

Aineisto-ohjattu ja avainsanaohjattu testausautomaatio

Pekka Laukkanen, Qentinel Oy

Tässä artikkelissa käydään läpi mitä ovat aineisto-ohjattu (data-driven) ja avainsanaohjattu (keyword-driven) automatisoitu testaus ja selitetään mitä etuja niistä on. Artikkelin tarkoitus on kaikille testaajille eikä esimerkiksi

```
SetFocus "Calculator"
Push "2"
Push "+"
Push "4"
Push "0"
Push "="
Check "Result=42"
```

Kuva 1 - Yksinkertainen testiskripti

minkäänlaista ohjelmointikoke-
musta tarvita.

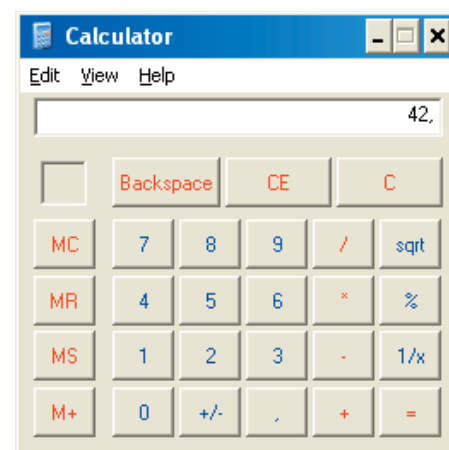
Yksinkertaiset testiskriptit, riippumatta siitä onko ne kirjoitettu käsin vai nauhoitettu, ovat yleisesti ottaen saman tyyppisiä kuin pseudokoodinen esimerkki kuvassa 1. Koodi kyseisessä esimerkissä on tarkoitettu testaamaan kuvassa 2 näkyvää Windows Calculator -ohjelmaa.

Ongelma kuvan 1 tyyppisissä skripteissä on se että käytettävä testidata on kovakoodattuna niiden sisällä. Jos testaaja haluaa testata jotain muuta - esimerkiksi laskun $0+0$ tai $1-1$ - on näille testeille tehtävä omat testiskriptit. Uuden skriptin tekeminen on periaatteessa helppoa sillä sehän onnistuu kopi-
oimalla vanha ja muokkaamalla sitä hieman. Tosielämässä testiskriptit eivät kuitenkaan ole niin yksinkertaisia kuin kuvassa 1 oleva

eikä pientenkään muutosten teko välttämättä onnistu ilman ohjelmointitietämystä. Vielä suurempi ongelma tulee eteen myöhemmin jos (tai paremminkin kun) testattava kohde muuttuu ja kymmenet tai sadat testiskriptit täytyy päivittää. Skriptien ylläpidosta tulee helposti niin iso ongelma että testit on nopeampi koodata tai nauhoittaa uudelleen. Tällöin automaatiolla haetut uudelleenkäytön edut jäävät toteutumatta ja automaation hyödyt ovat kyseenalaisia.

Osittainen ratkaisu näihin ongelmiin on testiaineiston erottaminen testiskripteistä. Testiaineisto voidaan laittaa erilliseen helposti editoitavissa olevaan tiedostoon ja se esitetään yleensä taulukkomuodossa kuten kuvassa 3.

Hyödyt edelliseen esimerkkiin ovat selvät. Testien editointi on helppoa ja ennen kaikkea uusien



Kuva 2 - Testattava ohjelma

testien lisääminen joukon jatkoksi on vaivatonta. Muutokset testattavassa kohteessa voivat tuki edelleen

hajoittaa testijärjestelmän, mutta nyt muutokset rajoittuvat vain yhteen testiskriptiin. Aineisto-ohjattu testaus siis tekee useiden saman tyyppisten testien suunnittelun ja suorittamisen helpoksi.

Itse testiskriptin ei tarvitse olla juurikaan kuvassa 1 olevaa esimerkkiä monimutkaisempi. Taulukkomuotoinen testidata täytyy toki ensin prosessoida ja varsinaisessa testiosassa korvata kovakoodattu testidata muutujilla. Näiden muutosten tekeminen omiin testiskripteihin ei kuitenkaan ole erityisen vaikeaa ja kaupallisissa testituotteissa tällaiset ominaisuudet ovat yleensä käytössä suoraan.

Kuvan 3 esimerkissä on kuitenkin yksi ikävä ongelma - testien täytyy olla saman tyyppisiä. Jos esimerkiksi kuvan 2 laskinta testattaessa

haluttaisiin tarkistaa että myös laskut $1+2+3$ tai $2*5-12+2$ toimivat ei se onnistuisi kuvan 3 taulukolla. Jos testaajille halutaan antaa vapaus tehdä eri tyyppisiä testejä on syytä viedä aineisto-ohjattu testaus astetta pidemmälle ja siirtyä avainsanaohjattuun testaukseen. Esimerkki tästä on kuvassa 4.

Kuten kuvasta 4 nähdään koostuu yksi avainsanaohjattu testi useasta askeleesta ja yhtä askelta taas kuvaa yksi avainsana sekä sille annettavat parametrit. Testaaja voi koota halumansa testin vapaasti olemassa olevista avainsanoista. Vaikka taulukkoon suunniteltu testi alkaa jo näyttää hiukan samalta kuin testiskripti kuvassa 1 on se kuitenkin huomattavasti helpompi ymmärtää ilman ohjelmointitaitoja. Avainsanat ovat myös yleensä hiukan korkeammalla tasolla kuin elementit koodissa - vertaa esimer-

	A	B	C	D	E	F
1	Testi	A	Oper	B	Tulos	
2	yht01	2	+	40	42	
3	yht02	0	+	0	0	
4	väh01	1	-	2	-1	
5						
6						
7						
8						

Kuva 3 - Esimerkki syötetiedostosta aineisto-ohjatussa testauksessa

kiksi sitä että luvun 20 syöttäminen onnistuu kuvassa 4 yhdellä rivillä kun siihen kuvassa 1 menee kaksi.

Avainsanaohjatun lähestymistavan hyvä puoli on juuri se että erilaisia testejä voi tehdä vapaasti. Avainsanaohjatun testijärjestelmän toteuttaminen on kuitenkin vähintään kertaluokkaa hankalampaa kuin aineisto-ohjatun. Aineisto-ohjattu testidata voi myös olla huomattavasti helpompi tehdä ja ymmärtää - vertaa esimerkiksi samaa asiaa testaavaa testiä "yht01" kuvissa 3 ja 4.

Sekä aineisto- että avainsanaohjattu testaus ovat erittäin käyttökelpoisia menetelmiä ylläpidettävien automaattiorakenteiden toteuttamiseen. Molemmissa on etuna mahdollisuus jakaa testien suunnittelu sekä automaation tekninen toteutus näihin tehtäviin erikoistuneille osaajille kokonaisuuden tehostamiseksi. Se kumpi menetelmä mihinkin tilanteeseen parhaiten sopii riippuu lähinnä testattavasta järjestelmästä sekä käytössä olevista työkaluista sekä osaamisesta.

Pekka Laukkanen toimii testaustehtävissä Qentinel Oyssä. Automaattisoinnin jäsentäminen ja kehittäminen on hänen keskeisiä kiinnostuksen kohteina.

	A	B	C	D
1	Testi	Avainsana	Parametri	
2	yht01	Syötä	2	
3		Paina	+	
4		Syötä	40	
5		Paina	=	
6		Tarkista	42	
7	pitkä01	Syötä	2	
8		Paina	*	
9		Syötä	-1,5	
10		Paina	+	
11		Syötä	3	
12		Paina	=	
13		Tarkista	0	
14				

Kuva 4 - Esimerkki syötetiedostosta avainsanaohjatussa testauksessa