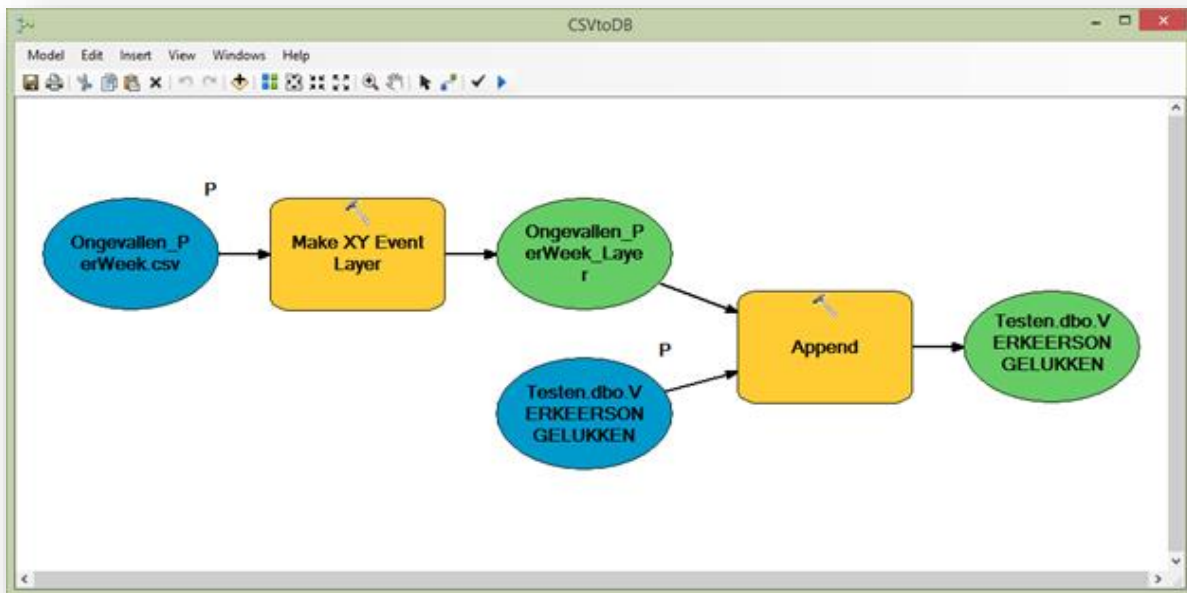


# Tools, tips, tricks: run tool run

Daar ga je weer, denk je misschien... Elke dag weer data importeren op dezelfde manier, dat moet toch makkelijker kunnen?! Het maken van ModelBuilder-modellen helpt hier natuurlijk bij, maar we willen meer. We willen het compleet automatisch! Waarschijnlijk heeft iedere gebruiker of beheerder wel één taak die net zo goed door een machine uitgevoerd kan worden. In deze Tools, Tips & Tricks: run tool run.

## Wat heb je nodig?

Allereerst heb je een model nodig waarin de taak is verwerkt, bijvoorbeeld het inlezen van een csv in een feature class.



Om dit model te automatiseren zijn er nog een paar andere zaken nodig:

- Een Python-script van het model of een Python-script om het model aan te roepen.
- Administrator-rechten om een taak in de Task Scheduler (Taakplanner) van Windows in te regelen.

Een model kun je niet direct als taak aan de Task Scheduler van Windows toevoegen, daarvoor heb je een Python-script nodig. Deze is zelf te maken, door het model te exporteren naar een Python-script of je maakt een script die het model aanroept. Ook kun je gebruik maken van het bijgevoegde generieke script om eenvoudig modellen en tools aan te roepen.

## Model exporteren naar Python

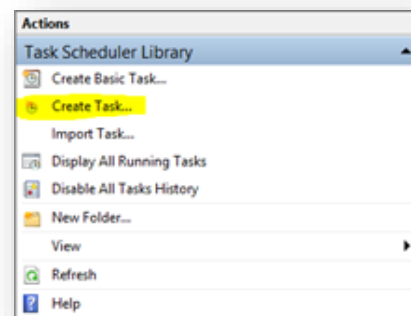
Als je het model exporteert naar een Python-script wordt er een script aangemaakt waarin alle tools uit het model worden aangeroepen. Je kunt kiezen of je de argumenten (in het script wilt opnemen of bij de taak wilt vermelden. Dit script kun je vervolgens gebruiken om als taak toe te voegen aan de Task Scheduler.

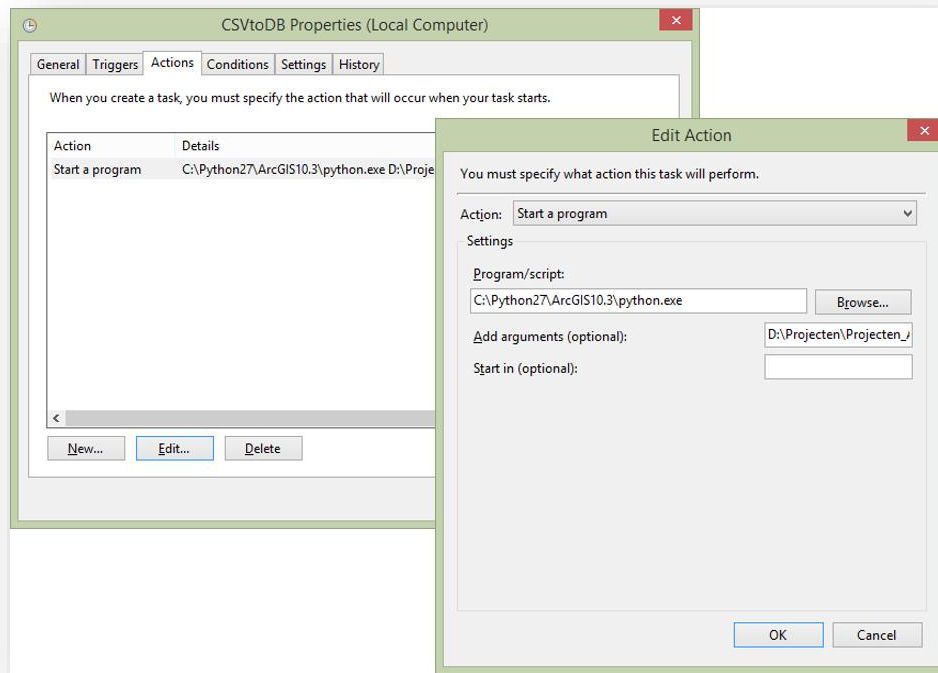
```
1 # -*- coding: utf-8 -*-
2 #
3 # CSVtoDB.py
4 # Created on: 2016-01-28 09:51:53.00000
5 # (generated by ArcGIS/ModelBuilder)
6 # Description:
7 #
8 #
9 # Import arcpy module
10 import arcpy
11
12 # Local variables:
13 Ongevallen_PerWeek_csv = "D:\\Projecten\\Projecten_ArcGIS\\Intern\\TTT_Arcpy\\Ongevallen_PerWeek.csv"
14 Ongevallen_PerWeek_Layer = "Ongevallen_PerWeek_Layer"
15 Testen_dbo_VERKEERSONGELUKKEN_3_ = Ongevallen_PerWeek_Layer
16 Testen_dbo_VERKEERSONGELUKKEN = "Database Connections\\Connection to Testen@esribx5020.sde\\Testen.dbo.VERKEERSONGELUKKEN"
17
18 # Process: Make XY Event Layer
19 arcpy.MakeXYEventLayer_management(Ongevallen_PerWeek_csv, "X", "Y", Ongevallen_PerWeek_Layer, "PROJCS['RD_New',GEOGCS[
20 'GCS_Amersfoort',DATUM['D_Amersfoort',SPHEROID['Bessel_1841',6377397.155,298.1528128]],PRIMEM['Greenwich',0.0],UNIT[
21 'Degree',0.0174532925199433]],PROJECTION['Double_Stereographic'],PARAMETER['False_Easting',155000.0],PARAMETER[
22 'False_Northing',463000.0],PARAMETER['Central_Meridian',5.38763888888889],PARAMETER['Scale_Factor',0.9999079],PARAMETER[
23 'Latitude_Of_Origin',52.1561605555555],UNIT['Meter',1.0]];-30515500 -30279500 10000;-100000 10000;-100000 10000;0.001;0
24 .001;0.001;IsHighPrecision", "")
25
26 # Process: Append
27 arcpy.Append_management("Ongevallen_PerWeek_Layer", Testen_dbo_VERKEERSONGELUKKEN, "NO_TEST", "Soort_ongeluk \
28 Soort_ongeluk\" true true false 100 Text 0 0 ,First,#,Ongevallen_week1_Layer,Soort,-1,-1;Betrokken_voertuigen \
29 Betrokken_voertuigen\" true true false 100 Text 0 0 ,First,#,Ongevallen_week1_Layer,Voertuigen,-1,-1;Aantal_slachtoffers
30 Aantal_slachtoffers\" true true false 2 Short 0 5 ,First,#,Ongevallen_week1_Layer,Aantal_slachtoffers,-1,-1;Lichtgewond
31 Lichtgewond\" true true false 2 Short 0 5 ,First,#,Ongevallen_week1_Layer,Lichtgewond,-1,-1;Zwaargewond \"Zwaargewond\"
32 true true false 2 Short 0 5 ,First,#,Ongevallen_week1_Layer,Zwaargewond,-1,-1;Opmerkingen \"Opmerkingen\" true true false
33 250 Text 0 0 ,First,#,Ongevallen_week1_Layer,Opmerkingen,-1,-1", "")
34
```

Met de Task Scheduler kun je taken op bepaalde tijdstippen automatisch laten uitvoeren, op je eigen computer of op de server. Zoek onder Windows naar Schedule Task om de Task Scheduler te openen. Het toevoegen van een script aan de Task Scheduler gaat eenvoudig. Klik op Create Task om een nieuwe taak aan te maken en vul de schermen naar wens in.

Let er wel op dat je naar het juiste Python-programma verwijst (ArcGIS for Desktop of ArcGIS for Server) en dat de argumenten met een spatie gescheiden zijn. Het eerste argument is altijd het pad naar het script. Mocht je argument spaties bevatten, dan kun je het tussen quotes zetten (denk bijvoorbeeld aan database connecties). Onder 'Start in' zet je de locatie van de logfiles.

Dit voorbeeld heeft geen argumenten, hier hoeft je dus alleen het pad naar het script neerzetten.





## Eigen script

Een gevolg van het exporteren van een model naar Python is dat je bij elke aanpassing het model opnieuw moet exporteren en dat gaat ten koste van het automatiseren. Om dit op te lossen, kun je ook een script maken om het model aan te roepen, net zoals je een standaard tool in ArcPy aanroept.

Het maken van een eigen script om een model aan te roepen gaat als volgt: (eventueel kun je het voorbeeld hieronder gebruiken). In het script importeer je de toolbox en voeg je het model toe, zoals je ook een standaard tool zal toevoegen, inclusief argumenten. In het voorbeeld hieronder is ook logging toegevoegd om eventuele problemen tijdens het uitvoeren van het model te registreren.

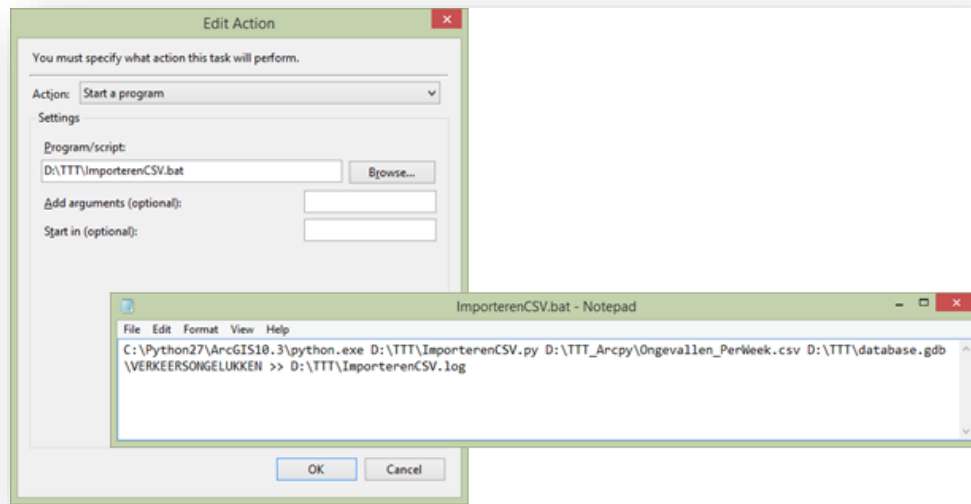
Als je dit script direct aan de Task Scheduler toevoegt, heb je niet de mogelijkheid om de verkregen logging weg te schrijven. Om dit op te lossen, kun je gebruik maken van een batchbestand (.bat). Dit is een tekstbestand waarin je alle argumenten wegschrijft, inclusief het pad naar Python. In de Task Scheduler verwijst je vervolgens naar het batchbestand in plaats van het Python-programma.

```

1 # importeer de benodigde modules
2 import arcpy
3 import sys
4
5 # Bepaal de argumenten
6 Ongevallen_PerWeek = sys.argv[1]
7 Verkeersongelukken = sys.argv[2]
8
9 # Importeer de toolbox
10 arcpy.ImportToolbox(r"D:\TTT\Importereren.tbx")
11 print arcpy.GetMessages()
12
13 # Voer het model uit
14 arcpy.CSVtoDB( Ongevallen_PerWeek, Verkeersongelukken)
15 print arcpy.GetMessages()
16

```

De argumenten moeten ook hier op de juiste volgorde geplaatst worden. Het gedeelte '>> D:\TTT\ImportererenCSV.log' verwijst naar het logbestand.



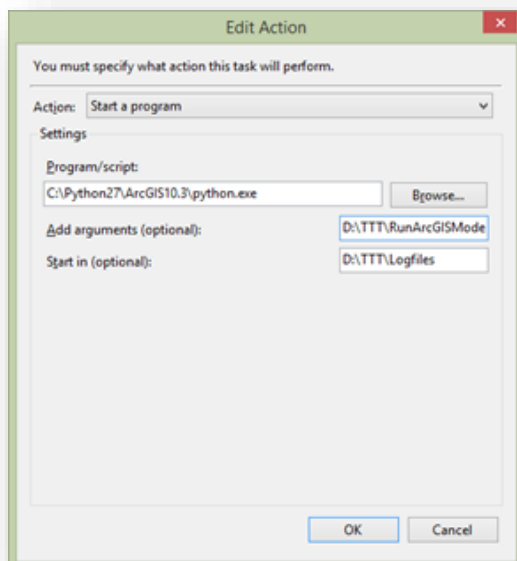
## Generiek script

Het generieke script kun je gebruiken om alle soorten tools of modellen te automatiseren. Het enige wat je moet doen is bepalen hoeveel argumenten het model gebruikt. Op basis hiervan geef je de argumenten op in de Task Scheduler. Zorg er wel voor dat je de argumenten in de juiste volgorde zet. Dit kun je achterhalen door het model te openen en te zien welke argumenten wanneer gevraagd worden.

Vervolgens kun je het script toevoegen als taak in de Task Scheduler en de argumenten opgeven. De eerste drie argumenten zijn het pad naar het script, het pad naar de toolbox, en de naam van de uit te voeren tool, gevolgd door de argumenten van de aan te roepen tool.

De argumenten:

D:\TTT\RunArcGISModel.py D:\TTT\Importereren.tbx CSVtoDB D:\TTT\Ongevallen\_PerWeek.csv  
"Database Connections\Connection to Testen@server.sde\Testen.dbo.VERKEERSONGELUKKEN"



Zodra je een taak hebt toegevoegd, kun je deze testen door deze handmatig uit te voeren. Werkt je taak zoals verwacht? Gefeliciteerd! Je hebt de taak geautomatiseerd.

```
1 # -----
2 # RunToolRun.py
3 # Usage: RunToolRun
4 # -----
5 # Import system modules
6 import sys, logging, datetime, importlib, arcpy
7
8 toolbox_pad = sys.argv[1]
9 toolbox_model = sys.argv[2]
10
11 # Local variables
12 vandaag = datetime.datetime.now().strftime("%Y%m%d%H%M%S")
13 logfile = toolbox_model + "_" + vandaag + ".log"
14 print "Logfile: " + logfile
15
16 # Initialize logging
17 logging.basicConfig(level=logging.DEBUG,format='%(asctime)s %(levelname)s %(message)s',filename=logfile,filemode='w')
18
19 try:
20     # Laden van de toolbox (eventueel gerefereerde toolboxes worden automatisch meegeladen)...
21     logging.info("Importeren toolbox: " + toolbox_pad)
22     toolbox_alias = "mijnalias"
23     arcpy.ImportToolbox(toolbox_pad,toolbox_alias)
24     arcpy.OverwriteOutput = 1
25
26     # Process: Model uitvoeren...
27     logging.info("==== Uitvoeren model gestart =====")
28
29     # Ophalen juiste functie op basis van de parameters
30     module = importlib.import_module('arcpy')
31     function_name = toolbox_model + '_' + toolbox_alias
32
33     logging.info("Starten functie: " + function_name)
34
35     function = getattr(module, function_name)
36
37     # Functie uitvoeren
38     logging.info(sys.argv)
39     if len(sys.argv) > 3:
40         function("sys.argv[3:]")
41     else:
42         function()
43
44     logging.info(arcpy.GetMessages(0))
45     logging.info("SCRIPT SUCCESVOL AFGEROND")
46 except Exception as e:
47     logging.error(str(sys.exc_info()[0]))
48     logging.error(arcpy.GetMessages(0))
49     logging.error("FOUT OPGETREDEN")
50
```

Het generiek script is beschikbaar op GitHub: <https://github.com/esrinederland/Tools-Tips-and-Tricks/blob/gh-pages/RunToolRun.py>

## Aanvullende informatie

[Lees meer over Scheduled Tasks](#)

Heeft u vragen? Of heeft u nog een tip voor een tool? E-mail naar [contact@esri.nl](mailto:contact@esri.nl).

Het is mogelijk om een training te volgen, ga daarvoor naar [het complete opleidingsaanbod](#).