Microsoft System Center Orchestrator 2012 R2 Essentials

With the increasing complexity of systems in our infrastructure designs, administrating these systems has become more complex and repetitive. This ultimately translates into an inefficient environment and a loss of time. System Center Orchestrator 2012 R2 is the tool that can help you regain that time again, by simplifying repetitive tasks (or complex ones) into a simple one that will require minimum intervention from you, or even none.

Starting off with the fundamentals of Orchestrator, this book helps you to learn the configuration and deployment of the Orchestrator 2012 R2 in a corporate environment to successfully deploy and mitigate real life issues that you might encounter. You’ll learn the most used part of Orchestrator, the Runbook Designer, with all its components and mechanisms to help you efficiently design and implement your Runbooks from top to bottom. You finally round off by gaining insights on the maintenance aspects of Orchestrator, while simultaneously covering the troubleshooting, backup, and recovery features.

Who this book is written for
This book is targeted at those who want to gain time in their administrative tasks and profit from it to also improve their environment by reaching a more autonomous infrastructure.

What you will learn from this book
- Plan, assess, design, and implement Orchestrator 2012 R2 in a corporate environment
- Master the Runbook Designer and discover all about the Runbook creation process
- Import and integrate more technologies through new imported functionalities and learn what they bring to your environment
- Extend your possibilities and interaction with your environment while you get to grips with the best practices for this system
- Implement High Availability, security, backup, and recovery procedures and fine-tuning of the platform
- Familiarize yourself with the Runbook design and its implementation with an insight into the methodology for a more dynamic data center

Miguel Oliveira

Microsoft System Center Orchestrator 2012 R2 Essentials
Design, implement, and improve your infrastructure administration with System Center Orchestrator 2012 R2’s automation process

In this package, you will find:

- The author biography
- A preview chapter from the book, Chapter 1 'Configuring and Deploying Orchestrator 2012 R2'
- A synopsis of the book’s content
- More information on Microsoft System Center Orchestrator 2012 R2 Essentials
About the Author

Miguel Oliveira is a versatile IT architect, system engineer, and information security enthusiast who has been living in Switzerland since 2012; before then, he lived in Portugal. During that time, Miguel worked mostly as an IT consultant, having worked with the major companies in Portugal (namely Novabase and Microsoft) with which he designed, developed, and implemented several System Center solutions across different clients, from the banking/financial industry to energy/oil companies. In Switzerland, he worked for international organizations and for a private company as an infrastructure architect and as a senior systems engineer at the time of publishing this book.

Miguel's experience in IT started early as huge curiosity developed in him when he was very young, and it hasn't stopped ever since. The will to develop his knowledge was so strong that at the age of 16, he had his own server at home, and with it, he achieved the first step of his career as an IT systems administrator in 2003, making him an expert on Microsoft products today. He also has good knowledge of VMware, Unix, Network, and Security, which he uses in his daily tasks as an infrastructure architect providing insight and expertise for solutions that are requested by his company.

This book is Miguel's first ever published book, and it has been an amazing and interesting adventure.
Preface

In the IT administration world, there has always been a huge demand for quicker ways of doing more with less (either time or material). An IT admin is constantly confronted with requests that either become a routine task and end up increasing their workload, or are just requests that bring a bit more inertia to the daily administrative process by adding either another component to manage, or requests to approve moving forward by looking into another console to monitor in order to take action to move forward. With technology always evolving and becoming more dynamic, IT administration becomes more difficult and demanding, which leads to newer software, tool implementation that will possibly need operations and tasks to be performed on a daily basis and information to be processed and debugged at several levels, exhausting the IT personnel resources on these administrative tasks and, or recurring actions that derivate from all this.

Microsoft System Center Orchestrator 2012 R2 can bring added value and make the IT admin's life easier by cutting short recurrent actions and allowing the IT admin to be focused on other tasks while Orchestrator carries on with those recurrent tasks. We identify and orchestrate these tasks through a workflow in order to make it more autonomous and only intervene when the workflow is not able to work out the task in it.
What is Microsoft System Center Orchestrator 2012 R2?

Microsoft System Center Orchestrator (SCORCH or Orchestrator, from now on) is a workflow automation software that allows IT administrators to automate monitoring and deployment tasks. Orchestrator's tasks are actually Runbooks that are designed through a drag-and-drop user interface and then translates them into .NET, PowerShell, or SSH commands to automate the tasks defined. It is possible to manage various System Center components, as well as AD and other technologies through the use of Integration Packs and can also be used to automate tasks in other operating systems.

This product comes with the Microsoft System Center Suite product family, following a name change from Opalis, which was acquired by Microsoft in December 2009.

We can integrate Orchestrator with other technologies through the use of Integration Packs or by using the Integration Toolkit. We can also develop our own Integration Pack and allow another technology to be managed by our Orchestrator environment, as well as interact with Orchestrator through a web service.

What this book covers

This book covers the many aspects of Orchestrator 2012 R2, including a successful deployment, Runbook design, best practices, how and what to use to develop your own Integration Pack, as well as how to integrate existing ones.

Chapter 1, Configuring and Deploying Orchestrator 2012 R2, covers the infrastructure design aspect of Orchestrator 2012 R2 in a corporate environment, from a simple deployment with an all-in-one server to a more redundant and complex deployment, while we address the prerequisites, firewall ports, and other aspects of the configuration.

Chapter 2, Runbook Designer, focuses on the Runbook Designer. This will be the most used part for the IT administrator, creating the workflows and automatisms, and therefore, we'll address everything there is to know about the Runbook Designer component of Orchestrator 2012 R2.

Chapter 3, Orchestrator Integration Packs, covers the Orchestrator Integration Packs, how they work, and what they are, as well as importing some to our environment for the upcoming chapters.
Chapter 4, *Extending Orchestrator*, is all about how and where we’ll address the Orchestrator capabilities of expansion and the services that surround it. We’ll address, in more detail, the different ways that Orchestrator can be accessed and through which methods it can be extended in its capabilities. We’ll also verify in more depth the best practices for these expansions, impacts, and troubleshooting scenarios.

Chapter 5, *Runbook Examples*, covers a series of sample Runbooks that are going to be explored and will serve as a base for you to create your own for the tasks you’ll need to cover in your environment. The Runbooks that we’re going to address at this point are going to be interconnected with their Integration Packs directly, therefore giving you a good understanding of how to work with them.

Chapter 6, *Maintaining an Orchestrator Infrastructure*, talks about maintenance being necessary to keep our Orchestrator happy and running. Troubleshooting will also be addressed, as well as backup and recovery.
Configuring and Deploying Orchestrator 2012 R2

This chapter will target the goal of having a proper Microsoft System Center Orchestrator 2012 R2 (SCORCH or Orchestrator) architecture in place. We’re going to cover the Orchestrator components, their capability to be scalable, and their requirement to better address the question of the design to be chosen. By the end of this chapter, you will be able to identify what requirements you should keep in mind and assess before drawing or implementing Orchestrator.

Microsoft System Center Orchestrator 2012 R2 is a workflow automation product that interconnects with multiple technologies. With Orchestrator, you can target the following points in your datacenter:

- Automation of repeated tasks (either system or manually triggered)
- Interaction between products

The expected result of the automation of repeated tasks is for the administrative workload to be reduced, thereby improving the quality and performance of the IT staff and also improving on the datacenter dynamics.

In this opening chapter, we will address the following points:

- Planning the Orchestrator architecture and deployment
- Deploying a single-server solution
- Deploying a multiple server solution
An overview of the Orchestrator components

To better understand Orchestrator, we need to know the basics of it. We can start off with the fact that it has seven components. The automated workflow is called a Runbook. The seven components are as follows:

- **Management Server**: This component coordinates the communication in the Orchestrator architecture between all the other components and it cannot be redundant as there can be only one Management Server per Orchestrator deployment.

- **Orchestrator Database**: As the name suggests, this is composed of a database that will be stored in a Microsoft SQL Server, and it will keep all the Runbooks in it, the status of those Runbooks, and keep the security delegation in place.

- **Runbook Server**: This is the component responsible for executing the Runbooks instances. When a Runbook is invoked, a copy of it is transmitted from the database into the Runbook Server, and then it's executed. The first Runbook Server to get installed gets the primary role.

- **Runbook Designer**: This component is the interface between you and the Orchestrator engine for creating the Runbooks. It has a subcomponent called Runbook Tester for you to use in order to validate the Runbooks you create.

- **Orchestrator Console**: This is a console that runs on Silverlight that uses the Orchestrator Web Service to communicate with Orchestrator.

- **Orchestrator Web Service**: This is the interface that allows applications to connect with Orchestrator and allows them to perform tasks through it, such as starting and stopping Runbooks, or viewing the status.

- **Orchestrator Deployment Manager**: The Deployment Manager is a tool used to deploy Integration Packs (IPs), Runbook servers, and Runbook Designers. This is a subcomponent of the Management Server.

Planning the Orchestrator architecture

The architecture of Orchestrator is, at its base, quite simple and light. At the very beginning you’ll find yourself in need of a multiple server Orchestrator deployment for performance and High Availability, and in the end, apart from the High Availability, you won’t feel the need for a big Orchestrator infrastructure.
At this point, we'll address a single-server architecture with the database in a separate SQL cluster. For this, the requirements are as follows:

- RAM: 1 GB (recommended 2 GB)
- Disk: 200 MB (available disk space)
- CPU: Dual-core Intel CPU, 2.1 Ghz (or above)
- OS: Windows Server 2008 R2 (with IIS, .NET 3.5 SP1, .NET 4.5, and Silverlight) or above
- SQL Server: Microsoft SQL Server 2008 R2 or above (with Collation: `SQL_Latin1_General_CP1_CI_AS` and Database Engine Service only)

As you can see, the requirements for the Orchestrator architecture following a single-server deployment are not really that demanding. However, let's observe the requirements per component for a multi-server deployment:

- Management server:
  - RAM: 1 GB (recommended 2 GB)
  - Disk: 200 MB (available disk space)
  - CPU: Dual-core Intel CPU, 2.1 Ghz or above
  - OS: Windows Server 2008 R2 (.NET 3.5 SP1) or above

- Orchestrator database:
  - SQL Server: Microsoft SQL Server 2008 R2 or above
  - Collation: `SQL_Latin1_General_CP1_CI_AS`
  - SQL Roles: Database Engine
  - Location: Local or Remote (cluster supported)

- Runbook server:
  - RAM: 1 GB (recommended 2 GB)
  - Disk: 200 MB (available disk space)
  - CPU: Dual-core Intel CPU, 2.1 Ghz or above
  - OS: Windows Server 2008 R2 (.NET 3.5 SP1) or above

- Runbook designer:
  - RAM: 1 GB (recommended 2 GB)
  - Disk: 200 MB (available disk space)
  - CPU: Dual-core Intel CPU, 2.1 Ghz or above
  - OS: Windows Server 2008 R2, Windows 7 SP1 or above
• Orchestration console / Web service server:
  - RAM: 1 GB (recommended 2 GB)
  - Disk: 200 MB (available disk space)
  - CPU: Dual-core Intel CPU, 2.1 Ghz or above
  - OS: Windows Server 2008 R2 (.NET 3.5 SP1, .NET 4.5, IIS, Silverlight) or above

Usually, the multi-server deployments are in fact connected to the High Availability questions, but keep with mind that the Management Server is a single component that can't be deployed in High Availability, even though the Runbook Servers are able to execute the Runbooks without that component.

On the **Active Directory (AD)** side, you'll need to also fulfill some requirements:

<table>
<thead>
<tr>
<th>User/Group</th>
<th>Name</th>
<th>Type</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>DOMAIN\scorchmssvc</td>
<td>Service account</td>
<td>This is the main management server's service account and it will be granted a logon as a service during the installation. It is also used for the Monitor service.</td>
</tr>
<tr>
<td>User</td>
<td>DOMAIN\scorchrbsvc</td>
<td>Service account</td>
<td>This is the account used for accessing resources that are specified on the Runbooks. This account might require access to remote computers.</td>
</tr>
<tr>
<td>Group</td>
<td>DOMAIN\Scorch_Users</td>
<td>Group</td>
<td>Users Security Global Group for allowing default access to Orchestrator</td>
</tr>
<tr>
<td>Group</td>
<td>DOMAIN\Scorch_Ops</td>
<td>Group</td>
<td>Operators Security Group to allow access as the Runbook operator on Orchestrator</td>
</tr>
<tr>
<td>Group</td>
<td>DOMAIN\Scorch_Admins</td>
<td>Group</td>
<td>This is the Full Administration Security Group for Orchestrator</td>
</tr>
</tbody>
</table>
The next screenshot shows these accounts and groups in Active Directory, and in the end you should have something quite similar in your environment. The users and groups don't need to be named exactly the same as the examples we give here; they can be adapted to your standards, but keep these in mind for your reference for the rest of this book.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>scorchrbsvc</td>
<td>User</td>
<td>Orchestrator Runbook Server Service Account</td>
</tr>
<tr>
<td>scorchmssvc</td>
<td>User</td>
<td>Orchestrator Management Server Service Account</td>
</tr>
<tr>
<td>Scorch_Users</td>
<td>Group</td>
<td>Orchestrator Console Users</td>
</tr>
<tr>
<td>Scorch_Ops</td>
<td>Group</td>
<td>Orchestrator Runbook Creators</td>
</tr>
<tr>
<td>Scorch_Admin</td>
<td>Group</td>
<td>Orchestrator Full Admins</td>
</tr>
</tbody>
</table>

On the network side, we'll need the following exceptions on the firewall:

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
<th>Default Port (TCP)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Runbook Designer</td>
<td>Management Server</td>
<td>135, 1024-65535</td>
<td>The Runbook Designer communicates with the management server over DCOM; by default, the DCOM uses port 135 to initiate the connection and then moves to a dynamic port between 1024 and 65535.</td>
</tr>
<tr>
<td>Management Server / Runbook Server / Web Service</td>
<td>Orchestration Database</td>
<td>1433</td>
<td>This takes care of the Microsoft SQL Server connectivity.</td>
</tr>
<tr>
<td>Client Browser</td>
<td>Orchestration REST-based web service</td>
<td>81</td>
<td>These are ports used for connecting to the Orchestration console.</td>
</tr>
<tr>
<td></td>
<td>Orchestration console</td>
<td>82</td>
<td></td>
</tr>
</tbody>
</table>

All the ports above can be configured to better suit your environment.
Now, in terms of the scalability of the Orchestrator, we can have redundancy in the Orchestrator Web Service / Console, Runbook Servers, and Orchestrator Database. However, we won't be able to have more than one Management Server.

You might consider having the components divided between servers due to service availability, but there's also the Runbooks workload expected and the number of simultaneous Runbooks that you'll have running to consider, as well as their importance.

Our advice would be to have one single-server deployed for the Runbooks development and preparation, and then for production, a redundant deployment of Runbook Server, Orchestrator Web Service / Console Server, and the Orchestrator Database under a SQL cluster. This way, you'll cover any need of simultaneously executing some Runbooks and High Availability.

Good! Now that we've defined all the requirements, and since all are in place, we'll do a sanity check so we can see if we're ready to go forward.

So, to move on to the next point, on installing the Orchestrator architecture you've decided to go with, you'll need to achieve the following steps:

1. Create Active Directory accounts:
   - Service accounts
   - AD groups

2. Install an operating system:
   - Windows Server (2008 R2 or above)
   - Add it to the domain
   - Add service accounts to the administrator's local group
   - Add the Orchestrator administrators group to the administrator's local group

3. Configure OS roles and features:
   - IIS
   - .NET 3.5 SP1
   - .NET 4.5
   - Silverlight

Don't forget that some of the OS roles and features are only required for some of the Orchestrator components. Check the list to see which component needs which piece of software.
Also, it's not on this list, but we're assuming you have a SQL cluster or a single-server in place to accommodate your Orchestrator database. Don't forget the permissions on that SQL server for the account running the Management Server.

The OS roles and features prerequisites can all be installed using PowerShell in order to achieve a faster single-server deployment. In order to do so, you can execute the following command:

```
Import-Module ServerManager
```

**Deploying a single-server solution**

As discussed in the previous section, we can use a single-server to host all the components of Orchestrator if we're targeting a small environment and/or we have the objective of having an Orchestrator for the development and quality assessment of our Runbooks before getting them into production. If you jumped straight into this part, we strongly suggest you read the *Planning the Orchestrator architecture* section before proceeding.

As of the time of writing, Windows Server 2012 R2 has been out for quite a while and it's the OS selected to host our Orchestrator. So, once you have chosen the OS and deployed it, and after having joined to the domain and having done the basic configurations, we can move on with deploying the prerequisites:

1. Verify that you have permissions on the SQL Server selected to host the database, that the collation is the one specified in the previous section, and finally that **Windows Authentication** is the authentication mechanism selected. If you choose to install the SQL Server locally, don't forget to get the **Database Engine Services options** and the share features by navigating to **Management Tools** | **Basic**.
2. Add the Orchestrator Administrators group and the Management Service account to the local Administrators group of the server.

3. Enable the **Web Server (IIS)** role, accept the features added, and proceed with the installation. The default settings of this role will suffice as the Orchestrator will configure it accordingly for you. If the Orchestrator installation doesn't detect this role on the server, it will install it for you.

4. Install the **.NET 3.5 SP1** feature. Don't forget to indicate the alternative source for the installation to occur.

5. (Only if you're installing on Windows Server 2008 R2): Install the **.NET 4.5** feature. If you forget to activate the **ASP.NET** and **HTTP Activation** features, the installer will take care of activating them for you.

6. Install **Silverlight**. If you start the console without installing it, you'll be required to install it.
7. Launch the installation using the SetupOrchestrator.exe. Select **Install** on the **System Center 2012 R2 Orchestrator Setup** section of the wizard.

![System Center 2012 R2 Orchestrator Setup](image)

8. On the **Product registration** page, enter your organization’s details and product key, and then click on **Next**.

9. Take your time to read the license terms, and if you accept them, continue the installation by clicking on **Next**.
10. On **Select features to install**, by default you’ll have all of them selected. As we’re installing a single-server, this is the desired selection, so just press **Next**.

11. As seen in step 4 of this installation, we didn't include the **ASP.NET** features on IIS, so Orchestrator will install them for us (press **Next**) and then it will confirm that everything’s OK.
12. At the **Configure the service account** step, we'll use the Management Service `scorchmssvc` account. The explanation on this screen can be a bit misleading, but this is the management service that we're talking about. In a multi-server deployment, this will be different. We'll get there…
13. Now, at **Configure the database server**, we'll use our server/instance (if any instance), and the port, and we'll verify the connectivity. If everything's OK, we'll move on with **Next**.
14. Verify the database settings at the **Configure the database** page and specify the database name if you want something different. The default is **Orchestrator**. Proceed with **Next**.
15. Now it's time to configure Orchestrators users group, but although the title of the page can be deceiving, this is not the normal users group, but rather the Administrators group. Click on Browse and select the proper group that you've defined (Scorch_Admins in this example). Click on Next when ready.

16. At the Configure the ports for the web services page, usually the default ports 81 and 82 are OK, but if these ports are not compliant with your environment or you wish to make the service available at some other port, this is the time to do it. If you do change the ports, don't forget to keep them in mind for the rest of this book as we'll use the default ones for every sample and configuration. Push Next.

17. Time to select the installation location. Usually, the default is OK, but, once again, you can specify another location that is more suitable for your needs. Push Next.

18. On the Help improve Microsoft System Center Orchestrator, select the options you prefer and click on Next.
19. Review **Installation summary**, and if everything's OK, click on **Install** to start the installation.

20. Once the installation finishes, you'll be given the status of the installation along with a few options to consider before finishing the setup:

   1. Launch Windows Update.
   2. Visit System Center Orchestrator Online.
   3. When the setup closes, start **Runbook Designer**.

All these steps can be automated with Powershell. The following is a sample of the command line to execute in order to install it in an unattended way with the Orchestrator in a single-server:

```
D:\Setup\Setup.exe /DbServer:SQLSERVER\INSTANCE
/DbNameNew:Orchestrator /WebServicePort:81 /WebConsolePort:82
/OrchestratorRemote /UseMicrosoftUpdate:1 /SendCEIPReports:0
/EnableErrorReporting:never /Components:all
/ServiceUserName:DOMAIN\scorchmssvc /ServicePassword:IWontTellYou
/OrchestratorUsersGroup:(Group's SID) /Key:XXXXX-XXXXX-XXXX-XXXX-
-XXXX /Silent
```

As we're setting up only one server, we're only going to use the Management Service account for the Management Service and for the Runbook Service. In this type of deployment, only one account is required. The Runbook Server service account will be used for any additional Runbook Servers that we may like to deploy in the future, and it's best practice to separate the two services into separate accounts to avoid a single point of failure in the future.

If you choose to go with the command-line installation, don't forget to retrieve the **Scorch Admns** group SID. If you don't want to specify the group, it will use the default one locally called **OrchestratorUsersGroup**. If you would like to install Orchestrator into another directory, you can use the **InstallDir()** function and specify the directory.

In any case, you need to verify the installation logs. They will be under

```
C:\Users\<USER>\AppData\Local\Microsoft System Center 2012\Orchestrator\Logs
```

With this complete, you will have a fully functional Orchestrator 2012 R2 deployment.
Deploying a multi-server solution

As mentioned previously in the Planning the Orchestrator architecture section, we can use a multi-server deployment to separately host the components of Orchestrator if we’re targeting a highly available environment or if we have a huge demand for Runbooks running at the same time. If you jumped straight into this part, we strongly suggest you read the Planning the Orchestrator architecture section before proceeding.

Management Server

On the multi-server solution, we have to start by deploying the Management Server along with the database. This is the main focus point of our Orchestrator architecture, so we have to start with it.

1. Verify that you have permissions on the SQL Server selected to host the database and that the collation is the one specified on the previous section, and, finally, that Windows Authentication is the authentication mechanism selected. If you chose to install the SQL Server locally, don't forget to get the Database Engine Services options, with the share features by navigating to Management Tools | Basic.

2. Add the Orchestrator Administrators group and the Management Service account to the local Administrators group of the server as explained above for the single-server instance.

3. Install the .NET 3.5 SP1 feature. Don't forget to indicate the alternative source for the installation to occur.

4. (Only if you're installing on Windows Server 2008 R2): Install the .NET 4.5 feature. You don't need to activate the ASP.NET as this is the Management Server, so it won't need it.

5. Launch the install using SetupOrchestrator.exe. Select Install on the System Center 2012 R2 Orchestrator Setup section of the wizard.

6. On the Product registration page, enter your organization's details and product key, and then click on Next.

7. Take your time to read the license terms, and if you accept them, continue the installation by clicking on Next.
8. **On Select features to install**, by default you'll have all of them selected. We're going to remove all of them except the required one—**Management Server**—and then proceed. Click on **Next**.

9. At the **Configure the service account** step, we'll use the Management Service (scorchmssvc) account, test, and proceed with **Next**.

10. Now, at **Configure the database server**, we'll use our server/instance (if any instance) and the port, and we'll verify the connectivity. If everything’s OK, we'll move on with **Next**.

11. Verify the database settings at the **Configure the database** page and specify the database name if you want something different. The default is **Orchestrator**. Proceed with **Next**.

12. In **Configure Orchestrator users group**, select the group defined earlier for the Orchestrators Admins function, in this case **TEST\Scorch_Admins**. After this is set, press **Next**.

13. Select the installation location that's most convenient for you and proceed with **Next**.
14. On the Microsoft Update screen, select the On (recommended) option and proceed with Next. It's very important to keep your Orchestrator up to date.

15. On Help improve Microsoft System Center Orchestrator, select the option you prefer and proceed with Next.

16. On Installation Summary, verify your options and proceed with Install if they're OK.

17. Once finished, you'll see a Setup completed successfully page with two options:

   - Launch Windows Update
   - Visit System Center Orchestrator Online

You'll then have finished your installation. We strongly suggest that you run the Windows Update option and update everything there is to update.

If you wish to install the Management Server component in an unattended way, don't forget to retrieve the Scorch_Admins group SID. If you don't want to specify the group, it will use the default one locally called OrchestratorUsersGroup. If you would like to install Orchestrator into another directory, you can use the InstallDir function and specify the directory.
In any case, you need to verify the installation logs. They will be under C:\Users\<USER>\AppData\Local\Microsoft System Center 2012\Orchestrator\Logs. You can do the unattended installation by executing the following commands:

```
Import-Module ServerManager
Add-WindowsFeature NET-Framework-Core -source D:\Sources\sxs
D:\Setup\Setup.exe /DbServer:SQLSERVER\INSTANCE
/DbNameNew:Orchestrator /OrchestratorRemote /UseMicrosoftUpdate:1
/SendCEIPReports:0 /EnableErrorReporting:never
/Components:ManagementServer /ServiceUserName:DOMAIN\scorchmssvc
/ServicePassword:IWontTellYou /OrchestratorUsersGroup:(Group's SID)
/Key:XXXXX-XXXXX-XXXX-XXXX /Silent
```

Runbook Server

Once you have deployed the Management Server, you can proceed with the next logical step on the installation, that is, the Runbook Server. The Orchestrator architecture needs at least one Runbook Server in order to function.

For the installation of the Runbook Server, these are the steps to take:

1. Have the Management Server installed and the Runbook Server Service account at hand (scorchrbsvc).
2. Add the Orchestrator Administrators group, the Management Service account, and the Runbooks Service account to the local Administrators group of the server.
3. Install the .NET 3.5 SP1 feature. Don't forget to indicate the alternative source for the installation to occur.
4. (Only if you're installing on Windows Server 2008 R2): Install the .NET 4.5 feature. You don't need to activate the ASP.NET as this is the Runbook Server, so it won't need it.
5. Launch the installation using the SetupOrchestrator.exe. Select Runbook Server under the Standalone installations in the System Center 2012 R2 Orchestrator Setup section of the wizard.
6. On the Product registration page, enter your organization's details and product key, and then click on Next.
7. Take your time to read the license terms, and if you accept them, continue the installation by clicking on Next.
8. Click on Configure the service account that was created for the Runbook Server service, in this case, scorchrbsvc. Proceed after testing by clicking on Next.
9. On the **Configure the database server** page, type the same server and instance (if applicable) that you used on the Management Server configuration. Click on **Next**.

10. On **Configure the database**, select the existing database (usually selected by default). Push **Next** to continue.

11. Click on **Select the installation location** and proceed with **Next**.

12. Regarding the **Microsoft Update** page, select your preference and click on **Next**.

13. On **Help improve Microsoft System Center Orchestrator**, select your preferences and proceed with **Next**.

14. Review the **Installation Summary**, and if everything's OK, click **Install** to start the process.

As mentioned previously, we recommend you take the time to get all the latest updates and then carry on with the installation of the other components.

If you prefer to run an unattended installation, you can do so by executing the following commands:

```
Import-Module ServerManager
Add-WindowsFeature NET-Framework-Core -source D:\Sources\sxs
D:\Setup\Setup.exe /DbServer:SQLSERVER\INSTANCE
/DbNameExisting:Orchestrator /UseMicrosoftUpdate:1 /SendCEIPReports:0 /EnableErrorReporting:never /Components:RunbookServer
/ServiceUserName:DOMAIN\scorchmssvc /ServicePassword:IWontTellYou
/Key:XXXXX-XXXXX-XXXX-XXXX-XXXXX /Silent
```

**Orchestrator console and web service**

After deploying both the Management Server and Runbook Server, you can deploy the console and web service getting the Orchestrator architecture finalized, given that the Runbook Designer will get installed on a Windows 7/8 client. There's no specific order to the installation of the components, except that the Management Server has to be always the first component to get installed.

To install the **Console and Web Service**, you'll need to execute the following steps:

1. Have the **Management Server** installed and the **Management Server Service account** at hand (**scorchmssvc**).
2. Add the Orchestrator Administrators group, the Management Service account, and the Runbooks Service account to the local Administrators group of the server.
3. Install the .NET 3.5 SP1 feature. Don't forget to indicate the alternative source for the installation to occur.

4. (Only if you're installing on Windows Server 2008 R2): Install the .NET 4.5 feature and activate the ASP.NET and HTTP Activation features. If you forget to activate this feature or don't want to activate it for now, Orchestrator will install it for you during the setup.

5. Install the Web Server (IIS) with the default settings.

6. Install Silverlight. If you don't install it now, when you get to the Console, it will request that you install Silverlight.

7. Launch the installation using SetupOrchestrator.exe. Select Orchestrator Console and Web Service under Standalone installations on the System Center 2012 R2 Orchestrator Setup section of the wizard.

8. On the Product registration page, enter your organization's details and product key, and then click on Next.

9. Take your time to read the license terms, and if you accept them, continue the installation by clicking on Next.

10. Click on Configure the service account that was created for the Management Server service. In this case, scorchmsssvc. Proceed after testing by clicking on Next.

11. On the Configure the database server page, type the same server and instance (if applicable) that you used on the Management Server configuration. Click on Next.

12. On Configure the database, select the existing database (usually selected by default). Push Next to continue.

13. At the Configure the ports for the web services page, usually the default ports 81 and 82 are OK, but if these ports are not compliant with your environment or you wish to make the service available at some other port, this is the time to do it. If you do change the ports, don't forget to keep them in mind for the rest of this book as we'll use the default ones for every sample and configuration. Push Next.

14. Click on Select the installation location and proceed by clicking on Next.

15. On Help improve Microsoft System Center Orchestrator, select your preferences and proceed with Next.

16. Review the Installation summary, and if everything's OK, click on Install to start the process.
17. If you want to install the Console and Web Service in an unattended way, you can do so by using the following commands:

```
Import-Module ServerManager

D:\Setup\Setup.exe /DbServer:SQLSERVER\INSTANCE
/DbNameExisting:Orchestrator /WebServicePort:81
/WebConsolePort:82 /OrchestratorRemote /SendCEIPReports:0
/EnableErrorReporting:never /Components:WebComponents
/ServiceUserName:DOMAIN\scorchmssvc
/ServicePassword:IWontTellYou /Key:XXXXX-XXXXX-XXXX-XXXX-XXXXX
/Silent
```

That's it for the Orchestrator architecture to be complete on the server side. Now all that's missing is the Runbook Designer that we will install on our Windows 8.1 (or 7) workstation on the multi-server solution.

## Runbook Designer

To achieve a successful installation on the Runbook Designer, you'll need to get .NET 3.5 SP1 and Silverlight installed. The Runbook Designer can be installed independently whether or not you have a Management Server and the proper database setup already, but it will be rather useless without it.

Before proceeding with this installation, please complete either a single-server or a multi-server implementation:

1. On the machine selected to install the **Runbook Designer**, enable the feature .NET 3.5 SP1 (if not already enabled).
2. Install **Silverlight**. If you don't install it now, you'll be prompted to install it if you have access to the console.
3. Launch the installation using the SetupOrchestrator.exe. Select Runbook Designer under the Standalone installations on the System Center 2012 R2 Orchestrator Setup section of the wizard.

4. On the Product registration page, enter your organization's details and product key, and then click on Next.

5. Take your time to read the license terms, and if you accept them, continue the installation by clicking on Next.

6. Click on Select the installation location and proceed with Next.

7. On Help improve Microsoft System Center Orchestrator, select your preferences and proceed with Next.

8. Review Installation Summary, and if everything's OK, click on Install to start the process.

9. Once the installation completes, you'll be given the status of the installation along with a few options to consider before finishing the setup:
   1. Launch Windows update.
   2. Visit System Center Orchestrator Online.
   3. When Setup closes, start the Runbook Designers.

10. As with all the other components of Orchestrator, you can also run the Runbook Designer installation in an unattended fashion. For that, you'll need to execute the following commands:

```
Import-Module ServerManager
Add-WindowsFeature NET-Framework-Core -source D:\Sources\sxs

D:\Setup\Setup.exe
/SendCEIPReports:0/EnableErrorReporting:never
/Components:RunbookDesigner /Key:XXXXX-XXXXX-XXXX-XXXX-XXXX
/Silent
```
When you start the Runbook Designer for the first time from a remote location (a workstation or server that's not the Management Server), you'll be presented with an error message saying **Could not find Orchestrator Management Service on 'WORKSTATION'.**

Confirm with **OK**, and then click on **Actions**. Select **Connect...** and insert the **Management Server** name into the box.
In the **Connection** window, insert the name of the Management Server that you'll be connecting to. This is a one-time setup for your user.

![System Center 2012 R2 Orchestrator Connection](image)

Once connected, you'll see **Runbook Designer** completely available.

![System Center 2012 R2 Orchestrator Runbook Designer](image)

Before you can use the **Orchestrator Console** option in **Runbook Designer**, you'll have to define it. To do so, go to **Options | Orchestrator Console** and insert the proper URL.
In our example here (with the multi-server implementation), the URL is http://PACKTSCORCHW:82.

After that, press Finish and you'll have everything configured as it should be.

Summary

So far, we've seen how to plan and deploy our Orchestrator architecture and the requirements and prerequisites for deploying successfully each component of Orchestrator in a single-server environment or multi-server environment.

All the actions completed on this chapter are essential for the successful usage of Orchestrator in your environment, so we recommend you plan your Orchestrator architecture thoroughly.

Welcome to Orchestrator! The journey has just begun...
Where to buy this book

You can buy Microsoft System Center Orchestrator 2012 R2 Essentials from the Packt Publishing website.

Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.

Click here for ordering and shipping details.