Kelverion Automation

Ticket and DevOps Synchronisation Solution

User's Guide

Version 1.1

Table of Contents

1. Overview....3
1.1. Ticket and DevOps Synchronisation Solution Operation....4
2. Pre-Installation Information....6
2.1. Kelverion Ticket and DevOps Synchronisation Solution Package Contents....6
2.2. Integration Packs Required....6
2.3. Microsoft Products Required....6
2.4. Persistent Data Store....7
3. PDS Creation....8
3.1. PDS Creation Steps....8
4. Solution Installation Steps....8
4.1. Runbook Solution Installation....8
5. Solution Configuration....9
5.1. System Center 2012 Orchestrator Configuration....9
6. Customising the Ticket and DevOps Synchronisation Solution....14
6.1. Default 'Out of the Box' behaviour....14
6.2. Triggering the Runbooks....14
6.3. Configuring the Runbooks to a specific Target Service Desk....15
6.4. Configuring the Solution to Target 'Atlassian Jira'....16
7. Installing Temporary License of Kelverion Integration Packs....17
8. Upgrade Warning....18
9. Notes....18

1. Overview

The existence of multiple Service Desks from different Vendors is now commonplace across many Organisations. However, incorporating reliable communication and transfer of information across such Service Desks in an Enterprise setting is a time consuming, labour intensive and an error prone process.

It typically involves the User raising a call into a Helpdesk (A) to log an Incident in a Service Desk (A). The incident is then processed and after a period of time the information is then manually replicated by another team using a different Help Desk (B). Only now can the information can be actioned by an engineer exposed to Service Desk (B). At any point during the life span of this incident a sequence of unpredictable scenarios can play out i.e. the 'urgency' may change in Service Desk 'A' while the engineer is investigating, The 'category' in Service Desk 'B' may change resulting in the 'engineer assigned ' also changing. The ticket may even become 'Closed' or 'Resolved' on either side. All the time during the lifespan of this Incident, this information needs to be relayed and transferred to the opposing Service Desk. Not only is this very labour intensive but it is also prone to time delays and human error. The human error element most likely arises during the process of 'identifying what has changed in the Incident' in the respective Service Desk.

Up to now automating this process has become a minefield of challenges. System Center 2012 Orchestrator provides the platform from which to succeed.

However, even for an experienced Orchestrator developer, there are two significant challenges with envisioning a HelpDesk to HelpDesk or Help Desk to Bug Tracking solution. The first challenge is being able to interface and parse data between the two Systems bi-directionally. The second challenge is more complex and entails being able to identify and dynamically update or communicate only the changes that have occurred to the Incident in the other Service Desk.

The Kelverion 'Ticket and DevOps Synchronisation' Solution meets these challenges and delivers. A User can simply raise the Incident in System Center 2012 Service Manger and the Incident will be created in Service Desk B (e.g. ServiceNow or Atlassian Jira). Subsequent changes to the Incident by either Service Desk will be automatically relayed in isolation to the opposing Service Desk via Orchestrator. The relayed information will be in the form of communication to the respective Work Notes (ServiceNow) or Comment Log (Service Manager). The Operator is the then equipped to update the Incident appropriately and accurately.

The Kelverion Ticket and DevOps Synchronisation Solution leverages the Persistent Data Store design philosophy and the Kelverion Orchestrator Integration Packs to provide a scalable and robust solution.

Each companies Service Desk and Bug Tracking systems are configured slightly differently and the Incident processes varies so the solution is provided as a flexible working framework which can be tailored to each specific customer implementation. The benefit of the solution is that the design and operational workflow is in place and only requires customising to accommodate the actual fields present in the Service Desk Incident Ticket or Bug Tracking system Issue Ticket.

This solution is available as a self-installation package for customers proficient with System Center.

For customers who are less familiar with System Center 2012, Kelverion or our partners can provide a complete installation and configuration solution where we work with you to customise the solution for your environment.

This document provides the guidance on how to setup and configure this Solution in your environment. It is aimed at an experienced System Center 2012 users. Users should also reference Microsoft supplied documentation for the System Center tools and Kelverion Integration Pack User Guides.

1.1. Ticket and DevOps Synchronisation Solution Operation

The Automation Solution utilises a common Persistent Data Store (PDS). From a very high level, the Solution operates as follows:

1. An IT user raises an Incident in Service Manager with an *Urgency, classification, short description and* 'Support Group' (By Default '3rd Party Help Desk' must be selected to activate the initial runbook monitor, though this is freely configurable –See Section 6.2).

2. Orchestrator raises a new Incident in ServiceNow with enrichment of the data retrieved from steps 1 *(e.g. Urgency, classification, short description)* as well as updating the work notes section *(e.g. 'Entered by: J Blogs').* Once the Record is inserted, the raw data is recorded in the PDS allowing for comparison later.

3. The IT User updates the Incident in Service Manager. The Runbooks identify this and update the PDS. The 'Work notes' section of the record is updated in ServiceNow with a description of the changes. This will typically be things such as changes to 'assigned to' which have been identified from the Action Log in Service Manager.

   OR/AND

The ServiceNow operator updates an Incident in ServiceNow and an entry is created in the PDS. The PDS data is compared with the original ‘New ServiceNow Incident’ or the most recent update to the ServiceNow Incident, whichever is newer. The runbooks identify where text() has previously existed for a value e.g. Urgency: 'Low' (was) 'High' (now). Note it will **NOT** identify Assigned to: 'NULL' (was) 'Beth Anglin' (Now). However it will identify Assigned to: 'Beth Anglin' (was) 'Bud Richman' (Now). The Service-Manager ‘Comment Log’ is updated with the deltas by Orchestrator.

4. The IT User updates an Incident as ‘resolved’ or ‘closed’ in SCSM. A record updated in the PDS. The ‘Work Notes’ section of the Incident is updated in ServiceNow including a description of the changes. The ServiceNow operator can now manually update the Incident accurately by referring to the ‘Work Notes’ section.

   OR/AND

   An active incident is ‘Resolved’ or ‘closed’ in ServiceNow by the ServiceNow Operator and an entry is created in the PDS. Raw data is compared with the original 'New ServiceNow Incident' or the most recent update to the service Now Incident, whichever is newer. The runbooks identify where text() has previously existed for a value i.e. State: 'Active' (was) 'Resolved' (now). The Service Manager ‘Comment Log’ is updated by Orchestrator with the deltas as identified. As this record has been resolved or closed in ServiceNow, resolving or closing it in Service Manager will be a manual operation using the information provided in the ‘Comment Log’ as above.

   Sufficient logic is provided within the runbooks to prevent any unnecessary relay of information being bounced between Service desks at any stage.

The Solution is based around our experience of building this type of automated offering for many customers. The Automation Solution provides the foundations out of the box and allows extension and modification to tailor the solution to exact customer requirements.

By using the Persistent Data Store approach any complex decision logic can be handled using the database and Orchestrator Runbooks. You can extend the Runbooks yourself or we can provide consultancy to help you with the design update and Runbook modifications.

2. Pre-Installation Information

2.1. Kelverion Ticket and DevOps Synchronisation Solution Package Contents

Kelverion Ticket and DevOps Synchronisation Solution contains the following elements:

- Kelverion Persistent Data Store creation script (Microsoft SQL Server)
- Kelverion Ticket and DevOps Synchronisation Solution Runbook Set
- Kelverion Ticket and DevOps Synchronisation User Guide
- Kelverion Data Manipulation File: 'Datamanip_TicketSync.xml'
- Kelverion Style Sheet File: 'Diff.xsl'

2.2. Integration Packs Required

The solutions requires the following Integration Packs:

Microsoft

- System Center 2012 R2 Service Manager Integration Pack

Kelverion

- SQL Server Integration Pack
- ServiceNow Integration Pack
- Atlassian Jira Integration Pack
- Data Manipulation Integration Pack
- Text Manipulation Integration Pack

Before importing any Runbooks please ensure these Integration Packs are installed in Orchestrator. If you do not already have Kelverion Integration Packs they can be downloaded for evaluation from our website.

2.3. Microsoft Products Required

In addition to System Center 2012 R2 Orchestrator the following products are also required:

- System Center 2012 R2 Service Manager

2.4. Persistent Data Store

The Persistent Data Store or PDS is a SQL Server database that is used by this Solution to allow all of the actions that the Runbooks take to be carried out in a robust way. The use of the database at each “step” allows us to design the Runbooks such that each Runbook is simple and can be considered a discrete unit. In programming terms it allows the Runbooks to be modular.

In your environment there may be a number of constraints that control the creation of a new database. For example the location of the log and data files, the recovery options that should be used, and the collation of the server. These requirements are typically specified by the DBA responsible for your database server.

Location

Typically the PDS is created on the same database instance as is used for the Orchestrator database. There is no specific requirement that this must be the case. In environments where there is very high load you may find that creating the PDS on a different database instance advantageous.

Database version

The Runbooks provided, have been tested against SQL2012 with the latest patches / updates applied. You may need to modify the SQL Script to get it to operate in your environment or to install it on older versions of SQL Server.

Collation

The Runbooks have all been developed on systems using **Case Insensitive** collations as per the Microsoft System Center 2012 Orchestrator implementation guidelines, the specific collation setting used for your environment must be case insensitive other than that though the setting can be chosen as appropriate for your environment.

Sizing

The minimum recommended size of the PDS is 1GB.

The amount of space required will depend on the two following factors:

- Number of requests processed
- Housekeeping frequency as defined by your DBAs

3. PDS Creation

Each Kelverion Runbook Solution uses a set of common tables within the PDS Database and a set of tables specific to itself.

As part of each solution package you are provided with a SQL script which will generate the PDS database tables required for the solution. When the SQL scripts are executed they check for the existence of each table they required in the PDS database. If this is a new installation they will create both the Common Tables and their Solution Specific database tables. If you are already using a Kelverion Runbook Solution then the script will detect that some of the tables this solution requires already exist and the script skips these table creation steps and creates only the tables which do not exist in your installation.

This means you can easily deploy one or more Kelverion Runbook Solution to an existing PDS database without damaging and tables which already exist. This also means when installing multiple Runbook Solutions in a new installation of the PDS you can run each Solution PDS creation script in any order you like and know that when done you will have all the tables you need for all your Runbook Solutions to operate.

3.1. PDS Creation Steps

1. Create a New Database on your SQL Server called PDS_LIVE or connect to your existing PDS_LIVE database
2. Then execute the SQL Script provided within the PDS_LIVE database you created.
3. Once the PDS_LIVE database is created you must ensure the Orchestrator Runbook Server Service Account has as a minimum Read and Write Access permissions to the PDS_LIVE database.

4. Solution Installation Steps

The installation steps below assume that the System Center 2012 applications are installed. These steps should be followed by an experienced user.

4.1. Runbook Solution Installation

The Runbook Solution installation steps are as follows:

1. Install the required Kelverion and Microsoft System Center Integration Packs into Orchestrator. Follow the guidance given as per respective IP User Guide.
2. Import the Ticket and DevOps Synchronisation Solution Runbook Set.
3. Place the 'Datamanip_TicketSync.xml' and the 'Diff.xsl' in a new directory called 'C:\Orchestrator\TicketSync\' on each runbook server.

5. Solution Configuration

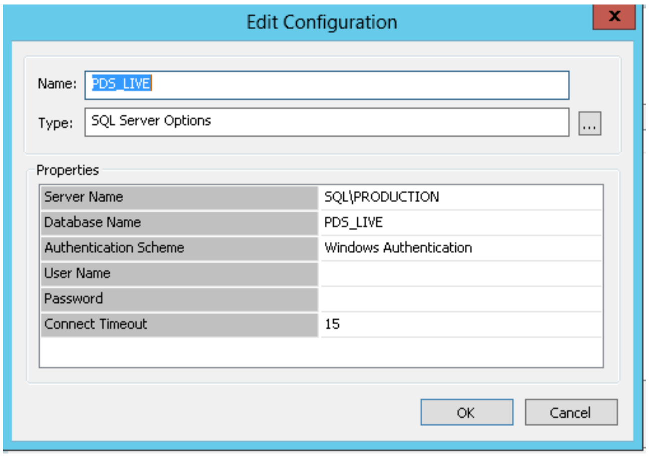
5.1. System Center 2012 Orchestrator Configuration

To use the Solution, you have to do a series of simple configuration steps to make the Runbooks operate in your environment.

KA SQL Server IP:

In the Options Menu of Runbook Designer find the KA SQL Server settings. In here you will need to create to database connections called (PDS_LIVE) which is required for the IP operation against the PDS.

Create a connection called PDS_LIVE and set the Database Name to (PDS_LIVE), set the database connection to the 'Server Name' and authentication details to values appropriate to your environment i.e.



Name: PDS_LIVE
Type: SQL Server Options

Server Name	SQL\PRODUCTION
Database Name	PDS_LIVE
Authentication Scheme	Windows Authentication
User Name	
Password	
Connect Timeout	15

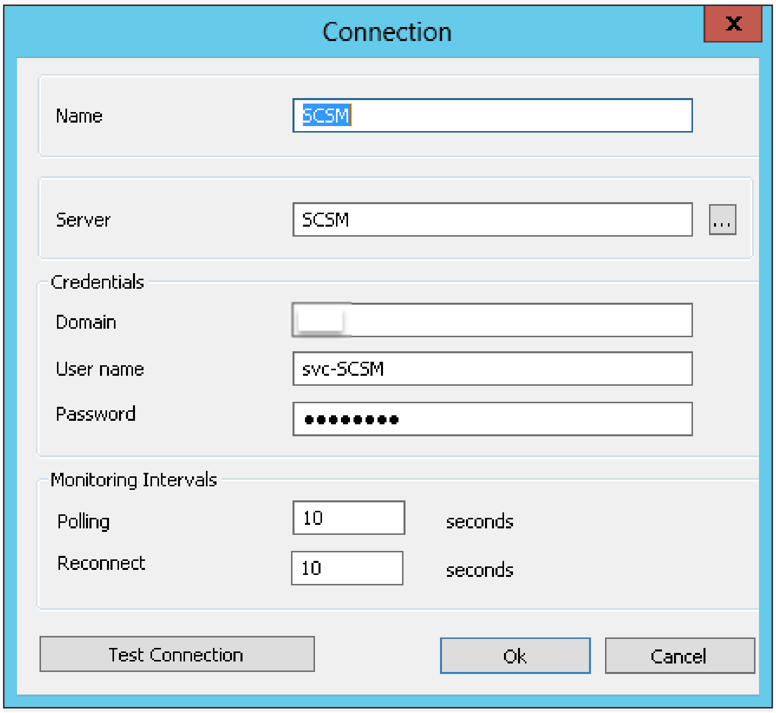
To set up a SQL Server configuration
1. In the Runbook Designer, click the **Options** menu, and select *KA SQL Server*. The **KA SQL Server** dialog box appears.
2. On the **Configurations** tab, click **Add** to begin the configuration setup. The **Add Configuration** dialog box appears.
3. In the **Name** box, enter **PDS_LIVE** as the name for the configuration.
4. Click the ellipsis button (...) next to the **Type** box and select *SQL Server Options.*
5. In the **Server Name** box, type the name of the SQL Server host.
6. Optionally, in the **Database Name** box, type the name PDS_LIVE as the database you want to connect to. If you leave the **Database Name** box empty, the activities in the integration will automatically include a cascading **Database Name** property that you can use to select a target database.

7. In the **Authentication Schema** box, select the method used to authenticate the connection.
8. If *SQL Server Authentication* was selected, in the **User name** and **Password** boxes, type the credentials that Orchestrator will use to connect to the SQL Server database
9. Click **OK** to close the configuration dialog box, and then click **Finish.**

Microsoft System Center 2012 Service Manager IP:

In the Options Menu of Runbook Designer, find the SC 2012 Service Manager settings. Create a connection called (SCSM), which is required for the IP operation.

Create a connection called SCSM and set the Server and authentication details to values appropriate to your environment i.e.



Name: SCSM
Server: SCSM
Domain:
User name: svc-SCSM
Password: ●●●●●●●●●
Polling: 10 seconds
Reconnect: 10 seconds

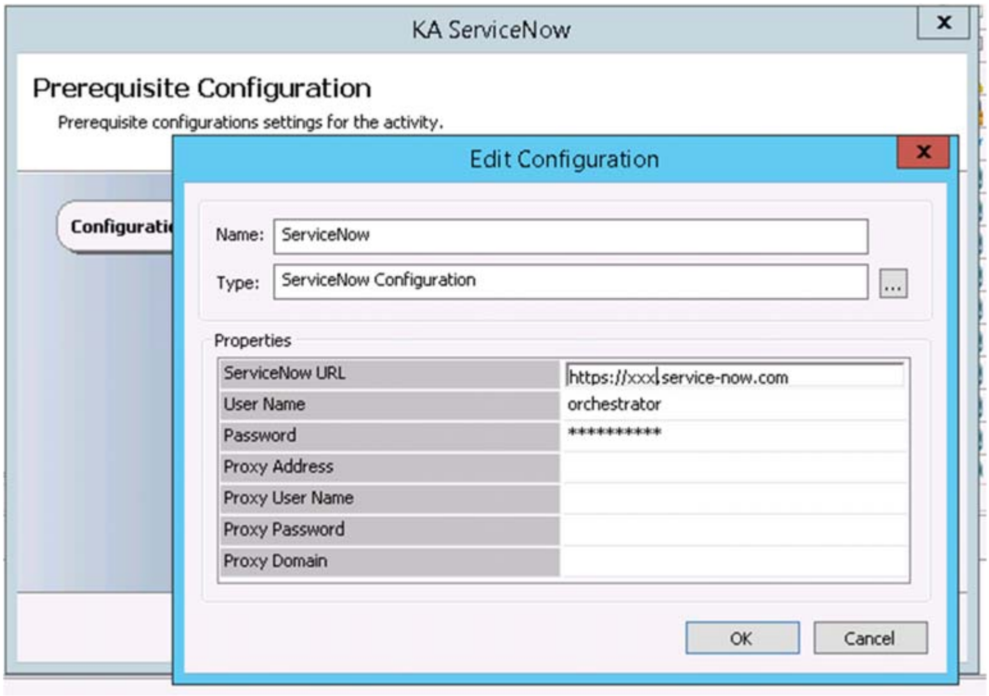
To set up a SC 2012 Service Manager configuration
1. In the Runbook Designer, click the **Options** menu, and select *SC 2012 Service Manager*.
The **SC 2012 Service Manager** dialog box appears.
2. On the **Configurations** tab, click **Add** to begin the connection setup. The **Configuration Entry** dialog box appears.
3. In the **Name** box, type **SCSM** as the name for the connection.

4. In the **Server Name** box, type the name of the SCSM Management Server host.
5. In the **Domain**, **Username** and **Password** boxes, type the credentials that will be used to connect to Service Manager.
6. Click **OK** to close the configuration dialog box, and then click **Finish**.

KA ServiceNow IP:

To set up a SC 2012 ServiceNow configuration

1. In the Runbook Designer, click the **Options** menu, and select *KA ServiceNow*. The **KA ServiceNow** dialog box appears.
2. On the **Configurations** tab, click **Add** to begin the connection setup. The **Configuration Entry** dialog box appears.
3. In the **Name** box, type **ServiceNow** as the name for the connection.
4. In the **ServiceNow URL** box, type your ServiceNow URL.
5. In the **Username** and **Password** boxes, type the credentials that will be used to connect to ServiceNow.
6. Click **OK** to close the configuration dialog box, and then click **Finish**.


KA ServiceNow
Prerequisite Configuration
Prerequisite configurations settings for the activity.
Configurati
Edit Configuration
Properties
...
OK
Cancel

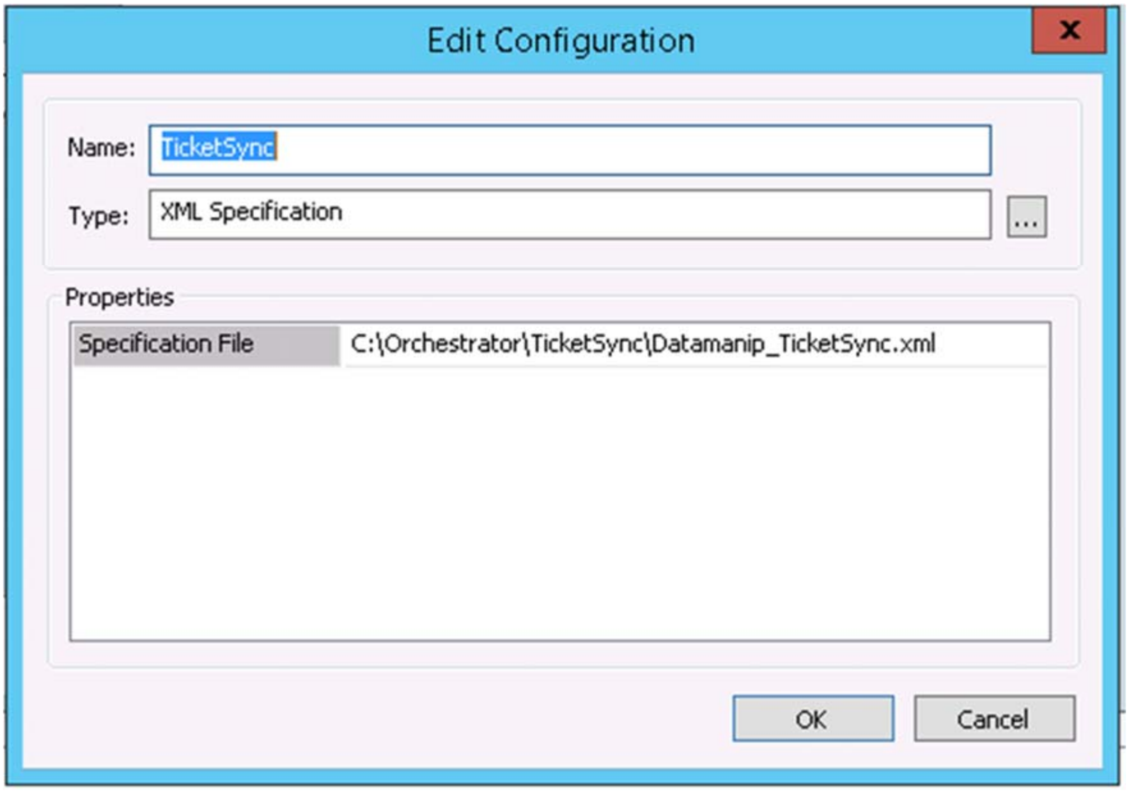
Name: ServiceNow
Type: ServiceNow Configuration

ServiceNow URL	https://xxx.service-now.com
User Name	orchestrator
Password	**********
Proxy Address	
Proxy User Name	
Proxy Password	
Proxy Domain	

KA Data Manipulation IP:

To set up a KA Data Manipulation configuration

1. In the Runbook Designer, click the **Options** menu, and select *KA Data Manipulation*.
The **KA Data Manipulation** dialog box appears.
2. On the **Configurations** tab, click **Add** to begin the connection setup. The **Configuration Entry** dialog box appears.
3. In the **Name** box, type **TicketSync** as the name for the connection.
4. In the **Specification File** box, locate the path of the data manipulation file i.e. *‘C:\Orchestrator\TicketSync\Datamanip_TicketSync.xml’.*
5. Click **OK** to close the configuration dialog box, and then click **Finish**.



Name: TicketSync

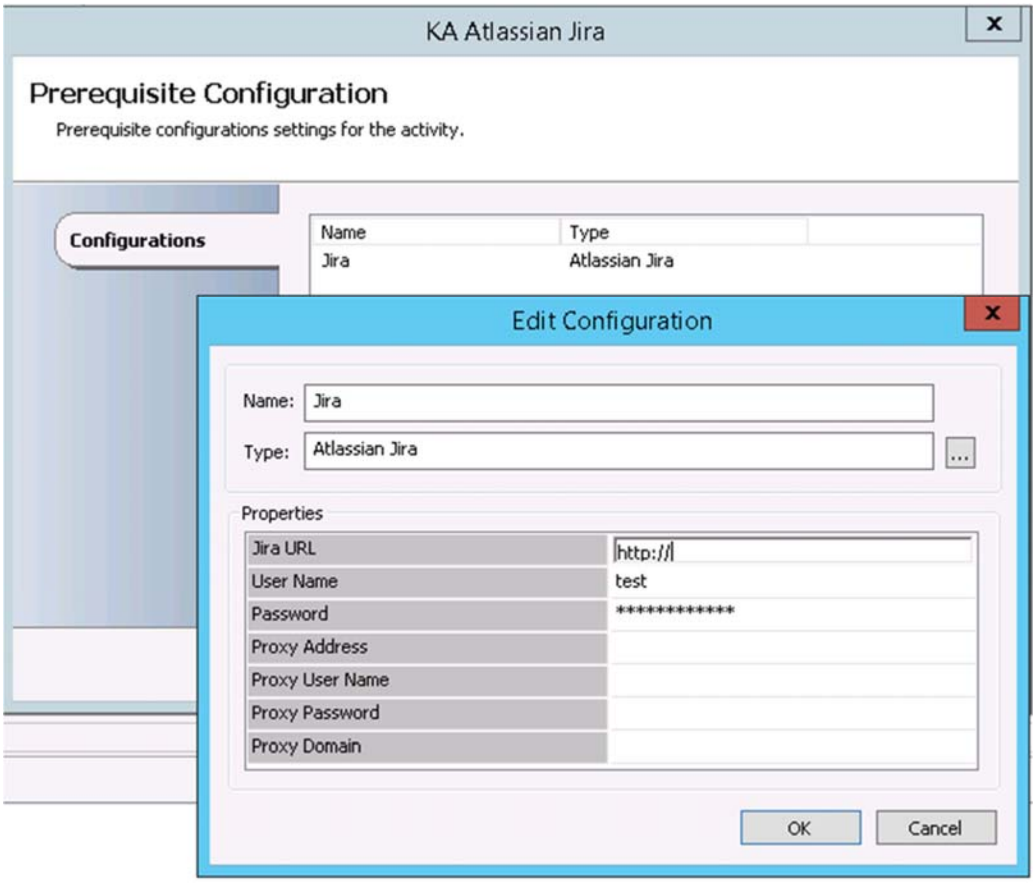
Type: XML Specification

Specification File: C:\Orchestrator\TicketSync\Datamanip_TicketSync.xml

KA Atlassian Jira:

To set up a KA Atlassian Jira configuration

1. In the Runbook Designer, click the **Options** menu, and select *KA* Atlassian Jira The **KA Atlassian Jira** dialog box appears.
2. On the **Configurations** tab, click **Add** to begin the connection setup. The **Configuration Entry** dialog box appears.
3. In the **Name** box, type **Jira** as the name for the connection.
4. In the **Jira URL** box, type your Jira URL.
5. In the **Username** and **Password** boxes, type the credentials that will be used to connect to Jira.
6. Click **OK** to close the configuration dialog box, and then click **Finish.**



Prerequisite Configuration

Prerequisite configurations settings for the activity.

Name	Type
Jira	Atlassian Jira

Name: Jira

Type: Atlassian Jira

Properties

Jira URL	http://
User Name	test
Password	************
Proxy Address	
Proxy User Name	
Proxy Password	
Proxy Domain	

6. Customising the Ticket and DevOps Synchronisation Solution

6.1. Default ‘Out of the Box’ behaviour

A customer may wish to use a different Service Desk product to drive or target the Ticket and DevOps Synchronisation Solution.

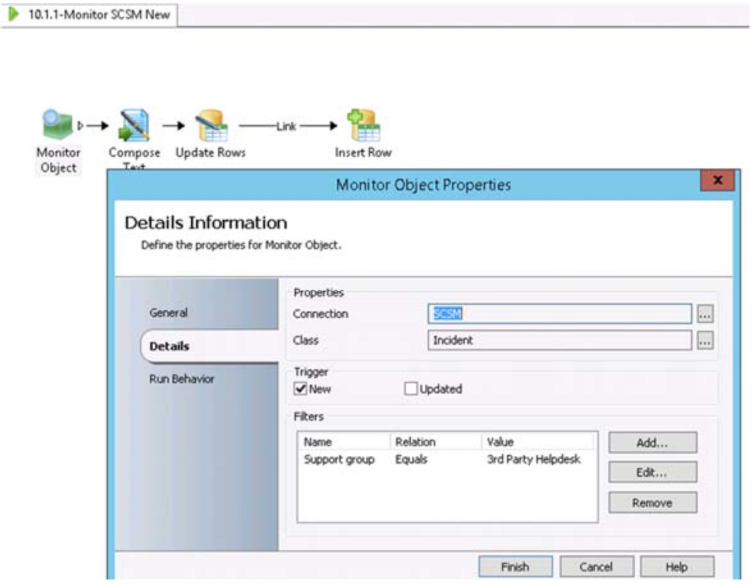
The Runbook sets are designed as a solution which provides the foundations out of the box and allows extension and modification to tailor the solution to exact customer requirements. It is expected therefore that a customer may wish to extend or customize the solution to talk to different Service Desks.

Out of the box, the Solution is configured to target ServiceNow. Using the Variable ‘TargetServiceDesk’

Section 6.4 details how the solution can also be configured to process Incidents between Service Manager and Jira.

6.2. Triggering the Runbooks

The runbooks are triggered on ‘New’ Service Manager Incidents where the Support Group = ‘3rd Party Help Desk’. It is expected that this should be adjusted/removed as required. It can be done as follows:



Connection: SCSM

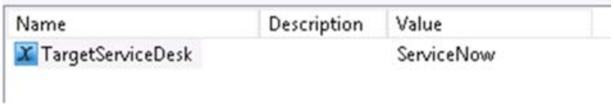
Class: Incident

☑ New ☐ Updated

Name	Relation	Value
Support group	Equals	3rd Party Helpdesk

6.3. Configuring the Runbooks to a specific Target Service Desk

To change the target Service Desk from 'ServiceNow', set the *'TargetServiceDesk' variable* Value to the appropriate Runbook Folder Name e.g. *'BMCRemedy'*



Name	Description	Value
TargetServiceDesk		ServiceNow

As the screen shot above shows, the Solutions currently have 'ServiceNow' configured as the Target Service Desk.

To extend the solution to use a new Target Service Desk, the new Target Runbooks would need to be built in the *'30.0 Target Helpdesk'* folder in line with the Service Manager Runbook folders. The Variable value could then be changed to refer to the new Target Runbook Folder Name.

These Runbooks are used as templates and should be copied to create Runbooks for your target Service Desk.

Out of the box, the Runbooks are provided to process Incidents between Service Manager and ServiceNow. Section 6.4 details how the solution can also be configured to process Incidents between Service Manager and Jira.

6.4. Configuring the Solution to Target 'Atlassian Jira'

Out of the box, the Runbooks are provided to process Incidents between Service Manager and ServiceNow. As well as providing integration into ServiceNow, the solution, with a measured amount of effort and tailoring, can be configured for Atlassian Jira. This requires a sequence of steps:

1. Stop the runbooks

2. Replace the default value of the variable 'TargetServiceDesk' from 'ServiceNow' to 'Jira'.

3. Ensure there is connectivity between Orchestrator and Jira after configuring the steps in section 5.1

4. *Configure the Jira 'Create Issue' Activity:* Open the Jira 'Create Issue' activity in Runbook '*32.1.1-Create Jira Issue*'. Using the same methodology applied for ServiceNow in runbook '*31.1.1-Create ServiceNow Incident*', subscribe the appropriate published data (i.e. Description) from the appropriate activities (i.e. 'Parse text.').

   Open the KA 'Run Procedure' activity. In the field 'SourceID', replace the hardcoded placeholder '*{<Issue ID from Create Issue>}*' with the 'Issue ID' subscribed directly from the 'Create Issue' activity.

   Open the KA 'Update Incident' activity. In the field 'TargetID', replace the hardcoded placeholder '*{<Issue ID from Create Issue>}*' with the 'Issue ID' subscribed directly from the 'Create Issue' activity.

5. *Configure the Jira 'Update Issue' Activity:* Repeat the same principles as above, where applicable, for the 'Update Issue' activity in the runbook *'32.1.2-Update Jira Issue'.*

6. *Configure the Jira 'Monitor Issue' Activity:* Open the Jira activity 'Monitor Issue' in the Runbook '*32.1.3-Monitor Jira Issue Resolved*'. Using the same methodology applied for ServiceNow in runbook '*31.1.3-Monitor ServiceNow Incident Resolved*', configure the filter.

Open the KA ‘Run Procedure’ activity. In the field ‘SourceID’, replace the hardcoded placeholder ‘*{<Issue ID from Monitor Issue>}’* with the ‘Issue ID’ subscribed directly from the ‘Monitor Issue’ activity.

7. *Configure the KA ‘Compose Text’ Activities:* Open the ‘Compose Text Jira’ activity in Runbook ‘*32.1.1-Create Jira Issue’*. Using the same methodology applied for ServiceNow in runbook ‘*31.1.1-Create ServiceNow Incident*’, subscribe the appropriate published data (i.e. Description) from the ‘Create Issue’ activity. If any properties are required to be added, removed or altered, use the ‘Datamanip_TicketSync.xml’ configuration file to achieve this.

   Repeat the same principles for the ‘Compose Text’ activity in the runbook *‘32.1.3-Monitor Jira Issue Resolved.*

8. Start the runbooks

7. Installing Temporary License of Kelverion Integration Packs

To run the solution you will need a full or evaluation licence key for Kelverion Integration Packs.

The licence files need to be copied into a folder called C:\Program Files (x86)\Kelverion Automation\Licenses. If this folder does not already exist on your system please first create the folder C:\Program Files (x86)\Kelverion Automation\Licenses and then copy the attached files into it.

The license key is regularly updated as it includes a specific license end date after which the product will no longer work. If you have a license or date format error on trying to run this product please contact info@kelverion.com detailing date of download and error details.

To purchase a license please contact your Kelverion representative, reseller or email info@kelverion.com

8. Upgrade Warning

The Runbooks provided in this Ticket and DevOps Synchronisation solution are provided for installation in a clean Orchestrator environment. If you have deployed any previous versions of this Runbook Solution then installing this version will overwrite any changes you have made to the currently deployed Runbooks.

You can either delete you existing Runbook deployment and then install this new Runbook Solution set or manually upgrade your existing deployment.

9. Notes

Kelverion Automation Ltd
65 High Street
Harpenden
Hertforshire, Al5 2SL
Email: info@kelverion.com
Web: www.kelverion.com