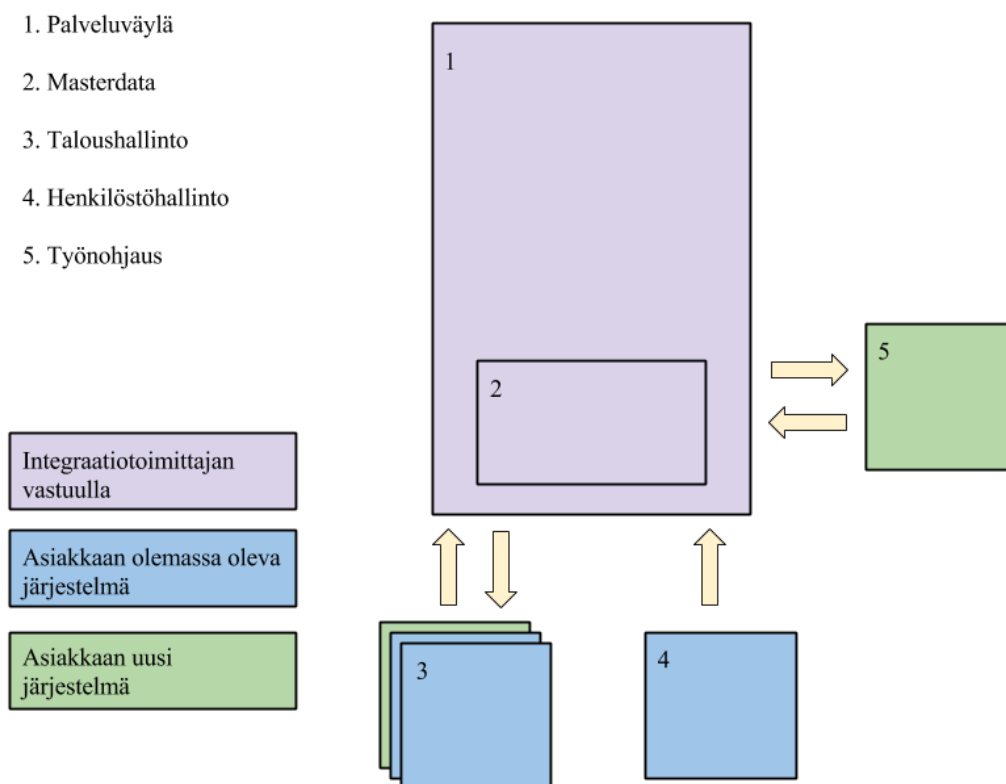
4.1. Projektin vaiheet

Projektia suunniteltaessa tavoitteena oli vaiheittain rakentaa tuki asiakkaan eri liiketoimintaprosesseille aloittaen perustiedon mallintamisesta, keräämisestä ja jakamisesta eri järjestelmille sekä asiakkaan uuden työnohjausjärjestelmän käyttöönoton mahdollistamisesta.



Kuva 4. Asiakkaan järjestelmien integraatiot ennen palveluväylää

Ensimmäiseen vaiheeseen liittyi asiakkaan viiden käytössäolevan järjestelmän integrointi toisiinsa palveluväylän avulla. Järjestelmät liittyivät lähinnä henkilöstöhallintaan, talouden järjestelmiin sekä työnohjaukseen. Ennen palveluväylän toteuttamista asiakkaan järjestelmien välille oli toteutettu suoria integraatioita järjestelmistä toisiin (Kuva 4).



Kuva 5. Integraatioarkkitehtuuri projektin ensimmäisen vaiheen jälkeen

Ensimmäisen kehitysvaiheen aikana käyttöön otettiin asiakkaan uusi työnohjausjärjestelmä, jonne oli tarkoitus siirtää työnohjaamiseen liittyvät toiminnot toimialakohtaisista järjestelmistä.

Seuraavan vaiheen aikana asiakkaan toimialakohtaisia järjestelmiä integroitiin työnohjausjärjestelmään ja laskutusprosessia tukevia integraatioita toteutettiin järjestelmien välille. Projektin aikana suurin osa yksittäisistä järjestelmien välisistä integraatioista korvattiin palveluväylän avulla.

4.2. Kehitystiimi ja sen työskentely

4.2.1. Henkilöresurssit

Kehitystiimiin kuului projektipäällikkö, kolme kehittäjää sekä ajoittain kaksi asiantuntijaa, jotka tukivat asiantuntemuksellaan palveluväylän ja perustietovaraston toteuttamista. Tiimin jäsenten työpisteet sijaitsivat lähekkäin, mikä helpotti kommunikointia. Cockburnin [2000] sekä Schwaberin [2004] mukaan fyysisesti lähekkäin sijoittuminen kehitystiimin jäsenten kesken tehostaa kehitystä. Tällä tavalla muun muassa tiedonkulku tehostuu.

Lisäksi työtä oli tukemassa tehtävähallintajärjestelmä, yhteistyöalusta sisältöjen jakamiseen tiimin sisällä ja asiakkaan kanssa sekä pikaviestin kommunikoinnin avuksi.

Tiimin sisällä osaamisalueet, yksittäisten jäsenten vahvuudet sekä integraatiokokemuksen määrä vaihteli, mutta lähekkäin sijoittuminen fyysisesti sekä aktiivinen tietämyksen jakaminen tehokkaasti tasoittivat näitä eroja.

4.2.2. Käytetyt prosessit ja menetelmät

Palveluväylän rakentamisessa ei noudatettu yhtä tiettyä ketterää menetelmää, vaan käyttöön otettiin projektia tukevia ketteriä käytäntöjä.

Schwaber [2004] kuvaa Scrum-menetelmää sekä sen käyttöä ketterässä kehityksessä monin esimerkein. Scrumin ideana on antaa paljon vastuuta kehitystiimille ja varautua muuttuviin vaatimuksiin lyhyillä iteraatioilla ja asiakkaan osallistamisella antamaan palautetta. Integraatioprojektit kuitenkin eroavat luonteeltaan ohjelmistokehitysprojekteista siten, että Schwaberin [2004] määrittelemän *projektin omistajan* (Product owner) ääni todennäköisesti pirstaloituu, koska integraatioihin liittyy useita organisaation liiketoimintalueita eikä yksi henkilö pysty vastaamaan suuressa organisaatiossa kaikista päätöksistä.

Kehitystiimi työskenteli itseohjautuvasti projektipäällikön tukemana, mikä mahdollistaa tehokkaan työskentelyn ja vähentää hallinnollista kuormaa [Cockburn 2000; Schwaber, 2004; Augustine et al. 2005]. Asiakkaan puolelta projektia tukemassa oli asiantuntijoita, joiden kanssa kehitystiimin jäsenet kommunikoivat jatkuvasti projektin aikana. Tehtävät työt myös priorisoitiin asiakkaan edustajien kanssa.

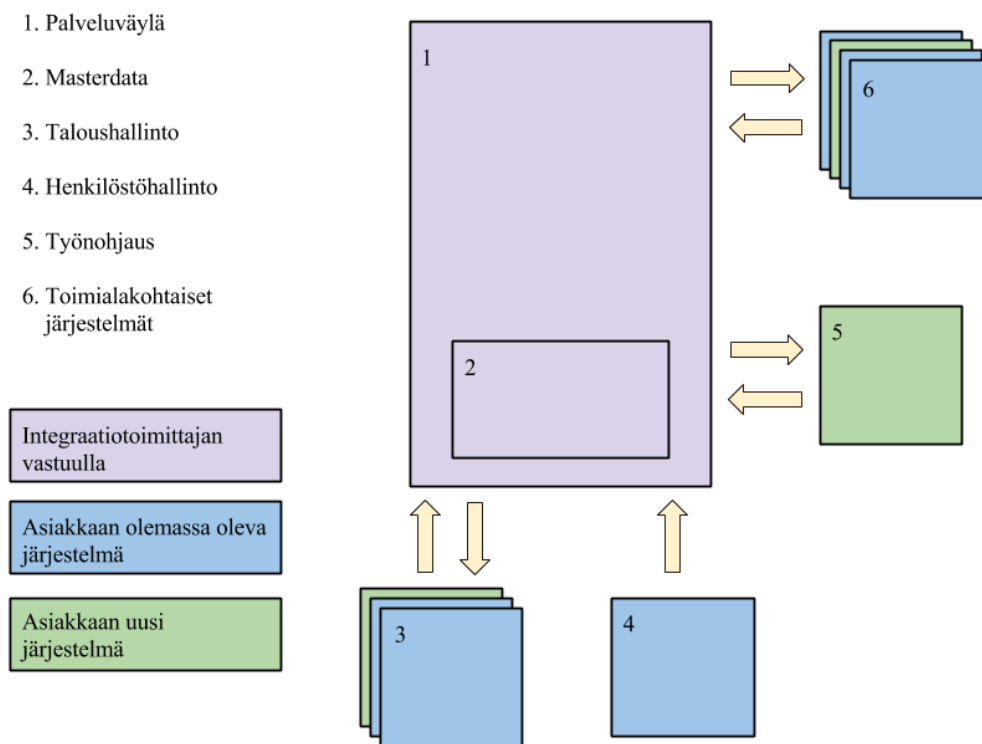
Ajoittain järjestettiin tilannepalavereja, joissa käytiin läpi projektin sen hetkinen tila ja haasteet, jotka pitäisi selvittää. Suurempien kokonaisuuksien osalta järjestettiin ennen toteutustyön aloittamista työpajoja, joissa paikalla olevien henkilöiden kanssa selkeytettiin tavoitteita ja määriteltiin vaatimuksia sekä haettiin avoimiin kysymyksiin vastauksia.

4.3. Työympäristö

On tärkeää kuvata työympäristöä, jossa kehitysprojekti toteutettiin. Se antaa tuloksille oman viitekehyksensä ja helpottaa niiden tulkintaa. Etenkin ketterään ohjelmistokehitykseen vertailtaessa toimintaympäristöjen eroavaisuudet

integraatioprojekteihin nähden selittävät osin, miksi ketterät menetelmät eivät ole suoraan sovellettavissa niihin.

Tutkittavan projektin työympäristö (kuva 6), johon integraatioita toteutettiin, koostui useista asiakasorganisaation tietojärjestelmistä. Osa järjestelmistä oli asiakkaan itse hallitsemissa ja osa kuului asiakkaan omille asiakkaille. Lisäksi osa järjestelmistä otettiin käyttöön samaan aikaan integraatiototeutuksen kanssa. Yhteensä integroitavana oli projektin kontekstissa kymmenen eri järjestelmää, joista osalla saattoi olla useita käytössä olevia instansseja eri toimipaikoissa.



Kuva 6. Havaintokuva asiakkaan it-ympäristöstä

Integraatioarkkitehtuuri rakentui asiakkaan uuden työnohjausjärjestelmän tarpeisiin sopivaksi. Uudet liiketoimintaprosessit pohjautuivat

työnohjausjärjestelmän, toimialakohtaisten järjestelmien sekä henkilöstöhallinnan yhdistämiseen toimivaksi kokonaisuudeksi. Lisäksi palveluväylän avulla laskutusprosesseja, kuten tehdyn työn laskutus, automatisoitiin järjestelmien välillä ja integraatiot mahdollistivat tarvittavan informaation saamisen raportoinnin tarpeisiin.

Ketteristä menetelmistä puhuttaessa yleensä oletetaan työympäristön koostuvan hyvin pienestä määrästä osapuolia, lähinnä asiakkaasta sekä kehitystiimistä [Cockburn, 2000; Schwaber, 2004; Little 2005; Augustine et al. 2005]. Integraatioprojektit taas koostuvat yleensä useista osapuolista, joihin lukeutuvat asiakkaan lisäksi useat järjestelmätoimittajat. Osapuolien lisääntyessä, myös projektin kompleksisuus lisääntyy ja eri toimijoiden välisten riippuvuuksien määrä lisääntyy. Tutkittavan projektin kontekstissa (katso liite 1) uuden palveluväylän version asentaminen tuotantokäyttöön vaati pelkästään asiakkaan omassa organisaatiossa useiden sidosryhmien yhteistyötä sekä tapauskohtaisesti myös useamman järjestelmätoimittajan osallistumista.

Palveluväylää kehittävä tiimi toimi ensisijaisesti tiiviissä yhteistyössä asiakkaan projektitiimin kanssa sekä tarvittaessa myös asiakkaan järjestelmätoimittajien asiantuntijoiden kanssa. Tutkittava projekti oli osa suurempaa muutosta asiakkaan tietojärjestelmäkentässä, joten kehitettävä palveluväylä ei ollut ainoa uusi, samaan aikaan käyttöön otettava järjestelmä. Tämä vaikutti toimintaympäristön osalta etenkin asiakkaan projektitiimin resursseihin sekä eri osapuolilta vaadittavaan yhteistyön määrään erilaisten muutosten yhteydessä.

5. Soveltamisen haasteita

Projektin aikana esiin nousi haasteita, jotka voidaan jakaa niiden luonteen mukaan eri kategorioihin. Havaitut haasteet liittyvät integraatioprojektien erilaiseen luonteeseen verrattuna ohjelmistokehitykseen. Muun muassa osapuolien määrä sekä projektin toteutukseen liittyvät eroavaisuudet aiheuttivat haasteita ketterien menetelmien soveltamiseen tutkittavaan projektiin.

Havainnot on jaettu neljään kategoriaan seuraavasti: osiossa 5.1 käydään läpi kehitystiimin ja asiakkaan sekä asiakkaan muiden järjestelmätoimittajien väliseen vuorovaikutukseen liittyviä haasteita. Sitä seuraavassa osiossa keskitytään käsittelemään tavallisesti etukäteen määriteltyjen toiminnallisuuden toteuttamista ketterästi. Osiossa 5.3 tarkastellaan ketterän näkökulman vaikutusta integraatioiden toteuttamiseen vaadittavan informaation ja tietämyksen saatavuuteen oikealla hetkellä ja osiossa 5.4 tarkastellaan järjestelmäintegraatioprojektin hallintaan liittyviä haasteita.

5.1. Kommunikaatio

Ketterien menetelmien käyttäminen jo perinteisessä asiakas-toimittaja -suhteessa vaatii toimivaa ja tehokasta vuorovaikutusta osapuolilta, jotta ennalta arvaamattomat ongelmat pystytään selvittämään nopeasti ja tehokkaasti. Asiakkaan antama palaute on myös tärkeää projektin kehittymiseksi oikeaan suuntaan [Alleman, 2002]. Kommunikaation tarkoituksena on olla väline tiedon hankintaan ja yhteisen kontekstin löytämiseen, silloin kuin päätöksen tekemistä

varten ei ole tarpeeksi informaatiota saatavilla tai osapuolten näkemykset eroavat toisistaan. Oletusten tekeminen kysymisen sijaan voi projektissa johtaa virheisiin, joiden huomiotta jääminen ja korjaaminen myöhemmin on kallista [Cockburn, 2000].

5.1.1. Tiedon kerääminen eri osapuolilta tehokkaasti

Tutkimuksen kohteena olevassa integraatioprojektissa kommunikointi asiakkaan kanssa oli kokonaisuudessaan monipuolista ja rikasta. Viestimiseen käytettiin moninaisia keinoja pikaviestimistä monen eri toimijan välisiin palavereihin. Tästä huolimatta projektin aikana esiin nousi toistuvasti seuraava ongelma:

Uusien integraatioiden tai palveluväylän toimintojen kehityksessä - vaatimusten havaitsemisen jälkeen - toteuttamiseen vaadittavan informaation kerääminen eri osapuolilta kesti ajoittain huomattavan pitkään samoin kuin ratkaisujen vieminen valmiina tuotantoympäristöön asti.

Tämä seikka vaikutti projektissa kokonaisuuden kannalta eniten tiimin ja muiden toimijoiden tehokkuuteen toistuvuutensa johdosta. Osittain ongelma johtui siitä, että asiakkaan projektitiimin jäsenten lisäksi tiedonkeruu kosketti myös muita kolmannen osapuolen toimijoita, joihin ei ollut yhtä suoraa yhteyttä kuin asiakkaaseen. Toisaalta, vaikka palveluväylää kehitettiin ketterästi asiakkaan kanssa, eivät kaikki muut toimijat toimineet samalla tavalla.

Projektin edetessä oli kuitenkin huomattavissa mainitun ongelman vaikutusten väheneminen, sillä kehitystiimin ymmärrys asiakkaan toimintaympäristöstä ja

vaatimuksista kasvoi jatkuvasti. Tämä tukee Schwaberin [2004] olettamusta ketterän tiimin toiminnan tehostumisesta projektin edetessä. Nonakan ja Toyaman [2003] mukaan tämä liittyy olemassa olevan tiedon yhdistämiseen uuteen, organisaation ulkopuolelta kerättyyn tietoon. Kun projektityötä tukevaa tietoa löytyy kehitystiimin sisältä tarpeeksi, uusien asioiden ymmärtäminen helpottuu.

5.1.2. Jatkuvan kommunikaation ylläpitäminen

Aiemmassa kappaleessa mainittuun tiedon keräämisen haasteeseen liittyen jatkuvan kommunikaation ylläpitäminen asiakkaan sekä muiden toimijoiden kanssa tietyn selvitettävän asian kontekstissa osoittautui ajoittain haastavaksi. Asiakkaan projektitiimin jäsenet eivät olleet mukana ainoastaan palveluväylän kehityksessä, vaan heidän muut tehtävänsä vaativat myös huomiota projektin aikana. Etenkin liikkuvaa työtä tekevien asiakkaan projektitiimin jäsenten kohdalla heidän tavoittamisensa saattoi olla hankalaa ajoittain.

Jos tavoiteltava henkilö oli pitkiäkin aikoja kiireinen muissa tehtävissään tai esimerkiksi työmatkalla Suomen toisella laidalla mobiiliyhteyksien varassa, yhteyden pitäminen ja projektiin liittyvien asioiden edistäminen muodostui haastavaksi. Asiakkaan kyky osallistua projektiin selvästi vaikuttaa kehitystiimin tehokkuuteen [Alleman, 2002; Augustine et al. 2005].

Projektin edetessä ja uusien monimutkaisempien kokonaisuuksien tullessa toteutettavaksi kehitystiimin jäsenet ymmärsivät kyllä paremmin asiakkaan liiketoimintaa ja tarpeita, jolloin ratkaisujen tekeminen helpottui ja vaati vähemmän resursseja asiakkaan projektitiimiltä. Toisaalta kehitystiimin tarvitseman tiedon luonne muuttui monimutkaisemmaksi kuin

projektin alkuvaiheessa ja vaati asiakkaan projektitiimin jäseniltä enemmän aikaa per selvittävä asia. Tämä johti usein keskeytyksiin muiden tehtävien vaatiessa välissä asiakkaan tiimin jäsenten huomiota ja yksittäisten asioiden selvittäminen pitkittyi. Asiakkaan organisaation laajuus sekä suora yhteys ainoastaan asiakkaan puolella projektitiimiin johtivat siihen, että joidenkin asioiden selvittely vaati kysymysten kierrättämistä useamman henkilön kautta ennen kuin vastauksen tietävä henkilö löytyi. Toisaalta asiakkaan projektitiimin jäsenet yleensä tiesivät suoraan keneltä tarvittava tieto löytyy.

Projektin kokonaisuuden kannalta yksittäisten asioiden pitkittyminen nosti sen riskiä, että jonkin tietyn asian selvittäminen pysähtyi tarpeettoman pitkäksi ajaksi kommunikoinnin katkettua.

5.1.3. Oikeiden asioiden kommunikointi

Haasteet kommunikoinnissa projektin aikana osoittavat, että kommunikaation luonnetta, muotoa sekä tavoitteita ei osattu ottaa huomioon tarpeeksi. Tehokkaan kommunikaation ollessa yksi tärkeimpiä ketterien menetelmien kulmakiviä Schwaberin [2004] mukaan, tulisi siihen panostaa huomattavasti enemmän.

Osaltaan hyviä tuloksia projektin aikana saatiin aktiivisesti asioita edistämällä ja pyrkimällä ymmärtämään asiakkaan ongelmia syvällisesti. Esimerkiksi erinäisistä syistä pitkään kestänyt kahden eri järjestelmän käsitteiden yhteensopivuuden varmistaminen nopeutui selvittämällä kummankin järjestelmän käyttötapoja yhdessä asiakkaan järjestelmäasiantuntijan kanssa sen sijaan, että varmistaminen olisi jätetty järjestelmäasiantuntijan vastuulle.

Nonakan ja Toyaman [2003] mukaan tietämyksen siirtymiseen henkilöltä toiselle organisaatorajojen yli vaikuttavat muun muassa ympäristö ja konteksti, jossa tietoa jaetaan sekä henkilöiden roolit tässä kontekstissa. Koska asiakkaan resurssit olivat projektissa rajalliset, suunnitelmallinen ja tarkoitushakuinen kommunikointi näytti tuottavan parempia tuloksia, kuin pelkkien kysymysten esittäminen ja vastausten odottaminen. Opastavan roolin omaksuminen asiakkaan kanssa käytävissä keskusteluissa tehosti asioiden selvittämistä, mutta toisaalta vaati kehitystiimiltä laajaa tietämystä asiakkaan toimialasta, liiketoimintaprosesseista sekä järjestelmistä.

5.1.4. Kehitystiimin sisäinen kommunikaatio

Kehitystiimin sisäinen vuorovaikutus sen jäsenten kesken on projektissa kuin projektissa tärkeää. Erityisen tärkeää se oli tutkittavassa projektissa, sillä eri jäsenten toimesta eri liiketoimintaprosessien mukaan toteutetut integraatiot liittyivät toisiinsa ja muutokset yhden integraation yhteydessä saattoivat vaikuttaa myös toisiin integraatioihin.

Vaikka työkalut mahdollistivat kommunikoinnin tehokkaasti ja kommunikointitavat olivat monipuolisia, projektissa tuli eteen tilanteita, joissa oleellinen tieto ei kulkeutunut toiselle kehitystiimin jäsenelle asti. Tämä johtui suurimmaksi osaksi projektin tiedonhallinnan käytännöistä, joita käsitellään myöhemmin, sekä projektin kontekstissa sirpaloituneista kommunikaatiokanavista.

Ketterässä kehityksessä tiimin tehokkaan toiminnan takaamiseksi tiimin jäsenet luovat yksinkertaiset säännöt, joita kaikki jäsenet noudattavat.

Projektipäällikön tehtävänä on seurata, että sääntöjä noudatetaan ja puuttuu sääntörikkomuksiin parhaaksi katsomallaan tavalla. [Augustine et al. 2005]

Tutkimuksen kohteena olevassa projektissa toimintaa ohjasivat lähinnä organisaation normit sekä suppea joukko laadittuja sääntöjä projektin hallinnan helpottamiseksi. Tällaisia käytäntöjä olivat esimerkiksi perustietovaraston muutostenseurantaprosessi muutosten viemiseksi luotettavasti kehitysympäristöstä aina tuotantoympäristöön asti sekä palaverien dokumentointiohjeet.

Schwaberin [2004] mukaan oleellinen osa kehitystyötä ovat päivittäiset tapaamiset, joissa hetkeksi pysähdytään katsomaan projektin tilaa ja mahdollisia haasteita kehitystyössä. Tutkittavassa projektissa tällaisia tilannekatsauksia ei ollut. Alkuvaiheessa tietyin väliajoin pidettiin koko kehitystiimin kesken tilannekatsauksia, mutta ne olivat epäsäännöllisiä eikä niitä myöhemmin projektin aikana järjestetty usein kehitystiimin pienentymisen takia.

Tiimin sisäisten kommunikoinnin haasteiden takia olisi ollut kannattavaa järjestää säännöllisesti tilannekatsauksia, joissa olisi jaettu tietoa projektin tilasta ja haasteista. Koska projektin monet integraatiot olivat riippuvaisia toisistaan, katsauksissa olisi myös voinut tuoda esille mahdollisesti muiden työhön vaikuttavia tekijöitä.

5.1.5. Asiakkaan näkemys toimintatapoihin

Tutkittu järjestelmäintegraatioprojekti oli vain osa asiakkaan suurempaa projektikonaisuutta, joten kehitystiimi ei voinut vaikuttaa muiden

osapuolien toimintaan kuin korkeintaan välillisesti ja hienovaraisesti. Projektin aikana huomattiin, että samalla tavalla kommunikoinnin suhteen toimivat osapuolet tekivät yhteistyötä tehokkaammin kuin erilaisia käytäntöjä suosivat osapuolet.

Joidenkin järjestelmätoimittajien kanssa projektin aikana pystyttiin suoraan sopimaan toimintatavoista integraatioiden toteuttamiseksi ja testamiseksi, toisiin taas ei pystynyt vaikuttamaan kuin asiakkaan projektitiimin kautta. Epäsuora kommunikointi johti usein projektin aikana asioiden viivästyneeseen selvittämiseen. Tästä syystä asiakkaalla tulisi olla näkemys toimintatavoista, joilla eri osapuolet tekevät työtä.

5.1.6. Kommunikoinnin käytännöt

Fahey ja Prusak [1998] kirjoittavat vaadittavasta henkilöiden välisestä yhteisestä kontekstista kommunikoinnin tiedon välittymisen onnistumiseksi. Tähän liittyen tutkittavassa projektissa kohdattiin tilanteita, joissa eri osapuolet keskustelivat asioista eri konteksteissa. Integraatioprojektiin liittyvien järjestelmien määrä voi olla suuri, mikä johtaa myös termien päällekkäisyyksiin.

Jokapäiväisessä kommunikoinnissa eri osapuolien henkilöt eivät välttämättä välittäneet tietoa kontekstista, mikä johti väärinymmärryksiin. Tämä ongelma oli sitä suurempi, mitä useamman järjestelmän kanssa henkilöt olivat tekemisissä. Toisaalta asiakkaan järjestelmätoimittajien ja kehitystiimin välisessä kommunikoinnissa termien sekoittumisen vaara oli vähäisempi, sillä järjestelmätoimittajien asiantuntijat eivät yleensä olleet tekemisissä asiakkaan

muiden järjestelmien kanssa. Tällöin kehitystiimi osasi päätellä kontekstin siitä, kenen kanssa oli tekemisissä.

Toinen pienempi haaste kommunikaation käytännöissä nousi esille kehitystiimin ja asiakkaan projektitiimin välillä tavoitettavuuteen liittyen. Vaikkakin projektilla oli käytössään monimuotoiset kommunikointitavat, yksittäisillä henkilöillä saattoi silti olla haasteellista tavoittaa toinen henkilö. Tämä johtui esimerkiksi liukuvista työajoista tai toisen osapuolen kiireistä sovittujen palaverien kohdalla.

Schwaber [2004] ja Augustine et al. [2005] huomauttavat kehitystiimin osalta yhteisistä säännöistä, joiden tarkoituksena on luoda yhteinen toimintaympäristö projektille. Tutkittavassa projektissa noudatettiin ennemminkin organisaation normeja kuin määritettiin projektille omia sääntöjä. Tämä toimi kohtuullisen hyvin projektin kontekstissa, mutta aiheutti ajoittain ristiriitoja projektin etenemisen suhteen. Yhteisiä sääntöjä ei myöskään sovittu asiakkaan kanssa kehityksestä, mikä myös silloin tällöin johti kontrasteihin projektityössä.

5.2. Epätietoisuuden hallinta

Lineaarisesti etenevissä projekteissa ennen ratkaisun toteuttamista tehtävän vaatimusmäärittelyn epäonnistuessa tai ollessa puutteellinen, projektin aikataulu tai budjetti saattavat ylittää sovitut rajat. Schwaberin [2004] mukaan asiakas ei välttämättä osaa itsekään kuvata ohjelmistotuotteelta vaadittuja ominaisuuksia etukäteen tarkasti ja vaatimukset voivat muuttua kehityksen aikana. Tähän ketterät menetelmät vastaavat lyhyillä iteraatioilla, joiden päätteeksi asiakkaan kanssa tarkastellaan tuloksia ja hiotaan tarvittaessa

vaatimuksia sekä asetetaan tavoitteita seuraaville iteraatioilla [Augustine et al. 2005].

5.2.1. Toimintatavat epätietoisuuden hallitsemiseksi

Integraatio-projekteissa toteutettaviin osakokonaisuuksiin, sekä yleensäkin yksittäisiin integraatioihin, saattaa liittyä kytköksiä useisiin eri järjestelmiin ja jonkin tietyn integraation valmiiksi saattamiseen voidaan tarvita panosta monelta eri toimijalta. Tästä syystä yksittäisten integraatioiden toteuttaminen lyhyissä iteraatioissa ei ole mielekästä eikä realistista.

Tutkimuksen kohteena olevan projektin alussa kokeiltiin kahden viikon mittaisia tarkastelujaksoja, joiden päätteeksi asiakkaan projektipäällikön kanssa käytiin läpi kehityksen alla olevien integraatioiden tilanne. Tästä kuitenkin luovuttiin, koska lyhyiden määrämittaisten iteraatioiden ei katsottu sopivan projektin ohjaamiseen sen luonteen takia. Tästä keskustellaan myöhemmin projektin hallintaa käsittelevässä osuudessa.

Lyhyiden iteraatioiden tarkoituksena ketterässä kehityksessä on antaa kehitystiimille mahdollisuus saada jatkuvaa palautetta työstään ja muokata kehityksen suuntaa tarvittaessa paremmin asiakkaan tarpeet toteuttavaan suuntaan. Koska toteutusta edeltävän vaatimusmäärittelyn osuus projektista on huomattavasti vähäisempi kuin lineaarisesti etenevissä projekteissa, kehitystiimi ei välttämättä tunne asiakkaan tarpeita laaja-alaisesti. Tämä kuitenkin muuttuu iteraatioiden myötä ja tiimin ymmärrys asiakkaan toimialasta ja tarpeista kasvaa projektin edetessä [Schwaber, 2004].

Iteraatioiden sijaan tutkittavassa projektissa yksittäisten integraatioiden edistämässä suurempi vastuu oli kehittäjillä itsellään. Yksittäiset kehittäjät pitivät huolen siitä, että heidän vastuullaan olevat integraatiot ja projektin osakokonaisuudet edistyivät. Projektipäällikkö pidettiin ajan tasalla edistymisen suhteen ja hän tarvittaessa priorisoi työtä sekä ratkoi eteen tulleita työtä hidastavia ongelmia. Tällä tavalla iteraatioiden sijaan projektin kompleksisuutta pienennettiin jakamalla vastuu osakokonaisuuksista kehittäjien kesken sen sijaan, että projekti pilkottaisiin ajan suhteen pienempiin palasiin.

5.2.2. Tulevaisuuteen varautuminen

Tutkittavassa projektissa asiakkaan toimialan tuntemuksen määrä oli projektin alussa kehitystiimillä hyvin pieni. Lisäksi asiakkaan prosessikuvaukset eivät kuvanneet kaikkia integraatioiden kannalta oleellisia asioita. Osa järjestelmistä oli myös asiakkaalle uusia, sillä niitä otettiin käyttöön samaan aikaan palveluväylän toteuttamisen kanssa.

Siten yksi suurimmista haasteista integraatioiden kehittämisen aikana oli epätietoisuus siitä, miten integroitavat järjestelmät toimivat ja miten niitä käytetään asiakkaan toimesta. Tämä kuitenkin täytyi ottaa huomioon kehityksen aikana, jotta palveluväylä toimisi kaikkien asiakkaan järjestelmien kanssa liiketoimintaprosessien mukaisesti.

Projektin edetessä epävarmuuden määrän huomattiin kuitenkin laskevan samalla kun kehitystiimin toimialatuntemus asiakkaan tietojärjestelmien ja toimintaprosessien osalta nousi. Tällöin esiin nousseet ongelmat saattoi ymmärtää sen sijaan, että ne olisi vain tiedostanut, eivätkä ne vaatineet yhtä

paljon perehtymistä kuin aikaisemmin projektissa. Schwaberin [2004] mukaan epätietoisuuden määrä laskee projektin edistymisen myötä ja tämä vaikuttaa myös työmäärän arviointiin uusien toteutuksia suunniteltaessa.

5.2.3. Muuttuvat vaatimukset

Samaan aikaan palveluväylän toteutuksen kanssa asiakas otti käyttöön myös muita uusia järjestelmiä. Uusien järjestelmien rajapintoja sekä toimintaa tietyissä tapauksissa kehitettiin käyttöönoton yhteydessä, mikä heijastui muuttuvina vaatimuksina palveluväylän kehitykseen sekä perustietovaraston tietomalleihin. Teknisestä näkökulmasta muuttuvat vaatimukset näkyivät lähinnä ylimääräisenä työnä kehitystiimille. Joissakin tapauksissa ongelmaksi muodostuivat arkkitehtuuriin ja mallinnettuihin prosesseihin teknisen toteutuksen sijaan vaikuttavat muutosvaatimukset.

Augustine et al. [2005] korostavat ketterien menetelmien käytössä valmistautumista muutoksiin. Yksittäisen ohjelmiston kehittämisessä muutosten läpivienti on kuitenkin helpompaa kuin palveluväylän tapauksessa. Projektissa asiakkaan näkemys liiketoimintaprosesseistaan tarkentui paljon ensimmäisten tuotantokäyttöönottojen jälkeen, kun integroituja järjestelmiä käytettiin aidosti tuotantoympäristössä.

Kun palveluväylä oli otettu tuotantokäyttöön, muihin järjestelmiin heijastuvia muutoksia ei voinut yhtä helposti enää toteuttaa. Etenkin palveluväylän tarjoamat rajapinnat olivat usean järjestelmän käytössä, joten niiden muuttamista vältettiin projektissa asiakkaan resurssien johdosta.

5.2.4. Integraatioiden skaalautuminen

Tutkittavassa projektissa vaatimuksena joillekin integraatioille oli skaalautuvuus, koska asiakkaan it-ympäristö oli sen tarjoaman palveluvalikoiman osalta heterogeeninen. Integraatioiden piti tukea myös tulevaisuudessa käyttöön otettavia järjestelmiä.

Asiakkaalla ei kuitenkaan ollut resursseja projektin aikataulun suhteen käydä läpi kehitystiimin kanssa kaikkia integroitavia järjestelmiä integraatioiden toteutuksia määriteltäessä. Haasteeksi tässä muodostui rajapintojen määrittäminen tukemaan myös tuntemattomia järjestelmiä.

Littlen [2005] mukaan projektin epävarmuus vaatii projektipäälliköltä kokemusta ja visiota etenemisestä. Tutkittavassa projektissa tuntemattomien järjestelmien osalta määrittelyt onnistuttiin tekemään muodostamalla yleinen tietomalli rajapintojen läpi kulkevista tiedoista tunnettujen järjestelmien perusteella.

5.2.5. Muutosten seuranta

Projektin tiedonhallintakäytännöistä johtuen informaatio oli jakautunut useisiin eri paikkoihin kehittäjien sähköpostilaatikoista muihin järjestelmiin asti. Osa projektiin liittyvästä informaatiosta liikkui ainoastaan kehittäjien ja asiakkaan projektitiimin välisissä keskusteluissa.

Tämän seurauksena kehityksen aikana muuttuneiden vaatimusten ja muiden muutosten seuraaminen aiheutti haasteita kehitystiimille - etenkin kehitystiimin pienentyessä projektin loppupuolella. Koska iteraatioiden sijaan

kompleksisuutta oli vähennetty vastuuttamalla kehittäjiä, yhden kehittäjän siirtyessä toiseen projektiin poistui myös suurin osa hänen tietämyksestään projektiin liittyen. Käytännössä poistuneet kehittäjät edelleen olivat niin fyysisesti toimistossa kuin virtuaalisestikin tavoitettavissa ja lähellä, mutta suuri osa heidän tekemäänsä kehitystyöhön liittyvästä informaatiosta oli läsnä vain heidän ja asiakkaan projektitiimin välissä tapahtuneessa kommunikaatiossa.

Tämä heijastui vastuita toiselle kehittäjälle siirrettäessä informaation puutteena ja epätietoisuutena tehdyistä muutoksista ja sovituista tavoitteista integraatioille. Näitä puutteita paikattiin kehittäjän lähdettyä kuormittamalla tätä usein kysymyksillä.

5.3. Projektin tietämyksenhallinta

Tietämyksenhallintaan osana ketterää projektia ei välttämättä kiinnitetä paljoa huomiota, mutta integraatioiden määrittelemine, toteuttamine ja ylläpito vaativat tiedon keräämistä, omaksumista, tallentamista sekä hyödyntämistä liittyen useisiin eri järjestelmiin ja toimijoihin. Oleellista tietoa ovat eri järjestelmien käyttämät rajapinnat, työnkulut sekä prosessit koko integraatioiden piirissä olevan organisaation laajuudessa. Tarkemmalla tasolla esimerkiksi eri sanomaformaattit ja niiden muunnokset integraatioissa sekä yleisimmät virhetilanteet ja niihin liittyvät prosessit ovat etenkin ylläpidon kannalta tärkeää tietoa.

Tutkimuksen kohteena olevassa projektissa päädyttiin kehitystiimin pienuuden johdosta suosia enemmän tiedon välittämistä kehitystiimin henkilöiden välillä kuin tiedon dokumentoimista projektia tukeviin tietojärjestelmiin. Muuten

tietämyksenhallinnassa luotettiin yksittäisten henkilöiden omiin käytäntöihin eikä projektilla ollut tietämyksenhallintaan liittyviä mainittavia yhteisiä käytäntöjä asiakkaalle ja ylläpitoa varten laadittavaa dokumentaatiota lukuunottamatta. Tämä käytäntö vähensi dokumentaation tekemiseen ja ylläpitoon vaadittavaa työmäärää, mutta toisaalta heijastui välillä projektin toimintaan myöhemmin tässä kappaleessa esitetyillä tavoilla.

Alikappaleissa on kuvattu mitä tiedonhallintaan liittyviä haasteita projektin aikana kohdattiin.

5.3.1. Tiedon jakaminen kehitystiimin sisällä

Tutkittavassa projektissa kehitystiimin jäsenet työskentelivät toisiinsa liittyvien, kuitenkin teknisesti erillisten kokonaisuuksien parissa. Tämä johti projektin tietämyksenhallintamallin mukaisesti eri osa-alueiden tietämyksen keskittymiseen yksittäisille tiimin jäsenille.

Tietointensiivisessä ja ketterässä kehitystyössä tiedon jakaminen eteen päin muille tiimin jäsenille on tärkeää projektin etenemisen suhteen [Cockburn, 2000]. Projektissa ei kuitenkaan aktiivisesti jaettu hankittua tietämystä tiimin jäsenten kesken, vaan suosittiin toimintatapaa, jossa tietoa tarpeen mukaan etsittiin muun muassa tiimin muilta jäseniltä. Reaktiivinen tiedonhaku proaktiivisen jakamisen sijasta näkyi projektissa myöhemmin mainittuina ongelmina.

5.3.2. Oleellisen tiedon löytäminen

Koska suurin osa projektiin liittyvästä tiedosta kulki kehitystiimin jäsenten mukana, oleellisen tiedon löytäminen ja saaminen tarvittaessa riippui siitä, oliko oikea henkilö tavoitettavissa vai ei. Tiimin jäsenten sijoittuminen lähelle toisiaan toimistossa sekä projektin kommunikointia tukevat tekniset ratkaisut mahdollistivat yleensä tiedon löytämisen sitä tarvittaessa. Tämä kuitenkin oli riippuvaista siitä, oliko tiedon omaava henkilö tavoitettavissa oikealla hetkellä.

Projektin aikana eteen tuli usein tilanteita, joissa tarpeellista tietämystä omaava henkilö ei ollut paikalla silloin, kun tietoa olisi tarvittu. Tällöin projektissa jouduttiin joko odottamaan, että kyseinen henkilö olisi taas tavoitettavissa, tai käyttämään aikaa tarvittavan tiedon etsimiseen muuta kautta. Tiedon etsiminen muuta kautta saattoi projektin kontekstissa tarkoittaa yhteyden ottamista asiakkaan asiantuntijoihin tai teknisen dokumentaation läpikäymistä uudestaan toisen tiimin jäsenen toimesta.

5.4. Projektisuunnittelu ja ohjaus

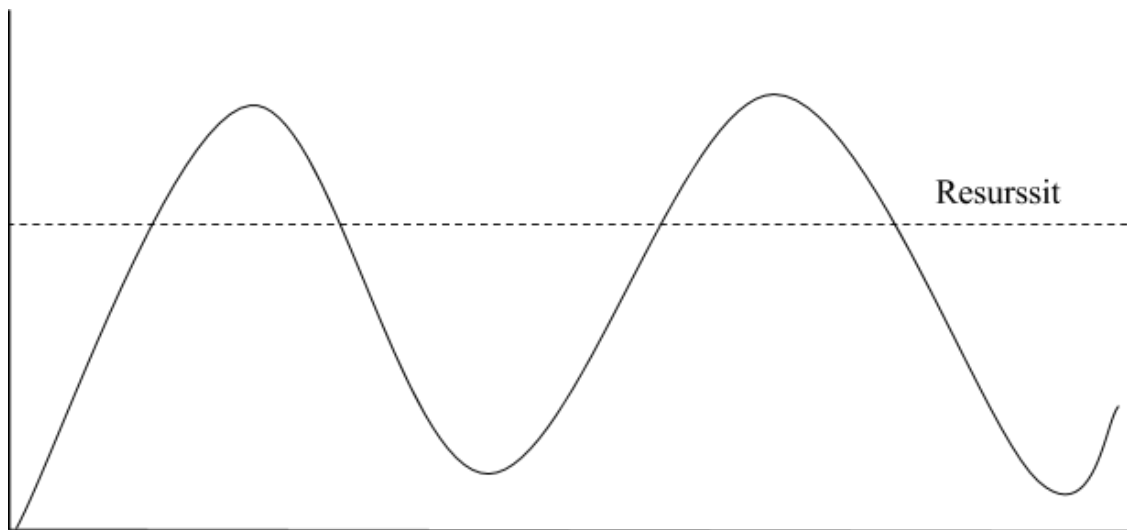
Kuten edellä on mainittu, ketterien menetelmien soveltaminen integraatioprojektiin tuo mukanaan haasteita. Ketterissä ohjelmistokehitysprojekteissa projektipäällikön tehtävä on yleensä ohjata projektia kokonaisvaltaisesti huolehtien eteen tulevista kehitystiimin työtä hidastavista esteistä ja asettaa projektille päämääriä [Schwaber, 2004]. Augustine et al. [2005] korostavat etukäteen suunnittelun sijaan valmistautumista muutoksiin projektin aikana, sillä asiakkaan näkemys projektista ja sen tavoitteista voi muuttua ajan kuluessa ja uusia vaatimuksia todennäköisesti ilmenee projektin aikana.

Järjestelmäintegraatio perustuu suurelta osin sovittuihin sanomaformaatteihin ja rajapintoihin, jotka pysyvät muuttumattomina [Hohpe, 2012]. Tästä syystä muutokset integraatioprojektin aikana todennäköisemmin koskevat integraatioiden tapaa toteuttaa määriteltyjä liiketoimintaprosesseja muuttuneiden vaatimusten muodossa. Tutkimuksen kohteena olleessa projektissa oli mukana asiakkaan kannalta uusia järjestelmiä, joiden rajapintoja ja sanomaformaatteja vasta määriteltiin palveluväylän kehityksen yhteydessä.

Suurimmat tässä luvussa käsiteltävät haasteet projektinhallinnan näkökulmasta liittyivät projektin eri osapuolien resursseihin edistää projektia sekä asiakkaan kannalta kestävien ratkaisujen läpiviemiseen.

5.4.1. Työn resursointi

Palveluväylää kehittävä tiimi teki aluksi täysipäiväisesti projektiin liittyvää työtä. Projektin edetessä kuitenkin huomattiin ettei asiakkaalla tai muilla toimijoilla välttämättä ollut resursseja selvittää kehityksen aikana eteen tulleita haasteita tai toteuttaa muutoksia omiin järjestelmiinsä yhtä tehokkaasti kehitystiimin kanssa. Tämä haastoi etenkin projektipäällikköä, jotta kehitystiimillä saataisiin teetettyä työtä yhtä paljon kuin heitä oli resursoitu projektiin.



Kuva 7. Kuva työmääristä. Pystyakselilla työn määrä ja vaaka-akselilla aika.

Littlen [2005] mukaan paljon projektinhallinnallista epävarmuutta sisältävät projektit vaativat kokenutta projektipäällikköä, joka pystyy ohjaamaan projektia sen haasteista huolimatta ja ratkaisemaan esiin nousevia haasteita ja ristiriitoja.

Tutkimuksen kohteena olleessa projektissa työmäärien tasaamisessa projektisuunnittelulla oli tärkeä osa projektin johtamisessa. Projektisuunnittelulla järjestettiin vaatimuksia aikajärjestykseen toteutusta varten ja tehtiin asiakkaalle ehdotuksia miten projektissa edettäisiin. Tällä tavalla pystyttiin vaikuttamaan työn määrään projektin eri vaiheissa ja tasaamaan kuvan 7 esittämiä työmääriä paljon tasaisemmiksi seuratuilla ajanjaksoilla.

5.4.2. Projektin edistäminen niukoilla resursseilla

Projektin edetessä asiakkaan resurssit suhteessa tarvittavaan osallistumiseen projektin edistämiseksi tehokkaasti pienenevät. Tästä syystä asiakkaan

asiantuntijat olivat ajoittain huonosti tavoitettavissa ja eri ominaisuuksien hyväksymistestaus venyi. Samaan aikaan asiakkaalla oli tarve saada tiettyjä uusia ominaisuuksia tai muutoksia vanhoihin toteutuksiin asennettua tuotantokäyttöön. Tämä aiheutti kehitystiimille ylimääräistä työtä eri ympäristöjen hallinnassa.

Projektissa tämä tilanne vaati ennakoivaa otetta projektipäälliköltä, jotta kehitystiimillä riittäisi työtä tehtäväksi. Käytännössä tämä tarkoitti projektin eteenpäin viemisen suunnittelua ja ehdotusten tekemistä asiakkaalle seuraavaksi aloitettavista toteutuksista. Augustine et al. [2005] mukaan projektipäälliköltä vaaditaan ketterissä projekteissa *ohjaavaa visiota* (Steering vision), jonka avulla projektipäällikkö määrittää ja ylläpitää tavoitteita samoin kuin edistää projektia.

Aktiivisesti ohjaamalla projektia, kehitystiimi pystyi myös vähentämään asiakkaan projektitiimiin kohdistuvaa rasiusta. Tämä kuitenkin vaatii asiakkaan luottamusta kehitystiimiä kohtaan, minkä saavuttamiseksi tiimin täytyy projektin aikana jatkuvasti lunastaa uudelleen ja uudelleen antamansa lupaukset projektin suhteen.

5.4.3. Kestävien ratkaisuiden tekeminen

Asiakkaan it-ympäristö sisälsi useita järjestelmiä, joiden toiminta vaikutti toisiinsa määriteltyjen liiketoimintaprosessien mukaisesti. Uusien järjestelmien käyttöönotto palveluväylän rakentamisen aikana heijastui projektiin muuttuvina vaatimuksina. Vaatimukset täsmentyivät tai muuttuivat ajoittain asiakkaan näkemyksen tarkentuessa liiketoimintaprosesseistaan ja eri

järjestelmien muilta järjestelmiltä tarvitsemista tiedoista. Näkemys muokkautui ajan myötä tuotantokäytön seurauksena.

Muutokset vaativat kestäviä ratkaisuja integraatioarkkitehtuurin kannalta, sillä palveluväylä vaikutti kaikkien muiden järjestelmien keskellä ratkaisevasti muiden järjestelmien toimintaan kokonaisuutena. Asiakkaan projektitiimillä oli ajoittaisesta resurssien vähyydestä johtuen taipumus pyytää nopeaa ja helppoa ratkaisua esiin nousseisiin ongelmiin.

Kokonaisuuden ja palveluväylän ylläpidettävyyden kannalta nopeat ja helpot ratkaisut eivät välttämättä aina olleet parhaita mahdollisia asiakkaan kannalta. Tästä syystä kehitystiimillä ja etenkin tiimin projektipäälliköllä oli muutostenhallinnan suhteen haasteita kommunikoida tilanne asiakkaan projektitiimille. Etenkin muutamissa arkkitehtuuriin liittyvissä muutostarpeissa pikainen ratkaisu olisi pidemmällä aikavälillä vaatinut ylläpidon kannalta enemmän resursseja.

Littlen [2005] mukaan kokeneen projektipäällikön ominaisuuksiin kuuluu tällaisten ristiriitatilanteiden ratkaiseminen projekteissa. Erityisesti integraatioprojekteissa, joissa osapuolia on useita ja päätökset vaikuttavat useisiin järjestelmiin, on tärkeää saada aikaan kokonaisuuden kannalta kestäviä ratkaisuja. Tämän huomattiin tutkittavassa projektissa helpottavan jatkokehitystä ja vähentävän tuotannossa olevan järjestelmän ylläpitoon kuluvia resursseja.

5.4.4. Kokonaisuuksien hallinta

Tutkittavassa projektissa kokonaisuuksia hallittiin jakamalla ne pienempiin osiin, joiden toteuttamista aluksi seurattiin kahden viikon jaksoina ketterien menetelmien mukaisesti. Tämä kuitenkin johti integraatioiden toteuttamisessa siihen, että tietyt seurantajaksoille merkityt integraatiot eivät valmistuneetkaan tietyssä ajassa ja niitä jouduttiin eri syistä siirtämään seuraaville tarkastelujaksoille jatkuvasti.

Lopulta lyhyistä tarkastelujaksoista projektissa luovuttiin, sillä niiden katsottiin sopivan huonosti integraatiotyöhön ja sen hallintaan. Vaihtoehtoisesti osakokonaisuuksien edistymistä seurattiin pidemmällä aikajänteellä ja esteitä ratkottiin sellaisten ilmaantuessa. Tällöin aikataulut joustivat tietyn aikarajan sisällä, mutta pitivät osakokonaisuuksien osalta kokonaisuudessaan.

6. Suosituksia

Tässä osiossa kerätään yhteen tiivistetysti aiempien havaintojen perusteella johdettuja suosituksia ketterien menetelmien soveltamisesta järjestelmäintegraatioprojektiin. Esitettynä on esiin nousseita haasteita, suosituksia paremmista käytännöistä sekä niiden tavoitteista.

6.1. Kommunikaatio

Kommunikaation sujuvuus ja tuloksellisuus eri osapuolten välillä on selvästi yksi tärkeimmistä ketterien projektien onnistumiseen vaikuttavista tekijöistä. Tämä tulee esille etenkin integraatioprojekteissa, joissa osapuolten määrä on selvästi suurempi kuin sovelluskehitysprojekteissa.

| Haaste | Tavoite | Suositus |
|---|---|--|
| Tiedon kerääminen osapuolilta | Tarvittavan tiedon tehokas kerääminen | <ul style="list-style-type: none"> • Ennakoiva ja tuloshakuinen kommunikaatio • Rikas ja jatkuva yhteydenpito osapuolten välillä |
| Kehitystiimin sisäinen kommunikaatio | Työn tehostaminen ja oleellisen tiedon jakaminen | <ul style="list-style-type: none"> • Kommunikaatiokäytäntöihin panostaminen ja niiden suunnitteleminen etukäteen • Tiimille yhteiset säännölliset palaverit projektin tilanteesta ja muutoksista |
| Vaihtelevat käytännöt eri osapuolten välisessä kommunikaatiossa | Osapuolista riippumattomat tehokkaat kommunikaatio-käytännöt | <ul style="list-style-type: none"> • Käytännöistä sopiminen osapuolien välillä pidemmässä projektissa • Asiakkaan kumppaniverkoston käytäntöjen tehostaminen |

6.2. Epätietoisuuden hallinta

Integraatiotyöhön liittyy paljon ratkaisujen kannalta oleellista tietoa, jota ilman integraatioiden onnistunut toteutus ei ole mahdollista. Vähentämällä etukäteen tehtävää määrittelyä voidaan muuttuvien vaatimusten ja niistä aiheutuvan

työn määrää vähentää. Tämä kuitenkin vaatii myös integraatiotyössä tarvittavan laajan järjestelmätuntemuksen huomioon ottamisen toteutusvaiheessa.

| Haaste | Tavoite | Suositus |
|--|---|---|
| Projektin kompleksisuuden hallinta | Integraatioiden tehokas toteuttaminen ilman kattavia määrityksiä | <ul style="list-style-type: none"> • Iteraatioiden sijaan vastuun jakaminen kehitystiimin jäsenten kesken • Projektin jakaminen osakokonaisuuksiksi |
| Muuttuvat ja osittain tuntemattomat vaatimukset sekä integraatioiden skaalautuvuus | Muutoksista ja lisäyksistä aiheutuvan työn minimoiminen | <ul style="list-style-type: none"> • Muutokseen varautuminen jo integraatioiden toteutusvaiheessa • Integraatioarkkitehtuurin suunnitteleminen joustavaksi ja skaalautuvaksi • Vaatimusten taustojen selvittäminen niiden ymmärtämiseksi paremmin |
| Muutosten seuranta | Muutosten todennettavuus ja ajantasaisten määritysten löydettävyys | <ul style="list-style-type: none"> • Muutosten suunnitelmallinen dokumentoiminen • Muutoksenhallintaprosessin kehittäminen |

6.3. Tietämyksenhallinta

Koska projektiin liittyvä hiljainen ja käsitteellinen tieto ovat tärkeässä asemassa tehokkaan työskentelyn kannalta, on tärkeää panostaa myös niiden hallintaan osana projektia.

| Haaste | Tavoite | Suositus |
|--|--|---|
| Oleellisen tiedon löytyminen sitä tarvittaessa | Integraatioiden kehityksen ja projektin hallinnan kannalta oleellinen tieto löytyy helposti | <ul style="list-style-type: none">• Tiedonhallinnan prosessien kehitys• Useisiin kanaviin hajautuneen informaation ja tiedon kerääminen talteen• Tiedon siiloutumisen estäminen• Aktiivinen tiedon jakaminen projektissa |

6.4. Projektisuunnittelu ja ohjaus

Projektinhallinnan näkökulmasta integraatioprojekteihin liittyy yleensä enemmän epävarmuutta kuin sovelluskehitysprojekteihin. Osapuolia ja toteutettavaan ratkaisuun liittyviä asioita on paljon, mikä monimutkaistaa kokonaisuuden hallintaa.

| Haaste | Tavoite | Suositus |
|---|---|--|
| Työn resursointi | Vältetään tilanteita, joissa keitystiimin jäsenillä on enemmän tai vähemmän työtä kuin resursseja | <ul style="list-style-type: none"> • Jatkuva ennakoiva projektisuunnittelu • Mahdolliset projektin ulkopuoliset tehtävät |
| Projektin edistäminen asiakkaan resursseista huolimatta | Projektin pysähtymisen välttäminen | <ul style="list-style-type: none"> • Suositusten tekeminen asiakkaalle projektin toteutuksesta ja aikatauluista • Projektin ohjaaminen aktiivisesti |
| Kestävät ratkaisut | Integraatioarkkitehtuurin kannalta kestävien ratkaisuiden tekeminen | <ul style="list-style-type: none"> • Asiantuntemuksen tarjoaminen asiakkaan kohtaamien haasteiden selvittämiseksi • Tehtäviin ratkaisuihin vaikuttaminen ottaen asiakkaan etu huomioon |

7. Tutkimukseen liittyvää pohdintaa

Ketterien menetelmien soveltaminen integraatioprojektiin näyttäisi mahdollistavan integraatoratkaisuiden nopeamman toteuttamisen. Samalla ketterät menetelmät eivät kuitenkaan kokonaisuutena sovellu suoraan integraatioprojektien käytettäväksi. Toimintaympäristön erilaisuus haastaa etenkin siinä tapauksessa ketterää integraatiotoimittajaa, että muut toimijat eivät pysty vastaamaan ketteryyteen omilla toimintatavoillaan.

Tämä vaatii projektin hallinnan kannalta ennakoivaa otetta, jotta myös muiden toimijoiden resurssit ja toimintatavat voidaan ottaa huomioon etukäteen. Pelkkä reagoiminen eteen tuleviin haasteisiin voi olla tehotonta ja tutkittavan projektin kontekstissa se olisi johtanut tehottomampaan työskentelyyn, sillä projektisuunnittelua käytettiin etenkin työmäärien tasaamiseen projektin aikana.

Projektissa tietämyksen ja projektiin liittyvän tiedon hallinta noudattelivat enemmän organisaation normeja kuin projektin luonteesta johdettavia tarpeita. Tässä tasapainotellaan projektien välisten käytäntöjen yhteensopivuuden ja yksittäisen projektin työn tehokkuuden välillä. Geneerisemmät työnkulut laskevat projektiin osallistumisessa toimintatapoihin perehtymiseen vaadittavaa aikaa näennäisesti, mutta eivät tue projektikohtaista työskentelyä kovin tehokkaasti. Tämä vaatii projektiin uutena henkilönä tulevalta paljon hiljaisen tiedon omaksumista, vaikka kehityksessä käytettävät tietojärjestelmät mahdollistaisivatkin työn paremman ohjaamisen dokumentoitujen käytäntöjen kautta.

Tutkimuksessa keskityttiin havainnoimaan yhtä integraatioprojektia. Tuloksien mukaan ketterien menetelmien soveltamisessa tässä projektissa haasteita oli etenkin menetelmien skaalaamisessa monen toimijan ympäristöön. Tämä näkyi ketterien menetelmien perusajatusten, kuten lyhyiden iteraatioiden toimimattomuutena laajemmassa toimijajoukossa.

Jatkotutkimuksena voitaisiin tarkastella sitä miten iteraatioiden sijaan eri osakokonaisuuksien vastuuttaminen yksittäisille kehittäjille toimisi muissa projekteissa.

Toisaalta tutkitussa projektissa havaitun proaktiivisen projektisuunnittelun hyötyjä voisi tarkastella myös ohjelmistokehitysprojektien näkökulmasta. Tutkimuksessa huomattiin tämän olevan tärkeää tehokkaan projektinhallinnan kannalta ja se voisi myös sovellettuna muihin ketteriin projekteihin tuottaa positiivisia tuloksia.

Tämän tutkimuksen tulosten vertaaminen toisiin samankaltaisiin integraatioprojekteihin voisi kertoa ovatko tutkimuksen tulokset yleistettävissä.

8. Yhteenveto

Myös järjestelmäintegraatiolta vaaditaan tulevaisuudessa yhä tehokkaampaa toteutusta. Läpimenoa integraatioiden määrittelystä tuotantokäyttöön halutaan pienentää ja tätä varten täytyy myös integraatioprojektien käytäntöjä uudistaa. Ketterät menetelmät ovat sovelluskehityksessä muuttaneet projektikäytäntöjä, mutta niiden soveltamisesta integraatiotyöhön ei ole olemassa vielä huomionarvoisia tutkimuksia.

Tutkielmaa varten seuratussa integraatioprojektissa sovellettiin ketteriä menetelmiä kehityksen nopeuttamiseksi. Ketterien menetelmien soveltaminen kuitenkin toi mukanaan myös haasteita, mikä johtuu integraatioprojektien luontaisesta erosta sovelluskehitysprojekteihin liittyen. Integraatioprojekteihin muun muassa liittyy yleensä laajempi toimintaympäristö, jossa on mukana useita osapuolia integraatiotoimittajan ja asiakkaan lisäksi.

Tämä haastaa ketteriä menetelmiä käytettäessä etenkin pohtimaan integraatioprojektin monimutkaisuuden hallintaa. Lyhyet iteraatiot aikaan sidottuna eivät välttämättä toimi kehitykseen liittyvien osapuolien määrän kasvaessa, vaan projektia täytyy hallita esimerkiksi jakamalla vastuuta kehitystiimin sisällä ja luottamalla yksittäisten tiimin jäsenten kykyyn viedä projektia eteenpäin.

Integraatioihin liittyy myös paljon projektin eri vaiheissa tarvittavaa tietämystä ja informaatiota. Erilaiset prosessikuvaukset, rajapintadokumentaatio, muutosten seuranta, järjestelmäkohtainen tietämys sekä toimialatuntemus vaativat integraatioita kehittävältä tiimiltä kykyä omaksua sekä hyödyntää eri

tietolähteitä. Kaiken tiedon ja informaation hallinnan tehokkuus vaikuttaa koko projektin toiminnallisiin edellytyksiin.

Toimintaympäristön laajentuessa kommunikoinnin merkitys projektin onnistumisen takaamiseksi kasvaa. Tutkittavassa projektissa huomattiin ennakoivan ja jatkuvan kommunikoinnin tuottavan hyviä tuloksia. Mitä välittömämpää kommunikaatio oli, sitä todennäköisemmin asiat saatiin selvitettyä nopeasti.

Projektinhallinnallisesti ketterien menetelmien käyttö haastaa projektipäällikköä, sillä eri osapuolien vaikutus integraatioiden kehittämiseen näkyy selvästi jokapäiväisessä työskentelyssä. Tämä vaatii kokenutta projektipäällikköä sekä paljon jatkuvasti tehtävää projektisuunnittelua eteen tulevien projektinhallinnallisten haasteiden ratkaisemiseksi.

Viiteluettelo

[Aidane, 2010a] Samad Aidane, *Agile Practices in Large System Integration Projects*, <http://www.guerrillaprojectmanagement.com/agile-practices-in-large-system-integration-projects>, viitattu 22.03.2014

[Aidane, 2010b] Samad Aidane, *Tailoring Agile Practices for Enterprise System Integration Projects - Part 1*, <http://www.guerrillaprojectmanagement.com/tailoring-agile-practices-for-enterprise-system-integration-projects-part-1>, viitattu 22.03.2014

[Aidane, 2010c] Samad Aidane, *Tailoring Agile Practices for Enterprise System Integration Projects - Part 2*, <http://www.guerrillaprojectmanagement.com/tailoring-agile-practices-for-enterprise-system-integration-projects-part-2>, viitattu 22.03.2014

[Aidane, 2012] Samad Aidane, *Tailoring Agile Practices for Enterprise System Integration Projects - Part 3*, <http://www.guerrillaprojectmanagement.com/tailoring-agile-practices-for-enterprise-system-integration-projects-part-3>, viitattu 22.03.2014

[Alavi and Leidner, 1999] M. Alavi and D. Leidner, *Knowledge Management and Knowledge Systems: Conceptual Foundations and Research Issues*, INSEAD Working Paper, 1999

[Alleman, 2002] Glen B. Alleman, *Agile project management methods for ERP: how to apply agile processes to complex COTS projects and live to tell about it*, Springer-Verlag, *Agile Universe LNCS 2418* (2002), 70-88

- [Augustine et al. 2005] Sanjiv Augustine, Bob Payne, Fred Sencindiver and Susan Woodcock, Agile project management: steering from the edges, *Communications of the ACM*, vol. 48, no. 12 (2005), 85-89
- [Cockburn, 2000] Alistair Cockburn, *Agile Software Development*, Highsmith Series Editors, 2000
- [Fahey and Prusak, 1998] Liam Fahey and Laurence Prusak, The eleven sins of knowledge management, *California Management Review*, **40**, no. 3 (1998), 256-276
- [Hobday et al. 2005] Michael Hobday, Andrew Davies and Andrea Prencipe, Systems integration: a core capability of the modern corporation, *Industrial and Corporate Change*, **14**, no. 6 (2005), 1109-1143
- [Hohpe, 2012] Gregor Hohpe, *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*, Pearson Education, 2012
- [Iivari, 2007] Juhani Iivari, A paradigmatic analysis of information systems as a design science, *Scandinavian Journal of Information Systems*, **9**, (2007), no. 2, 39-64
- [Josuttis, 2007] Nicolai M. Josuttis, *SOA in Practice, the Art of Distributed System Design*, O'Reilly Media, 2007
- [Little, 2005] Todd Little, Context -adaptive agility: managing complexity and uncertainty, *IEEE Computer Society* (2005), 28-35

- [Nonaka and Krogh, 2009] Ikujiro Nonaka and Georg von Krogh, Tacit knowledge and knowledge conversion: controversy and advancement in organizational knowledge creation theory, *Organisation Science*, **20**, no. 3 (2009), 635-652
- [Nonaka and Toyama, 2003] Ikujiro Nonaka and Ryoko Toyama, The knowledge-creating theory revisited: knowledge creation as a synthesizing process, *Knowledge Management Research & Practice*, **1**, (2003), 2-10
- [Piirainen and Gonzalez, 2013] Kalle A. Piirainen and Rafael A. Gonzalez, Constructive synergy in design science research: a comparative analysis of design science research and the constructive research approach, *Liiketaloudellinen Aikakauskirja*, 3-4, (2013), 206-234
- [Salmela et al. 2010] Hannu Salmela, Mikko Hallanoro, Seppo Sippa, Tommi Tapanainen ja Jari Ylitalo, *Ketterän organisaation IT*, Talentum Media, 2010
- [Schwaber, 2004] Ken Schwaber, *Agile Project Management with Scrum*, Microsoft Press, 2004
- [Schön, 1983] Donald A. Schön, *The reflective practitioner, how professionals think in action*, Basic Book Inc., 1983
- [Wolter and Haselden, 2006] Roger Wolter and Kirk Haselden, *The what, why, and how of master data management*, Microsoft Corporation, 2006