



Testaus hajautettuna suunnitelmavetoisessa ja ketterässä mallissa

Kari Kakkonen toimii Testaus ja menetelmät -osastokeskuksen johtajana ja johtavana testauskonsulttina Enderossa. Kari on toiminut testauksen parissa jo vuodesta 1996 ja on konsultoinut ja kouluttanut lukemattomia asiakkaita, projekteja ja yrityksiä eri liiketoiminta-alueilla niin yksittäisissä projekteissa, koulutuksissa kuin kotimaisissa ja ulkomaisissa seminaareissa. Kari on jäsenenä Testaus-osaamisyhteisön ohjausryhmässä, Finnish Software Testing Boardin hallituksessa sekä ISTQB:n kansainvälisessä Expert Level -työryhmässä.

Testaustoimintaa harjoitetaan yhä useammin eri paikassa kuin kehitystiimi tekee kehitystyötään. Monista eri syistä tapahtuvaa testauksen hajautusta yhdistävät yhteiset haasteet. Nämä ovat ratkaistavissa tiiviillä kommunikoinnilla ja yhteisillä tietoteknisillä palveluilla, kuten portaalit ja hallintatyökalut.

Testauksen hajautuksen määrittelyä

Testaus hajautettuna ymmärretään usein erityisesti merkityksessä "testaus ulkoistettuna eri yrityksen tekemäksi eri maantieteelliseen sijaintiin". Maantieteellinen sijainti on tietysti eri jo silloin, kun syntyy kehitystiimistä erillinen testaus tiimi oman yrityksenkin kasvaessa. Toisaalta saman testaus-tason testauskin voidaan hajauttaa useisiin sijainteihin. Järjestelmätestaus tiimi voi olla useassa eri yrityksen toimipisteessä ja se voi olla jakanut tuon testaus-tason tehtäviä keskenään. Myös testaustasojen määrä johtaa hajautukseen. Kullekin testaus-tasolle muodostuu usein oma tiimi, joka usein istuu omissa tiloissaan. Jotkin testaus-tasot, kuten hyväksymistestaus, usein organisoituvat oman luonteensa takia automaattisesti hajautetuksi. Jos henkilöt työskentelevät muutenkin erillään, niin myös testaustehtävissä.

Taustaa hajautuksen lisääntymiselle

Testausammattilaisuuden kasvu

Testausta hajautetaan yhä enemmän. Tämä tapahtuu erityisesti testausammattilaisuuden ja yleisen testausymmärryksen kasvun ajamana, jolloin puoliautomaattisesti syntyy kehitystiimistä erillisiä testauskeskittymiä. Erillinen testauskeskittymä yleensä toivoo itselleen oma tilan saadakseen aikaan tiimihenkeä.

Isot hankkeet – monta toimittajaa

Toinen merkittävä tekijä ovat yhä isommat järjestelmä-hankkeet, jotka jakavat testauksen eri sijainteihin. Isoissa hankkeissa on usein eri asiain keskittyviä testaustasoja (tai perättäisiä vaiheita). Samoin isoissa hankkeissa on usein eri toimittajille tehtäväksi annettuja osia, joka ajaa testauksen tapahtumaan sekä kunkin toimittajan tiloissa että tilaajan tiloissa. Isoissa hankkeissa osa kehitystiimeistä voi toimia suunnitelmavetoisilla toimintatavoilla (esim. V-malli tai vaikkapa iteratiivinen RUP) ja osa käyttää ketteriä toimintatapoja. Tällöin hankkeen tasoinen testaus on hajautettua vaikka yksittäiset osaprojektit voivat välttää hajauttamista hyvinkin tehokkaasti. Ks. kuva 1.

Isossa hankkeessa voi olla eri ohjelmistokehityksille eri osaprojekteissa.

Isot hankkeet – monta tilaajaa

Toisaalta järjestelmän tilaaja itsessäänkin on usein maantieteellisesti hajautunut. On useita toimipisteitä ja käyttäjät istuvat yhdessäkin toimistossa ympäri taloa ilman jatkuvaa kontaktia toisiinsa. Käyttäjistä kootaan usein hyväksymistestaaajat, jolloin hyväksymistestaus helposti on nimenomaan hajautunutta. Isoissa yhteisissä järjestelmissä tilaaja voi olla myös joukko organisaatioita, jolloin hyväksymistestaaajat ovat myös organisatorisesti erillään. Hajautus on astetta laajempaa. Tällaisia hankkeita on paljon mm. julkisella sektorilla ja eläkevakuutussektorilla.

Ulkoistus

Lisäksi hyvän testausosaamisen etsintä ja kustannuspaineet ajavat ulkoistamaan testaustoimintaa eri testausyrityksille ja sitä kautta usein, joskaan ei välttämättä, eri maantieteellisiin sijainteihin. Yksittäinen testaus-taso tai -vaihe, tyy-

pillisimmin järjestelmätestaus, rajataan omaksi tehtäväalueekseen, jolle haetaan osaamiseltaan ja kustannuksiltaan sopiva testausyritys tekijäksi. Tällainen ulkoistus voi olla projektiluonteinen toimeksianto tai pysyvä kumppanuus.

Vastavoimia hajautukselle

Ketterät toimintatavat samoissa tiloissa

Vastavoimana testauksen hajautumiselle ketterät toimintatavat lähentävät testausta kehitystiimin ympärille. Ketterissä toimintatavoissa on useimmiten tavoitteena saada päivittäinen kommunikaatio toimimaan yhdistämällä kaikki ohjelmistokehitystyön roolit määrittelystä kehitykseen ja testaukseen samoihin tiloihin. Tämä toimiikin hyvin riittävän pienen ohjelmiston tai järjestelmän kehityksessä. Kun henkilömäärä ei juuri ylitä kymmentä henkeä, kaikki mahtuvat samoihin tiloihin. Hieman isommassa mittakaavassa muutama kymmenen henkeä mahtuu vielä päivittäin yhteiseen projektipalaveriin, esim. Scrum-mallia käytettäessä päivittäiseen 15 minuutin scrumiin.

Ketterät mallit roolittuvat

Usein ketterissä malleissa esitetään utopistisia tavoitteita, joissa jokainen ketterän tiimin jäsen pystyy samoihin tehtäviin. Osaamisen kannalta tämä on lopulta täysi mahdottomuus. Kukaan ei pysty osaamaan kaikkea. Käytännössä kaiken osaamisella tarkoitetaan ohjelmistokehittäjien tasolla suhteellisen monipuolista osaamista ja tehtävien jakoa; jokainen osallistuu itselleen osaksi tulleen toiminnallisuuden määrittelyyn, toteutukseen ja yksikkö- sekä integrointitestaukseen. Sen sijaan määrittelyjen tekemisen ja järjestelmätestauksen osaamiset eriytyvät omiksi rooleikseen. Määrittelyt voivat olla liiketoimintavaatimus- tai arkkitehtuuritason asioita. Järjestelmätestausta laaja-alaisempaakin testausta voi olla. Palataan siis heiluriliikkeenä V-mallista ketterään toimintatapaan menon jälkeen V-mallimaiseen roolitukseen. Eri roolit voivat tietysti silti tapahtua samassa aikaikkunassa. Hajautuksen kannalta nämä eri roolit liikkuvat helposti, yhteisten tilojen käydessä ahtaiksi, erillisiin sijainteihin. Silti ketterä toimintatapa pitää hajautettuakin ryhmää hyvin kasassa – tämä on yksin hajautuksen hallinnan avainasioita.

Osaamisen puute vähentää hajautusta

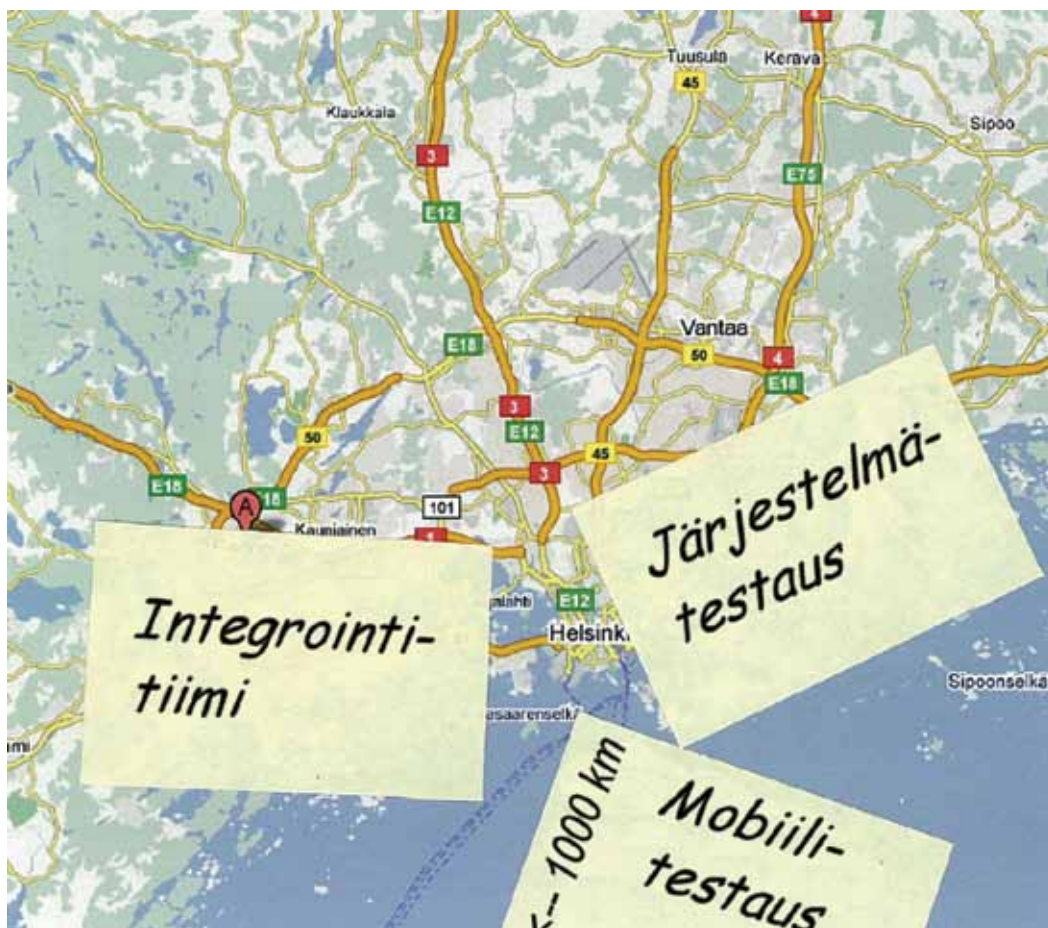
Liiketoimintaosaamisen puute johtaa nykyään yhä useammin ulkoistettuun sijaintiin asti hajautetun testauksen palauttamiseen lähelle kehitystiimiä. Tällöin on ensin ajaututtu ongelmiin ajattelematta, mitä ja miten voi tehokkaasti ulkoistaa. Ratkaisuna tuodaan esim. järjestelmätestaus takaisin omien ihmisten, tai järjestelmäkehityksen kohdalla ohjelmiston toimittajan, tehtäväksi. Ulkoistettu testaus voidaan myös tuoda kehitystiimin lähelle samoihin tiloihin, vaikkakin organisatorinen ulkoistus pidetään osaamis- ja kustannussyiden vuoksi voimassa. Joskus tietyn teknisen arkkitehtuurin testausosaamisen puute voi johtaa samaan tilanteeseen.

Hajautuksen onnistumisen keinoja

Kommunikaatio

Kommunikaatio on aina tärkeässä roolissa ohjelmistokehityshankkeissa. Hajautetun testauksen ollessa kyseessä kommunikaatio on kriittinen asia. Sen täytyy toimia hyvin ja sen täytyy toimia tarvittaessa päivittäin. Kommunikointiin löytyy keinot, kun ne etsitään. Tärkein asia on pitää kommunikointi tiiviinä, kun hajautusta ensimmäistä kertaa luodaan. Ei anneta tilanteen liukua kerran kuussa tai harvemmin tapahtuvaan kommunikointiin. Pikaviestimet kuten Skype, videoneuvottelut, sähköposti ja puhelin on keksitty ja niitä on syytä käyttääkin.

Kuva1.
Isossa hankkeessa voi olla eri ohjelmistokehitysmalleja eri osaprojekteissa.
Kuva: Matti Vuori



Keskitetty seuranta ja suunnittelu

Oivallisia keinoja säännöllisyyden aikaansaamiseen ovat keskitetty korkean tason suunnittelu ja seuranta. Käytännössä tarvitaan kokonaistestausuunnitelma ja vastaavasti kokonaistilanteen etenemisraportti, joiden kautta kaikki näkevät kokonaisuuden, eivätkä vain omaa osaansa kokonaisuudesta. Näiden menetelmien tarkkuustaso luonnollisesti vaihtelee valtavasti riippuen ohjelmistokehitysprosessista – suunnitelmavetoinen vai ketterä. Jotta nämä menetelmät toimivat, tarvitaan yleensä keskitetty testauksen koordinaattorin rooli, joka katsoo kokonaisuutta yli mahdollisten maantieteellisten ja organisaatorajojen. Roolin ei tarvitse olla täysipäiväinen työ, vaikka se sellaiseksi isossa hankkeessa helposti muuttuukin. Onko ensin olemassa keskitetty rooli vai kokonaiskuvan aikaansaamiseen ja ylläpitämiseen tarvittavat menetelmät, on tyypillinen monikana-ongelma. Molemmat kuitenkin tarvitaan.

Läpinäkyvyys

Toisaalta läpinäkyvyys eri sijainneissa tapahtuvan testauksen yksityiskohtiin on sekin tärkeää. Pelkällä kokonaiskuvalla ei pääse pitkälle. Voidaan ajatella, että on aktiivista ja passiivista kommunikointia. Aktiivisessa kommunikoinnissa järjestetään esim. puheyhteys hajautettujen ohjelmistokehityksen eri tahojen välille, esim. statuspalaveri kerran päivässä Skypen välityksellä. Passiivisessa kommunikoinnissa luodaan läpinäkyvyys. Annetaan mahdollisuus kaikille osapuolille ja etenkin koko testauksen koordinaattorille nähdä, mitä tapahtuu, periaatteessa milloin vain. Käytännössä tämä saadaan aikaan käyttämällä keskitettyjä tietoteknisiä ratkaisuja. Pitäisi pystyä luomaan läpinäkyvyyden aikaansaaminen automaattiseksi. Käytännössä tehdään työtä keskitettyjen ratkaisujen kautta, eikä niin, että vain raportoidaan keskitettyihin ratkaisuihin.

Julkaisukäytännöt

Toki käytössä täytyy olla muitakin toimintatapoja. Näistä olennaisimpia ovat päivittämisen koonnin mahdollisuus, sovittu tiivis julkaisusykli testausympäristöihin ja julkaisuselosteet. Päivittäinen koodin koonti eheäksi, yksikkötestatuksi, ajokelpoiseksi ohjelmistoksi tai järjestelmäksi pitää ohjelmiston tai järjestelmä eheänä. Samalla se poistaa hajautuksen perisyntejä – suuret viiveet ja liialliset rinnakkaiset koodin kehityshaarat. Päivittämisen koonnin asennus kuitenkin pitäisi olla mahdollisuus, ei pakote. Testausjärjestelmän tarvitsee myös testausrauhaa. Toisaalta vanhaa versiota ei kannata liian pitkään testata. Pitäikin sopia tiivistä julkaisusyklistä, jolloin saadaan jo raportoituja vikoja varmistustestaukseen uuden version myötä. Sopiva sykli voi olla esimerkiksi asennus kaksi kertaa viikossa. Lopulta julkaisuseloste, tuo kaikille testausympäristöille olennainen asia, nousee uuteen arvoonsa hajautuksen kohdalla. Hyvästä julkaisuselosteesta selviää, mikä

on muuttunut aiempiin versioihin nähden: mitä uutta, mitä korjattua. Hyvä julkaisuseloste syntyy automaattisesti.

Osaaminen ja tuki

Ehdoton edellytys hajautuksen onnistumiselle on myös se, että kaikilla hajautetuilla testausosapuolilla on riittävät tiedot ohjelmiston käytöstä, teknisestä arkkitehtuurista sekä testauksen toimintatavoista ja työkaluista. Ohjelmiston käytön osaaminen on siis ns. liiketoimintatuntemusta. Kaikkien osapuolien ollessa yhdessä tämä tieto on helpostikin saatavissa. Hajautetusti jokin tieto jää monesti puuttumaan. Täytyy siis organisoida riittävä aktiivinen tuki. Tämä jää keskitetyn testauksen koordinoijan vastuulle.

Keskitetty ympäristö

Tekninen ympäristö on merkittävin tekijä, joka mahdollistaa hajautetun testauksen toiminnan lähes kuin oltaisiin samassa paikassa. Puhutaan erilaisista etätyöratkaisuksista. Testausympäristöihin pitää olla reaaliaikainen pääsy kaikilla osapuolilla, toki käyttötarpeiden mukaan rajattuna. Virtualisoidut testausympäristöt mahdollistavat tämän vielä varhaisempia ympäristöjen jaon teknologioita paremmin, yleisimmin käytetään VMWarea. Tarvitaan yhteinen projektikommunikaation alusta, työtila tai portaali, johon jokainen pääsee lukemaan ja lisäämään tietoja, niin tekstiä kuin tiedostojakin – yleisimmin käytetään wikiä. Myös tiedostopalvelimet tulee jakaa. Testauksen hallinnan ja konfiguraation hallinnan työkalut täytyy myös saada toimimaan paikasta riippumatta.

Yhteenveto

Testauksen hajautus yleistyy, se on tosiasia. Oli syy sitten iso hanke ja sen tuomat useat testaus- tasot tai ulkoistus, testaukselle on järkevää löytää toimivia hajautuksen muotoja. Hajautuksesta saadaan menestystarina, kun hoidetaan hyvin kommunikointi, keskitetty riittävä suunnittelu ja seuranta, hyvät julkaisukäytännöt, läpinäkyvyys eri osapuolten kesken, osaamisen tuen tarjoaminen sekä toimiva hajautettu tekninen ympäristö. Samaan aikaan kannattaa tehdä tiiviissä tiimissä samassa paikassa niitä asioita, mitkä ilman hajautusta voi tehdä. Ei siis luoda hajautettua tiimiä turhaan viereiseen huoneeseen, vaan nautitaan integroidun tiimin nopeasta kommunikaatiosta.