

Altair® Monarch Server™ v2023.0
INSTALLATION GUIDE

TABLE OF CONTENTS

Monarch Server System — Server Installation	1
Software Requirements.....	1
Hardware Requirements	2
Recommended Software Versions.....	3
Before You Begin	3
Distributable and Non-distributable Components.....	3
Database Server Preparation	4
Installer Preparation.....	5
Monarch Server Installation	5
Single Server Installation	6
Distributed Installation.....	19
Server Post-Installation Actions	21
Configuring SSO	21
Setting User Session Timeout.....	25
Setting Administrator Password and E-mail Address	25
Client Installation	26
Software Requirements.....	26
Installation Maintenance	27
Adding or Removing Components	27
Repairing a Damaged Installation	30
Removing All System Components.....	30
Upgrading Installation	32
Upgrading Using Multi-installer	32
Configuring Software to Work with Monarch Server	39
Configuring Web Services.....	39
Configuring Web Applications	44
Client.....	44
RMSCClient.....	44
Admin.....	45
System Configuration	46

Starting Configurator	46
Connecting to Configurator	46
Backing Up CONFIG.XML	48
Configuring Instances	49
Creating an Instance	49
Creating a New Database	52
Updating Database	53
Managing Locations	54
Migrating Data	55
RMS Migration	57
Unlocking System Administrator	57
Configuring Servers	59
Managing Server Modules	64
Configurator Settings	68
Global Settings	69
Configuring Global License Settings	71
Configuring Modules	76

MONARCH SERVER SYSTEM — SERVER INSTALLATION

This document describes how to install *Monarch Server* system components on your servers.

SOFTWARE REQUIREMENTS

Server software:

- .NET framework 4.8
- SQL Server 2016/2017/2019 (if the Monarch Server database is present on the local machine)

NOTE You have the option to deploy SQL Server Express 2019 and SQL Server Management Studio 18 (64-bit installation only).

SQL Server Express is only suitable for RMS or Automator; it is not viable for instances servicing Report Warehouse functionality.

If SQL Server is not installed on the Common Components server, then a basic installation of SQLClient and SQLCMD is required.

Operating systems:

- Windows Server 2016 (IIS 10.0)
- Windows Server 2019 (IIS 10.0)
- Windows Server 2022 (IIS 10.0)

NOTE Core versions of the above operating systems are NOT supported.

Additional system software components:

The following components are installed automatically by the installation program if they are not present on the machine. You may consider installing them at the most convenient time (for example, a scheduled maintenance window) as the installation may require system reboots.

- .NET framework 4.8
- Microsoft Access 2013 Runtime
- Microsoft SQL Command Line Utilities 11 for SQL Server (if the Monarch Server database is not present on the local machine)
- Microsoft ODBC Driver 11 for SQL Server (if the Monarch Server database is not present on the local machine)

- ❑ SQL 2017 Local DB

NOTE You can download SQL 2017 Local DB 64-bit from <https://download.microsoft.com/download/E/F/2/EF23C21D-7860-4F05-88CE-39AA114B014B/SqlLocalDB.msi>.

Client software:

- ❑ One of the following browsers:
 - Mozilla Firefox
 - Google Chrome
 - Microsoft Edge (Chromium)

NOTE Each release of Monarch Server is tested with the latest publicly available browser versions available at that time. Due to the frequency of browser version updates, Altair does not include specific browser version numbers in the system requirements. In case of changes in browser functionality affecting UI functionality, Altair will address the same as part of a subsequent patch release.

HARDWARE REQUIREMENTS

Processors:

- ❑ **Recommended:** Multi-core or multi-processor Intel Xeon 2 GHz w/ 2 MB cache (or better)

Memory:

- ❑ **Recommended:** 16 GB RAM or greater

Network Adapter:

- ❑ **Recommended:** Ethernet 100 Mbps or higher (Gigabit recommended)

Hard drive:

- ❑ **Recommended:** 100 GB or more (depending on processing requirements, report sizes, number of active concurrent users, mining types). Disk performance has a significant influence on application scalability and viewing response.

- NOTE**
- ❑ Additional disk space may be required if leveraging Monarch Classic-based processes. The requirement depends on the number of concurrent processes and the data volumes being processed by each and subsequent configurations of the SDK Database. Altair Support can assist with identifying the space considerations in such instances.
 - ❑ Monarch Server is a distributed application. Typically, better functionality is obtained by deploying 1+ servers as opposed to increasing the resource of a single server. Altair Support can assist with advising on options to address aspects such as performance, failover, and high availability.

Monitor:

- ❑ **Recommended:** Screen resolution of 1920 × 1080 (or 1366 × 768 for wide screens)

RECOMMENDED SOFTWARE VERSIONS

The following software versions are recommended for the Monarch Server System.

SOFTWARE	VERSION
Metadata database: MS SQL Server	2019
Application server: <ul style="list-style-type: none">• Operating System• IIS• .NET Framework	<ul style="list-style-type: none">• Windows Server 2019• 10.0• 4.8
For end users: Browser	Google Chrome

BEFORE YOU BEGIN

Before you begin the installation process, you should plan the setup of your Monarch Server System in a manner suited for your tasks.

Distributable and Non-distributable Components

Monarch Server System is a *distributable* system. This means that different *components* of the system can be installed on different machines within a network. There are two types of components: distributable and non-distributable.

Each of the non-distributable components can be installed only once per Monarch Server system.

Distributable components can be installed several times on different computers to form a desired configuration of the Monarch Server system that is tailored to your requirements.

The system consists of five main components:

- ❑ Configuration Service
- ❑ Agent Service
- ❑ Services
- ❑ Web Applications (Admin, Client, and RMSClient)
- ❑ Repository

The **Configuration Service** is a non-distributable component. Install only one instance of the **Configuration Service** per sold copy.

The **Agent Service** is a distributable component. It should be installed on every server used for MS installation. All instances of **Agent Service** use one **Configuration Service**.

The **Agent Service** launches the following services:

- Datawatch.Config.ConfigServer.exe
- Datawatch.ModulesManagement.Agent.exe
- Datawatch.RemoteServices.AdminServicesHost.exe
- Datawatch.RemoteServices.ClientServicesHost.exe
- Datawatch.RemoteServices.FilingDispatcherServiceHost.exe
- Datawatch.RemoteServices.MaintenanceDispatcherServiceHost.exe
- Datawatch.RemoteServices.NotificationsDispatcherServiceHost.exe
- Datawatch.RemoteServices.VolumeServicesHost.exe
- Datawatch.RemoteServices.DataPumpDispatcherServiceHost
- Datawatch.RemoteServices.DataPumpServicesHost
- Datawatch.RemoteServices.DataPumpStorageServicesHost
- Datawatch.ModulesManagement.Agent.exe
- Datawatch.RemoteServices.RMSClientServicesHost.exe
- Datawatch.RemoteServices.RMSContentServiceHost.exe

These services can run on different servers.

The **Web Applications** is a distributable component. It can be installed several times on different computers according to the number of instances.

The **Repository** is a distributable component. It can be installed on different computers. In most cases, only one instance of **Repository** should be installed.

Database Server Preparation

When the Database Service component is chosen during the installation, the setup will create the Monarch Server database on the database server. You will need to have a special database user that will be used by the Monarch Server setup for database creation and under which Monarch Server will further connect to the database.

SQL Server

There are two levels of security that may be set for the Monarch Server database user in SQL Server. They differ in partitioning maintenance task support.

In a Monarch Server database, servicing Report Warehouse functionality, all report-related data are partitioned by date ranges, and each partition can be managed separately — backed up, removed, joined with another partition, etc. For Monarch Server to perform these operations automatically, partition maintenance tasks were designed.

For these tasks to function properly, the Monarch Server database user will require more rights. This requirement may be considered unsecure in some environments, particularly when the database server is used for several databases and not only for Monarch Server. In this case, these rights may be assigned temporarily as the partitioning maintenance task is performed. Another option is to perform partitioning maintenance operations manually.

For setup to create the Monarch Server database — and support the partitioning maintenance task functionality — you will need the **Database Creators** server role to be assigned to the user.

NOTE This role will allow the user to destroy any database in the system (without reading data). Thus, it is recommended to assign this role only during the time of installation

and during partitioning maintenance task operations.

To create the user, login to the database server under System Administrator (sa) user, and execute the following commands:

```
exec sp_addlogin 'ES_LOGIN', 'ES_PASSWORD'  
exec sp_adduser 'ES_LOGIN', 'ES_USER'  
exec sp_addrolemember 'db_ddladmin', 'ES_USER'  
exec sp_addrolemember 'db_datareader', 'ES_USER'  
exec sp_addrolemember 'db_datawriter', 'ES_USER'  
exec sp_addsrvrolemember 'ES_LOGIN', 'dbcreator'
```

If you decide to disable the **Database Creator** role after the Monarch Server database is created (i.e., after installation is complete), you will need to issue the following command under System Administrator:

```
exec sp_dropsrvrolemember 'ES_LOGIN', 'dbcreator'
```

NOTE

The db_owner role may be removed, and the db_executor role may be added. If the db_executor role does not exist, it may be created as follows:

```
CREATE ROLE db_executor  
GRANT EXECUTE TO db_executor  
EXEC sp_addrolemember 'db_executor', 'ES_USER'
```

Installer Preparation

Depending on a computer's local security policies and configuration, the execution of downloaded files might be blocked. In particular, when downloading a file through a Windows OS to an NTFS partition, the Windows OS considers this file as an attachment by default. As a result, the opening, unpacking, or execution of the file is "blocked," resulting in errors. To avoid the installer file of Monarch Server from being blocked, the following steps are recommended.

To unblock the installer file after download

1. Find the ZIP file of the Monarch Server installer.
2. Right-click on the ZIP file and select **Properties** in the context menu.
3. Navigate to the **Security** section of the **General** tab in the **Properties** window. Find the **Unblock** check box beside the text "This file came from another computer and might be blocked to help protect this computer."
4. Select the **Unblock** check box to activate it.
5. Click **OK** to apply the change.

MONARCH SERVER INSTALLATION

You need to start **NetTcpPortSharing** in Windows Services if the application uses TCP (by default).

1. Click **Start**, point to **Administrative tools**, and double-click **Services**.

2. Right-click on **NetTcpPortSharing**, and select **Properties**.
3. In the **Startup type** drop-down list, select **Automatic**, and click **OK**.

Next time you start Windows, the service will be started automatically.

It is recommended that you stop Internet Server Services before you begin the installation.

You must be logged on as a local administrator on the machine where you are going to install the Monarch Server system, or the UAC should be turned off during the installation process.

NOTE If some system files are to be replaced during the installation, you will see a message informing of a Windows restart requirement at the end of the installation.

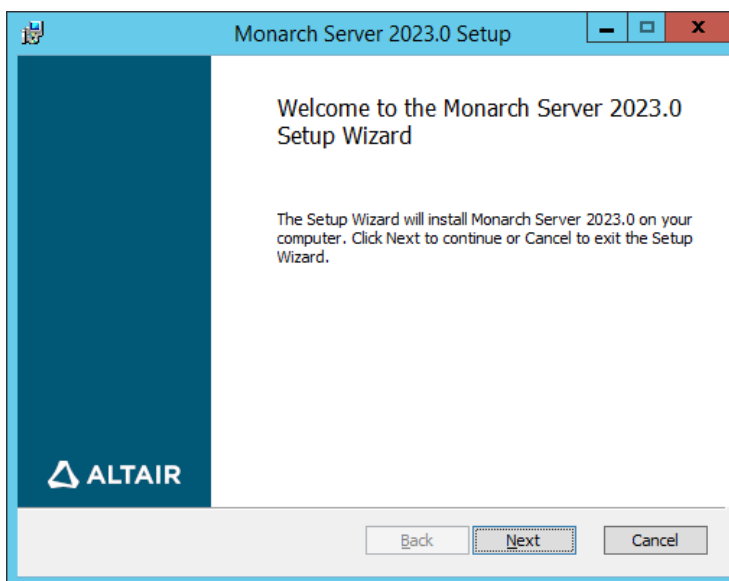
Open the directory with the installation files, and check whether it contains the license file labeled as “ds.lic.” If not, copy this file from the original Monarch Server installation media to the directory from which you are going to install Monarch Server.

NOTE If the configuration machine name in the license file is not the same as the machine name of installation, the warning message is displayed: “Config machine name in license does not match current machine name. Config service will not start properly.” You can proceed with the installation.

For the recent versions of Monarch Server, you can proceed with the installation without a license. The licensing can be configured after completing the installation. Details are available [here](#).

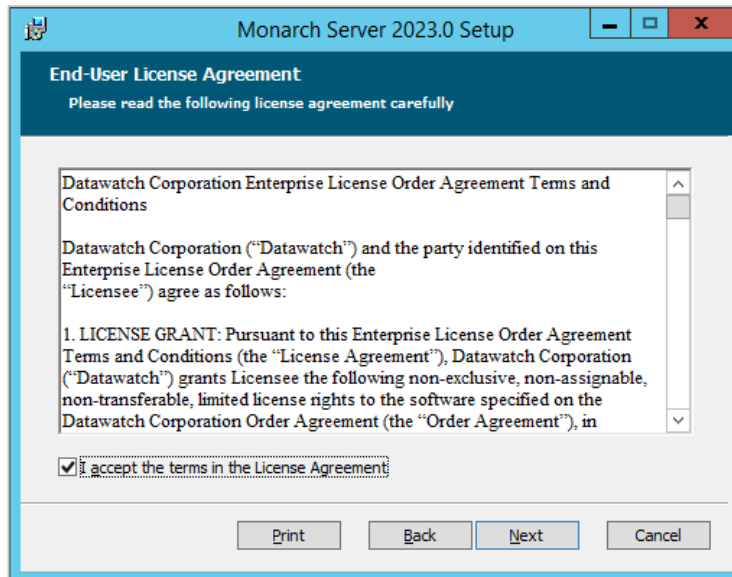
Single Server Installation

1. In the directory with the installation files, double-click **MonarchServerv2023x64.msi** in case of 64-bit Windows to launch the Monarch Server Setup Wizard.



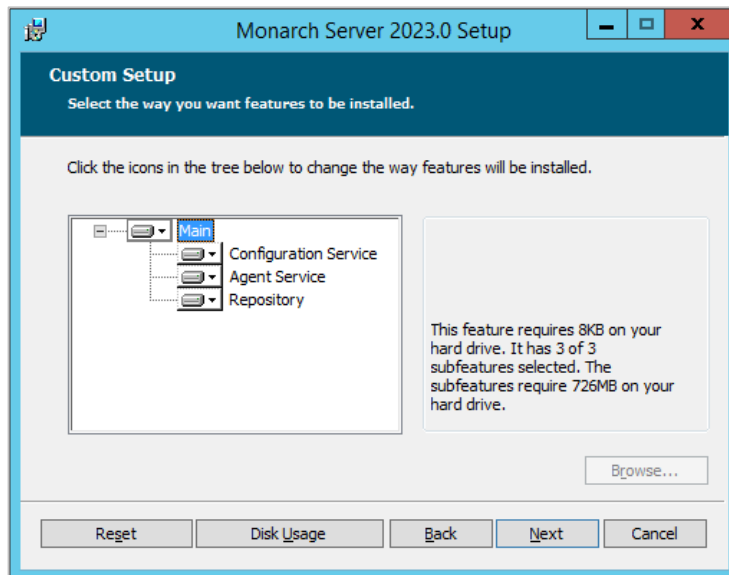
Click **Next**.

2. The End-User License Agreement dialog box appears.



If you agree to the license terms, select the check box below the text, and click **Next**.

3. The **Custom Setup** dialog box appears.

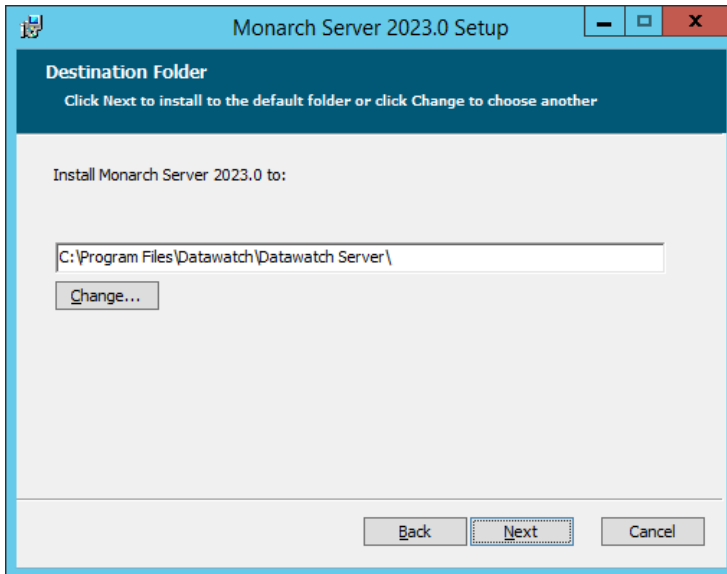


You can select the following components to be installed:

- **Configuration Service** manages the system configuration and tracks license agreements. It can serve more than one instance. In most cases, only one instance of Configuration Service should be installed.
- **Agent Service** manages services (modules) on a particular server. It should be installed on every server used for MS installation.
- **Repository** is a folder containing module packages. A package is a 7-Zip archive containing all assemblies required by a module. In most cases, only one instance of Repository should be installed.

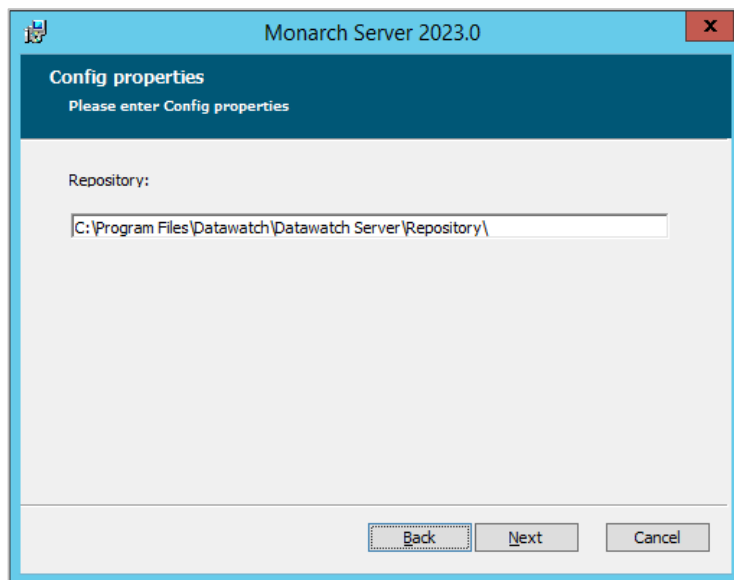
Click **Next**.

4. The **Destination Folder** dialog box appears.



Click **Next** to install Monarch Server in the default folder, or click **Change** to select another folder.

5. The **Config properties** dialog box appears.

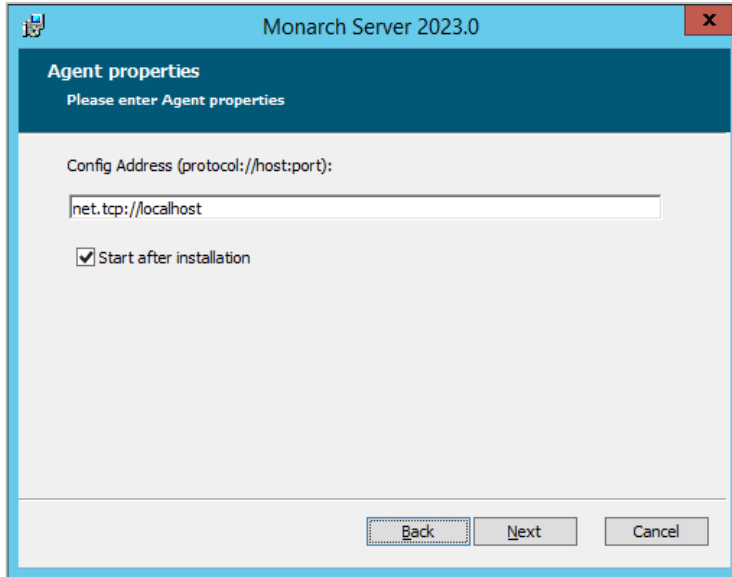


Set the following properties:

- In the **Repository** field, specify the path to the folder where the **Repository** will be located.

Click **Next**.

6. The **Agent properties** dialog box appears.

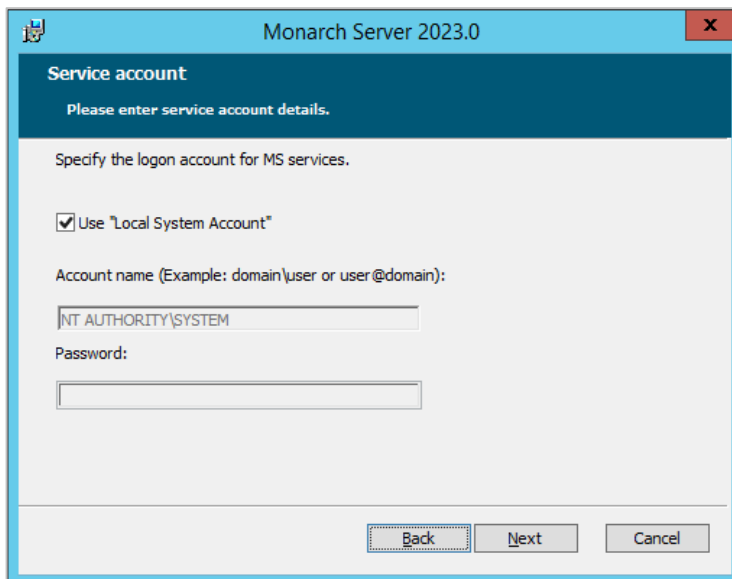


Set the following properties:

- In the **Config Address** field, specify the address where the **Configuration Service** will run; 'localhost' should be replaced with the FQDN of your server (e.g., net.tcp//myserver.domain).
- Select the **Start after installation** check box if you want the **Agent Service** to start immediately after the installation.

Click **Next**.

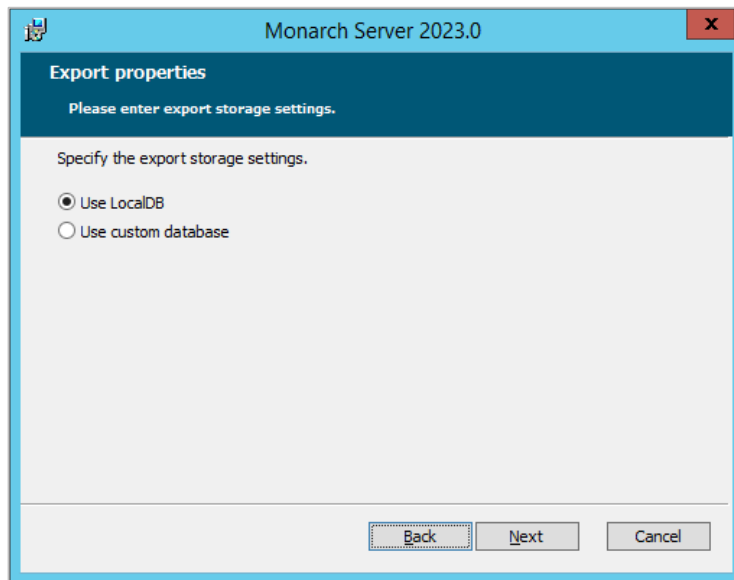
7. The **Service account** dialog box appears.



Select the **Use “Local System Account”** check box if you want to use a local system account. Otherwise, clear the check box and specify a custom user in the **Account name** box.

NOTE In case of a [distributed installation](#), if you specify a custom user in the **Account name** field, **NTLM** authentication is applied.

8. The **Export Properties** dialog box appears.



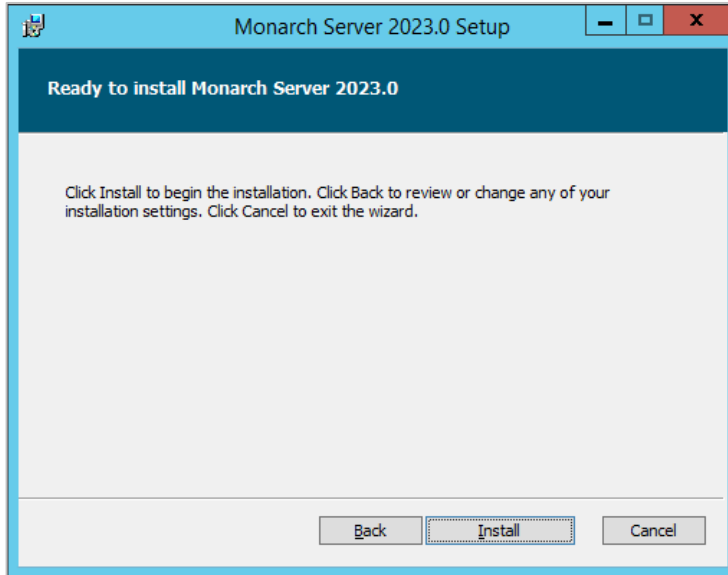
Specify the export storage settings:

- Select the **Use local DB** option if you want to use a local database as an export storage.
- Select the **Use custom database** option if you want to use a custom database as an export storage, and then specify its parameters:
 - ◆ Server name
 - ◆ Authentication credentials: user name and SQL password. If you want to use Windows authentication, select the corresponding check box.

NOTE Using a custom database means Monarch Server will need to be able to create/destroy work databases dynamically. Contact Altair Support for further information and guidance if required.

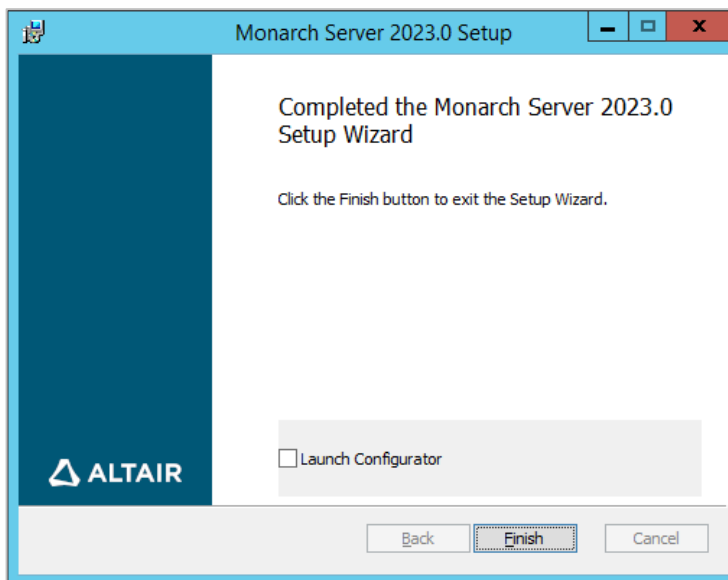
Click **Next**.

9. The following dialog box appears.



Click **Install** to begin the installation.

10. After the installation is complete, the following dialog box appears.



Select the **Launch Configurator** check box to run the **Configurator** immediately. Click **Finish** to exit the Setup Wizard.

Now you can access the following options via the **Start** menu, under **All Programs, Monarch Server, Monarch Server 64-bit**:

- Run **Agent Service** in console mode
- Run Configuration Service in console mode
- Run Configurator
- Uninstall

Creating Instances

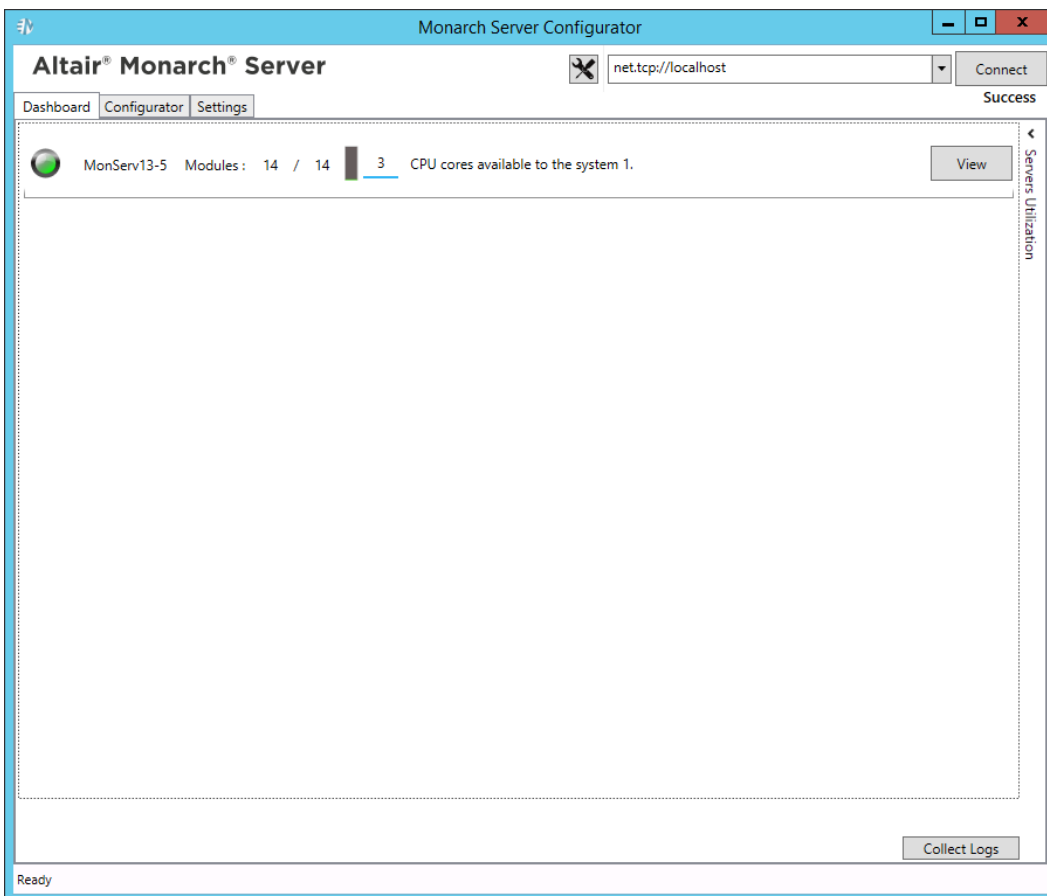
You can create instances using the **Configurator** tool.

The **Configurator** will launch immediately after installation if you select the corresponding check box.

NOTE You can run it manually by selecting the **Run Configurator** option on the **Start** menu.

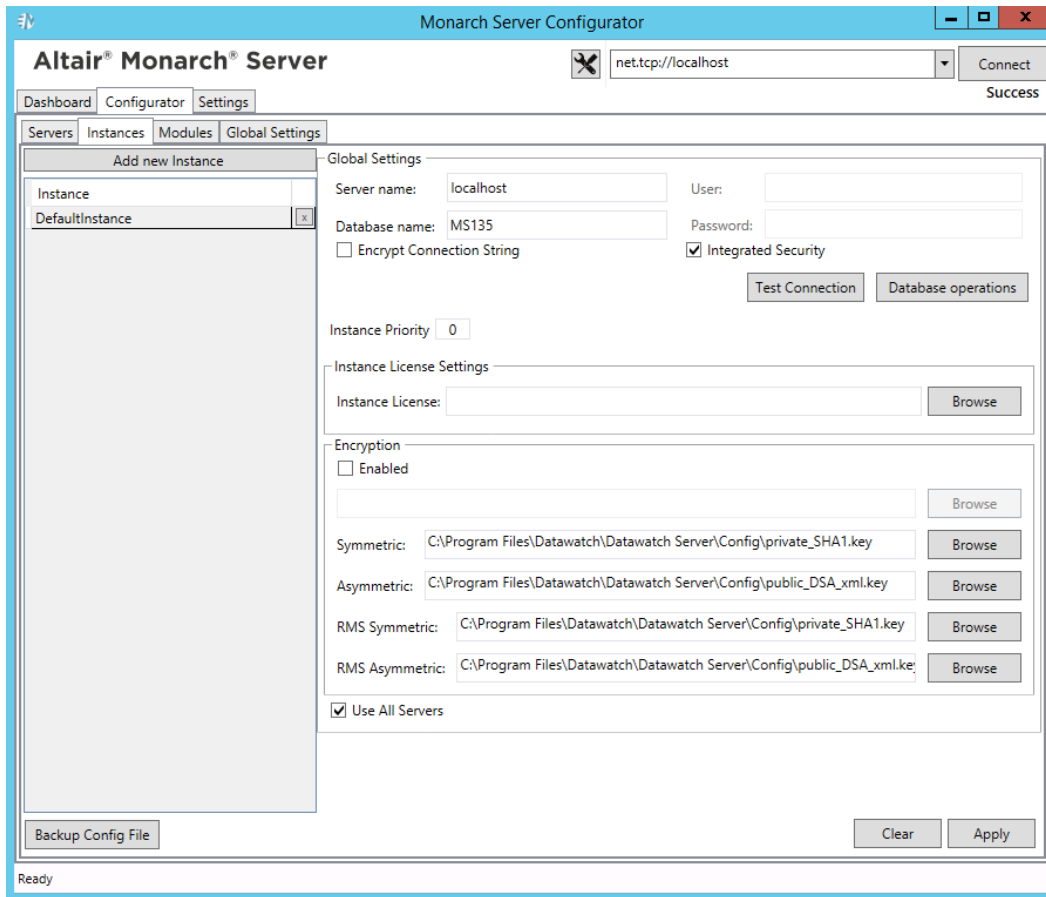
To create an instance

1. From the drop-down list in the top right corner in the **Monarch Server Configurator** window, select the domain server name and port where the **Configuration Service** runs, and click **Connect**.



NOTE The default service URI can be net.tcp://localhost. However, the **Configuration Service** can be launched on any computer. In this case, the URI should contain the domain server name.

2. Select the **Configurator** tab, and click the **Instances** tab.



- Under **Global Settings**, specify the connection string parameters: server name, database name, user, and password. Click **Test Connection**. You will be prompted to update the database.

NOTE Select the **Integrated Security** check box if you want to access SQL Server via Windows Authentication. This option is based on the context of the user running the Configurator application and NOT the Config Service.

If there is no database with the specified name, a message will appear asking if you want to create a new database.

Note that you need to configure a valid license to make any database operations.

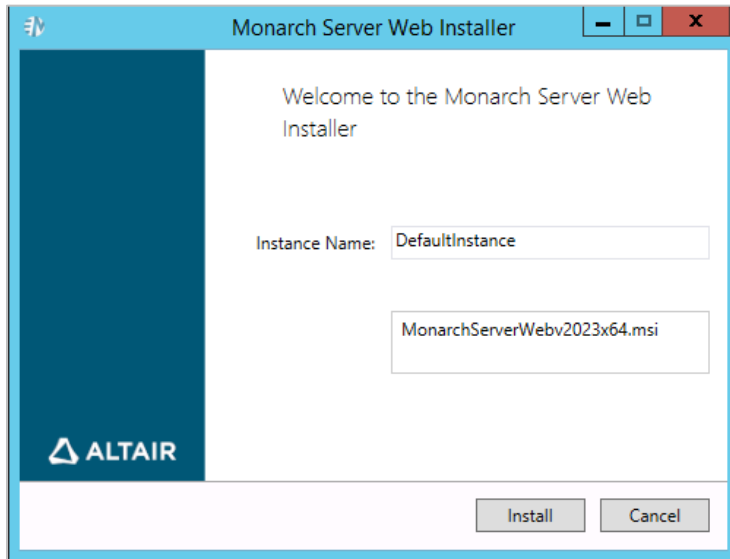
- Under **Instance License**, specify the path to the license file.
- Click **Apply**.
- If you want to create several instances, each using its own database, click the **Add new Instance** button, type the new instance name, and specify the parameters described in Step 2.

NOTE For additional information about setting instance parameters, click [here](#).

Installation of Web Applications

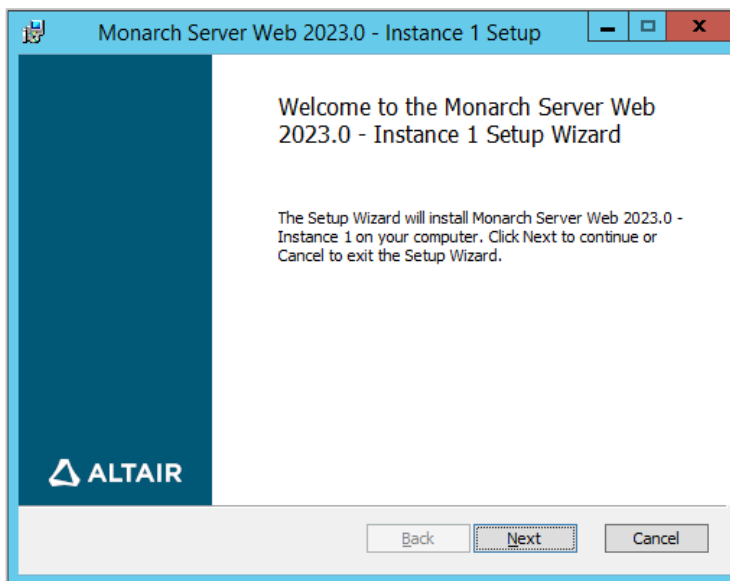
You need to install the **Admin** and **Client** applications for each instance.

1. Go to the directory with the installation files, and double-click **Web.Installer.exe**.
2. In the **Monarch Server Web Installer** dialog box, type the instance name (or leave the field empty if you plan to install only one instance), and click **Install**.

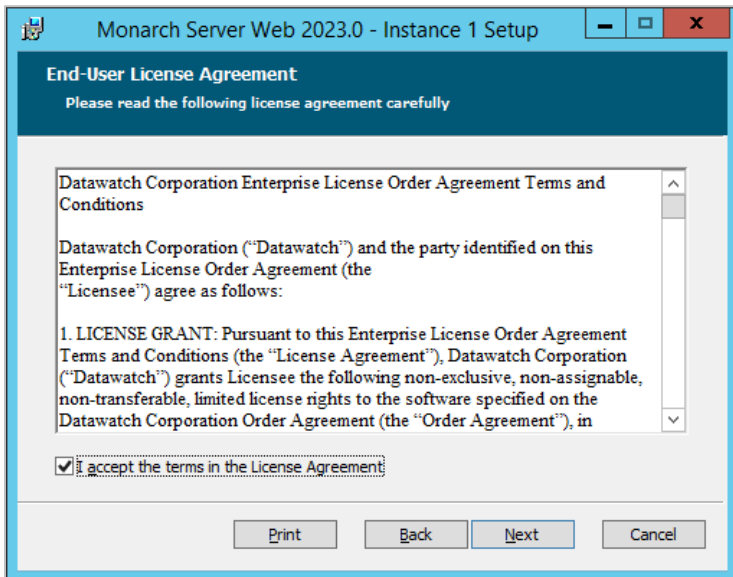


NOTE If you install only one instance, double-click **MonarchServerWebv2023x86.msi** (or **MonarchServerWebv2023x64.msi** in case of 64-bit Windows) to launch the Monarch Server Web Setup Wizard.

3. The **Setup Wizard** dialog box appears. To proceed, click **Next**.

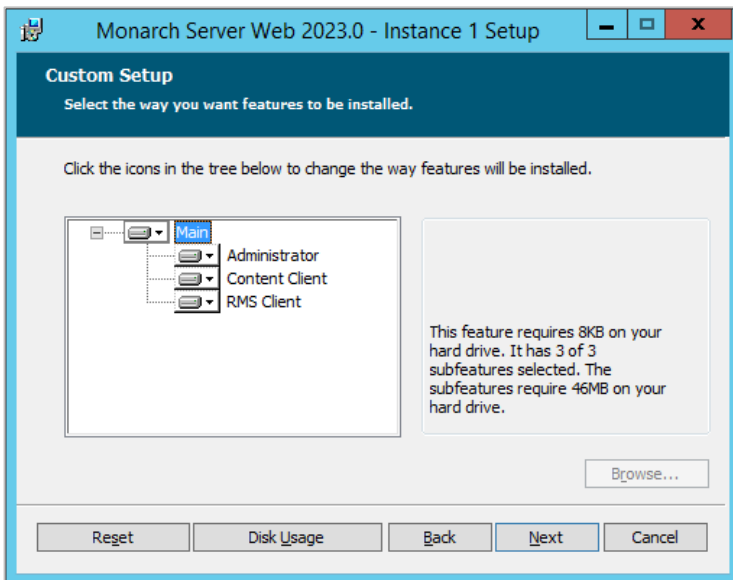


4. The End-User License Agreement dialog box appears.



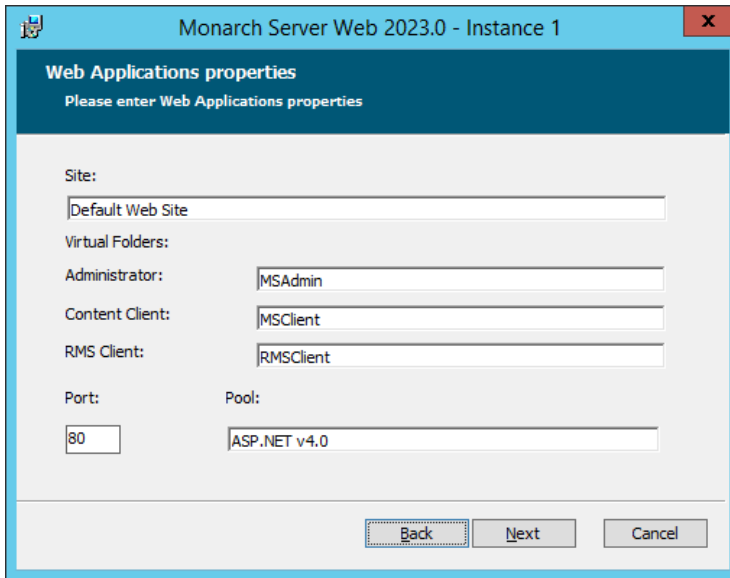
If you agree to the license terms, select the check box, and click **Next**.

5. The **Custom Setup** dialog box appears.



Select the components to be installed. For single server installation, install all the components. Click **Next**.

6. The Web Applications properties dialog box appears.



In the **Site** field, specify the site name where the Web applications will be installed.

NOTE Select this site name from those listed in the Internet Information Services (IIS).

Under **Virtual Folders**, specify the names of the virtual folders for **Administrator**, **Client**, and **RMS Client**.

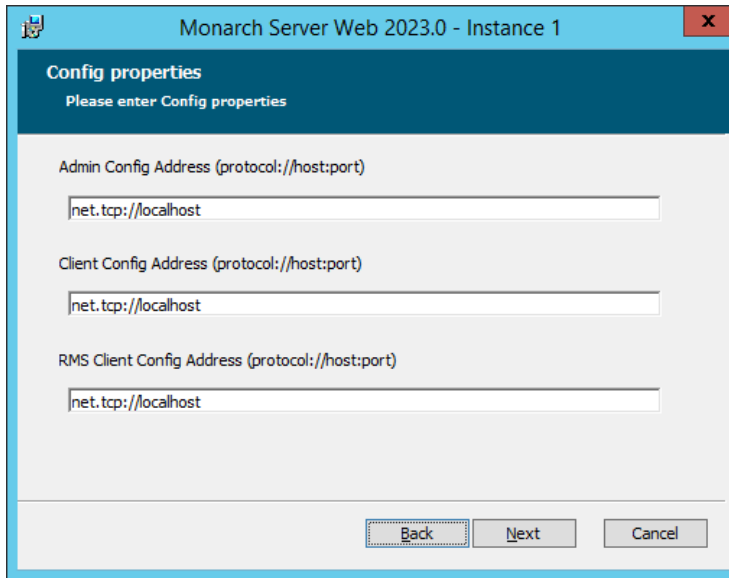
NOTE The installer creates virtual folder names using the template **DsAdmin%instance_name%** and maps them to the corresponding folder at C:\inetpub\wwwroot.

In the **Port** field, specify the port where the IIS runs.

In the **Pool** field, specify the name of the application pool to be used by the Web applications.

Click **Next**.

7. The **Config properties** dialog box appears.



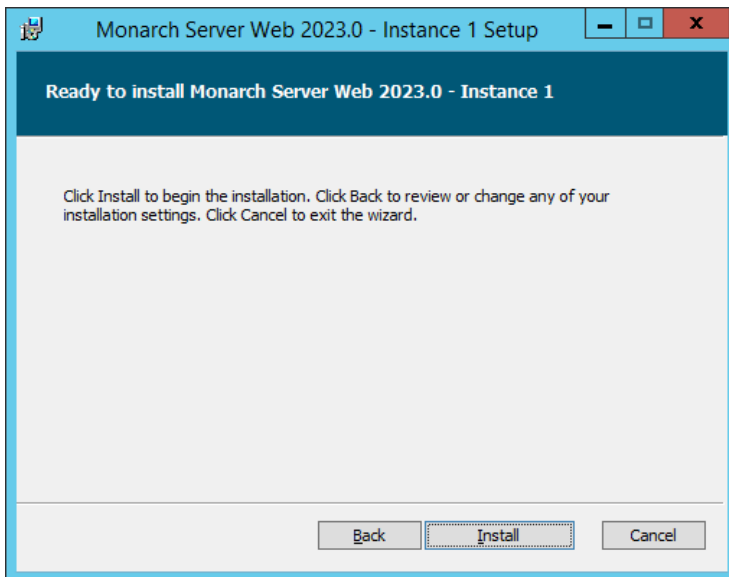
In the **Admin Config Address** field, specify the address where the Configuration Service runs for the Admin application.

In the **Client Config Address** field, specify the address where the Configuration Service runs for the Client application.

In the **RMS Client Config Address** field, specify the address where the Configuration Service runs for the RMS Client application.

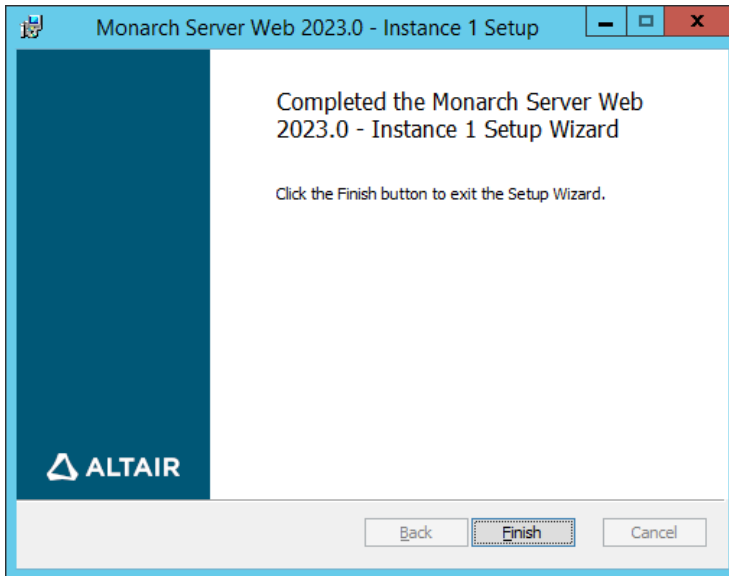
Click **Next**.

8. The following dialog box appears.



Click **Install** to begin the installation of the chosen applications.

9. After the installation is complete, the following dialog box appears.

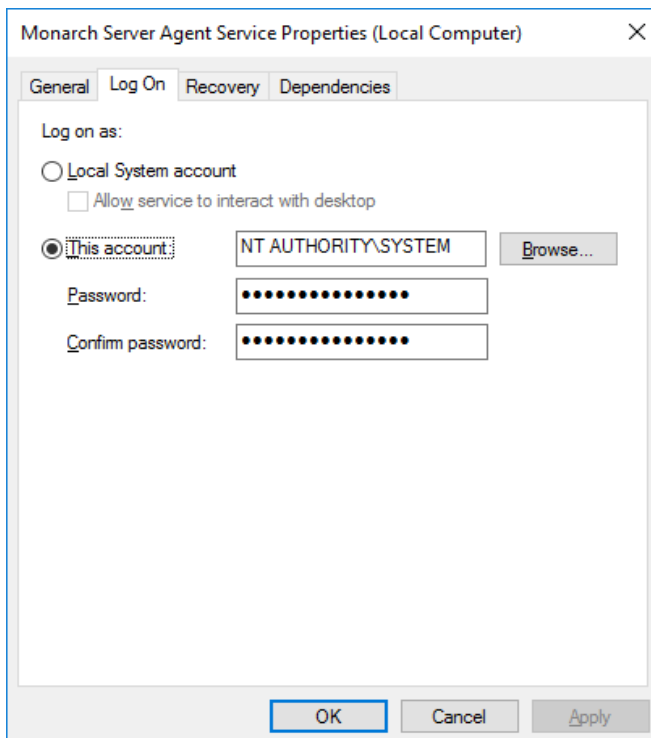


Click **Finish** to exit the Setup Wizard.

Changing the Account

To change the account

1. Click Start, select Control Panel, and point to Administrative Tools.
2. Double-click **Services**.
3. Double-click Monarch Server Agent Service.
4. Select the **Log On** tab and specify account properties.



- Click **OK** to save the changes and start the service.

Distributed Installation

All installation steps for single server installation are valid for distributed installation with the following remarks:

- In Step 7, in the **Service account** dialog box, you specify the logon account for MS services. If you specify a custom user in the **Account name** field, **NTLM** authentication is applied.

The following table provides details on the interaction of services in case of distributed installation.

Client \ Server	Local User	Local System	Domain User	Domain Machine
Local User	NTLM	NTLM	NTLM	NTLM
Local System	Anonymous NTLM	Anonymous NTLM	Anonymous NTLM	Anonymous NTLM
Domain User	NTLM	NTLM	Kerberos	Kerberos
Domain Machine	NTLM	NTLM	Kerberos	Kerberos

- You may install **Client**, **Admin**, and **RMS Client** applications together or separately on any server allowed by your license.
 - Install **Agent** on each server.
 - Configure the list of services that should run on each server.

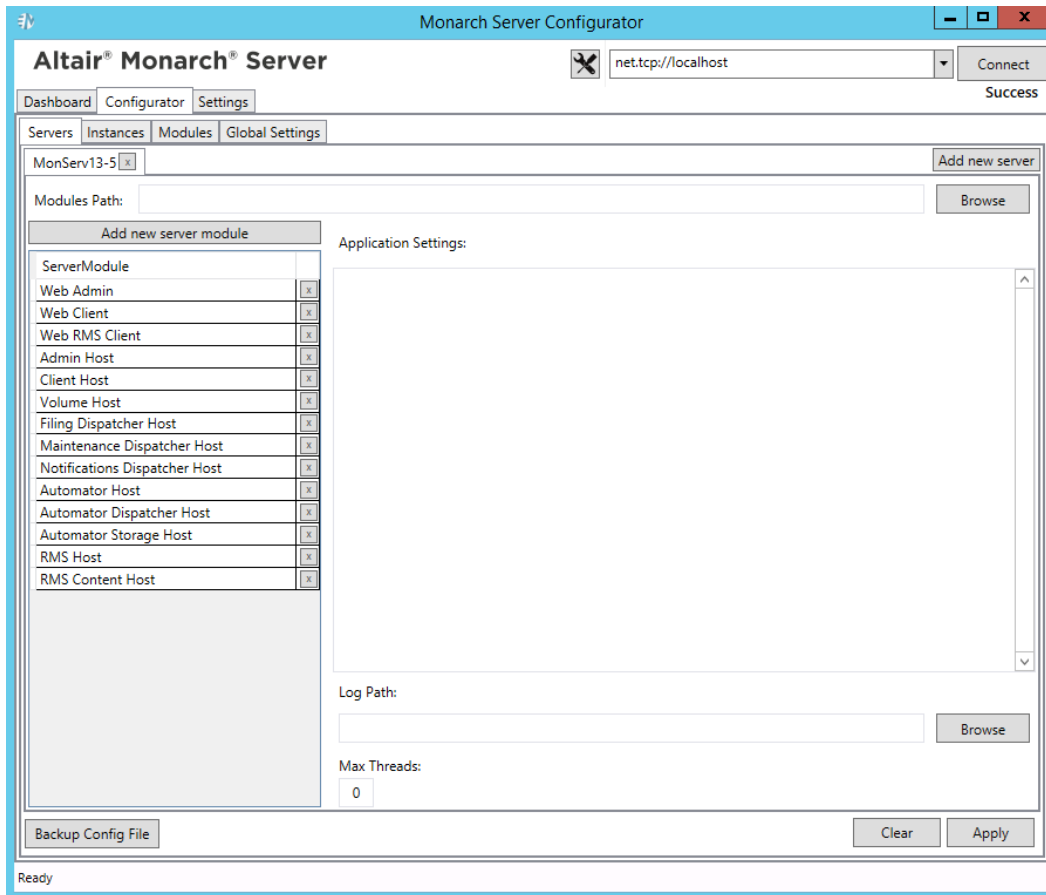
NOTE LOAD BALANCING

Every server tries to execute a task locally (with the tasks sorted according to execution time). It then checks the count of threads in the thread pool.

- If the thread count in the thread pool is the same across all servers, CPU utilization is >95% (check the Configuration Dashboard for the specific server for a given parameter or logs similar to "Server server_name utilization is percentage% processor time").
- If the thread pool lacks free threads (see logs similar to "Server server_name pump thread pool used used_number of pool_count (percentage%)"), the task being executed is forwarded to another server with low utilization.

To configure the list of services

- Select the Run Configurator option from the Start menu.
The **Configurator** will be launched.



2. From the drop-down list in the top right corner in the **Monarch Server Configurator** window, select the domain server name and port where the **Configuration Service** runs, and click **Connect**.
3. Under the **Configurator** tab, select **Servers**, and configure the list of services that should run on each server.

NOTE If you use the **Active Directory** security provider:

- Launch the Admin Host using an account that has the permission to receive the Active Directory (AD) users list.
- Launch the Client Host using an account that has access to the AD.

SERVER POST-INSTALLATION ACTIONS

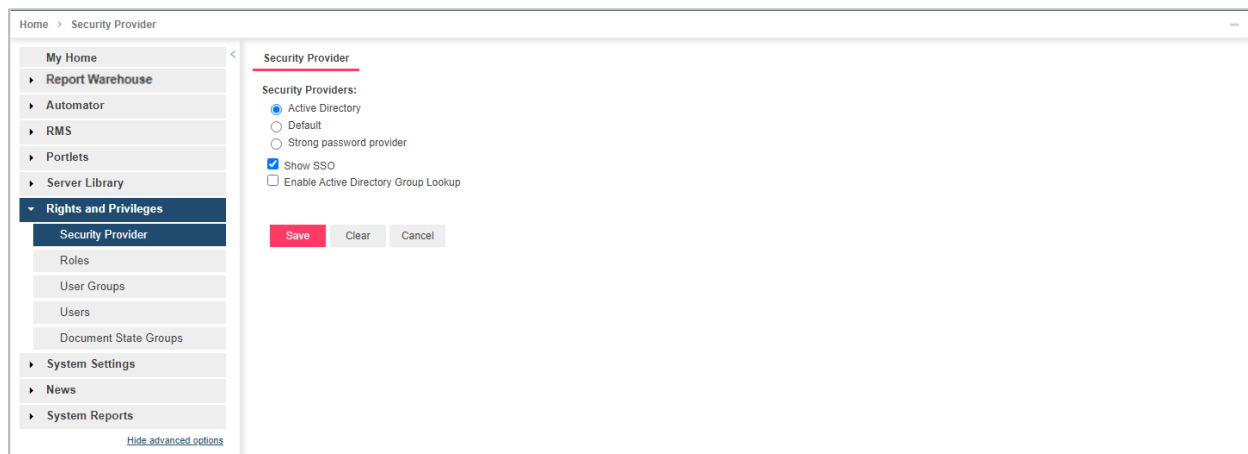
CONFIGURING SSO

To use SSO

1. Open Administrator.

NOTE In Monarch Server that is upgraded from previous versions (e.g., v13.2), the virtual folders could still be labeled as “DSAdmin” (or “DSClient”). In this case, use “DSAdmin” (or “DSClient”) instead of “MSAdmin” (or “MSClient”) to open the Admin (or Client) page.

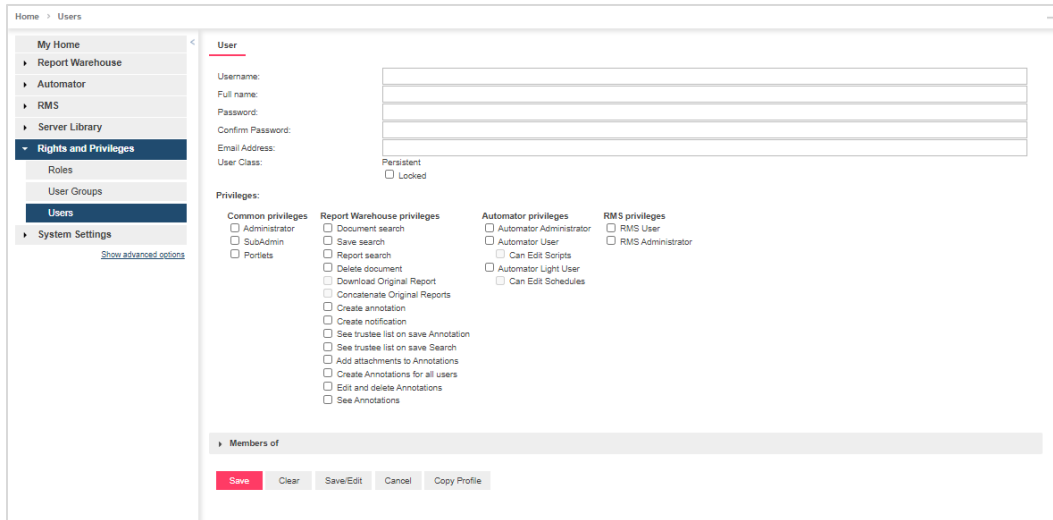
2. Under the Navigation Tree, click **Show Advanced Options**.
3. Click on Rights and Privileges, and then choose Security Provider.
4. Under Security Providers, select **Active Directory**, and then select **Show SSO**.



NOTE Mapping/aliasing of at least one administrator user is required. If there are no users mapped (MS and AD users are the same), the MS application is inaccessible without direct database modification of the Configuration table to the default state.

To add an administrator user account, choose **Users** under Rights and Privileges in the Navigation Tree. In the *User* window, click on “+” to add a new user.

In the *User* window, provide the necessary details. To check if the user account you are creating already exists, click on **Check User**. Do not forget to click **Save**.



5. Modify the file **web.config** of the Administrator and Client applications (the paths are *C:\inetpub\wwwroot\MSAdmin* and *C:\inetpub\wwwroot\MSClient*, respectively).

NOTE This step is crucial. One simple mistake can result in serious application issues. To avoid potential problems, make a backup of the **web.config** file before making any changes.

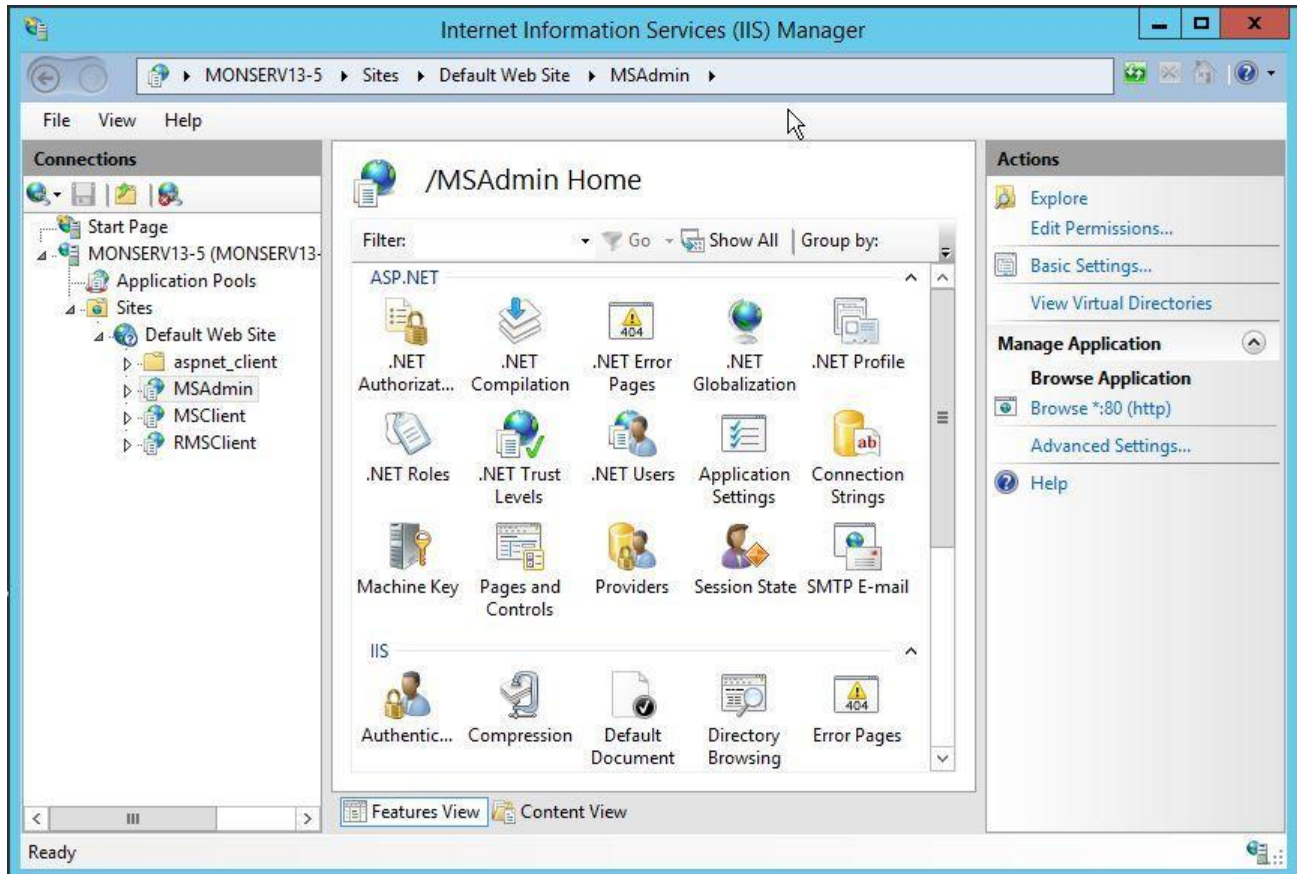
- a. Disable the following code:

```
<authentication mode="Forms">
  <forms loginUrl="~/Login/LoginRedirect" defaultUrl="~/Login/LoginIndex"
    cookieless="UseCookies" timeout="15" name=".ASPXFORMSAUTH"
    slidingExpiration="true"/>
</authentication>
```

- b. Enable the following code:

```
<authentication mode= "Windows" />
<authorization>
  <deny users="?" />
</authorization>
```

6. Make sure that **Windows, ASP.NET Impersonation** authentication on IIS for the Administrator and Client applications is disabled.
 - a. Go to Start > All apps > Windows Administrative Tools > Internet Information Services (IIS) Manager.

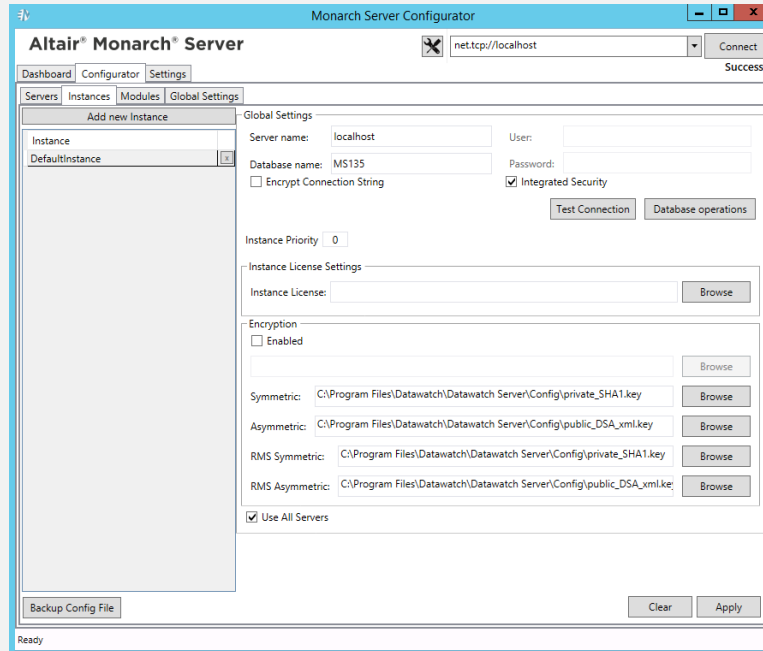


- c. Go to MSAdmin > Authentication, and disable ASP.NET Impersonation status.
 - d. Go to MSCClient > Authentication, and disable ASP.NET Impersonation status.
7. Restart the Monarch Server Config and Monarch Server Agent Services, IIS.
 8. On the login page of the Administrator and Client applications, click the **SSO** button.

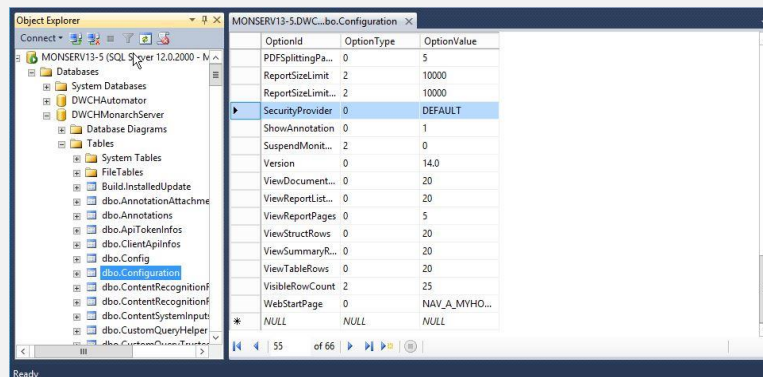
NOTE

If you fail to define at least one administrator user and cannot login to the Administrator page, edit the Configuration table, and set the OptionValue of SecurityProvider to DEFAULT. The steps are as follows:

- a. Open Microsoft SQL Server Management Studio.
- b. In the **Object Explorer** panel, click **Databases**, and choose the database you are using. (You may check the database name using the Monarch Server Configurator. Under Configurator, click the Instances tab. The Database Name is shown in the Global Settings window. Refer to the image below.)



- c. Under **Tables**, right-click on **dbo.configuration**, and then choose **Edit Top 200 Rows**.
- d. Look for the OptionId **"SecurityProvider,"** and then change its OptionValue to **DEFAULT** (case sensitive).



SETTING USER SESSION TIMEOUT

The server provides two parameters that allow setting up a timeout period, after which user sessions within the application expire and need to be renewed. These parameters are stored in the file **web.config**, the path to which is:

C:\inetpub\wwwroot\MSAdmin(MSClient or RMSCClient)

One parameter is located in the Session State mode **In Proc**; this parameter checks when the last request was sent. Session timeout period is the amount of time from the last request.

In the Monarch Server applications, the parameter used by the system is the one with the smaller value.

NOTE The following parameter is valid only for MSAdmin and MSClient.

The other parameter is located in the authentication mode **Forms**; this parameter checks the time of the last request from the user when half of the time defined in the parameter passes (for example, if the specified time is 10 minutes, the check will be performed after 5 minutes).

SETTING ADMINISTRATOR PASSWORD AND E-MAIL ADDRESS

Before you use the Monarch Server system, you should set the password and e-mail address for the administrator account.

1. Open a browser and type the MSAdmin address in the **Address** bar (for example, <http://localhost/MSAdmin>).
2. Log in using the default administrator account (the login name is "admin," the password is "password").
3. On the Navigation Tree, double-click **Rights and Privileges**, and select **Users**.
4. Click **admin**.
5. Type a password in the **Password** and **Confirm Password** boxes, and then click **Save**.
6. Click **Save** on the User page to save the changes.
7. On the Navigation Tree, double-click **System Settings**, and select the **Application Settings** tab on top of the page.
8. Enter a valid e-mail address in the **Administrator E-mail Address** field.
9. Click **Save**.

CLIENT INSTALLATION

No special client installation is required. The user needs the following:

- ❑ One of the following browsers:
 - Mozilla Firefox
 - Google Chrome
 - Microsoft Edge

NOTE Each release of Monarch Server is tested with the latest publicly available browser versions available at that time. Due to the frequency of browser version updates, Altair does not include specific browser version numbers in the system requirements. In case of changes in browser functionality affecting UI functionality, Altair will address the same as part of a subsequent patch release.

SOFTWARE REQUIREMENTS

Monarch Server was tested with the following client browsers and operating systems:

SOFTWARE	VERSION
Operating System: Windows	Windows Server 2016 (IIS 10.0 v1709) Windows Server 2019 (IIS 10.0 v1809) + Core Edition
Browsers: <ul style="list-style-type: none">❑ Google Chrome❑ Mozilla Firefox❑ Microsoft Edge	NOTE: Each release of Monarch Server is tested with the latest publicly available browser versions available at that time. Due to the frequency of browser version updates, Altair does not include specific browser version numbers in the system requirements. In case of changes in browser functionality affecting UI functionality, Altair will address the same as part of a subsequent patch release.

INSTALLATION MAINTENANCE

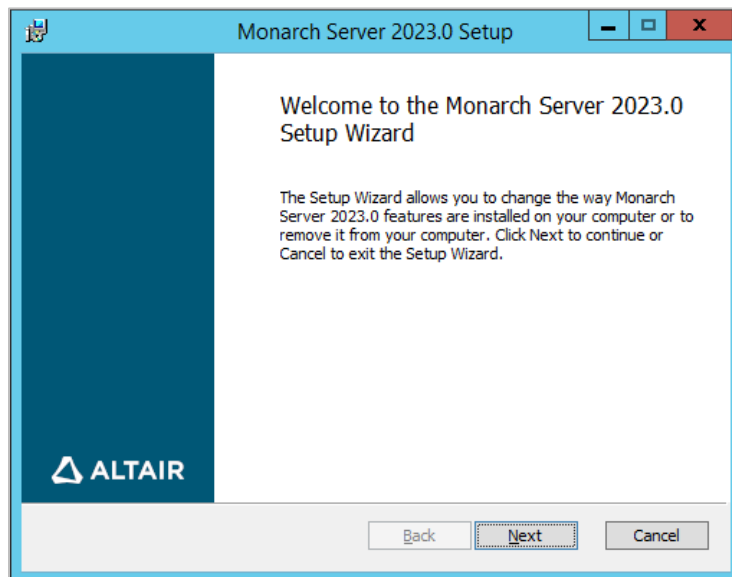
This section describes how to add/remove Monarch Server components on a computer and how to repair a damaged installation.

NOTE You must shut down Monarch Server application and the Web application engine before you attempt to do any of the tasks below.

ADDING OR REMOVING COMPONENTS

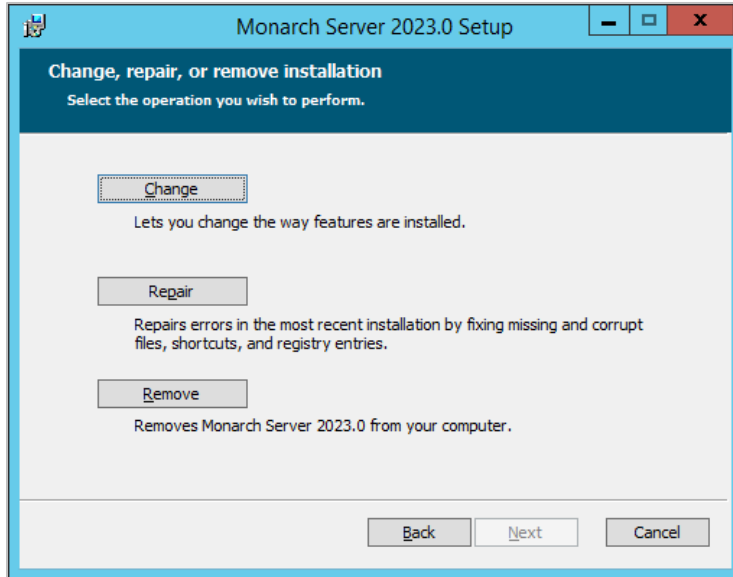
NOTE Adding/removing components via **Add/Remove Programs** will not work in the multi-instance mode. In this case, perform this operation manually.

1. Run the **MonarchServerv2023x86.msi** file from your installation media (**MonarchServerv2023x64.msi** in case of 64-bit Windows), or use **Add/Remove Programs** in the system's **Control Panel**. The Monarch Server Setup Wizard is launched.



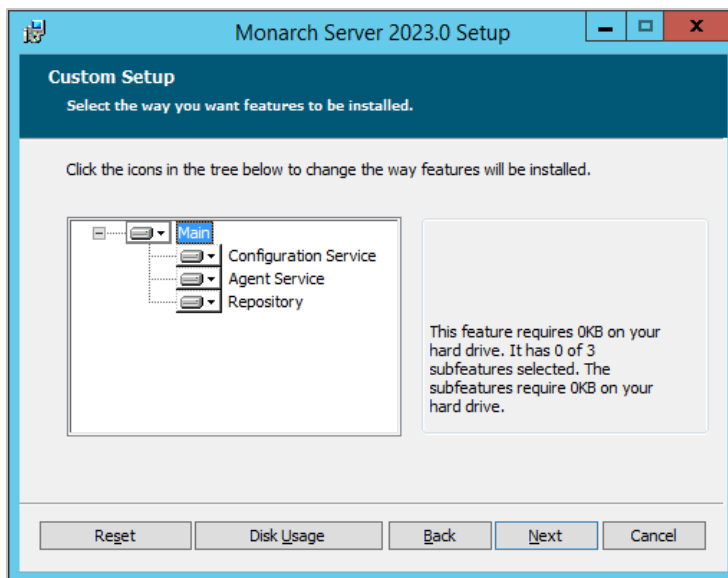
Click **Next**.

2. A dialog box appears with the following options.



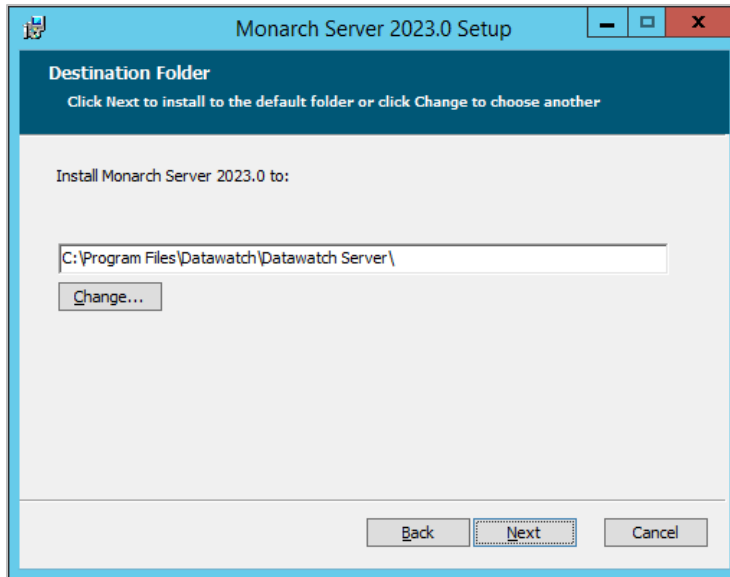
Click the **Change** button.

3. The **Custom Setup** dialog box appears.



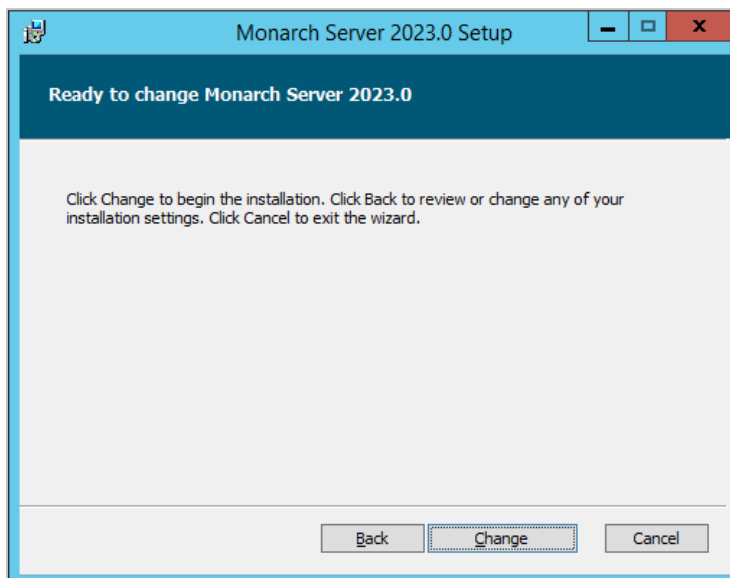
Select the components you want to install, or deselect the components you want to remove. Click **Next**.

4. The **Destination Folder** dialog box appears.



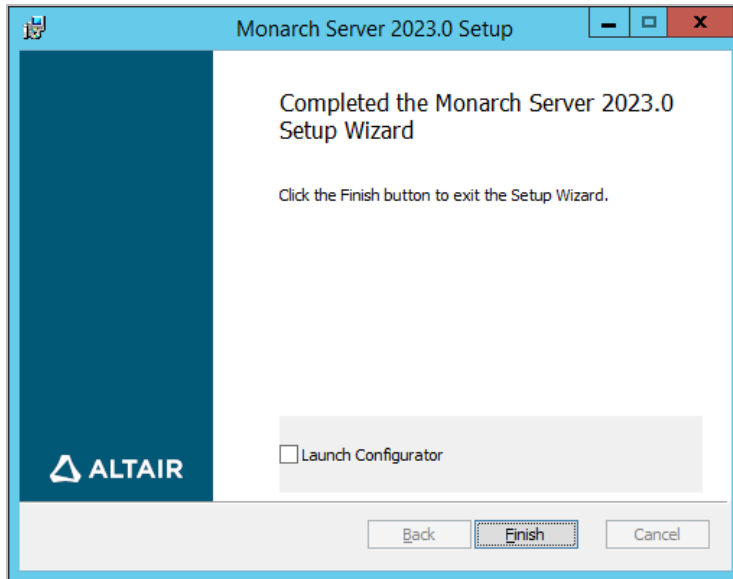
Click **Next** to install Monarch Server to the default folder, or click **Change** to select another folder.

5. The following dialog box appears.



Click **Change** to change the way features are installed.

6. When the maintenance is complete, the following dialog box appears.



Click **Finish** to exit the Setup Wizard.

REPAIRING A DAMAGED INSTALLATION

The Repair function of the Setup Wizard repeats the last installation to replace missing or damaged files of the Monarch Server System. Use this option if you accidentally deleted some Monarch Server System files or if the Monarch Server System becomes unworkable due to damaged files.

NOTE The Repair function removes all changes you made in property files or registry settings. Therefore, before beginning a repair, backup all property files and registry keys first.

Do not try to work with the system registry if you are unfamiliar with it. Incorrect actions can render your system unusable.

To repair a damaged installation

1. Run the **MonarchServerv2023x86.msi** file from your installation media (**MonarchServerv2023x64.msi** in case of 64-bit Windows), or use the **Add/Remove Programs** shortcut in the system's **Control Panel**. The Monarch Server Setup Wizard is launched.
2. Click **Next** and select the **Repair** option. Follow the instructions on the screen. When the repair is complete, the corresponding message will appear.
3. Click **Finish** to exit the Setup Wizard.

REMOVING ALL SYSTEM COMPONENTS

This function removes all Monarch Server components and files installed on the computer.

To remove all system components

1. Run the **MonarchServerv2023x86.msi** file from your installation media (**MonarchServerv2023x64.msi** in case of 64-bit Windows), or use the **Add/Remove Programs** shortcut in the system's **Control Panel**. The Monarch Server Setup Wizard is launched.
2. Click **Next** and select the **Remove** option. Follow the instructions on the screen. When the operation is complete, the corresponding message will appear.
3. Click **Finish** to exit the Setup Wizard.

UPGRADING INSTALLATION

If you have the Monarch Server application v13.0/v12.5 installed on your computer, you need to remove the previous version of the application before installing Monarch Server v2023.0.

NOTE Before uninstalling the previous version, ensure that you are able to restore the original environment, either wholly (e.g., full server/resource backups) or at minimum; the backups of the Monarch Server database; and all file-based resources dependent on the RMS, Automator, and Report Warehouse functionalities being utilized.

To remove the previous version of Monarch Server

1. Run the **MonarchServerv13x86.msi** file from your installation media (**MonarchServerv13x64.msi** in case of 64-bit Windows), or use the **Add/Remove Programs** shortcut in the system's **Control Panel**. The Monarch Server v13.0/v12.5 Setup Wizard is launched.
2. Click **Next** and select the **Remove** option. Follow the instructions on the screen. When the operation is complete, the corresponding message will appear.
3. Click **Finish** to exit the Setup Wizard.

You also need to remove the previous version of Web Installation.

To remove the previous version of Web Installation

1. Run the **MonarchServerWebv13x86.msi** file from your installation media (**MonarchServerWebv13x64.msi** in case of 64-bit Windows), or use the **Add/Remove Programs** shortcut in the system's **Control Panel**. The Monarch Server Web v13.0/v12.5 Setup Wizard is launched.
2. Click **Next** and select the **Remove** option. Follow the instructions on the screen. When the operation is complete, the corresponding message will appear.
3. Click **Finish** to exit the Setup Wizard.

After performing these steps, you need to run the version installation as specified in the [Single Server Installation](#) section of this guide.

If you have the **distributed** installation of Monarch Server v13.0/v12.5, you need to remove the application and Web Installation from all servers. Then, you need to run the version installation as specified in the [Distributed Installation](#) section of this guide.

NOTE Before uninstalling the previous version, you can back up the config.xml file and web.config files to use some settings from these files after the installation of Monarch Server v2023.0.

UPGRADING USING MULTI-INSTALLER

If you have the Monarch Server application v13.5 (or later) installed on your computer and launch the latest version's installer to upgrade it, it is NOT necessary to remove the previous installation.

To prepare for the upgrade

1. Backup the config.xml/web.config(s).
2. Close the Configurator.
3. Stop the Application Pool (IIS) used by the MS web applications (MSAdmin, MSCClient, RMSClient) or stop the sites where the web applications are deployed.

To upgrade to the latest version

1. Run **MonarchServerv2023x64.msi** or **MonarchServerv2023x86.msi**, whichever is applicable to the previous installation. Make sure that the license file is available in the same folder as the installer.
2. Proceed with the installation as usual (Steps [1–3](#), [7](#), and [9–10](#), [Single Server Installation](#) section).
3. Stop the Monarch Server Agent service.
4. In Monarch Server Configurator, click the Configurator tab, and then click the Instances tab.
5. In the **Instances** tab, click the **Database Operations** button to launch the **Database Operations** window.
6. In the **Database Operations** window, click **Update Database**, and then close the window.
7. Click **Apply** in the **Instances** tab to save the changes.
8. After the upgrade, web.config reverts to the old state for the multi-installer. Thus, in each web.config, the version numbers should be manually changed as follows:
 - a. Copy the following content and open the corresponding web.config file ([Admin](#), [Client](#), [RMS](#)):

ADMIN

```
<assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity.Interception"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Newtonsoft.Json"
publicKeyToken="30ad4fe6b2a6aeed" culture="neutral" />
    <bindingRedirect oldVersion="0.0.0.0-13.0.0.0" newVersion="13.0.0.0" />
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.EnterpriseLibrary.Common"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
    <bindingRedirect oldVersion="0.0.0.0-6.0.0.0" newVersion="6.0.0.0" />
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity.Configuration"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Web.Http"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
    <bindingRedirect oldVersion="0.0.0.0-5.2.7.0" newVersion="5.2.7.0" />
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Net.Http.Formatting"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
    <bindingRedirect oldVersion="0.0.0.0-5.2.7.0" newVersion="5.2.7.0" />
  </dependentAssembly>
</assemblyBinding>
```

```

    <dependentAssembly>
      <assemblyIdentity name="System.Web.Optimization"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-1.1.0.0" newVersion="1.1.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="WebGrease" publicKeyToken="31bf3856ad364e35"
culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-1.6.5135.21930"
newVersion="1.6.5135.21930" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="WebMatrix.Data"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="System.Web.WebPages.Razor"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="WebMatrix.WebData"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="System.Web.Razor"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="IronRuby" publicKeyToken="7f709c5b713576e1"
culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-1.1.3.0" newVersion="1.1.3.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="Microsoft.Scripting"
publicKeyToken="7f709c5b713576e1" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-1.1.0.30" newVersion="1.1.0.30" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="Microsoft.Dynamic"
publicKeyToken="7f709c5b713576e1" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-1.1.0.20" newVersion="1.1.0.20" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="System.Web.WebPages.Deployment"
publicKeyToken="31bf3856ad364e35" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="BundleTransformer.Core"
publicKeyToken="973c344c93aac60d" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-1.8.32.0" newVersion="1.8.32.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="Datawatch.Core.Logging.Contracts"
publicKeyToken="47f651eeea82774c" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-16.0.0.88" newVersion="16.0.0.88"
/>
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="Antlr3.Runtime"
publicKeyToken="eb42632606e9261f" culture="neutral" />
      <bindingRedirect oldVersion="0.0.0.0-3.5.0.2" newVersion="3.5.0.2" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="System.Web.Helpers"
publicKeyToken="31bf3856ad364e35" />
      <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0" />

```

```

    </dependentAssembly>
  </dependentAssembly>
  <assemblyIdentity name="System.Web.WebPages"
publicKeyToken="31bf3856ad364e35" />
  <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.Mvc"
publicKeyToken="31bf3856ad364e35" />
  <bindingRedirect oldVersion="0.0.0.0-5.2.7.0" newVersion="5.2.7.0" />
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="Mono.Reflection"
publicKeyToken="0738eb9f132ed756" culture="neutral" />
  <bindingRedirect oldVersion="0.0.0.0-2.0.0.0" newVersion="2.0.0.0" />
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="DelegateDecompiler"
publicKeyToken="93b26a10a04705bd" culture="neutral" />
  <bindingRedirect oldVersion="0.0.0.0-1.0.0.0" newVersion="1.0.0.0" />
</dependentAssembly>
</assemblyBinding>

```

CLIENT

```

<assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
  <dependentAssembly>
    <assemblyIdentity name="System.Web.Http"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-5.2.7.0" newVersion="5.2.7.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity.Interception"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.EnterpriseLibrary.Common"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-6.0.0.0" newVersion="6.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity.Configuration"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Net.Http.Formatting"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-5.2.7.0" newVersion="5.2.7.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Web.Optimization"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-1.1.0.0" newVersion="1.1.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="WebGrease" publicKeyToken="31bf3856ad364e35"
culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-1.6.5135.21930"
newVersion="1.6.5135.21930"/>
  </dependentAssembly>
</dependentAssembly>

```

```

    <assemblyIdentity name="System.Web.WebPages.Razor"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Web.Razor"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Web.WebPages.Deployment"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Newtonsoft.Json"
publicKeyToken="30ad4fe6b2a6aeed" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-13.0.0.0" newVersion="13.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Datawatch.Core.Logging.Contracts"
publicKeyToken="47f651eeea82774c" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-16.0.0.88" newVersion="16.0.0.88"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Antlr3.Runtime"
publicKeyToken="eb42632606e9261f" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.5.0.2" newVersion="3.5.0.2"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Web.Helpers"
publicKeyToken="31bf3856ad364e35"/>
    <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Web.WebPages"
publicKeyToken="31bf3856ad364e35"/>
    <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="System.Web.Mvc"
publicKeyToken="31bf3856ad364e35"/>
    <bindingRedirect oldVersion="1.0.0.0-5.2.7.0" newVersion="5.2.7.0"/>
  </dependentAssembly>
</assemblyBinding>

```

RMS

```

<assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity.Interception"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.EnterpriseLibrary.Common"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-6.0.0.0" newVersion="6.0.0.0"/>
  </dependentAssembly>
  <dependentAssembly>
    <assemblyIdentity name="Microsoft.Practices.Unity.Configuration"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
  </dependentAssembly>
</assemblyBinding>

```

```

    </dependentAssembly>
  </dependentAssembly>
  <assemblyIdentity name="ICSharpCode.SharpZipLib"
publicKeyToken="1b03e6acf1164f73" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-0.86.0.518"
newVersion="0.86.0.518"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.Http"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-5.2.7.0" newVersion="5.2.7.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Net.Http.Formatting"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-5.2.7.0" newVersion="5.2.7.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="Newtonsoft.Json"
publicKeyToken="30ad4fe6b2a6aeed" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-13.0.0.0" newVersion="13.0.0.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.Optimization"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-1.1.0.0" newVersion="1.1.0.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="WebGrease" publicKeyToken="31bf3856ad364e35"
culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-1.6.5135.21930"
newVersion="1.6.5135.21930"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.WebPages.Deployment"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.Razor"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.WebPages.Razor"
publicKeyToken="31bf3856ad364e35" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="Datawatch.Core.Logging.Contracts"
publicKeyToken="47f651eeea82774c" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-16.0.0.88" newVersion="16.0.0.88"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="Antlr3.Runtime"
publicKeyToken="eb42632606e9261f" culture="neutral"/>
  <bindingRedirect oldVersion="0.0.0.0-3.5.0.2" newVersion="3.5.0.2"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.Helpers"
publicKeyToken="31bf3856ad364e35"/>
  <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.WebPages"
publicKeyToken="31bf3856ad364e35"/>
  <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0"/>
</dependentAssembly>
</dependentAssembly>
  <assemblyIdentity name="System.Web.Mvc"
publicKeyToken="31bf3856ad364e35"/>

```



```
<bindingRedirect oldVersion="1.0.0.0-5.2.7.0" newVersion="5.2.7.0"/>
</dependentAssembly>
</assemblyBinding>
```

- b. In the web.config file, highlight the entire **assemblybinding** section and then paste the copied text from Step 8a to replace it.
 - c. Save the changes and close the config file.
9. Run the Application Pool.
 10. In the Services tool, restart Monarch Server Config Service, and start Monarch Server Agent Service to activate the update.

The upgraded version should now be available.

NOTE

- Restart the server if requested during the upgrade.
- The initial load of the application may take a few seconds to deploy the update.
- This upgrade supports Monarch Server v13.5 or later.
- In case of distributed installation, **MonarchServerv2023x64.msi** may be used for every machine.
- MonarchServerWebv2023x64.msi** or **MonarchServerWebv2023x86.msi** does not need to be run during the upgrade. Web applications are upgraded automatically. Only **MonarchServerv2023x64.msi** (**MonarchServerv2023x86.msi**) needs to be run.

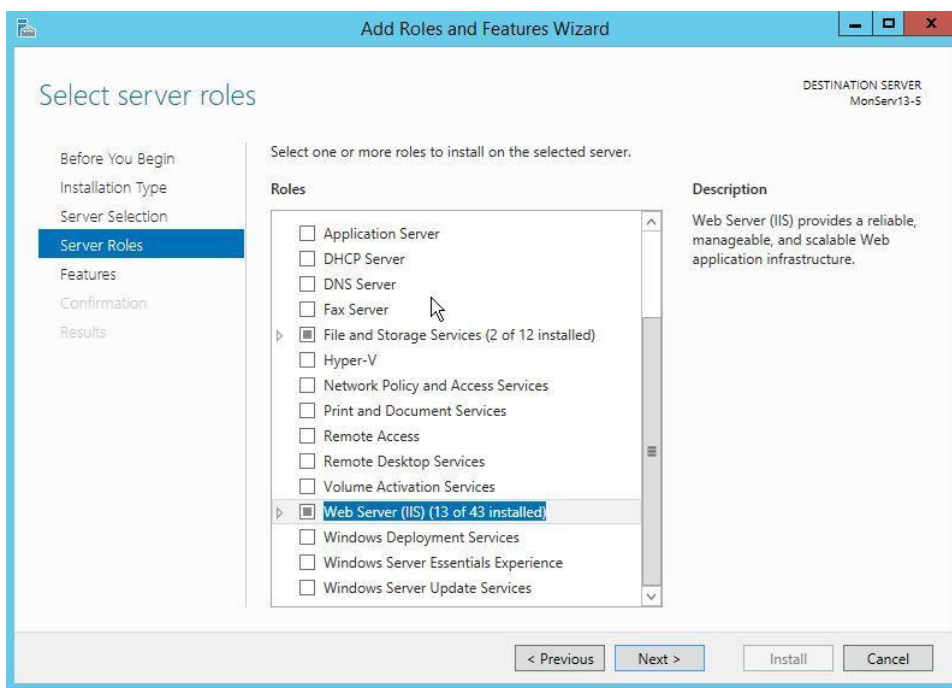
CONFIGURING SOFTWARE TO WORK WITH MONARCH SERVER

CONFIGURING WEB SERVICES

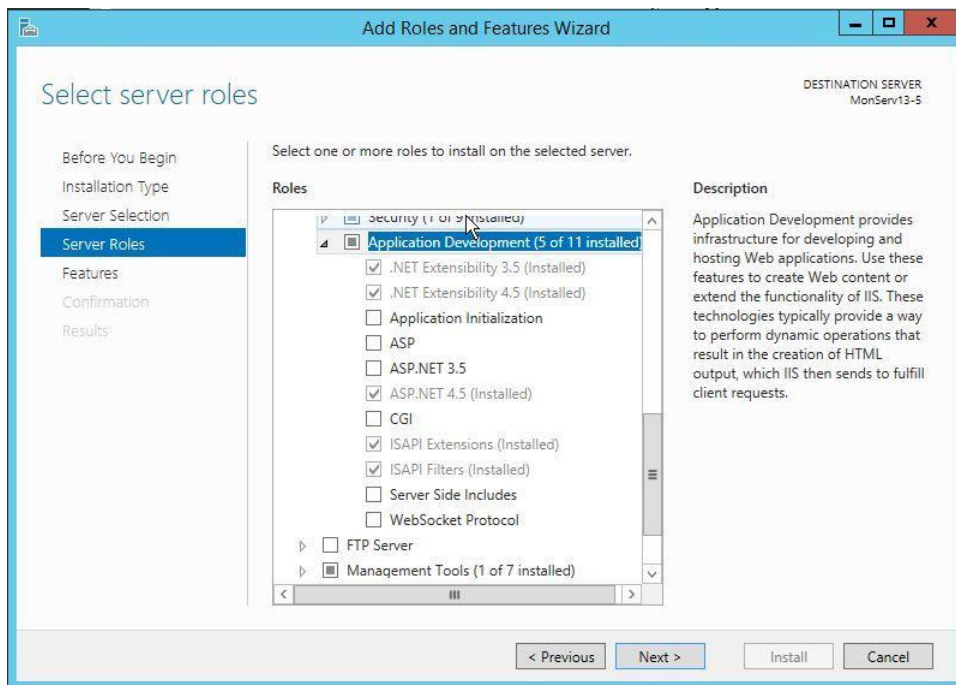
If you install the Web Applications for the first time, you need to configure IIS in the following way.

To install IIS

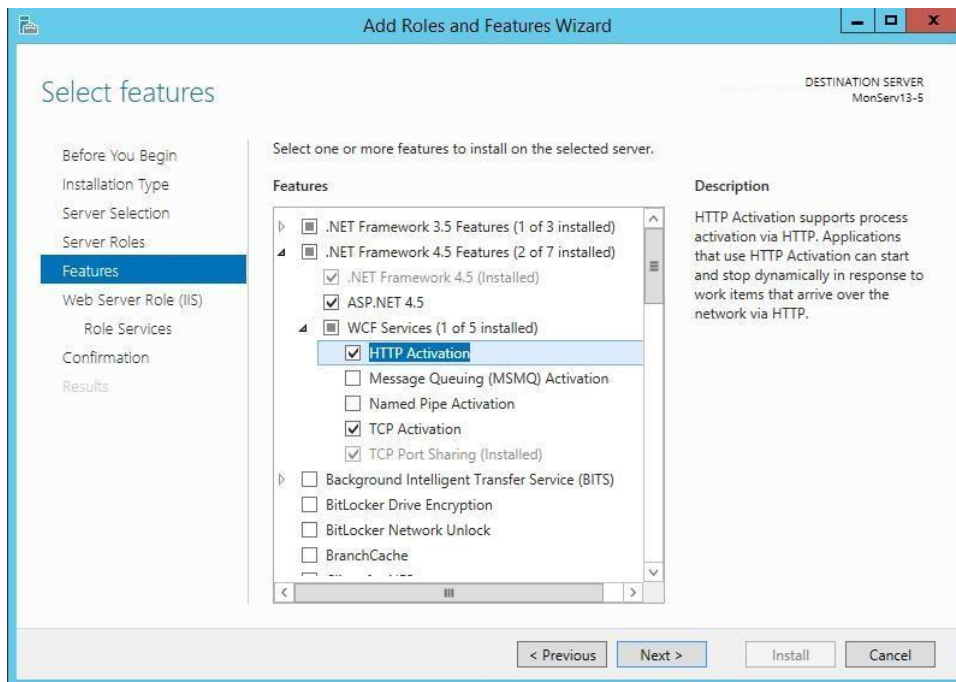
1. Click Start, point to Administrative Tools, and then click Server Manager.
2. Under **Roles**, click **Add Roles**.
3. Select Web Server (IIS) on the Server Roles page.



4. Select the following services:
 - ASP.NET
 - .NET Extensibility
 - ISAPI Filters



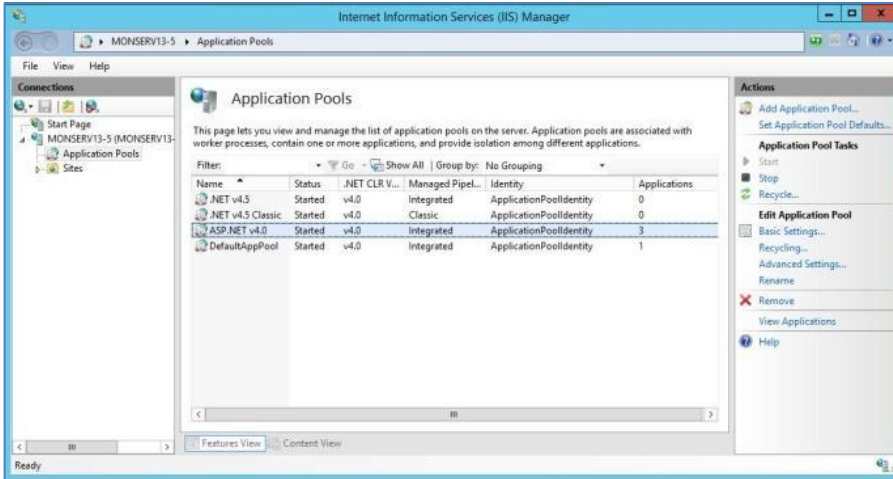
If you use Windows Server 2012, select HTTP Activation of the .NET 4.5 WCF Services.



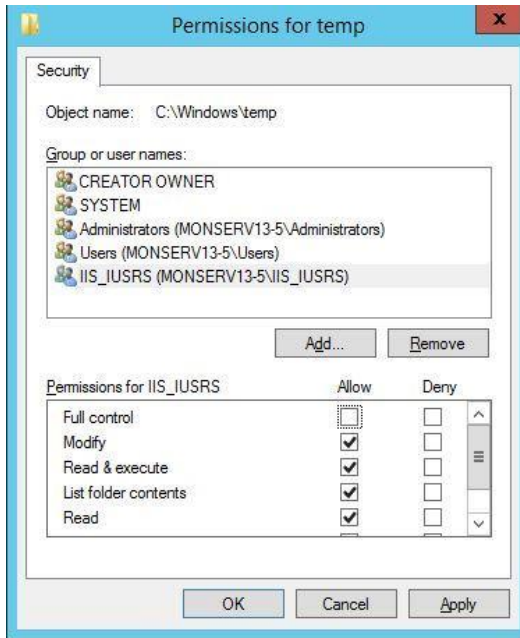
5. Register .NET Framework:

- 32-bit %WINDIR%\Microsoft.Net\Framework\v4.0.30319\aspnet_regiis.exe -i
- 64-bit %WINDIR%\Microsoft.Net\Framework64\v4.0.30319\aspnet_regiis.exe -i

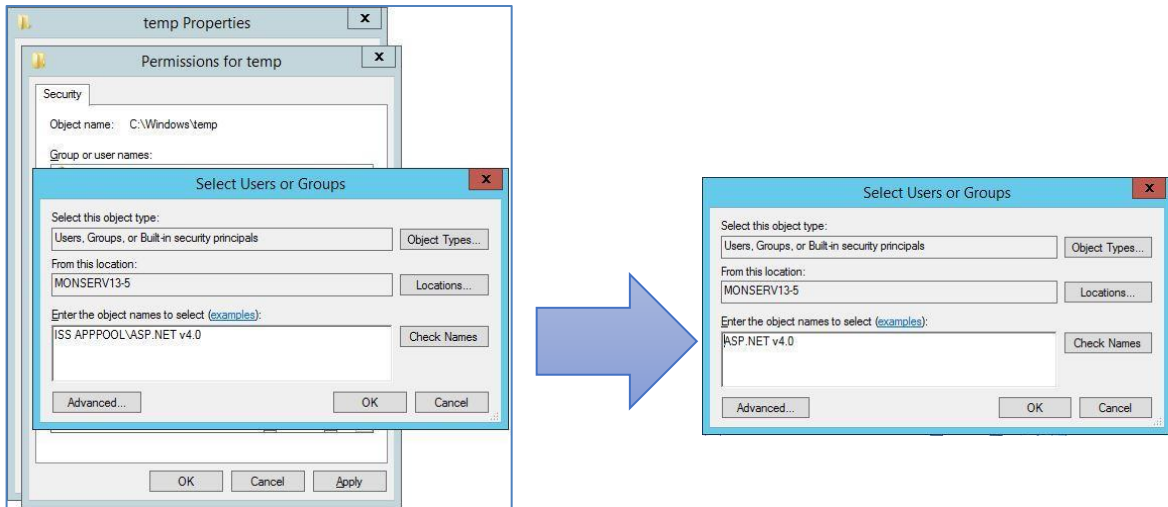
6. In the IIS Manager, look up the **Identity** of the ASP.NET v4.0 application pool.



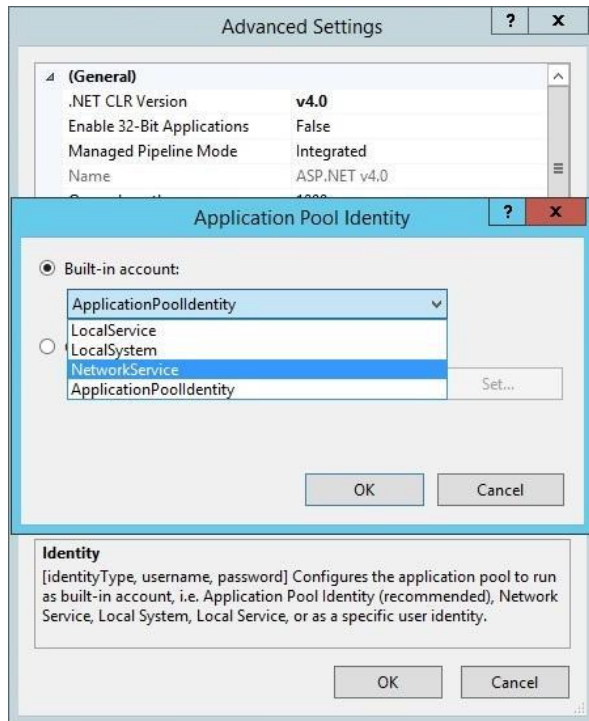
7. Based on the **Identity**, give write permissions to the “%WINDIR%\temp” folder:
 - a. Right-click on the folder name and select **Properties**.
 - b. Select the **Security** tab and click **Edit**.
 - c. Select the **Modify** check box and click **OK**.



- d. Click **Add** in the **Security** dialog box.
- e. Enter the pool identity name and click **Check Names** to look up available application pool identity names. Click **OK**.

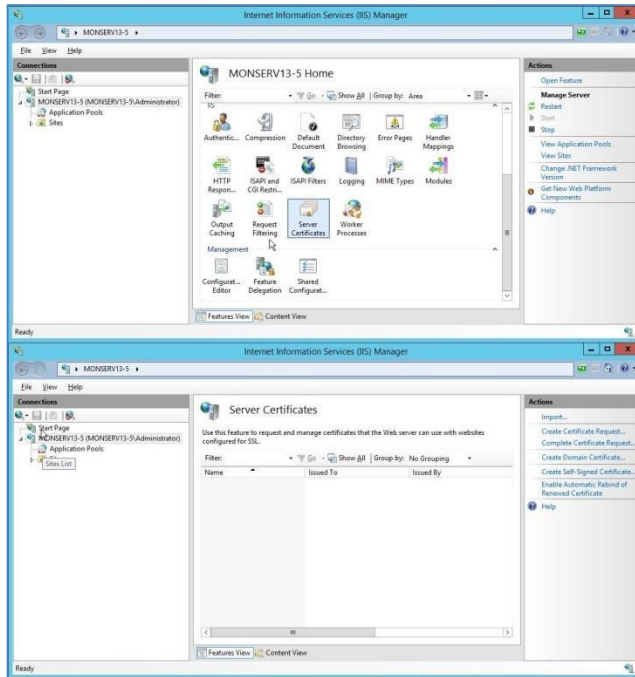


8. You can change the application pool identity if necessary.
 - a. In the **IIS Manager**, click the application pool name.
 - b. In the Actions pane, click Advanced Settings.
 - c. Under **Process Model**, select **Identity**, and click the (...) button.
 - d. Select another identity from the list, and click **OK**.

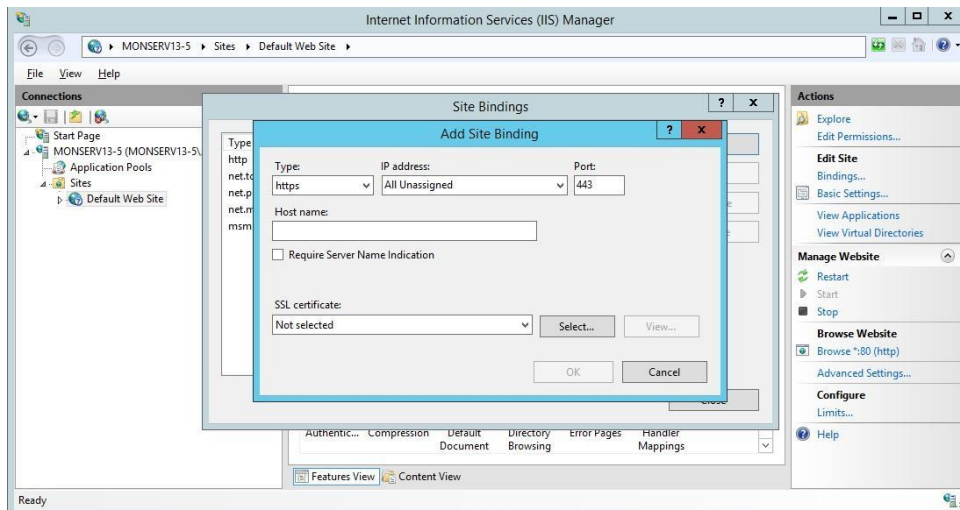


9. To support SSL, you need to set up a certificate or create a self-signed certificate:
 - a. Open the **IIS Manager** and navigate to the level you want to manage.
 - b. In the **Features** view, double-click **Server Certificates**.
 - c. In the Actions pane, click Create Self-Signed Certificate.

- d. On the **Create Self-Signed Certificate** page, type a friendly name for the certificate in the **Specify a friendly name for the certificate** box, and then click **OK**.



10. Assign the certificate to a site:
 - a. In the **IIS Manager**, select a site.
 - b. In the **Actions** pane, click **Bindings**.
 - c. Click **Add**.
 - d. Select the **Https** type, select the certificate, and then click **OK**.



CONFIGURING WEB APPLICATIONS

After installing Monarch Server v2023.0, you can manually edit some of the Clients applications' parameters in the web.config file.

CLIENT

The default path to the file is C:\inetpub\wwwroot\MSClient\web.config.

The parameters listed below are located in the <appSettings> of the MSClient.

The recording format of each of the parameters is as follows:

```
<add key="parameterName" value="parameterValue" />
```

PARAMETER NAME	DESCRIPTION	DEFAULT VALUE
allowXformPdfView	Enable ES Style PDF View.	true
enableTrainingLink	Enable Lesson link.	true
ClientEndSessionRedirectUrl	URL to be opened on clicking to End Session link.	
usePdfSubgroupHeaders	Enable Subgroup Headers in PDF export from Dynamic View.	true
loginOutOnBrowserCloseEnabled	Enable immediate session termination when the browser or browser tab is closed.	false
loginOutOnBrowserCloseTimeout	Waiting time for the termination of current session after the browser or browser tab is closed.	5000 (ms)

RMSCLIENT

The default path to the file is C:\inetpub\wwwroot\RMSClient\web.config.

The parameters listed below are located in the <appSettings> of the RMSClient.

The recording format of each of the parameters is as follows:

```
<add key="parameterName" value="parameterValue" />
```

PARAMETER NAME	DESCRIPTION	DEFAULT VALUE
userGroupName	Name of the group whose rights are applied to users when logging in to the RMS Client.	RMSGroup
exportPDFInSameWindowEnabled	Enable export to PDF in the current window.	true
usePdfSubgroupHeaders	Enable Subgroup Headers in PDF export from Dynamic View.	true
loginOutOnBrowserCloseEnabled	Enable immediate session termination when the browser or browser tab is closed.	false

PARAMETER NAME	DESCRIPTION	DEFAULT VALUE
loginOutOnBrowserCloseTimeout	Waiting time for the termination of current session after the browser or browser tab is closed.	5000 (ms)

ADMIN

The default path to the file is C:\inetpub\wwwroot\MSAdmin\web.config.

The parameters listed below are located in the <appSettings> of the MSAdmin.

The recording format of each of the parameters is as follows:

```
<add key="parameterName" value="parameterValue" />
```

PARAMETER NAME	DESCRIPTION	DEFAULT VALUE
loginOutOnBrowserCloseEnabled	Enable immediate session termination when the browser or browser tab is closed.	false
loginOutOnBrowserCloseTimeout	Waiting time for the termination of current session after the browser or browser tab is closed.	5000 (ms)

SYSTEM CONFIGURATION

After you install the Monarch Server web applications on the servers allowed by the license, you need to configure the servers, instances, and modules to get your system up.

Monarch Server Configurator is a PC-based software tool for configuring the Monarch Server System.

Monarch Server Configurator functions include the following:

- Setting up servers, instances, and service modules
- System analysis and workload control

STARTING CONFIGURATOR

There are two ways to start the Monarch Server Configurator:

- Immediately after server installation by selecting the **Launch Configurator** check box in the setup window
For more information, see the [Installation Guide](#).
- From the **Start** menu if you want to launch the Configurator later.

To run Configurator from the Start menu

1. On the **Start** menu, select **All Programs**.
2. From the displayed list, select **Altair Monarch Server**.
3. Select **Run Configurator**.

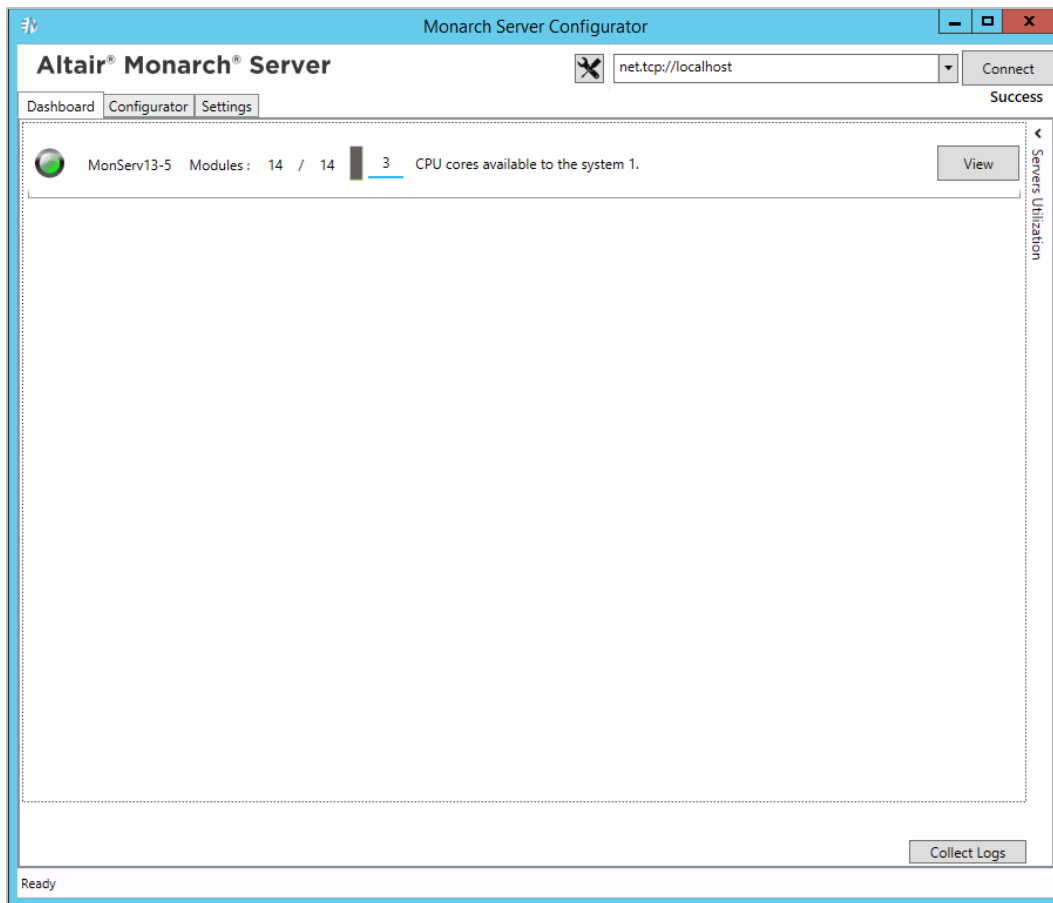
CONNECTING TO CONFIGURATOR

To set a connection to Configurator

1. From the drop-down list in the upper-right corner of the **Configurator** window, select the domain server name and port where the **Configuration Service** runs, and click **Connect**.

NOTE The default value is `net.tcp://localhost`.

2. If the connection is completed successfully, the Dashboard page displaying a list of servers appears.



The Dashboard page contains the following items:

- **State Indicator:** The circle on the left side of each server indicates the state of server modules by changing its color:
 - ◆ Green — all modules are running
 - ◆ Yellow — some modules were stopped
 - ◆ Red — all modules were stopped
- **Servers Utilization diagram:** Click the arrow on the upper-right corner of the page to view the **Servers Utilization** diagram.
- **View** button: To view the log information and to manage server modules, click the **View** button.
- **Collect logs** button: Allows Configurator to immediately collect and archive all Log (*.log) files, and save them to a local file.


NOTE Configurator is able to collect log files only from those machines where Agent service is started.

BACKING UP CONFIG.XML

Vital information about the Monarch Server System is stored in the **config.xml** file of the Config Service. The automatic backup of every changed version of the config.xml file enables you to restore a working configuration in cases of application failure caused by modified settings in the config.xml file. You can also manually create backups of the config.xml file via the Monarch Server Configurator.

NOTE The backup folder (C:\ProgramData\Datawatch\Monarch Server\ConfigBackups) is not removed after an upgrade or uninstall. The folder content should be removed manually when a subsequent reinstallation is NOT planned as the backup files in the folder may contain sensitive information (e.g., connection string information).

To enable/disable the automatic backup of config.xml

1. Launch the **Monarch Server Configurator**.
2. In the **Configurator** window, click the **Troubleshooter** button  found near the drop-down menu on the upper-right corner.
The Troubleshooter window displays.
3. In the **Advanced Settings** panel below the Troubleshooter window, click the **Enable auto backup** or **Disable auto backup** button.

- NOTE**
- The automatic backup function produces up to 25 copies. When this limit is reached, the oldest copy is removed to accommodate the latest copy made.
 - Elevated permissions are required to modify the system environment variables.
 - The Monarch Server Config Service (and Agent Service if running) requires a restart to apply the new settings.

To manually backup the config.xml file


1. Launch the **Monarch Server Configurator**.
2. Click the **Configurator** tab.
3. In the Configurator tab, click the **Backup Config File** button found on the lower left corner.

The notification “Config file backup has been successfully created” is then displayed in the status bar below.

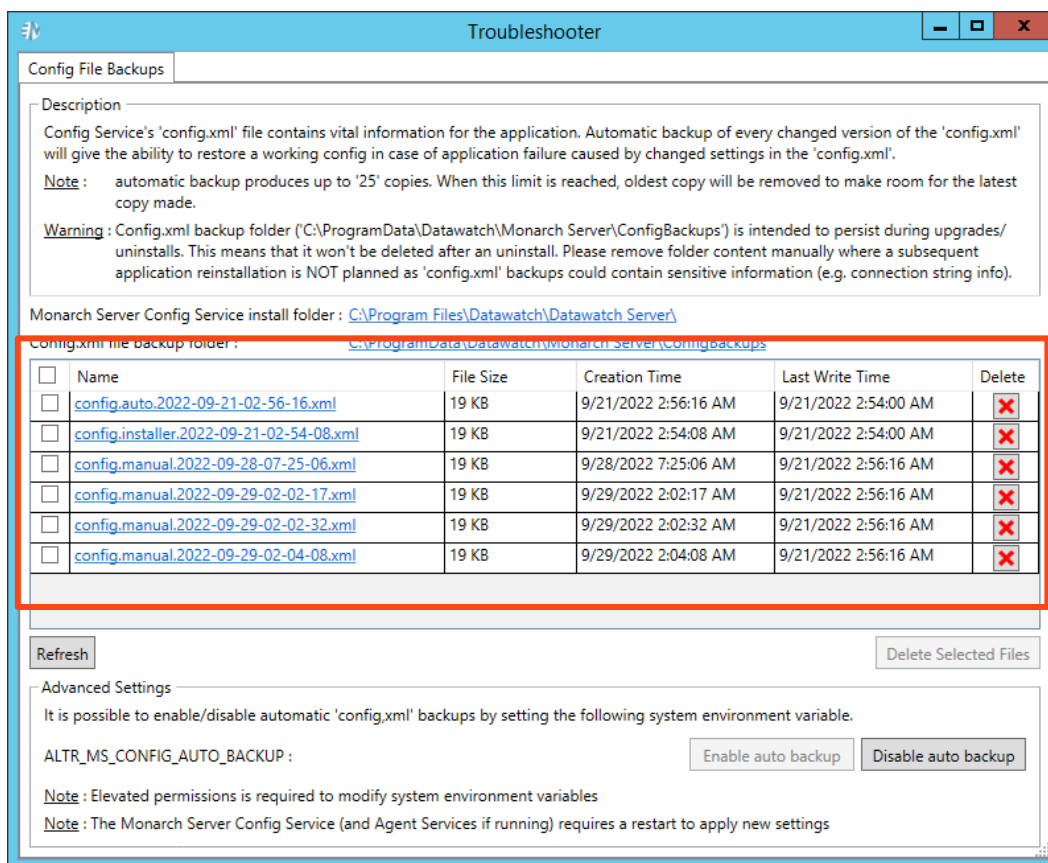


To view/delete the backup config.xml files


1. Launch the **Monarch Server Configurator**.

- In the **Configurator** window, click the **Troubleshooter** button  found near the drop-down menu on the upper-right corner.

The Troubleshooter window displays.



The list of backup files is displayed just below the Description panel. The list includes the following information:

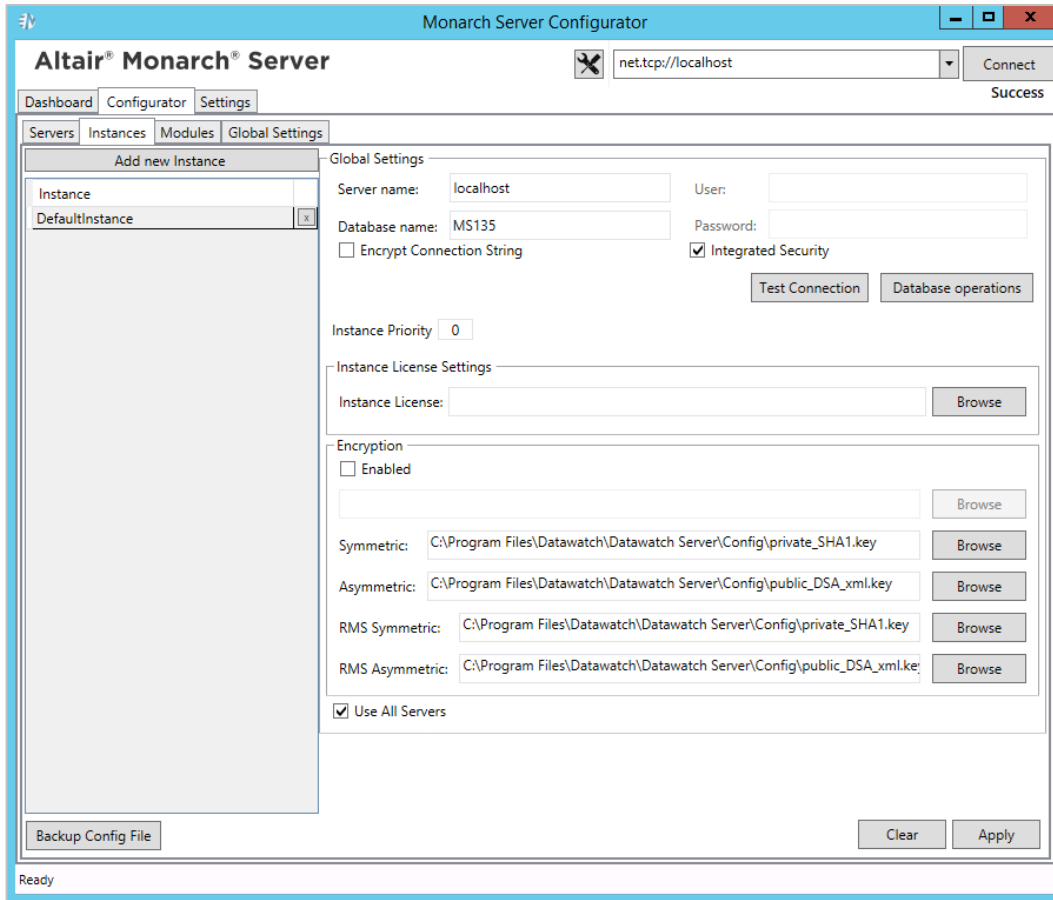
- Name:** File name of the backup file created.
 - File Size:** Size of the backup file.
 - Creation Time:** Date and time the backup file was created.
 - Last Write Time:** Date and time the backup file was last written to.
- To delete a backup file, click the corresponding **Delete** button  found in the **Delete** column of the backup list. You can also delete more than one backup file at a time by ticking their corresponding checkboxes and then clicking **Delete Selected Files**.

CONFIGURING INSTANCES

Creating an Instance

To create an instance:

- Select the **Configurator** tab, and click the **Instances** tab. The **Instance** page is displayed.



2. Under **Global Settings**, specify the database connection string parameters:

- Server name
- Database name
- User
- Password

NOTE This connection string will be used by services for database access.

3. Select the **Encrypt Connection String** check box if you want to encrypt connection string.
4. Select the **Integrated Security** check box if you want to access SQL Server via Windows Authentication. If you do so, the fields **User** and **Password** become unavailable.
5. Click **Test Connection**. If connection is successful, you will be able to perform database operations.

NOTE If there is no database with the specified name, create a new database.

6. In the **Instance Priority** field, enter value to define instance priority.
7. In the **Instance License** field, specify the path to the license file.

NOTE This license is used only for a particular instance. To set the global license used for all instances, see [Global Settings](#).

8. Under **Encryption**, specify the path to the encryption file. **Note:** You can specify the encryption file only if the **Enabled** check box selected.
9. Under **Symmetric**, specify the path to the private key file for checksum in symmetric mode for MS.
10. Under **Asymmetric**, specify the path to the public key file for checksum in asymmetric mode for MS.
11. Under **RMS Symmetric**, specify the path to the private key file for checksum in symmetric mode for RMS.
12. Under **RMS Asymmetric**, specify the path to the public key file for checksum in asymmetric mode for RMS.
13. The **Database Common Time-out** setting is configured manually in the config.xml file. Path to this file: C:\Program Files\Datawatch\Datawatch Server\Config\config.xml

This parameter is located in the <Database> of the <Instance>:


```
<Instance Name="DefaultInstance" Priority="1">
  <Database>
    ...
    <CommandTimeout>0</CommandTimeout>
  </Database>
  ...
</Instance>
```

The default timeout is 0 (it means no timeout).

To define the servers available for an instance:

1. Under **Servers available for Instance**, click the  button and select servers from the drop-down list.

NOTE In this case, an instance will be able to use only those services that are installed on the selected servers.

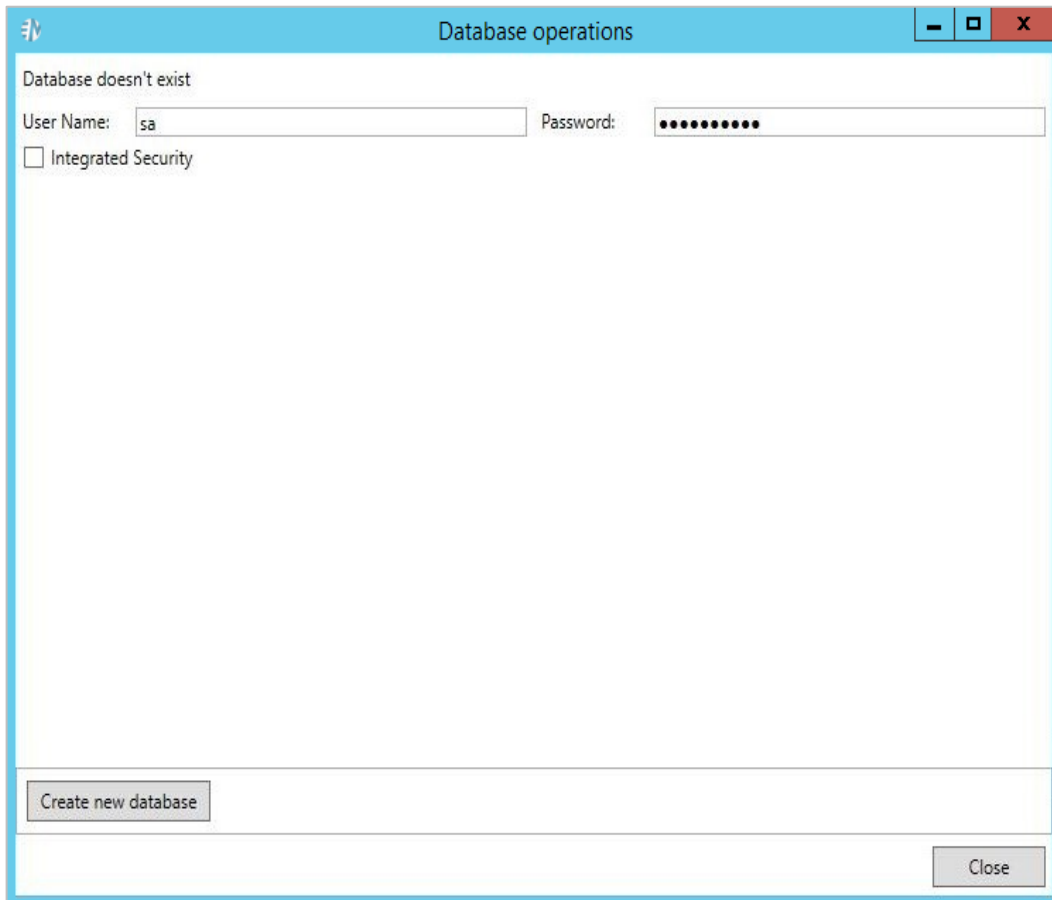
2. Select the **Use All Servers** check box, to allow instances use all servers.
3. Click **Apply**.
4. To add a new instance, click the **Add new Instance** button.
5. To delete an instance, click the  button by the instance name.

Creating a New Database

To create a new database:

1. On the **Instances** tab, click the **Database Operations** button.
2. In the **Database operations** window, enter your username and password.

NOTE Select the **Integrated Security** check box to use Windows Authentication. The specified user must have system administrator rights to create a new database.



Database operations

Database doesn't exist

User Name: sa Password:

Integrated Security

Create new database

Close

3. Click Create new database.
4. Click **Close**.

Updating Database

Before updating the database, the following points should be considered:

- ❑ **The user carrying out this updating process should be familiar with and have experience in administrating an SQL Server instance and its connectivity, availability, and maintenance.**

This upgrade operation involves the backup, possible reconfiguration, and restoration/recovery of the application SQL Server database in the event of a problem. In addition, a fuller version of SQL Server may be potentially used as part of the SDK functionality. In validating the upgrade, there may be a need to perform activities on the SQL Server database instance via the Management Studio.

- ❑ **The database should be backed up.**

A backup will allow for regression if problems occur and cannot be resolved within the desired upgrade window. This involves the following:

- The backup should be made when the Altair Monarch Server services are wholly stopped. This will involve the **Configurator and Agent Service**.
- In case of the use of more than one server, then the Agent Service on all other servers hosting modules must also be stopped (refer to your Configurator to determine your server(s) and their hostname(s)).
- For Virtual Servers, snapshots are also viable, but they should also be taken when the services are stopped to ensure that the snapshot is in a clean, restorable state.

NOTE Monarch Server can be deployed such that all components reside on one server, including its SQL Server database. However, for your upgrade, you should be able to identify if more than one server requires a snapshot. You should also verify whether a separate SQL Server backup should be performed if you are using an external SQL Server instance.

Consideration should also be given to the input and output resources as in some instances, these will need to be backed up as well to facilitate a clean recovery of all required resources in the event of total or partial regression. This process might also help assist in testing as elements can be restored post-testing to remove the effects of said test activities.

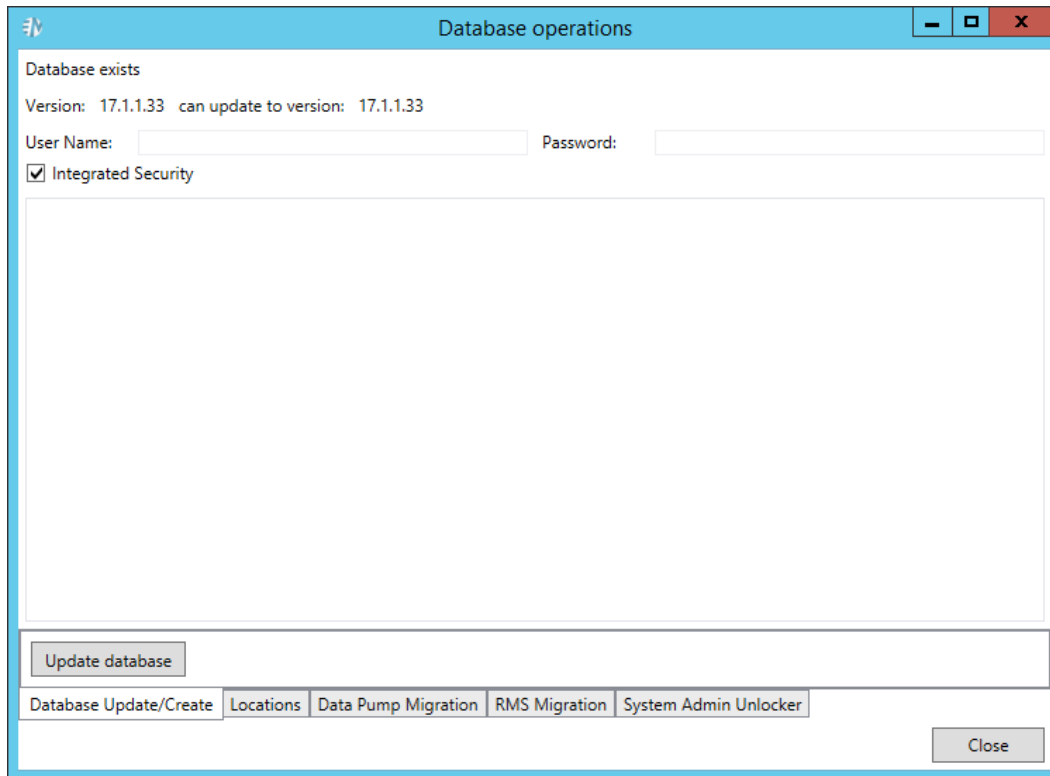
- ❑ A copy of the log file should be made by either copying and pasting the same or by grabbing the log file labeled as follows:

```
C:\Users[CurrentUser]\Documents\Datawatch\Log\update_dwch-net_db_YYYYMMDDHHMMSS.log
```

To proceed with updating the database:

1. On the **Instances** tab, click the **Database Operations** button.
2. In the **Database operations** window, enter your username and password.

NOTE Select the **Integrated Security** check box to use Windows Authentication. The specified user must have system administrator rights to update the database.



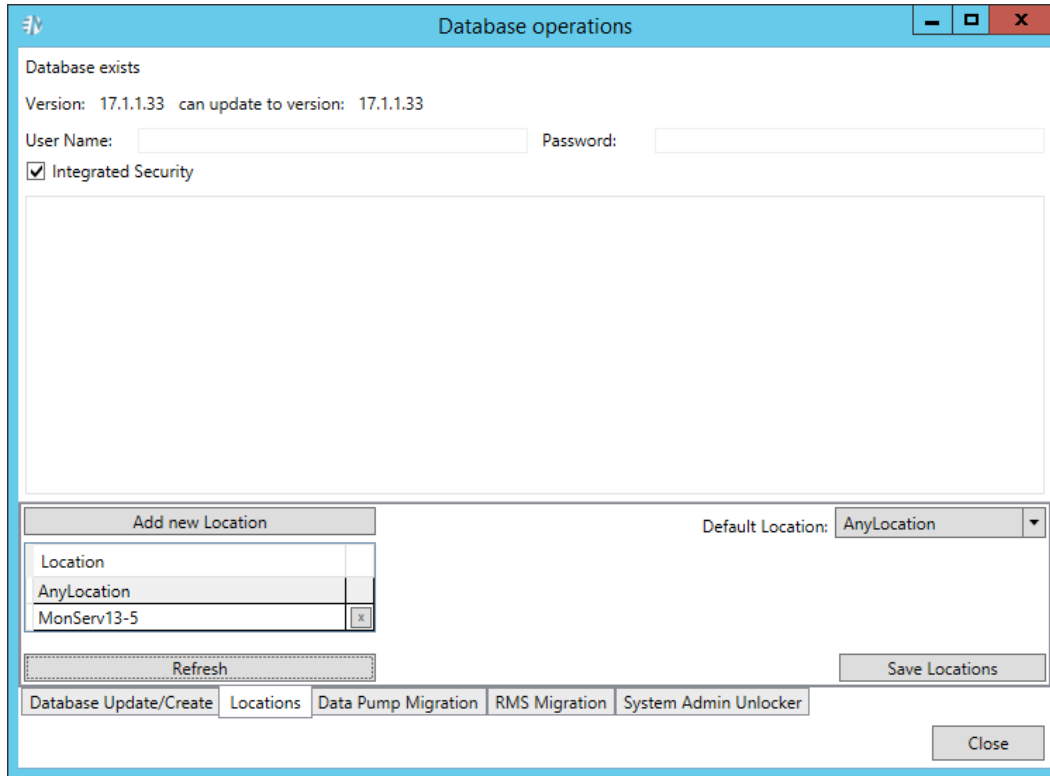
3. Click Update database.
4. Click **Close**.

Managing Locations

To add a new location:

1. On the **Instances** tab, click the **Database Operations** button.
2. In the **Database operations** window, enter your username and password.

NOTE Select the **Integrated Security** check box to use Windows Authentication. The specified user must have system administrator rights to update the database.



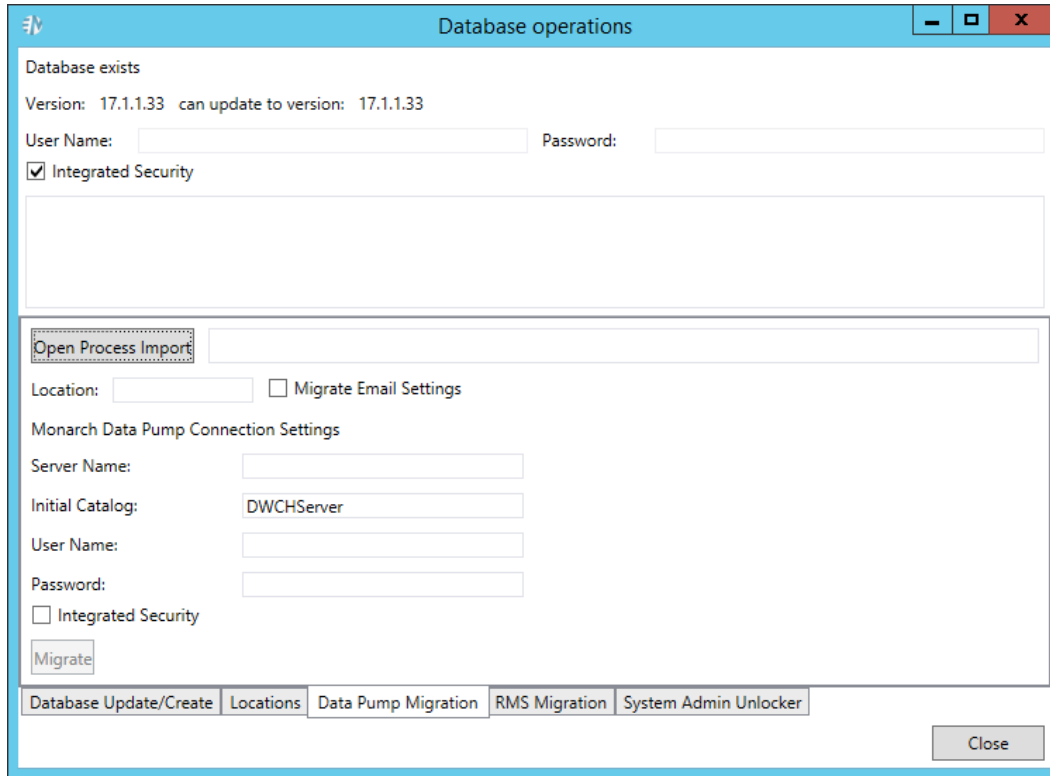
3. Click the **Refresh** button to display existing in the database locations.
4. Click the **Add New Location** button to add new location.
5. To edit a location, select its name and click the **Edit Location** button.
6. In the **Default Location** field, select default location from the drop-down list.
7. Click the **Save Locations** button to save changes.

Migrating Data

To import data:

1. On the **Instances** tab, click the **Database Operations** button.
2. In the **Database operations** window, enter your username and password.

NOTE Select the **Integrated Security** check box to use Windows Authentication. The specified user must have system administrator rights to update the database.



3. Select the Data Pump Migration tab.
4. Click the **Open Process Import** button to import processes. In the displayed *Open File* dialog box, choose the files with the imported from the previous Automator versions processes.
5. Select the *Migrate Email Settings* check box to transfer settings.
6. In the *Location* field, enter location name. The specified location will be used for the imported processes.
7. To set connection to the database of the previous Automator version, under Automator Connection Settings specify the following parameters:
 - Server Name
 - Initial Catalog
 - User Name
 - Password

NOTE Select the **Integrated Security** check box to use Windows Authentication.

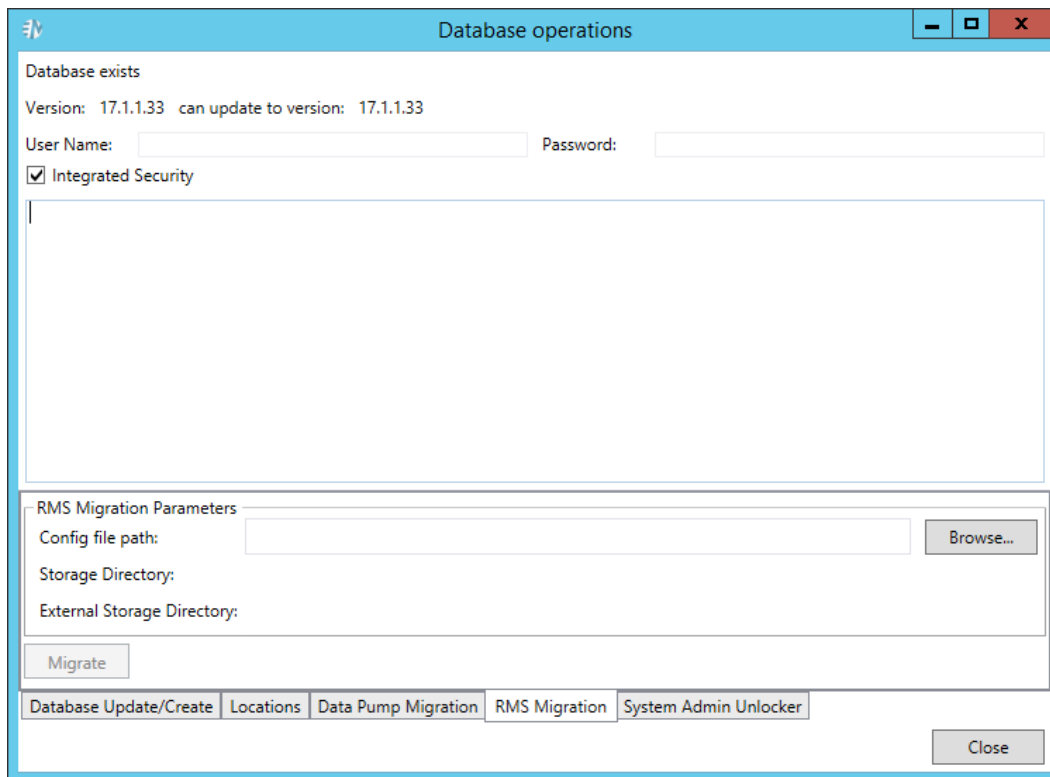
8. Click the **Migrate** button. The following data will be migrated to the current database:
 - Global distributions
 - Global scripts
 - Alerts
 - Email settings (if the **Migrate Email Settings** check box was selected)
 - Default job disposition settings.

RMS Migration

To import RMS data:

1. On the **Instances** tab, click the **Database Operations** button.
2. In the **Database operations** window, enter your username and password.

NOTE Select the **Integrated Security** check box to use Windows Authentication. The specified user must have system administrator rights to update the database.



The screenshot shows a window titled "Database operations" with a blue header. The main content area is divided into several sections. At the top, it says "Database exists" and "Version: 17.1.1.33 can update to version: 17.1.1.33". Below this are input fields for "User Name:" and "Password:". A checkbox labeled "Integrated Security" is checked. There is a large empty text area below the checkbox. The bottom section is titled "RMS Migration Parameters" and contains three input fields: "Config file path:" with a "Browse..." button, "Storage Directory:", and "External Storage Directory:". A "Migrate" button is located below these fields. At the very bottom, there is a tabbed interface with five tabs: "Database Update/Create", "Locations", "Data Pump Migration", "RMS Migration" (which is selected), and "System Admin Unlocker". A "Close" button is in the bottom right corner.

3. Select the **RMS Migration** tab.
4. Click the **Browse** button to select web config file from previous RMS versions that contains storage settings.
5. Click the **Migrate** button. The following data will be migrated to the current database:
 - RMS system settings
 - Users with their profiles
 - Models
 - View definitions

Unlocking System Administrator

To unlock system administrator:

1. On the **Instances** tab, click the **Database Operations** button.
2. In the **Database operations** window, enter your username and password.

NOTE Select the **Integrated Security** check box to use Windows Authentication. The specified user must have system administrator rights to update database.

Database exists
Version: 17.1.1.33 can update to version: 17.1.1.33
User Name: Password:
 Integrated Security

User	Reset password	Attempts count	Reset attempts	Locked	Default Password:
admin	<input type="checkbox"/>	0	<input type="checkbox"/>	<input type="checkbox"/>	password

Refresh Submit

Database Update/Create | Locations | Data Pump Migration | RMS Migration | System Admin Unlocker Close

3. Select the System Admin Unlocker tab.
4. Click **Refresh** to display available system administrators in the table.
5. Select the *Reset password* check box if you want to reset the current admin password to the default one.

NOTE You can modify the default password in the **Default Password** field.

6. Select the **Reset attempts** check box if you want to reset the number of login attempts to 0 for the corresponding administrator.

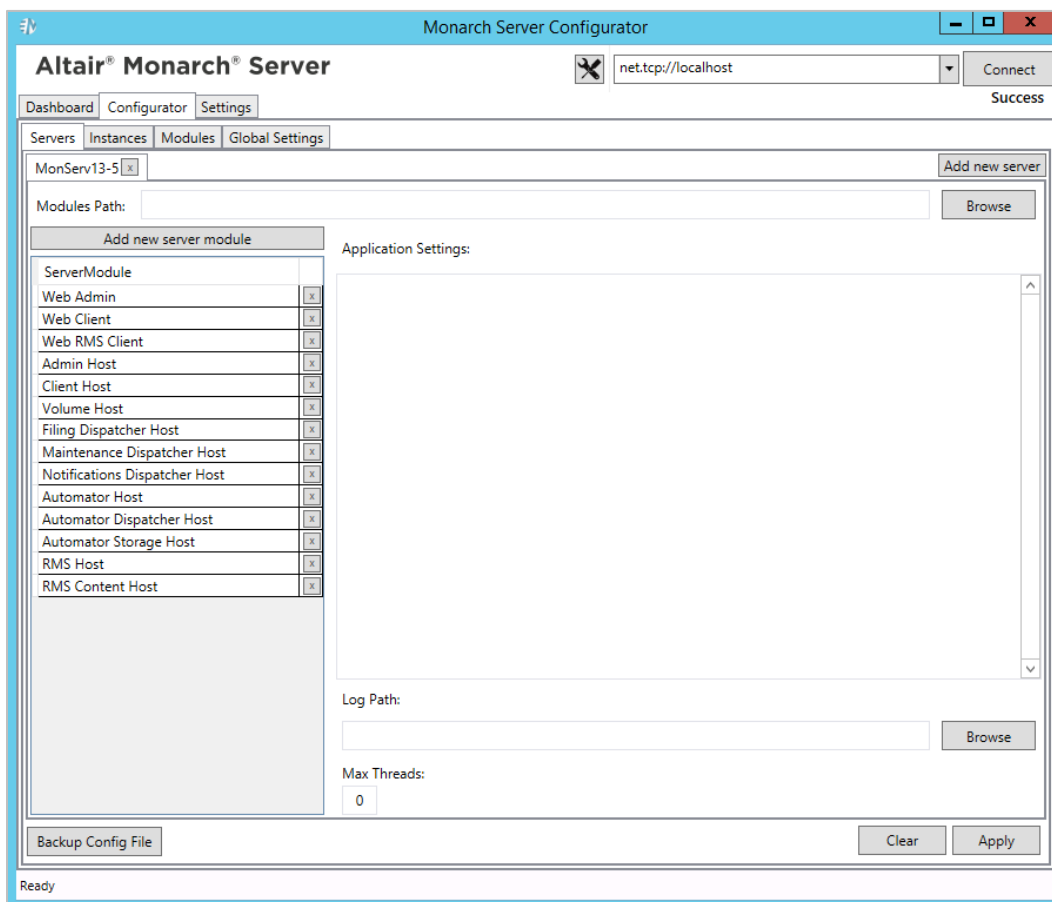
NOTE The number of failed attempts is displayed in the **Attempts count** field.

7. Select the **Locked** check box if you want to lock the corresponding administrator. To unlock the administrator, clear the check box.
8. Click **Submit** to apply changes.
9. Click **Close**.

CONFIGURING SERVERS

To configure the list of services that should run on the server:

1. Select the **Configurator** tab, and click the **Servers** tab. The Servers page is displayed.



2. Click the **Add New Server** button to add new services.
3. In the **Modules Path** field, specify the path for modules.
4. In the **Add New Server** window, specify the server module you want to add.
5. In the **Application Settings** field, specify the application settings for the required host.

Host application settings should observe the following format:

```
<configuration>
  <appSettings>
    <add key="parameterName" value="parameterValue" />
    ...
  </appSettings>
</configuration>
```

```
</appSettings>
</configuration>
```

The table below specifies host application settings.

MODULE NAME	PARAMETER NAME	DESCRIPTION	DEFAULT VALUE
Admin Host	LdapPort	Indicates the port for LDAP communication	636
	WorkPath	Indicates the location where SDK creates temp files	
Filing Dispatcher Host	AutoStartFilingInterval	Interval between which the job for auto start Filing Tasks will be started.	00:00:10
	FilingDirListenerInterval	Interval between which new job for auto start Filing Tasks will be created in the ScheduleJobs table.	00:00:10
	AutoStartFilingPortion	Indicates how many automatic filing tasks will be run during AutoStartFilingInterval interval.	10
	FileLockAttempts	Indicates how many times the application attempts to check if a file is unlocked Note: If the file was not unlocked after the defined combination of FileLockAttempts and FileLockWaitPeriod, an email notification is sent to the application Admins advising of the filing task/process, affected filename, and processing location/server.	6
	FileLockWaitPeriod	Indicates the interval between attempts to check if a file is unlocked Note: If the file was not unlocked after the defined combination of FileLockAttempts and FileLockWaitPeriod, an email notification is sent to the application Admins advising of the filing task/process, affected filename, and processing location/server.	500
	SdkRegexTrapTimeout	Indicates the time interval during which RegEx checks every row of processed reports for pattern matching	30
	TempDbCleanEnabled	Indicates whether TempDB is checked for cleaning. Note: If set to "True" TempDB is checked every 1 minute (default value). TempDB is cleaned if it exceeds 1000 MB (default value) and if localDB is selected for SDK.	False
TempDbCleanInterval	Indicates the time interval during which TempDB is cleaned. Note: The value should be formatted as d.hh:mm[:ss.ff]. The number of days should be less than or equal to 24. Days are counted after the last restart of the host.	00:01:00	

MODULE NAME	PARAMETER NAME	DESCRIPTION	DEFAULT VALUE
	TempDbCleanSizeThresholdMb	Indicates the size of TempDB, at which point it should be cleaned. Note: The value is counted in megabits.	1000
	WorkPath	Indicates the location where SDK creates temp files	
Automator Host	SecurityProtocols	Allows the configuration of supported security protocols. Note: By default, security protocols depend on the .NET framework version. Possible values are "SystemDefault," "Ssl3," "Tls," "Tls11," and "Tls12."	Tls11, Tls12
	AutoStartPumpTaskInterval	Indicates the time interval during which the pump process will be started.	00:00:10
	PumpMonitoringInterval	Interval between which the system is monitored for new files.	00:00:10
	AutomatorMonitoringSuspended	Suspends all Automator monitoring.	False
	PumpTaskLegacyBehavior	Indicates the use of the old behavior of process handling Note: The old and new behaviors are described in the Process Handling Scenarios section of the Administrator's Guide.	True
	SuppressIndexingExternalLookupTables	Manages the creation of an appropriate index on each external lookup table during its building process Note: If set to "true," no indexes will be created. If set to "false," indexes will be created.	True
	SdkDeleteDbOnExit	Allows SDK to delete unused temp databases after export	False
	SdkShrinkDbOnExit	Allows SDK to compress temp databases after export	False
	SdkMaxCatalogs	Specifies the maximum number of cache databases that the SDK creates during export	3
	SdkRegexTrapTimeout	Indicates the time interval during which RegEx checks every row of processed reports for pattern matching	30
	SdkEnableAceDaoExportByDefault	Refers to the behavior of the process when exporting to .accdb, .dbf, and .mdb. Note: The default value is "true" (ACE DAO). Thus, it is not necessary to add the key unless this needs to be set to "false" (use ACE OleDb by default).	True
	WorkPath	Indicates the location where SDK creates temp files	
	BulkBehaviorExportBatchSize	Allows batch size to be configured for bulk insert and update	10000
	BulkBehaviorExportCommandTimeout	Allows timeout to be configured for bulk insert and update	30
Automator Dispatcher Host	TempDbCleanInterval	Indicates the time interval during which TempDB is cleaned.	00:01:00

MODULE NAME	PARAMETER NAME	DESCRIPTION	DEFAULT VALUE
		Note: The value should be formatted as d.hh:mm[:ss.ff]. The number of days should be less than or equal to 24. Days are counted after the last restart of the host.	
	TempDbCleanEnabled	Indicates whether TempDB is checked for cleaning. Note: If set to "True" TempDB is checked every 1 minute (default value). TempDB is cleaned if it exceeds 1000 MB (default value) and if localDB is selected for SDK.	False
	TempDbCleanSizeThresholdMb	Indicates the size of TempDB, at which point it should be cleaned. Note: The value is counted in megabits.	1000
	SecurityProtocols	Allows the configuration of supported security protocols. Note: By default, security protocols depend on the .NET framework version. Possible values are "SystemDefault," "Ssl3," "Tls," "Tls11," and "Tls12."	Tls11, Tls12
Automator Storage Host	StorageDirectory	Path to the Storage Directory for temp files of Automator.	
	TimeToStoreFiles	Time for how long temp files are stored in Storage Directory.	00:10:00
Notifications Dispatcher Host	OfflineTaskCacheDirectory	Path to the Storage Directory for Offline Tasks data.	
Client Host	TempDbCleanEnabled	Indicates whether TempDB is checked for cleaning. Note: If set to "True" TempDB is checked every 1 minute (default value). TempDB is cleaned if it exceeds 1000 MB (default value) and if localDB is selected for SDK.	False
	TempDbCleanInterval	Indicates the time interval during which TempDB is cleaned. Note: The value should be formatted as d.hh:mm[:ss.ff]. The number of days should be less than or equal to 24. Days are counted after the last restart of the host.	00:01:00
	TempDbCleanSizeThresholdMb	Indicates the size of TempDB, at which point it should be cleaned. Note: The value is counted in megabits.	1000
	LdapPort	Indicates the port for LDAP communication	636
	WorkPath	Indicates the location where SDK creates temp files	
RMS Content Host	ApiUploadStoragePath	Path to the temporary folder, which is created with the uploaded reports (default path: <host name>\C:\Windows\Temp\Datawatch RmsApiUploadStorage) Note: The folder containing the uploaded reports are removed at the end of the session.	

MODULE NAME	PARAMETER NAME	DESCRIPTION	DEFAULT VALUE
	ApiUploadSessionTimeOut	The time at which the session ends. Note: The resulting link will no longer be available at the end of the session.	01:00:00
RMS Host	TempDbCleanEnabled	Indicates whether TempDB is checked for cleaning. Note: If set to "True" TempDB is checked every 1 minute (default value). TempDB is cleaned if it exceeds 1000 MB (default value) and if localDB is selected for SDK.	False
	TempDbCleanInterval	Indicates the time interval during which TempDB is cleaned. Note: The value should be formatted as d.hh:mm[:ss.ff]. The number of days should be less than or equal to 24. Days are counted after the last restart of the host.	00:01:00
	TempDbCleanSizeThresholdMb	Indicates the size of TempDB, at which point it should be cleaned. Note: The value is counted in megabits.	1000
	LdapPort	Indicates the port for LDAP communication	636
	WorkPath	Indicates the location where SDK creates temp files	

6. In the **Log Path** field, specify the log path.

NOTE



- The path specified is resolved in the context of the server that the setting relates to. Therefore, in clicking the Browse button when adding a log path to a new server, the path is assumed to be resolvable to the specified server. This case should be manually verified.
- Some actions (such as test connection when using Windows authentication) are carried out in the context of current users and NOT in the context of the service account.

7. In the **Max Threads** field, define the maximum number of threads.

NOTE

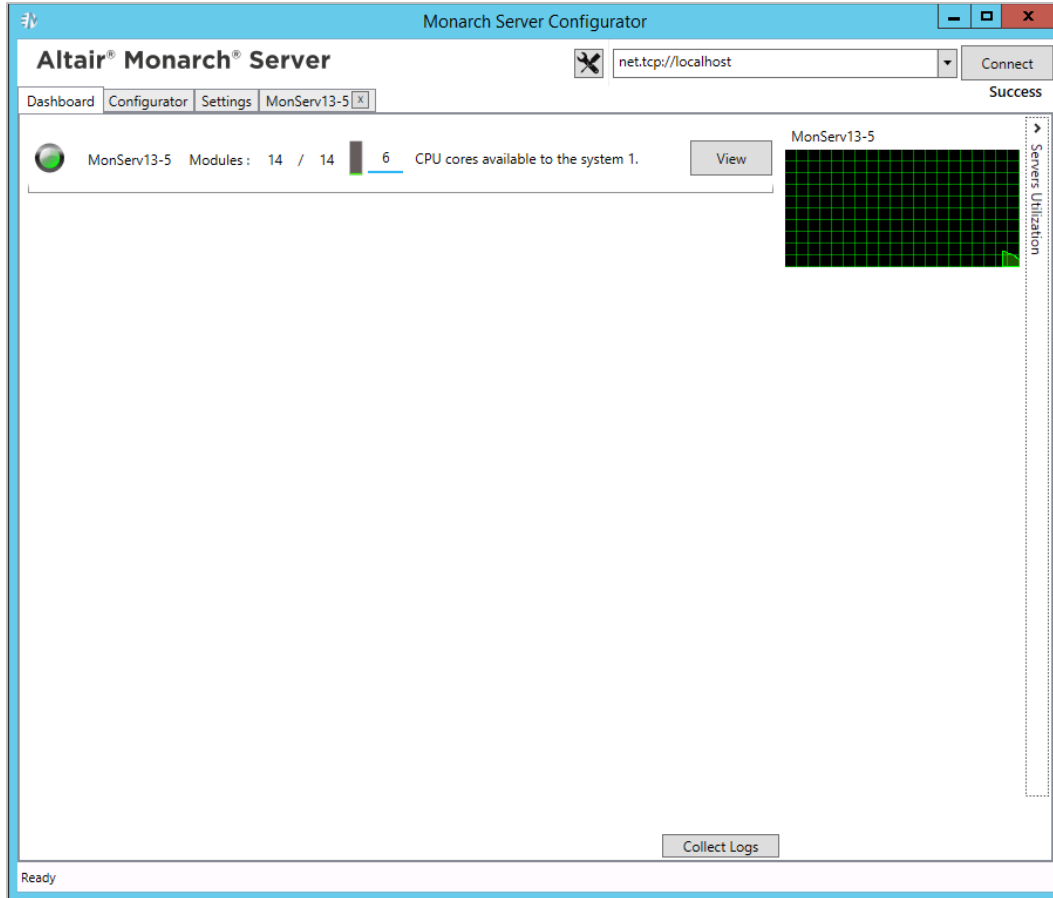
- The Max Threads setting is useful for the following hosts, which are able to run some jobs or tasks:
 - Maintenance Dispatcher Host
 - Filing Dispatcher Host
 - Automator Host

- Automator Dispatcher Host
- Notifications Dispatcher Host
- This option stands for the number of tasks/processes that are fulfilled at a time for the corresponding module.
- The Max Threads setting is used in the SmartThreadPool, which accepts a maximum of 25 threads (the same for 0 Max Threads or empty). Increasing the Max Threads setting for some hosts could affect the performance of other hosts. Moreover, large Max Threads could affect the performance of the host itself because of the context switching between all thread pools. Thus, the optimal value lies in the middle and depends on the environment, number of processor cores, and network.

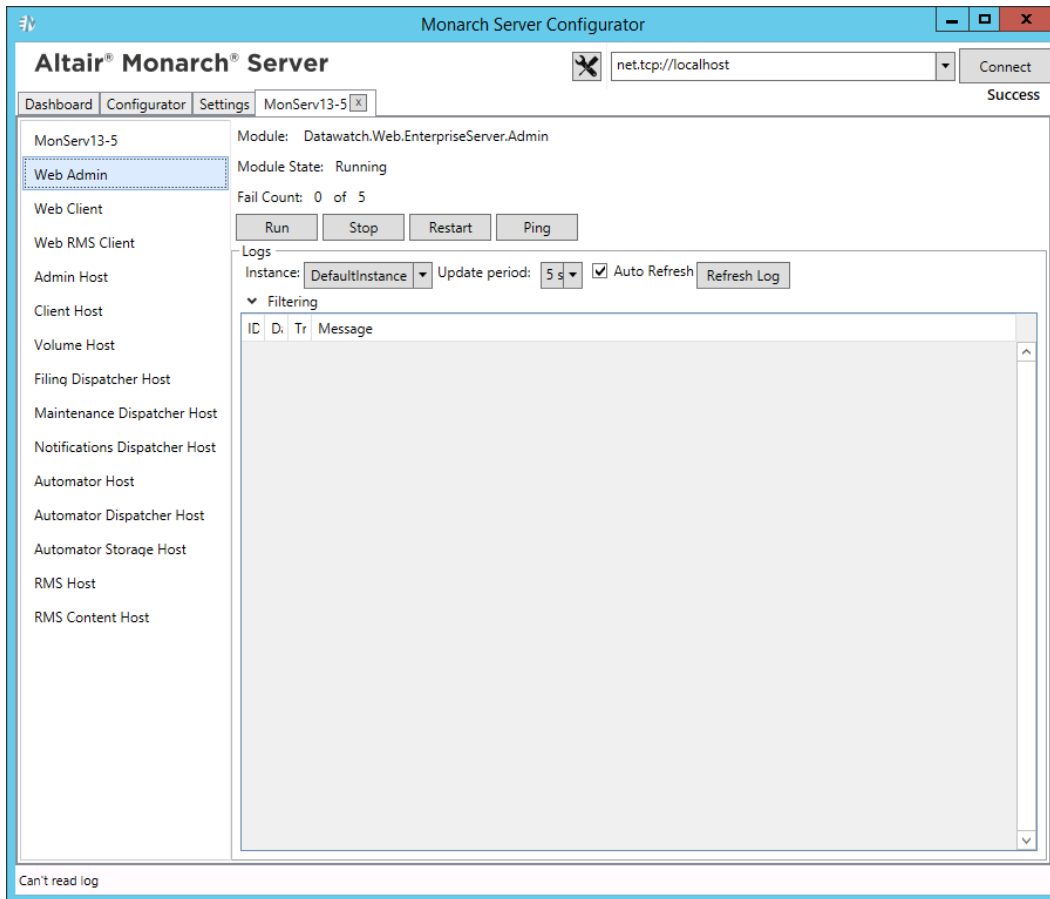
8. In the **SDK Settings** section, specify the export storage settings:
 - Select **Local DB** database type, if you want to use SQL Local DB as an export storage and specify **Database File Path** (if Database File Path is not specified, database files will be created in the user's directory %LOCALAPPDATA%\Datawatch\DatawatchEngine).
 - Select **Remote DB** database type, if you want to use SQL Server as an export storage and specify its parameters:
 - ◆ Server Name
 - ◆ Authentication credentials: user name and password. If you want to use Windows authentication, select the corresponding check box.
9. To enable SDK trace logging, set **Trace Log Folder** and **Trace Log Level** (0 – none, 1 – min, 2 – max).
10. Click **Apply** to save changes.
11. To delete a service, click the  button by the service name.
12. To add a new server, click the **Add new Server** button.
13. To delete a server, click the  button by the server name.

MANAGING SERVER MODULES

To view the server log information and manage server modules, click the **View** button on the Dashboard page.



The Server page opens in a new tab.



To perform an action to all server modules

1. Select the name of the server from the list in the left side of the window.
2. Click one of the following buttons: **Run All**, **Stop All** or **Reload All**.

NOTE When you click the **Stop All** button, you can specify the timeout value in seconds to stop all the running processes in the **Timeout** field.

To perform an action to a single module

1. Select the module name from the list in the left side of the window.

NOTE You can check the Module State and Fail Count for every module. Max fails can be specified for all modules in the [Configuring Modules](#).

2. Click one of the following buttons: **Run**, **Stop**, **Restart** or **Ping**.

To customize the table with the log information, the following options are available:

- Instance:** The drop-down list shows all instances created for all servers.

NOTE This option is only available for the Web Admin, Web Client, and Web RMS Client modules.

- Update Period:** Select value for the update period from the dropdown list.
- Auto Refresh** check box: Select this check box, to enable automatic log refreshing.
- Refresh Log** button: Click the **Refresh Log** button, to refresh the log information.
- Filtering:** Click the drop-down arrow and select filters. To select all filters, click the **Check All** button. To cancel selection of all filters, click the **Uncheck All** button.
- Details row:** Click anywhere on a particular row in the table to view details.

NOTE The table displays the log information about the item currently selected in the left side of the window.

To restart hosts/modules with CLI/CMD

Hosts and modules can be restarted via the command line interface. The file path to the terminal is

```
C:\Program Files\Datawatch\Datawatch
Server\Configurator\Datawatch.ModulesManagement.Configurator.Terminal.exe
```

The following commands are available:

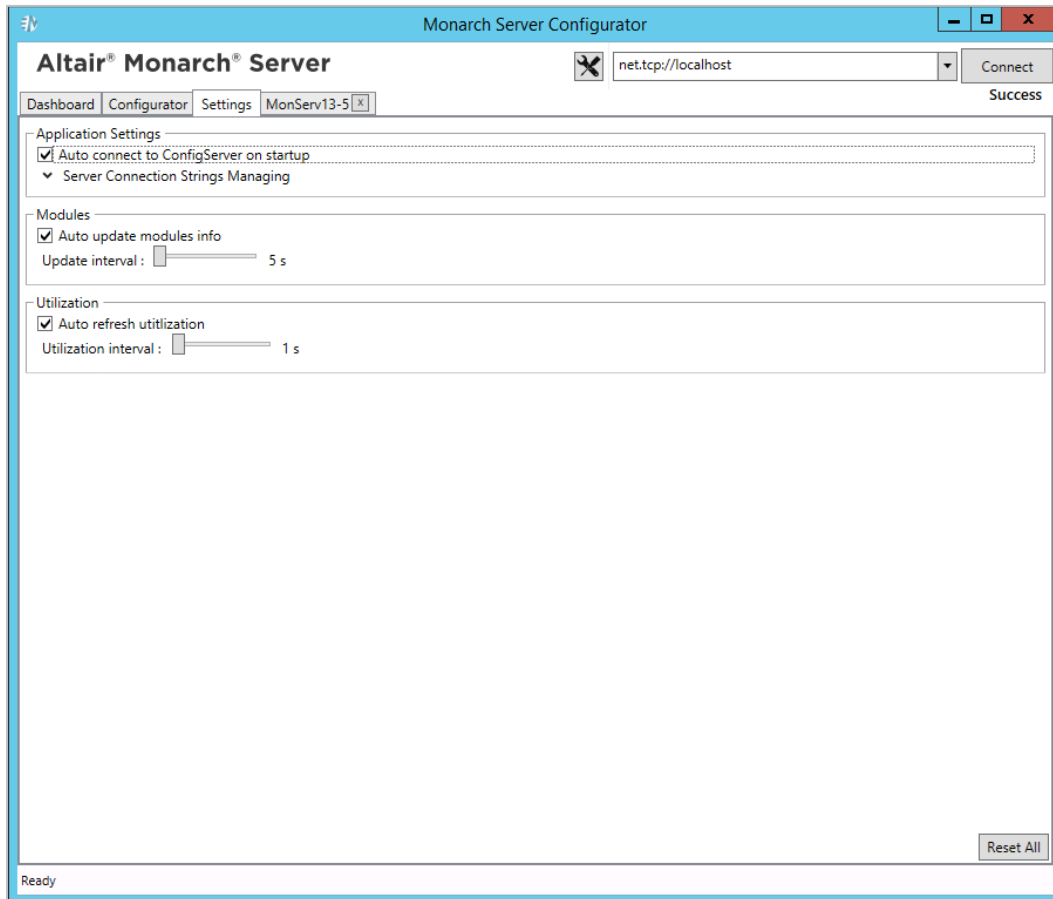
COMMAND	DESCRIPTION
--?, --help	Displays the help screen
--restartAllModules	Restarts all modules in specified server; requires serverName as parameter
--restartAllModulesInAllServers	Restarts all modules in all servers
--restartModulesInServers	Restarts module(s) in server(s); requires moduleName and serverName as parameters

The following parameters are required in some commands:


PARAMETER	DESCRIPTION
--serverName	Parameter to set server name for restarting the modules in that server. Multiple server names can be provided by separating them with a colon. Example: --serverName=ServerA.domain:ServerB.domain --
--moduleName	Parameter to set module names to restart. Multiple module names can be provided by separating them with a colon. Example: --moduleName=Datawatch.RemoteServices.ClientServicesHost:Datawatch.RemoteServices.DataPumpServicesHost

CONFIGURATOR SETTINGS

To proceed to the Configurator Settings page, click the **Settings** tab.



The following groups of settings are available:

- Application Settings
 - Select the **Auto connect to ConfigServer on startup** check box, to connect automatically to the ConfigServer on launching Configurator.
 - Click the drop-down arrow, to view the list of Config Services addresses. To delete address, click the  button by its name.
- Modules
 - Select the **Auto update modules info** check box, to update the modules information automatically.
 - Move the slider, to set the update interval value.

NOTE You can set the update interval only if the **Auto update modules info** check box is selected.

- Utilization

- Select the **Auto refresh utilization** check box, to refresh utilization automatically.
- Move the slider, to set the utilization refresh interval value.

NOTE You can set the utilization refresh interval only if the **Auto refresh utilization** check box is selected.

To reset all settings, click the **Reset All** button.

GLOBAL SETTINGS

To customize global settings, select the **Configurator** tab, and click the **Global Settings** tab.

The following settings are available on the Global Settings page:

- ❑ **Global License Settings:** Modify the global license source to be used. The specific steps are detailed [here](#).
- ❑ **Global Settings:**
 - **Log level:** Select the log level from the drop-down list (the default value is **All**).
 - **Log size (kb):** Enter the log size (the default value is **1024**).
 - **Log max archived files:** Enter the max number of archived files (the default value is **50**).

- **Global timeout:** Enter to set receiveTimeout for all URI schemes. The receiveTimeout setting can be configured manually for each particular scheme via config.xml.

Path to this file: C:\Program Files\Datawatch\Datawatch Server\Config\config.xml

This parameter is located in the <binding> of the <basicHttpBinding>, <wsHttpBinding>, <netTcpBinding>, <netNamedPipeBinding>:

```
<bindings>
  <basicHttpBinding>
    <binding name="basicHttpBinding"
      openTimeout = "00:10:00"
      sendTimeout = "00:10:00"
      closeTimeout="00:10:00"
      receiveTimeout = "01:00:00"
      maxReceivedMessageSize="2147483647"
      transferMode="Streamed">
      <readerQuotas
        maxStringContentLength="2147483647"
        maxArrayLength="2147483647" />
    </binding>
  </basicHttpBinding>
  ...
</bindings>
```

To save changes, click **Apply**.

NOTE User session ID may be logged by modifying the config file as follows:

- Open the config.xml file and locate the logging template field.
 - Modify this field by adding {keyvalue(sessionId)}.
- The field should appear as:
 - template="{[timestamp(local:G)] {severity} {keyvalue(sessionId)}} {message}"

[For local:G, additional information is available [here](#).]

For a new installation of Monarch Server

The config.xml file may be modified as follows:

```
<formatters>
  <add
    type="Microsoft.Practices.EnterpriseLibrary.Logging.Formatters.TextFormatter,
    Microsoft.Practices.EnterpriseLibrary.Logging, Culture=neutral"
    template="{[timestamp(local:M/d/yyyy h:mm:ss.ffff tt)] {severity}
    {keyvalue(sessionId)}} {message}"
    name="Text Formatter" />
</formatters>
```

For an upgrade to the latest version of Monarch Server

- a. Perform the [upgrade](#) as usual.
- b. Open the config.xml file and locate the old template field:

```
<formatters>
  <add
    type="Microsoft.Practices.EnterpriseLibrary.Logging.Formatters.TextFormatter,
    Microsoft.Practices.EnterpriseLibrary.Logging, Culture=neutral"
    template="[{timestamp(local:G)} {severity}] {message}"
    name="Text Formatter" />
</formatters>
```

- c. Replace this field with following:

```
<formatters>
  <add
    type="Microsoft.Practices.EnterpriseLibrary.Logging.Formatters.TextFormatter,
    Microsoft.Practices.EnterpriseLibrary.Logging, Culture=neutral"
    template="[{timestamp(local:M/d/yyyy h:mm:ss.ffff tt)} {severity}
    {keyvalue(sessionId)}] {message}"
    name="Text Formatter" />
</formatters>
```

- d. Save the changes.
- e. Restart Services to apply the changes made.

Configuring Global License Settings

Monarch Server supports two global license sources: Datawatch license and Altair license. This section details these two sources.

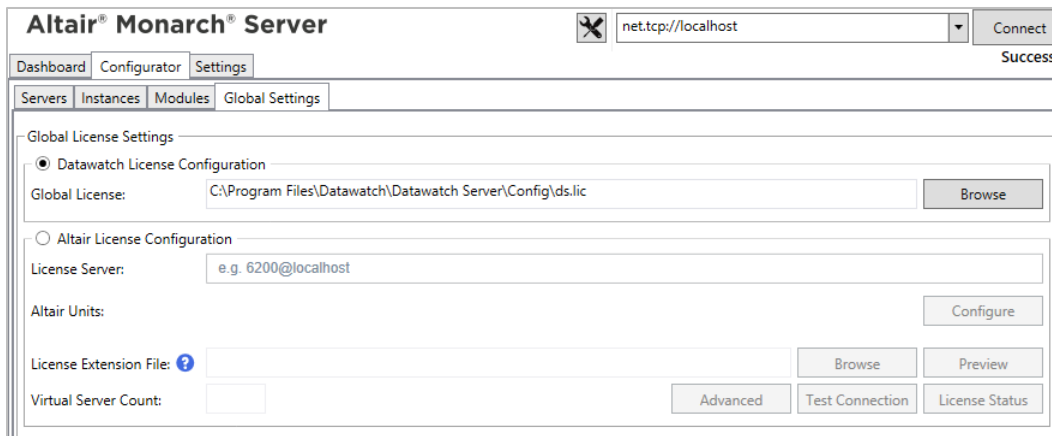
Datawatch license configuration

Use this configuration if you have a license file located in a local folder or network.

NOTE Global license is used for all instances. To set a separate license for a particular instance, see [Configuring Instances](#), point 6.

Steps:

1. Launch **Monarch Server Configurator** from the **Start** menu.
2. In the Configurator window, click the **Configurator** tab.
3. From the available tabs, select **Global Settings**.
4. Under *Global License Settings*, select **Datawatch License Configuration**.



5. Specify the full path (e.g., “C:\License\ms.lic”) or relative path (e.g., “ms.lic,” with the base directory being the “[InstallDir]\Config” folder) of the license file or click **Browse** to manually search for it.
6. To save your settings, click **Apply**.

Altair License Configuration

Use this configuration if you have access to Altair’s License server or Altair Units. Altair Units is a cloud-based licensing system that provides seamless access to Altair licenses through an internet connection.

NOTE Altair Units requires valid Altair Connect credentials. Check with your administrator for access.

Steps:

1. Launch **Monarch Server Configurator** from the **Start** menu.
2. In the Configurator window, click the **Configurator** tab.
3. From the available tabs, select **Global Settings**.
4. Under *Global License Settings*, select **Altair License Configuration**.
5. If your license source is **Altair’s License Server**, perform the following steps:
 - a. Specify the location of the Altair License Server in the **License Server** field with the following format: port@hostname (e.g., 6200@localhost).

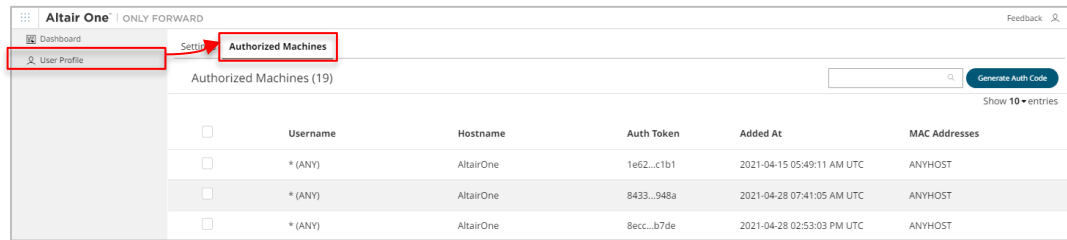
NOTE Multiple license servers can be specified. Use the semicolon to separate the license servers (e.g., 6200@localhost;6200@externalALS).

- b. Save your settings by clicking **Apply**.

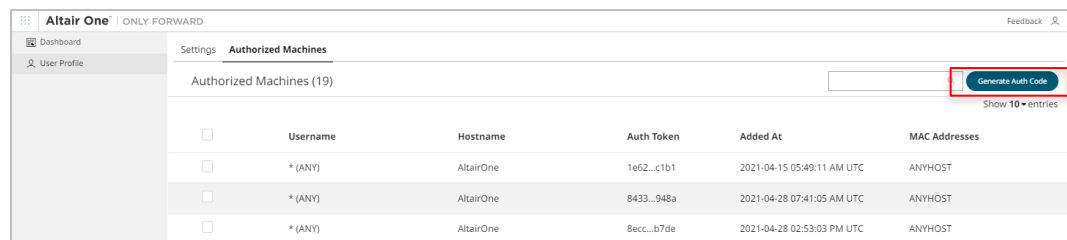
If your license source is **Altair Units**, perform the following steps:

- a. Generate the **Auth Code** first:

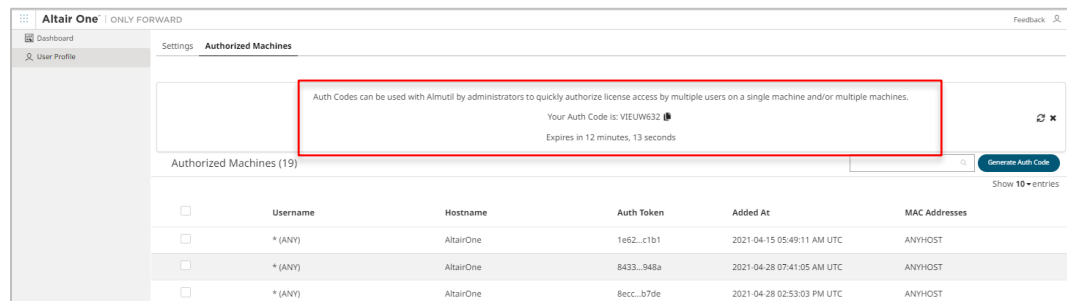
- i. Launch Google Chrome and navigate to the Altair One Admin portal at <https://admin.altairone.com/login>. Use your credentials to log in.
- ii. In the portal, click **User Profile** and then click the **Authorized Machines** tab in the main display.



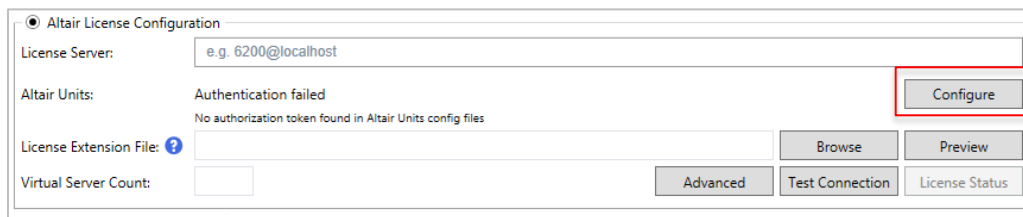
- iii. Click **Generate Auth Code**.



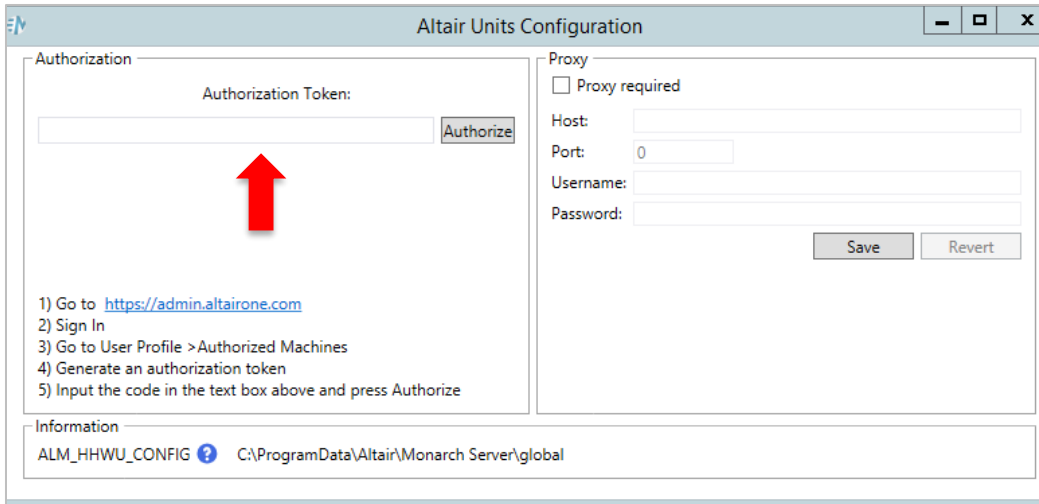
The page displays the auth code and a timer indicating the code's expiration.



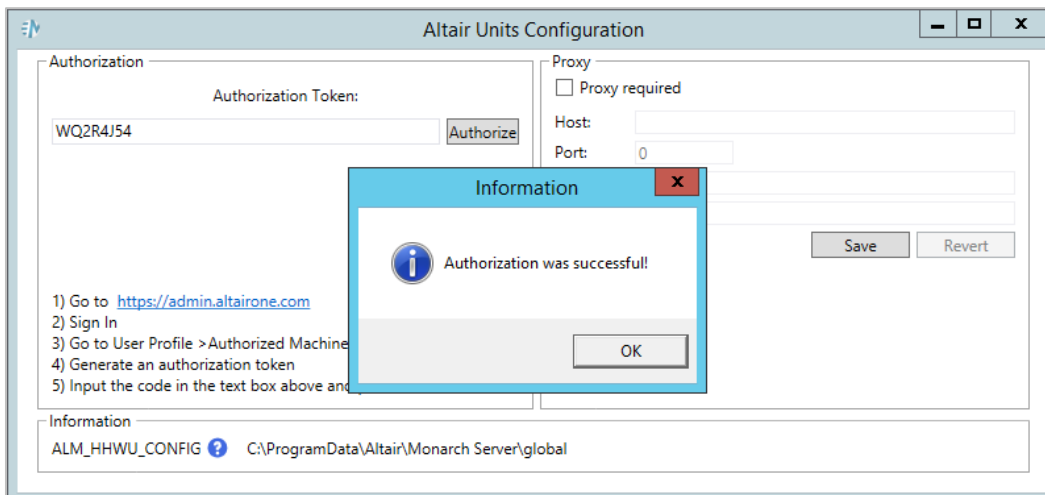
- b. Return to the Configurator and launch the **Altair Units Configuration** dialog by clicking **Configure**.



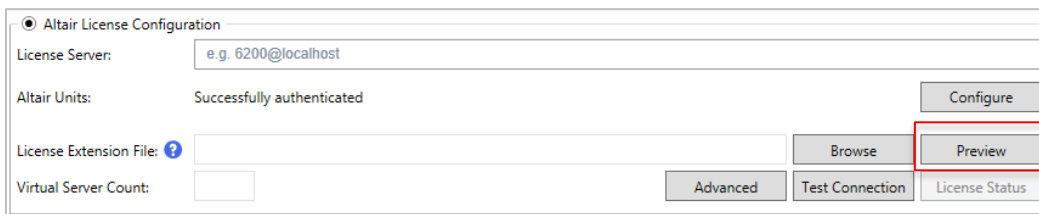
- c. In the Altair Units Configuration dialog, enter the **Auth Code** you generated in Step iii and then click **Authorize**.



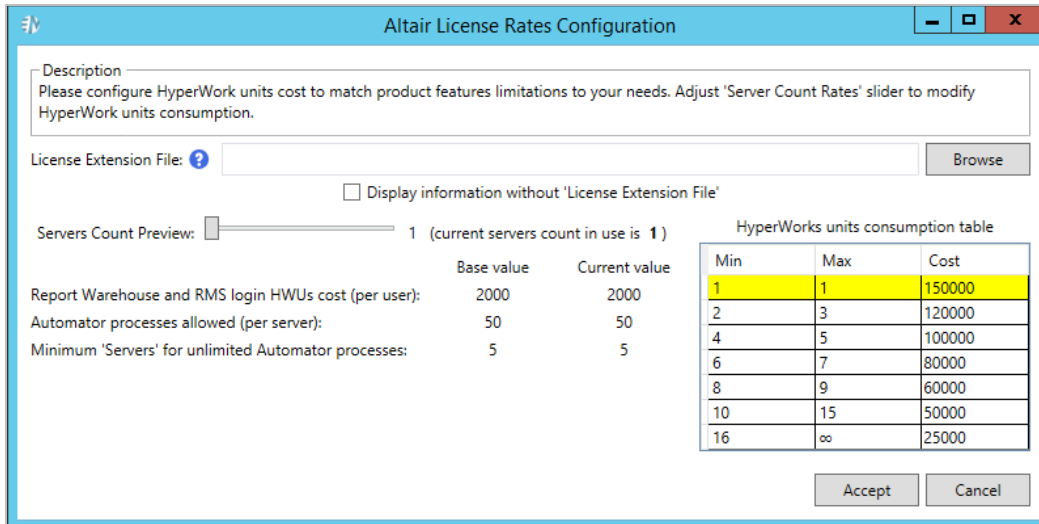
If the auth code is successfully authorized, the following prompt displays:



- d. Click **OK** and then close the Altair Units Configuration dialog.
- e. You can then modify the number of Altair Units that will be consumed to meet your specific requirements. For this purpose, click the **Preview** button in the Global Settings tab of the Configurator to launch the **Altair License Rates Configuration** window.



- f. In the **Altair License Rates Configuration** window, you can adjust the **Server Count Rates** slider to modify the unit consumption. Click **Accept** to save your settings.



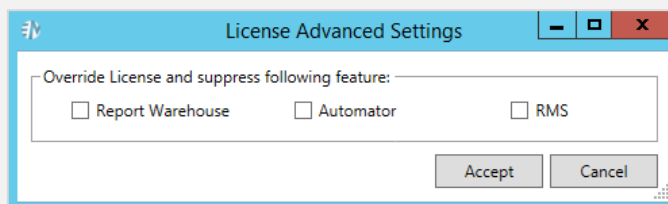
NOTE The default unit consumption is as follows: 2,000 units per user when logged in to MSCClient and RMSClient, 50 Automator processes per server, and a minimum number of 5 servers for an unlimited number of processes.

The default costs may be changed when using a license extension file.

- g. Click the **Test Connection** button in Configurator to check the properties of the currently specified Altair license and view their detailed information.
- h. Click the **License Status** button to view the currently applied Altair license and its detailed information.
- i. Save all your settings by clicking **Apply**.

NOTE Altair licensing (i.e., License Server, Altair Units) for Monarch Server covers all three components, namely, Report Warehouse, Automator, and RMS. If you do not intend to use all three components, you can override the license and suppress specific components. To do this, perform the following:

- a. Launch **Monarch Server Configurator** from the **Start** menu.
- b. In the **Configurator** window, click the **Configurator** tab.
- c. From the available tabs, select **Global Settings**.
- d. Under **Altair License Configuration** in the **Global License Settings**, click the **Advanced** button.
The **License Advanced Settings** dialog box displays.

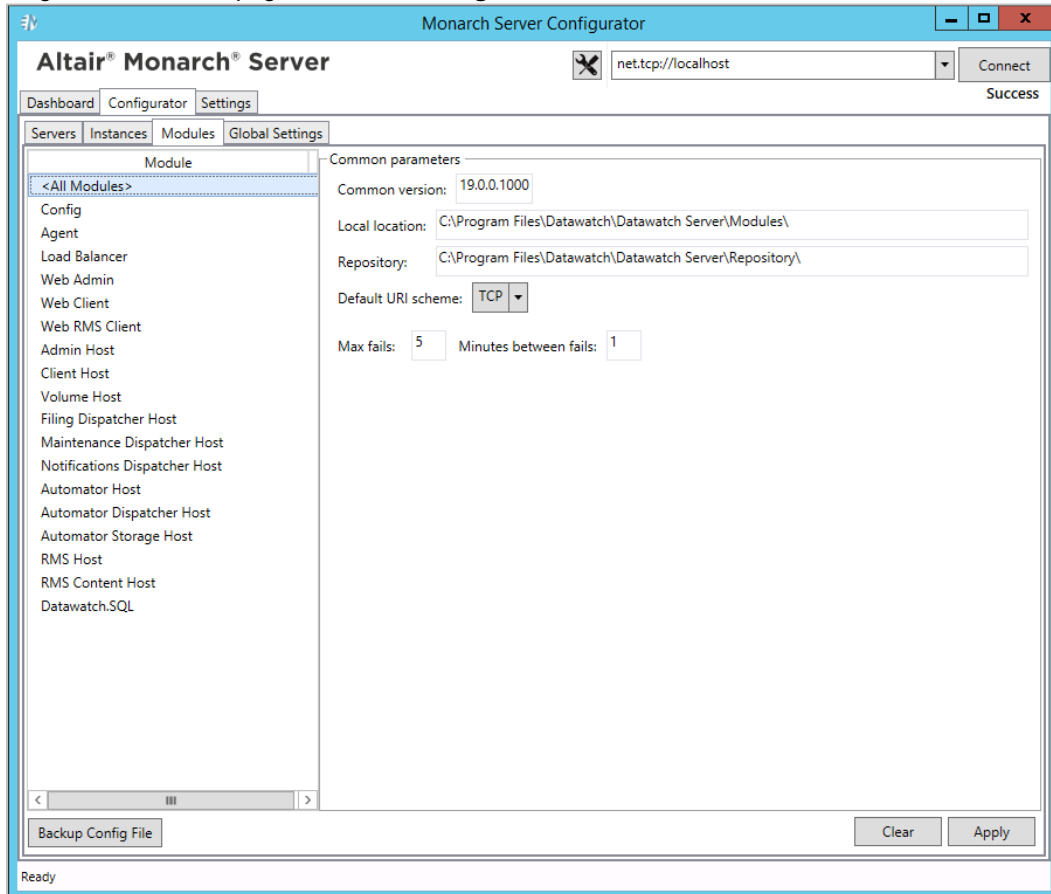


- e. Select the checkboxes corresponding to the components you want to suppress. Note that one checkbox should remain unselected as not all

- components can be suppressed; doing so displays the following alert: *At least one feature should not be suppressed.*
- f. Click **Accept** to save your settings.

CONFIGURING MODULES

To go to the Modules page, select the **Configurator** tab, and click the **Modules** tab.



To set the common modules parameters:

1. Select **All Modules** from the list in the left side of the window.
2. Fill in the following fields:
 - **Common version:** Specify the common version, you want to use.
 - **Local location:** Specify the local modules location.
 - **Repository:** Specify the repository.
 - **Default Uri scheme:** Select the default URI scheme, from the drop-down list.
 - **Max fails:** Enter the maximum number of fails.
 - **Minutes between fails:** Enter the amount of minutes between fails.

3. Click **Apply**.

To set a single module parameter:

1. Select the module name from the list in the left side of the window.
2. Select the URI scheme, from the drop-down list.
3. Specify the module version, if you want to use a version different from common for this module.
4. Click **Apply**.

CONTACT US

GET IN TOUCH

We'd love to hear from you. Here's how you can [reach us](#).

SALES CONTACT INFORMATION

Portal: [Contact Altair](#)

US: + 1.800.445.3311

International: + 1.978.441.2200

SUPPORT CONTACT INFORMATION

Customer Portal: <https://community.altair.com/community>

Email: dasupport@altair.com

US: +1 800.988.4739

Canada: +1 978.275.8350

Europe, Middle East, Africa: +44 (0) 8081 892481