

SYSTEMITYÖ

Systeemityöyhdistys SYTYKE ry:n jäsenlehti N:o 4/2008



**Hypetystä
tervejärkisesti**

SYTYKE ry on vuodesta 1979 toiminut valtakunnallinen systeemityöntekijöiden ammatillinen yhdistys, joka kehittää alan ammattilaisten välistä yhteistyötä ja tutkimustoimintaa.

Teemayhdistyksen jäseneksi voivat liittyä kaikki systeemytyöstä kiinnostuneet yksityiset henkilöt, yhdistykset ja yritykset. SYTYKE ry:n toiminta-alueena on koko Suomi. SYTYKE on Tietotekniikan liitto Ry:n jäsenyhdistys.

Lisätietoja SYTYKE ry:stä: www.sytyke.org

TOIMISTO

Susanna Koskinen
Systeemytyöyhdistys Sytyke ry
Talvikkitie 40 A 33
01300 Vantaa
p. 09 56075363
f. 09 56075365
sytyke@hennax.fi

Hallitus 2009

Puheenjohtaja Mitro Kivinen

p. +358 40 589 2724
mitro.kivinen@kolumbus.fi

Seppo Takanen

p. +358 50 581 0140
seppo.takanen@codebakers.fi

Marianne Malila

malila68@gmail.com

Juha Jääskinen

juha.jaaskinen@digia.com

Timo Sundman

timo.sundman@tietoenator.com

Tiina Kiuru

tiina.kiuru@ri.fi

Jyri Partanen

jyri.partanen@sulake.com

Varajäsenet

Ilkka Pirttimaa

p. +358 50 389 0022
ilkka.pirttimaa@stockmann.fi

Altti Lagstedt

altti.lagstedt@haaga-helia.fi

Liittokokous- edustajat

Minna Oksanen

minna.oksanen@gmail.com

Seppo Takanen

seppo.takanen@codebakers.fi

Lauri Laitinen

lauri.laitinen@nokia.com

SYTYKE ry:n hallituksen sähköpostilista:

hallitus@sytyke.ttlry.fi

Sytyttääkö? - Liity jäseneksi



Systeemytyöyhdistyksen jäseneksi liitytään Tietotekniikan liiton kautta (<http://www.ttlry.fi/>, 020 741 9898, jasenasiat@ttlry.fi) valitsemalla jäsenyhdistykseksi Systeemytyöyhdistys ry. Nykyinen Tietotekniikan liiton jäsen voi liittyä joko vaihtamalla jäsenyhdistystä tai liittymällä lisäjäseneksi.

Tietotekniikan liiton henkilöjäsenmaksu vuonna 2009 on alkaen 49€ ilman lehtiä, 1 lehti 63€. Eri-tyisryhmien hinnoittelusta lisätietoja Tietotekniikan liitosta. Lisäjäsenyys maksaa 12€/yhdistys.

Osaamisyhteisöt 2008

Systeemytyöyhdistyksessä toimitaan niin yhdistystasolla kuin aihepiireittäin erikoistuneissa osaamisyhteisöissä. Monipuolisessa tarjonnassamme löytyy jokaiselle jotakin. Vaihtoehtona on myös perustaa omalle kiinnostukselleen uusi osaamisyhteisö - SYTYKE-hallitus toivottaa toimintaehdotukset tervetulleeksi. Osaamisyhteisön toimintaan pääset mukaan liittämällä postia vetäjälle.

JavaSIG - on Javan käyttäjien ja harrastajien intressiryhmä, vetäjänä Simo Vuorinen.
simo.vuorinen@tietoenator.com

ProjektiOSY - ProSY pyrkii yhdistämään Systeemytyön projektitoiminnasta ja sen kehittämisestä kiinnostuneet, vetäjänä Markku Niemi.
make.niemi@kolumbus.fi

TestausOSY - FAST on testauksen keskustelu- ja yhteistyöverkosto, vetäjänä Maaret Pyhäjärvi.
maaret.pyhajarvi@iki.fi

DAMA Finland keskittyy tiedon, informaation ja tietämyksen hallintaan, yhteyshenkilönä Minna Oksanen.
minna.oksanen@gmail.com

ViestintäOSY järjestää yhteistoimintaa viestintäsovellusten alueella, vetäjänä Tapani Ranta.
tapani.ranta@generum.fi

RELA keskittyy relaatiotietokantoihin vetäjänä Lauri Pietarinen.
lauri.pietarinen@relational-consulting.com

MallinnusOSY jakaa tietoa tietojärjestelmien mallintamisesta, vetäjänä Juha Jääskinen.
juha.jaaskinen@digia.com

KäytettävyysOSY vaihtaa kokemuksia käytettävyyden kehittämiskeinoista sekä kehittää omaa käytettävyysosaamistaan. Vetäjänä ja yhteyshenkilönä Helena Venäläinen.
helena.venalainen@op.fi

SOA SIG on keskittynyt palveluarkkitehtuuriin (Service-Oriented Architecture, SOA). Vetäjänä Janne J. Korhonen.
jjk@jannekorhonen.fi



Julkaisija

Systeemityöyhdistys Sytyke ry
Susanna Koskinen
Talvikkitie 40 A 33, 01300 Vantaa
p. 09 5607 5363
f. 09 5607 5365
sytyke@hennax.fi

Päätoimittaja

Minna Oksanen
minna.oksanen@gmail.com
puh. 040 577 6640

Toimitussihteeri

Susanna Koskinen
sytyke@hennax.fi

Taitto

Speaking Bark Tmi
Katja Tamminen
sb@cabaroo.com

Toimituskunta 4/2008

Mika Hynnä (AKE)
Timo Sunndman (TietoEnator)
Marianne Malila (Tietotekniikan opiskelija)

Lisätietoja lehdestä

www.sytyke.org/lehti

Tilaukset

Systeemityölehti sisältyy yhdistyksen
Tietotekniikan liiton suositusten
mukaiseen yhdistyksen jäsenmaksuun.
Vuositilaus 30 €
Irtonumerot 8 €
Hyvissä ajoin ennen painatusta tehty
vähintään 50 kpl lisätilaus 2 €/kpl.
Tilaukset yhdistyksen toimistosta.

Kansikuva

Ilkka Pirttimaa

Seuraava numero

QA toimii / ei-toimi
Toimituskunta:
TestausOSY & KäytettävyysOSY
Ilmestyy: ti 10.3.2009

Painopaikka

T-Print
Ahokaari 1-3
05460 Hyvinkää
Puh. (019) 475 8500

Painos: 2500 kpl
ISSN 1237-0525
15 vuosikerta, nro. 4

Ilmoitushinnat

Takakansi A4 1200 €
Sisäkannet A4 1000 €
Sisäsivut 1/1 800 €
Sisäsivut 1/2 600 €
Sisäsivut 1/4 400 €

Arvonlisävero 0%
Vakiopaikan vähintään vuodeksi
varanneille 20% alennus.

Teksti: Minna Oksanen, lehden päätoimittaja,
BI/DW asiantuntija Diglassa ja
DAMA Finlandin perustajajäsen

Pääkirjoitus: Hypetystä – ei höpötystä tervejärkisesti

Vuonna 2008 halusimme ensimmäisten 3:n lehden kautta esitellä, millainen on sytykeläinen, miksi meitä tarvitaan ja miten teemme työtä. Tämä 4:s lehti kuvaa enemmän sitä, miksi Sytyke on tärkeä.

Noin vuonna 1 ja 2, kun Sytykeen ja DTry:n ”järkiavioliitto” oli astunut voimaan, olimme me DTläiset ihan innoissamme. Toimintaan saatiin ammatillisesti kehittävää sisältöä. Olihan DTry:ssäkin ollut toimintaa – mutta enemmän virkistys- ja viihdepuolella. Sytyke on ammattilaisten yhteisö, jossa löytyy jokaiselle ohjelmistokehitykseen osallistuvalla jotain. Ja mikä parasta sytykeläisyys

on aina ajan tasalla, ellei jopa askeleen edellä – Hypeä, mutta tervejärkisesti, ammattilaisilta ammattilaisille.

Käsissäsi on taas melkoinen lukupaketti, vaikkapa joulunpyhien lepoaikaan kinkun sulaessa, sillä nyt on aika rentoutua ”jouluhöpötykseen” ja keskittyä vuoden vaihduttua uusiin haasteisiin.

Eikä sovi unohtaa, että vuonna 2009 Sytyke täyttää kunnioitettavat 30 vuotta!

Hyvää ja Rauhallista Joulua sekä Sytyttävää Uutta Vuotta!



Sisällys

- | | | | |
|-----------|--|-----------|--|
| 3 | Pääkirjoitus
<i>Minna Oksanen</i> | 18 | Ketterästi tehtyä ja laadukasta – onnistuuko?
<i>Risto Nevalainen</i> |
| 4 | Miksi Sytyke on tärkeä?
<i>Mitro Kivinen</i> | 21 | Kallialan kolumni |
| 6 | Vaatimukset vaatimusten hallinnalle
<i>Sakari Lehtonen</i> | 22 | Mielikuvittele – ideoista innovaatioiksi!
<i>Tarja Raussi</i> |
| 10 | Prosessista koodia napin painalluksella!
<i>Tarja Raussi</i> | 25 | Mukavia kokemuksia SYTYKE-aktiiviudesta
<i>Markku Niemi</i> |
| 13 | Onko TOGAF oikotie onneen?
<i>Sakari Olli</i> | 26 | Seminaarit Sytykkeen toiminnassa
<i>Juha Jääskinen</i> |
| 16 | Kun aika on oikeasti rahaa – enterprise-kehityksen tulevaisuus
<i>Antti Akonniemi</i> | 28 | LinkedIn Sytykkeen toimintaa tehostamassa
<i>Seppo Takanen</i> |
| | | 30 | Kuutamolla |



Mitro Kivinen
on Sytyke Ry:n
puheenjohtaja ja
toimii itsenäisenä
projekti- ja kehi-
tysvalmentajana
perustamassaan
Alter Chief Oy:ssä.

Miksi Sytyke on tärkeä?

Suomessa on noin viisikymmentätuhatta ohjelmointiammattilaista. Sytykkeellä on osaamisyhteisöineen yli parituhatta jäsentä. Koska systeemityö on pohjimmiltaan ohjelmistojen tekemistä, olemme Sytykkeessä ottaneet visioksemme tavoittaa kaikki ohjelmistokehittämisen ammattilaiset.

Ohjelmistokehityksen ammattilaisten joukkoon kuuluu muiden muassa erilaiset suunnittelijat ja määrittelijät, ohjelmoijat ja testaajat sekä iso joukko projekti- ja järjestelmäpäälliköitä. Alan ammattinimikkeitä on runsaasti.

Vaikka yhdistystoiminta ei olekaan muotia, olemme Sytykkeessä ottaneet haasteen vastaan ja haluamme kääntää yhdistyksen jäsenmäärän taas kasvuun. Tämä onnistuu vain, jos olemme oikealla asialla ja tarjoamme jäsenistöllemme sitä, mikä tekee jäsenyyden Sytykkeessä kiinnostavaksi ja tavoitelluksi.

Suomen ohjelmistoala on varsin hajanainen

Ohjelmistoammattilaisia löytyy luonnollisesti ohjelmistoyrityksistä, joiden päätuotteena on jokin myytävä ohjelmisto. Näitä voivat olla erilaiset liiketoimintaa tukevat ja mahdollistavat ohjelmistot, upotettavat ohjelmistot tai vaikkapa tietokone- ja konsolipelit. Koko tietoliikenne- ja telekommunikaatioala on ohjelmistoriippuvainen. Ohjelmistoja tehdään myös muualla teollisuudessa osana jotain muuta myytävää tuotetta tai järjestelmää.

Ohjelmistoja tarvitaan organisaation omiin tarpeisiin, muun muassa pankeissa ja valtion organisaatioissa. Tällaisten ohjelmistojen tekemiseen tarvitaan usein ammattilaisia sekä asiakkaan, että toimittajan joukoissa. Erilaiset integraattorit ja tietotekniikan palvelutalot pullistelevat ohjelmistoammattilaisia.

Eri alojen ohjelmistoammattilaiset valmistuvat eri oppilaitoksista ja ovat usein taustaltaan varsin erilaisia. Myös ammatillinen toimintakulttuuri voi

poiketa erittäin paljon jopa käytetyn kielen osalta niin paljon, että ensialkuun on vaikeaa nähdä mitään yhteistä. Kuitenkin lähes kaikki osallistuvat projektityöhön ja kaikkien työn tuloksena on jonkinlainen tietokoneessa ajattava abstrakti tuotos, jonka tekeminen vaatii sekä teknistä osaamista, että aihealueen asiantuntemusta.

Sytyke Ry tarjoaa kaikille yhteisen riippumattoman foorumin, jolla erilaiset ammattilaiset voivat keskenään keskustella itselleen tärkeistä asioista. Yksi Sytykkeen arvokkaimpia asioita onkin tarjota mahdollisuus keskustella asioista ilman asiakastoimittajasuhteesta johtuvia pidäkkeitä. Tavoitteena on puhua asioista niiden omilla nimillä ja jakaa kokemuksia eri ammattilaisten välillä.

Sytyke korostaa vahvan asiantuntijuuden merkittävyyttä

Perinteinen urapolku suorittavasta työstä esimiesasemaan on yhä harvemman nuoren tavoite. Palkitsevinta ei ole enää ylennys pois asiantuntijatehtävistä, vaan kannustus asiantuntijuuden syventämiseen. Taustalla on sekä asiantuntijuuden yleisen arvostuksen nousu että työmarkkinoiden epävarmuus: parhaille osaajille on aina kysyntää.

Ohjelmistotekniikan ja toimintamallien kehitysvauhti on niin nopeaa, että osaamistason säilyttäminen ja syventäminen edellyttää jatkuvaa teknologian, menetelmien ja kokemusten seuranta. Arkipäivän kiireiden keskellä pysyminen tilanteen ja trendien tasalla on jatkuvasti haastavampaa. Tiedon saannin ja osaamisen kehityksen tueksi tarvitaan yhteisöjä, joissa asiantuntijuus vahvistuu ja välittyy.

Sytykkeen tavoitteena on jäsenten kehittymisen alojensa vahvimiksi asiantuntijoiksi. Näin jäsenyys hyödyttää sekä yksilöitä että työnantajia. Sytykkeen työkaluina ovat kovan tason seminaarit ja painavaa sisältöä tarjoava oma lehti. Yhteisöllisyys ja viihtyminen ovat myös tärkeä osa toimintaa.

Osaamisyhteisöt tarjoavat erityisasiantuntemukseen keskittyvän viiteryhmän, Sytyke koko ohjelmistoalan keskinäisen verkoston

30-vuotiaan yhdistyksen jäsenkunnassa on sekä nuorta innokkuutta, että huikea määrä kokemusta, mukana myös opiskelijoita, jotka saavat näkyvyyttä käytännön työelämään. Tätä valtaisa kokemus- ja osaamiskeräytymää hyödynnetään erilaisissa osaamisyhteisöissä, joissa kuhunkin asiaan erikoistuneet ihmiset pääsevät luomaan omia verkostojaan muiden samoista asioista kiinnostuneiden kanssa.

Yhdistys tarjoaa osaamisyhteisöille toimintalustan, käytännön tukea ja yhteydet muihin yhteisöihin ja yhdistyksiin. Osaamisyhteisö saa yhdistykseltä vuosittaisen budjetin, jonka avulla yhteisö voi järjestää erilaisia seminaareja ja muuta toimintaa. Sytyke tarjoaa osaamisyhteisöilleen mahdollisuuden tuottaa kotivisut Sytykkeen sivujen yhteyteen ja oman sähköpostilistan yhteisön omaa tiedottamista varten. Lisäksi osaamisyhteisö voi ilmoittaa omista tapahtumistaan kaikille Sytykkeen jäsenille sähköpostilla ja kaikille Tietotekniikan Liiton jäsenille erilaisilla Tietokonelehden ja TIVI:n jäsenpalstoilla sekä Liiton tarjoamilla tapahtumakalentereilla.

Yhtenä erittäin toimivaksi havaittu tapa on tehdä osaamisyhteisöjen yhteisiä tilaisuuksia. Näin kahden eri osaamisalueen asiantuntijat voivat kokoontua yhteen saamaan uutta näkökulmaa ajankohtaisiin ja kiinnostaviin asioihin. Vaikka suurin osa yhdistyksen toiminnasta tapahtuukin osaamisyhteisöjen puitteissa, on myös yhdistystasolla oma merkityksensä.

Yhdistyksessä järjestetään kevät- ja syyskokoukset, joiden yhteydessä on perinteisesti pidetty myös asiantuntijaluentoja. Lisäksi yhdistyksen jo 11 kertaa järjestetty laivaseminaari huippuluentoineen kuuluu monen sytykeläisen jokasyksyiseen ohjelmaan. Sytyke toimii myös Tietotekniikan Liitossa sen suurimpana valtakunnallisena asiantuntijayhdistyksenä. Sytyke on mukana koko suomalaisen tieto- ja viestintäalan verkostoissa, ja Sytykkeessä tehdäänkin mielellään yhteistyötä alan eri toimijoiden kanssa.

Sytyke Ry on kokemuspankki

Parhaimmillaan Sytyke toimii kokemuspankkinä, jossa jokainen jäsen muodostaa tilin, jolle kokemukset ja osaaminen talletetaan ja jolta kokemuksia voi jakaa muille. Toisin kuin raha, ei kokemus tililtä jakaessa vähene, vaan se lisääntyy. Aina, kun kokemusta jaetaan, lisääntyy koko yhteisön osaaminen kokonaisuutena.

Kiinnostavat asiat muuttuvat vuosien varrella, yhteisöjä syntyy, yhdistyy ja lakkaa jäsenten tarpeen ja kiinnostuksen mukaan. Kehittyvien toimintamallien soveltaminen vaatii pohdintaa ja ammattilaisten on osattava yhdistää oikea toimintamalli käsilläoleviin tarpeisiin. Vertaistuki tuo asiantuntijan käyttöön kokemuseräistä tietoa ja luo ammattilaisille vahvan selkänojan. Vertaistuen jakamista varten tarvitaan yhteisöjä, joissa asiantuntijuus vahvistuu ja välittyy. Siksi Sytyke on mukana tarjoamassa puitteita kokemusten vaihtoon.

SAMCOM – LUOTETTAVA KUMPPANI



Samcom tuottaa korkeatasoisia tietojärjestelmäratkaisuita ja -palveluja, joiden avulla asiakkaamme kehittävät liiketoimintaansa ja parantavat kilpailukykyään. Samcom yhdistää alan parhaan osaamisen ja parhaat teknologiaratkaisut takaamaan hankkeiden pysymisen budjetissa ja aikataulussa.

Tutustu Samcomiin osoitteessa www.samcom.fi



Sakari Lehtonen työskentelee ARIS-konsulttina IDS Scheer Finland:illa nimikkeellä Senior Consultant. Hänellä on 30+ vuoden tausta it-arkkitehtuurien ja tietojärjestelmämäärittelyjen parissa. Hän on pitkään ollut aktiivinen Sytykeläinen.

Vaatimukset vaatimusten hallinnalle

- tervettä järkeä eli liiketoimintaohjausta vaatimusmäärittelyyn

Vaatimusmäärittely ja vaatimustenhallinta ovat systeemityön yksi perinteinen näkökulma. Parhaimmillaan hyvä ote vaatimuksiin tuo hyötyä tietojärjestelmän kehittämisen koko elinkaaren ajalla. Tietojärjestelmien kehittämisen luonne muuttuu, pohjautuen mm. palveluarkkitehtuurin tuloon ja yhä laajemman valmissovellusten rooliin. Mitä puutteita voisi olla koetellussa tavassa käsitellä vaatimuksia? Mitä uutta otetta prosessilähestymistapa ja jatkuva toiminnan parantaminen tuo vaatimustenhallintaan? Miten liiketoiminta ja IT tässä pääsevät lähemmäksi toisiaan? Näihin kysymyksiin haetaan vastausta seuraavassa. Artikkelivalottaa vaatimustenhallinnan aseman kehittymistä myös pitemmällä perspektiivillä, ja ottaa kantaa aiheen eri oppisuuntiin.

Perinteisen systeemityön ongelmia vaatimustenhallinnan alueella lähdetään tässä tarkastelemaan systeemityön vaiheistuksen kautta. Olkoon valittu työmenetelmä mikä hyvänsä, karkeasti keskeiset vaiheet tyypillisesti ovat vaatimusmäärittely, tietojärjestelmämäärittely, toteutus, ja testaus.

Vaatimusmäärittelyssä paras käytäntö perinteisesti on ollut kerätä Excel:iin tarkalla tasolla vaatimuslueteloita, jotka ovat käytössä koko kehittämisen ajan, niin että testauksessakin verrataan järjestelmän piirteitä vaatimuksiin (Kuva 1.). Testaus kehittyvä yksittäistapausten testauksesta yhä enemmän

män kokonaisten liiketoimintaprosesseihin toimivuuden testaukseen, jolloin vaatimustenhallinnan haasteet vain kasvavat.

Ongelmana on pitkät vaatimuslistat, on helposti tuhansia vaatimuksia, ja ainoa tapa kuvata vaatimusta selostusteksti. Miten teet eroa keskeisen vaatimuksen ja pikku yksityiskohdan välillä? Miten tekstistä saat käsityksen mitä vaatimus tarkoittaa? Miten vaatimukset linkittyvät keskenään? Tuntuuko tutulta! Miten tätä tilannetta voisi parantaa?

Vaatimusten linkitys käyttötapauksiin

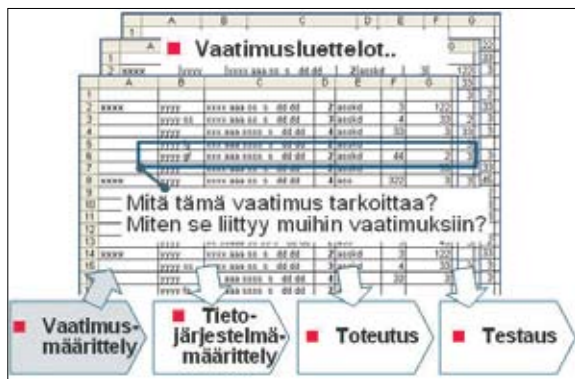
Merkittävä kehitysaskel vaatimusmäärittelyssä on ollut UML:n myötä käyttötapausmallinnus (Use case). Vaatimuslistoissa on osa vaatimuksia sellaisia, että niihin linkittyy tiettyjä käyttötapauksia (Kuva 2.). Käyttötapaus on erittäin selkeä ja konkreettinen tapa kuvata tietyn toiminnallisen vaatimuksen sisältöä. Samalla on välinepuolella menty pelkästä Excel:istä eteenpäin, esimerkiksi Rational-tuoteperheen välineistössä vaatimukset tallettavat tietokantatauluihin, ja vaatimuksen kohdalla voi olla sujuva linkki toisessa välineessä olevaan käyttötapauskuvaukseen.

Onko tässä tutussa työtavassa vielä puutteita? Miten käyttötapaukset puhuttelevat liiketoiminnan puolta? Ongelma on käyttötapauskuvausten it-lähtöisyys, ne kyllä hyvin kuvaavat it-palvelujen sisältöä, mutta linkitys toimintaprosesseihin puuttuu! Miten tätä tilannetta voisi parantaa?

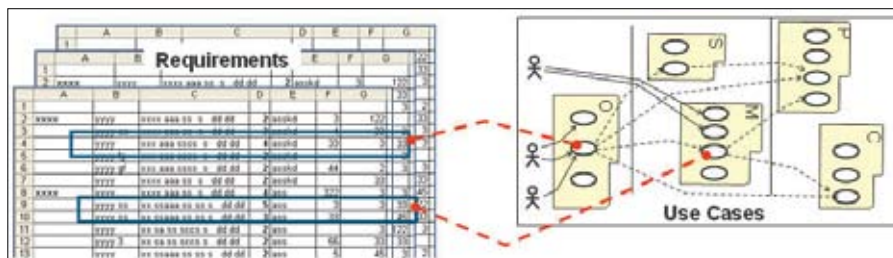
Prosessilähestymistapa ja uudet tuulet!

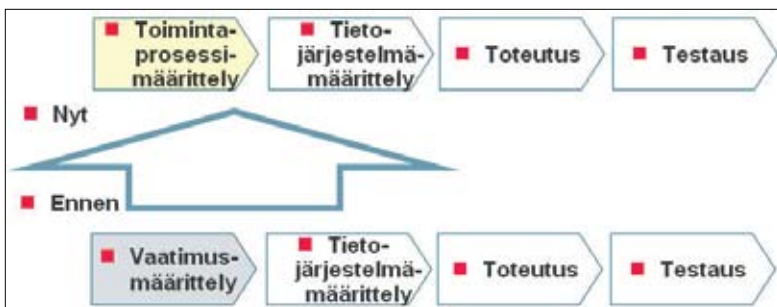
Tarvittava kehitysaskel on lisätä vaatimusmäärittelyn kuvaan mukaan prosessilähestymistapa ja liiketoimintalähtöisyys. Selkeät prosessikuvaukset ovat silta liiketoimintapuolen ja it-puolen keskinäisessä kommunikoinnissa. Prosessimäärittely tuottaa prosessikuvauksia, joihin pohjautuvat it-puolella asettavat vaatimukset. Voisiko vanha vaatimusmäärittelyn kohta vaihejaossa ollakin nimeltään toimintaprosessimäärittely (Kuva 3.)?

Kuva 1.



Kuva 2.





Kuva 3.

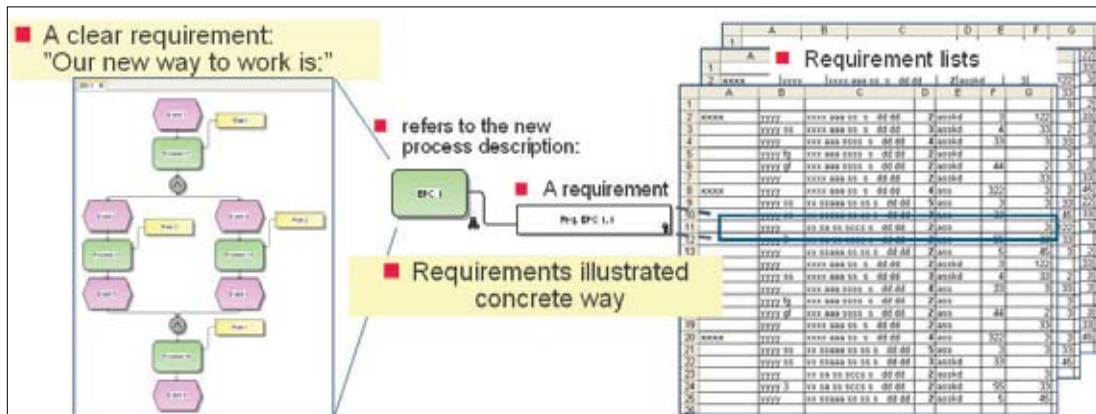
Parhaimmillaan liiketoiminnan toimintaprosessit (BPM) ovat selkeitä, visuaalisia kuvauksia, ja prosessien tietotarpeet IT:n suuntaan myös selkeästi kuvattuja.

Keskeinen uudistus vaatimustenhallintaan on, että osa vaatimuksista linkittykin prosessikuvauksiin. Konkreettinen vaatimuksen kuvaus on, kuvata uusi tapa toimia (Kuva 4.). Välinetueella vaatimukset tallettavat tietokantaan, ja vaatimuksesta voi olla joustava linkki, josta aukeaa prosessikuvaus, joka kuvaa uuden tavan toimia. Työkalussa on vaatimusten molemmat puolet, on sekä UML-käyttötapauslinkitys, että BPM-prosessikuvauslinkitys. Edistyskelliset yritykset ovat rakentaneet jo vuosia hyvällä menestyksellä kokonaisuuksia, jossa prosessihierarkiaan linkitetään käyttötapaus. Parhaimmillaan näin toteutuu muun muassa myös liiketoimintaroolien kuvauskantapohjainen yhteiskäyttö BPM- ja UML-kuvauksissa. Onko tämän uudistuksen jälkeen vielä puutteita? Mitä voisi vielä parantaa?

Jatkuva toiminnan parantaminen

Jatkuvan kehittämisen periaate on vielä ratkaiseva lisäaskel menetelmään. Sinänsä tämä on koeteltu periaate, tunnetaan Demingin ympyränä (Kuva 5.). PDCA (Plan, Do, Check, Act), sen sovellutuksia on tänä päivänä useita, muun muassa Six Sigma DMAIC (Define, Measure, Analyze, Improve, Control). Sykli toistuu tasoittain, kuten kuvassa ylemmän strategiatason syklin Do-osassa toistuu sama sykli tarkemmalla tasolla. Sykli sykliltä ympyrän kieriessä toiminta paranee, mittareina toiminnan laadun ja tehokkuuden valitut mittarit. Toiminnan parantamisen syklin pituus vaihtelee prosessista riippuen.

Vaativana lisänä edellä tarkasteltuun kertaluonteiseen vaatimustenhallinnan tapaan on tämä jatkuva toiminnan mittaaminen ja vertaaminen sykli sykliltä. Vaatimustiedot on koko ajan oltava käytettävissä, ja aina uudella syklillä verrataan uutta tapaa toimia aiempaan ja haetaan parannuskohdat. Esimerkiksi ARIS tarvittaessa vertaa keskenään vaatimuksia, eli uutta tapaa toimia ja aiempaa, ja osoittaa tavoiteltavat parannuskohdat visuaalisesti.



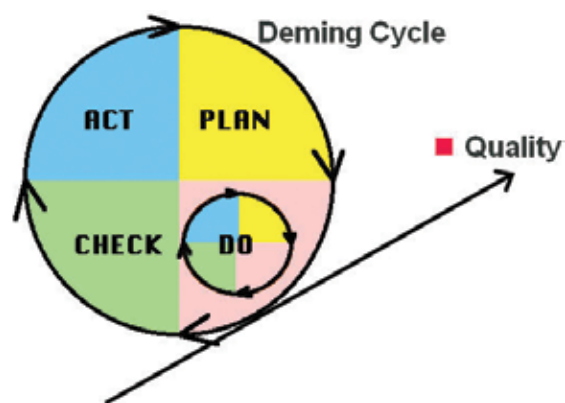
Kuva 4.

Vaatimusten "VOCipannu"

Wokkipannu olkoon vertauskuvana siitä miten kokonaisuus toimii ilman, että pienten osien epäsojivuus pilaa kokonaisuutta. Otsikon VOC ei ole kirjoitusvirhe, vaan Six Sigma-maailmassa yleisesti käytössä oleva lyhenne. VOC eli Voice of customer tarkoittaa asiakasvaatimusten huomiointa ottamista. Ja tässä on liiketoiminnalla ja IT:llä on yhteisiä haasteita. VOCiin liittyvät nimittäin attributit epämääräinen (eri asiakassegmentit, Pareto-sääntö), epästabiili (strategia muuttuu joka vuosi) ja epätasapainossa (kaikki on tärkeää). Epämääräisyyteen on perinteisiä lääkkeitä olemassa, se on määrittelytyötä. Epästabiiliisuus on usein liiketoimintamuutosten syy, ja yhdessä järjestelmähäilyksen ja ylläpidon elinkaarin kanssa se on perinteisesti ollut IT:n ja liiketoiminnan yhteistoiminnan vaikein osa-alue.

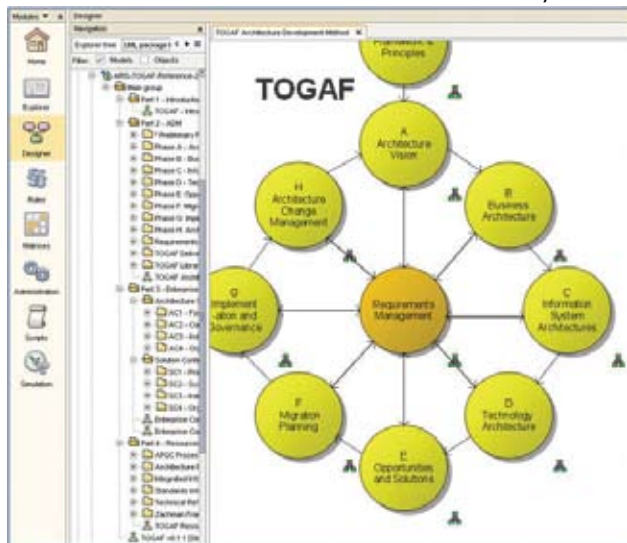
Uudet teknologiat kuten palveluarkkitehtuuri yrittävät vastata tähän haasteeseen, ja myös tämän artikkelin pääsanoma on loppujen lopuksi epästabiiliisuuden hallinnassa. Six Sigma tarjoaa tähän vielä muita työkaluja, kuten Conjoint analyysi ja Design of Experiments (DoE). Vaatimusten hallintaan liitetään usein vielä ominaisuus "epäsuosittu", koska se on kuin "tervaan juontia". Oleellinen haaste tänä päivänä on luoda kehikko liitetointavaatimusten joustavaan hallintaan kuitenkin siten, että myös IT:n vaatima tarkkuustaso toteutuu.

Kuva 5.



Vaatimustenhallinta keskiössä

Vaatimustenhallinta on jatkuvassa toiminnassa keskiössä, tätä asemaa hyvin korostaa toiminnan kehittämisen TOGAF-kehikon visuaalinen ilme (Kuva 6.). TOGAF ei ylipäättään ota kantaa kuvaustapoihin, eikä myöskään siihen, miten vaatimusten hallintaa käytännössä hoidetaan. Työkalun



Kuva 6.

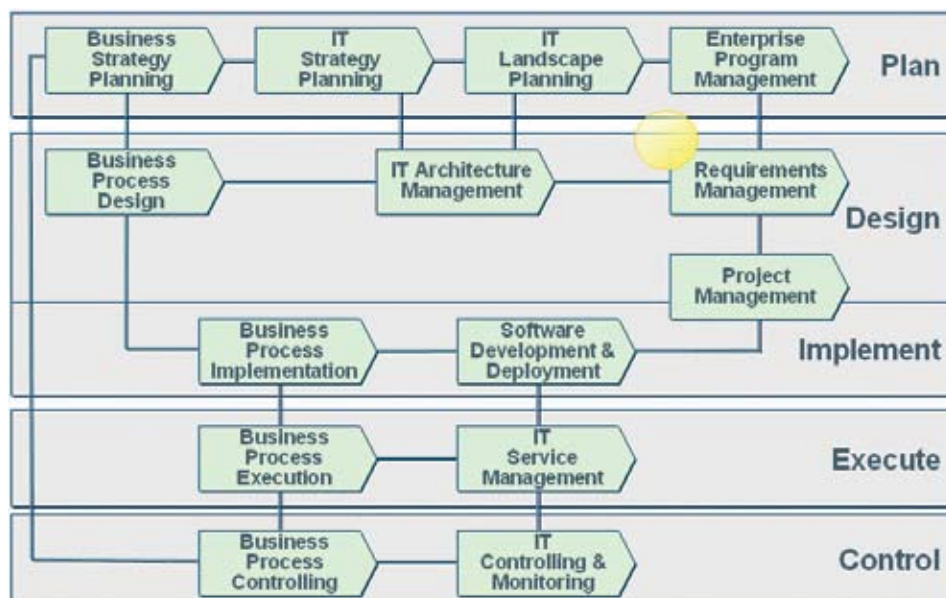
Yrityssuunnittelun kehys

Tässä artikkelissa seuraavaksi laajennetaan näkökulmaa yrityssuunnittelun tasolle. Yrityssuunnittelussa Gartner näkee kolme aluetta (Kuva 7.), yritysstrategia, (Enterprise Strategy), yritysarkkitehtuuri (EA Enterprise Architecture) ja kehittämisohjelmat (PMO Program Management Office). Kaavakuvan mukaisesti, nämä kolme aluetta ovat toisiaan leikkaavia, ne painottuvat eri yrityksissä eri tavoin, ja myös orgnisointitapa vaihtelee. Joka tapauksessa, nämä kolme aluetta aina ovat olemassa, tavalla tai toisella. Yrityssuunnittelun jatkuvan kehittämisen syklin kaavakuvassa nämä kolme aluetta ovat mukana, ja mukana myös keskiön asemassa oleva vaatimustenhallinta (Kuva 8.).

Kuva 7.



Kuva 8.



Yhtenäisen kuvauskannan rooli

Yrityssuunnittelun kolme aluetta tehokkaasti toimiakseen tarvitsevat yhteiskäyttöisiä tietoja, jotka eheä yhtenäinen kuvaustietokanta mahdollistaa. Yrityssuunnittelun tietokanta on vaativa kokonaisuus, jota havainnollistakoon vertaus stadionin rakenteeseen ja rakentamiseen (Kuva 9.). Stadionilla ovat yritysstrategian kuvaukset oma tasonsa. Yritysarkkitehtuuri tyypillisesti jaetaan osa-alueisiin (ks. kuva) prosessit, yhteiskäyttöiset tiedot, palvelut ja sovellukset, teknologiavalinnat ja infra. Kehittämisohjelmien osaan sisältyvät kehittämisohjelmat, niiden projektit, sekä vaatimustenhallinta ja vaatimusten allokointi projekteihin. On huomattava, että kaikkien näiden kuvausten kesken on lukuisia tarpeellisia linkityksiä, tavallaan siitä muotoutuu stadionrakenteeseen verrattava haaste.

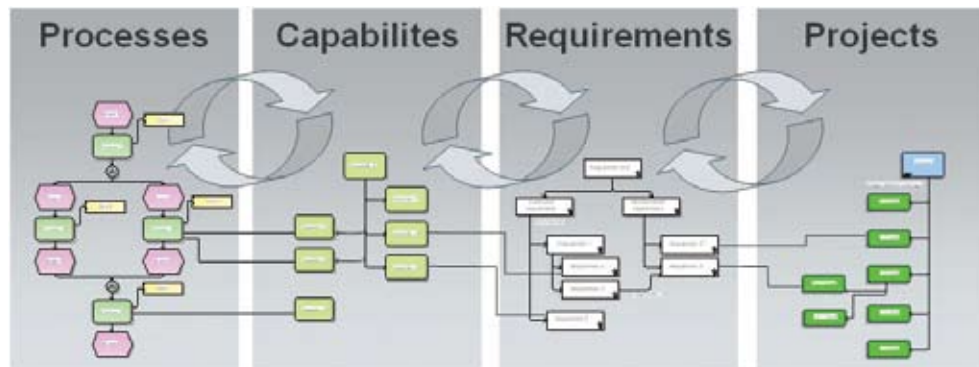
Yhtenä osana kuvauskantaa ovat tässä artikkelissa aiheena olevat vaatimukset, jotka linkittyvät saumattomasti muihin kuvauksiin (Kuva 10). Kyseisessä kuvassa esilläolevat kyvykkyydet ovat prosessien ja vaatimusten välisiä selvittäviä tulintoja, ja vaatimuksista edelleen on allokointi projektien tehtäviksi.

Suunnittelutietokannassa tämä on käyttökelpoinen tapa linkittää eri asioita yhteen ja hahmottaa kokonaiskuvaa. Enää vaatimukset eivät ole irrallisia lauseita, samoin projektien tehtävillä on selkeä linkity vaatimusasetantaan. Miten muutoin tätä kokonaiskuvaa voisi hahmottaa, ellei suunnittelutietokanta sisältäisi näitä elementtejä ja niiden linkityksiä?

On korostettava eroa ja työnjakoa suunnittelutietokannan ja operatiivisen tietokannan kesken. Esimerkiksi kehittämisohjelmien projektisuunnittelussa (Program Management Office) suunnittelutietokanta ja sen tarkkuus on oikea, sensijaan



Kuva 9.



Kuva 10.

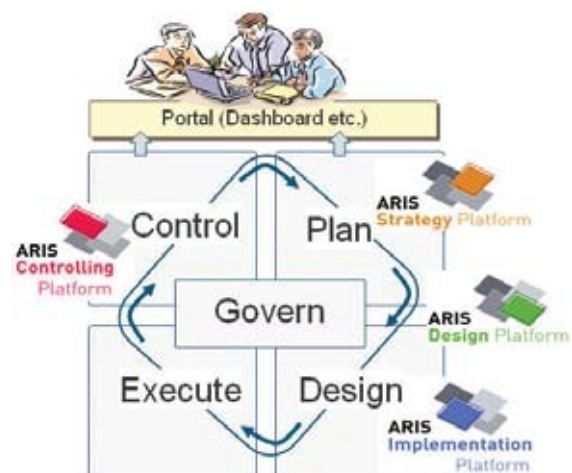
suunnittelutietokanta ei ole operatiivinen projektin läpiviennin tietokanta, vaan sitä varten ovat projektiohjelmistot.

Vaativuuden jäljitettävyys

Omana haasteena on tietysti vielä yksi vaatimusten hallinnan ulottuvuus eli jäljitettävyys. Yhtenäisen kuvauskannan ansiosta on työkalumielessä hyvä valmius olemassa. mutta liiketoimintaihmisille riittävän "kurin" aikaansaaminen kuvausten linkityksissä on monesti vielä edessäpäin oleva karikko. Jäljitettävyyttä ei voi enää kokonaan delegoida IT-ammattilaisten tehtäväksi.

Artikkelin lopuksi, olkoon ARIS esimerkkinä jatkuvan kehittämisen periaatetta tukevasta tuoteperheestä (Kuva 11.). ARIS tuoteperhe on esimerkkisovellutus Demingin syklistä ja sen mukaisesti tuotteet nivoutuvat yhteen. Yhtenäinen eri ARIS-tuotteiden käyttämä ARIS-kuvauskanta puolestaan kattaa yritysstrategian, yritysarkkitehtuurin ja kehittämisohjelmien sisältämät kuvaukset, eli on esimerkki yhtenäisestä stadionlaajuisesta yrityssuunnittelun suunnittelutietokannasta.

Kuva 11.



Prosessista koodia napin painalluksella!?



Tarja Raussi toimii Tieturi Oy:ssä systeemyön asiantuntijana ja tiiminvetäjänä. Viime vuosina hänen tehtäväkuvansa on painottunut erityisesti prosessien mallintamisen ja vaatimusmäärittelyn kouluttamiseen, konsultointiin ja tutorointiin.

Ei kai? Taas joku yrittää vippaskonsteja? Kyllä ja ei. Siihen suuntaan ollaan menossa, mutta päästäkö siihen koskaan – se jää nähtäväksi.

Prosesseja on kautta aikain kuvattu erilaisilla kuvaustavoilla ja kaavioilla. Voi melkein sanoa, että jokaisella konsultilla on oma kuvaustapansa. Mutta nyt on olemassa standardoitu kuvaustapa, joka pyrkii paitsi yhtenäistämään prosessien kuvaamisen, myös viemään sen sille tasolle, että prosesseista tosiaan saataisiin suoritettavaa koodia.

Business Process Modeling Notation (BPMN) -kuvauskielen standardi julkaistiin alkuvuodesta 2006 ja tämän vuoden alussa siitä tuli 1.1 versio. Mikä sitten erottaa tämän aiemmista kuvaustavoista? Ja miksei esimerkiksi UML:n toimintokaavio kelvannut - sehän on myös standardoitu?

BPMN on kaksitasoinen

BPMN-kuvauskieli on kaksitasoinen: kaavio symboleineen ja sitten niihin liittyvät attribuutit.

Kaavion elementeissäänkin on kaksi tasoa: ydinelementit ja sitten niiden täydennykset. Ydinelementtejä ovat erilaiset vuoelementit (flow objects), valintaportit (gateways), tapahtumat (events) ja näitä toisiinsa liittävät yhdistävät elementit (connecting objects). Näiden lisäksi ovat vielä erilaiset uimaradat (swimlanes).

Näiden täydennykset, "koristeet", ovat itse asiassa visuaalisia attribuutteja, jotka laajentavat ydinelementin merkitystä. Visuaalisten attribuuttien lisäksi koko prosessiin ja sen eri elementteihin voidaan liittää sanallisia attribuutteja, jotka täydentävät kaaviota ja vievät sen huomattavasti tarkemmalle tasolle.

Ykkös- eli kaaviotaso on tarkoitettu kenelle tahansa, jonka täytyy kuvata toimintaprosesseja. Kakkostaso eli syvemmälle menevät sanalliset attribuutit erityisesti toteutustavoista on tarkoitettu liiketoiminnan analysoijille ja automatisoijille. Usein tämä taso kiinnostaa IT-ihmisiä ja varsinkin SOA-projekteja tekeviä.

Kaaviot liiketoimintaihmisille

BPMN:n kaavioelementit ovat sellaisia, että kuka tahansa oppii kuvaamaan prosessin kulun kaavioilla. Varsinkin jos pidättäytytään ns. ydinelementeissä, niin kaaviot ovat helppotajuisia ja selkeitä.

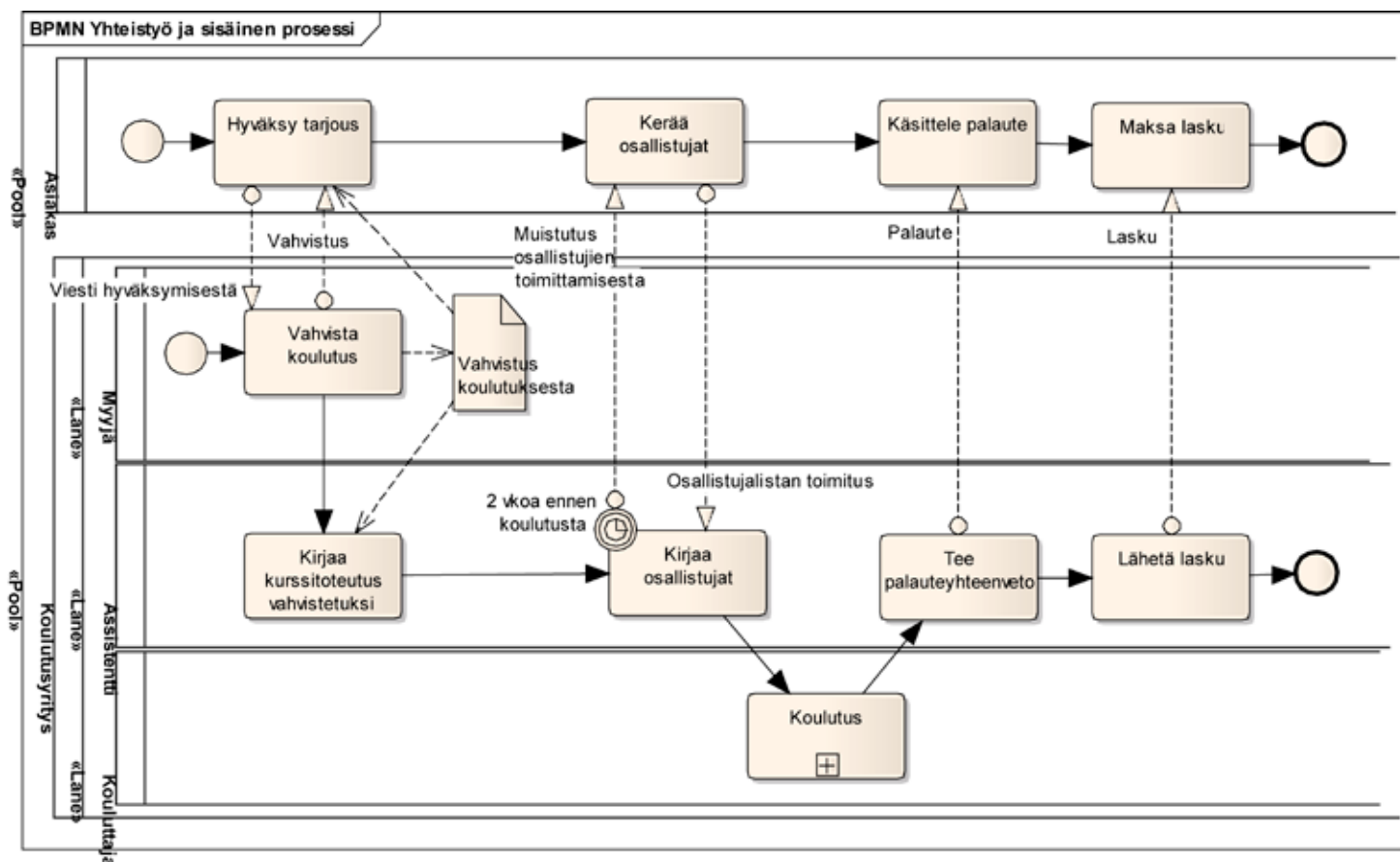
Mistä prosessi koostuu? Prosessikaavioon piirretään tyypillisesti tehtäviä, päätöksentekopisteitä, tuotoksia ja näitä yhdistäviä siirtymiä. Ja vielä kun kuvataan, kuka tekee, niin peruskuvio alkaa olla selvä.

Tehtävä (task) on yksi vuoelementeistä, muita ovat osaprosessi (sub-process) ja transaktio (transaction). Päätöksentekopiste (decision) on eräs BPMN:n valintaportteista. Yleisimmän datan tutkimiseen perustuvan päätöksentekopisteen (data-based exclusive XOR) lisäksi päätöksen perusteena voi olla myös jokin tapahtuma (event-based XOR). Näiden lisäksi BPMN:ssä valintaportteilla esitetään myös samanaikaisuutta ja muunkinlaisia valintavaihtoehtoja.

Siirtymistä toimintavuot (sequence flow) yhdistävät vuoelementtejä, valintaportteja ja tapahtumia toisiinsa. Ne kertovat siis missä järjestyksessä asiat tehdään. Tapahtumia ovat paitsi prosessin alku (start) ja loppu (end) myös välipysähdys (intermediate). Tämä välipysähdys voi kertoa esim. että prosessissa pysähdytään hetkeksi odottamaan jonkin ajan täyttymistä tai viestin saapumista.

Tuotoksien esittämiseen BPMN:ssä on yksi symboli: tieto-objekti (data object). Tämä onkin mielestäni toistaiseksi BPMN:n yksi harvoista heikkouksista. Nimittäin tieto-objektin rakennetta ei voi mitenkään kuvata - paitsi ihan tekstinä. Myöskään tieto-objektien välisiä suhteita ei BPMN kuvaa laisinkaan. Syynä lienee se, että haluttiin keskittyä prosessin kulun kuvaamiseen päästä päähän.

Uimaradoissa eli 'kuka tekee' on BPMN:n eräs vahvuus. Uimaratoja on paitsi aiemminkin tuttu rooli tai toiminnallisuusrata (lane), myös uusi liiketoiminnan osapuoli (pool). Tämä liiketoiminnan osapuoli tarkoittaa aina eri yritystä tai isompaa



Kuva 1.
Yhteistyöprosessi ja
sisäinen prosessi.

organisaatiota, esim. valtionhallinnossa ministeriötä, lääninhallitusta tms. Näin BPMN-kaavioiden avulla voidaan kuvata prosesseja, jotka menevät yritysrajojen yli. Ja tähän on nykypäivää - harvemmin yritys toimii ilman alihankkijoita tai kumppaneita. Näitä liiketoiminnan osapuolia yhdistävät viestivuot (message flow), muualla viestivuota ei saa käyttää.

Erilaiset kaaviot eri tarkoituksiin

Kaavioilla voidaan kuvata puhtaasti yrityksen tai organisaation sisäisiä prosesseja (private process). Tällöin kaaviossa on vain yksi liiketoiminnan osapuoli (pool), joka voidaan siten jättää jopa kuvaamatta. Sen sijaan tyypillisesti kaaviossa on useita rooleja (lane), jotka tekevät asioita.

Jos halutaankin kuvata pelkästään ns. julkista prosessia (abstract, public process) eli miten meidän firman kanssa voi asioida, kuvataan 2-n kpl liiketoiminnan osapuolia, meidän firman sisään vain ulospäin näkyvät tehtävät ja sitten viestivuot toiseen liiketoiminnan osapuoleen.

Toki on mahdollista kuvata myös näiden yhdistelmä, jolloin kaavio alkaa muistuttaa yhteistyöprosessia (collaboration process). Esimerkkinä kuvan 1 prosessikaavio, jossa on kaksi yritystä Koulutusyritys ja Asiakasyritys.

Kaavioiden täydentäminen

Lisää ilmaisuvoimaa kaavioihin saadaan käyttämällä elementeille visuaalisia attribuutteja. Yleisimmät näistä ovat tapahtumiin liittyvät viesti-

ja aikaherätteet. Esimerkikkaaviossa (kuva 1) näemme aikaherätteen: kaksi viikkoa ennen kurssin alkua pitää muistuttaa osallistujalistan toimittamisesta.

Osa visuaalisista attribuuteista voi kuvata esimerkiksi toistoa (luupit) tai satunnaisuutta (ad-hoc). Esimerkiksi artikkelin kirjoittaminen ei välttämättä etene aina niin, että ensin kirjoitetaan otsikko, sitten ingressi, tekstiä, väliotsikko jne. Useimmiten nämä tehtävät pitää kyllä tehdä, mutta ne eivät seuraa toisiaan missään tarkassa järjestyksessä – ne suoritetaan siis ad-hoc (kuva 2).

Prosessille ja sen elementeille voidaan antaa myös sanallisia attribuutteja. Osa näistä on yleisiä, kuten prosessin tai elementin nimi, tunnistetyyppi ja suorittajat. Osa on elementtikohtaisia ja hyvinkin tapahtuman tai toiminnon toteuttamiseen liittyviä.

Kaavioista koodia

Jotta kaavioista saataisiin suoritettavaa koodia, täytyy sen elementeillä määritellä juuri näitä toteuttamiseen liittyviä attribuutteja. Toiminnoista tehtävälle (task) pitää määritellä esimerkiksi syötteet ja tulosteet (input, output) ja toteutustapa (implementation: web service, other) ja sitten tietysti nimetä tuo web service. Eli tässä määritellään siis jo jokaiselle prosessin tehtävälle, mitä proseduuria tai palvelua kutsutaan.

BPMN-kuvauskielen attribuuteista voidaan siis johtaa BPEL-kielen vastineet. BPEL eli Business

Process Execution Language on XML-pohjainen kieli, joka sekin on standardoitu. BPEL-kieli käyttää tyypillisesti palveluita eli web servicejä, tämän takia sitä kutsutaan myös nimellä BPEL4WS eli Business Process Execution Language for Web Services.

Ideana on, että kun BPMN-kaaviosta on generoitu BPEL-kieltä, se voitaisiin sitten suorittaa missä tahansa teknisessä ympäristössä. Jaa-has, ja tässä sitten tullaan siihen fiktion. Perusidea ja standardi ovat hyviä, MUTTA. Jokainen välinevalmistaja, joka on toteuttanut BPMN-kielen välineeseensä, käyttää usein omaa "murretaan" BPEL-kielestä. Niinpä täydellinen yhteensopivuus ja siirrettävyys jäävät usein unelmaksi.

Ja itse asiassa vaikka tuo toimisikin, niin tästä saadaan vasta palvelukutsut, ei varsinaista sisältöä ohjelmalle. Se pitää kuitenkin koodata.

Liiketoimintasääntöjen kuvaaminen toisi lisäpotkua

BPMN ei sisällä liiketoimintasääntöjen kuvaimista. Ne toisivat kuitenkin lisäsisältöä varsinaiseen prosessiin, sillä liiketoimintasäännöt ovat aina päätöksentekokohtien takana - millä ehdolla tehdään mitään. Liiketoimintasäännöt (business rules) ovat tällä hetkellä maailmalla iso hype ja liiketoimintasääntöperustaisia järjestelmiä rakennetaan kovaa vauhtia. Ongelmana on toistaiseksi se, että sääntöjen kuvaustapaa ei ole standardoitu, vaan jokainen välinevalmistaja käyttää omaa kuvaus- ja toteutustapaansa. Standardin työstö on menossa OMG:n (Object Management Group) alaisuudessa. Uskon kuitenkin, että tästä tulemme kuulemaan lisää ihan lähiaikoina.

Miksei UML:n toimintokaavio kelpannut?

UML:n toimintokaaviota (activity diagram) on käytetty jo pitkään mm. toimintaprosessien kuvaamiseen. Miksi sitä ei sitten otettu pohjaksi? Ja varsinkin kun UML 2:n kehitystyössä eräs tehtäväksi anto oli toimintokaavion kehittäminen paremmin prosessien kuvaamiseen sopivaksi.

Sytä lienee useita. Todennäköisesti prosessien kuvaamisen standardointia odottavat kyllästyivät odottamaan UML2:n mammuttiurakan valmistumista - hankehan myöhästyi useamman vuoden. Toisaalta UML on koettu aina enemmän ohjelmistokehityksen kuvauskieleksi, jota puhtaasti liiketoiminnan kanssa työskentelevät saattoivat vieroksua. Eräs teknisempi syy lienee se, että kaikissa mallinnusvälineissä toimintokaavio on ollut aina vain pelkkä piirustus - siihen ei ole koskaan laitettu "paukkuja", saati sitten harkittu sen yhdistämistä ohjelmakoodiin. Tätä itse olin aina ihmetellyt.

Kuitenkin jos UML:n toimintokaavion hallitsee, BPMN:ään siirtyminen on suhteellisen helppoa.

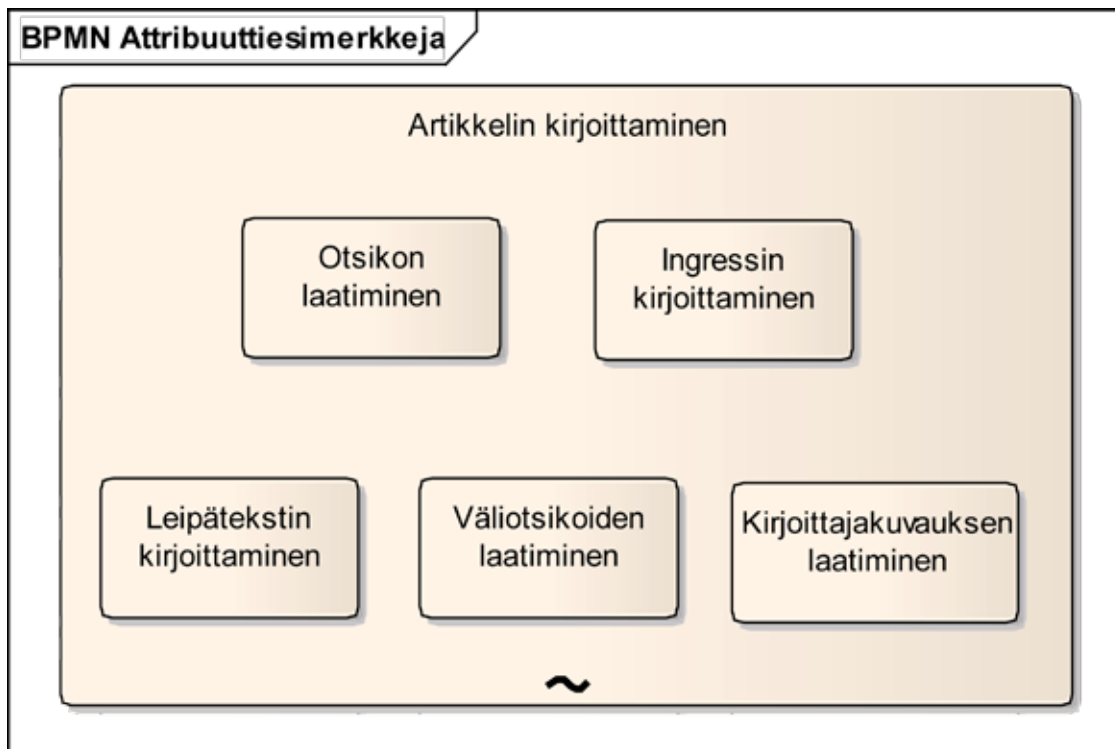
BPMN Suomessa

Itse törmäsin viime keväänä parissa firmassa ensimmäisiin projekteihin, joissa oli lähdetty käyttämään BPMN:ää. Kesän aikana ja tänä syksynä sen käyttö on selvästi lisääntynyt. Kiinnostusta lisää varmasti se, että julkishallinnon neuvottelukunta JUHTA on nimennyt suosituksessaan BPMN:n prosessien kuvaamisessa käytettäväksi tavaksi (JHS 152).

Myös ICT-yrityksissä BPMN herättää kiinnostusta palvelukeskeisen arkkitehtuurin (SOA) hankkeissa.

SOA-projektin onnistumisen perustuu prosessien kuntoon laittamiseen ja niiden kuvaamiseen - ja mikä olisikaan mukavampaa kuin saada koodia SOA-arkkitehtuurin orkestrointikerrokseen suoraan prosesseista.

Kuva 2. Artikkelin kirjoittamisen ad hoc -prosessi.





Sakari Olli työskentelee Tieturi Oy:ssä tiiminvetäjänä sekä asiantuntijana osaamisalueenaan yritys- ja tietojärjestelmäarkkitehtuurit. Hän toimii mm. TOGAF-sertifiointikurssien valmentajana.

Onko TOGAF oikotie onneen?

Parhaita käytäntöjä formalisoivat viitekehukset ovat olleet viime vuosina suosittuja. Suosituista kehyksistä ITIL kokoaa yhteen it-palvelujen hallinnan parhaita käytäntöjä ja COBIT kerää yhteen hyvän tietohallintotavan periaatteita. TOGAF-viitekehys on suosittu väline yritysarkkitehtuurin kokonaisvaltaiseen suunnitteluun. Tämän artikkelin tavoite on auttaa ymmärtämään TOGAF-viitekehystä ilman liiallista hypetystä.

Yritysarkkitehtuuri liittää organisaation liiketoiminnan tavoitteet, avainprosessit sekä informaatiojärjestelmät yrityksen strategiaa tukevaksi kokonaisuudeksi. Integroitu kokonaisuus mahdollistaa parhaimmillaan ydinprosessien ja informaatiojärjestelmien saumattoman ja kustannustehokkaan yhteentoimivuuden. Yritysarkkitehtuuri on myös väline monimutkaisten riippuvuuksien sekä kehittämisprojektien salkunhallintaan.

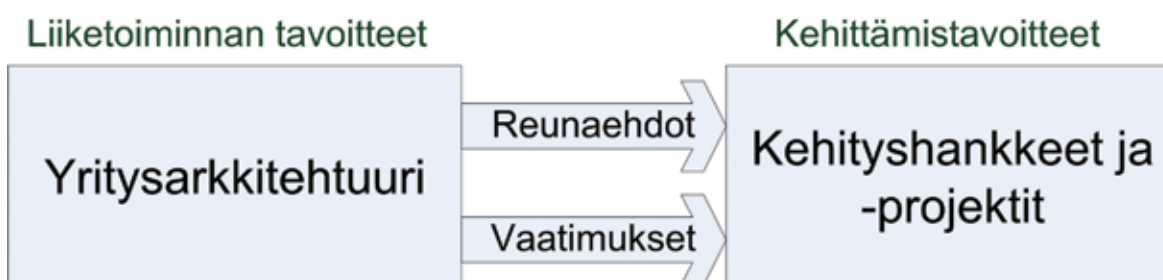
Markkinoilla on useita yritysarkkitehtuurin rakentamiseen tarkoitettuja viitekehyksiä. Suosituimmat ovat tällä hetkellä Zachmann Framework ja TOGAF (The Open Group Architecture Framework). Suurin ero näiden kehysten välillä on se, että TOGAF on vapaasti saatavilla ja tarjoaa metodin yritysarkkitehtuurin suunnitteluun. Vaikka TOGAF-dokumentaatio on vapaasti saatavilla, vaatii sen kaupallinen soveltaminen lisenssin. TOGAFin suosioon Suomessa vaikuttaa mm. sen toimittajaneutraali luonne sekä valinta valtion

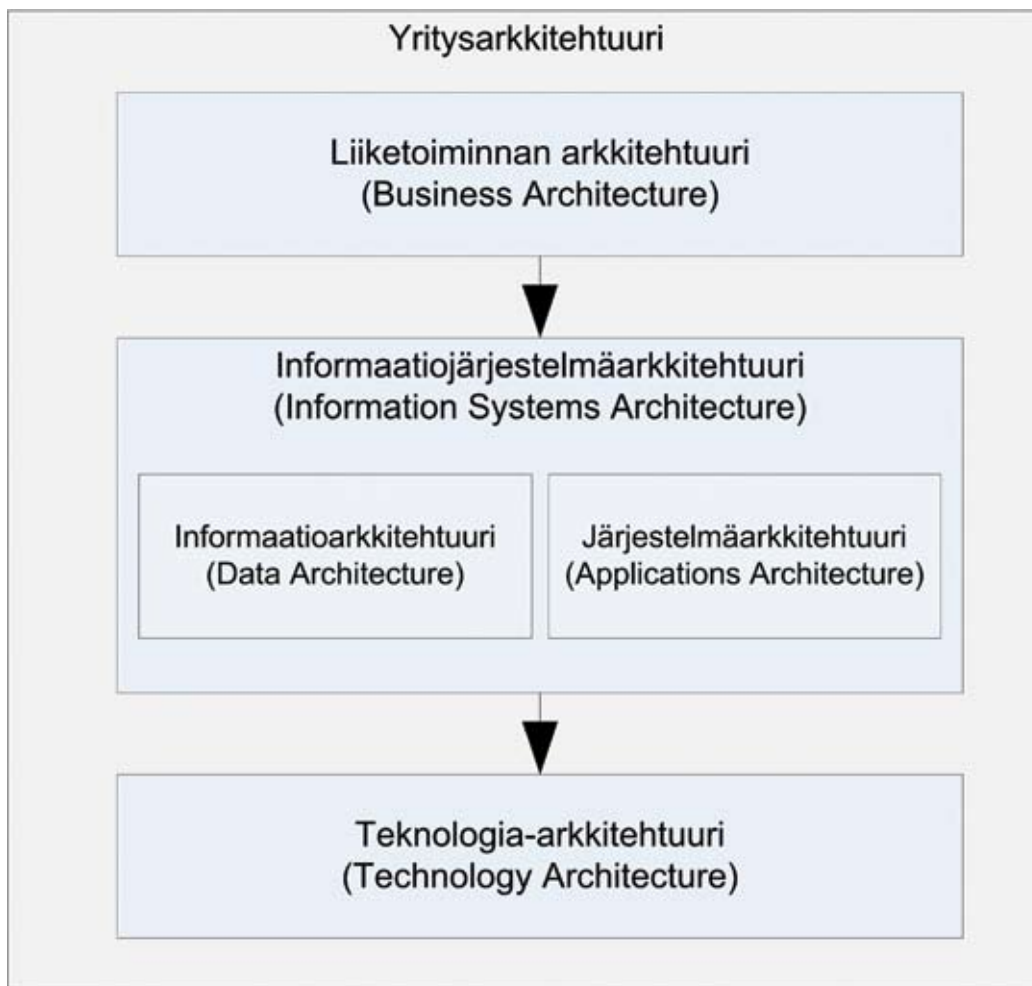
yritysarkkitehtuurityön menetelmäksi. Kehyksenä TOGAF on joustava ja antaa mahdollisuuden räätälöintiin erilaisten organisaatioiden ja toimialojen tarpeita silmälläpitäen.

TOGAF-viitekehyyksen osat

TOGAFin versio 8 jakaantuu kolmeen osaan: ADM (Architecture Development Method), Enterprise Continuum ja Resource Base. ADM on metodi yritysarkkitehtuurin suunnitteluun. Enterprise Continuum on luokittelu, jonka avulla hallitaan arkkitehtuurin uudelleenkäytettäviä rakennuspalasia. Käytännössä sen avulla voidaan esimerkiksi kirjata ylös se, millä tuotteella organisaatio toteuttaa tarvitsemansa käyttäjätunnistuspalvelun. Resource Base on dokumenttikokoelma suunnittelutyön tueksi.

ADM on viitekehyyksen avainkomponentti, joka kuvaa selkeät vaiheet yritysarkkitehtuurin suunnitteluun sekä jakaa vaiheet tavoitteellisiin askeleisiin. Metodien ensimmäisissä vaiheissa etsitään liiketoimintalähtöisesti tarpeita tietyn tyyppisille toiminnoille sekä palveluille. Vasta myöhemmissä vaiheissa mietitään, millä välineillä ne toteutetaan. Esimerkiksi suunnittelun alkuvaiheissa voidaan tunnistaa tarve tunnistamispalvelulle. Sen jälkeen pohditaan, miten löydetty palvelutarve voidaan toteuttaa. Enterprise Continuum tarjoaa luokittelun, joka auttaa kokoamaan yhteen organisaation arkkitehtuurin suunnittelussa ja toteuttamisessa vakioidut komponentit.





Yritysarkkitehtuurin suunnittelu ADM:n avulla

ADM rakentaa yritysarkkitehtuurin neljän arkkitehtuuriulottuvuuden avulla. Käytännössä ADM tarjoaa "rautalankamenetelmän" suunnittelun tueksi ja liittää TOGAFin muut osat toisiinsa. Metodi ei keskity pelkästään perinteiseen yritysarkkitehtuurisuunnitteluun vaan antaa lisäksi ohjeita hankkeiden kannattavuuden arvioimiseen ja arkkitehtuurin käytön ohjaamiseen. Mukana on myös ohjeistusta muutosten ja vaatimusten hallintaan. ADM on iteratiivinen työtapana, ja jokaiselle iteraatiolle täytyy asettaa selkeä tavoite.

Yritysarkkitehtuurin suunnittelutyö lähtee liikkeelle liiketoiminnan arkkitehtuurin suunnittelusta tunnistamalla nykytila ja luomalla arkkitehtuurivisio liiketoiminnan tavoitetilasta. Merkittävä lähtökohta on myös liiketoiminnan tavoitteiden sekä strategian tunnistaminen. Tarkastelussa käytetään apuna kehitystä tarvitsevia näkökulmia. Esimerkkinäkökulmina ADM tarjoaa prosessi-, liiketoimintafunkti- sekä tietonäkymiä. Vaiheen lopuksi verrataan nykytilaa sekä tavoitetilaa toisiinsa ja tunnistetaan puutteet.

Liiketoiminnan rakenteiden ja haasteiden ymmärtämisen jälkeen ADM pureutuu informaatiojärjestelmiin, joita tarkastellaan sekä tieto-

järjestelmätasojen avulla. Molempien osalta suunnittelutyössä on tavoitteena tunnistaa nykytila ja luoda tavoitetila sekä pohtia toimet ja kehittämisprojektit tavoitetilaan pääsemiseksi. Tasojen suunnittelujärjestys voi vaihdella.

Informaatioarkkitehtuurin suunnittelun tavoitteena on ymmärtää liiketoiminnan avainkäsitteet ja tunnistaa niiden suhteet toisiinsa. Suunnittelun tavoitteena on ymmärtää myös tiedon yhteys liiketoiminnan avaintoimintoihin. ADM ehdottaa tietotason analysoinnin välineeksi esimerkiksi liiketoiminnan käsitelmalleja, loogisia tietomalleja sekä tietoturvallisuuden ja tiedonelinkaaren tarkastelua.

Järjestelmäarkkitehtuurin suunnittelussa on alkuvaiheessa tavoitteena tunnistaa millaisia palveluita tuottavia järjestelmiä liiketoiminta ja tiedonhallinta tarvitsevat. Tavoitteena ei ole suunnitella yksittäisten järjestelmien arkkitehtuuria tai valita teknologioita, vaan selvittää millaisia järjestelmiä tarvitaan. Vaiheessa lähdetään liikkeelle organisaation nykyisen sovelluskentän selvittämisestä sovellusten systemaattisen dokumentoinnin avulla. Järjestelmäarkkitehtuurin osalta luodaan tavoitetila ja pohditaan mitkä nykyiset sovellukset ovat siinä mukana.

Teknologia-arkkitehtuuri mahdollistaa liiketoiminnan avainjärjestelmien toiminnan jatkuvuuden. Teknologiatason suunnittelun tavoite on ymmärtää edellisten vaiheiden aikana tunnistetut ei-toiminnalliset vaatimukset ja rakentaa vaatimukset toteuttava kokonaisuus. Toteutuksen välineinä ovat laitteistoihin, tietoverkkoihin sekä ohjelmistoalustoihin perustuvat ratkaisut. TOGAFin teknologia-arkkitehtuuri rakentaa liiketoimintalähtöisen suunnitelman, joka luo perustan toteutustyölle.

Kuinka taataan arkkitehtuurintyön onnistuminen?

ADM tarjoaa apuvälineitä yritysarkkitehtuurin toteutuksen suunnittelemiseen ja ohjaukseen. Menetelmä tarjoaa tukea kehityshankkeiden ja niiden toteuttamismahdollisuuksien arviointiin. Tarjolla on myös tukea kehityshankkeiden kannattavuuden sekä toteuttamisjärjestyksen suunnitteluun.

TOGAF painottaa arkkitehtuurin hallintokäytäntöjä. Viitekehys kehottaa luomaan organisaation, joka vastaa arkkitehtuuriperiaatteiden toteutumisesta, sekä päättämään, miten kehityshankkeita

valvotaan ja ohjataan. Ohjaamisen tueksi on tarjolla apudokumentteja esimerkiksi arkkitehtuurisopimusten ja -katselmointien tekemiseksi.

ADM ei näe yritysarkkitehtuuria muuttumattomana kokonaisuutena. Metodi painottaa, että muutostarpeita on seurattava. Jos liiketoiminnassa tai teknologioissa tapahtuu merkittäviä muutoksia, täytyy ADM kierroksen vaiheet toistaa.

Toimiiko TOGAF todellisuudessa?

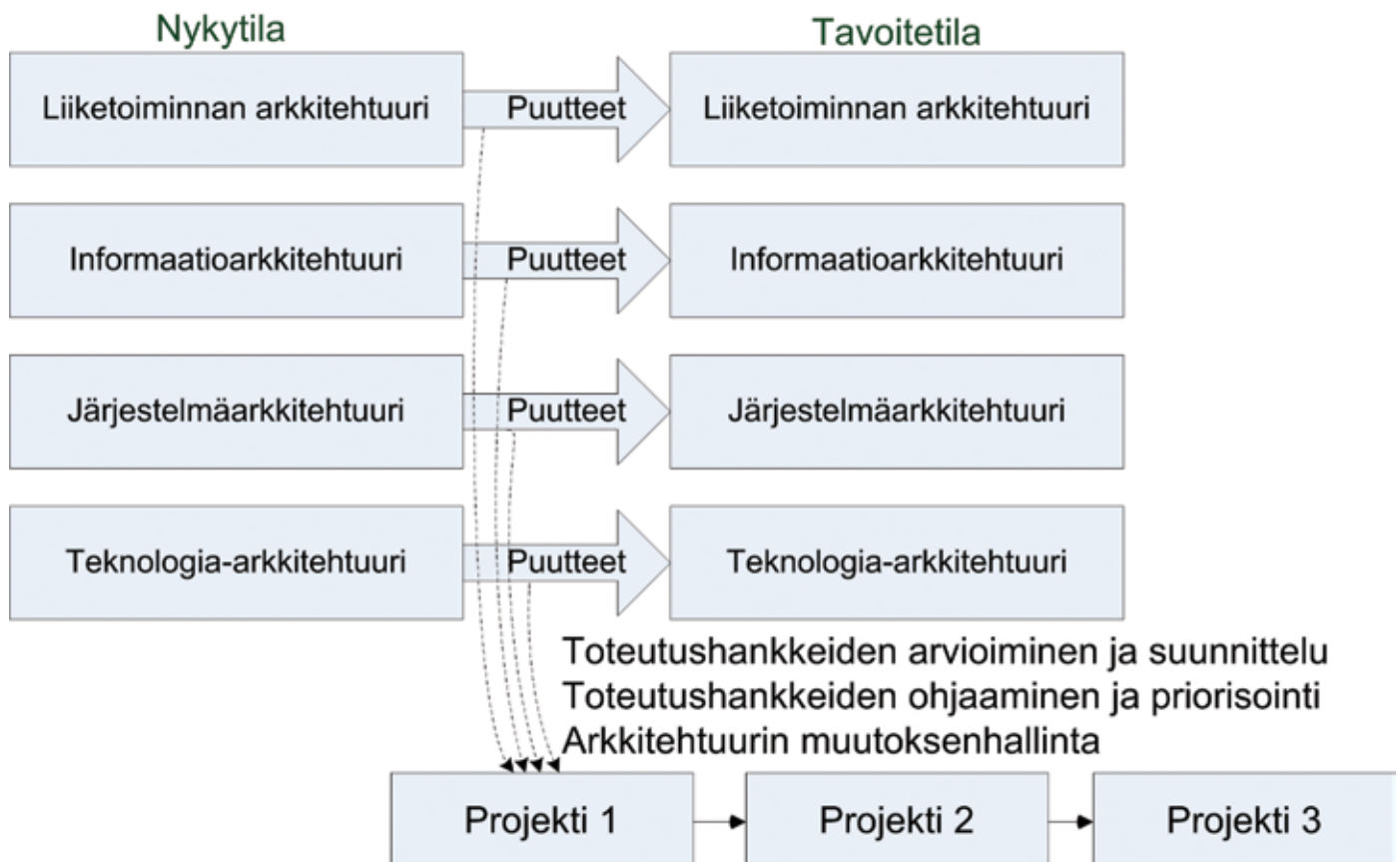
TOGAF tarjoaa iteratiivisen ja räätälöitävissä olevan menetelmän yritysarkkitehtuurin suunnitteluun. Oikein sovellettuna TOGAF tarjoaa tukea arkkitehtuurityöhön, mutta ilman syvällistä osaamista se johtaa umpikujaan. Viitekehyksen käyttö vaatii soveltamista, ja jokaisen askeleen orjallinen toteuttaminen luo massiivisen dokumentaation, joka ei sinällään takaa onnistumista. Dokumentaation tarkkuustasoa huomattavasti merkittävämpi asia on hankkia yritysarkkitehtuurihankkeelle kestävä tuki organisaatiossa ja pyrkiä takaamaan hankkeen maksimaaliset edut organisaatiolle. Myyntityö on haastavaa, sillä kyseessä ei ole lyhyen aikavälin investointi. Ketterä soveltaminen on avain menestyksekkääseen TOGAFin käyttöön.

Valmiit viitekehykset ovat tällä hetkellä suosituja, sillä niiden kuvitellaan tarjoavan yksinkertaisia välineitä suunnitteluun. TOGAF helpottaa suunnittelutyötä, mutta ilman arkkitehtuurityön sudenkuoppien ymmärrystä se ei tarjoa automaattista

tista tautta onnistumisesta. Yritysarkkitehtuurien suunnittelu on aina vaikeaa, sekä aikaa että kommunikointia vaativaa työtä. Yksikään malli ei voi taata automaattista onnistumista monimutkaisten asioiden kanssa toimittaessa.

Meneekö TOGAF 8 sileäksi?

Keväällä 2009 julkaistaan virallisesti TOGAF-viitekehyksen 9. versio. Tuleva versio on edeltäjänsä kaltainen ja esimerkiksi ADM-menetelmä säilyy samankaltaisena. Uusi versio tulee kuitenkin tarjoamaan aikaisempaa enemmän tukea mallin räätälöimiseen sekä kattavamman tukidokumenttaation suunnittelun tueksi.





Antti Akonniemi toimii ohjelmistokehittäjänä ja toimitusjohtajana Kisko Labs Oy:ssä. Antti on myös yksi Suomen Ruby Brigaden (www.rubybrigade.fi) perustajia.

Haluatko vaihtaa leiriä? Ota yhteyttä ja liity taisteluun saamattomuutta vastaan!

Kun aika on oikeasti rahaa – enterprise-kehityksen tulevaisuus

Ruby on Rails on alun perin 37Signals -yhtiön David Heinemeier Hanssonin kehittämä web-ohjelmointikehys, jonka pohjalla on japanilaisen Yukihiko Matsumoton kehittämä Ruby-ohjelmointikieli. Rails kehitettiin auttamaan 37Signalin web-kehitystä. Se vastasi kysymyksiin siitä, kuinka toteuttaa web-sovelluksia ja -palveluita mahdollisimman helposti, nopeasti, elegantisti ja haus-

kasti. Railsin avulla on mahdollista tehdä helposti ylläpidettävää koodia. Uudet yksikkötestikehukset mahdollistavat sujuvan testien kirjoituksen. Ei enää tekosyitä sille, että testien kirjoittaminen olisi tylsää, hankalaa tai inhottavaa. Täysin yksikkötestattua koodia on ilo ylläpitää ja muokata. Näin ylläpito on myös halpaa. Uusina tuulina mainittakoon tuki ja työkalut Behaviour Driven Developmentille (BDD) sekä Story Driven Developmentille. Ylläpitoa helpottaa myös Railsin korkea abstraktiotaso. Haluatko vaihtaa tietokannan MySQListä Oracleen – no problem! Platformista toiseen – ei hätää! Toisin kuin monessa muussa kehityksessä, Railsissa abstrahointi toimii.

Toisin sanoen, Rails muuttaa ohjelmistokehityksen taloutta visiolla. Sovitut käyttäytymismallit ja tietynlainen järkeäisyys nopeuttavat ohjelmistokehitystä sekä laskevat kuluja.

Bisnes mielessä

Kehittäminen Railsilla on nopeaa. Ensimmäinen prototyyppi on valmis minuuteissa tai viimeistään tunneissa. Osallistuin äskettäin Java-projektiin, jossa olin hämmentynyt projektin starttauksen kestäessä viikkoja. Senkään jälkeen projektin tuotosta ei mielestäni voinut kutsua prototyyppiä. Onhan se XML-tiedostojen konffaus varmasti suurta hupia, mutta riittääkö hupia todella viikoiksi?

Liiketoimintakin lienee kiinnostunut nopeasta kehityksestä. Enkä puhu nyt ison sinisen konsultitalon ikuisuusprojekteista, vaan sellaisista, joissa asiakas saa mahdollisimman nopeasti arvoa tuottavan sovelluksen käyttöönsä pienellä investoin-

nilla. Ajattele, jos liiketoiminnan ideoima sovellus olisi käytössä jo kuukauden päästä. Ei pelkästään säästöä kehityskuluissa vaan sovellus ulkona tuottamassa rahaa huomattavasti nopeammin. Liian hyvää ollakseen totta? Kokeile itse.

Investointi on muuten hauska sana. Se odottaa jotain vastineeksi. Kulu taas ei. Kuinka moni ohjelmistoprojekti on yritykselleen kulu sen sijaan, että sitä voisi kutsua investoinniksi? Ja tuleeko kehittäjiä piitata tästä? Ei varmasti kaikkien, mutta jokaisen, joka kehtaa kutsua itseään ammattilaiseksi, olisi hyvä uhrata asialle ajatus. Miksi? Ymmärrän, että nämä ovat monelle tylsiä ja ehkä jopa turhia asioita pohdittavaksi. Väitän silti, että kannattaa miettiä, miten leipä pöydällesi saapuu. Kaikki ei pyöri mahdollisimman nopean algoritmisi ympärillä, vähiten nykyisessä finanssikriisissä. Samaan hengenvetoon voisin haastaa teidät yrittäjiksi, mutta jätetään se toiseen kertaan.

DRY - Don't Repeat Yourself

Railsin yksi tärkeimmistä ideologioista on DRY. Älä toista itseäsi. Se pätee omaan koodiisi ja sitä kautta myös muihin projekteihin. Ihmiset julkaisevat moduleitaan toisten käytettäväksi ja muokattavaksi. Paljon tärkeitä ohjelmistojen osia löytyy valmiina käytettäväksi ja vieläpä abstrahoituna sille tasolle, että se ei rajoita ohjelmaasi. Tarjolla on työkaluja tarpeeseen kuin tarpeeseen.

Nuorella nuoren ongelmat

Ennen kuin alan kuulostaa Railsin provikapalkkaiselta myyjältä, on hyvä käsitellä Railsin puutteita. Rails ei ole valmis. Sovelluskehys on vielä nuori ja on paljon tehtävää, ennen kuin voidaan puhua kypsästä työkalusta. Useimmiten Railsia koskeva kritiikki koskee skaalautuvuutta. Suurimpana syynä tähän pidetään Twitter.comia, jonka palvelu tökki pahasti noin vuosi sitten. Tekijäkaarti kaatoi syyt Railsin päälle. Osittain syy oli Railsin ja osittain Twitterin arkkitehtuurin. Tämä tapah-tuma kuitenkin herätti Railsin kehitystiimin panostamaan suorituskykyyn. Työ jatkuu edelleen, mutta kokemuksen ansiosta merkittäviä parannuksia on jo tehty.

Mainittu Twitter on todennäköisesti käytetty Railsilla tehty sovellus. Toinen esimerkki korkean kuorman sovelluksesta on Friends for Sale Facebook -applikaatio. 300 miljoonaa sivulatausta kuukaudessa. Kuinka monta sivua teidän Java-portaalinne jakeikaan?

Kuten muillakaan kielillä, suurimmat ongelmat skaalautuvuuden kanssa eivät johdu kielivalinnasta vaan toteutuksesta. Rails ei ole yksin alkuaikojen skaalautuvuusongelmiensa kanssa. Myös Javalla oli nuoruudessaan vastaavia ongelmia, jotka nyt on ratkaistu. Rails ei myöskään ole kaikkea kaikille. Tärkeintä onkin valita paras työkalu käsillä olevaan työhön.

Meressä on paljon kalaa

Entä muut vaihtoehdot, kuten Python ja Django? Django on suunniteltu enemmän web-sivustojen ja -sivujen rakentamiseen, kun taas Rails pikemminkin web-palveluja ja -ohjelmistoja varten. Python ja Django ovat varmasti parempia vaihtoehtoja kuin Java tai PHP, mutta väitän, että Rubylla ohjelmointi on silti tehokkaampaa. En ole Python-asiantuntija, mutta en voi ymmärtää, kuinka pääsisin samaan tai nopeampaan kehitystahtiin käyttäen Djangoa/Pythonia, kuin minkä nyt saavutan Railsin ja Rubyn avulla. Ehkä tämä johtuu siitä, että päähäni sattuu, kun luen Python-koodia. Kieli ei vain aukea minulle. Onneksi tämän artikkelin maksimipituus estää minua kertomasta, miltä Java-koodin kirjoittaminen tai lukeminen tuntuu. Jos haluat, voin kuvailla filistä joskus naamatusten.

Mistä koodaajia?

Ammatikseen Railsia käyttäviä on vielä harvassa, saati sitten kilpailevia Django-/PHP-ohjelmistokehityksiä käyttäviä. Sen sijaan harrastajia, kiinnostuneita ja kohta työttömiä Java-koodaajia on paljon. Fiksujen ja innostuneiden kaverien kouluttaminen Rails-osaajiksi on helppoa, koska kieli on niin looginen. Täysin oppineeksi ehtiminen kestää hiukan kauemmin. Rails-osaajien lukumäärä kasvaa koko ajan ja isotkin organisaatiot ovat lähteneet leikkiin mukaan. Muun muassa Nokia, IBM ja Futurice tuottavat Suomessa Rails-koodia. Töitä siis pitäisi piisata.

Rubyn suosio on kovassa nousussa (kuva 1), mutta mistään mainstream-kielestä ei vielä puhuta. Ruby on viimeisimmän (lokakuu 2008) TIOBE Programming Community Indexin mukaan kymmeneksi suosituin kieli. Top5: 1. Java, 2. C, 3. C++, 4. (Visual) Basic, 5. PHP.

Kuoleva Java

Vaikka puhun tässä Railsista, suuri kiitos Javan kuolemasta menee PHP:lle. PHP muutti ohjelmistokehitystä oikeaan suuntaan todistaessaan, että ohjelmistokehitysprojektin budjetiksi ei tarvita miljoonia euroja vaan pienemmälläkin investoinnilla selvittää. PHP todisti, että ei tarvitse olla insinööri tuottaakseen web-palveluja. Ja kyllähän

Javaakin ymmärtää, vaikkei olisi penaaalin terävin kynä. Kokonaisuuden hanksaaminen vaatii kuitenkin hieman enemmän.

Java on ns. yleiskieli ja olemassa olevien Java-ohjelmistojen määrä on huikea. Java-ohjelmoijia on todella paljon. Yleiskielisyys ja kooderilauma eivät kuitenkaan mielestäni oikeuta Javan valintaan ohjelmistoprojektin toteuttamiseen. Se, että organisaatiossa käytetään muissa järjestelmissä Javaa, saattaa jo olla parempi syy. Tosin nykyään Railsinkin saa toimimaan Java-alustalla käyttäen jRubya tai jopa .NETin päällä IronRubylla.

Leikki sikseen

Mutta olkoon provosointi. Henkilökohtaisesti käytän Rubya siksi, että se on vilpittömästi ensimmäinen ohjelmointikieli, jonka ymmärrän. Railsia siksi, että saan sillä asioita aikaiseksi. Paljon. Ohjelmointikieliä on monia, mutta sen sijaan, että tekisi kaiken kuten aina ennenkin, tulisi miettiä myös sitä, mitä kielivalinta merkitsee projektille, liiketoiminnalle ja organisaatiolle. Hyvä kysymys on myös se, mitä ja millä työkalulla haluat tulevaisuudessa tehdä töitä. Töitä... Heh. Kun työskentelen Rails-projektin parissa, se ei tunnu työltä. Tekisin samaa hommaa ilman palkkiotakin. Aina-kin melkein.

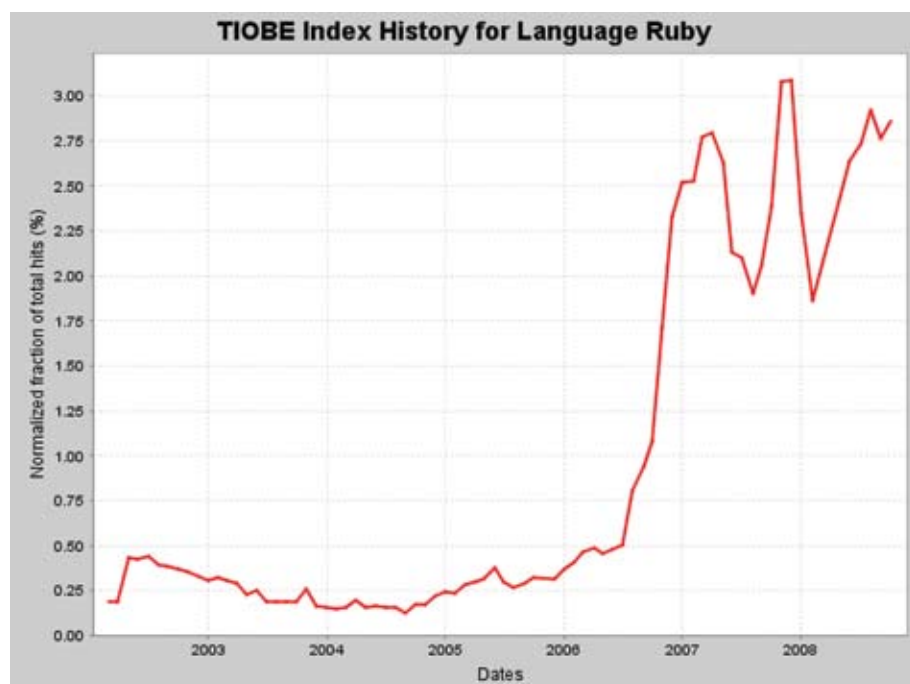
Rails lyhyesti

- Pohjautuu MVC-arkkitehtuuriin
- Tarkoitettu lähinnä WWW-palvelujen rakentamiseen
- Mm. sisäänrakennettu AJAX- ja XML-tuki
- Tietokantatuki mm. MySQL-, PostgreSQL-, SQLite-, Oracle- ja DB2-kannoille

Ruby on Railsin asennukseen löytyy helppo ohjeistus osoitteesta:
www.rubyonrails.org/down

Hyviä kirjoja Railsin opetteluun:
Agile Web Development with Rails (Thomas, Hansson, Breedt ja Clark)
The Rails Way (Fernandez)
Sekä kotimaisiin voimin tehty Beginning Ruby on Rails E-Commerce: From Novice to Professional (Hellsten ja Laine)

Kuva 1.





Risto Nevalainen,
vanhempi neuvonantaja,
FISMA ry.
Risto Nevalainen on toiminut pitkään systeemityön menetelmien ja laadun parissa. Hän toimi Sytyke ry:n puheenjohtajana 1980-luvun lopulla.

Ketterästi tehtyä ja laadukasta – onnistuuko?

Nykyään hypetetään paljon ketterästä kehityksestä. Se sopii jotenkin luontaisesti ohjelmistokehittäjän ajatteluun, ja kokemukseni mukaan koetaan melkein poikkeuksetta positiiviseksi. Mutta ei sekään ole tae projektin ja toimituksen onnistumisesta ja tuloksen laadusta. Ketteryyden tuo mieleeni ajat, jolloin seinätekniikka koki läpimurron ja siihenkin kohdistui melkein päällemaallisia odotuksia. Paljon niissä onkin samaa – mm. tiimityö, nopearytmyisyys ja nopeasti jotakin valmista. Nykyinen ohjelmistotekniikka ja hyvät välineet mahdollistavat sen, mikä jäi parikymmentä vuotta sitten vielä haaveeksi. Nyt on ilo nähdä ohjelmistotuotannon ottavan ison askeleen kohti ammattimaista tekemistä ja korkeaa ennustettavuutta tekemisen ja tuloksen osalta.

1. Johdantoa

Kun näin Sytykkeen jäsenviestin tämän lehden teemasta ”hypetystä tervejärkisesti”, se kolahti minuun heti. Mieleeni tuli pikimmiten ketterä kehittäminen ja siihen kohdistunut hypetys – joka on mielestäni valtaosiltaan ansaittua. Siksi päätin tehdä tämän (puolustus)kirjoituksen, muistellen samalla että vastaavanlaista hypetystä on ollut jo tietotekniikan alkuajoista saakka. Olen nähnyt useita satoja eri tavoin hallittuja projekteja arvioijaurani aikana, niistä parikymmentä ketterän kehityksen periaatteiden mukaisesti. Ja aina olen nähnyt saman: porukka on tavattoman innostunut eikä koe minkäänlaista tarvetta palata menneeseen. Jotakin erinomaisuutta siinä siis täytyy olla.

Käsittelem ketterän kehittämisen hypetystä kolmessa asiassa: 1) tuottaako ketterä kehittäminen parempaa laatua kuin esim. V-malliin perustuva kehitystapa? 2) ovatko ohjelmistotuotantoon kehitetyt kyvykkyys- ja kypsyyssmallit ristiriidassa vai samansuuntaisia ketterän kehityksen kanssa? ja 3) onko Scrum-tyyppinen projektijohtaminen jotenkin erilaista kuin perinteisempi peräkkäiseen

tekemiseen tai iteraatioihin perustuva malli? Alustava vastaukseni kysymyksiin 1 ja 2 on pääosin myönteinen ja kysymykseen 3 varauksellinen.

Tarkoitan ketteryydellä (agile) tässä artikkelissa teknisen tekemisen ja projektinhallinnan muodostamaa kokonaisuutta, esim. Scrum-tyyppinen projektinhallinta yhdistettynä muutamaan tai useimpiin XP-periaatteisiin. Esimerkiksi käyköön pienehkö kehitysprojekti, jossa on 4 – 5 kahden viikon iteraatiota kiivaimman kehityksen aikana sekä sitä ennen ja jälkeen kuukauden tai parin esisuunnittelu ja järjestelmätestaus, päättyen onnistuneeseen toimitukseen. Oleellisia XP-periaatteita ovat jo määrittelyjen tekeminen rinnalla aloitettu testauksen suunnittelu, jatkuva integrointi kunkin iteraation sisällä ja niiden välillä sekä pari/tiimityöskentely toteutuksessa.

2. Tuottaako ketterä kehitys parempaa laatua?

Olen siis pääasiassa sitä mieltä, että ketterä kehitys tuottaa parempaa laatua kuin muut elinkaarimallit. Tai kyseessä onkin joukko ohjelmistokehityksen periaatteita pikemmin kuin mikään uusi malli. Mielestäni parempi laatu on tärkein yksittäinen ketteryyttä puoltava tekijä. Juu on siinä, että lopputulos saadaan nopeasti näkyväksi ja kriittisimmät osat ovat mukana useissa tekemisen ja testaamisen kierroksissa. Ne ehtivät vakiintua ja täydentyä projektin aikana. Ehkä jotakin muuta vähemmän tärkeää jää tekemättä, mutta se onkin yksi ketteryyden perusajatuksista. ”Customer on your side” periaate tarkoittaa, että tehdään asiakastarpeen mukaisia ratkaisuja ja vakiinnutetaan ne nopeasti. Ja mikäs sen parempi tapa kuin tehdä lopputulosta alusta alkaen, esim. toimiva proto jota täydennetään ja kasvatetaan.

Laatua mitataan usein virheiden määrällä per testausvaihe. Tässä suhteessa ketterä kehitys voi olla huononäköisistä aluksi, koska vajaan tuotteen testauksessa eli toteutuksen alkuvaiheessa löytyy väkisin paljon virheitä. Mutta lopputulos ratkai-

see, eli isoja uusia virhekesauomia ja yllätyksiä ei pitäisi enää tulla loppuvaiheessa. Virheiden määrä voi olla siksikin suuri, kun testaus on näkyvää työtä jo toteutuksen rinnalla.

Sudenkuoppa paremmalle laadulle on määritysten liiallinen epämääräisyys. Tämä voi johtua siitä, että asiat ovat oikeasti avoinna eikä vaatimuksia siis tiedetä taikka sitten asiakastarpeet ovat turhan epämääräisiä ja jatkuvasti muuttuvia. Tämä johtaa monenkertaiseen tekemiseen ja raskaaseen iterointiin, eikä työ tahdo edistyä – päättyen usein pettymykseen ja projektin lopettamiseen. Uusi projekti tehdään sitten jollakin koetellummalla tavalla ja ketteryys muuttuu kirosanaksi. Asiakaskin valittaa, että tämä sitoo liikaa heidän avainhenkilöidensä aikaa eikä tunnu tehokkaalta.

Vaatimusten epämääräisyys on toisaalta peruste käyttää ketteriä menetelmiä. Aloitetaan sieltä, mikä jotenkin tunnetaan ja työn aikana opitaan loput. Perinteisen kiinteähintaisen projektin kannalta tämä on kuitenkin mahdoton tilanne, kun projektin laajuudesta ei ole mitään tietoa.

Koetan havainnollistaa toimittajan ja asiakkaan vuorovaikutusta oheisella kaaviolla (kuva 1), jossa vasemmalla puolella on toimittajan kyvykkyys (kuvassa valittu kolmostaso), ja oikealla asiakkaan vastaava kyvykkyys. Tilanteessa A toimittaja ei kykene vastaamaan asiakkaan koviin ja selkeisiin vaatimuksiin ja on siis juurisyy ongelmille ja huonolle laadulle. Tilanteessa B on hyvä tasapaino, eli ammattimainen toimittaja on hiukan parempi kuin ainutkertaista järjestelmää haluava asiakas. Tilanteessa C taas asiakas on ongelma, kun sen kyvykkyys on heikko ja toiminta siten ennustamaton. Toimittaja on silloin ikäänkuin jojon päässä, koettaen venyä milloin mihinkin.

3. Onko ketterä kehitys kypsyyssmallien mukaista vai niiden vvastaista?

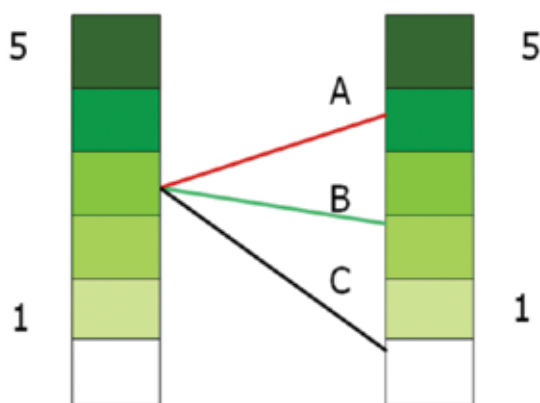
Minulta kysytään usein, onko ketterä kehitys jotenkin uutta ja jopa vastakkaista esim. tunnetun kypsyyssmallin CMMI vaatimuksille. Vastaavia muita malleja ovat esim. SPICE kyvykkyysmalli ja monilla toimialoilla käytössä olevat elinkaarimallit. Vastaan aina, että ei ole vaan ketterä kehitys on hyvinkin kyvykkyys- ja kypsyyssmallien vaatimusten mukaista ja jopa edellytys nousta nopeasti ja tehokkuutta lisäten ylemmille tasoille.

Syynä epämääräiseen, ristiriitaiseen ja mielestäni väärään vastakkainasetteluun on liian puhdasoppinen ja kapea ajattelu puolin ja toisin. Koetan purkaa sitä sekä malleista että ketteryydestä lähtien.

Monet kyvykkyys/kypsyyssmallien soveltajat ja arvioijat ottavat mallien vaatimukset liian kirjai-

mellisesti, eli vaaditaan ne toteutettavaksi sellaisenaan. Mallien tarkoitus on kuitenkin koota hyviä käytäntöjä ja yleisiä vaatimuksia, jotka voidaan toteuttaa monella eri tapaa. Yksi näistä tavoista voi olla mielestäni ketterä kehitys. Otetaan esimerkiksi CMMI-vaatimus "Perform peer reviews", joka saa usein heikon arvosanan suomalaisissa ohjelmistoyrityksissä. Monien mielestä se tuo mieleen vain ulkopuolisen tarkastuksen ja lisäbyrokratian, vaikka katselmointin toteutustapa voisi olla yhtä hyvin projektiryhmässä tehty paritarkastuskin – täydennettynä vaikkapa yksikkötesteillä. Tällainen katselmointitapa taas on hyvinkin ketterien periaatteiden mukaista, pidetäänhän edistymisestä ja teknisistä tuloksista joka-aamuinen kokouskin!

Ketterän kehityksen mukaisesti mielestään elävät toimivat usein perusajatusten vastaisesti tai tulkitsevat niitä liikaa edukseen. Otetaan esimerkiksi idea "documented as needed". Se tarkoittaa, että tehdään vain tarpeelliseksi koettuja kuvauksia, muut jätetään tekemättä. Käytännön tilanne



Kuva 1: Toimittajan ja asiakkaan periaatteelliset asetelmat keskinäisissä kyvykkyystasoissa. Toimittajan oletettu kyvykkyys vasemmalla ja asiakkaan oikealla.

on usein, että dokumentaatio jää tekemättä tai lopuksi harsitaan jotakin asiakasdokumentaatiota muistuttavaa. Tästä saa tietysti huonon arvosanan CMMI-arvioinnissa, ja mallin vaatimus koetaan byrokratiaksi ja turhan työn teettämiseksi. Mutta ei "documented as needed" suinkaan tarkoita, että dokumentteja tehtäisiin vain suunnittelijoita itseään varten ja muista viis. Kyllä niitä tarvitaan myöhemminkin mm. ylläpidon ja jatkokehityksen tarpeisiin. Eli lopputuloksena on jotakuinkin myös CMMI-vaatimuksia vastaava dokumentointitapa eli en näe tässä mitään ristiriitaa.

Toki monet standardoidut mallit ovat aika raskaita etenkin projektinhallinnan puolella, eivätkä ketterän kehityksen noudattajat pidä niistä. Mutta vika onkin näissä vakioituissa toimintatavoissa, joissa ei ole otettu tarpeeksi huomioon ketterän kehityksen tarpeita. Ei kahden viikon sprinttiä kannata suunnitella täysimääräisen projektin lailla. Mutta ei se tarkoita, ettei suunniteltaisi lainkaan. Ketterän kehityksen tyypillinen toteutus on pitää kunkin sprintin alussa työpaja, jossa kootaan edellisen kauden tulokset ja suunnitellaan

seuraavan kauden (esim. kahden viikon) työt. Itse asiassa ne suunnitellaan usein hyvinkin tarkasti, ja tekijäporukka niihin sitouttaen.

Olen nähnyt suuren määrän ketterän kehityksen mukaisia projekteja, joissa kokonaisuuden suunnittelu on jäänyt vähälle, kun luotetaan pala palalta tekemiseen. Tämä on yksi sudenkuopista, johtaen useimmiten paljon suurempaan työmäärään ja sprint-askeleisiin kuin oli alunperin aavistukseen. Eli ei ketterä kehitys poista projektinhallinnan ja "top down" näkemyksen tarvetta, päinvastoin korostaa sitä jotta kukin yksittäinen askel menee oikeaan suuntaan.

Mallien vaatima mittarointi toteutuu mielestäni erinomaisesti ketterässä kehityksessä. Työmaata seurataan sekä backlog tilanteen että burn-down asteen avulla, sekä tehdään kehityssyklien välissä katselmointeja (niitä kutsutaan usein retrospective review – nimellä). Tämä jos mikä on esimerkiksi CMMI:n vaatimusten mukaista ja on itse asiassa ponnahduslauta tasojen 4 ja 5 vaatimaan mittarointiin. Onkin itse asiassa niin, että ketterän kehityksen mukainen toiminta saa kokonaisuutena parempia tasolukuja prosessiarvioinneissa kuin perinteiset tavat. Tähän vaikuttaa itse systematiikan lisäksi hyvät välineet, joiden avulla tekemistä tuetaan tai automatisoidaan.

4. Onko Scrum parempi tapa hallita ohjelmistoprojekteja?

Ketterän kehityksen projektinhallintana on useimmiten Scrum eli hyvin mietitty jaksottainen kehityskausi (usein käytetään nimitystä sprint), jonka aikana panostetaan tiimityöhön, viestintään, näkyvään suunnitteluun ja tiimikeskeiseen kehittämiseen. Kukin sprint sisältää enemmän tai vähemmän sekä määrittelyä, toteutusta, testausta että toimittamista. Tietysti ihan projektin alussa painottuu määrittely, lopussa enemmän testaus, virheenkorjaus ja lopulta täyden toimituksen tekeminen.

No, eihän tuossa kuulosta olevan mitään uutta. Eikä olekaan! Vanha kunnon hyvän suunnittelun periaate toteutuu, ja useimmat meistä näkevät tekemisessä jopa peräkkäisiä vesiputouksen tapaisia vaiheita, nyt vain useampaan kertaan tehtynä. Tältä osin vanhat opit siis toistavat itseään. Jo seinätekniikan kultakaudella 1980-luvulla näimme vastaavia elinkaaria kukkuloineen kuin nyt RUP-mallin kalvoissa.

Hyvänä puolena scrum-menettelyssä pitäisin edistymisen hyvää tietämistä. Hidastempoisemmassa projektissa ei paljastu heti, jos jotakin vaatimusta ei osata toteuttaa. Sitä pallotellaan, ja uskotellaan edistymistä tapahtuvan. Karu piirre scrumissa on, että myös osaamattomuus paljastuu nopeasti eli jotkut meistä kokevat putoavansa kärryiltä. Onhan se toisaalta hyväkin että vastuita

joudutaan miettimään nopeasti uusiksi, kunhan se vain tehdään harkitusti ja muitakin kuin teknisen tekemisen taitoja kunnioittaen.

Koko ketterän kehityksen pahimpana sudenkuoppana pitäisin sen suurinta ansiotakin eli nopeatempoisuutta. Koko ajan on joku takaraja lähellä saavuttaa tuloksia. Se panee yrittämään sekä yksilön että tiimin tasolla, onhan yhdessä onnistumisen kokemus hieno asia. Mutta tähänkin voi turtua ja väsyä.

Pari vuotta sitten pidimme FiSMAn piirissä työpajan projektinjohtamisen ja ohjelmistokehityksen kokemuksista. Emme hyväksyneet sinne muita kuin todistuskelpoisia, eli tilastoihin ja analyysihin perustuvia tuloksia. Pääpuhujamme Khaled-El Emam näytti käyriä, joiden mukaan noin kolmen kuukauden iteraatio näyttäisi olevan kokonaisuuden kannalta paras ja pitkän päälle tehokkain tapa. Tämä on aika kaukana siitä viikon tai kahden rumbasta, jota puhdaslinjaisimmat ketterän kehityksen yritykset käyttävät. Mutta tästä näyttäisi seuraavan burn out ennemmin taikka myöhemmin. Tämä ilmiö ei ole vielä tullut vahvasti esiin suomalaisessa työ kulttuurissa, jossa on ollut tilaa omalle vastuunotolle ja räätälöidyille tehtäväkuville.

5. Lopuksi

Olen koettanut yllä karsia ketteryyden hypeystä rajatulta osin ja omiin kokemuksiini perustuen. Kaikista pienistä epäkohdista huolimatta näen sen suurena kehitysaskeleena ohjelmistotuotannossa, ja jopa ikaikaisten ohjelmoinnin ihanteiden toteutumisenä. Muutama vuosi sitten kuuntelin Scott Amblerin palopuhetta ketterän kehityksen ylivoimaisuudesta. Se kuulosti minun korviini aivan siltä, kuin olemme aina haaveilleet tekevämme ohjelmistoja ja systeemyötä. Itse asiassa se kuulosti siltä, miten olen nähnyt systeemyötä tehtävän suomalaisissa yrityksissä jo pitemmän aikaa. Teknologia ja välineet eivät ole vain antaneet periksi aiemmin eivätkä ole olleet tarpeeksi joustavia. Nyt ovat, eli tämä hype on toteutumassa!

"Koko ketterän kehityksen pahimpana sudenkuoppana pitäisin sen suurinta ansiotakin eli nopeatempoisuutta."



Eija Kalliala

ihmemaa nielee
harhailijan, jokainen
tie vie jonnekin

kirkas tavoite
selättää hypen, johtaa
oppijan maaliin

Sosiaalista mediaa tervejärkisesti opetukseen

Sosiaalinen media luikertelee hitaasti korkeakoulujen opetukseen suljettujen oppimisalustojen ja tietoturvatujen palvelinten rinnalle. Skype- ja mese-osoitteet viuhuvat ryhmätyön alkumetreillä ja oppimisalustoille tallentuvat kokousmuistiot välittävät vain kalpean kuvan ryhmän intensiivisistä palavereista ideamyrskyineen. Oppimisalusta voi tarjota wikin, mutta miksi wikitellä suljetussa ympäristössä, jonne innovaatiot hautautuvat kurssin päätyttyä, kun avoimissa ympäristöissä yhteiskirjoitetaan asiantuntijaverkoston kanssa ja luovan työn tuotokset linkitetään omaan CV:hen LinkedInissä?

Jotkut opettajat ovat vieneet kursejaan Habboon, Facebookiin, Second Lifeen tai Ningiin olettaen, että sosiaalinen media synnyttää sosiaalisuutta ja että kaikki opiskelijat ovat jo siellä. Seurauksena voi olla oppimisympäristö, jossa opettaja huhuilee yksin tyhjässä keskusteluavaruudessa ja joutuu arvailemaan lempinimien taakse kätkeyvien passiivisten olentojen identiteettiä reaali maailmassa.

Ketterinkin tietojärjestelmä tai verkkopalvelu tarvitsee kivijalakseeseen kunnollisen määrityksen, samoin verkko-oppimisympäristö ja -opiskelu. Ennen sukellusta kaninkolosta teknologisen hypen ihmemaahan opettajan pitäisi selvittää itselleen - ja mielellään myös muille - mikä on opiskelun tavoite ja kohderyhmä, mitä halutaan oppia ja missä opittua soveltaa? Liian usein näkee, miten uuden teknologian alttarille uhrataan oppimistavoitteet ja sisälöt, joskus opiskelijatkin.

Sosiaalinen media tarjoaa uudenlaisia mahdollisuuksia vuorovaikutukseen, yhteisöllisyyteen ja toimivien verkkopalvelujen koostamiseen. Niiden käyttöön otosta voi syntyä menestystarinoita ja varoittavia esimerkkejä.

Kollegani on vuosia innostanut opiskelijoita laatimaan laadukkaita seminaariesitelmää asiantuntijateksteiksi Wikipediaan. Toinen kollegani kerää oppiaineensa linkkejä sosiaalisiin kirjanmerkkeihin oppimisalustalle vinkiksi opiskelijoille. Korkeakoulujen TieVie-verkko-opetuksen asiantuntijakoulutuksessa työstimme ryhmätyötä Google Docissa ja kaivoimme dokumenttimme oppimisalustalta oppimisportfolioksi blogiin tai omalle verkkosivustolle näkymään vielä kurssin jälkeenkin.

Opiskelijani ovat vuosikausia projektityönään laatineet verkkopalveluita todellisille yrityksille - vaihtelevalla menestyksellä. Viime keväänä eräs opiskelijaryhmä valitsi perinteisten toteutusvälineiden sijaan sosiaalisen median työkalut. Näyttävä ja toimiva verkkopalvelu valmistui ajoissa ja tyytyväinen tilaaja koki opiskelijatyön ylittäneen odotuksensa. Toinen opiskelijaryhmä taas käytti perinteisiä työvälineitä ja turhautui taistelllessaan verkkopalvelua testausympäristöön oppilaitoksen tietoturvaliselle palvelimelle, johon toimeksiantaja ei päässyt palvelua testaamaan.

Viime aikoina yhä useammat kurssit ovat alkaneet kukkia erilaisia sosiaalisen median palveluja. Teknologian integrointi on vaivatonta: sosiaalisen median kirjon

Skypestä, opiskelijablogeista ja wikeistä Delicious-linkkeihin ja Google-kalentereihin voi linkittää perinteiselle oppimisalustalle vinkkeineen ja ohjeineen. Jos projektitoita tehdään oikeille yrityksille, niin toimeksiantajien edustajien liittymisen Ning-yhteisöön Ning-tunnuksin voi olla vaivattomampaa kuin pääsy oppilaitoksen suljetulle oppimisalustalle. Jos työskennellään Second Lifessa toimivan yrityksen kanssa, Second Life voisi luontua oppimisympäristöksi. Jos kaikki opiskelijat ovat Facebookissa, opiskelun voisi viedä sinne.

Oppimisympäristön voi kitkatta loihtia Ningiin ja saada sinne joukon opiskelijoita ja yrityskumppaneita. Mutta jos kukaan ei suunnittele, mitä Ning-ympäristössä tehdään, niin jääkö virtuaalivaruuteen vain seminaarimateriaalin ja sammuneen keskustelun muistomerkki? Entä ehtivätkö kiireiset aikuisopiskelijat paneutua sosiaalisen median aarreaitaan ja päivittää kaikkia kaikkialle näkyviä virtuaalisen olemassaolonsa dokumentteja?

Sosiaalinen media ei synnytä keskustelua ilman keskusteluaihetta ja -tarvetta eivätkä blogitekstit kommentteineen tai yhteisöllisyys synny käskyttämällä. Virtuaalikalenterit eivät muistuta tapahtumista, jos kukaan ei niitä katsele eivätkä wikit synnytä yhteisöllistä kirjoittamista, jos ryhmän jäsenet eksyvät eri wikeihin.

Onhan opettajilla edelleen selkeänä mielessään opetuksen tavoite, kohde-ryhmä ja sisältö?

Mielikuvittele – ideoista innovaatioiksi!



Tarja Raussi on laivase-minaarioimikunnan jäsen ja Sytyke ry:n hallituksen varajäsen vuonna 2008. Tarja toimii systeemityön asiantuntijana Tieturi Oy:ssä kouluttaen tutoroiden ja konsulttien systeemityön eri osa-alueita, erityisesti prosessien mallintamista ja vaatimusmäärittelyä.

Sytyke oli taas hypettämässä tervejärkisesti, sillä mediaa seuranneet ovat varmaan havainneet, että innovaatio on ollut tämän syksyn trenditeema. Joukko innovaatiosta kiinnostuneita sytykeläisiä kokoontui 10. - 12.9. Viking Gabriellalle. Mikä mukavinta, joukossa oli uusia kasvoja!

Katajanokan terminaalissa sytykeläiset saivat käsiinsä seminaarin ensimmäisen innovaation: kirjekuori, jossa oli kumma naru. Innovaation oli tehnyt Ilkan tytär Sesilia. Sitäpä olikin hauska seurata, kuinka moni oivalsi keksinnön käytön. Katso kuvaa ja päättele itse – vastauksen löydät toisaalta tästä lehdestä.

Ilkka Pirttimaa avasi seminaarin ja luovutti puheenvuoron Sytyke ry:n puheenjohtaja Mitro Kiviselle, joka pohti lyhyesti, mitä Sytyke ja systeemityö oikein on. Mieleeni jäi virke: "meistä jokainen on Sytyke-kokemuspankin kokemustilejä, joista jokainen sytykeläinen saa nostaa osaamista ja jota voimme kartuttaa muiden sytykeläisten kokemuksella". Juuri siksihän me haluamme kokoontua: jakamaan kokemuksiamme, oppimaan toisiltamme, verkostoitumaan ja viihtymään yhdessä.

Edison keksi hehkulampun – humpuukia!

Pentti Juhala Suomen Keksijäin Keskusliitosta kertoi, kuinka maailmalla haaskataan vuosittain tuotekehitysvaroja miljoonia euroja, kun pyörää keksitään uudelleen. Toisin sanoen, ei osata käyttää julkista patentti-informaatiota. Jos on fiksua, voi saada nimensä historiankirjoihin ostamalla patenteja. Näin teki Edison, joka osti kilpailijoiden patentit hehkulampulle ja kehitti sen tuotteeksi! Eri maissa on kuitenkin eroja siinä, mitä voi patentoida. Esimerkkinä tulkinta, jonka mukaan patentoitavaksi kelpaa vain tuote, joka on keksinnöllinen, uusi ja teollisesti hyödynnettävä. Siis siitä pitää syntyä jotain käsin kosketeltavaa - idea tai menetelmä ei kelpaa patentoitavaksi. Esitys herätti runsaasti keskustelua ja kysymyksiä.

Lauri Laitinen Nokiasta kertoi, kuinka hänen illuusionsa patenteista ovat karisseet vuosien mitaan. Hän luuli, että patentissa on aina kyse suurista hienoista keksinnöistä, mutta saikin todeta, että suurin osa on ihan normaalia insinööritöitä.

Suurin osa patentoinnista on itse asiassa hakeusten tekemistä, lainopillista puurtamista jne. Patentointia voidaan "harrastaa" ihan vain kilpailijoiden kiusaksi, sillä patentti estää toisia hyödyntämästä keksintöä. Kaiken lisäksi patentti pitää useimmiten hakea joka maahan erikseen! Kalista ja aikaa vievää touhua. Laurilla itsellään on kymmenittäin patenteja, joten hän tosiaan tiesi, mistä puhui.

Voiko tieteellisestä tutkimuksesta tulla innovaatioita?

Lyhyen kahvitauon jälkeen Petri Myllymäki Helsingin Yliopistosta kertoi, että puhtaasti tieteellinen huippututkimus saattaa synnyttää kaupallisesti hyödyllisiä innovaatioita. Mutta kuinka tämä saataisiin aina toteutumaan ja siitä vielä rahaa, niin se onkin arvoitus! Complex System Computation (CoSCo)-tutkimusryhmä tutkii mm. koneoppimista ja data-analyysiä. Tämä ryhmä sai tehtäväkseen laatia algoritmin kolmiulotteiseen pakkausongelmaan: miten erimuotoiset, eripainoiset jättimäiset paperirullat pakataan optimaalisesti rekan konttiin. Pitkän kehittelyn tuloksena syntyi yhdessä TietoEnatorin kanssa Koptimi-ohjelmisto, joka on ollut tuotantokäytössä StoraEnsolla vuodesta 1999. Toisentyypinen innovaatio on uudenlainen hakukone AINO, joka hakee netistä tekstidokumentteja niiden sisällöstä tehdyn tilastollisen analyysin pohjalta.

Uudenlainen näkökulma työn "hankaluuteen" tuli siitä, että Petri kertoi huippututkijan tyypillisesti pitkästyvän, jos homma tai ongelma on liian helppo. Eli jos tuntuu siltä, että toinen osapuoli ei innostu, tee ongelmasta vieläkin haastavampi - silloin hän saattaa kiinnostua!

Kuviteltua vai faktaa?

Mikko Holmberg ja Perttu Karvinen CodeBakers Oy:stä kertoivat, millaista apua voi välineistä olla yhteistyön tekemisessä. Tyypillinen ongelma projektissa on, kuinka paljon on valmista ja paljonko on kesken. Yleensä esitetään mutu-arvioita eli 'musta tuntuu'. "Joo, koodi on lähes valmis!" - lue: homma on aloitettu. Vaan kuinka saadaan faktaan perustuvaa tietoa projektin tilanteesta?

Yhteistyötä tehdään nykyään lähinnä sähköpostin, palaverien, wikin tms. avulla. Mutta riittääkö se? Tietävätkö kaikki, mitä seuraavaksi



Kirjekuori-innovaatio; mikä on naran virka? (vastaus alla)

Vastaus: kirjettä on kirjepöytä.

pitäisi tehdä? Mistä löytyy johonkin tuotokseen liittyvät keskustelut, dokumentaatiot jne.? Application Lifecycle Management (ALM) lupaa tukea

- erilaisia viestintätapoja,
- ohjata erilaisissa rooleissa toimivien käyttäjien toimintaa prosesseilla,
- tuotosten välistä jäljitettävyyttä,
- hjelmien välistä integroitavuutta.
- ALM-ohjelmistoja on monella eri toimitajalla.

Mikko ja Perttu esittelivät RTC-pilottiprojektin tuloksia. Pilottiprojektissa käytettiin IBM Rationalin Team Concert -tuotetta ja siinä oli mukana 6 yritystä ja 12 henkeä. Pilotin aikana pidettiin 3 työpajaa ja 2 online-sessiota. Kaiken kaikkiaan kokemukset olivat positiivisia, joskin haasteitakin tuli vastaan. Saimme nähdä myös demon työkalusta - vaikutti ihan mielenkiintoiselta.

Yubikey – autentikointi napinpainalluksella

Saimme päivän päätteeksi esimakua seuraavan aamun yritysvierailijan aiheesta, kun Ilkka Pirttimaa esitteli lyhyesti varsin näppärän innovaation, Yubikeyn. Yubikey on USB-näppis, joka tuottaa yhdellä napin painalluksella salatun "blaablaan" tunnistamista varten. Oikealla salausavaimella pystyy tunnistamaan "blaablaasta", mikä Yubikey on kyseessä, tuottiko se avaimeen liitetyn salaisen sisällön. Lisäksi ns. toistohyökkäys on estetty siten, että koodin lisäksi varmennepalvelu tuottaa joka kerran kasvavan numerokoodin eli tiettyä avainta voi käyttää vain kerran.

Ilta jatkuikin sitten buffet-illallisella, jossa jokainen sai innovoida, miten nauttia runsaista antimista ja kuitenkin välttää ähkyltä. Porukka viihtyi pöydissä keskustelemassa aikaisempien vuosien tapaan niin, että meidät lopulta heitettiin ulos ravintolasta. Kas kun on kerrankin paikka, jossa voi istua ja jutella liian taustamelun häiritsemättä!

Aamukymmeneltä Tukholmassa kokoonnuimme jälleen seminaaritiloihin kuulemaan, miten Yubikey oikein syntyi. Simon Josefsson Yubico Ab:stä tuli kertomaan heistä ja tuosta innovaatiosta. Firma on ollut olemassa puolitoista vuotta ja sillä on 9 henkeä töissä Tukholmassa ja Kaliforniassa. Firma perustettiin innovaation ympärille. Yubikey on mekaanisesti todella yksinkertainen, mutta tuottaa 128 bittisen kertakäyttöisen autentikointikoodin. Se käy kaikkiin tietokoneisiin - käyttöjärjestelmäriippumaton, eikä maksa paljon mitään. Sen käyttäminen ja integrointi sovelluksiin perustuu täysin avoimeen lähdekoodiin. Mitään mittavia ja kalliita taustaohjelmistoja ei tarvita. Simon kertoi, että eräs asiakas oli päättänyt käyttää Yubikeyä etuoven aukaisemiseen sähköisesti. Ilkka kommentoi, että hänkin on ajatellut sitä.

Tehdäänkös me vai valitetaankos vaan?

Sitten olikin aika lähteä jaloittelemaan Tukholmaan ennen iltapäivän sessiota. Tosin kirjoittajan aika kului tällä kertaa töiden merkeissä - piti hoitaa parit puhelinneuvottelut.

Tauon jälkeen ihmiset saapuivat virkistyneinä kuulemaan, miten Mitro Kivinen Alter Chief Oy:stä saa asioita tapahtumaan, oli sitten kyseessä prosessin parannus, tekninen ongelma tai kahvi-automaatin hankinta. Mitrolla oli loistavasti kiteytettynä täysin käytännönläheiset aktiviteetit aina ongelman määrittelystä sen ratkaisemiseen. Ja ennen kaikkea: muistetaan kiittää lopuksi kaikkia mukana olleita!

Voiko toimittajan ja tilaajan yhteistyö olla avointa?

Sami Kettunen Samcom Oy:stä ja Kari Aho Elisa Oyj:stä kertoivat, miten yhteistyö voi toimia avoimessa hengessä. Sami Kettunen totesi, että jos henkilökemiat eivät toimi projektissa, on syytä vaihtaa ihmisiä. Muuten toiminta ei ole tehokasta ja toimivaa. Samcomilla on käytössä tavoitehintamalli: Jos arvioitu työmäärä alitetaan, saadaan pieni bonuskorvaus. Jos arvioitu työmäärä ylitetään, Samcom veloittaa ylimenevistä työtunneista vain 40%. Kuitenkin tuntikatto rajaa asiakkaan riskin - sen ylittyessä toimittaja vastaa täysin kustannuksista. Tämän toimiminen nojaa täysin osaavaan henkilöstöön. Osaamisen lisäksi panostetaan henkilöstön viihtyvyyteen - näin saadaan henkilöstön vaihtuvuus minimoitua. Yhteistyössä käytetään monenlaisia työvälineitä apuna.

Kari Aho kertoi puolestaan, miten he ovat kokeneet yhteistyön: tietotaidon päälle on helppo rakentaa. Lisäksi tekijöiltä voi vaatia enemmän ja sanoa suorempaan, kun asioita tehdään aidosti yhdessä. Elisassa on käytössä sekä ITIL että ASL IT:n hallintaan ja sovellusten elinkaaren hallintaan. Hän esitteli todellisen esimerkin, jossa erään monitoimittajaympäristössä toteutetun Navitas-palvelun ylläpito ja kehittäminen on pääosin ulkoistettu Samcomille. Liiketoimintapuolesta vastuu on kuitenkin aina Elisalla.

Jotta yhteistyöstä voisi tulla kumppanuus, toimittajalta vaaditaan:

- Koodin on toimittava!
- Dokumentaation sovellushallinnasta on oltava relevanttia ja ajan tasalla
- Kokemusta ja näyttöjä osaamisesta
- Koodaajat ja arkkitehdit samaan pöytään asiakkaan kanssa
- Henkilöt osaavat käyttäytyä

Kumppanuuteen on päästy, sillä nämä toimivat. Osaaminen on avaintekijä, mutta luottamus ratkaisee!

Sponsoreina olivat:



Liimaa liiketoiminnan ja IT:n väliin?

Ilkka Pirttimaa ja Antti Everi Stockmann Oyj:stä kertoivat, miten he ovat lähteneet kehittämään yritykselle kokonaisarkkitehtuuria. Stockmann on globaali konserni: 8 maata, 600 myymälää, 18 000 työntekijää, ja siten sillä on globaalin organisaation haasteet. Alun perin jokaisessa yrityksessä on ollut omat työkalut, sovellukset, kuvaustavat jne. Vähitellen on luotu yritysarkkitehtuuri, jossa on kaksi päänäkökulmaa: liiketoiminta-arkkitehtuuri ja tekninen arkkitehtuuri. Kun käytännössä tulee erilaisia muutostarpeita, niin yritysarkkitehtuuri antaa puitteet ja ohjeistaa, miten investoidaan ja priorisoidaan hankkeita. Toisaalta käytännön prosessit, järjestelmät ja infrastruktuuri tuottavat palautetta, joka vaikuttaa yritysarkkitehtuuriin.

Stockmann on valinnut IDS-Scheerin ARIS-tuotepereheen kokonaisarkkitehtuurin luontiin. Välineistön käyttö on tuonut helpotusta, mutta toisaalta kuvaaminen pitää tehdä "oikein", jotta välineestä saa hyötyjä irti. Esimerkiksi aiemmin sovellusten elinkaarta "hallittiin" excel-taulukoiden avulla, nykyään saadaan suoraan raportit välineestä. Välineistön käyttöönotosta on nyt kulunut 9 kuukautta ja mallintajia on useista liiketoimintaryhmistä. Projektin sijaan päätettiin luoda ns. mallinnuskerho, jossa työestetään malleja ja ratkotaan yhdessä ongelmia. Näin kukaan ei jää yksin opettelemaan, vaan saadaan yhteistyöstä lisää voimaa. Haasteena on edelleen perinteinen: kuinka saada ihmiset dokumentoimaan enemmän? Tulevaisuuden näkymistä yksi on mallinnuksen tuominen dokumentaation lähteeksi.

Voiko tietovarastointi olla ketterää?

Tomas Stenlund Ineosta kertoi uudesta Data Vault -tietovarastointitavasta. Hän aloitti raflaavasti toteamalla, ettei relaatiomallia ole koskaan tarkoitettu tietovarastointiin. Data Vault -tietovarastoinnin peruselementit ovat Hub (ydinelementti), Satelliitti (liittyy hubiin tai linkkiin ja se

kuvaakin usein ajan suhteen eri tavoin käyttäytymistä) ja Linkki (yhdistää elementtejä). Jos malli muuttuu liiketoiminnan muuttumisen takia, edellinen versio mallista menee automaattisesti historiatietoihin. Näin liiketoiminnan muuttumisen vaikutusta tietovarastomalliin voidaan seurata - ja tarvittaessa palauttaa aiempi versio.

Peruselementteihin lisätään lisäksi metatietoa esim. niiden käsittelystä (audit trail). Data Vault sisältää myös liiketoimintasääntöjä (business rules). Dataan voidaan liittää tietoa myös sen laadusta. Kirjoittajan kommentti: Data Vault yhdistää siis aidon oliomallin ja metatiedon. Perusrakenne muistuttaa paljon oliotietokantojen rakennetta.

Koska malli pitää sisällään sekä metatietoa, historiaversiointin ja tietoa laadusta, se soveltuu hyvin iteratiiviseen ja ketterään kehittämiseen, jossa tietovarastoakin pitäisi kehittää iteroiden. Alussa ei siis ole "valmista" mallia, vaan se rakentuu vähitellen.

Lopussa kiitos seisoo

Ilkka veti yhteen seminaarin annin ja antoi sitten puheenvuoron sponsoreille, jotka mahdollistivat jälleen kerran seminaarin. Sitten siirryimmekin nauttimaan á la carte -illallisesta ja toistemme innovoivasta seurasta. Perinteisen tietokilpailun lisäksi jokaisen piti miettiä ja laskea, kuinka monen uuden ihmisen kanssa tuli keskustelleeksi. Allekirjoittanut jääväsi itsensä lipunjakajan ominaisuudessa - ei sitä oikein voi kunnon keskusteluksi mainita, kun ojentaa liput ja muut materiaalit :) Eniten uusia kontakteja oli kerännyt päätoimittajamme Minna Oksanen, joka sitten vielä lisäsi saldoaan kiertämässä jakamassa palkintonamuja kaikille ja samalla kyselemässä ideoita lehteen.

Kiitokset kaikille osallistuneille mukavasta seminaarista!

Jokaisen piti miettiä ja laskea, kuinka monen uuden ihmisen kanssa tuli keskustelleeksi.





Teksti: Markku Niemi

Mukavia kokemuksia SYTYKE-aktiiviudesta

On taas syssy eli se aika vuodesta kun houkutellaan uusia aktiiveja mukaan SYTYKEN ja sen osaamisyhteisöjen toimintaan. Lupasin kirjoittaa muutaman rohkaisevan rivin sel- laista harkitseville :-).

Johtokunnan jäsenenä ja myöhemmin osaa- misyhteisöaktiivina on minun mielestä ollut hyvä seurata alan menoa ja trendejä. Järjestämällä aktiviteetteja kuten esitelmä- ja puhetilaisuuksia ja tekemällä Systeemityö-lehteä oppii väkisin itse substanssistakin. Siinä sivussa tulee kokemusta myös työelämässäkin hyödyllisistä käytännön taidoista. SYTYKEN laivaseminaarin järkäily oli itsel- leni kokemus, jonka nojalla uskalsin päivätöissäkin sellaisen tarjoilla eräälle kansainväliselle jengille ja ihan hyvinhän se meni.

Eihän se puhujien ja kirjoittajien, sponsoreiden ym. metsästäminen, yhdistyksen lakisääteisten velvoitteiden hoitaminen ja muu toiminnan jär- jestäminen toki aina herkkua ole, mutta SYTYKEN tapaisessa hyvähenkisessä porukassa kommel- luksetkin naurattavat. Meno ei ole niin kuoleman vakavaa kuin päivätöissä, joten uusiakin juttuja tulee kokeiltua herkemmin yhdistyksen sallivassa ilmapiirissä.

Jos meno ei yhdistyksen vakielimissä silti miel- lytä, voi SYTYKEN siipien suojiin perustaa oman

aihepiirin mukaisen kerhon eli osaamisyhteisön. Siihen yhdistys suorastaan kannustaa ja tarjoaa monenlaista, rahallistakin, tukea.

Puhdasta viihdettäkin on ollut mukavasti tar- jolla etenkin laivaseminaareissa ja johtokunnan suunnitteluseminaareissa. Hyvässä porukassa on hauskaa melkein missä tahansa ja ruotsinlaivojen meinigin kaikki tietävätkin. Suunnitteluseminaa- reista mieleenpainuvuvin oli ehkä urheiluhenkisen Vierumäen viikonloppu, joka aloitti johtokunnan kevät/syysuunnitteluseminaarien perinteen. Puheenjohtajana tuolloin ollut Helena Venäläinen kunnostautui sen toiminnan käynnistäjänä. Eikä hullumpi meno ollut keväisessä Suomenlinnassa- kaan, vaikka retkeilyhousut punaviiniillä tuhriin- tuivatkin Helenan purjeveneeseen keikkeessä... vai mikähän se syy olikaan.

Mutta nyt on siis minulle tullut aika väistyä SYTYKE-aktiviteettien järjestämisestä; aika aikaa kutakin. Toivottavasti yhdistys löytää uusia, innokkaita toimijoita. Vapaaehtoistyö on mukavaa ja opettavaista kun sen oikein oivaltaa. Oma jut- tunsu on sekin, että CV:ssä tai työhaastattelussa antaa vapaaehtoistyöhön aktiivinen osallistumi- nen hyvän kuvan. Tärkeintä on kuitenkin se, että antamalla saa, Tommi Hellstenin paradokseja lai- natakseen.



*Markku Niemi,
ent. SYTYKE ry:n johtokun-
nan jäsen ja ProSY-aktiivi.*



Juha Jääskinen on
MallinnusOSYn vetäjä
ja hallituksen jäsen
2009.

Seminaarit Sytykkeen toiminnassa

Aloitin Sytykkeen MallinnusOSYn vetäjänä alkuvuodesta 2007, jolloin omana motivaationani oli järjestää itseäni ja samalla muitakin kiinnostavia seminaareja; tähän mennessä olen itse järjestänyt 7 seminaaria ja yhden verkkoseminaarin yhteistyössä ja lisäksi ideoinut muita seminaareja.

Aiemmissa käymissäni seminaareissa olen kiinnittänyt huomiota hyviin ja ei niin hyviin seminaareihin ja usein suuri osa tästä mielikuvastani on muodostunut seminaarin järjestelyistä. Siksi halusin kiinnittää erityistä huomiota järjestelypuoleen, ilman sen suurempaa kokemusta järjestämisestä. Huomasin, että kun tekemisen aloittaa heti, niin asioiden tekemiseen ei menekään niin kovin paljon aikaa – järjestäminen kun muodostuu monista pienistä asioista. Seminaarien järjestämisessä onkin hyvänä mallina ketterien menetelmien periaatteet. Yksi tärkeimmistä järjestämisen pienistä asioista on ajoissa aloittaminen. Pieniä asioita ei tarvitse suunnitella raskaasti ja kun aloitat tekemään heti, ei niistä muodostu edes mielessäsi isoja ja stressaavia. Kun järjestelyt on tehty ajoissa, ei tule kiire loppumetreillä. Pari kertaa flunssa on iske-

nyt juuri seminaarin alla, voin vain kuvitella sitä stressin määrää jos nämä pienet järjestämisen asiat olisivat olleet tekemättä. On tainnut sama ongelma viime tipassa tekemisestä ollut ainakin itselläni monesti työssä mukana – onkohan kiire pakollista, vai onko se vain lopulta itse aiheutettua ja jos on niin voisiko olla toisin? Aikaakin ketterään tekemiseen menee vähemmän, eli aika on harvoin syy kiireeseen.

Jos mahdollista, aloitan seminaarin ”järjestämisen” jo puoli vuotta ennen varsinaista seminaaripäivää. Tällöin ideoin seminaarin teeman ja esitykset, varaan aikataulun, paikan/tarjoilut ja tiedotan ajankohdasta kohderyhmälle sekä kysyn sopivia esiintyjiä. Esiintyjät on helpompi löytää ja motivoida, kun kalenterissa on tilaa ja järjestämiselle aikaa – esiintyjät saavat myös nimetä teemaan sopivat esitysten aiheet. Näiden tekemiseen ei kulu aikaa kaikkine tekemisineen hyvin tehtynäkään juuri paria tuntia pidempään. Jos idea seminaarista tulee vasta kuukautta ennen järjestämistä, ei ajankäyttö ole ongelma – suurin ongelma tällöin on sopivien esiintyjien löytäminen juuri halutulle päivälle.



Viimeistään kuukautta ennen seminaaripäivää kysyn esiintyjiltä esitysten edistymisestä ja jos esiintyjä niin haluaa, mielelläni kommentoin esitystä; tärkeintä on muistuttaa ja aktivoida esiintyjät ja samalla herättää uudestaan mielenkiinto kohderyhmässä, jolle lähetän samalla myös uusintakutsun. Tämäkään kokonaisuus ei vie aikaa puolta tuntia enempää kutsua ja kommentointikierrosta kohden ja harvemmin näitä kovin montaa iteraatiota teen. Myös esiintyjille on usein helpompaa, kun joku sparraa esityksen tekemistä.

Viikkoa ennen seminaaripäivää teen saman kyselyn kuin kuukautta ennen, tällä kertaa vähän aktiivisemmin. Kaikissa kyselyissä käytän usein vanhoja sähköpostipohjia, jolloin niidenkään kirjoittaminen ei montaa minuuttia kestä. Esiintyjä pyydän lähettämään esitykset minulle viimeistään seminaaripäivänä, jotta saan nekin jaettua kuulijoille – yleensä jaan esitysmateriaalit vain sähköisesti seminaarin jälkeen. Toki materiaalit voisi myös tulostaa, tähän pitää varata vähän enemmän aikaa. Viimeistään viikkoa ennen teen myös palautekyselyn valmiiksi, jotta voin lähettää sen sähköisesti heti seminaarin jälkeen. Palautekyselyn tekemiseen löytyy erinomaisia helppokäyttöisiä ja edullisia ohjelmistoja. Palautekyselyn tekeminen ensimmäistä kertaa vie aikaa maksimissaan pari tuntia ja jatkossa vanhan peruskyselyn pohjalta muokkaaminen ei kauan kestä. Tulostetut palautekyselyt ovat lisäksi tarpeen, kun jaan kuulijoille palkintoja. Tällöin arvonta on helppo tehdä palautekyselyiden antaneiden kesken. Lisäksi lähetän aina myös sähköisen palautelomakkeen, johon vastaajalla on rauhallisempi vastata.

Kun kaikki edellä mainitut toimenpiteet on tehty, on seminaarin järjestäminen aika helppoa. Omaan seminaarin alustuksen tekemiseen

pitää varata 1-2h, riippuen vaatiiko kokonaisuus erityistä alustusta, vai voiko shown antaa lähes kokonaisuudessaan asiansa tunteville esiintyjille. Itse seminaarissa oma roolini on jaksottaa se pitää huolta tauoista. Erityisen tärkeää on palautekyselyn lähettäminen läsnäolijoille heti seminaarin jälkeen siten, että se odottaa sähköpostissa seuraavana aamuna. Tähän kyselyyn liitän myös seminaariesitykset. Tällä kyselyllä saan tyypillisesti n. 50% vastausprosentin ja kun lähetän saman uusintakyselynä viikon päästä, on kyselyn vastausprosentti vähintään 80%. Tämän pohjalta onkin taas entistä helpompaa järjestää seuraava seminaari, kun tiedän mitä kohderyhmä oikeasti haluaa kuulla!

Seminaarin järjestämisessä ei ole mitään vaikeaa, helpoista asioista tosin voi itse tehdä vaikeita tai stressaavia tai sitten ei. Kiteytetysti seminaarin järjestämisessä tärkeää on tehdä pieniä asioita pitkin matkaa – tee vain se mitä sinulle järjestäjänä kuuluu, älä myöskään lykkää mitään! Yksittäisen toimenpiteen tekemiseen ei mene yleensä puolta tuntia enempää ja jos tähän ei ole aikaa, voi kysyä miksi ylipäättään haluat seminaaria järjestää. Sama pätee kaikkeen muuhunkin tekemiseen ja tärkeintä on ottaa vastuu itse valituista tekemisistä. Seminaarin järjestäminen on lopulta helppoa – on antoisaa jakaa ja saada tietoa ja erityisesti samalla keskustella tauoilla muiden samasta asiasta kiinnostuneiden kesken.

Hyvää syksyn jatkoa ja tervetuloa Sytykkeen asiantunteviin ja hyvin järjestettyihin seminaareihin!

Hyvä tietotekniikan liiton jäsen!

Näin päivität tietosi

Voit päivittää jäsentietosi verkkosivuillamme www.ttlry.fi. Tietojen päivittämiseen tarvitset käyttäjätunnuksen (= jäsennumerosi, merkitty jäsenlehtiin) ja salasanasasi (= postinumerosi). Jos olet muuttanut salasanasasi tai kirjautuminen ei muutoin onnistu, voit lähettää tunnusten tarkistuspyynnön osoitteella jasenasiat@ttlry.fi.

Toivomme sinun erityisesti varmistavan, että sähköpostiosoitteesi jäsentiedoissa on oikea.

Tietotekniikan liitto ry

Lars Sonckin kaari 12	www.ttlry.fi	jasenasiat@ttlry.fi
02600 Espoo	etunimi.sukunimi@ttlry.fi	p. 020 741 9898
		f. 020 741 9889



Henkilökohtaisempaa palvelua - Sinun eduksesi

Tietotekniikan liitto jäsenyhdistyksineen, osaamisyhteisöineen ja kerhoineen haluaa palvella jäseniään henkilökohtaisemmin ja paremmin, tarjota tietoa juuri Sinua kiinnostavista aiheista. Palvelun parantamiseksi olemme uusineet verkkopalvelumme.

Päivität vain tiedot itseäsi kiinnostavista aiheista ja saat tietoa juuri niistä. Voit päivittää valintasi aina halutessasi. Tietoja ei anneta ulkopuolisille tahoille vaan niitä käytetään ainoastaan TTL:n ja sen piirissä toimivien yhteisöjen tarkoituksiin.



Seppo Takanen on Sytyke ry:n varapuheenjohtaja ja hän toimii CodeBakers Oy:ssä vastualueenaan palvelujen ja liiketoiminnan kehittämisen sekä projektien johtaminen.

LinkedIn Sytykkeen toimintaa tehostamassa

Henkilökohtaisten LinkedIn-verkoston rakentaminen on entistä suositumpaa. Tällä hetkellä LinkedIn:iä käyttää 25 miljoonaa ihmistä ja määrä kasvaa 1,2 miljoonalla joka kuukausi. Motiivina verkoston rakentamiselle ja omien tietojen julkaisemiselle on entistä enemmän oman cv:n ja siihen liittyvien tietojen esittäminen sekä omalle verkostolle, että mahdollisille muille kiinnostuneille. Tiedä vaikka oma osaaminen kiinnostaisi jotain niin paljon, että sillä perusteella saisi tarjouksen uudesta työpaikasta. Useille yksi verkosto ei riitä, vaan niitä rakennetaan erilaisiin tarpeisiin lukuisiin tarjolla olevista vaihtoehdoista. Mm. Facebook, MySpace ja Plaxo kilpailevat käyttäjien suosiosta.

Henkilökohtaisten verkoston rakentamisen lisäksi palvelut tarjoavat mahdollisuuksia tilojen tai ryhmien muodostamiseen yhteisten kiinnostusalueiden ympärille. Tätä mahdollisuutta haluamme käyttää nyt myös Sytykkeen toiminnan tehostamisessa. Vaihtoehtoja vertailtaessa LinkedIn tuntuu hyvältä vaihtoehdolta, koska se on nimenomaan ammattilaisten yhteydenpitoon tarkoitettu yhteisö.

LinkedIn tarjoaa Sytykkeen jäsenille hyvän ympäristön, josta näkee keitä muita yhdistykseen kuuluu. Lisäksi tulevat mahdollisuudet yhdistyk-

sen jäsenten väliseen verkottumiseen ja keskustelujen käymiseen yhteisistä aiheista. Ryhmä ei ole täysin avoin kenelle hyvänsä, vaan tarkoitus on, että se toimii pääasiassa Sytykkeen jäsenten välisenä kanavana.

Ryhmään liittyminen

Ryhmään liittyminen edellyttää luonnollisesti LinkedIn-tunnuksen käyttöä. Jos sinulla ei ole vielä omaa tunnusta, niin sellainen on helppo perustaa LinkedIn-palvelussa (<http://www.linkedin.com>). Sinulla ei tarvitse olla yhtään kontaktia liittyäksesi Sytyke-ryhmään. Ryhmän kautta voi sitten käynnistää oman verkoston rakentamisen tai kartuttaa jo olemassa olevaa verkostoa.

Liittyminen käy varsin yksinkertaisesti joko suoraan ryhmän sivulta (<http://www.linkedin.com/groups?gid=137972>) tai "Groups Directory" toiminnon kautta (<http://www.linkedin.com/groups-directory>). Sytyke-ryhmä löytyy hakupalvelun avulla syöttämällä sana "sytyke" hakukenttään. Haun jälkeen ryhmän nimen vierestä löytyy toiminto "Join this group", jota kautta pyyntö välittyy ryhmän koordinaattoreille (tällä hetkellä koordinaattoreina toimivat yhdistyksen puheenjohtaja Mitro Kivinen ja varapuheenjohtaja Seppo Takanen sekä muutama muu aktiivinen yhdistyksen jäsen). Ryhmän koordinaattori hyväksyy sitten pyyntösi.



Omat asetukset

Jotta ryhmän jäsenyydestä saisi parhaan mahdollisen hyödyn, niin omat ryhmään liittyvät asetukset kannattaa tarkastaa ryhmään hyväksymisen jälkeen. Erityisesti tärkeää on "Contact Settings" -osion tiedot.

Ryhmän keskustelualueet

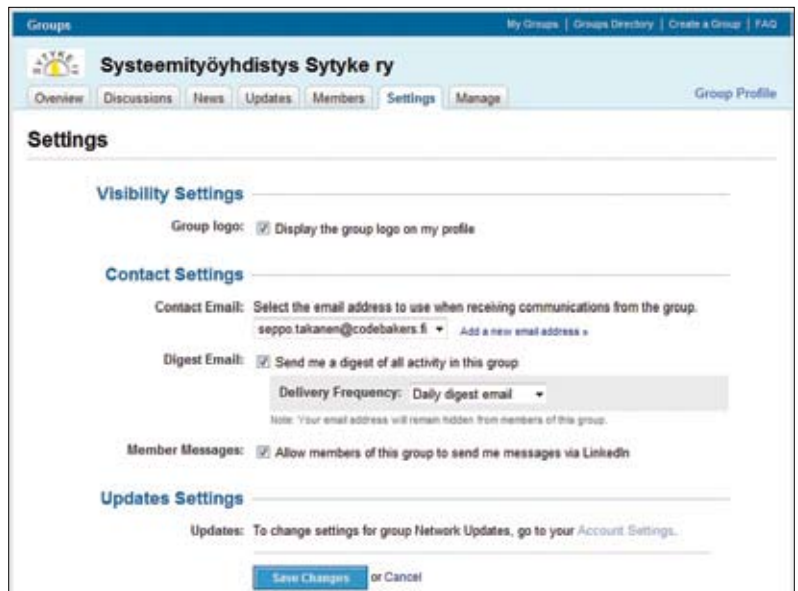
Keskustelualue mahdollistaa keskustelujen käynnistämisen mistä hyvänsä jäsenistöä kiinnostavasta aiheesta. Tärkeitä aiheita ovat esim. vuoden 2009 toiminnan suunnittelu, Syteemityölehden sisällön kehittäminen vuoden 2009 aikana ja yhdistyksen 30-vuotisen taipaleen juhlistaminen vuoden 2009 aikana. Samalla tavalla kaivataan keskustelua OSYjen toiminnan kehittämisestä ja tukemisesta.

OSYt ja LinkedIn

LinkedIn tarjoaa myös mielenkiintoisen mahdollisuuden OSYjen toiminnan kehittämiseen. OSYt voisivat perustaa omia ryhmiä, jolloin OSYn jäsenistön hallinta ja sisäiset keskustelut hoituisivat samaan tapaan kuin Sytyke-ryhmässä. Tästä olisi myös se synergia, että tiedot OSYistä ja niiden toiminnasta saataisiin laajempaan tietoisuuteen kuin tällä hetkellä. OSYja onkin pyydetty perustamaan omat ryhmänsä mahdollisimman pian.

LinkedIn-kehittää palveluitaan jatkuvasti. Tutkimme näitä uusia palveluita ja otamme niitä käyttöön ryhmän puitteissa tarpeen mukaan. LinkedIn ei kuitenkaan ole se ainoa vaihtoehto sosiaalisten verkostojen hyödyntämiseen Sytykeen piirissä. Tarpeen ja kiinnostuksen mukaan muita kanavia voidaan myös avata. Lähettäkää kommentteja ja mielipiteitä tästä Sytyke-ryhmän hyödyntämisestä suoraan ryhmään uusiksi keskustelun avauksiksi. Luonnollisesti kommentit tulevat perille myös sähköpostitse Mitro Kiviselle (mitro.kivinen@alterchief.fi) tai Seppo Takaselle (seppo.takanen@codebakers.fi).

Olkaa aktiivisia!



Korjaus

Lehdessä 1/2008 on Irmeli Sinkkosen tiedot väärin, lehdessä 4/2007 ne ovat oikeat eli;

Irmeli Sinkkonen on Adage Oy:n kehittämisjohtaja. Irmeli on käytettävyyden pioneereja Suomessa. Hän myös kouluttaa käyttäjäkeskeisen

suunnittelun käyttöä verkkoprojekteissa. Ennen käytettävyytutkimukseen siirtymistään hän on toiminut neljännesvuosisadan suunnittelijana ja suunnittelun esimiehenä.

Toimitus pahoittelee virhettä.



Hypetystä tervejärkisesti

Käsittelen tässä artikkelissa hypetystä korkeakulttuurisen tieteellisesti. Hypetys on monivaiheinen ilmiö, joka on jatkuvasti ympärillämme ja jonka johonkin vaiheeseen jokainen ihminen on koko ajan kietoutunut. Käsittelen kutakin vaihetta kahdesta eri näkökulmasta: IT ja PJ. Tarkemmin sanottuna IT:hän tietysti on meille kaikille tuttu informaatioteknologia ja PJ tarkoittaa pikkujoulua.

Ensimmäinen vaihe on kaikkein yleisin, nimittäin ignoranttiuden vaihe. Suurin osa maailman ihmisistä on jatkuvasti lähes kaikkien asioiden suhteen tässä tilassa. On aivan turha kuvitella, että maailmassa eläisi ihmisiä, jotka tietäisivät kaikesta edes vähääkään. Parhaimmatkin tietävät hyvin vähistä asioista aika vähän.

IT: "En minä oo mistään SQL:stä kuullu, onkse taas niitä four-letter-words?"

PJ: "No kuule ei täällä vielä oo näkynyt niitä markkinoinnin uusia mimmejä"

Seuraavaksi ihminen altistuu kohteena olevaan asiaan ensimmäistä kertaa. Tapahtuu tutustumisen:

IT: "Olin kaverin luona, sillä oli Vista"

PJ: "Katos mitä eukkoja tuolta tulee, hör hör"

Kolmantena vaiheena voidaan pitää euforisen kiinnostuksen vaihetta. Tutustumisen kohde on jostain syystä palautunut mieleen, sen toimivuutta on verrattu johonkin nykykäytäntöön ja havaittu se mahdollisesti hieman toimivammaksi, tai ainakin sen cool-indeksi on korkeampi.

IT: "Tuo JSR-168 standardi on suht jees API, sillä voi kato ihan tuoteriippumattomasti vääntää portletteja."

PJ: "Hei..katos vähän tota, onpas aika kissa, kylläpäs on vetävän näkönen. Oikein syötävän elikkäs suuhunpantavan näköinen."

Neljäs vaihe, hyperaktiivinen innostus alkaa olla jo hieman rasittava, mutta niin kovin tuttu vaihe. Kukapa ei tietäisi ympäristöstään useita tässä vai-

heessa likoavia ihmisiä. Teinit ovat hullaantuneita hevosiin, bändeihin, idoleihin, mopoihin. Aikuisemmat kuka mihinkin.

IT: "Jukra muuten se uusiin kesäkuussa julkaistu JSR-286 standardi on ihan sikamakea API, sillä voi kato tehdä inter-portlet -kommunikaationkin ihan kybällä"

PJ: "Siis hei oikeesti kato, kato nyt tota mimmiä. Uuh, sori, että mä kuolaan tähän sun takille. Tunneksä sitä, eikös sun vaimos ollu samalla osastolla? Mikä sen nimi on? Mun täytyy käydä kysymässä."

Viides vaihe on seitsemäs taivas niille, jotka sinne pääsevät. Tämä vaihe on Flow-tilan ylimmän konkretian ilmentymä, joka tuottaa siellä olijoille suunnatonta tyydytystä, mutta heidän lähiympäristölleen tämä on äärimmäisen turhautumisen ja sekaannuksen olotila.

Evangelioiminen on vaihe, jossa ihmisen päätä ei saa käännettyä muuhun suuntaan, ajatusmallirakenne on sementoitunut jo niin vahvasti, että siitä jää elinikäisiä vaurioita. Potilaalla on pakonomainen tarve levittää uskoaan kaikille, jotka liikkuvat tai eivät liiku parin sadan metrin läheisyydessä tai sitä kauempana.

IT: "Mä olen vahvasti sellasen vision kanssa liikenteessä, että tämä sovellus on tehtävä palveluorientoituneen arkkitehtuurin mukaisesti tai sitten sitä ei tehdä ollenkaan. Ei meidän kannata lähteä ollenkaan tollaseen vääräuskoiseen toteutusmenetelmään mukaan, koska SOA on ihan paras tähän tarkoitukseen. Ja kaikkiin tarkoituksiin. Meidän koko IT eräajot pitää alistaa konvertointisuunnitelman alaisiksi ja sillä lailla me saatais kaikesta SOA-complianttia ihan muutamassa vuodessa. Se olisi aika rankkaa, mutta oikean asian takia kannattaa taistella."

PJ: "Hei sen mimmin nimi on joku Johanna, ihan mieletön muia tiäksä, se on ollu semmoses voimisteluryhmäs ja nyt se hoitelee koko markkinointiosastoa, tai siis mä tarkoitan, että sillä on niitä joitaki markkinointivastuita ja se on joku pomo siellä puolella ja sillä on toinen pikkurilli vääntynyt sillälailla aivan ihanan sympaattisesti ja arvaa, se osaa röyhtäillä ihan niinku miehet, siis se on oikeesti aivan ihana nainen, sun pitäis kans saada joku shemmoinen nainen ittelles, mä voi kysyä, onko sillä siks..siks..siskoa tai jotain, mut hei kato nyt kuinka upea se on..."

Joulua kaikille!



Evankelioiminen on vielä kanssaihmisille vielä suhteellisen helppo vaihe, koska potilaan ulosanti on jollain lailla vielä ymmärrettävissä. Seuraavassa vaiheessa tilanne on muuttunut jo katastrofaaliseksi. Kielilläpuhuminen on vaihe, jossa potilaan viestintä on puuroutunutta ja vain hyvin harvojen asiantuntijoiden (tai vertaisryhmään kuuluvien oppilaiden) ymmärrettävissä olevaa spesifikoitunutta jargonia. Tässä vaiheessa, kun kaikki alkaa olla jo menetettyä, potilaan mielenterveyden rippeiden eheyttämisen ja ennenkaikkea ulkoisen tyytyväisyyden kannalta olisi tärkeää saada potilas kommunikoidaan samalla tavalla ajattelevien yksilöiden kanssa, niin vieraannuttavalta kuin se saattaa omaisista tuntuakin.

IT: "Nää WSRP:t on ihan ehdottomia, se tägi-kirjasto ja pääsy CC/PP-dataan sen vanhemman API:n kautta on ihan uskomattoman hieno juttu. Siinä kuule semanttinen ja sosiaalinen webbi nuri-see hoosiannaa, kun BPELin kautta rytkästään tietomallit UML:stä XML:ään Apachen serverispecifikaation läpi portaaliin, josta ECMAScriptillä ne voidaan incorporoida servletti-pohjaisiin referenssi-implementaatioihin. Mä voin briiffata sua näiden kanssa, niin pääset noista vanhoista Javapelleilyistä eroon ja tekemään jotain oikeesti tuottavaa hommaa, nimittäin tää web2.0 on kyllä silleen hanskattavissa, jos on kunnon toolkitit ja JVM, jolla sitä voi puukottaa."

PJ: "Thiätskä hei... shen nimi on Ohana, siis Joo.. Haanna. Sen manikhyristiset kurvit on .. hoh hoijjaa, ja kuule mitat on ihan mallissa, niinku 29-84-20-44 tai ainaki ja sit lishäks ne shilmät.. ja sit se...hei mä tarjoon sulle kaljan, otaksä viskin? Ai ei vai viksi maistu, sinishet kato ne shilmät, niinku mun kisshalla kun mä olin pieni, sholi hei ihan maailman viksuin kihh..kissah. Ääliäli, hoh ho. Krhmöö. Ööö, kalijaa, misshä ne kaikki nais-het on!! Rai-Rai-Rai!!"

Kielilläpuhuminen normaalisti kestää varsin vähän aikaa. Se on tietyllä tavalla kulminatiopiste, hypetyksen voima ei kestä siellä huipulla kovin kauaa. Mikäli vaihe kestää poikkeuksellisen kauan, on pakko siirtyä kemialliseen lääkitykseen, koska lobotomialeikkaukset kaikessa tehokkuudessaan ovat nykyään jostain hatarasta syystä kiellettyjä. Näiden vaiheiden läpikäynti on kokonaisuuudessaan ottaen terveellistä ja lisäksi niiden

sivutuotoksina saamme nauttia useista maailmanhistoriaa muuttaneista keksinnöistä. Tuli, sähkö, puhelin, Big Brother -formaatti ja jääkaappimagneetit, kaikki nuo elämää helpottavat asiat ovat syntyneet jossain hypetyksen vaiheessa. Seuraavaa - normaalielämään orientoituvaa - vaihetta kutsutaan kirjainyhdistelmällä LAH (Life After Hype). LAH on vaiheena hyvin monimuotoinen, se saattaa sisältää aineksia monista multilabiileista projektisoivista tunteista kuten häpeä, tapahtuman kieltäminen, flegmatia ja vastustus.

IT: "No se WSAD nyt oli sellanen hetken leimahdus, kyllähän sillä silloin jotain sai aikaiseks, mutta eihän kukaan täysjärkinen sillä nykyään vois kuvitella mitään koodaavansa."

PJ: "Siis hyivvit..si, mikset sä hei kertonu mulle eilen, millanen haukka se oli. Mulla oli niin karmeen kova migreeni ja olin juonu varmaan kolmattakymmentä viskiä, enhän mä muuten sel-lasen kanssa olis ees tanssinu. Sitäpaitsi sillä oli niin tyhmat jutut eikä se ymmärtäny ees wrestlingistäkään mitään. Mä en tajuta, mikä muhun meni. En kyllä aio juoda enää ikinä".

Kaikki vaiheet voidaan käydä läpi nopeasti, varsinkin katalyyttisten aineiden vaikutuksen alaisena. Joillakin ihmisillä jokin asia saattaa jäädä aiempiin vaiheisiin, esimerkiksi evankelioimiseen. Evankelioijat voivatkin saada ansaittua omaisuuksia esimerkiksi myyntialalla. Tai yritysvalmennuksessa.

Hypetyshän on suoraan käsitettävissä yhtenä osa-alueena Richard Dawkinsin Meemi-teoriasta. Geenithän, kuten kaikki tietävät, ovat solutasen perusyksiköitä. Meemit taas ovat sosiologisen tason perusyksiköitä. Kuten geenit, myös meemit periytyvät, joskaan eivät solutasolla vaan oppimisen kautta.

No niin, nyt olemme käsitelleet hypetyksen eri vaiheet tervejärkisesti. Joko muuten joululahjat on hankittu?

Rauhallista Joulua ja Onnellista Uutta Vuotta!

IKUUTTAMILLA



HYPE JA TERVE JÄRKI

ICT-ala on tunnettu lyhenteistään. Hype-käyrän alussa ei ole täyttä varmuutta lyhenteen taustalla olevan menetelmän tai teknologian elinvoimaisuudesta. Hype-käyrän kohotessa kiinnostus kasvaa ja käytännön toteutus lisääntyy. Soveltamisessa on oltava kosolti tervettä järkeä ja ymmärrystä, jotta energiaa ei haaskata turhaan. Energiasi säästämiseksi tärkeimpään Tieturi tarjoaa valmennuksia niin hype-teemojen tulkintaan kuin vakiintuneiden käytäntöjen tervejärjestykseen soveltamiseen. Säästä aikaasi ja valitse sopiva valmennus!

Muutosjohtamisen valmennusohjelma	26.1. alk.
Tuloksekas tiimidynamiikka	9.2.
BPM pähkinänkuoressa	9.2.
Liiketoimintaprosessien kehittäminen	29.-30.1.
CRM mitattavassa liiketoiminnassa	5.-6.2.
Projektiosaajaksi iisisti	30.1. alk.
Taitavan projektipäällikön leadership-taidot	19.-20.1.
Prince2 Foundation	25.-27.3.
COBIT Overview	22.1.
ITIL Foundation	12.-14.1. • 26.-28.1.
IT Service Manager	14.1. alk.
Sovellusten elinkaaren hallinta	19.-21.1.
Ketterät käytännöt ja niiden käyttöönotto	26.-27.1.
UML määrittelyjä	29.-30.1.
ISEB Intermediate Certificate in Software Testing -valmennus	2.-4.2.
ISEB ISTQB Foundation Certificate -valmennusohjelma	16.-19.2.
ISEB Practitioner Certificate in Test Analysis -valmennus	23.-25.2.
SOA-palveluiden määrittely ja suunnittelu	26.-27.2.
Expression Web	26.-27.1.
Visual Studio Team System 2008 Test Edition	25.3.
Ketterä Visual Studio Team System Workshop	21.-22.1.
JavaFX-ohjelmointi	18.-19.3.
Qt programming	28.-30.1.
Introduction to Mobile Linux	2.2.

Jotain ihan uutta www.tieturi.fi. Tule kokeilemaan!

puh. 09 4315 5333 | kurssit@tieturi.fi | Helsinki | Turku | Tampere

OLE
AIKAASI
EDELLÄ

 **Tieturi**
www.tieturi.fi