

TestApp by Unicus

Konsol och schemalagd körning

Författare: Joakim Odermalm

Dokumentversion: 1.0, 190626

Innehåll

Installation.....	2
Kortfattad hjälp.....	2
Lista tester och senaste resultat.....	3
Filter-alternativ.....	3
Köra tester.....	4
Rapportera resultat genom email.....	4
Konfiguration.....	5
Schemalagd körning.....	7

Installation

Hämta

Paketet finns (för närvarande) på <http://joakim.dreamhosters.com/testrunnerapp/console/>

Installera

Ladda ner senaste versionen och zippa upp den på lämplig plats.

Inställningar för test management

För närvarande stödjer konolversionen inte managment-integration.

Kortfattad hjälp

Nedan är hjälp-texten så som den ser ut i version 2.2.1. För att se denna hjälp i ev. uppdaterad version, använd kommandot

```
TestRunnerApp help
```

Genomgående förklaringar återfinns senare i detta dokument.

```
Help for TestRunnerApp console
```

```
=====
```

```
Usage: TestRunnerApp Example.testapp command params
```

```
Available commands:
```

```
list [filter options]
```

```
run [filter options]
```

```
mail /to 'email' [filter options]
```

```
Filter:
```

```
by last outcome: /o [Pass] [Fail] [Warning] [NotRun]
```

```
by test id: /id 'regEx'
```

```
Change testdata before run:
```

```
/d TestId#TestDataIndex:NewTestData
```

```
Note: New testdata cannot contain forward slash.
```

```
Ex: /d T1#1:foo T1#2:bar T2#1:far T2#2:'b o o'
```

Lista tester och senaste resultat

`list [filter options]`

Ex. - Lista alla tester i svitfilen MyTests

```
TestRunnerApp MyTests.testapp list
```

Det är möjligt att filtrera vilka tester som visas, se avsnittet **Filter-alternativ**.

Filter-alternativ

Det går att filtrera listning, körning och resultatrapportering på senaste utfall eller test-id.

Utfall

Efter kommandot, använd flaggan **/o** och en efterföljande lista med utfall som ska visas. Alternativen är:

Pass Fail Warning NotRun

Ex. 1 – lista tester i svitfilen MyTests som aldrig körts

```
TestRunnerApp MyTests.testapp list /o NotRun
```

Id

Efter kommandot, använd flaggan **/id** och ett reguljärt uttryck som matchar de id som ska visas.

Ex. 2 – lista tester i svitfilen MyTests med id 1.

```
TestRunnerApp MyTests.testapp list /id '^1$'
```

Ex. 3 – lista tester i svitfilen MyTests med id 1 tom. 9.

```
TestRunnerApp MyTests.testapp list /id '^\d$'
```

Kombinera

Används båda är filtreringen additiv, dvs. båda villkoren måste vara uppfyllda.

Ex. 4 – lista tester i svitfilen MyTests med id 1 tom. 9 eller 30 tom. 49, och vars senaste utfall var *Fail* eller *Warning*.

```
TestRunnerApp MyTests.testapp list /o Fail Warning /id '^(\\d|(3|4)\\d)$'
```

RegEx

Här är en bra resurs för att lära sig mer om reguljära uttryck: <https://regex101.com/>

Jmf. även <https://xkcd.com/208/>

Köra tester

Körning

run [filter options]

Ex. - Kör alla tester i svitfilen MyTests

```
TestRunnerApp MyTests.testapp run
```

Det är möjligt att filtrera vilka tester som körs, se avsnittet **Filter-alternativ**.

Redigera testdata

Det är möjligt att ändra testdata innan körning, te.x. för att skripta flera körningar av samma test med olika testdata. För närvarande finns begränsningen att testdatan inte kan innehålla framåtlutande snedstreck: /

Change testdata before run:

```
/d TestId#TestDataIndex:NewTestData
```

Note: New testdata cannot contain forward slash.

Ex: /d T1#1:foo T1#2:bar T2#1:far T2#2:'b o o'

Välja test-cykel

För närvarande stödjer konsolverisoner inte testcykler.

Välja web-driver

För närvarande kan man inte välja web-driver från kommandoraden, detta val görs istället i konfigurationsfilen (se nedan).

Rapportera resultat genom email

mail /to 'email' [filter options]

Ex. – Skicka rapport med alla tester i svitfilen MyTests

```
TestRunnerApp MyTests.testapp mail /to 'recepient@customer.foo'
```

Det är möjligt att filtrera vilka tester som rapporteras, se avsnittet **Filter-alternativ**.

Konfiguration

I konfigurationsfilen **TRAC.cfg** finns möjlighet att ange inställningar.

Kolumner för rapport

Inställning för vilka kolumner som visas vid listning och mailrapport sätts med **columns="kolumn1 kolumn2 etc"**

Ex.

columns="id name previousDateTime previousOutcome webDriverType failStep message eType"

Följande kolumner kan användas:

Kategori	Namn	Beskrivning
Om testet	id	Test-id
Om testet	name	Testnamn / beskrivning
Om testet	totalRuns	Totalt antal körningar
Om testet	estimatedTime	Förväntad körningstid
Om senaste körning	previousDateTime	Starttid för körning
Om senaste körning	webDriverType	Använd webdriver
Om senaste körning	runNotes	Anteckningar om körning
Om senaste resultat	previousOutcome	Utfall
Om senaste resultat	previousRuntime	Total körningstid
Om senaste resultat	failStep	Steg där fel uppstod
Om senaste resultat	message	Resultat-meddelande
Om senaste resultat	eType	Undantags-typ från resultat
Om senaste resultat	eMessage	Undantags-detaljer från resultat

Webdriver

Ex.

webDriverType="Firefox"

Giltiga val: **Firefox Chrome**

Mail-rapportering

```
### Mail settings ###
smtpHost="smtp.gmail.com"
smtpPort="587"
smtpTimeout="10000"
enableSsl="True"
useDefaultCredentials="False"
sendFrom=""
subject="TestApp by Unicus report"
sendFromAccount=""
sendFromAccountPw=""
```

Management-integration

Obs. att konsolversionen för närvarande inte stödjer management-integration, dessa inställningar användas alltså inte.

```
### Managment settings ###  
exportAfterExecution="True"  
createNewCycle="False"
```

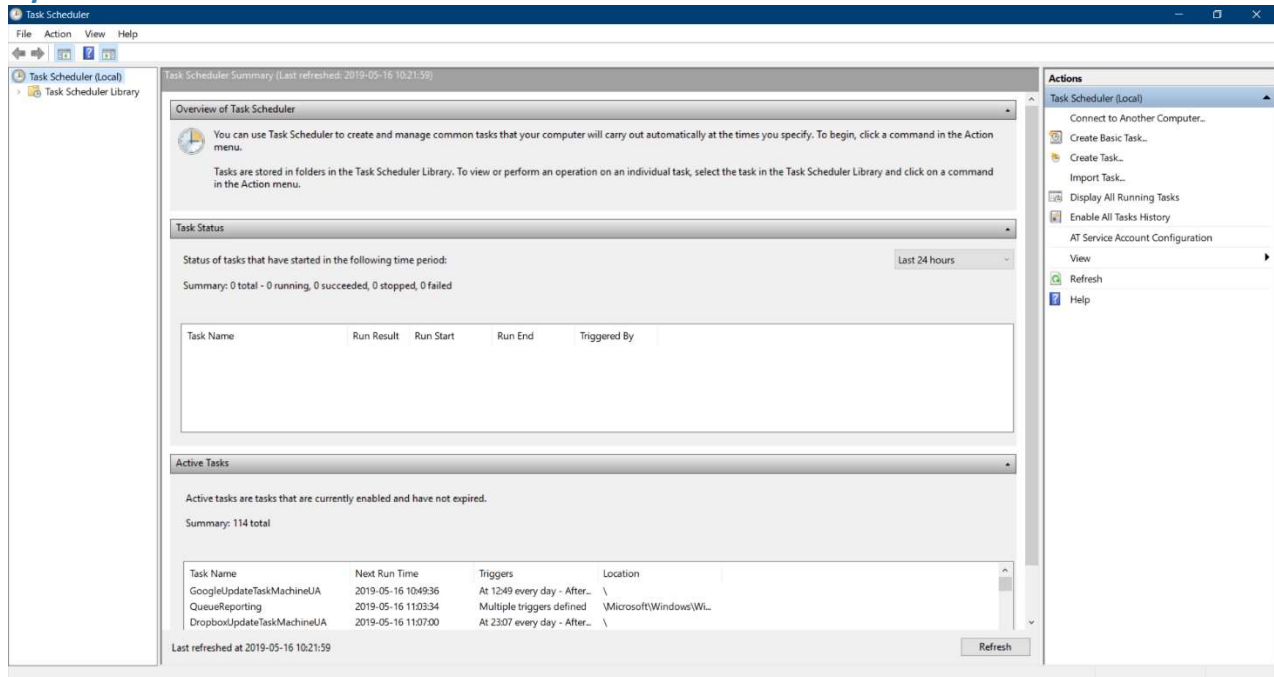
```
### Jira Cloud with Adaptavist TM4J ###  
jiraCloudInstance=""  
jiraCloudUser=""  
jiraCloudKey=""  
tmjCloudToken=""
```

Schemalagd körning

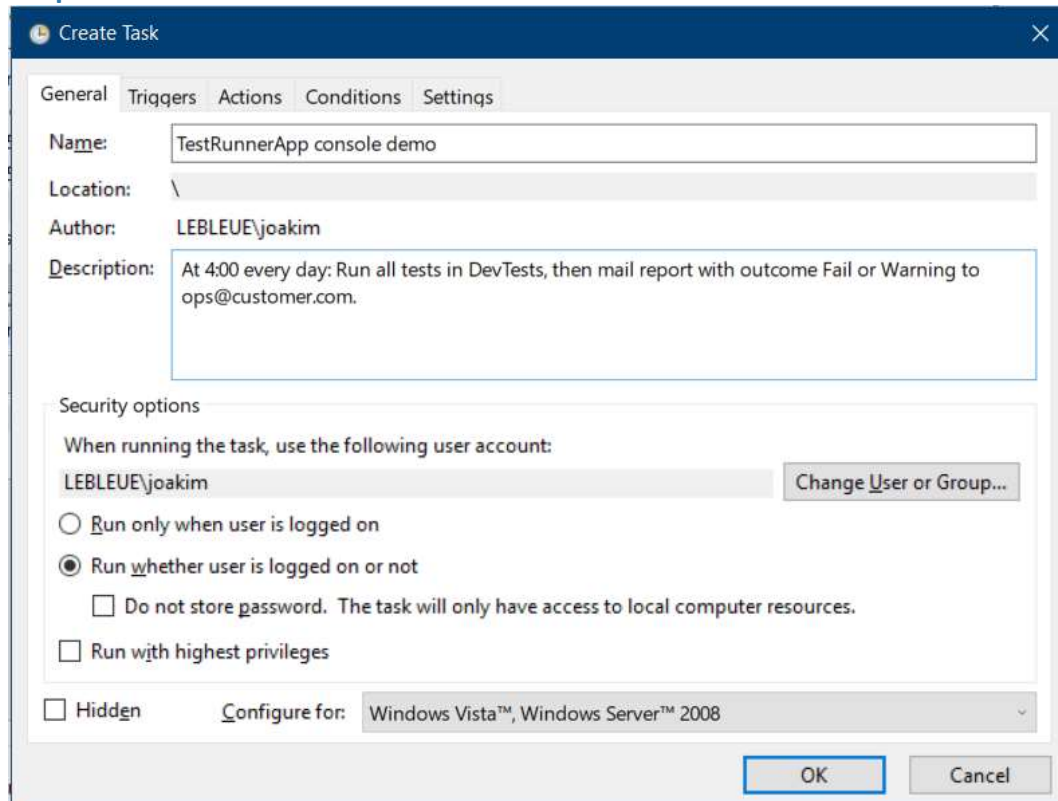
I detta avsnitt visas ett exempel på hur man kan sätta upp återkommande schemalagda körningar mha. det inbyggda Windows-verktyget **Task Scheduler**. Många andra alternativ finns.

I detta exempel sätter vi upp en schemalagd körning som varje dag kl 04:00 kör alla tester i svitfilen DevTests och sedan mailar en rapport med alla tester som fick utfallet **Fail** eller **Warning** till ops@customer.com.

Vy vid start



Skapa en task



Ange när vår task ska köras (trigger)

The 'New Trigger' dialog box is shown with the following settings:

- Begin the task:** On a schedule
- Settings:**
 - ☐ One time
 - ☒ Daily
 - ☐ Weekly
 - ☐ Monthly
- Start:** 2019-05-16, 04:00:00
- ☐ Synchronize across time zones
- Recur every:** 1 days
- Advanced settings:**
 - ☐ Delay task for up to (random delay): 1 hour
 - ☐ Repeat task every: 1 hour for a duration of: 1 day
 - ☐ Stop all running tasks at end of repetition duration
 - ☒ Stop task if it runs longer than: 30 minutes
 - ☐ Expire: 2020-05-16, 11:33:16
 - ☐ Synchronize across time zones
 - ☒ Enabled

Buttons: OK, Cancel

Ange vad som ska utföras (action)

Bilden visar den första av våra två actions.

Obs. att **Start in** måste fyllas i med mappen där konsolversionen av TestRunnerApp finns.

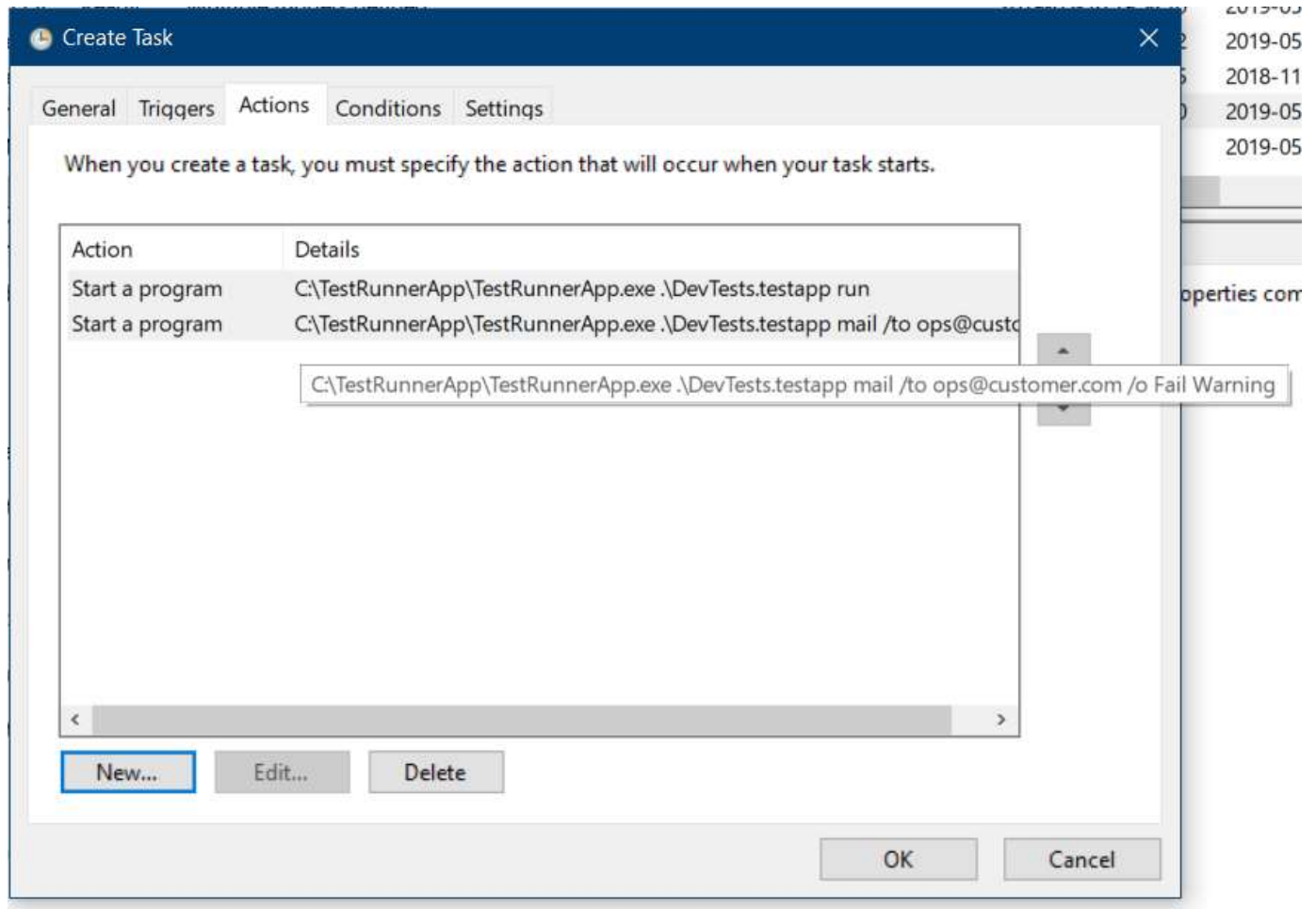
The 'New Action' dialog box is shown with the following settings:

- Action:** Start a program
- Settings:**
 - Program/script:** C:\TestRunnerApp\TestRunnerApp.exe
 - Add arguments (optional):** .\DevTests.testapp run
 - Start in (optional):** C:\TestRunnerApp\

Buttons: OK, Cancel

Kontrollera och bekräfta

Här har båda våra två actions lagts till, se tooltip på bilden för kommandoradsalternativen som lagts in i action #2.



Task Scheduler-vy med inlagd schemakörning

