

Muudatusehaldus

Muudatusehalduse eesmärgiks on kindlustada, et süsteemis tehtavad muudatused oleksid tehtud, kasutades standardseid ja efektiivseid protseduure ja et muudatuse tegemine häiriks võimalikult vähe süsteemi tööd.

Muudatusehalduse võib jagada etappideks:

- Planeerimine
- Riskihaldus
- Testimine
- Juurutamine

Riskihalduse teemaga tegeletakse vastavas loengus. Järgnevas teeme aga ülevaate ülejäänud etappidest.

Planeerimine

Kõik süsteemis läbi viidavad muudatused (ka uue süsteemi loomine) saavad alguse planeerimisest. Nagu alati, on ka siin oluline planeerida mõistlikus mahus, mitte liiga vähe - et töö käigus mitte hätta jääda, ning mitte liiga palju - nii ei jõua ialgi tegudeni.

Planeerimisega alustatakse juba siis, kui alles hakatakse otsustama, kas üldse on vaja hakata muudatust tegema (näiteks käivitada mõni uus süsteem). Juba selles faasis on võimalik tõsiselt eksida. Eelkõige on oluline hakata tegelema süsteemi muutmise vastavalt ettevõtte reaalsele vajadusele, mitte mingi müügimehe jutu või ülemuse tuju tõttu. Vajaduste puudumisel või valesid probleeme lahendades on uute süsteemide loomine ja vanade muutmine lihtsalt raha raiskamine.

Plaani koostamisel on vajalik püstitada konkreetsed ja mõõdetavad eesmärgid, mille saavutamine on ettevõtte võimetekohane. Samuti tuleb kindlaks teha hetkeseis ning vahendid, mida saame eesmärkide saavutamisel kasutada. Seejärel asutakse koostama plaani, mis kirjeldab sammud, kuidas kasutada olevate vahendite abil ja hetkeseisust alustades jõutakse püstitatud eesmärgideni.

Tegevusplaani loomisel ei maksa unustada varianti, et muudatus ei õnnestu esialgse plaani järgi - mingite ootamatute probleemide tõttu. Nii on oluline luua ka varuprotseduur või plaan B, mis püüab saavutada eesmärki teisi teid pidi. Võib ka juhtuda, et muudatust ei õnnestu vajaliku aja jooksul vajalikus mahus läbi viia või selgub, et ei põhi- ega varuplaan ei aita eesmärki saavutada. Sellistel puhkudel kasutatakse taganemisplaani, mis taastab süsteemi muudatuse eelses seisus.

Uue süsteemi käivitamisel ning olemasoleva muutmisel on planeerimise etapis mõningaid erinevusi. Üldiselt on uue süsteemi juurutamisel aega vabamalt ning riskid väiksemad, kui vanas süsteemis muudatuste tegemisel. Samas on uute süsteemide käivitamisel rohkem segadust ning teadmatust, kui juba käivasse süsteemi muudatuste tegemisel. Töötava süsteemi muutmisele võivad omakorda rakenduda väga ranged ajapiirangud.

Igasuguste muudatuste puhul tuleks selgeks teha, milliseid süsteemi osi plaanitud muudatus puudutab ning millised valmisprotseduurid on olemas ning millised tuleb luua. Näiteks: andmebaasi kolimisel ühelt platvormilt teisele (näiteks Oracle PC-lt SUNi serverile) võib kasutada valmis varundamise ja taastamise protseduure mis on olemas manuaalides. Samas näiteks kolimiseks ühest süsteemist teise (andmebaaside näitel näiteks Postgresist Oraclesse) tuleb protseduur vastavalt kasutatavatele rakendustele ise luua.

Süsteemi muudatuste tegemisel on oluline riskide maandamine - et süsteem ei satuks muudatuste tagajärjel mittetöötavasse situatsiooni. Siin tuleb kindlasti planeerida süsteemi tarkvarast ning andmetest varukoopiate tegemine või mingid muud meetodid muudatuste tagasivõtmiseks, kasulik oleks need ka enne ära proovida. Kriitiliseks teguriks on ka ajapiirangud, nii on oluline võimalikult täpse ajakava kasutamine. Sealjuures ei maksa teha väga optimistlikke prognoose süsteemide töö ja inimeste jõudluse kohta, optimistlike prognooside puhul kipub reaalsus prognoositust hullem olema.

Uute süsteemide käivitamisel tuleb teha plaan kogu süsteemi elukäigu kohta: alates sellest, kuidas ta luuakse, kuidas juurutatakse, kuni selleni, millal võiks süsteemi pensionile saata. Mida detailsem on see plaan, seda kallim on planeerimine ning seda keerulisem, aga ka olulisem, on hilisemate muudatuste puhul plaani muutmine. Samas võimaldab detailne plaan suurte meeskondade paralleelset tööd ning teisalt lihtsamalt ja täpsemat kontrolli kogu protsessi üle. Näiteks: ühemehefirma raamatupidamistarkvara juurutamine pole kuigi keeruline, ning ei vaja palju planeerimist, samas kui kümnete tuhandete töötajate ja rahvusvahelise haardega suurfirma raamatupidamise juurutamine vajab väga detailset planeerimist.

Edasi on oluline süsteemi ulatuse valik: ei maksa lasta kahuriga kärbest ega õngega vaala püüda. Mida suurem ja võimsam on süsteem, seda keerukam ta on, seda mahukam on juurutus ning seda rohkem on süsteemil ka eeldusi. Nii on suurematel süsteemidel, peale selle, et nende alghind on suurem, ka varjatud kulud suuremad.

Planeerimise vahendid

Erinevalt tarkvaratehnikast on süsteemihalduse poolne planeerimise faas metoodikate ja vahenditega üsna katmata. Enamjaolt tegutsetakse kogemuste ja harjumuste baasil. Töövahendiks kasutatakse tekstiredaktorit ja vahel ka tabelarvutust. Muidugi võib enda tarbeks kohandada infosüsteemide projekteerimisest tuttavaid vahendeid ja metoodikaid, seda enam, et suur hulk süsteemihalduse-alaseid planeerimisi käib käsikäes mõne tarkvaraprojekti planeerimisega.

Abiks on ka kõikvõimalikud visuaalsed vahendid – skeemi- ja graafikaredaktorid. Arvutisüsteemide visuaalne kirjeldamine parandab kõigi seotud osapoolte arusaamist tulevases süsteemist või muutustest. Praktikas kasutatakse seda väga tihti arvutivõrkude ja arvutivõrkude muudatuste planeerimisel. Peale süsteemi visuaalse planeerimise saab visuaalselt planeerida ka süsteemi erinevaid eluetappe ja seotud tegevusi: sõltuvusgraafid, *timeline*'d, otsustusdiagrammid.

Mida rohkem on uues süsteemis valmisprodukte (nii riistvara kui tarkvara poolt), seda olulisem on ka tootjatepoolse dokumentatsiooni osa planeerimises. Tähtsamad tegevused on siin tootjadokumentatsiooni süstematiseerimine ja kontroll – kas kõik plaanitud tegevused on dokumentatsioonis piisavalt kirjeldatud või tuleb lisainfot leida.

Planeerimise tulemus

Muutuste planeerimise tähtsaimad tulemused on ülevaade muutuste eesmärkidest, nende seos süsteemi eesmärkidega, muutuste ulatus, kasutadaolevad ressursid, muutuste täpne kirjeldus (ehk plaan) koos samm-sammuliste protseduurikirjeldusega, edukuse kriteeriumid, esialgne ajagraafik (täpsustub juurutusprotsessis), vastutusosalad. Tegelikus elus on planeerimise väljund (pahatihti minimaalne) alamhulk sellest loendist.

Eesmärgid on tavaliselt esitatud tekstina, ressursid ja süsteemi komponendid tabelina, süsteemi ülesehituse juures on tähtsad joonised, skeemid, eluetapid *timeline*'na. Konkreetseid ning täpseid juhendeid protseduuride läbiviimiseks pannakse kirja lihtsalt nn. „*checklistidena*“, ehk nimekirjadena tegevustest, reeglina läbiviimise järjekorras.

Ka mitmed tootjad annavad erinevate protseduuride kohta välja oma nimekirju. Näiteks kogemustega serveritootjate manuaalid (eelkõige riistvaramanuaalid) sisaldavad sageli väga täpseid protseduure alates sellest, kuidas seadet elektrivõrku ühendada, kuni selleni, kuidas seadet loodida.

Testimine

Ka süsteemihaldus hõlmab testimist, testitakse nii tarkvara kui riistvara. Peamiseks eristavaks jooneks süsteemihalduse puhul on see, et testitakse pigem erinevatest komponentidest loodud süsteemi tervikuna, testitakse neid omadusi, mida omab süsteem kui tervik ja mida komponentidel pole, samuti testitakse süsteemi käitumist erinevates konfiguratsioonides, süsteemi vastavust talle esitatud nõuetele.

Testimismeetodid süsteemihalduses

Suuremas osas kasutatakse süsteemihalduses tarkvaratehnika testimismeetodeid, mida on vastavalt vajadusele kohandatud. Tarkvaratehnikas kasutatakse mõistet „süsteemitest“, mis tähendab seda, et testitav tarkvara ühendataksegi enne testimist valmissüsteemiks. Süsteemitesti mõiste on süsteemihalduses loomulik, s.t enamik siin läbi viidavaid teste ongi süsteemitestid.

„Musta kasti“ testimismeetod käsitleb süsteemi kui tervikut, vaadeldakse kasutaja poolt antud sisendit ja selle tagajärjel tekkinud süsteemi väljundit, selle vastavust spetsifikatsioonile. Süsteemi siseehitusele testimisplaani koostamisel tähelepanu ei pöörata. Võib öelda, et sel meetodil testitakse süsteemi kasutaja vaatepunktist, testija kehasub tavaliseks kasutajaks, kes teeb igapäevast tööd.

„Valge kasti“ meetodil testides koostatakse aga testid süsteemi siseehitust, struktuuri ja konfiguratsiooni silmas pidades. Ideaaljuhul on testitud iga mooduli erinevad harud, iga konfiguratsioonidirektiiv. Suuremate arvutisüsteemide puhul võib see liiga töömahukaks osutuda ja nii valitakse välja olulisemad, enimkasutatavad moodulite funktsioonid ja konfiguratsioonidirektiivid.

Süsteemide arendamisel ja ülesseadmisel kasutatakse verifitseerimisteste, mis testivad süsteemi hetkeseisu, võttes aluseks spetsifikatsioonid, millele süsteem selleks hetkeks peaks vastama. Nii saadakse arendus- ja juurutustööde käigus tagasisidet selle kohta, kuidas antud etapid õnnestuvad, kas ajakavaga ollakse kooskõlas. Vajadusel saab välja tulnud kõrvalekaldeid kiiresti korrigeerida.

Süsteemi lõplikul vastuvõtmisel kasutatakse valiidsusteste, mis testivad süsteemi lõplikku vastavust talle püstitatud ülesannetele. Valiidsustestid keskenduvad pigem lõppkasutajatele pakutavale funktsionaalsusele kui süsteemisistele toimemehhanismidele.

Süsteemihalduses kasutatakse tihti teste fiktiivse(te) kasutaja(te) ja fiktiivse(te) mooduli(te) abil. Testimine fiktiivse kasutaja abil tähendab, et süsteemile „mängitakse“ kasutajat, keda tegelikult pole olemas. Tavaliselt on selle kasutaja jaoks välja mõeldud kindel roll, kindlad käitumisharjumused ja kindlad vajadused ning tööülesanded, mida peab kasutaja süsteemi abiga täitma.

Fiktiivne moodul on tavaliselt kood, mis on küll täpselt sama liidesega, kui moodul, mida ta asendab, kuid reaalselt täidab ta selle mooduli ülesannetest vaid alamosa: näiteks mingit osa kontrollitult korrektselt, mingis osas simuleerib vigu. Tihti annab fiktiivne moodul ka mingit lisaväljundit selle kohta, millised andmeid tema sisendisse antakse. Mõningatel juhtudel on „fiktiivseks mooduliks“ hoopis testija ise.

Tarkvara testimine

Traditsiooniline tarkvaratestimine kuulub süsteemihalduse alla vaid osaliselt. Ideaalis võiks süsteemihaldaja eeldada, et tarkvara on enne tema kätte jõudmist juba korralikult testitud. Kuigi vahetevahel see nii ei ole, jääb traditsiooniline testimine siiski süsteemihalduse skoobist välja.

Süsteemihalduse jaoks tähendab tarkvaratestimine hoopis tarkvaramoodulite, erinevate pakettide koostöö ja tarkvara konfiguratsiooni testimist. Seejuures ei pruugi need moodulid ja paketid olla pärit samast allikast – tegelikult enamasti ongi tegu erinevate tootjate tarkvaraga. Moodulite ja pakettide mingil viisil ühendamise ja sobival viisil konfigureerimise kaudu loodud süsteem omab mingeid uusi omadusi ja uut funktsionaalsust, mida koostisosadel üksikuna võttes pole. Neid omadusi ja funktsionaalsust süsteemihalduses testitaksegi.

Vahel on vaja testida paigaldatava süsteemi sobivust konkreetsesse keskkonda. Selle käigus tehakse esmalt kindlaks keskkonna nõuded ja eripärad, mille alusel koostatakse testiplaan. Kui suurem osa testidest sellises plaanis keskendub just nõuetele ja eripäradele vastavuse testimisele, siis ära ei tohi unustada ka seda, et lisaks tuleb testida süsteemi "tavapärast" funktsionaalsust, millele nõuded ei laiene ja eripärad ei mõju (ei tohiks mõjuda).

Süsteemide jõudlustestides ei kontrollita otseselt süsteemi funktsionaalsust, vaid seda, kas funktsionaalsust suudetakse pakkuda ka ajaliste-mahuliste piirangute raames ning seda, kas funktsionaalsus säilib suurte koormuste all. Jõudlustestides koormatakse süsteemi või selle osasid ning registreeritakse tulemused erinevate koormuste juures. Võib püüda jäljendada nn. "igapäevakoormust", kas siis eelnevalt tüüpilisi suhtlusepisooide salvestades ja neid maha mängides või siis kunstlikult sarnaseid episooide genereerides. Tulemusi võib kasutada süsteemi pudelikaelte leidmiseks või tulevikuhinnangute tegemiseks.

Testiplaan ja testi tulemus

Testimise alusdokumendiks on testiplaan. Laias laastus kirjeldab testiplaan seda, millised peaksid olema süsteemi oodatud väljundid etteantud sisendite korral. Sisendeid võib kirjeldada nii sisestatavate andmete kujul kui ka sammudena, mida testi läbi viimiseks teha tuleb. Erinevatele testidele võib anda järjekorranumbri või nime, samuti võib ette anda testide läbimise järjekorra ja ajagraafiku. Iga testi juurde võib lisada ka eeltingimusi, mis peavad enne testi algust täidetud olema. Testid võivad jaotuda erinevatele tasemetele ka kriitilisuse järgi - mõne testi läbimine võib olla kohustuslik, mõne läbimine soovituslik.

Tulemuste juures võib kirjeldada nii seda, millised peavad olema väljundandmed, millised peab olema süsteemi reaktsioon kasutaja tehtud sammudele ja millistele ajalistele ning mahulistele piirangutele peab kogu kasutusepisood vastama.

Musta kasti meetodil testides on testiplaani kuuluvad testid kirjeldatud pigem oodatava kasutusstsenaariumina, kus kordamööda kirjeldatakse nii testija samme kui süsteemilt oodatud vastuseid neile sammudele. Testi läbi viies pannakse oodatud vastuste kõrvale kirja tegelikult saadud vastused ja hiljem otsustatakse neid võrreldes, kas test läbiti või mitte. Valge kasti meetodil testides on testid tavaliselt väiksemad ja konkreetsemad, testide kogus jällegi suurem. Esitada saab neid näiteks tabelina.

Testimise tulemuseks on täidetud testiplaan. Testiplaani osatulemuste alusel tuletatakse lõpptulemus (kui see on ette nähtud). Testija võib testiplaani täites lisada ka veel oma kõrvalmärkuseid - kõrvalmõjusid, mis ei sobi vormilt testi tulemusena kirjeldamiseks, soovitusi testide modifitseerimiseks jms. Läbi viidud testide lõppkokkuvõtte alusel algatatakse süsteemis ümberseadistusi ja ümberstruktureerimisi, koostatakse veareporteid tootjatele.

Testimine ja turve

Kui peamiselt testitakse positiivset funktsionaalsust, s.t testitakse, kas süsteem omab mingit soovitud funktsionaalsust, siis andmeturbe seisukohalt tuleb testida ka negatiivset funktsionaalsust, s.t testitakse, kas süsteem ei oma mingit soovimatut funktsionaalsust.

Andmeturbe tavade järgi on vajalik nii musta kui valge kasti metoodika kasutamine, sest nii on võimalik katta eritüübilisi turbevigu. Samuti tuleb testida nii disaini- ja konfiguratsioonivigade kui ka tarkvaravigade leidumise vastu. Arvestama peab, et tootjapoolne tugi katab reeglina vaid tarkvaravigu, vigu süsteemi komponentide omavahelistes seostes ja vigu konfiguratsioonis tuleb hallata ja testida peamiselt ise. On olemas tarkvaratooteid, mis püüavad tüüpilisi konfiguratsioonivigu tuvastada ja selleteemalisi soovitusi anda.

Juurutamine

Juurutamine on protsess, mille käigus plaanitud muudatused viiakse ellu. Eesmärgiks on muudatusplaani võimalikult täpne järgimine (et kindlustada võimalikult väike mõju töötavatele teenustele), samuti muudatusprotsessi toetamine kommunikatsiooniga. Kui testimisfaasis on süsteem terviklikult käima saadud ning toimimine kontrollitud, siis juurutamise põhiline raskus on inimeste ja asutuse reeglite kohandamine uue süsteemiga.

Juurutusprotsessi alustatakse sageli juba varakult, testimise ajal, ning lõppeb see protsess siis, kui (muudetud) süsteem on käivitunud ning seatud eesmärgid saavutatud – ehk siis mitte sel päeval, kui muudatused tehtud, vaid pigem siis, kui „lastehaigustest“ ja kõrvalmõjudest üle on saadud.

On mitmeid erinevaid variante juurutusprotsessi toimimise koha pealt. Näiteks on võimalik nn. spiraalne mudel: uue süsteemiga käivitatakse algselt tuumikprotsessid ning sealt mingite ajavahemike vahel rakendatud funktsionaalsust laiendatakse. Samuti on võimalik täielik süsteemi või muudatuse rakendamine kogu mahus. Erinevate võimaluste vahel tuleks valida vastavalt süsteemile, muudatuse ulatusele ja organisatsiooni vajadustele.

Teavitamine

Kindlasti tuleks teavitada kõiki, keda plaanitud muudatus puudutab. Seda nii konkreetse süsteemi otseste kasutajate puhul, kui ka süsteemi kaudsete kasutajate puhul. Nii näiteks tuleks teavitada andmebaasiserveri uuendamisest nii otse andmebaasi kasutavate rakenduste kasutajaid, kui ka veebi-intraneti-kasutajaid. Teavitada tuleb vastavalt muudatusele piisavalt ette, et kasutajad saaks oma tööd planeerida ning vajadusel saaks ka süsteemi muudatusi muule ajale viia. Teavitada tuleks korduvalt, nii varajaselt kui ka otse muudatuste eelsel päeval. Samuti võib suuremate muudatuste tulemustest kasutajatele teada anda. Rutiinsete muudatuste (mis ei tekita töökatkestusi ka ebaõnnestumise puhul) pole teavitamine tarvilik. Kui on oht, et muudatus tekitab katkestuse (aga plaanikohase täitmise korral mitte), võib kasutada „passiivset“ teavitamisviisi, näiteks kasutajatele saadetud e-kirja asemel teade ettevõtte siseveebis.

Teavitamisel tuleks kasutajaid teavitada sellest, mis toimub, mis peaks juhtuma, mis võib juhtuda, millised on plaanitava tegevuse tulemused, ning kindlasti: millal miski juhtuma peaks. Samuti ei maksa unustada kontaktandmeid, juhuks kui kasutajatel on küsimusi plaanitava muudatuse osas.

Muudatustest teada andva dokumendi olulisemad komponendid:

- Mida tehakse?
- Kes teeb?
- Keda muudatused mõjutavad?
- Ajakava
- Kontaktandmed lisainfo saamiseks

Juurutamisel on oluline kiire tagasiside kõikidelt osapooltelt, kes võivad tuvastada probleeme (või nende puudumist) süsteemi mingites osades. Seetõttu on oluline kogu juurutuse faasis plaanida sagedast infovahetust: kas siis elektrooniliste kanalite vahendusel või (suuremate ja pikemat aega nõudvate muudatuste puhul) koosolekute näol.

Ajagraafik

Vastavalt plaanitud protseduuridele tuleb valida inimesed, kes seda muudatust läbi hakkavad viima. Siin kerkib üles küsimus, millistel aegadel on inimesed võimalik tööle saada.

Edasi on vajalik koostada ajakava: millal on muudatuse jaoks tarvilikud protseduurid valmis ning selged muudatust tegevatele inimestele, millal on protseduuri võimalik läbi viia - kas tööajal või väljaspool tööaega. Rutiinsete muudatuste puhul tavaliselt ajapiiranguid pole. Kriitilistele muudatustele on soovitatav koostada ka läbiviimise täpne ajakava: millisel hetkel tuleb olla millises punktis, millal saab käivitada veel plaani B, millal on selge, et antud ajaaknas protseduuri läbi ei saa viia, ning selle peale käivitada taganemiskava.

Tiheda ajakava puhul tuleks tegelikult läbi mõelda ning soovitatavalt kirja panna kõik, mis vähegi untsu võib minna, sest Murphy seadused kipuvad siin sageli kehtima. Mida tihedam ajakava, seda rohkem tuleb tekkivateks probleemideks valmistuda.

Tegeliku muudatuse tegemine

Süsteemi käivitamise tehniline protsess tuleb kindlasti täpselt plaanida: kokku panna meeskond ning täpne tegevuskava. Samuti tuleb valmis panna plaan B ehk varuplaan juhuks, kui kõik ei lähe nii nagu tarvis. Kirjas peaks olema, kus on vajalikud materjalid ja vahendid, ning nende asjade olemasolu näidatud kohtades ka kontrollitud.

Kuigi tootjad pakuvad enamasti oma tarkvarale/süsteemidele paigaldus-, hooldus- ja käivitusprotseduuride kohta dokumentatsiooni, ei pruugi see olla 100% kohaldatav teie keskkonnaga, nii et see materjal tuleks eelnevalt kriitilise pilguga üle vaadata.

Peale tootjate dokumentatsiooni tasub vaadata ka internetti ja otsida lehti, mis viitavad teiega sarnasele olukorrale: kui juurutamisel pole just spetsiaalselt teie asutusele loodud tarkvara, siis on üpris tõenäoline, et keegi on selle kadalipu juba läbi teinud ning oma kogemused kirja pannud.

Ka süsteemide käivitamisel on võimalik kaasata välisabi: sageli pakuvad tootjafirmad koos tootega ka mingit paigaldustuge, kust on probleemide korral võimalik abi saada. Samuti on võimalik kaasata kas tootjaid või toodete edasimüüjaid ka käivitamise läbiviimisesse. See variant maksab muidugi reeglina parasjagu, ent võib suurte süsteemide puhul ära tasuda.

Juhul kui füüsiline ligipääs on vajalik ning vähegi küsimusi tekitab (serveriruumidesse ligipääs pole reeglina väga avalik), siis tuleks ka need protseduurid kindlaks teha ning vajadusel kirja panna.

Muudatuste läbiviimisel ei maksa unustada, et kui on oht, et reaalse juurutustöö ajal elektrooniliselt neid dokumente lugeda ei saa, tuleks vajalikud dokumendid eelnevalt paberile välja trükkida.

Koolitus

Väga tähtis uute süsteemide juurutamise puhul on see, et inimesed tuleks uue süsteemiga kaasa, harjuks selle süsteemiga ning omandaks normaalseks tööks vajalikud töövõtted. Koolitus on süsteemiga kohandamiseks hädavajalik, samuti on vajalik hea dokumentatsioon. Koolituse korraldamine ei pruugi muidugi alati võimalik olla: kas on kasutajate arv liialt suur, või keeruline määrata või väga voolav.

Lõppkasutajad jagunevad sageli mitmesse gruppi: näiteks andmesisestajad ja ülevaatajad. Koolitada tuleb ka süsteemiadministraatoreid ja operaatoreid. Kindlasti ei maksa unustada kasutajatuge: see ala saab süsteemi käivitamisel kindlasti lisakoormuse ning korralik ettevalmistus on hädavajalik.

Suuremate projektide puhul tasub mõelda professionaalse koolitusfirma palkamisele, kuna seal on tööl professionaalsed pedagoogid, kellele arvutisüsteemide õpetamine on igapäevane töö.

Paljudel valmissüsteemidel on olemas ka abimaterjalid kasutajate koolitamiseks. Mõnikord on olemas ka nn. konvertimismaterjalid ehk juhendid kasutajate oskuste ühest süsteemist teise üle kandmiseks. Keerukamatel tarkvaratoodetel on sageli olemas ka oma juurutusfirmad, kus siis juurutuse käigus saab tellida ka koolituse.

Kasutajatugi

Kui süsteemi kasutajad tööle tulevad ja nad isegi teavad, et süsteem on vahetatud/muudetud ja nad on ka koolitatud uut süsteemi kasutama, siis sellest hoolimata võib ette tulla probleeme. Inimeste õppimisvõime ja -meetodid on erinevad ning seetõttu on alguses küsimusi ja segadust. Ühest küljest aitavad segadust vähendada koolitus ja eneseabimaterjalid, ent teisest küljest on hädavajalik korralik kasutajatugi.

Kasutajatuge võib pakkuda teie asutuse kasutajatoerühm (kui on selline) või süsteemiadministraatorid/operaatorid. Samuti on võimalik kasutada ka kasutajaid endid, koolitades osasid kasutajaid teistest põhjalikumalt ning nemad siis toetavad neid, kelle oskused nõrgemad on. Tuleb hoolitseda selle eest, et muudatusejärgsel perioodil oleks kasutajatugi ja süsteemihaldusrid (tavalisest enam?) kättesaadavad.

Organisatsiooni töökorraldus

Organisatsiooni töökorraldus ei ole küll väga tehniline teema, ent kindlasti on see seotud juurutamisega. Nimelt olenevalt juurutatavast muudatusest võivad olla vajalikud suured muudatused töökorralduses ning sellega tuleb ka muudatusehalduses arvestada.

Uusi arvutisüsteeme või vanade süsteemide muutmist kasutatakse sageli peale automatiseerimise ka organisatsiooni töökorralduse ja kultuuri muutmiseks. Sellistest kaalutlustest peavad juurutajad teadlikud olema, et saavutada võimalikult suur efekt.

Asutusesisesed reeglid võivad vajada kohandamist, kuna näiteks võib muutuda mingite andmete kasutamise vajadusega inimeste ring vms. Seetõttu tuleks vastava valdkonna reeglid süsteemi juurutamisel kindlasti üle vaadata.