

# **Praktisk hantering av automatiserade testfall**

**T U A C A P T M F T W !**

# Das TOC (Hålla låda i ca 35 min)

- Presentation... quien soy yo?
- Bakgrund... vad, varför...?
- Hur man kan testa (in-) parametrar och regelverk på en strukturerad och förvaltningsbar sätt
- Det mjuka... softies...
- T U A C A P, T... M FTW, que?

# Presentation

- Manuel... mindre bra namn...
- Konsult på Alskami
- Testare med specialisering på tekniska tester och testautomatisering



# Bakgrund till tester av ett system

- Spansk fiktiv system vid namn S O L, låter som sol och står för Super Ologico Losning
- System bestående av 30 WSDL-baserade webbtjänster
  - Tjänster med 100 inparametrar, varav 40% spökisar
  - Okänd (till viss del) beteende, odokumenterad
  - System som används av två helt olika typer av aktörer
- Test med SoapUI med hjälp av Groovy och MS Excel... inga gem, ingen tuggummi eller andra myska tingesten... bara skicklighet

# Bakgrund – Två olika sätt

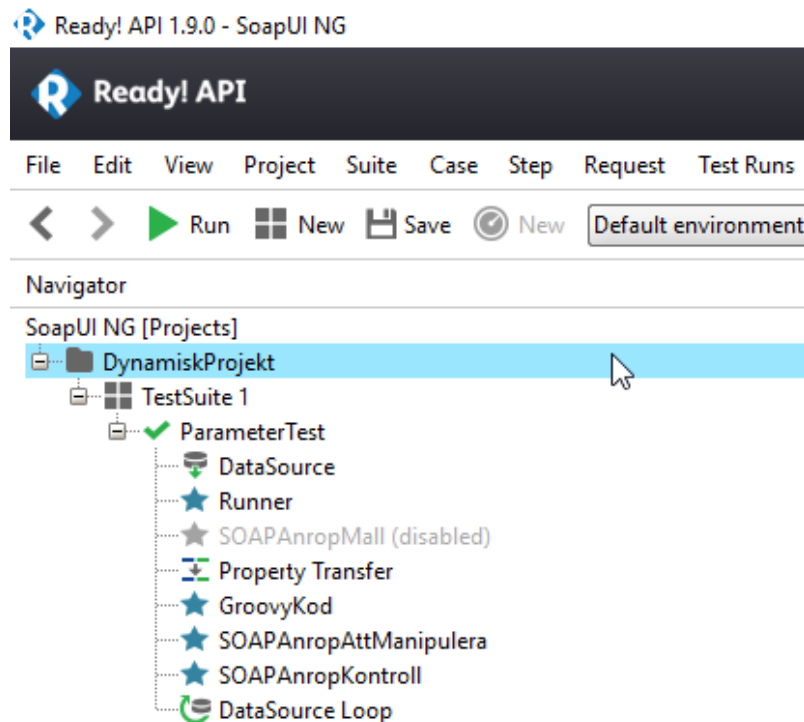
- Del 1 – Test av inparametrar
  - Samtliga inparametrar testas på ett likartad sätt oavsett om det handlar om en skapa-tjänst eller en uppdatera-tjänst
  - Dynamisk manipulering av anrop ("svårt")
- Del 2 – Test av regelverk
  - Regelverk testas genom att kombinera två eller fler inparametrar eller parametrar (indata) och tillstånd (testdata)
  - Parameterersättning ("lätt") med t.ex. DataGen

# Del 1 – Tester av indata, vad menas med det?

- Inparametrar, t.ex. ålder
  - Null, dvs. parametern i fråga skickas inte i anropet
  - Negativa värden, t.ex.  $-1$
  - $0, 1, 2, 3\dots$
- Eller kanske namn
  - Null
  - För få tecken, t.ex. tom sträng (dvs. `""`) eller för många tecken, t.ex.  $\{="t"*65\}$
  - Minst eller max antal tecken, t.ex.  $\{="t"*64\}$

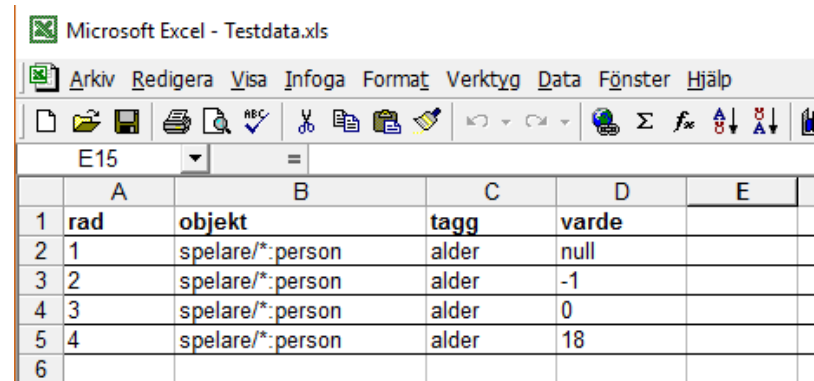
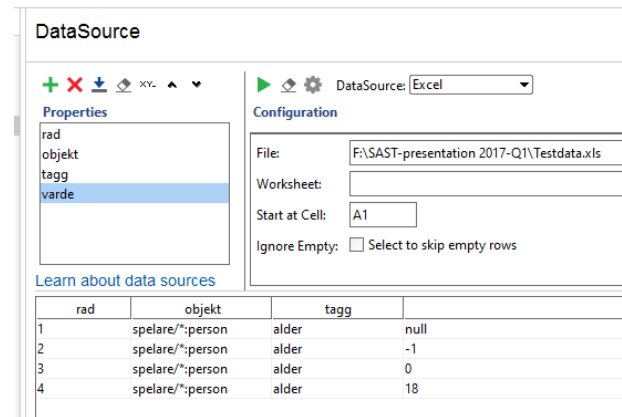
# Del 1 – Struktur på TestCase

- Innehåll i en TestCase
  - DataSource
  - Runner
  - Mall, Transfer och Anrop
  - GroovyKod
  - Kontroll
  - Loop



# Del 1 – DataSource

- Teststeg med MS Excel som styrdata
- Rad, objekt, tagg och värde
- Värde kan vara SoapUI/Groovy-kod
  - `${=’’’’}`
  - `${#Project#person_1}`





# Del 1 – Runner

- Groovy-kod för att avgöra huruvida ett test ska exekveras eller inte
- Om man har för många tester, då vill man kanske med hjälp av kod styra hur stor procentandel man vill exekvera
- Testfall med buggar vill man nog hoppa över
- Vissa testfall vill man kanske alltid exekvera

# Del 1 – Mall, Anrop och Transfer (MAT)

- Mall (disabled)
  - Mall som innehåller alla fält ifyllda med lämpliga värden
  - Ska ej ändras, skall ej exekveras
- Anrop
  - Anropet manipuleras innan den exekveras
- Transfer
  - Steg som kopierar mallen till teststeget där själva anropet ska äga rum

# Del 1 – Er syn har inte försämrats radikalt [GroovyKod]

**/\* BESKRIVNING**

Denna gulliga Groovy-kod-snutt används för att (dynamiskt) manipulera ett WSDL-anrop innan anropet i fråga skickas. Obs! Denna kod ska inte ändras...  
\*/

**// IMPORT**

import groovy.xml.XmlUtil

**// VARIABLER**

```
def groovyUtils = new com.eviware.soapui.support.GroovyUtils( context )
def generatedXML = groovyUtils.getXmlHolder( "SOAPAnropAttManipulera#Request" );
def xmlObjekt = context.expand( '${DataSource#xmlObjekt}' )
def xmlTagg = context.expand( '${DataSource#xmlTagg}' )
def varde = context.expand( '${DataSource#varde}' )
```

**// GULLIG KOD**

// Ta bort parametern eller sätt värde på parametern.

```
if( varde == "null" ) {
    // Parametern i fråga ska vid "null" inte skickas alls, dvs. parametern ska tas bort från anropet.
    generatedXML.remove( "/*:$xmlObjekt/*:$xmlTagg" )
}
else {
    // Annars så ska parametern få värdet i fråga.
    generatedXML.setNodeValue( "/*:$xmlObjekt/*:$xmlTagg", varde )
}
```

// Snygga till anropet och uppdatera den.

```
boolean prettyPrint = true
generatedXML.updateProperty( prettyPrint )
```

# Del 1 – Kontroll

- Assertions ligger inte i anropet i fråga utan i steget efter
- Kontroll sker med if, else if och else satser i kombination med assert-kod och containsKey() getNodeValue() metoder

	forv_svar	forv_obj	forv_tagg	forv_varde
Ok		person	alder	null
Error		n/a	n/a	n/a
Ok		person	alder	0
Ok		person	alder	18

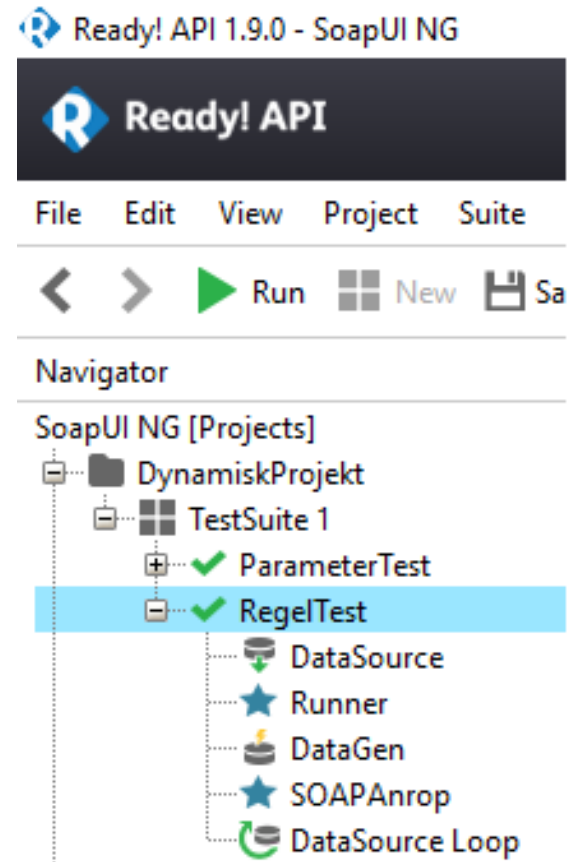
	E	F	G	H	
	forv_med	forv_obj	forv_tagg	forv_varde	
Ok		person	alder	null	
Error		n/a	n/a	n/a	
Ok		person	alder	0	
Ok		person	alder	18	

# Del 2 – Verksamhetsregler

- Verksamhetsregler blir lätt komplicerade
- Permutationer blir ensdöd
- Trevligt att med MS Excel kunna strukturera regler som blir till testfall som blir till styrdata
- Dölja kolumn
- Filtrera kolumn

# Del 2 – Struktur på TestCase

- Innehåll i en TestCase
  - DataSource
  - Runner
  - DataGen
  - Anrop
  - Loop



# Del 2 – MS Excel

- MS Excel används som styrdata
- Varje rad är ett test
- Styrdata kan vara så enkel som en ”null” eller en ”ja” eller kanske en mer komplex variant som ”ja, ibland”

	A	B	C	D	E
1	rad	medlem	aktiv	resultat	kommentar
2	1	nej	nej	1	Person att söka upp...
3	2	nej	ja	2	Kanske dags att skicka några torpeder...
4	3	ja	nej	3	Ja ja... åtminstone är han medlem...
5	4	ja	ja	4	Kontrollera nästa år...
6					

# Del 2 – DataGen

- Varje parameter i en DataGen styrs från en DataSource med avseende på ifall parameteren i fråga ska skickas eller inte osv.

DataGen

▶ + ✖ ^ ▾

---

Properties

Name	Type
medlem	Script
aktiv	Script

---

Configuration

▶ Edit ▾

```
1 def medlem = context.expand( '${DataSource#medlem}' )
2
3 switch( medlem ) {
4
5     case "null":
6         return ""
7         break
8
9     case "nej":
10        return "<medlem>false</medlem>"
11        break
12
13    case "ja":
14        return "<medlem>>true</medlem>"
15        break
16
17    default:
18        assert false
19        break
20 }
```



# Del 2 – Anrop, dvs. XML med DataGen

- Smuggla in DataGen-parametrar:

<forNamn>Manuel</forNamn>

<efterNamn>Farías H.</efterNamn>

#{DataGen#medlem}

#{DataGen#aktiv}

<alder>43</alder>

# Resultat – Vad får jag och vad kostar det?

- Köp 3, betala för 2!
- 5 000 testfall x 2, dvs. 10 000 testfall!
- En TestCase som realiserar 5k och en annan som realiserar 32k kombinationer av tester
- Få TestCases, tonvis med testfall
- Ett dokumenterat system
- 7 månader minus n antal månader... fika...

# Softies

- Uppmana folk att tänja på gränserna, att utveckla sig. För varje testare i Stockholm som växer, så växer Gran Estocolmo... och Sverige... och världen!
- Hjälp och stötta era familjer, vänner, bekanta och främlingar! Hjälp era arbetskamrater!
- Tack till don Tomaso, Stefan Rydin, Johan Larsson och Robert Fred! Och er! Och alla andra!

T U A C A P T M F T W !

Tomas' Über Awesome Kick-Ass  
Parameter Testing Methodology For the  
Win!