

Automazione delle API con Groovy Script

Sommario

[Introduzione](#)

[Creare un progetto soapUI](#)

[Crea una richiesta API soapUI](#)

[Crea un test case soapUI](#)

Introduzione

In questo documento viene descritto come creare una richiesta API (Application Programmers Interface) soapUI e come creare un test case soapUI che esegue un ciclo sulle fasi del test che automatizzano le richieste API a Quantum Policy Suite (QPS).

L'esempio di test case soapUI in questo articolo implementa una procedura di test che legge un file di ID sottoscrittore e quindi crea e invia una query SubscriberRequest a QPS.

Creare un progetto soapUI

Prima di iniziare questa procedura, installare l'applicazione soapUI sul desktop. È possibile scaricare il file eseguibile dell'installazione soapUI da www.soapui.org.

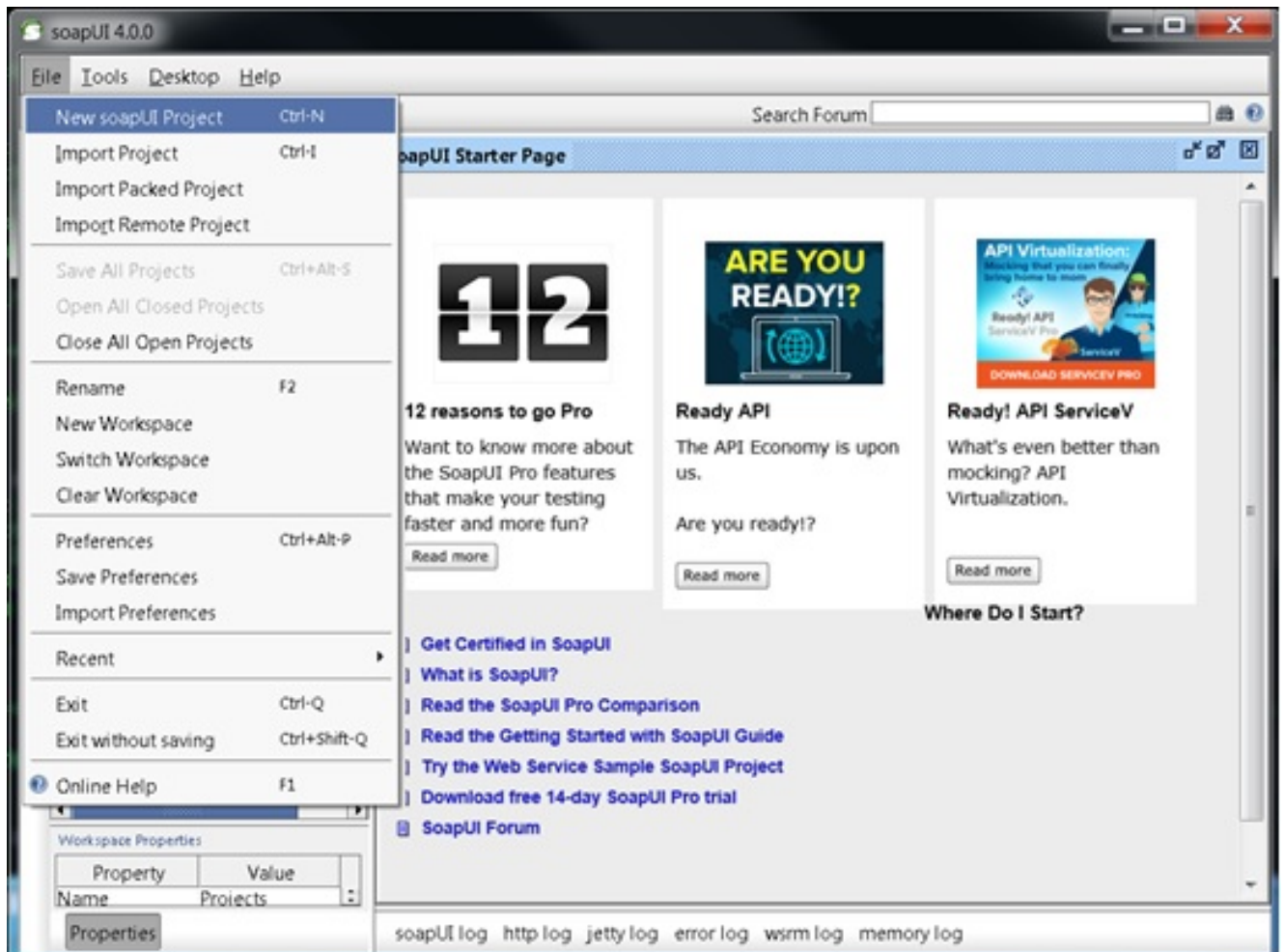
Prima di creare una richiesta API o un test case è necessario creare un progetto soapUI. Per creare il progetto sono necessari il file WSDL (Web Services Description Language) e il file XSD (XML Schema Description). WSDL specifica le API supportate. In genere, è possibile ottenere WSDL e XSD da QPS quando si eseguono questi comandi da Load Balancer (LB):

- `wget http://lbvip01:8080/ua/wsd/UnifiedApi.wsd`
- `wget http://lbvip01:8080/ua/wsd/UnifiedApi.xsd`

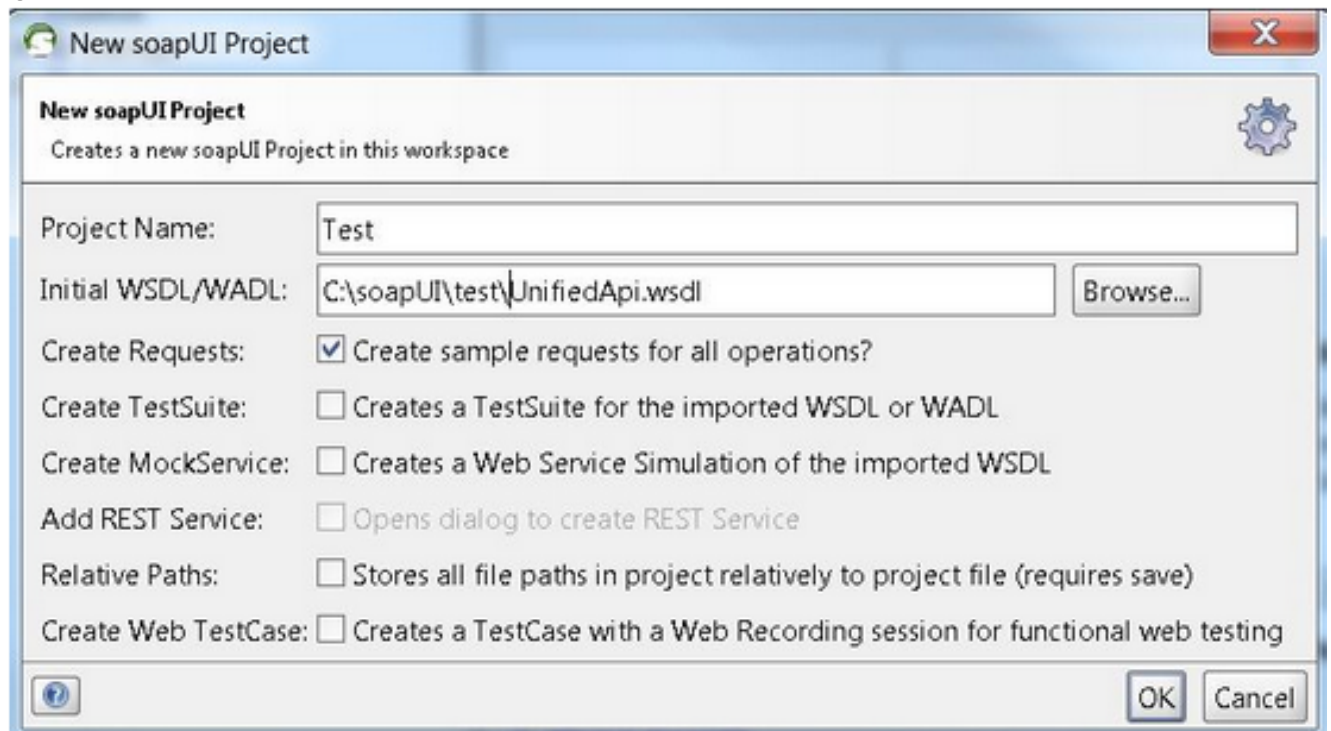
Memorizzare WSDL e XSD nella stessa directory sul desktop in cui si intende eseguire l'applicazione soapUI.

Completare questi passaggi per creare il progetto soapUI:

1. Scegliere **File > Nuovo progetto soapUI** dalla finestra soapUI:



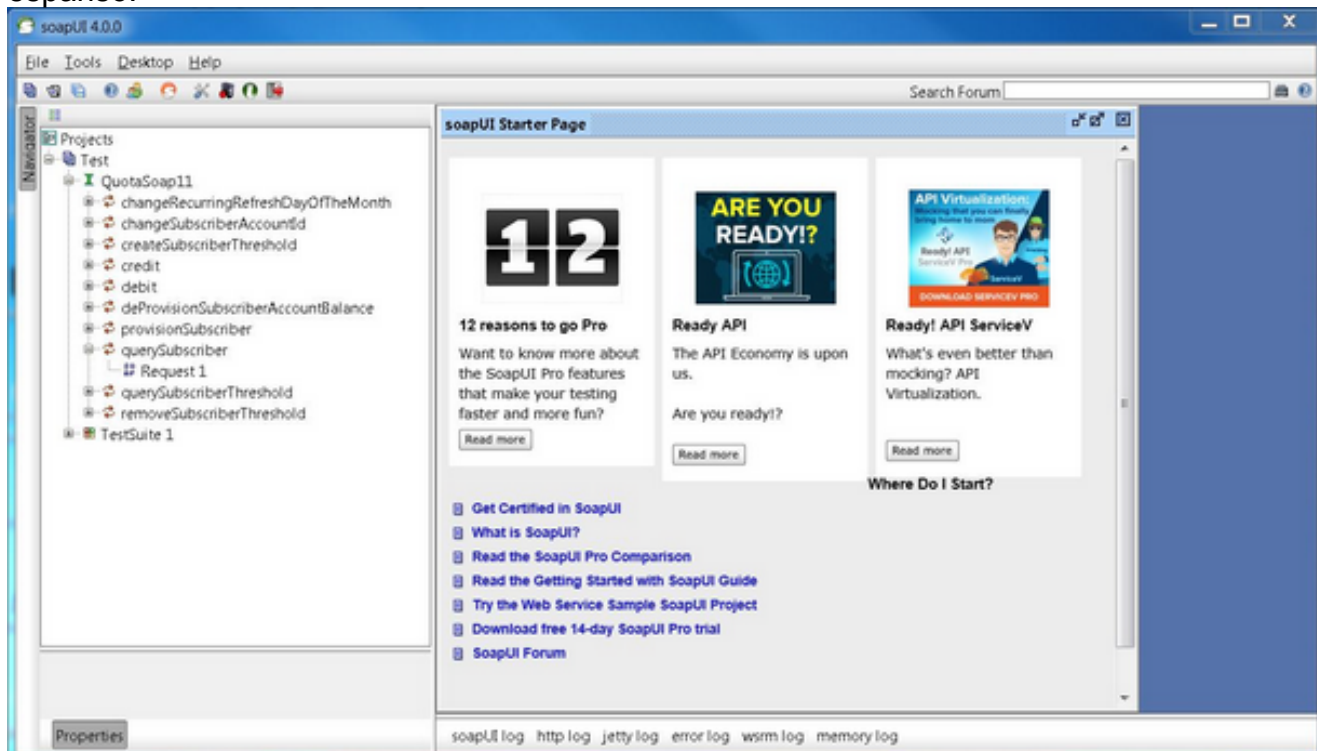
2. Nella finestra Nuovo progetto soapUI immettere un nome per il progetto nel campo Nome progetto e la posizione in cui il file WSDL è memorizzato nel campo WSDL/WADL iniziale. Al termine, fare clic su **OK**.



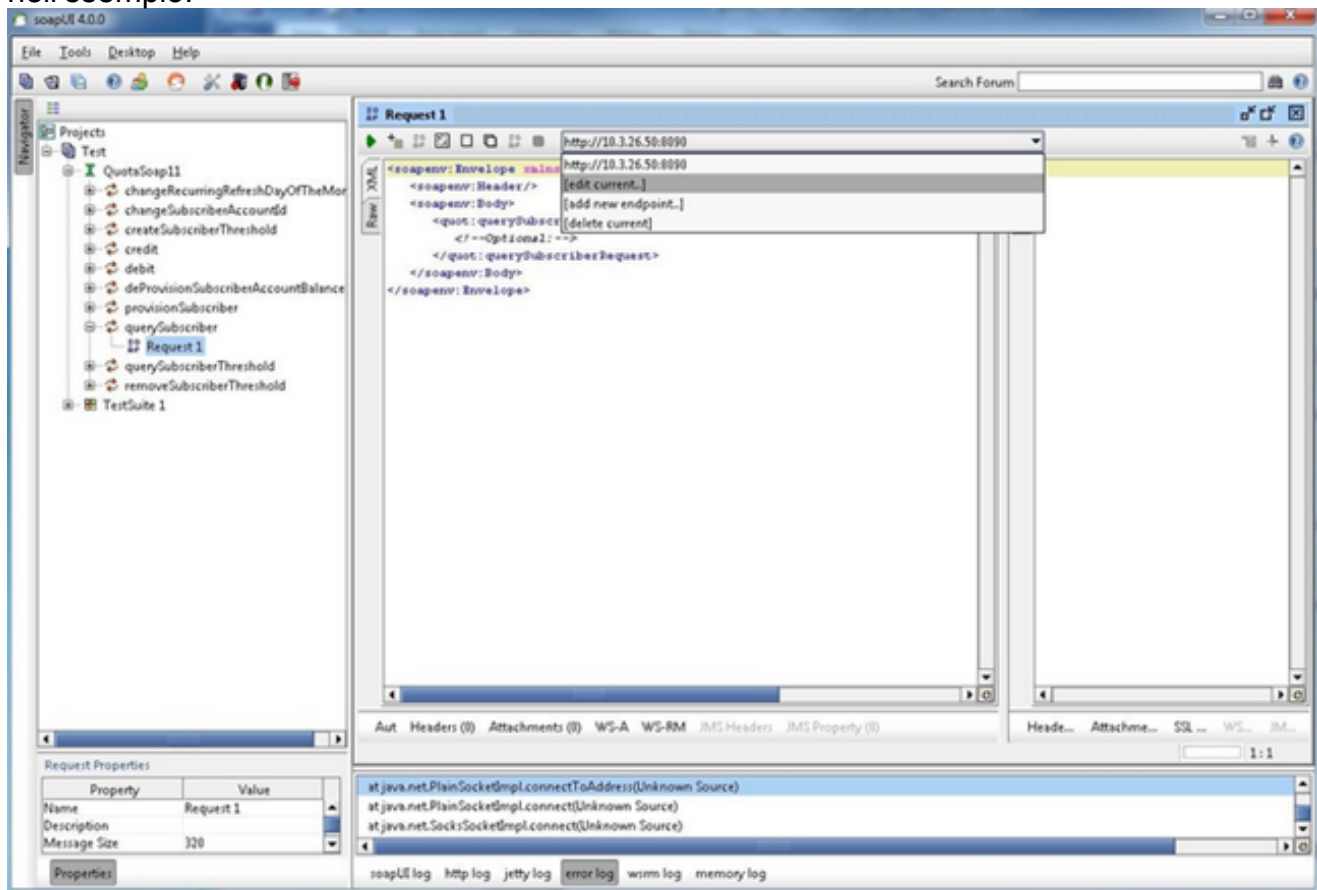
Crea una richiesta API soapUI

Completare questi passaggi per creare una richiesta API soapUI:

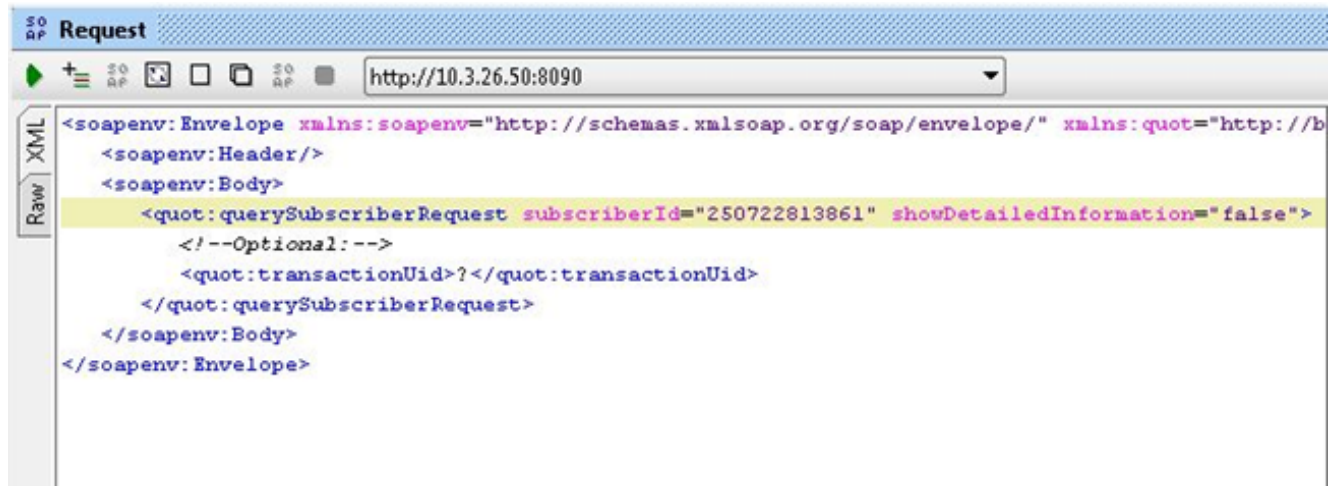
1. Espandere il progetto soapUI creato per visualizzare le API. È inoltre possibile espandere una delle API per visualizzare la richiesta. Nell'esempio, `querySubscriberRequest` è espanso:



2. Aprire la richiesta per visualizzare la finestra della richiesta con il codice XML che forma la query. Nella finestra Richiesta modificare l'indirizzo IP `http://` nell'indirizzo IP e nella porta. Di norma, si tratta dell'indirizzo IP `lbvip01` e della porta a cui inviare la richiesta, come mostrato nell'esempio:



3. Modificare i campi nel codice XML con i dati che si desidera inviare nella richiesta. In questo esempio, la richiesta è querySubscriberRequest. Modificare l'ID del sottoscrittore per il quale si desidera eseguire la query e impostare showDetailedInformation su **false**:



The screenshot shows a SOAP client window titled "Request" with a URL bar containing "http://10.3.26.50:8090". The XML content is displayed in a "Raw" view. The XML structure is as follows:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:quot="http://b">
  <soapenv:Header/>
  <soapenv:Body>
    <quot:querySubscriberRequest subscriberId="250722813861" showDetailedInformation="false">
      <!--Optional:-->
      <quot:transactionUId?</quot:transactionUId>
    </quot:querySubscriberRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

4. Per eseguire la query, fare clic sul pulsante verde **Esegui** nella parte superiore della finestra Richiesta.

Crea un test case soapUI

In questa procedura viene illustrato come creare un gruppo di test in grado di automatizzare l'invio delle API a QPS.

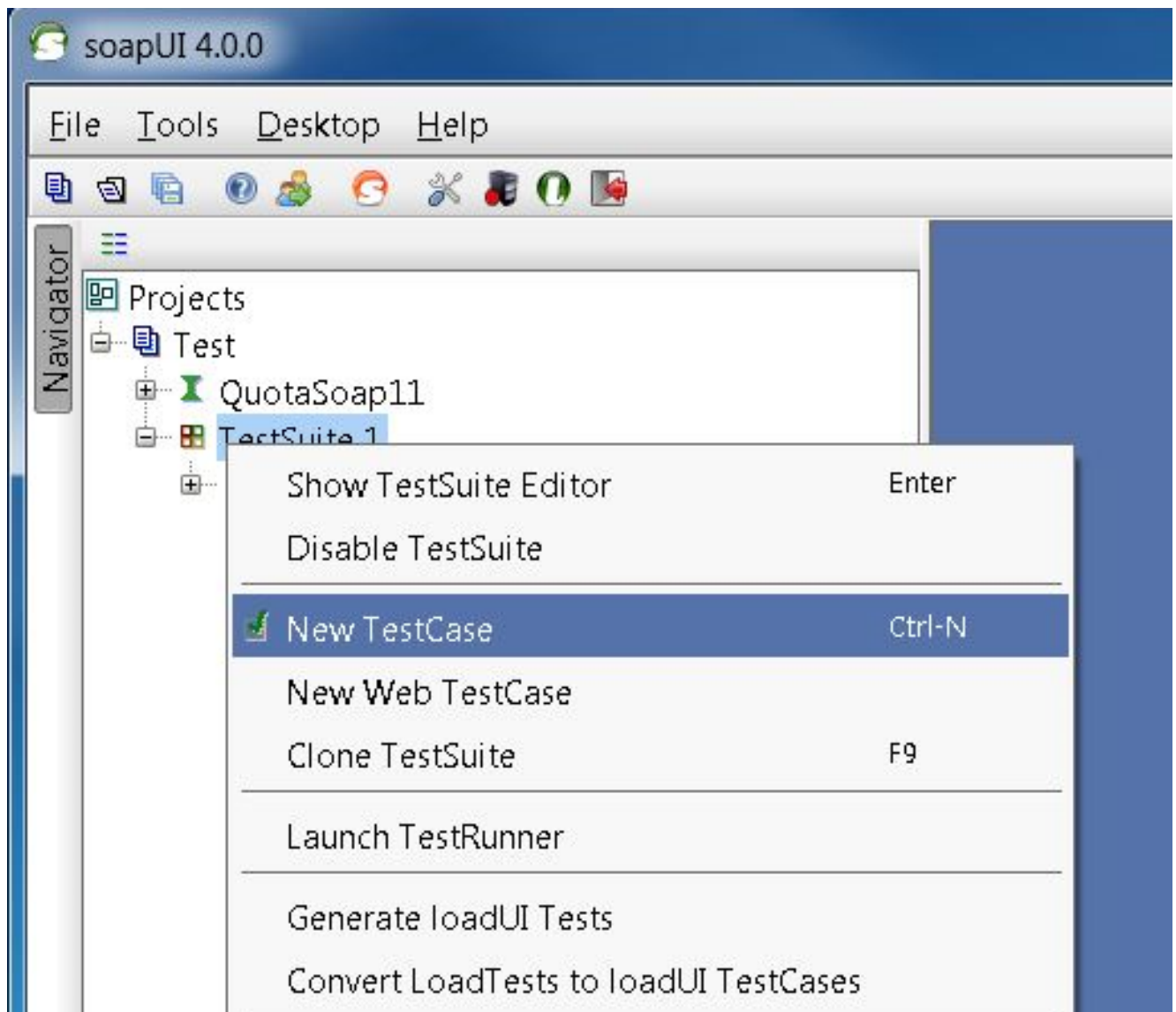
In questa procedura di esempio il gruppo di test esegue un ciclo su un elenco di ID di sottoscrittore e quindi utilizza tali ID di sottoscrittore nella querySubscriberRequest che invia a QPS. L'elenco degli ID sottoscrittore si trova su una singola riga in un file di testo denominato **subid.txt**.

Per creare il gruppo di test, completare i seguenti passaggi:

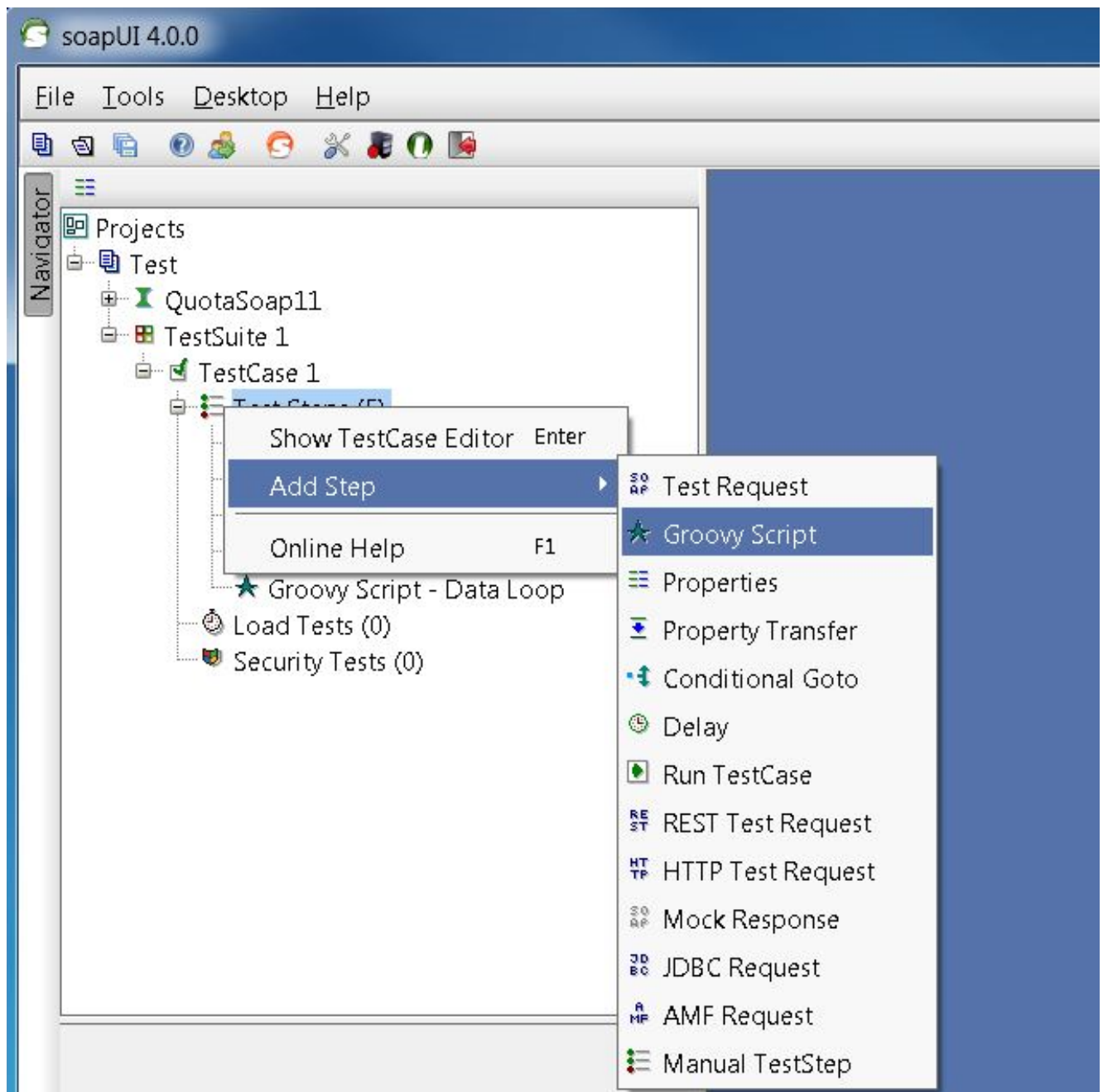
1. Nel progetto soapUI creato creare una nuova suite di test. Fare clic con il pulsante destro del mouse su soapUI e scegliere **New TestSuite**.



2. Fare clic con il pulsante destro del mouse sulla suite di test e scegliere **Nuovo test case**.



3. Per aggiungere un passo di test di Groovy Script, fare clic con il pulsante destro del mouse sul test case e scegliere **Aggiungi passo > Script Groovy**. Denominarlo **Origine dati**:

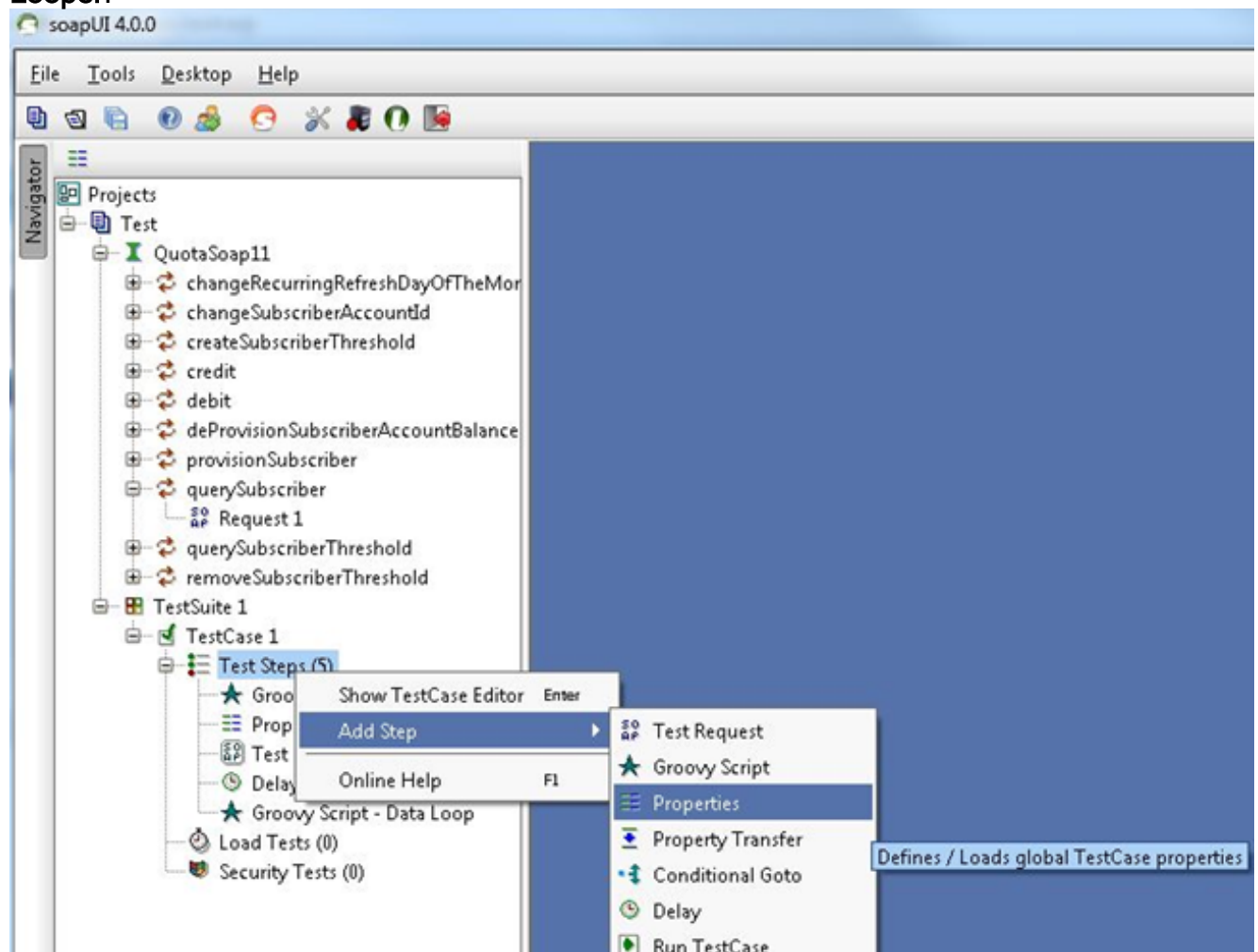


4. Nel file Origine dati incollare il codice. Questo codice legge il file **C:/subid.txt** che contiene un ID sottoscrittore su ciascuna riga:

```
import com.eviware.soapui.support.XmlHolder def myTestCase = context.testCase
def counter,next,previous,sizeFile tickerEnumFile = new File("C:/subid.txt") //subscriber
IDs separted by new line (CR). List lines = tickerEnumFile.readlines() size =
lines.size.toInteger() propTestStep = myTestCase.getTestStepByName("Property - Looper")
// get the Property TestStep propTestStep.setPropertyValue("Total", size.toString())
counter = propTestStep.getPropertyValue("Count").toString() counter= counter.toInteger()
next = (counter > size-2? 0: counter+1) tempValue = lines[counter]
propTestStep.setPropertyValue("Value", tempValue) propTestStep.setPropertyValue
("Count", next.toString()) next++ log.info "Reading line : ${counter+1} /
$lines.size"propTestStep.setPropertyValue("Next", next.toString()) log.info
"Value '$tempValue' -- updated in $propTestStep.name" if (counter == size-1) {
propTestStep.setPropertyValue("StopLoop", "T") log.info "Setting the stoploop property
now..."}
else if (counter==0) { def runner = new
com.eviware.soapui.impl.wsdl.testcase.WsdlTestRunner
(testRunner.testCase, null) propTestStep.setPropertyValue("StopLoop", "F") } else{
propTestStep.setPropertyValue("StopLoop", "F") }
```

5. Fare clic con il pulsante destro del mouse sul passo di test e scegliere **Add Step > Properties** per aggiungere un passo di test Property e denominarlo **Property -**

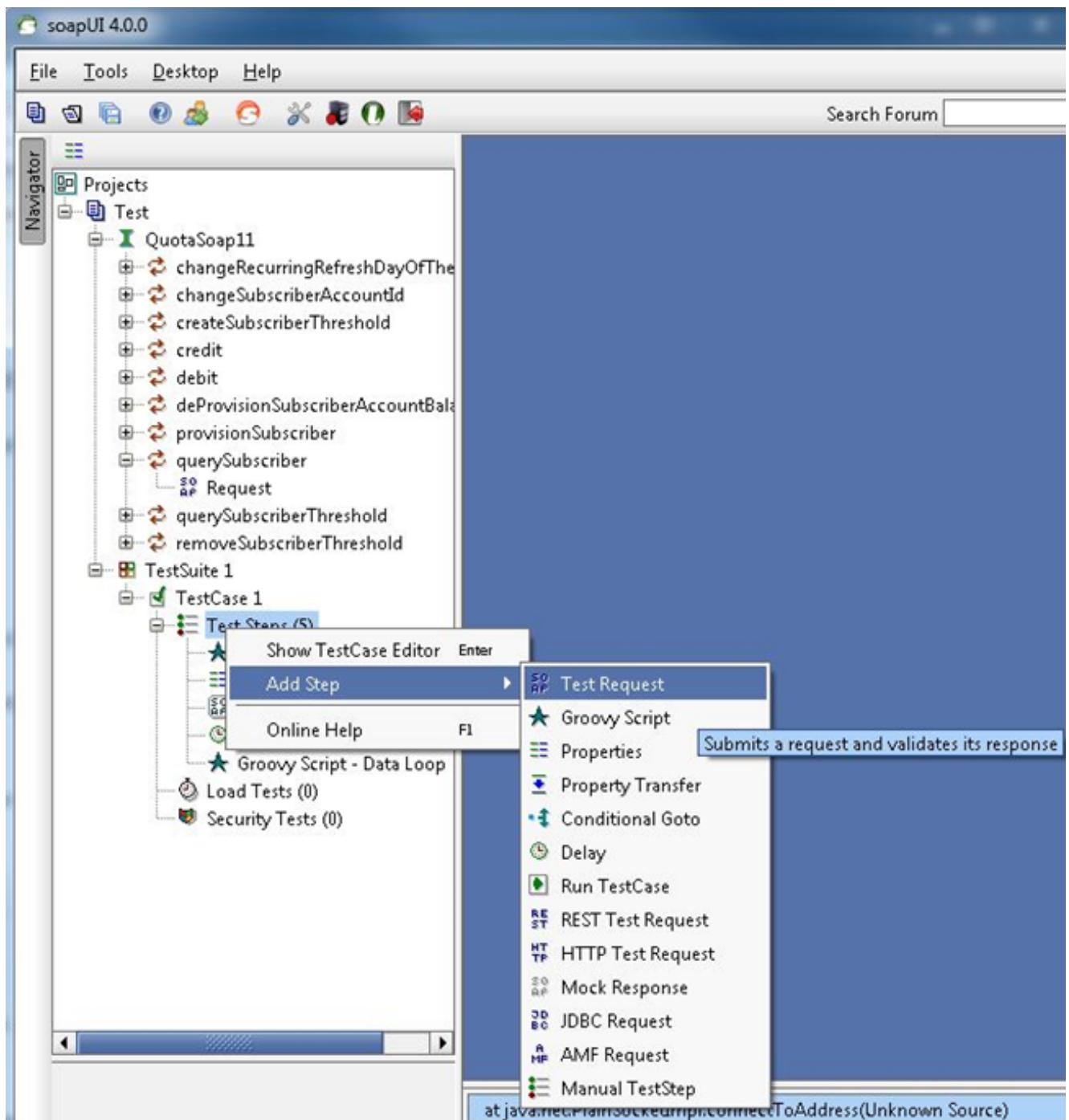
Looper.



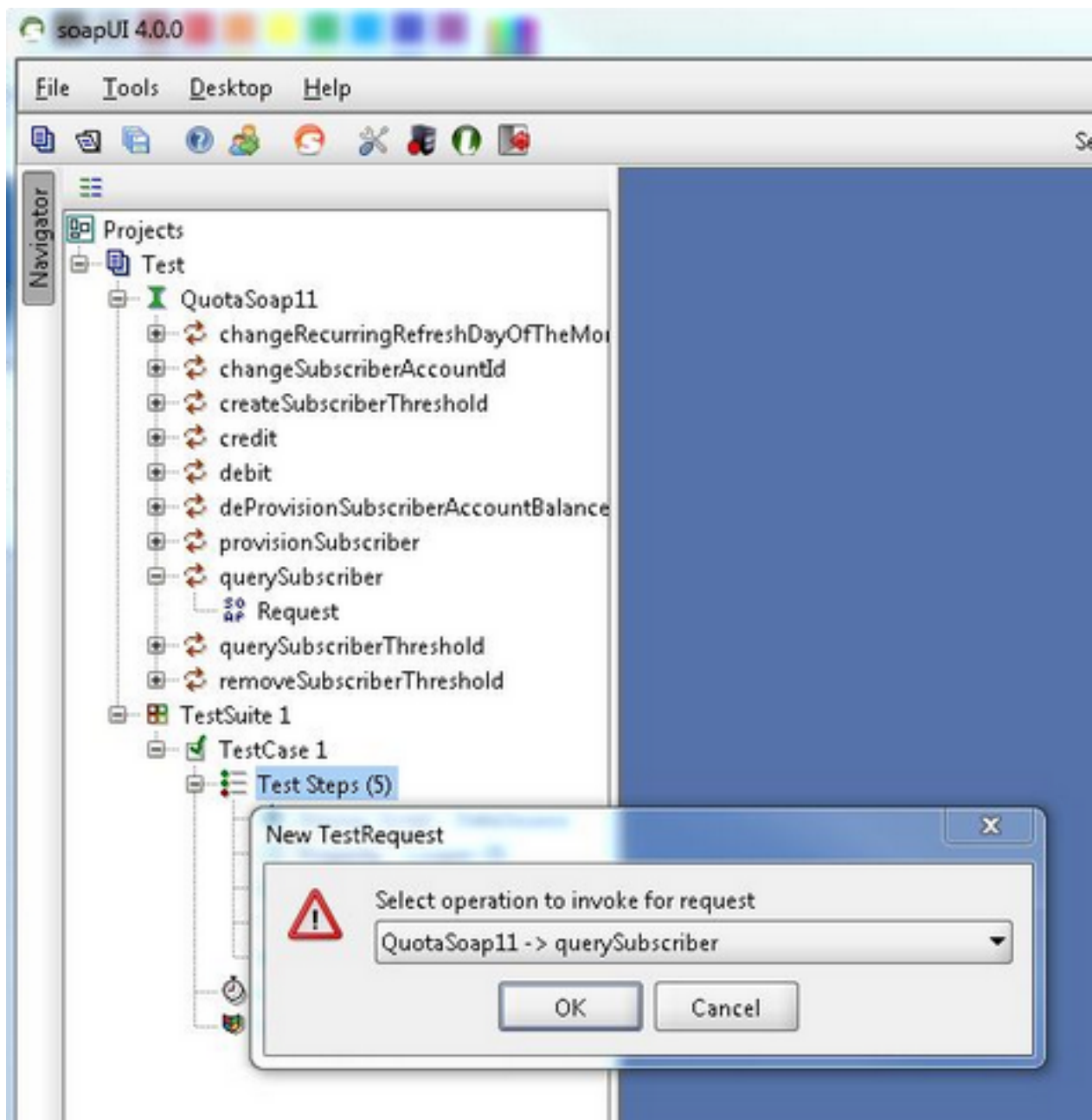
6. Aggiungere le seguenti proprietà definite dall'utente del passaggio del test Looper:
TotaleValore - Nell'esempio riportato sotto, l'ID del destinatario viene letto dagli ID del destinatario del fileConteggioAvantiInterrompiCiclo

Name	Value
Total	
Value	
Count	
Next	
StopLoop	

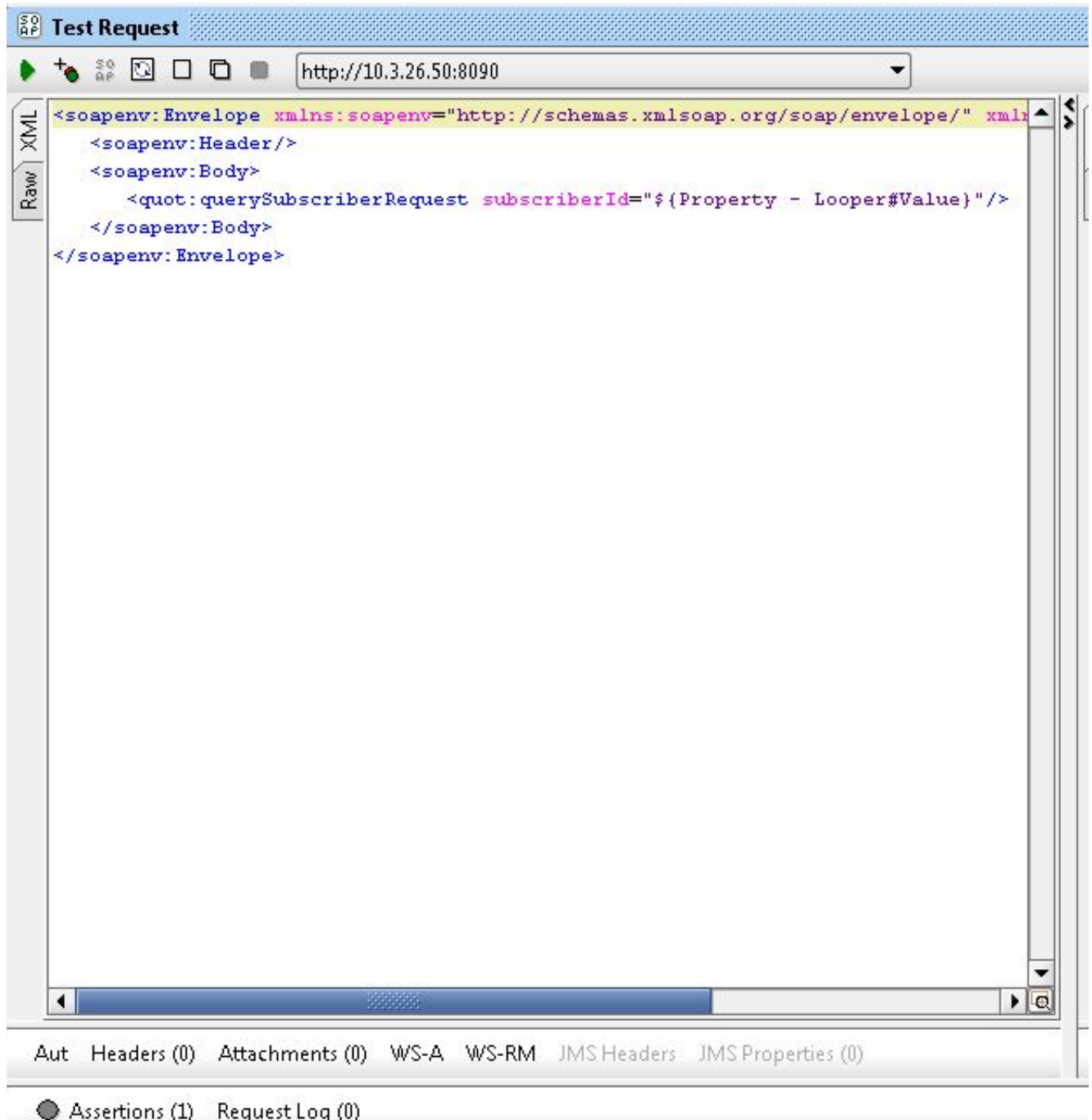
7. Fare clic con il pulsante destro del mouse sul passo di test e scegliere **Add Step > TestRequest** per aggiungere un passo di test della richiesta di test e scegliere la richiesta da richiamare:



Nell'esempio viene utilizzato querySubscriberRequest.

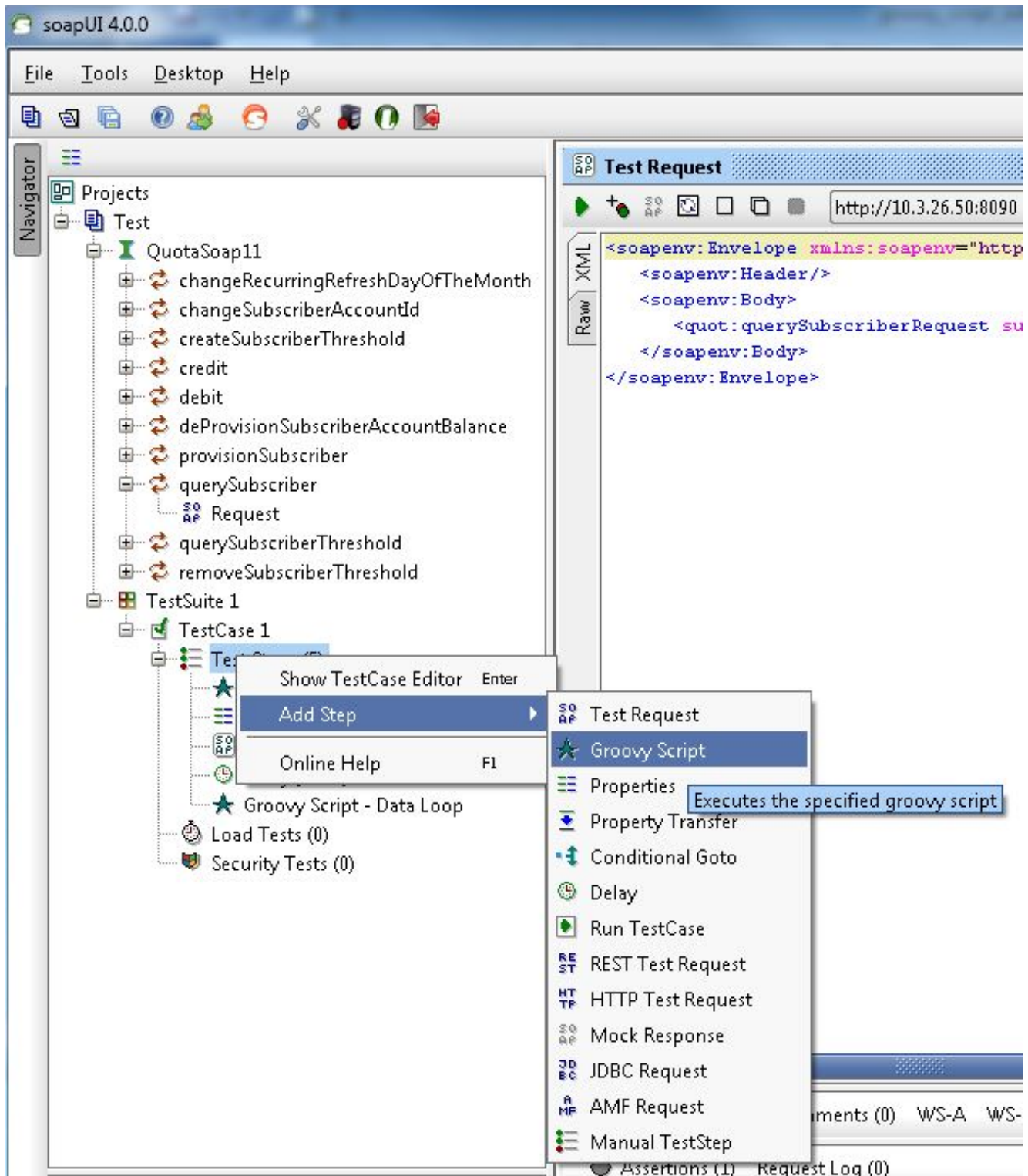


8. Nella richiesta, il codice di espansione sostituisce i valori dei campi dell'elemento interrogato. In questo esempio, il ? del **SubscriberId=?** in querySubscriberRequest è sostituito dal codice di espansione **#{Property - Looper#Value}** (soap_test_req_expansion_code):



Proprietà - Ciclo è il nome della proprietà TestStep creata in precedenza e **Valore** contiene l'ID sottoscrittore corrente letto dal file degli ID sottoscrittore.

9. Fare clic con il pulsante destro del mouse sul passo di test e scegliere **Aggiungi passo > Script Groovy** e denominarlo **Ciclo dati**:



10. Incollare il codice seguente nel ciclo di dati dello script di Groovy:

```

def myTestCase = context.testCase
def runner
propTestStep = myTestCase.getTestStepByName("Property - Looper")
endLoop = propTestStep.getPropertyValue("StopLoop").toString()
if (endLoop.toString() == "T" || endLoop.toString()=="True"
|| endLoop.toString()=="true")
{
log.info ("Exit Groovy Data Source Looper")
assert true
}
else
{
testRunner.gotoStepByName("Groovy Script - DataSource") //go to the DataSource
}

```

11. In questa procedura di esempio, viene aggiunto un ritardo di 1000 ms tra ciascun loop. Questo passaggio è facoltativo. Con il ritardo sono ora disponibili cinque passaggi del test:

The screenshot displays the JMeter interface. On the left, the Navigator tree shows a project named 'Test' containing a 'QuotaSoap11' folder with various test elements and a 'TestSuite 1' folder containing 'TestCase 1'. 'TestCase 1' has five test steps: 'Groovy Script - DataSource', 'Property - Looper (5)', 'Test Request', 'Delay [1000]', and 'Groovy Script - Data Loop'. The right pane shows the 'TestSteps' list for 'TestCase 1' with the same five steps. Below the TestSteps list are tabs for 'Description', 'Properties', 'Setup Script', and 'TearDown Script'. At the bottom, the 'Test Properties' table is visible.

Property	Value
Name	TestCase 1

at java.net.PlainSocketImpl.connectToAddress(Unknown Source)
at java.net.PlainSocketImpl.connect(Unknown Source)
at java.net.SocksSocketImpl.connect(Unknown Source)

12. Fare clic sul pulsante **Esegui** verde per eseguire i cinque passaggi del test nella finestra TestCase.