

Automating SCOM Tasks Using SMA



Microsoft MVP Community Camp 2015 Melbourne

Tao Yang

MVP: System Center Cloud and Datacenter Management

tyang@tyang.org

Microsoft®
System Center
About Myself

- System Center Cloud and Datacenter Management (SCCDM) MVP
- Heavily focused on System Center Operations Manager (SCOM)
- Also working in SCCM, Orchestrator, SMA, VMM, SCSM, Hyper-V etc.
- Management Pack Author
 - OpsMgr Self Maintenance Management Pack
 - ConfigMgr 2012 (R2) Client Management Pack
 - SCOM Maintenance Mode Scheduler Management Pack
- Communities
 - Blog: <http://blog.tyang.org>
 - Cross Blog: <http://www.systemcentercentral.com>
 - Twitter: @MrTaoYang



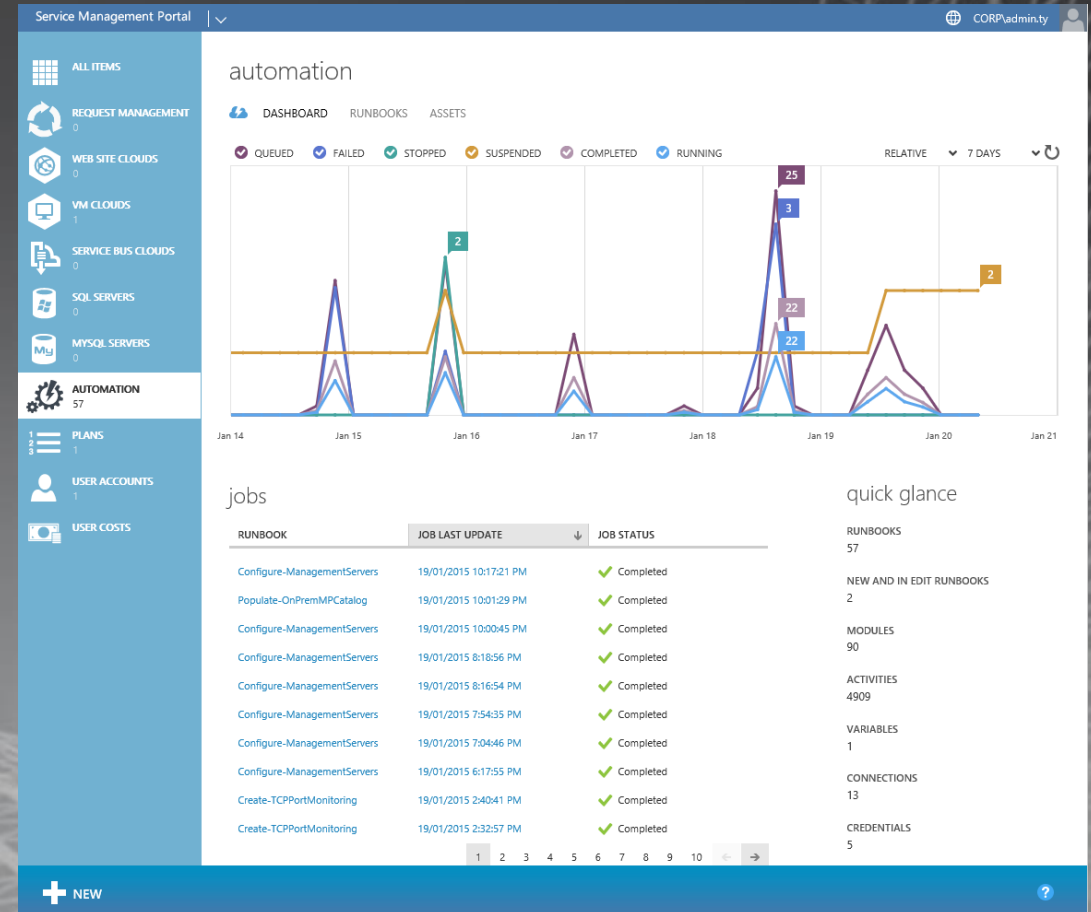
Agenda

- Short SMA Introduction
- Challenges in SCOM Automation
- SCOM Automation Using SDK
- Lots of Demos
 - Copy Management Group Settings
 - Create New Management Pack
 - Create Performance Monitor
 - Populate a MP List on SharePoint
 - Copy MP (Between Management Groups)
 - Create TCP Port Monitoring
- Q & A



What is Service Management Automation (SMA)

- New Member of the System Center 2012 R2 family
- Automation engine based on PowerShell Workflow
- Integrated with WAP Admin Portal (Under Automation Tab)
- 64-bit PowerShell
- Using Existing PowerShell Modules



SMA & Orchestrator – Better Together?

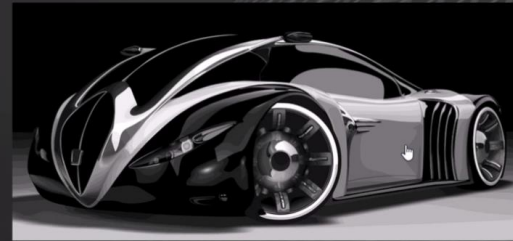
- Good Analogy from SCCDM MVPs Michael Rüefli and Markus Klein in their 2014 SCU Europe Presentation

SC ORCHESTRATOR, SMA ->?

SC Orchestrator



SMA

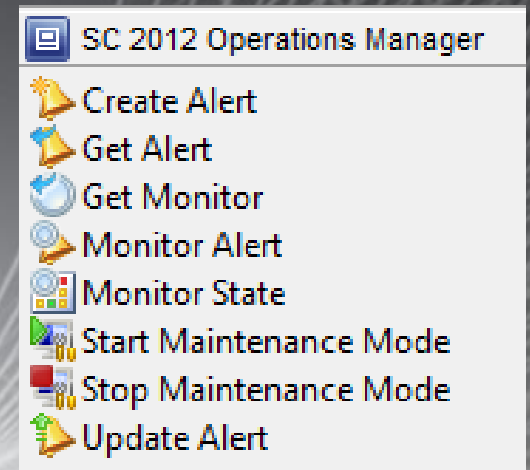


Better together ?



SCOM Automation - Challenges with Orchestrator

- Can't really do much with only 8 activities!
- Must install SCOM Operations Console of the same version on the runbook designer computer and the runbook server
- What if I'm running both SCOM 2007 and 2012?
 - Cannot mix SCOM components from different versions
 - Must use a separate runbook server for SCOM 2007 runbooks.



SCOM Automation - Challenges in SMA

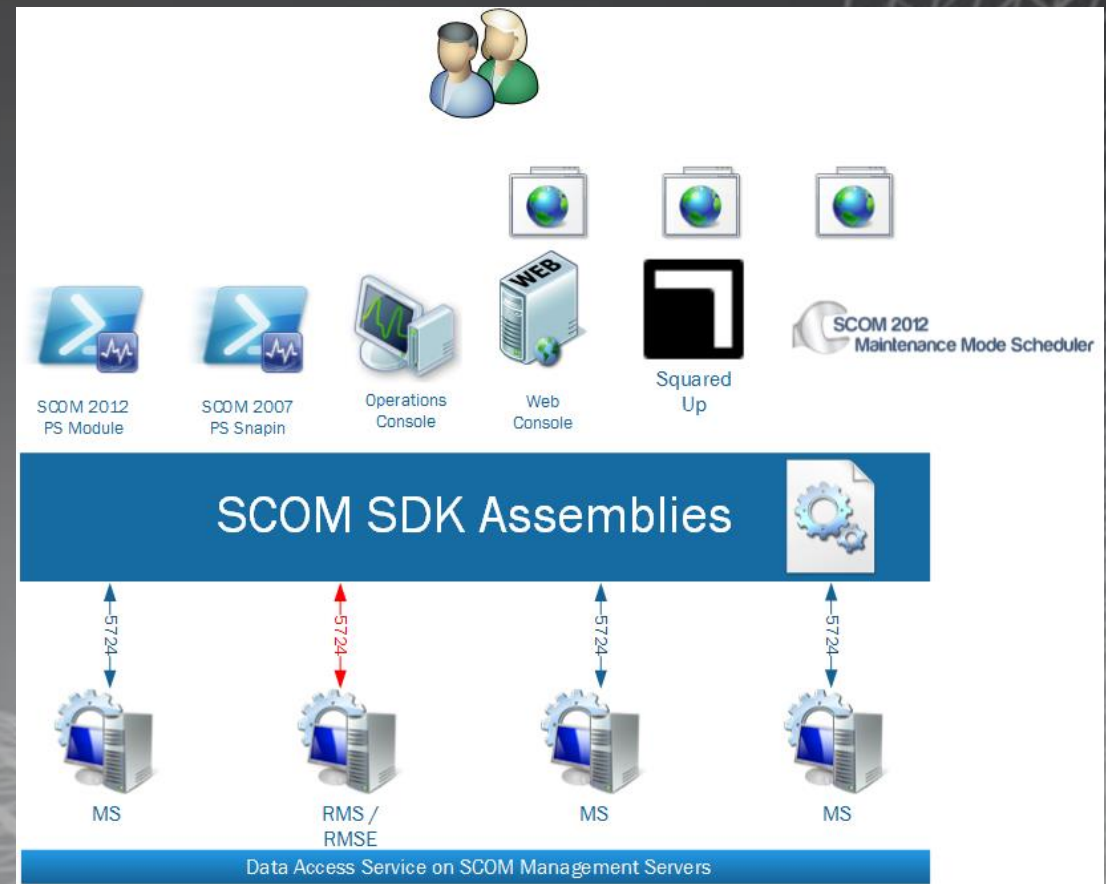
- OperationsManager module is a Portable Module
 - Proxying commands to the original OperationsManager module
 - Requires the original OperationsManager module to be installed on all Runbook Workers

```
try { $params = $PSBoundParameters; OperationsManager\Export-SCOMManagementPack @params } catch {  
$message = '' + $env:COMPUTERNAME + ' does not have the OperationsManager module installed.  
Either this cmdlet must be called from an inlinescript against a machine with the  
OperationsManager module installed, or you must install the OperationsManager module on ' + $env:  
COMPUTERNAME; if($_.FullyQualifiedErrorId -eq "CouldNotAutoLoadModule") { throw $message } else {  
throw $_ } }
```

- SMA today is lacking monitoring activities
 - i.e. monitor alerts,
 - monitor SharePoint list items, etc.

Scripting in SCOM (Overview)

- All SCOM user interfaces use SDK to communicate with management servers
- SCOM 2012 SDK:
 - Microsoft.EnterpriseManagement.Core.dll
 - Microsoft.EnterpriseManagement.OperationsManager.dll
 - Microsoft.EnterpriseManagement.Runtime.dll
- SDK DLLs are installed into the Global Assembly Cache (GAG) with:
 - Management Server
 - Operations Console
 - Web Console
- SDK DLLs are also located on Mgmt Servers:
 - “<SCOM Install Dir>\Server\SDK Binaries” folder
- What can I do with the SDK assemblies?
 - **EVERYTHING!!**



Scripting in SCOM (Loading SDK)

- PowerShell can leverage the SCOM SDK DIRECTLY!
- Loading SDK Assemblies from Global Assembly Cache (GAC):

```
#Load SCOM SDK DLLs from GAC
[System.Reflection.Assembly]::Load("Microsoft.EnterpriseManagement.Core, Version=7.0.5000.0, Culture=neutral, PublicKeyToken=31bf3856ad364838")
[System.Reflection.Assembly]::Load("Microsoft.EnterpriseManagement.OperationsManager, Version=7.0.5000.0, Culture=neutral, PublicKeyToken=31bf3856ad364838")
[System.Reflection.Assembly]::Load("Microsoft.EnterpriseManagement.Runtime, Version=7.0.5000.0, Culture=neutral, PublicKeyToken=31bf3856ad364838")
```

- Loading SDK DLLs from a folder:

```
#Load SCOM SDK DLLs from a folder
[System.Reflection.Assembly]::LoadFrom("C:\Temp\Microsoft.EnterpriseManagement.Core.dll")
[System.Reflection.Assembly]::LoadFrom("C:\Temp\Microsoft.EnterpriseManagement.OperationsManager.dll")
[System.Reflection.Assembly]::LoadFrom("C:\Temp\Microsoft.EnterpriseManagement.Runtime.dll")
```

Scripting in SCOM (Connecting to MG)

- Define MG Connection Setting

```
#Define Connection Setting
$ConnectionSetting = New-Object Microsoft.EnterpriseManagement.ManagementGroupConnectionSettings("OpsMgrMS01")

#Optionally, specify username and password
$UserName = "Domain\SCOMAdmin"
$Password = ConvertTo-SecureString -AsPlainText "password1234" -force
$ConnectionSetting.UserName = $Username
$ConnectionSetting.Password = $Password
```

- Connect to MG and Get MG Admin object

```
#Connect to SCOM Management Group
$MG = New-Object Microsoft.EnterpriseManagement.ManagementGroup($ConnectionSetting)

#Get Management Group Administration
$Admin = $MG.Administration
```

Scripting in SCOM (SDK Tips)

- Use the ManagementGroup instance (\$MG) to:
 - Retrieve Management Group information
 - Get and create Management Pack elements (classes, rules, monitors, discoveries, tasks, module types, monitor types), and much more.
- Use the AdministrationManagement object (\$Admin) to perform administrative tasks such as:
 - Configure Management Group settings
 - Add or delete agents / management servers, etc.
- SCOM SDK Fully documented on MSDN:
 - <http://msdn.microsoft.com/en-us/library/hh329086.aspx>
- Many good examples from OpsMgr Self Maintenance MP
 - <http://blog.tyang.org/2014/06/30/opsmgr-2012-self-maintenance-management-pack-2-4-0-0/>

SMA / PowerShell Module: OpsMgrExtended

- Place DLLs in the PS Module folder
- No need to install any SCOM components
- Can be used as:
 - SMA Integration Module
 - Normal PowerShell module
- Cannot use native SCOM module cmdlets

```
Function Import-OpsMgrSDK
{
    <#...#>
    #OpsMgr 2012 R2 SDK DLLs
    $DLLPath = (Get-Module OpsMgrExtended).ModuleBase

    $arrDLLs += 'Microsoft.EnterpriseManagement.Core.dll'
    $arrDLLs += 'Microsoft.EnterpriseManagement.OperationsManager.dll'
    $arrDLLs += 'Microsoft.EnterpriseManagement.Runtime.dll'
    $DLLVersion = '7.0.5000.0'
    $PublicKeyToken = '31bf3856ad364e35'

    #Load SDKs
    $bSDKLoaded = $true
    Foreach ($DLL in $arrDLLs)
    {
        $AssemblyName = $DLL.TrimEnd('.dll')
        #try load from GAC first
        Try {
            Write-Verbose "Trying to load $AssemblyName from GAC..."
            [Void][System.Reflection.Assembly]::Load("$AssemblyName, Version=$DLLVersion, Culture=neutral, PublicKeyToken=$PublicKeyToken")
        } Catch {
            Write-Verbose "Unable to load $AssemblyName from GAC. Trying PowerShell module base folder..."
            #Can't load from GAC, now try PS module folder
            $DLLFilePath = Join-Path $DLLPath $DLL
            [Void][System.Reflection.Assembly]::LoadFrom($DLLFilePath)
            Write-Verbose "Successfully loaded $DLL from PowerShell module base folder"
        }
        $bSDKLoaded = $false
    }
}
$bSDKLoaded
```

OpsMgrExtended

- 29 Functions in the initial release:

SDK Connection Functions

- Import-OpsMgrSDK
- Install-OpsMgrSDK
- Connect-OMManagementGroup

Administrative Tasks

- Approve-OMManualAgents
- Backup-OMManagementPacks
- Add-OMManagementGroupToAgent
- Remove-OMManagementGroupFromAgent
- Get-OMManagementGroupDefaultSettings
- Set-OMManagementGroupDefaultSetting

Basic Authoring Tasks

- Get-OMManagementPack
- New-OMManagementPack
- Remove-OMManagementPack
- Copy-OMManagementPack

- New-OMManagementPackReference
- New-OM2StateEventMonitor
- New-OM2StatePerformanceMonitor
- New-OMPerformanceCollectionRule
- New-OMEventCollectionRule
- New-OMServiceMonitor
- New-OMInstanceGroup
- New-OMComputerGroup
- New-OMConfigurationOverride
- New-OMPropertyOverride
- New-OMOverride
- Remove-OMGroup
- Remove-OMOverride

Advanced Authoring Tasks

- New-OMTCPPortCheckDataSourceModuleType
- New-OMTCPPortCheckMonitorType
- New-OMTCPPortMonitoring

Configure OpsMgrExtended in SMA

- Connection Type:
 - Operations Manager SDK
- ComputerName:
 - One of the SCOM management servers
- UserName:
 - A Service Account that has SCOM administrator access (i.e. Data Access Service Account)
- Password:
 - Password for the service account

ADD CONNECTION

Configure connection

CONNECTION TYPE
Operations Manager SDK

NAME
OnPrem_MG

DESCRIPTION
OpsMgr SDK connection to the On-Premise Management Group

1 3

ADD CONNECTION

Configure connection properties

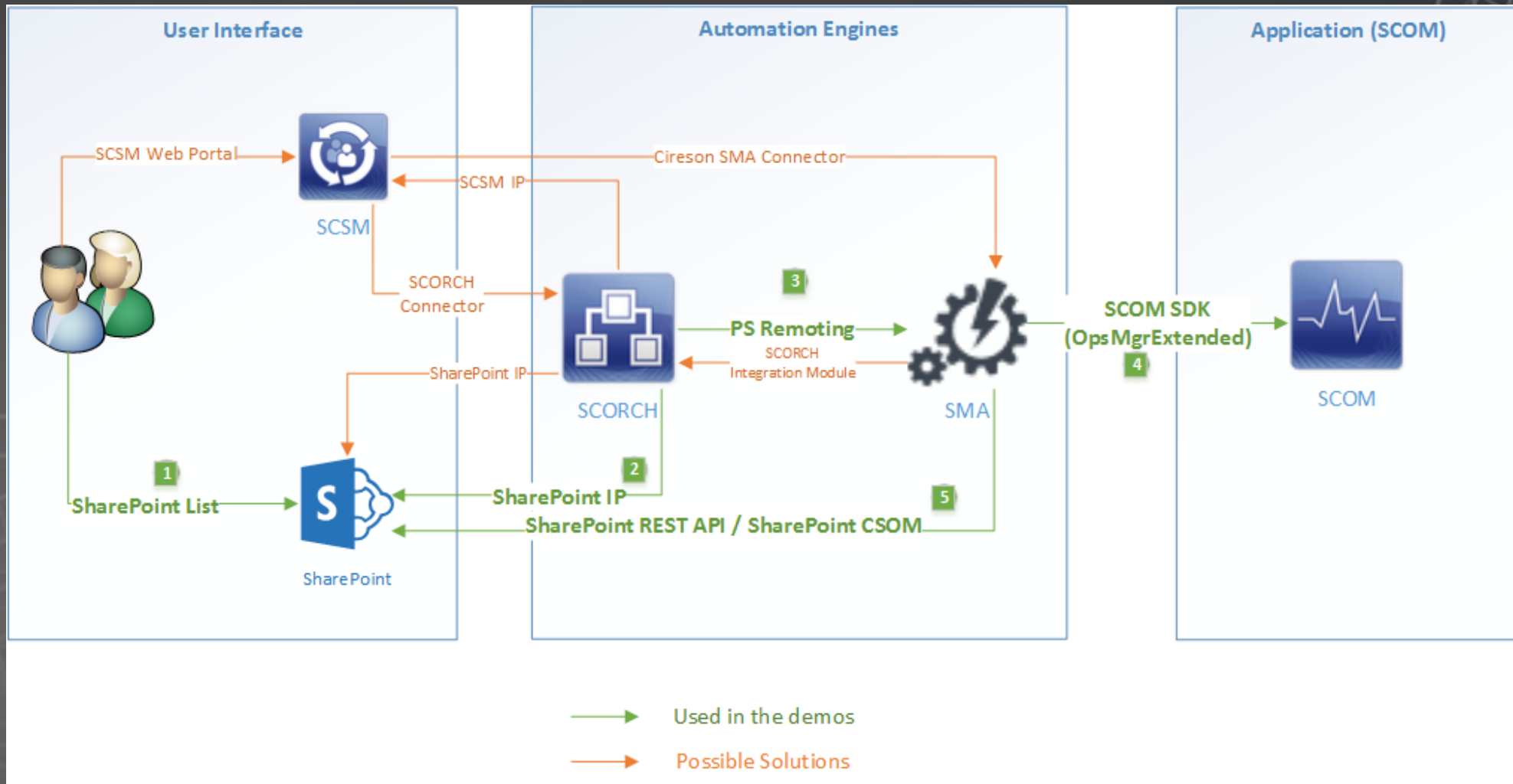
PASSWORD
••••••••

COMPUTERNAME
OpsMgrMS01.corp.tyang.org

USERNAME
Corp\SVC_OpsMgrSDK

1 2

Systems Integration Options



Demo 1: Configuring SCOM

- Copying Management Group settings between multiple MGs

Demo 2: Basic Authoring Task

1. Create a Blank Unsealed MP (In “Test” Environment)
2. Create a 2-State Performance Monitor (In “Test” Environment)
3. Populate a MP List on SharePoint
4. Copy the newly created MP & Monitor to another MG (Move to “Production” environment).

Demo 3

- Create TCP Port Monitoring
 - Equivalent to the TCP Port Monitor template
 - With many improvements
- What MP elements are created?
 - Class Definition for TCP Port Watcher and various groups
 - Class Relationships
 - Class and Relationship Discoveries
 - Data Source Module Type
 - Monitor Type
 - Performance Collection Rule
 - 4 Unit Monitors and a dependency monitor
 - Discovery Overrides

Modules / Runbooks Used in the Demo

- SMA Module: **OpsMgrExtended**:
 - During final testing phase
 - To be released after been tested / reviewed by a selected group
 - Stay tuned!
- SMA Module: **SharePointSDK**:
 - <http://blog.tyang.org/2014/12/23/sma-integration-module-sharepoint-list-operations/>
- SMA Modules: **SendEmail & SendPushNotification**:
 - <http://blog.tyang.org/2014/10/31/simplified-way-send-emails-mobile-push-notifications-sma/>
- SMA Runbook: Update SharePoint 2013 List Item:
 - <http://blog.tyang.org/2014/08/30/sma-runbook-update-sharepoint-2013-list-item/>

Additional Resources

- Service Management Automation (SMA)
 - SMA Whitepaper (Michael Rüefli): <https://gallery.technet.microsoft.com/Service-Management-fcd75828>
 - Master Class: Orchestrating Daily Tasks Like a Pro (Pete Zerger and Anders Bengtsson): <https://www.youtube.com/watch?v=7vpR6CdacL8>
 - Building a Real self-service platform with SCSM, SMA PowerShell Workflows (Travis Wright): <http://channel9.msdn.com/Series/SCUE2014/Building-a-Real-self-service-platform-with-SCSM-SMA-PowerShell-Workflows>
 - Service Management Automation – Introduction (Jakob Gottlieb Svendsen): <http://channel9.msdn.com/Series/SCUE2014/Service-Management-Automation-Introduction>
 - Service Management Automation (SMA) deep dive (Markus Klein and Michael Rüefli): <http://channel9.msdn.com/Series/SCUE2014/Service-Management-Automation-SMA-deep-dive>

Additional Resources

- System Center Operations Manager (SCOM) – MP Dev and SDK Related
 - SCOM 2012 SDK Documentation: <https://msdn.microsoft.com/en-us/library/hh329086.aspx>
 - SCOM SDK – Automating Management Pack Development: <https://msdn.microsoft.com/en-us/library/hh329043.aspx>
 - Russ Slaten's blog (many good code examples): <http://blogs.msdn.com/b/rslaten/>
 - OpsMgr Self Maintenance MP (SDK code examples): <http://blog.tyang.org/2014/06/30/opsmgr-2012-self-maintenance-management-pack-2-4-0-0/>
 - Jonathan Almquist's blog (Good MP Authoring articles): <http://blog.scomskills.com/>
 - Online MP Authoring Course: <http://channel9.msdn.com/Series/System-Center-2012-R2-Operations-Manager-Management-Packs>

Additional Resources

- SCOM Blogs to follow (the “usual suspects”, ordered by name):
 - Bob Cornelissen: <http://www.bictt.com/blogs/>
 - Cameron Fuller: <http://blogs.catapultsystems.com/cfuller/default.aspx>
 - Dieter Wijckmans: <http://scug.be/dieter>
 - Kevin Greene: <http://kevingreeneitblog.blogspot.com>
 - Kevin Holman: <http://blogs.technet.com/b/kevinholman>
 - Marnix Wolf: <http://thoughtsonopsmgr.blogspot.com>
 - SCOM Product Team: <http://blogs.technet.com/b/momteam/>
 - Stefan Roth: <http://stefanroth.net/>
 - System Center Central: <http://www.systemcentercentral.com>
 - Tim McFadden: <http://www.scom2k7.com/>

Q & A