



System Center Orchestrator to Azure Automation Migration Guide



Keverion



Introduction

For companies using Microsoft System Center Orchestrator versions 2016, 2019 or earlier who are looking for a replacement for their aging Orchestrator systems, they have an option to migrate to a new 64 bit version Orchestrator 2022 (more details on this can be found in the [Kolverion Orchestrator Best Practices Guide](#)) but many companies are considering replacing their Orchestrator system entirely and are looking for guidance.

Equally along with the increasing move away from on-premises into the cloud, companies are looking at what the migration strategy is for their Orchestrator system and how to automate their cloud and any legacy on-premises components going forward.

You could, of course, host Orchestrator on VMs in the cloud, but that does defeat the point of cloud migration because companies ideally want a cloud-first Software as a Service (SaaS) solution.

The migration from Orchestrator into a cloud automation solution is confusing, so this guide is intended to provide you with some clarity on how to progress.

Table of Contents

- Introduction 1
- Microsoft Automation Platforms 3
- Migrating from System Center Orchestrator to Azure Automation 5
- System Center Orchestrator vs. Azure Automation Capability Comparison 9
- Azure Automation Costs & Runbook Analysis 14
- Runbook Conversion 16
- Conversion 17
- Hybrid 19
- Summary 20
- About Kolverion 21



Microsoft Automation Platforms

Microsoft has several SaaS automation options which starts the confusion early on, particularly as some of the messaging does not reflect what the products are designed to do.

There are three SaaS automation platforms available from Microsoft:

- Power Automate – RPA (Robotic Process Automation) platform
- Logic Apps - Integration Platform as a Service
- Azure Automation – Dedicated IT Process Automation as a Service platform

So, let's assess each as their suitability for replacing System Center Orchestrator.

Power Automate

RPA tools are not designed or intended for infrastructure and IT process automation and Power Automate is no exception, so it is not a replacement for System Center Orchestrator.

Logic Apps

While Logic Apps does offer No Code automation, the main disadvantages are that it does not have a capability to execute on-premises tasks; it is a cloud first API automation system and you cannot execute PowerShell natively within a Logic App. So while Logic Apps do have some useful features and may well make up part of your wider solution, on its own it is not a replacement for System Center Orchestrator.

“RPA tools are not designed or intended for infrastructure and IT process automation and Power Automate is no exception, so it is not a replacement for System Center Orchestrator.”

Senior Automation Consultant,
Kelderion



Azure Automation

Azure Automation is a cloud-based automation platform that provides an expansive automation service while remaining cost effective thanks to both low upfront and long-term running costs.

Azure Automation has all the capabilities of Orchestrator and being a modern PowerShell-based platform, adds significant new capabilities that are not easily available in Orchestrator, such as Global Deployment, cross-region resiliency, GIT integration for collaborative working and change management.

Azure Automation provides the ability to create Graphical PowerShell Runbooks; and when paired with the Keverion Runbook Studio, you get the same No Code – Low Code authoring experience you have come to expect with Orchestrator.

In simple terms Azure Automation is the replacement for System Center Orchestrator.

If you want more details on how the three automation platforms compare please [see our blog on Microsoft Automation Platforms](#) and how they compare.

“Azure Automation has all the capabilities of Orchestrator, and being a modern PowerShell-based platform adds significant new capabilities that are not easily available in Orchestrator”

Senior Automation Consultant,
Keverion



Migrating from System Center Orchestrator to Azure Automation

When companies are looking to move from Orchestrator to Azure Automation, there are three common concerns:

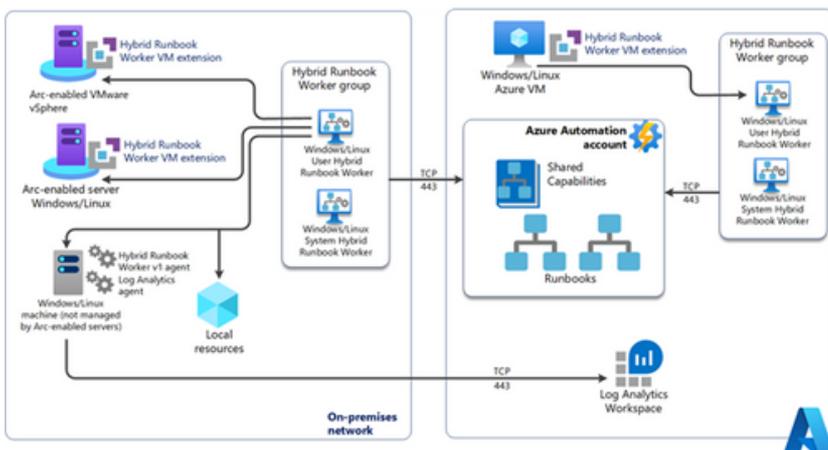
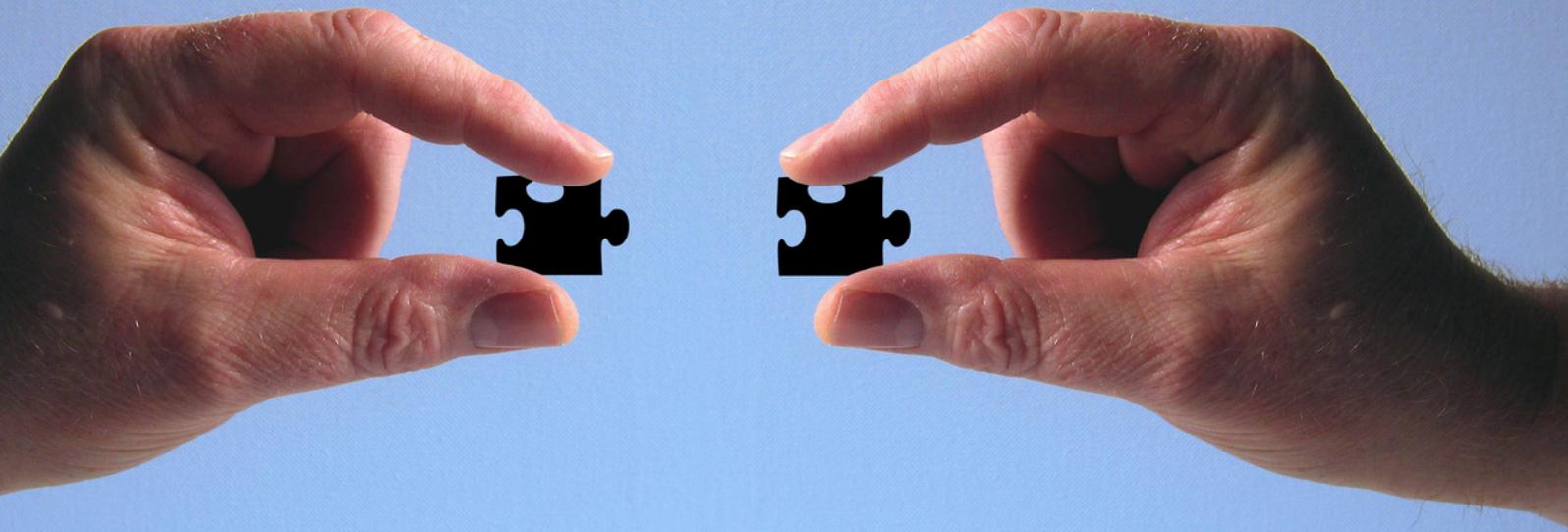
- On-Premises Automation
- Security
- Replacing the Monitor Activities in their Orchestrator Runbooks

On-Premises Automation

While being a cloud-based Automation as a Service solution, Azure Automation also has the capability to integrate and automate on-premise infrastructure and systems. It has a component called a Hybrid worker, which is an agent you deploy onto a server within your datacenter where runbooks execute when they need to interact with on-premises systems.

All the network communications are outbound from the Hybrid Worker over HTTPS using port 443; there is no inbound communications or trusts from the Azure Cloud.

The Hybrid Worker has a set of Integration Modules deployed onto it; and periodically, it reaches out to Azure Automation and says, “Do you have any work for me to do?” If there is work for it to do, then Azure Automation will pass over the automation Runbook to execute and some input data, “in effect metadata for the automation”, then the hybrid worker will execute the automation locally using the account credentials configured locally on the Hybrid Worker – so Azure Automation in the cloud does not have direct access to your on-premise servers or any credentials; you manage and control that all locally for optimum security.



Hybrid Runbook Worker

- Client Side Initiation
- Redundancy Available
- Runs Local PowerShell Modules
- Windows and Linux Options

It is worth noting that from 1st October 2023 onwards any new On-Premise VM hosting the Hybrid Worker must be an Azure-Arc enabled machine to allow you to deploy the Hybrid Worker VM extension function. For more details see this [Microsoft article](#).

Security

When it comes to security, the major concern with Azure Automation is around on-premises automation and do I have to open up my datacenter to Azure to use the automation.

The short answer is no – the Hybrid Worker resolves that issue because it is deployed and runs on a server in the datacenter, so all execution and key company data stays on-premises.

If HTTPS is not secure enough and you want further control, Azure Automation also supports Azure Private Endpoint and Private Link, effectively bringing the Automation service into your VNet eliminating exposure from the public internet. For further information [see this Microsoft Page](#).



Monitor Activities

Orchestrator has the capability of a Monitor Activity; the Monitor sits at the beginning of a Runbook and is targeted either:

- At a System to look for records, typically once every 15 seconds, which meet certain criteria, and if found, process those records.
- Check a Date or Time to execute the Runbook steps on a given frequency.

Azure Automation does not have a Monitor Activity equivalent, and you would not want to use it in Azure Automation. This is because for a Monitor Activity to work, the Runbook must be constantly running and you would then be paying execution minutes to Microsoft for a Runbook which was not doing any work most of the time.

In Azure Automation, rather than a monitor, the approach is to execute a

Runbook on a scheduled frequency which goes to the target and looks to see if there is any new data to process, a scheduled 'Look for Work to Do' Runbook.

There are two ways to schedule the Runbook execution frequency:

- The native Runbook Schedule function in Azure Automation
- A Logic Apps Recurrence Trigger

Choosing which one to use comes down to *complexity and frequency*. The Azure Automation Runbook schedule is easy to configure but the maximum frequency of scheduling is once an hour.

Logic App Recurrence is a bit more complex to setup as you are using two Azure Services, but you can schedule things as frequently as every second.



Runbook Schedule

- Azure Automation
- Max Hourly
- Easy to Associate
- Same Cost as Runbook

Logic App Recurrence

- Azure Service
- Max Every Second
- Calls a specific Azure Automation Runbook Webhook
- \$0.000125 per Web Hook Call

Using the execution frequency approach, the Orchestrator monitor equivalent in Azure Automation becomes a design similar to this:

Orchestrator	Azure Automation
Monitor a System to look for records, typically once every 15 seconds, which meet certain criteria and, if found, process those records.	Use a Logic App to trigger Azure Automation, typically once every 5 minutes, and execute a Get request from a system of all records which match certain criteria and, if found, process those records.
Check a Date or Time to execute the Runbook steps on a given frequency.	Use Runbook Schedule or Logic App to check a Date or Time to execute the Runbook steps on a given frequency.



System Center Orchestrator vs. Azure Automation Capability Comparison

Parameters	Features	System Center Orchestrator	Azure Automation
Functionalities	Process Automation	✓	✓
Runbook Authoring	PowerShell 5.1, 7.1, 7.2	5.1 in SCO 2022	✓
	Python 2, Python 3		✓
	Graphical	No Code Low Code with PowerShell editor	No Code Low Code PowerShell with <u>Kelverion Runbook Studio</u>
	Drag-and-Drop activities	✓	With <u>Kelverion Runbook Studio</u>
	User interface for authoring	✓	With <u>Kelverion Runbook Studio</u>
User Interface	User interface for at-scale management of resources provided		Azure Portal and <u>Kelverion Runbook Studio</u>
Architecture	High availability, load balancing, resiliency	Setup installed and managed by customer	Automatically handled by the service



Parameters	Features	System Center Orchestrator	Azure Automation
Assets	Variables	✓	✓
	Connections	✓	✓
	Schedules	✓	✓
	Modules	✓	✓
Integration packs Architecture	SCOM, SCCM, SCSM, DPM, SCVMM, Active Directory, Exchange, REST, SharePoint	Provided by MS	PowerShell modules available in gallery
	Third-party packs provided by Kolverion	<u>31 packs</u>	<u>20 packs</u>
	Integration toolkit to create integration packs	✓	✓
Runbook management	Centralized storage & management of runbooks	✓	✓
	Ability to test runbooks	Runbook Tester	✓



Parameters	Features	System Center Orchestrator	Azure Automation
Runbook management	Set of cmdlets for managing and starting runbooks		✓
	Ability to migrate runbooks and assets between instances and accounts		✓
	Runbook scheduling frequency	Seconds	Hourly
Runbook Execution Environment	Cloud based runbook execution		✓
	On-Premise runbook execution	✓	✓



Parameters	Features	System Center Orchestrator	Azure Automation
Security Features	Private links support		✓
	Customer Managed Key (Preview)		✓
	Role Based Access Control to regulate access to the Automation account and its resources.		✓
	Service tag can be used in place of specific IP addresses when you create security rules.		✓
Monitoring	Track success and failure jobs		✓
	Integration with Azure Monitor to send emails		✓



Parameters	Features	System Center Orchestrator	Azure Automation
Platforms supported	Windows	✓	✓
	Linux		✓
Runbook Gallery	GitHub & PS Gallery integration		✓
Source Control integration	This feature promotes configuration as code where runbooks or configurations can be checked into a source control system.		✓ Enhanced with Keverion Runbook Studio
<u>VM lifecycle management</u>	Schedule VM start/stop based on calendar or resource utilization using built-in runbooks		✓
Upcoming New Capabilities	PowerShell 7 support, native Az module support, Azure CLI		✓
Billing	License based	✓	
	Pay-as-you-Use		✓



Azure Automation Costs & Runbook Analysis

Azure Automation Costs

The Azure Automation core service, the Automation Account is provided free by Microsoft. The charging for Azure Automation is done based on Runbook Execution time. The Runbook Execution is measured in minutes and is charged at \$/£/€ 0.002 per minute.

For more details on how the Azure Automation pricing works and how your costs would work based on your automation usage read our [pricing blog](#).

Runbook Analysis

Having determined that Azure Automation is the right platform for your Orchestrator migration, the next task is to understand the scope of the migration ahead of you.

The typical Orchestrator system will be populated with many Runbooks which have been built over a number of years, which may or may not be in use,

and exactly how they operate may not entirely be clear.

Many of the Runbooks will be linked together to form a common process - 'a Solution'. So you need to determine the linkages, those that are dependent on or call another Runbook.

You will need to understand:

- Whether Runbooks are active and in use or have been deprecated
- How they are linked
 - And for those linked Runbooks how the process is initiated
- Which Integration Packs are used in the Runbooks
- For any Integration Pack used, you will need to know if equivalent PowerShell Integration Modules exist, of if there are other cmdlets which can be used in their place.

The good news for Keverion customers is that we have equivalent Integration Modules for most of our Orchestrator Integration Packs.



Solution

- How is the automation initiated (Service Desk or some other system)
- List of all Orchestrator Integration Packs (IPs) Used

Name	Type	Any IPs	Initiated	List of IPs
Runbook1	Management	No	Time Driven	None
Runbook2	Hardware	Yes	Triggered	Kelverion IP for SQL Kelverion IP for ServiceNow
Runbook 3	Email	Yes	Triggered	Codeplex IP for Email
Runbook3	Deprecated	Yes	Triggered	Kelverion IP for SQL Microsoft IP for Active Directory

Once you know which Runbooks need migrating and any Integration Pack dependencies, you are ready to look at the conversion of the Runbooks themselves.



Runbook Conversion

When you are ready to start the Runbook conversion, there is a key consideration that you need to consider; while Orchestrator and Azure Automation both provide graphical Runbooks, the way their automation framework processing operates is different. This is largely because of the advances in technology which have happened since Orchestrator was conceived and Azure Automation now exploits.

This means that logic that works in an Orchestrator Runbook may not be efficient in Azure Automation; or in fact, will not work at all and will need manual redesign – this affects how you can convert Runbooks into Azure Automation.

There are three approaches available for the actual Runbook Conversion, which will discuss:

Migration

- Use Microsoft’s System Center Orchestrator Migration Toolkit

Conversion

- Use Existing Runbooks as a guide to build New Azure Runbooks

Hybrid

- Develop a strategy to leave SCO in place while new or updated work is done in Azure Automation

System Center Orchestrator Migration Toolkit

While Microsoft have provided the Orchestrator migration toolkit, companies who have looked at it have found challenges in its use and have followed the Conversion or Hybrid approach. For more details on this see this [Microsoft article](#).



Conversion

When we are talking with companies about migration from Orchestrator to Azure Automation there are often some typical themes:

- The people who built the Runbooks in Orchestrator have left the business.
- The new Orchestrator team are not actually sure how they work.
- Although the Runbooks automate part of the process, we have some new systems which include tasks it does not automate.
- It would be good if the automation also did something they did not include before as part of the process.

In all of these cases, it does not make sense to migrate the existing Runbooks into Azure Automation because they don't meet the requirements anymore or the knowledge about how they work has been lost. In this case, it is actually better to convert the automated task into a completely new Azure Automation Runbook that delivers the new required process:

- You take the Orchestrator runbook as a template for the interactions and data that needs to be passed around.
- You look at the actions being performed by Orchestrator.
- Add in any new actions or tasks which need to be performed.

Once you understand these aspects, you can design Azure Automation Graphical Runbooks which perform all of the required tasks in an optimal manner for execution in the new platform.

Just like Microsoft has provided a Migration Toolkit, Keverion has a set of tools that we use for conversion; together that are the Keverion Runbook Suite:

Keverion Runbook Studio

- Low code/No code authoring environment for Azure Automation
- Familiar interface for current Orchestrator users



- Powerful tools for building, managing, and testing azure runbooks.
- Connect to multiple Azure AD Tenants, Subscriptions and Automation Accounts all from within one view.
- Removes the need for deep PowerShell or scripting knowledge.
- GIT Integration for Runbook version control

Kelverion Smart Integration Modules

- Supported Enterprise Class Integration Modules
- Provide design time information about integration target

Kelverion Automation Portal

Self-Service Portal for people to request that automated tasks are performed.

- Built from an ethos of 'Simple and Flexible.'

Provides rapid development and deployment of catalog items.

- Produces automation quality request data easily accessible by Azure Automation.

These tools make it simple and straightforward to design and manage Runbooks like you would in Orchestrator but allow you to leverage all the modern benefits of Azure Automation.

You can design, test, and manage your automation with Kelverion and rely on the power and security of Microsoft to deliver your 'Automation as a Service' platform anywhere in the world.

"Kelverion's integrations worked right out of the box with no Runbook server modifications, a huge point I made to my manager...Thanks for all your help!"

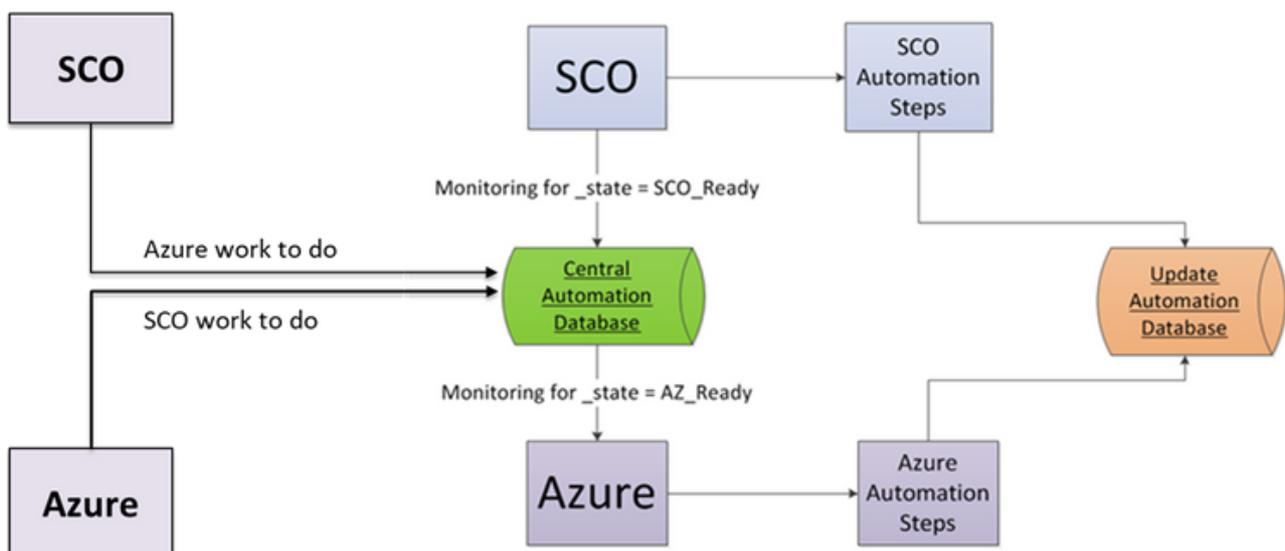
Technical Administrator, US Financial Services Organization

Hybrid

If your existing Orchestrator Runbooks are still meeting the desired needs of on-premises automation or you have a large number of working Orchestrator Runbooks, but you have completely new automation tasks to perform, particularly in the cloud, then using a hybrid approach of Orchestrator and Azure Automation in parallel makes sense.

Your Orchestrator system continues to operate, and you build all new automation in Azure Automation and over time as your Orchestrator processes need major rework, you rebuild in Azure Automation. In the fullness of time the bulk of the Runbooks in Orchestrator will become redundant, and you will eventually turn the platform off.

If you want to get both tools working together to handle on-premises with Orchestrator and cloud tasks with Azure Automation, then you can join them using a shared automation database where they record tasks for each other to do in a central database, and then watch for the notifications that the other party has completed its tasks.





Summary

Whatever your current position is with Orchestrator, if you are looking to move to a more modern No Code – Low Code authoring platform, then Azure Automation is the right platform for you.

The number of Runbooks you have performing actual work in Orchestrator today will define how you transfer across to Azure Automation, but the conversion or hybrid approach is proving to be the best and most efficient methods for customers who have already made the move, so you would be well advised to go down one of those two paths.

Further Reading

[What is Azure Automation?](#)

[Azure Automation costs explained](#)

[Top 10 IT automation mistakes and how to avoid them](#)

[Downloadable templates for runbook documentation](#)

[Azure Automation Best Practices](#)

[Planning an IT Automation Project](#)

About Keverion

Experts in Cloud, On-Premise and Hybrid automation, Keverion provide solutions and integrations that remove the manual process tying up IT staff; transforming the productivity, efficiency, and supportability of IT service automation. Our products utilize and enhance the power of Microsoft Azure and System Center Orchestrator.

Working closely alongside Microsoft, we have developed our integrations and automation solutions to help bridge the gap between Microsoft's automation platforms and third-party systems, in the process building key alliance partnerships with multiple vendors to ensure our products are fully certified.

Since 2010, Keverion has expanded to become a global company, with offices now in the UK, Canada, and the US. Through this, we are able to offer and support products and professional services engagements to enterprise-level organizations no matter where they are.

There's a smarter approach to IT Service Management

Get in touch to find out more



www.keverion.com



info@keverion.com



US: +1 289 801 0559
UK: +44 203 875 8035



<https://www.linkedin.com/company/keverion-automation/>