# Table of Contents

**Introduction** ................................................................................................................................. 3  
Document Overview ............................................................................................................................. 3  
Audience ................................................................................................................................................ 3  
**Kelverion Test Drive** ......................................................................................................................... 4  
Before You Begin .................................................................................................................................. 4  
References ............................................................................................................................................... 4  
**The Kelverion Runbook Studio** ......................................................................................................... 5  
Kelverion Licensing ............................................................................................................................... 5  
  * Kelverion Runbook Studio License ................................................................................................... 5  
  * Kelverion Integration Module Licenses ........................................................................................... 6  
Kelverion Runbook Studio Layout .......................................................................................................... 7  
  * The Toolbar ....................................................................................................................................... 7  
  * The Resources Pane ......................................................................................................................... 7  
  * The Runbook Canvas ....................................................................................................................... 8  
  * Runbook Properties ......................................................................................................................... 8  
**Connecting the Kelverion Runbook Studio to Azure** ........................................................................ 9  
Setup Azure Connection ........................................................................................................................ 9  
Connect to Azure Subscription .............................................................................................................. 10  
Add Evaluation Licenses to the Connection Assets ............................................................................... 12  
Create the “AutomationData” Design-time Connection ...................................................................... 12  
**Configure the Kelverion Automation Portal** .................................................................................... 15  
Licensing the Portal ............................................................................................................................. 15  
Submitting a Request ........................................................................................................................... 16  
**Creating Your First Runbook** ......................................................................................................... 17  
Tutorial ................................................................................................................................................ 17  
**Deeper Discovery Example** ............................................................................................................ 21  
**Next Steps** ...................................................................................................................................... 39  
Suggested Exercise ............................................................................................................................... 39  
**Contact Information** ....................................................................................................................... 40
Introduction

The Kelverion Runbook Studio is a graphical Runbook authoring environment, that can be used to generate Graphical PowerShell and Graphical PowerShell Workflow Runbooks for Azure Automation, as well as generating PowerShell snippets which can be used within SMA and your own scripts.

Kelverion have created a Test Drive facility which enables you to try out the Kelverion Runbook Studio in a sandpit environment without having to set up any infrastructure.

Document Overview

This document contains two tutorials that will take you through creating a Graphical Runbook using the Kelverion Runbook Studio and show you how exploit the power of Smart Discovery.

Audience

This document is targeted at readers who are reviewing the Kelverion Runbook Studio to enhance their Runbook creation capabilities. It is intended as a companion to the Kelverion Test Drive and provides configuration information on how to create Runbooks shown in the video tutorials.
Kelverion Test Drive

The Kelverion Test Drive is a preconfigured environment that provides a sand pit area to experiment with Kelverion technologies. It is hosted in the cloud, so it can be connected to using only an RDP client and an Internet connection.

To arrange for a Kelverion Test Drive please get in touch with Kelverion using the Contact Information at the end of this document.

Before You Begin

This tutorial is a follow-on companion to the 'Introduction to Kelverion Runbook Studio’ video which demonstrates using the Kelverion Automation Test Drive. If you haven’t watched the video yet, you are strongly encouraged to do so.

In the video you will complete the following configuration of the Kelverion Test Drive environment.

- Added the Evaluation License to the Kelverion Runbook Studio
- Configured the Kelverion Runbook Studio to connect to the Test Drive automation account
- Added the evaluation license to the Automation Portal
- Created a request in the Automation Portal
- Created your first Runbook
- Understood where to find any authentication information.

References

The video mentioned above is called Introduction to the Test Drive for the Kelverion Runbook Studio which can be found here:

Introduction to the Test Drive for the Kelverion Runbook Studio

Other instructional videos on using the Kelverion Runbook Studio and on other Kelverion products can be found on our YouTube channel here:

https://www.youtube.com/user/Kelverion

User guides and further information for all our products can be found on our website here:

www.kelverion.com
The Kelverion Runbook Studio

The Kelverion Runbook Studio can offer the greatest acceleration to your Runbook development when used in conjunction with the Kelverion Integration Modules. These Integration Modules can be used in any PowerShell environment and provide a rich integration experience while developing your Runbooks and provide a supported method to integrate into many applications.

To further simplify the design and maintenance of your Runbooks, the Kelverion Runbook Studio provides “Smart Discovery” when used in conjunction with the Integration Modules. Smart discovery allows you to use the Kelverion Runbook Studio GUI to explore the application, cmdlets, and API that you are interacting with and generate Runbooks with a few quick clicks.

Graphical Runbook authoring within the Kelverion Runbook Studio allows you to focus on the logic flow of your automation without getting bogged down with syntax, this change of focus opens the doors for many more people to exploit the power of Azure Automation.

Kelverion Licensing

The Kelverion Runbook Studio and the Kelverion Integration Modules that it uses both require licensing.

Kelverion Runbook Studio License

An evaluation license file is saved on the Test Drive virtual machine for the use of the Test Drive.

When the Kelverion Runbook Studio first starts it requests that you supply a valid license.

Click the “Add License” button and browse to the location that the KASL file is stored.
Ensure that the License Status changes to green with a valid date.

Licensing the Kelverion Runbook Studio outside the Test Drive is achieved in the same way by using a license file supplied by Kelverion. Use the contact details at the end of this document for further assistance.

**Kelverion Integration Module Licenses**

The Integration Modules are Licensed from this same window by clicking on the “Integration Modules” section. If you have closed this window, then you can open it again from “File \ About \ License Information”

Click on “Add License” and browse to C:\__TestDrive__

There are no diagrams or images in the text.
Ensure that the license has a **green** tick and a valid date.

---

**Kelverion Runbook Studio Layout**

In this section we will examine the parts of the Kelverion Runbook Studio in a little more detail. The image below has the key areas highlighted and are described in more detail below.

---

**The Toolbar**

The area highlighted in yellow is already familiar to us and as you would expect it allows quick access to many of the key functions.

**The Resources Pane**

The area highlighted in blue is the Resources Pane. There are 2 modes for the Resources pane, Toolbox and Azure:

- **The Azure category is for managing the resources that exist within Azure. These are the Azure Assets such as variables, connections, credentials, and certificates along with Runbooks.**
The Toolbox category provides access to the Activities that we will add to our Runbooks.

Switching between the “Toolbox” category or the “Azure” category toggles between those 2 modes.

You may also have noticed the search box at the top of the Resources pane. Both the Toolbox and Azure modes can be searched and filtered.

This example shows the list of cmdlets being filtered using the search facility.

The Runbook Canvas

The Green area in the middle of the screenshot above is the Runbook canvas, this is where we will place the Activities that go to make up our Runbook. You will see that a new empty Runbook is created on the Runbook canvas.

Right clicking on the Canvas brings up a context menu (shown highlighted in red) shows Use commands depending on when and where you right click on an area of the Kelverion Runbook Studio.

Runbook Properties

The purple area to the right of the image above is the properties pane, which is also context sensitive, containing the Activity properties or the Runbook properties.

Tip: Click an empty area in the Runbook canvas to get back to the Runbook properties after you have configured an activity.
Connecting the Kelverion Runbook Studio to Azure

The Kelverion Runbook Studio can work in either an on-line or off-line mode. Here we see how to work connected to your Azure subscription by setting up the configuration needs to build a Runbook.

We must configure the Connections that are used at design time and those used at runtime separately. The “Connection Assets” within your Azure Automation account are used at the time of Runbook execution.

The “Smart Connections” used within the Kelverion Runbook Studio are used at Runbook design time to discover the properties of the target system and to assist your configuration of the Activities.

Setup Azure Connection

This step needs to be carried out the first time you connect to an Azure Subscription.

In the Runbook Studio click on “File” and then “Account”.

![Account section of the Runbook Studio](image)
Click on “Manage Tenants” to open the Tenants window.

Then click on “New” to enter a new Azure Tenant.

Enter the name “Kelverion Test Drive”

And the Azure Active Directory ID “587672c8-f540-4522-a1ed-20367b32bc4a”

**NOTE: This ID can be found in the file “C:\_TestDrive_\ConfigInfo.txt”**

Click OK and OK to close these windows.

Ensure you make the tenant active by selecting it.

**Connect to Azure Subscription**

In the Kelverion Runbook Studio follow the guidelines below.

When the Kelverion Runbook Studio starts up it will not be connected to your Azure subscription.

Click on “Connect” on the toolbar to start the login process.
You will see the familiar Microsoft prompt to login to your Azure subscription.
Copy the login information from the ConfigInfo.txt file and log in to Azure.

Once you have logged in you will notice that the subscription and account dropdowns are populated in the Toolbar.

*Note: If you have multiple Automation Accounts associated with your subscription, then you can easily manage them all with the Kelverion Runbook Studio.*

In the Kelverion runbook Studio click on the Azure component in the Resources Pane and expand the configuration.

Here you will see all the Accounts and Assets that are contained in the Automation Account.
Add Evaluation Licenses to the Connection Assets

When you first opened the runbook studio you selected the evaluation licenses for the Runbook Studio (design time) and the Integration Modules (run time.) Any connection assets that are created in your automation accounts that utilize the Kelverion Smart Integration Modules will need to have the run time licenses information associated with them, so they can execute correctly in Azure.

In the Azure Resources view

Right click on the connections category within the Automation account that you are updating

And select “Update License keys”

After a moment you will see the following confirmation

Create the “AutomationData” Design-time Connection

The steps below show how to configure the design-time connection. This enables the Runbook Studio to make the connection to the databases in SQL Server during the development process.
In the Kelverion Runbook Studio, click on “File” and then “Smart Connections” to open the “Smart Connections” window.

Click on “New”
Add the “Name” “AutomationData”
This should represent the type of data that the Connection represents
Select: “Kelverion. SqlServer” as the Connection type.
Enter the “ServerName”, “UserName” and “Password” from the ConfigInfo.txt file.
Click OK
Click OK on the “Smart Connections” screen to complete the process.
Configure the Kelverion Automation Portal

The Kelverion Automation Portal is a service request management system that has been deployed into the Test Drive environment to provide some context to the data being automated.

Licensing the Portal

As with the other Kelverion products the Automation Portal requires a license file to activate it. Follow the instructions below to do this.

Open Internet Explorer and go to the localhost address http://127.0.0.1

Login using the SQL Username and SQL Password credentials that are stored in the ConfigInfo.txt file.

*Note: The first time this site loads takes a while to initiate.*

Accept any Internet Explorer first time setup options.

When the “Invalid License” page loads, use the cog icon to go to the “Admin” > “Settings” menu.
When the “Settings” page loads scroll down to the “License” section
Click on the “Upload” button.

Browse to the “__TestDrive__” folder and select the Kelverion “AutomationPortal.KAL” file.
Click the “Import” button.
Check that the “Status” now shows as “Valid”

**Submitting a Request**

Once the Portal is licensed it can be used to enter a Request for processing with the Runbooks. This will record the data you enter and post it into a SQL table that the Runbook can access and subsequently process.

Select the “Services” menu from the top of the screen.
Click on the “Test Drive Service 1” item
Select the “Test Drive Offering 1” item.
Fill out the form with test data and click on the “Submit” button.
At the confirmation screen click the “Submit” button again.
You can check that the record has been submitted by opening the “Requests” menu at the top of the screen and looking for the entry you just made.

*Note: You will need the Request ID later in the Tutorial.*
Creating Your First Runbook

The Kelverion Runbook Studio allows you to create both “Graphical PowerShell Runbooks” and “Graphical PowerShell workflow” Runbooks. For most use cases you should employ “Graphical Runbooks”. Both types are broadly the same, with some extra functionality in the workflow variant, at the expense of some increase in complexity and a performance penalty.

Tutorial

Follow the instructions below to create the first Graphical Runbook.

In the Kelverion Runbook Studio click the “New” button on the toolbar.

Select the “Graphical Runbook” option.

You should have an empty Runbook in the canvas, and the Runbook properties open in the properties pane.

Enter a name for the Runbook in the Runbook Properties Pane.

Optionally add a description too.

Note: Runbook names need to be unique in Azure.

Now, ensure that the Resources pane is in “Toolbox” mode

In the search box type “Kelverion” to see all the Kelverion Activities or type “SqlRow” to filter the Activities further.
Drag and drop an “Select-SqlRows” Activity onto the Runbook Canvas.

Note the exclamation mark icon depicting that more configuration is required.

Click on the “Discovery” button in the “Activity Properties” pane and set the “Connection”, “Database Name” and “Table Name” using the drop-down boxes.

The Connection to select is called “AutomationData” however this is a bit confusing since we will be using it to connect to the “AutomationPortal” database.

Once the Connection is selected then the available databases are presented. Select the “AutomationPortal” database.

Once the database is selected the integration module can discover the tables it contains and present a list for you to choose from.

Pick the “dbo.Request” table from the drop-down when they are presented.

As each option is selected the next level of granularity is discovered. This is an example of Smart Discover in action.
Click on the “Filters” button and configure the settings to only get the SQL record with an “ID” equal to “10”.

**Tip:** It is a good practice to always set the filter first since they are easy to forget, and it will make a big difference to your Runbook without them.

Click on the “Parameters” button and select the mandatory “Connection” (highlighted with a red exclamation mark icon).

Select “Connection asset” and the “AutomationData” run-time connection.

Further parameters can be set by clicking on the “(n) Optional” button to reveal other properties that the Smart Discovery has detected.

Now the Runbook is complete, the draft can be published to the Azure Automation Account using the “Upload” button.

To test the Runbook, return to the Azure mode of the Resources pane and browse to the “Runbooks” section to find the Runbook that you just published.

Right click on the Runbook name and select the “Test Draft” option.
The Test windows opens where you can select where you want to run the test, in Azure or in a Hybrid Worker Group.

Click on the “Start” button to run the test.

The Runbook will be tested with in-progress information shown in both the main Output window and the Job status bar at the bottom of the screen.

Once the Runbook has completed, the output of the Runbook is displayed.

The Output window shows each field of the SQL row that has been returned by the Runbook Activity.

*Note the ID field value is “10” as per the Filter set in the Activity.*

In a real-world scenario, you would make any changes that are required then Publish the Runbook to make it ready for production.

This can also be done from the Runbook Studio using the menu as shown.
Deeper Discovery Example

In our first example Runbook, we quickly explored Smart Discovery. Let’s take a deeper dive into discovery. This example is going to integrate our Runbook with the Kelverion Automation Portal, we will see how to use the Automation Portal to prompt an end user for specific validated information that we can use to drive our automation.

The Automation Portal has already been configured with a simple service and a couple of simple offerings for this example. Once you have completed this example you should create a service and offering of your own, so you can see the power and flexibility of the Automation Portal.

In this example, we’re going to create a more complex Runbook with outputs from the activities used as inputs to the subsequent ones. Just as we would in most of the real Runbooks we create.

First, create a new graphical Runbook and name the Runbook “Portal-Integration1”.

Switch the Resources pane to “Toolbox” so we can find the activities we want.

We’re going to use several activities from the SQL Server integration module. Type “Kelverion” in the search box and then scroll down display the activities from the SQL Server Integration Module.
Drag the following activities onto the Runbook canvas
Select-SqIRow
Update-SqIRow
Insert-SqIRow

Then right click on the canvas to pop up a context menu and add a “code” activity
Join the activities up with pipeline links so that your runbook looks like this

rename the activities:

Select-SqIRow -> Get Request
Update-SqIRow -> Update Request
Code -> Process Request
Insert-SqIRow -> Create AutomationData Record
Your Runbook should now look like this

You will notice that each of the activities has the exclamation mark icon to indicate that they are not fully configured.

We’ll sort that out now, starting with “Get Request”. Click on the activity and switch to the discovery tab in the properties pane.
Click on the connection drop down, and then select the “AutomationData” configuration.

The Runbook Studio will discover the available databases. Select the “AutomationPortal” Database. The Runbook studio will now discover the all the tables that exist within the database.

Select the dbo.Requests Table.
The discovery process has enough information now so that it can give us the appropriate options while we configure the runtime properties on the “parameters” tab.

Switch to the parameters tab in the Properties pane.

Set the connection parameter to use the “AutomationData” Connection Asset

Once the connection is set the activity will no longer have the exclamation mark Icon (as all the mandatory properties have values) however we want to set a filter, so switch to the filters tab in the properties pane and click on “Add Filter”.

The “Edit Smart Filter” dialogue will pop up.

Click on the filter dropdown and you will see a list of the columns that exists within the table.

Select State.
In the ‘Operation’ field, you will see that we have text specific operators (contains, like, etc.) If you change the filter column to “Created” and check those operations again you will see that the operators are now specific to Date’s and Times. That’s an important feature of the discovery, and part of what makes it Smart discovery. The Runbook Studio in conjunction with the integration modules know how to interrogate the system that you are interacting with and take care of the details so that you don’t have to worry about it. It helps you to get the right low-level syntax for whatever the target application is, so you don’t need to be a SQL, or ServiceNow Guru. You can focus on the Runbook logic instead.

Switch the Filter field back to State, use the Equal operations and set the comparison to a constant “New”

![Edit Smart Filter](image)

Click on OK.

On the parameters we will set the optional properties so that a single row is processed each time the Runbook is executed. In the real world you would probably build the Runbook to process all the outstanding requests, but processing things one at a time make it easier for us to test our Runbook.

Set the “top” parameter to the constant “1”

![Activity Properties](image)

We’re done with the configuration of the “Select Requests” activity, and it will “publish” each of the fields onto the pipeline for the current request for us to “subscribe” to in the later activities.
Let's configure the “Update Request” activity to update the state of the request to “locked”

Configure the discovery within the properties pane to point to the “AutomationPortal” database, and the “dbo.Requests” table

In the “parameters” tab of the properties pane set the Connection again, and then set the Optional property ‘State’ to ‘Locked’.

We now switch to the Filters and create a new filter.

When the “Edit Smart Filter” dialogue opens select the Filter Column “ID” and “Equals” as the operation.
Now we're going to use data that's output from the previous activity rather than a constant value as we did before.

In the Data Source field select “Activity output”.

The Activity drop down is added, and this allows you to select the activity that we want to consume output from, select the “select department” activity.

Finally, “field path” input will allow you to select the property that you want to use.

If you look at the fields available there and check the able definition you will notice that each of the fields in the database table have been made available to us.

Select the “ID” field in the field path.

Click on OK to finish editing the filters.
Next, we will configure the “Process request” code activity.

Click on the activity in the canvas and notice that the code activity has different tabs in the activity properties.
Switch to the “Code” tab and paste in the above PowerShell. This PowerShell will take the XML data from the AutomationPortal request and perform some
processing. It will then make the information available to the subsequent activities in the Runbook.

For Example: when the user completes the request in the portal they specify their first name and last name;

The PowerShell takes that information from the data and produces the account name which is the first letter of the first name + the lastname, the SAMAccountName which is the account name + the domain, and the User Principal Name.

Our final Activity in this runbook is to Insert a record in the AutomationData database.

Click on the “Create AutomationData Record” activity in the runbook and configure the discovery.

Connection - > AutomationData
Database - > AutomationData (Notice: different Database to the previous activities)
Table - > TestDrive.Users
Then switch to the “parameters” tab

Set the connection to use the Connection Asset “AutomationData”

Now we will set the optional properties.

The “Enabled” field should be set to the constant value “True”

The “GivenName” field should be set to the “FirstName” activity output from the “Process Request” activity.
The “Name” field should be set to the “Name” activity output from the “Process Request” activity. This was one of the fields we manipulated in the PowerShell.

The “ObjectClass” field should be set to constant value “User”

The objectGUID is going to be generated “on the fly”.
Set the value to be the result of a PowerShell expression; then set the PowerShell to `[guid]::newguid().toString()`
The “SamAccountName” field should be set to the “SamAccountName” activity output from the “Process Request” activity. This was one of the fields we manipulated in the PowerShell.

![Image of SamAccountName activity]

The “Surname” field should be set to the “LastName” activity output from the “Process Request” activity.

![Image of Surname activity]

The “UserPrincipalName” field should be set to the “UserPrincipalName” activity output from the “Process Request” activity. This was one of the fields we manipulated in the PowerShell.

![Image of UserPrincipalName activity]

Finally, our runbook is complete.

Upload the Draft Version of the runbook to the Automation Account.
There will already be a request in the Automation Portal with user information set. Let’s verify that before we test the runbook.

Open the AutomationPortal by starting Internet Explorer and navigating to http://127.0.0.1/

You will be prompted to login (use the credentials from the file c:\__TestDrive__\ConfigInfo.txt)

Once you are logged in click on the requests menu, and then select “all requests”.

You should see that request number 8 was created some time ago. If you click on the + next to the request, you will be able to see the request details (as shown above.)

*If you want to create a new request, then click on the Kelverion logo to take you to the Portal home page, and then select the “Test Drive Service 1” and “Test drive Offering 1”. Remember though we configured the runbook to only process the first “New” request when it executes.*
Switch back to the Runbook Studio and select the Azure tab from the resources view.

Right Click on our draft runbooks “Portal-Integration1” and “Test Draft”

You will see the Runbook testing pane as you did in our previous test.

Eventually the runbook will complete. The output from the runbook (without increasing the logging) will just be the ID of the record that was added into the “TestDrive.Users” table in the AutomationData Database.

You should open SQL Management studio and check that table where you will see the values from the request after the effects of the PowerShell manipulation.

You should also check the state of the request in the Automation Portal, you should see that the request is still in a “locked” state, as our user has only been logged to the database so far.
<table>
<thead>
<tr>
<th>ID</th>
<th>Service</th>
<th>Offering</th>
<th>Created</th>
<th>Updated</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>Test Drive Service</td>
<td>Test Drive Offering</td>
<td>1/17/2018 3:09:29 PM</td>
<td>1/17/2018 3:09:29 PM</td>
<td>Locked</td>
</tr>
<tr>
<td>6</td>
<td>Test Drive Service</td>
<td>Test Drive Offering</td>
<td>1/17/2018 3:09:29 PM</td>
<td>1/17/2018 3:09:29 PM</td>
<td>New</td>
</tr>
</tbody>
</table>
Next Steps

Suggested Exercise

The next step on your automation journey would be to develop the Portal-Integration1 Runbook to:

- Manage the Lifecycle of the User record in the “TestDrive.Users” table to include a closed or deleted state.
- Manage the Request record state past Locked to Completed.
- Explore the Automation Portal to extend the “Test Drive Service1” Service to add a “Create PC” Offering that works in a similar way to the create user offering.
- Create a Runbook that manages the Create PC Request from the Automation Portal in the same way as the user Offering does.
- Try to integrate the “Test Drive Offering 2 – Ping a hostname” Offering in the Automation Portal with the Create PC Runbooks you created above.

Enjoy exploring this Kelverion Test Drive and thank you for your interest in our products.
Contact Information

Kelverion Automation Ltd
Unit 31, Thrales End Business Centre
Thrales End Lane
Harpenden
Hertfordshire AL5 3NS

Email: info@kelverion.com
Web: www.kelverion.com