

Infrastructure as Code

by J{a}son Bourne

Denis Buco

Microsoft Deutschland GmbH

Agenda

- DevOps
- ARM Templates
- Azure Automation
- PowerShell DSC

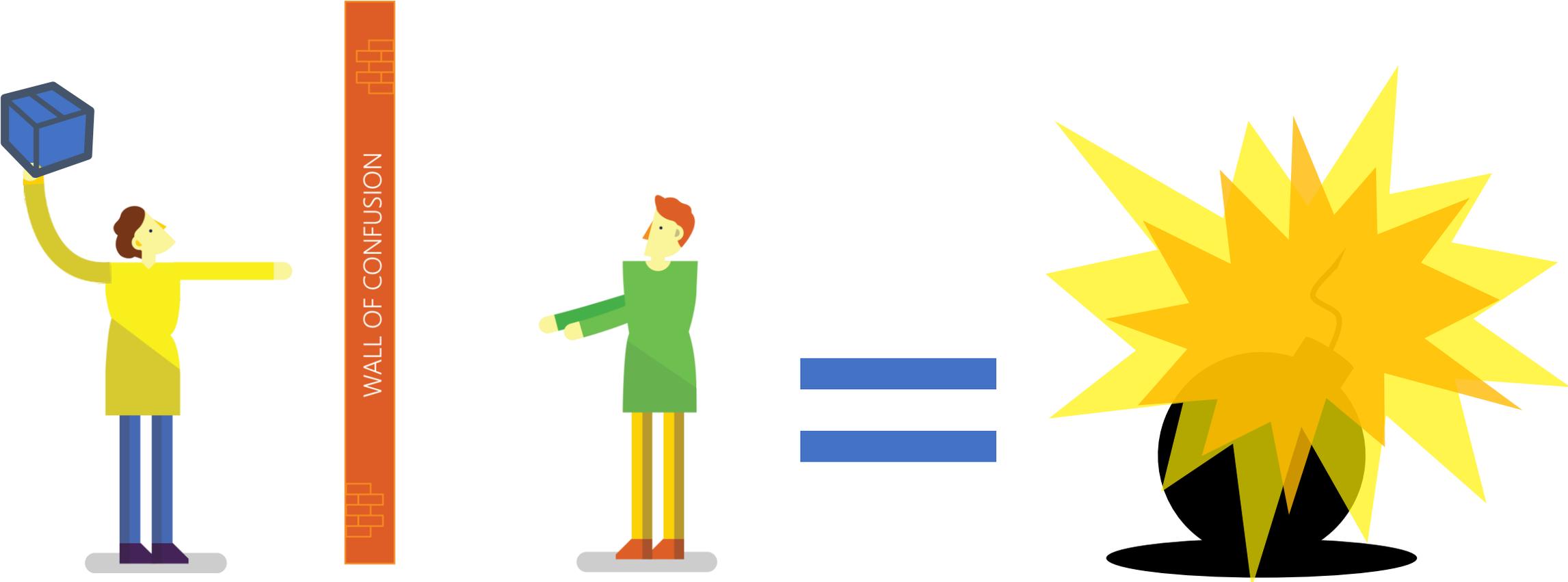


DevOps

DevOps beschreibt einen Prozessverbesserungs-Ansatz aus den Bereichen der Softwareentwicklung und Systemadministration. *DevOps* ist ein Kunstwort aus den Begriffen Development (englisch für *Entwicklung*) und IT Operations (englisch für IT-Betrieb). *DevOps* soll durch gemeinsame Anreize, Prozesse und Werkzeuge (englisch: *Tools*) eine effektivere und effizientere Zusammenarbeit der Bereiche *Dev*, *Ops* und Qualitätssicherung (QS) ermöglichen.^[1] Mit *DevOps* sollen die Qualität der Software, die Geschwindigkeit der Entwicklung und der Auslieferung sowie das Miteinander der beteiligten Teams verbessert werden.
<https://de.wikipedia.org/wiki/DevOps>

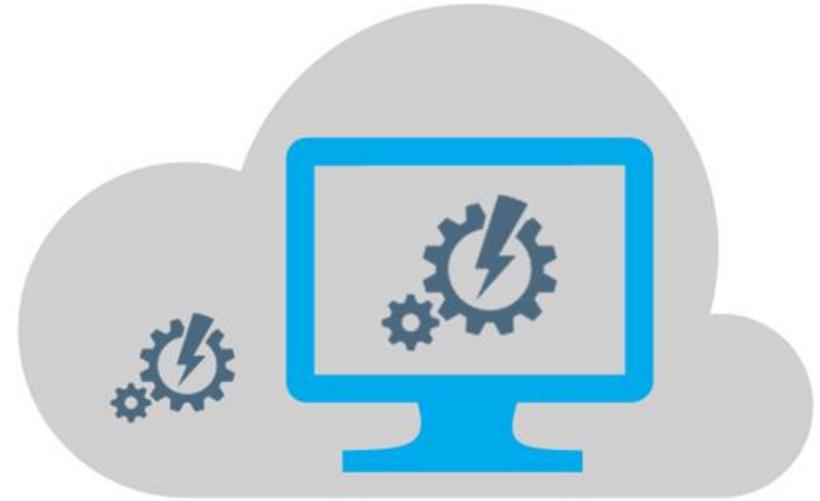


Traditional Development and Operations



DevOps

- Automated Build
- Automated Testing
- Automated Deployment
- Automated Monitoring
- Automated Infrastructure Provisioning
- Self Healing Infrastructure

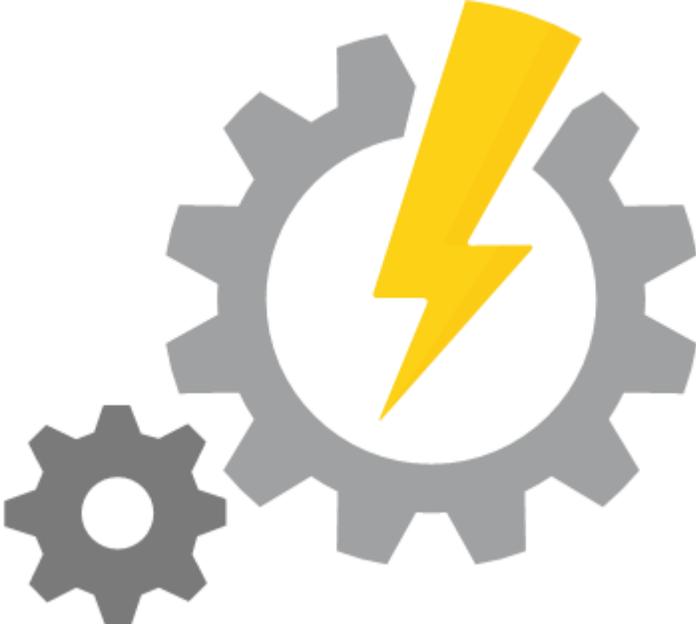
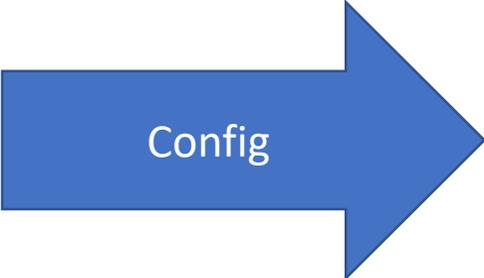


ARM Templates

- Parameter
- Variablen
- Ressourcen

```
1 {
2   "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": {
5     "autoscalesettings_free_htdemo_name": {
6       "type": "string",
7       "defaultValue": "free-htdemo"
8     },
9     "components_htms_name": {
10      "type": "string",
11      "defaultValue": "htms"
12    },
13    "autoscalesettings_free_htdemo_metricResourceUri": {
14      "type": "string",
15      "defaultValue":
16        "/subscriptions/a48a924d-6007-4c39-a3c0-5466b9012f42/resourcegroups/htdemo/providers/Microsoft.
17        Web/serverfarms/free"
18    },
19    "autoscalesettings_free_htdemo_metricResourceUri_1": {
20      "type": "string",
21      "defaultValue":
22        "/subscriptions/a48a924d-6007-4c39-a3c0-5466b9012f42/resourcegroups/htdemo/providers/Microsoft.
23        Web/serverfarms/free"
24    },
25    "autoscalesettings_free_htdemo_targetResourceUri": {
26      "type": "string",
27      "defaultValue":
28        "/subscriptions/a48a924d-6007-4c39-a3c0-5466b9012f42/resourcegroups/htdemo/providers/Microsoft.
29        Web/serverfarms/free"
30    }
31  },
32  "variables": {},
33  "resources": [
34    {
35      "comments": "Generalized from resource:
36        /subscriptions/a48a924d-6007-4c39-a3c0-5466b9012f42/resourceGroups/htdemo/providers/microsoft.
```

Azure Automation



Virtual machine



Virtual machine



Virtual machine

Azure Automation

Account

- Einen **Automation Account** für Production, Development, On-Premise (max. 30)

Assets

- Assets sind Variablen, Credentials, Zertifikate, Connections, welche im RunBook verwendet werden können.

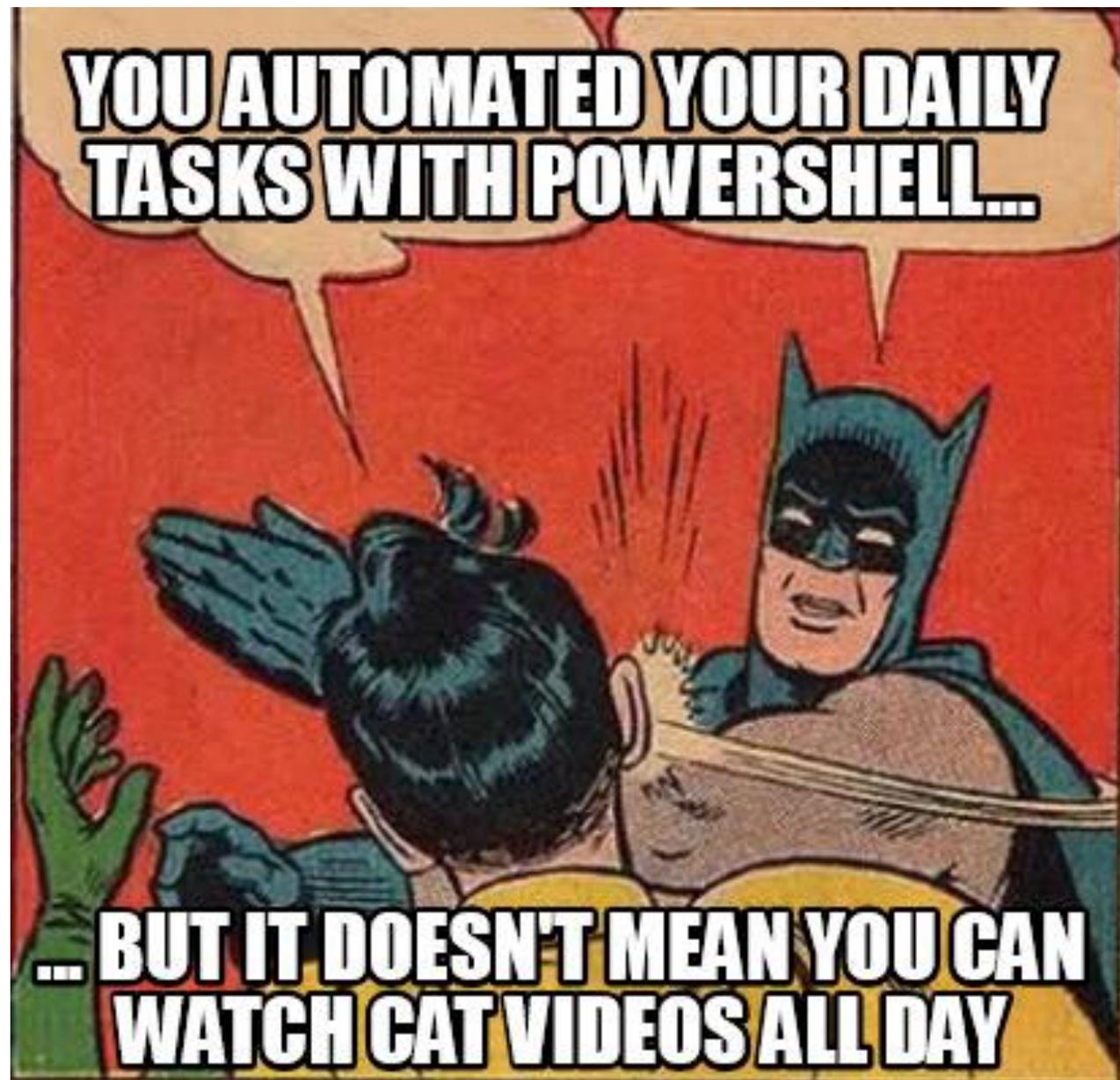
Source Control

- Codeverwaltung über GitHub und Visual Studio Online (bald verfügbar)

Manage

- Runbooks können für Azure, OnPremise und Amazon Web Services (AWS) Ressourcen verwendet werden

PowerShell
DSC



PowerShell - Desired State Configuration (DSC)

- Definiere “**WAS** soll gemacht werden?”. Das “**WIE** wird es gemacht?“, wird vom System erledigt.
- Push / Pull Ansatz
 - (Push) Mit “Start-DscConfiguration” wird die Konfiguration (MOF-Datei) auf die jeweiligen Zielsysteme kopiert und ausgeführt.
 - (Pull) Die Zielsysteme holen sich von einem Pull-Server die notwendigen Informationen/Konfiguration.
- Voraussetzung PowerShell ab V4 (Windows Server 2012 R2 Windows 8.1)
- Verwendet ‘Local Configuration Manager’ und ‘Windows Workflow Manager’



Wer? Wie? Was?

- Resources
 - Powershell Module (Get-DscConfiguration)
 - X / C Module = Experimental / Community
- Configuration
 - Definition „Was soll passieren?“ -> MOF Datei
- Node
 - Ziel (Server auf dem die Konfiguration ausgeführt wird)
- Local Configuration Manager
 - Ausführende Komponente auf dem jeweiligen Node



PowerShell(DSC)

PowerShell

```
# BitLocker Drive Encryption Service

$service = get-service | select-object Name, Status |
where-object {$_.Name -eq "BDESVC" -and $_.Status -eq 'Running'}

If (!$service){
  Start-Service -Name "BDESVC"
}
```

PowerShell DSC

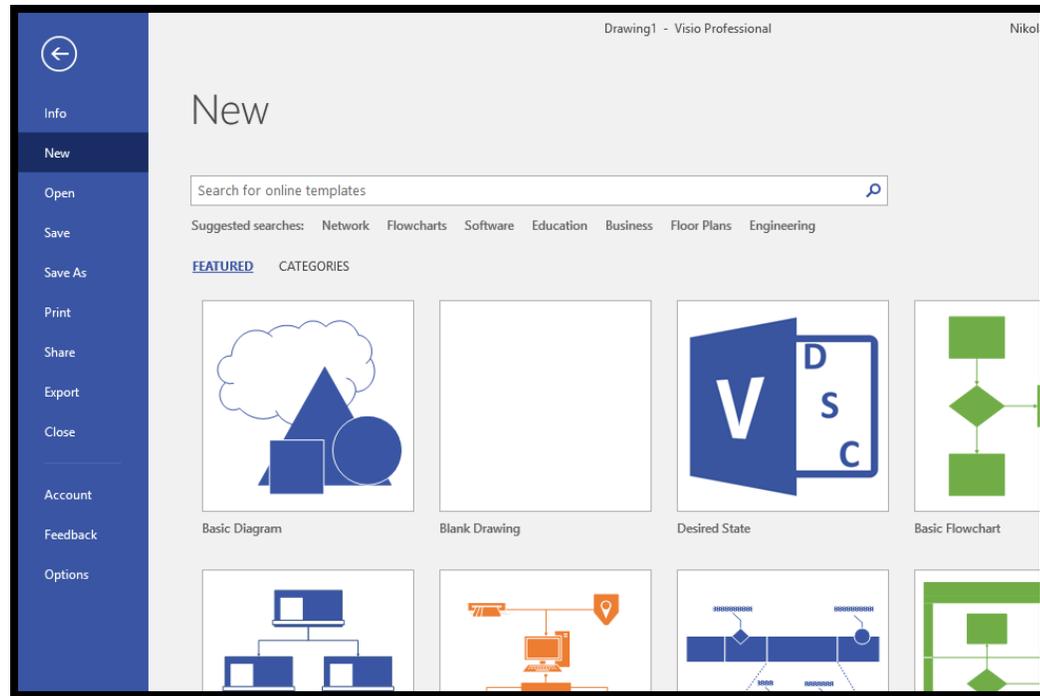
```
Service BDESVC {
  Name = 'BDESVC'
  State = 'Running'
}
```



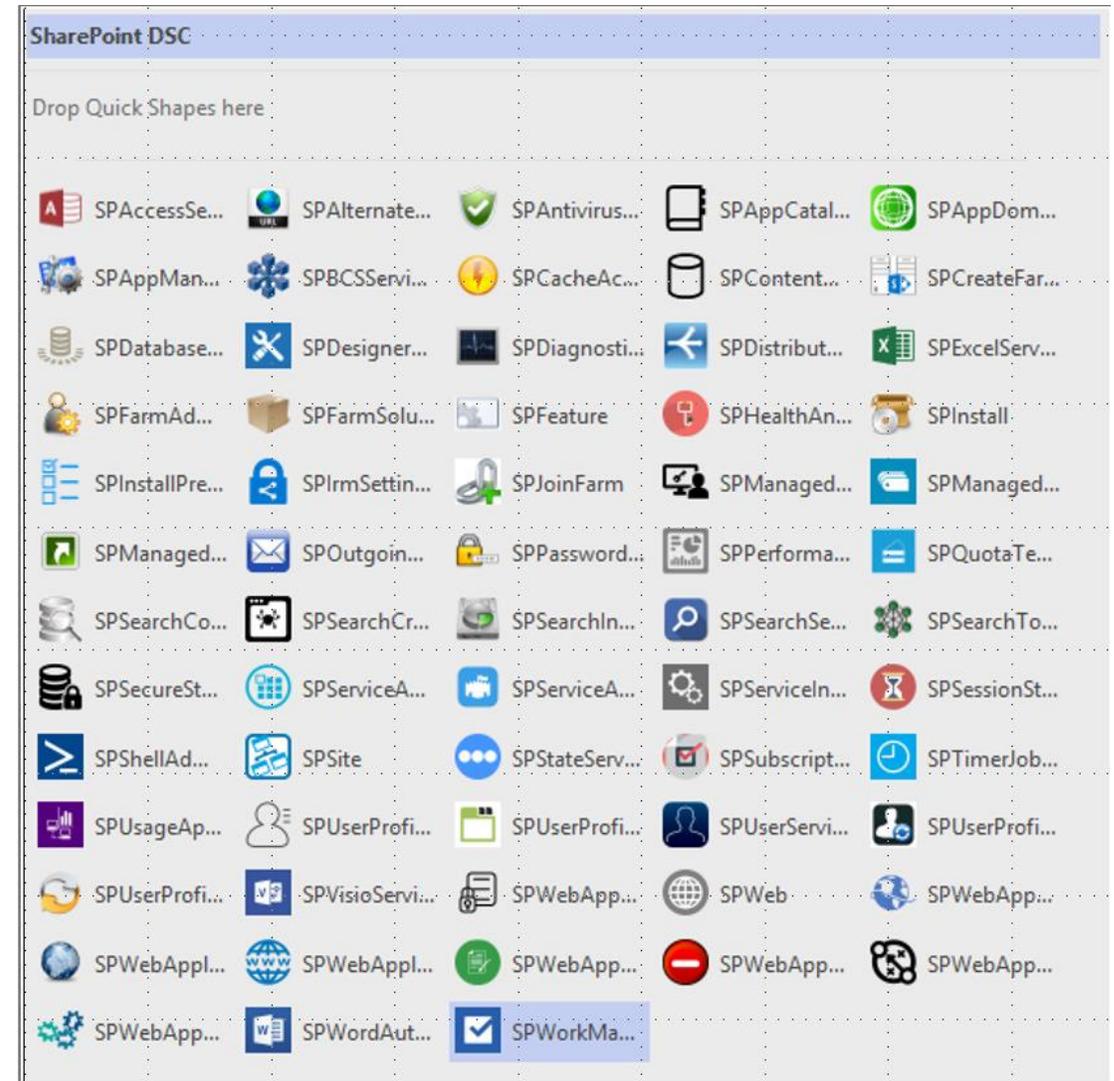
Reverse DSC!

- Wie sieht eigentlich meine aktuelle Umgebung aus?
- Get-TargetResource -> exportiert meine aktuelle Config als DSCScript
- Kopie der aktuellen Umgebung erstellen -> MOVE2AZURE

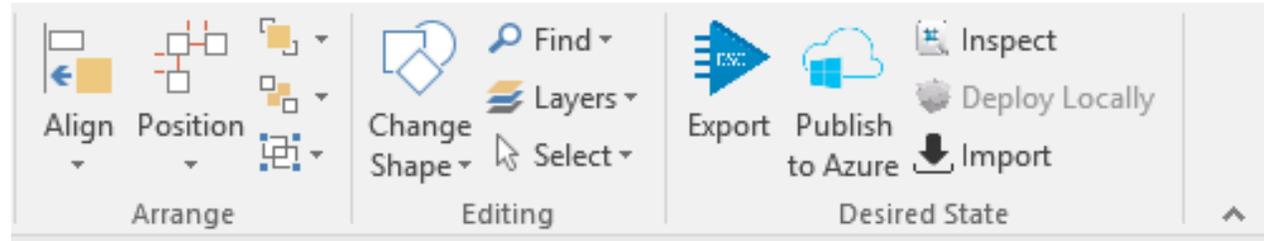
VisualDSC



Visio 2013 und 2016



VisualDSC



The image shows the VisualDSC interface. On the left, a diagram of a SPWebApplication is displayed on a grid. The diagram consists of a central blue globe with a network of white lines and nodes, surrounded by a square grid of eight white nodes. Below the diagram, the text "SPWebApplication" is written in a large, bold, black font. On the right, a window titled "DSC Inspector" is open, showing the DSC configuration code for the SPWebApplication. The code is as follows:

```
SPWebApplication Nik
{
    AllowAnonymous = $FALSE
    ApplicationPoolAccount = "contoso\sp_farm"
    UseSSL = $FALSE
    AuthenticationMethod = "NTLM"
    ApplicationPool = "SharePoint-80"
    Name = "Nik"
    PsDscRunAsCredential = $farmAccountCreds
}
```