



PowerShell Conference Europe

# Mastering Azure using Cloud Shell, PowerShell and Bash!

*@ThomasMaurer*

*[www.thomasmaurer.ch](http://www.thomasmaurer.ch)*



# Many thanks to our sponsors:





# Thomas Maurer

- Senior PM Azure Hybrid @Microsoft
- Former Microsoft MVP
- Blogger: [www.thomasmaurer.ch](http://www.thomasmaurer.ch)
- PowerShell Enthusiast

# Tools for Managing Azure

## Management tools

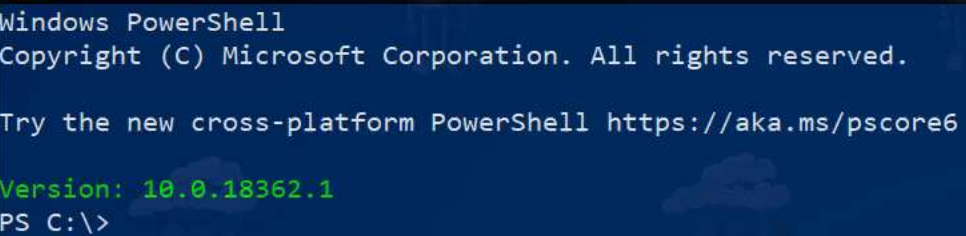
- Azure Portal
- Azure CLI
- Azure PowerShell
- Azure Cloud Shell



## Automation tools

- ARM templates
- Azure Automation
- Azure Functions





# A primer on automation

## Why

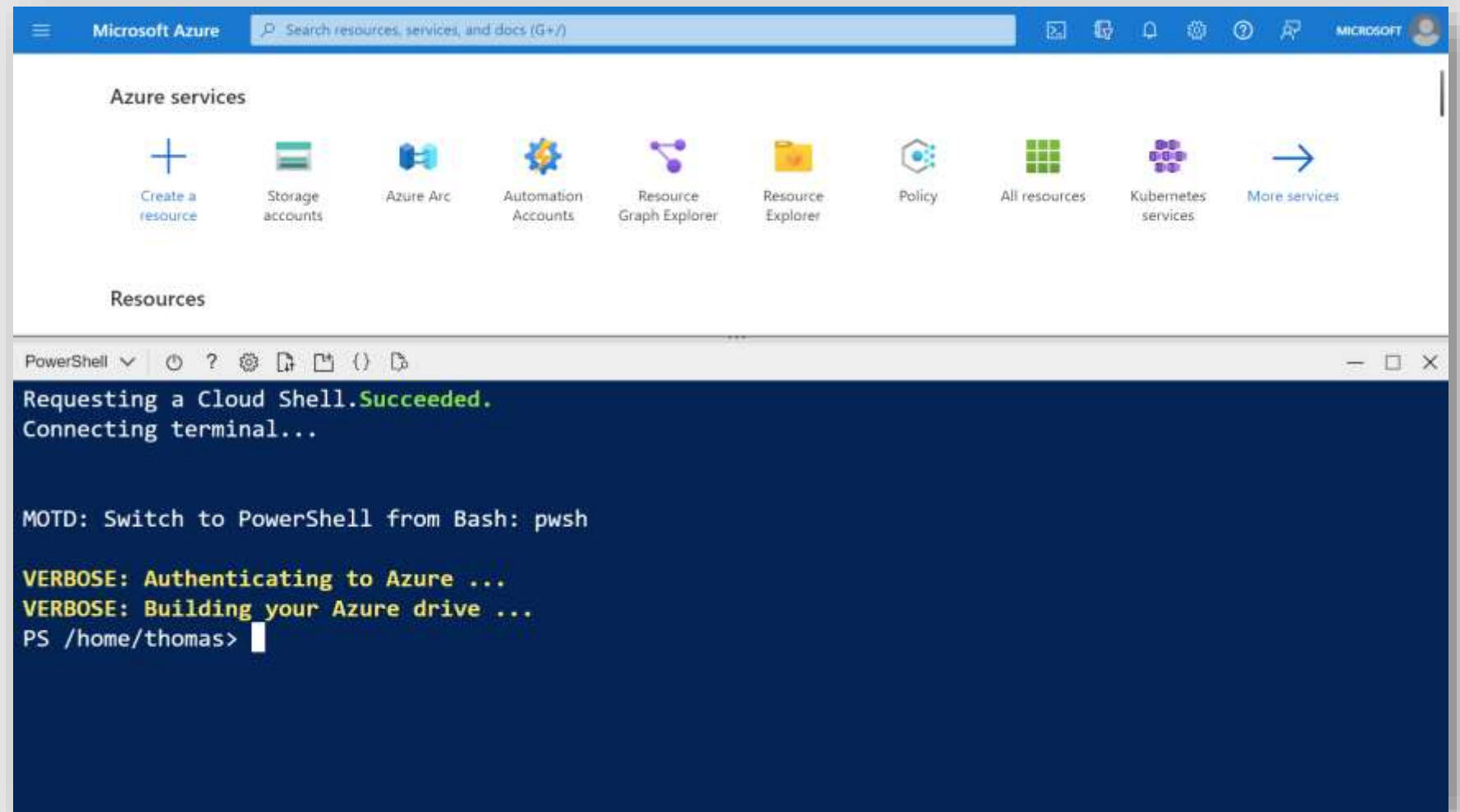
- Save time
- Lower overhead cost
- Consistency

## When

- Resources at scale
- Configuration
- Orchestration



# Cloud Shell



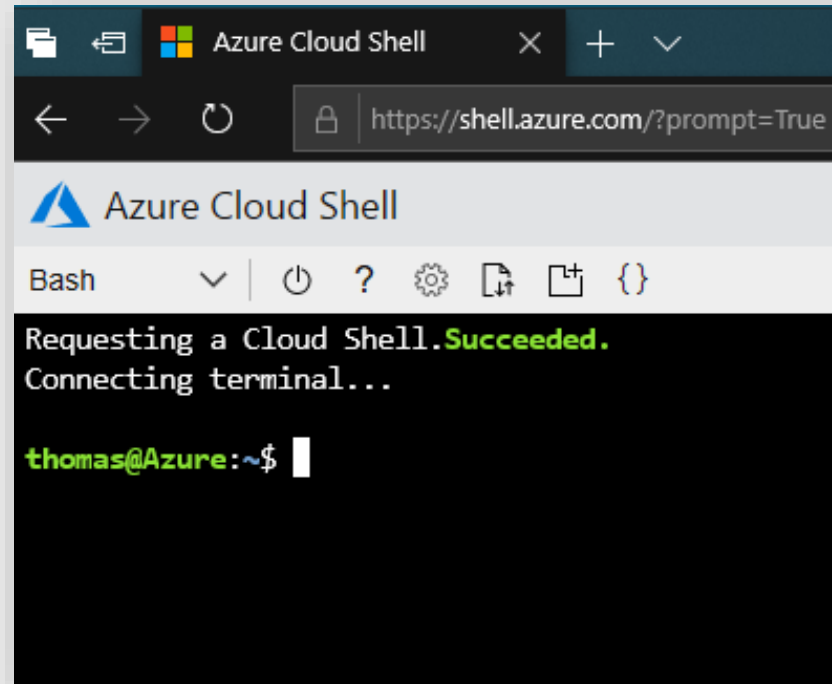
# Azure Cloud Shell

Authenticated access

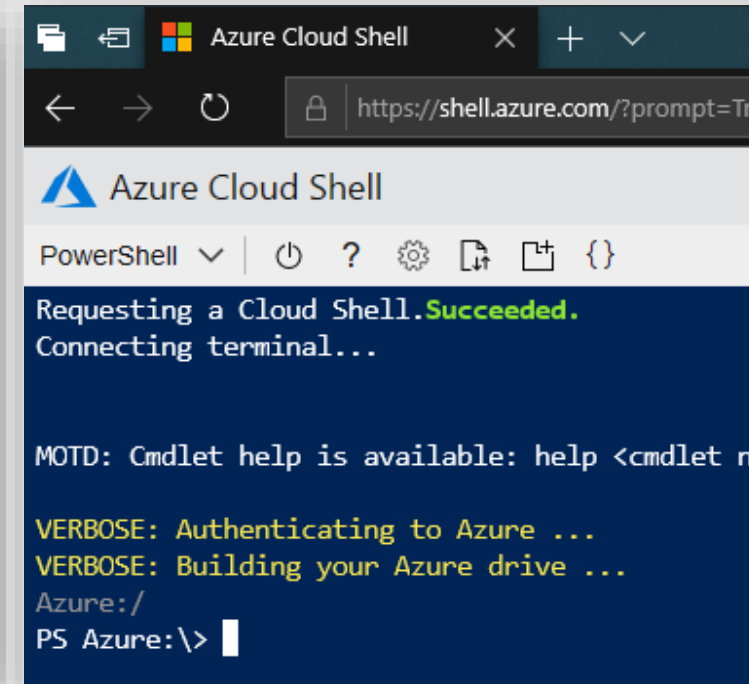
Bash & PowerShell

Private & secure environment

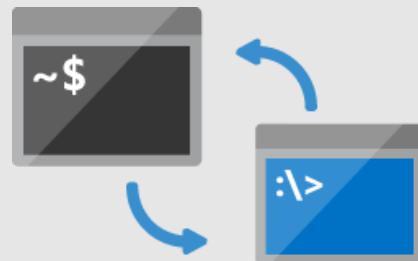
Common languages & tools



```
Azure Cloud Shell
https://shell.azure.com/?prompt=True
Bash
Requesting a Cloud Shell.Succeeded.
Connecting terminal...
thomas@Azure:~$
```



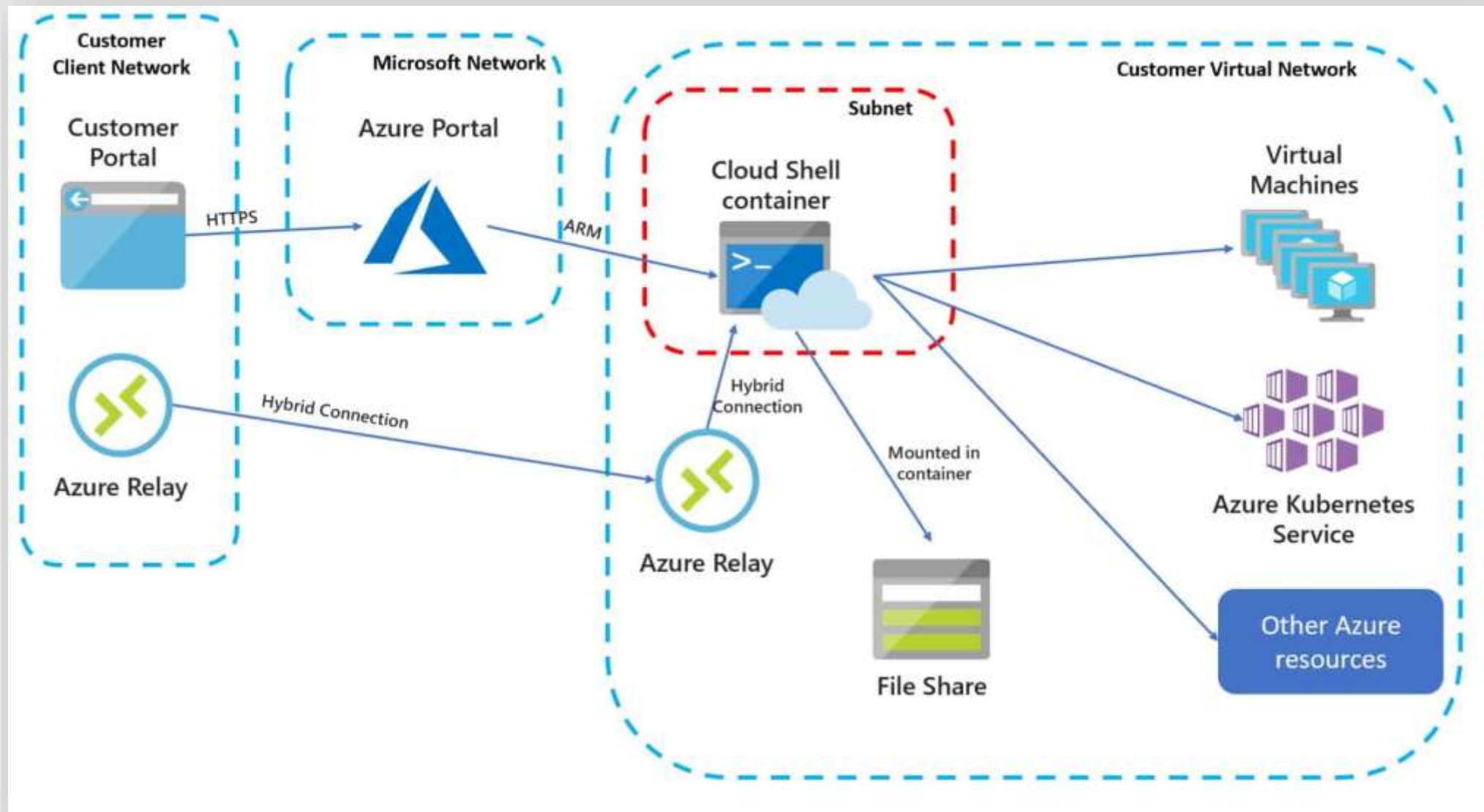
```
Azure Cloud Shell
https://shell.azure.com/?prompt=True
PowerShell
Requesting a Cloud Shell.Succeeded.
Connecting terminal...
MOTD: Cmdlet help is available: help <cmdlet n
VERBOSE: Authenticating to Azure ...
VERBOSE: Building your Azure drive ...
Azure:/
PS Azure:\>
```





# Exploring the Azure Cloud Shell

# Connect Azure Cloud Shell to Virtual Network vNet



# Multiple Experiences

[Azure portal](#)

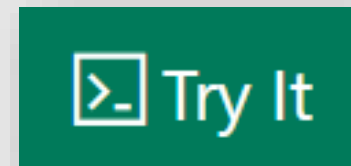
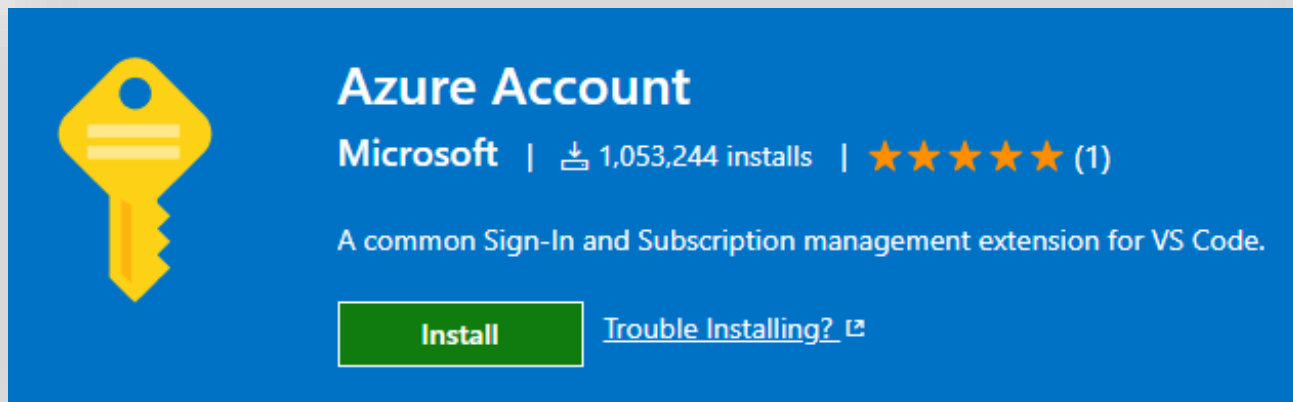
[Shell.azure.com](#)

[Azure Extension](#) in Visual Studio Code

Windows Terminal

[“Try It” integration](#) in docs.microsoft.com

[Azure Mobile App](#)





GatherUp!

14:51 | 19°

07:20



To-Do β



Spotify



Training - GSC ...



OneNote



Azure



Microsoft



Clock



Calculator



Hue



Messenger



LinkedIn



Slack



Skype



Travel



Teams



Twitter



Instagram



Facebook



Gallery



Play Store

Search



**Az Predictor**

# Invoke Command PowerShell and Azure CLI

## PowerShell:

```
Set-PSReadLineOption -PredictionViewStyle ListView
```

## Azure CLI:

```
Set-PSReadLineOption -PredictionViewStyle InlineView
```

# How to Run Scripts in your Azure VM using Run Command

# Invoke Command PowerShell and Azure CLI

## PowerShell:

```
Invoke-AzVMRunCommand -ResourceGroupName 'TT-AZUREVMS-RG' -Name 'ttazurewinvm01' -CommandId 'RunPowerShellScript' -ScriptPath 'C:\Users\thoma\script.ps1'
```

## Azure CLI:

```
az vm run-command invoke -g myResourceGroup -n myVm --command-id RunShellScript --scripts "sudo apt-get update && sudo apt-get install -y nginx"
```



```

AzVMRunCommand-multi.ps1 • script.ps1
AzVMRunCommand-multi.ps1 >
4 $resourceGroup = "test-azurevms-rg"
5
6 #Select the right Azure subscription
7 Set-AzContext -Subscription $subscriptionId
8
9 #Get all Azure VMs which are in running state and are running Windows
10 $myAzureVMs = Get-AzVM -ResourceGroupName $resourceGroup -status | Where-Object {$_.PowerState -eq "VM running" -and $_.StorageProfile.OSDisk.OSType -eq "Windows"}
11
12 #Run the script again all VMs in parallel
13 $myAzureVMs | ForEach-Object -Parallel {
14     $out = Invoke-AzVMRunCommand
15         -ResourceGroupName $_.ResourceGroupName
16         -Name $_.Name
17         -CommandId 'RunPowerShellScript'
18         -ScriptPath .\script.ps1
19     #Formating the Output with the VM name
20     $output = $_.Name + " " + $out.Value[0].Message
21     $output
22 }
    
```

```

#Get all Azure VMs which are in running state and are running Windows
$myAzureVMs = Get-AzVM -ResourceGroupName $resourceGroup -status | Where-Object {$_.PowerState -eq "VM running" -and $_.StorageProfile.OSDisk.OSType -eq "Windows"}

#Run the script again all VMs in parallel
$myAzureVMs | ForEach-Object -Parallel {
    $out = Invoke-AzVMRunCommand
        -ResourceGroupName $_.ResourceGroupName
        -Name $_.Name
        -CommandId 'RunPowerShellScript'
        -ScriptPath .\script.ps1
    #Formating the Output with the VM name
    $output = $_.Name + " " + $out.Value[0].Message
    $output
}
    
```

Name	Account	SubscriptionName	Environment	TenantId
ms-cxa- azvm001 HostName	@microsoft.com	ms-cxa-	AzureCloud	111
AMS211020210034				
azvm003 HostName				
AMS060101070555				

**SSH to Arc enabled Servers**

# SSH Remoting with Azure Arc

```
az ssh arc --resource-group tt-azurearcvms-rg --name  
ttubuntu02 --local-user thomas
```

# PowerShell in Windows Admin Center in the Azure Portal

**Want to learn more?**

# Blogs and Links

[aka.ms/azops-cloudshell](https://aka.ms/azops-cloudshell)



## MASTERING AZURE WITH CLOUD SHELL

POSTED IN [CLOUD](#), [MICROSOFT](#), [MICROSOFT AZURE](#), [POWERSHELL](#), [VISUAL STUDIO](#), [VISUAL STUDIO CODE](#), [WORK](#)

There are multiple ways to interact and manage resources in [Microsoft Azure](#). You can use the Azure Portal or command line tools like the Azure PowerShell module or the Azure CLI, which you can install on your local machine. However, to set up a cloud management workstation for administrators and developers can be quite a lot of work. Especially if you have multiple machines, keeping consistency between these machines can be challenging. Another challenge is keeping the environment secure and all the tools up to date. This and many more things are addressed by the Cloud Shell.

Cloud Shell is not brand new, [Microsoft announced Cloud Shell](#) at Build 2017. This [blog](#) post is about how you can master Azure with Cloud Shell and to give you an overview about the possibilities of Cloud Shell.

## What is Cloud Shell



# Q&A

15 minutes



# Demos

We love Demos!

