Datawatch Corporation makes no representation or warranties with respect to the contents of this manual or the associated software and especially disclaims any implied warranties of merchantability or fitness for any particular purpose. Further, Datawatch Corporation reserves the right to revise this publication and make changes from time to time to its contents without obligation to notify anyone of such revisions or changes.

Datawatch Monarch Swarm software is offered and is to be used in accordance with a SOFTWARE LICENSE AND MAINTENANCE AGREEMENT. This agreement stipulates that this software be used only in the computer system designated in that agreement. The agreement further stipulates that the customer shall not copy or alter, or permit others to copy or alter, the software or related materials in whole or in part, in any media for any purpose, except to make an archive (back-up) copy or to make a copy as an essential step in the use of the software with the customer's computer.

Datawatch Corporation hereby grants the buyer the right to reprint this documentation for internal uses only. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, for any other purposes, without the prior written permission of Datawatch Corporation.

Datawatch Monarch Swarm Installation Guide
Copyright © 2018 by Datawatch Corporation
All rights reserved. Printed in the U.S.A.
Unpublished - Rights reserved under the copyright law of the United States.

Datawatch Monarch Swarm is a trademark of Datawatch Corporation. Other products mentioned herein may be trademarks or registered trademarks of their respective owners in the United States or other countries.

For U.S. Government End Users, the software is a "Commercial Item(s)," as that term is defined at 48 C.F.R. Section 2.101, consisting of "Commercial Computer Software” and “Commercial Computer Software Documentation,” as such terms are used in 48 C.F.R. Section 12.212 or 48 C.F.R. Section 227.7202, as applicable. Consistent with 48 C.F.R. Section 12.212 or 48 C.F.R. Sections 227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the Datawatch Software License and Maintenance Agreement.

DATAWATCH CORPORATION

CORPORATE HEADQUARTERS
4 Crosby Drive
Bedford, MA 01730, USA
Tel.: +1 800.445.3311 / +1 978.441.2200

GREENSBORO, NC
101 S Elm #225
Greensboro, NC 27401
Tel.: +1 800.445.3311

NEW YORK
415 Madison Avenue, Suite 1421
New York, NY 10017
Tel.: +1 800.445.3311

EUROPE, MIDDLE EAST, AND AFRICA

EUROPEAN HEADQUARTERS
Siena Court, Broadway
Maidenhead, Berkshire SL6 1NJ
Tel.: +44 203.868.0230

DATAWATCH AB
Eriksbergsgatan 10
Stockholm, Sweden SE-114 30
Tel: +46 853.480.483

ASIA PACIFIC

MANILA
U2011 20th Flr Jollibee Plaza Condominium
F. Ortigas Jr. Ave., Ortigas Center
Pasig City 1605 PH
Tel.: +63 2.633.5583
# Table of Contents

## Introduction

## System Requirements
- Minimum Requirements
- Recommended Specifications
- Ports
- Known Limitations

## Running the Setup Program
- Post-Installation Steps
  - Increasing the Heap Size of the Application
  - Configuring Tableau Components
- Log File Locations

## Starting Monarch Swarm
- Setting Up the HTTPS Protocol
- Setting Up LDAP SSO

## Configuration File Settings
- Datawatch Monarch Swarm
- Datawatch Monarch Swarm Data Engine
- Datawatch Monarch Swarm ML and Spark

## Upgrading Monarch Swarm
- Upgrading the Monarch Swarm Application
- Upgrading Java Runtime Environment

## Troubleshooting Monarch Swarm
- Uninstalling the Application
- Starting the Service
- Starting the Cassandra Service
- Backing up the Monarch Swarm PostgreSQL Database
- Restoring the Monarch Swarm PostgreSQL Database
Backing up the Cassandra Database ................................................................. 51
  Backing up the Cassandra Database for Monarch Swarm Versions 2.0 and Earlier 52
Restoring the Cassandra Database .................................................................... 53
Product Support ................................................................................................. 54
Introduction

Datawatch Monarch Swarm allows users across organizations and regions to explore, prepare, and distribute data seamlessly and automatically.

The application allows you to:

- Access data from both structured and semi-structured sources, including PDFs, reports, and all major databases, into orderly tables for subsequent business analytics
- Open workspaces saved directly from Data Prep Studio in the Swarm library, a local environment, or shared by other users
- View workspace and table information, including all input data sources, data preparation operations applied, published tables, exports, and schedules, prior to working with it
- Clean and prepare data for visualization or further analysis
- Share data from a single-user environment to clustered server environments to deliver the necessary information to vast communities
- Export data to a number of a number of the most popular visualization and advanced analytics tools
- Capture data preparations in portable data source definitions, workspaces, and processes to enable easy re-use and sharing across users and environments
- Schedule process execution for automatic data distribution
- Enable global collaboration and improve productivity by saving workspaces into the Swarm library for retrieval in Data Prep Studio or sharing workspaces with other Swarm users
- Socialize data use by sharing and liking objects, subscribing to them and other users, and following users; popular items are recommended to other users based on social interactions and usage patterns

This installation guide describes the steps necessary to install and access Monarch Swarm.

More information on Monarch Swarm is available via the following links:

- [Monarch Swarm Factsheet](#)
- [Monarch Swarm Quick Start](#)
- [Monarch Swarm Online Help](#)
System Requirements

To run Monarch Swarm successfully, we suggest that your system meet or exceed the minimum requirements specified below. Note that administration rights and permissions are required to install Monarch Swarm. Running the application once installed, however, does not. Note also that your requirements may vary depending on the volume of data you intend to work with, the number of concurrent users you expect, and several other factors. Contact us to ask about specific requirements for your deployment.

Minimum Requirements

Application Server:

- OS: Windows Server 2016 (recommended), Windows Server 2012 64-bit, or Windows Server 2012 R2 64 bit
- RAM: 32 GB or higher
- Disk Space: 500 GB disk space (depending on the volume of data to be processed)
- CPU: 8 cores or higher
- Java Runtime Environment 8u181 or higher (major version 8, 64-bit)
- For AWS deployments, an m5.2xlarge instance or better

Client Browser (choose one):

- Google Chrome (recommended) – latest version
- Internet Explorer 11
- Mozilla Firefox – latest version
- Microsoft Edge – latest version

Recommended Specifications

Monarch Swarm is a highly scalable and flexible application. We strongly advise contacting your Datawatch account manager to obtain recommended specifications for the deployment you wish to implement.
## Ports

The following ports must be opened for Monarch Swarm to function correctly:

- If you are using HTTP, open port **8080**
- If you are using HTTPS, open port **8443**
- Port **9091** must be opened to allow the socialization and machine-learning components of the application to work correctly. When installing the application on AWS, port 9091, as well as all other ports (i.e., 8080 or 8443, 4040) must be added to your AWS instance security group and the port must be open in the server instance.
- Port **4040** must be opened to allow Spark to work correctly.
- Port **8081** must be opened for proper communication between the Datawatch Monarch Swarm service and the Datawatch Monarch Swarm Data Engine service.

Note that the protocol type for these ports is TCP.

## Known Limitations

The current version of Monarch Swarm includes, among others, the following limitations:

- If user selects the "Row Number" metadata column in a table, the column operations are not displayed.
- Some file types, such as .log and .xps, are not supported in Monarch Swarm v2.2.
- Workspaces imported from Data Prep Studio and containing data from CDATA drivers are not supported in the current version of Monarch Swarm.
- The options for a Google Analytics data source (e.g., Account, Views, Properties) made from one Google Analytics connection do not display when this connection is changed to another Google Analytics connection.
- When a table in a workspace is published and then shared with the setting "Share all required input data sources,” a share notification for only the table is sent to users with whom the table and data sources are shared.
- No support for SAS files.
- While Microsoft Excel files in the 97–2003 workbook format are supported in Monarch Swarm, the application does not support Excel 95 .XLS files. To continue using the data in a 95 .XLS file in Monarch Swarm, open the file in Excel and then save it to a supported format.
- Data Prep Studio allows more encoding options than are currently supported in Swarm. By default, Swarm will treat ANSI/ASCII encodings as UTF-8. UTF-16 and UTF-32 are also supported.
- When tables with over 255 columns or very large numbers of rows are exported from Data Prep Studio to Monarch Swarm, the export result in Data Prep Studio may read as 100% completed but Monarch Swarm yields an empty table except for column headers.
This error is logged in the app.log file as follows.

```
com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
at com.datastax.driver.core.exceptions.InvaliRequestException: Batch too large
```

A list of other known limitations may be found [here](#).
Running the Setup Program

The steps below describe how to install Monarch Swarm in a Windows environment.

Steps:

1. Manually download the Windows 64-bit version of Java SE Runtime Environment 8 and install it.
   Note that while the minimum version required by Monarch Swarm is update 181, newer updates are also supported by the installer so long as the major version is 8.

2. Extract the contents of the zip file you obtained from the download link provided by your product specialist. This file should contain:
   - The application installer
   - A prerequisites folder
   The prerequisites folder should include installers for:
   - Datawatch JDBC Drivers
   - PostgreSQL 9.6
   - A libs folder containing CDATA JDBC drivers to enable the creation of connections
   If any of these items is not in the file you downloaded, contact your Datawatch product specialist immediately.

3. Your license will be provided to you by Datawatch. Rename this file to license.lic and copy it to the folder containing the Monarch Swarm prerequisites folder and installer.

4. Double click on the application installer (DatawatchMonarchSwarm.exe) to run it.

5. In the next screen, read the license agreement. If you agree with its terms, select I accept the agreement and then click Next.
6. The *Configuration* screen allows you to specify whether the HTTP or HTTPS protocol should be used. The default port for each protocol is also indicated. Choose the protocol you would like to apply. Unless you want to change them, accept the default port values indicated and then click **Next**.

![Configuration Screen](image)

If you wish to use a different set of ports, enter them into the appropriate fields provided and then click **Next**. Note that the ports you specify must not be used by other applications.

7. The *Database Configuration* screen allows you to specify a database name, a database user name, and a database user password. The default for all three items is `newserver`. If you wish to create a new database, user, and password, enter the required names and passwords into the corresponding fields.

The *Database Configuration* screen requires you to specify a PostgreSQL administrator login and password to use for new PostgreSQL installations.
Take note of the administrator login and password you supply as these details will be required when you upgrade the Monarch Swarm application. If preexisting installation of PostgreSQL is available, the administrator login and password should be provided here. Enter the necessary details and then click **Next**.

8. The *Select Destination Location* screen allows you to specify a folder in which the application will be installed. If you do not wish to change the default location indicated, click **Next**. Otherwise, specify a new location and then click **Next**.

![Select Destination Location](image)

9. The *Select Components* screen allows you to specify which Monarch Swarm components to install.

![Select Components](image)

The Tableau SDK and Tableau Hyper components are provided to enable table download/export to Tableau Server in .tde format and Tableau .hyper format, respectively.
Note that only one of these components can be installed at any one time. Deselect the component(s) you do not wish to install and then click Next.

10. The Select Start Menu Folder screen allows you to specify a Start Menu folder into which the program’s shortcuts will be placed. Change it to a different folder if you wish and then click Next or simply click Next to accept the default value indicated.

11. The Ready to Install screen provides a summary of your installation folders. Click Install to begin installing Monarch Swarm.

The installation begins.
12. When installation has completed, the following message displays. Click **Finish**.

When the application is correctly installed, the following services become available:

- Datawatch Monarch Swarm
- Datawatch Monarch Swarm Data Engine
- Datawatch Monarch Swarm ML
- Datawatch Monarch Swarm Spark

Note also that while Postgres and Cassandra are automatically installed when Monarch Swarm is installed, these applications are reserved for Monarch Swarm use only. Thus, addition of custom tables/data is not supported.
Post-Installation Steps

INCREASING THE HEAP SIZE OF THE APPLICATION

The following procedure represents an important and mandatory post-installation configuration step to increase the heap size of the application and maximize performance. These instructions should be performed by an experienced system administrator.

Steps:

1. Calculate the maximum heap size to allocate to the application. While requirements may vary according to the environment in which the application is being deployed, a heap size of 75% of the physical memory of the server is recommended.

   For example, if the physical memory of the server is 64 GB, the maximum heap size should be 48 GB.

2. Modify **monarchswarm.xml** and **monarchswarm-dp.xml** as follows, replacing the numerical value of the parameter "-Xmx16g" with the value calculated in Step 1:

   ```
   -Xms4g -Xmx16g
   ```

   For example:

   ```
   <arguments>-jar "core-api.jar" -Xms4g -Xmx16g --spring.profiles.active=prod</arguments>
   ```

   where:

   "-Xms4g” and "-Xmx16g” indicate the minimum and maximum heap sizes, respectively.

CONFIGURING TABLEAU COMPONENTS

When installing Monarch Swarm, you are asked to select whether to install the Tableau SDK (TDE) or Tableau Hyper component. These components cannot work at the same time, and one file format must be enabled while the other is disabled. Once installed, no further configuration is necessary, and you can download/export tables according to your chosen component (e.g., Tableau SDK).

If, however, you wish to switch to the other component (e.g., Tableau Hyper), you must run the installer once more and select this component to install it. Once done, the environment variables are modified.
Steps:

1. In Windows, launch the System Properties dialog (typically by right-clicking on **Computer > Properties > Advanced System Settings**) and then click **Environment Variables**.

   The Environment Variables dialog displays.

   ![Environment Variables Dialog](image)

2. Edit the **Path** property in the **System variables** section of the **Environment Variables** dialog.

3. From the list of system variables, delete `C:\Program Files\Datawatch Monarch Swarm\tableau_hyper\bin` and leave `C:\Program Files\Datawatch Monarch Swarm\tableau_sdk\bin` (if you previously installed Tableau Hyper and now want to use Tableau SDK).

   If you previously installed Tableau SDK and now want to use Tableau Hyper, delete `C:\Program Files\Datawatch Monarch Swarm\tableau_sdk\bin` and leave `C:\Program Files\Datawatch Monarch Swarm\tableau_hyper\bin`.

4. Save changes and restart Windows.
Log File Locations

Monarch Swarm log files may be found in the following locations:

<table>
<thead>
<tr>
<th>LOG FILE</th>
<th>LOCATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cassandra</td>
<td><code>&lt;Monarch Swarm folder&gt;\cassandra\logs</code></td>
</tr>
<tr>
<td>Datawatch Monarch Swarm service (core-api.log)</td>
<td><code>%TEMP%\MonarchSwarm\Logs\</code></td>
</tr>
<tr>
<td>Datawatch Monarch Swarm Data Engine service (data-engine-api.log)</td>
<td>e.g., C:\Windows\Temp\MonarchSwarm\Logs</td>
</tr>
<tr>
<td>Datawatch Monarch Swarm ML and Spark services (ml-app.log)</td>
<td></td>
</tr>
<tr>
<td>Installation log file</td>
<td>The installation log file is saved in the</td>
</tr>
<tr>
<td></td>
<td>user’s Temp folder (i.e., the person who ran</td>
</tr>
<tr>
<td></td>
<td>the installer)</td>
</tr>
<tr>
<td></td>
<td>e.g., C:\Users&lt;User’s name&gt;\AppData\Local\Temp</td>
</tr>
<tr>
<td></td>
<td>Installation log files are named as follows:</td>
</tr>
<tr>
<td></td>
<td>Setup Log YYYY-MM-DD #NNN.next</td>
</tr>
<tr>
<td></td>
<td>where:</td>
</tr>
<tr>
<td></td>
<td>YYYY – current year</td>
</tr>
<tr>
<td></td>
<td>MM -current month</td>
</tr>
<tr>
<td></td>
<td>DD – current day</td>
</tr>
<tr>
<td></td>
<td>NNN – a unique number provided to each</td>
</tr>
<tr>
<td></td>
<td>installation starting from 1</td>
</tr>
</tbody>
</table>
Starting Monarch Swarm

If you opt not to restart your computer immediately after installing the application, the Monarch Swarm login page displays as soon as you click Finish on the final screen of the installer.

If you opted to restart your computer after installation or if you are accessing Monarch Swarm from a different machine, launch your browser, enter the IP address of the machine you are connecting to into the address bar (i.e., "10.0.0.101"), and then append the application port specified during installation (in this case ":8080") to it (e.g., "10.0.0.101:8080"). Press Enter on your keyboard afterward.
The Monarch Swarm login page displays.

If you selected the HTTPS protocol, your address bar should display as follows:
You may need to modify the firewall settings of your server if an error page, such as that shown below, displays when trying to access the Monarch Swarm login page. Contact your network administrator if this problem persists.

![Error Page](image)

Login using the credentials provided to you by your product specialist and then click **Sign in**.

The **Monarch Swarm dashboard** displays.

## Setting Up the HTTPS Protocol

An SSL certificate is required to run Monarch Swarm installations using the HTTPS protocol.

*Note that this procedure is best performed by a knowledgeable system administrator.*

In the steps outlined below, the keystore file **key-store.jks** and the security certificate **key-store.cer** are generated. You can generate the keystore and certificate using any name you wish.

**Steps:**

1. Generate a security certificate.

   1.1 Go to the bin directory of JRE (i.e., the path where Java is installed, usually `C:\Program Files\Java\jre1.8.x_x\bin`), paste the following script and run it. In this example, replace "localhost" with your domain name in `CN=localhost`. Replace “localhost” with the machine name in `san=dns:localhost`. Alternatively, change the value in `ip` to the IP of the machine.
keytool -genkey -alias server -keyalg RSA -keysize 4096 -keystore key-store.jks -validity 3652 -dname "CN=localhost, O=Datawatch Corporation, L=Bedford, ST=MA, C=US" -ext san=dns=localhost,ip:127.0.0.1 & keytool -certreq -alias server -file key-store.csr -keystore key-store.jks

1.2 You will be asked to set a **keystore password** and a **key password**. These passwords should be the same. You will be asked to repeat your keystore and key passwords at a later time so take note of this information.

1.3 Run the following command to export the certificate:

```
keytool -export -alias server -keystore key-store.jks -rfc -file key-store.cer
```

A security certificate is generated in C:\Program Files\Java\jre1.8.0_XXX\bin."

1.4 Copy the keystore file (*.jks) to the **certificate** folder of the Swarm directory (C:\Program Files\Datawatch Monarch Swarm\certificate).

2. Configure Monarch Swarm.

2.1 Edit the Monarch Swarm configuration file.

2.1.1 Open **application-prod.yml**. This file is normally located in C:\Program Files\Datawatch Monarch Swarm.

2.1.2 Set the following parameters:

```
server:
  port: 8443
  ssl:
    key-store: key-store.jks # Path to generated keystore
    key-store-password: Pa$$word # Keystore password
    key-password: Pa$$word # Key password
    enabled: true

application:
  data-engine-api:
    url: https://machine_name:8081
```

Parameter settings:

- server.port – The port to use for the HTTPS protocol (8443)
- server.ssl.key-store – The path to the generated keystore
- server.ssl.key-store-password – The keystore password
- server.ssl.key-password – The key password
- server.ssl.enabled – Set this item to **true** to enable SSL

2.2 Edit the Monarch Swarm ML configuration file.
2.2.1 Open `social-application.yml`. This file is normally located in C:\Program Files\Datawatch Monarch Swarm\ml.

2.2.2 Set the following parameters:

```yaml
server:
  ssl:
    enabled: true
    keystore.password: Pa$$word # Keystore password
    keystore.path: ..\\certificate\key-store.jks # Path to the generated keystore
```

Parameter settings:

- **server.ssl.enabled** – Set this item to `true` to enable SSL
- **server.ssl.keystore.password** – The keystore password
- **server.ssl.keystore.path** – The path to the generated keystore

2.3 Edit the Monarch Swarm Data Engine configuration file.

2.3.1 Open `application-prod.yml` in the dp folder (C:\Program Files\Datawatch Monarch Swarm\dp).

2.3.2 Set the following parameters:

```yaml
server:
  ssl:
    enabled: true
    key-store: ..\\certificate\key-store.jks
    key-store-password: Pa$$word # Keystore password
    key-password: Pa$$word # Key password

application:
  core-api:
    url: https://machine_name:8443
```

Parameter settings:

- **server.ssl.enabled** – Set this item to `true` to enable SSL
- **server.ssl.key-store** – The path to the generated keystore
- **server.ssl.key-store-password** – The keystore password
- **server.ssl.key-password** – The key password
2.4 Import the generated certificate to the Java keystore.

2.4.1 Open Command Prompt with administrator rights, paste the following script, and then run it.

```bash
keytool -import -noprompt -v -trustcacerts -alias datawatch_monarch_swarm_certificate -file "C:\Program Files\Datawatch Monarch Swarm\certificate\key-store.cer" -keystore "C:\Program Files\Java\jre1.8.x_xx\lib\security\cacerts" -storepass changeit
```

Note that the path to the Java keystore is usually `C:\Program Files\Java\jre1.8.x_xx\lib\security\cacerts` and the path to the certificate file is usually `C:\Program Files\Datawatch Monarch Swarm\certificate\key-store.cer`.

2.3.2 If the certificate is renewed, delete the previously imported certificate from the Java keystore using the script below and then import the new certificate as described above.

```bash
keytool -delete -alias datawatch_monarch_swarm_certificate -keystore "C:\Program Files\Java\jre1.8.x_xx\lib\security\cacerts" -storepass changeit
```

2.4 Restart the Datawatch Monarch Swarm, Datawatch Monarch Swarm Data Engine, Datawatch Monarch Swarm Spark, and Datawatch Monarch Swarm ML services.

3. Configure clients.

3.1 Add the generated certificate to Trusted Root on all machines on which Monarch Swarm will be used.

3.1.1 Run Certificate Manager.

3.1.2 Import the file key-store.cer to Trusted Root Certification Authorities.
3.2 Launch Monarch Swarm in your browser using the following form:
https://<ipaddress>:8443.

Setting Up LDAP SSO

LDAP Single Sign-On (SSO) can be set up to log into Monarch Swarm to implement
organizational security and allow larger number of users to access the application without
needing to create a user profile for each one.

Note that this procedure is best performed by a knowledgeable system
administrator.

The steps outlined in this section assume that:

- There exists a domain controller server in which LDAP has been enabled and configured.
- There exists a server in which Monarch Swarm has been installed. This server must be
  part of the domain.
- There exists a user with a client machine who is a member of the domain for which
  Monarch Swarm is being configured

Note that the scripts to be run and config file are extremely sensitive to formatting and case.
Exercise caution when running the necessary scripts and making additions to the Monarch
Swarm config file.

Steps:

A. On the server with Active Directory Domain Controller:

1. In the machine with the AD domain controller, create the AD user tomcat with the password
   Password#.

2. Run the following scripts in Powershell in Administrator mode:

   * setspn -A HTTP/<full computer name of Monarch Swarm server>
     tomat
   * ktpass /out c:\tomcat.keytab /mapuser tomat@<DOMAIN NAME>
     /princ HTTP/<full computer name of Monarch Swarm server>@<domain
     name> /pass Password# /ptype KRB5_NT_PRINCIPAL /crypto All

For example, assuming that WIN-SWARMSERVER.test.local is the complete computer
name of the Monarch Swarm server and the domain name is test.local, the following
scripts should be run:

   * setspn -A HTTP/WIN-SWARMSERVER.test.local tomat
   * ktpass /out c:\tomcat.keytab /mapuser tomat@TEST.LOCAL
     /princ HTTP/WIN-SWARMSERVER.test.local@TEST.LOCAL /pass Password#
     /ptype KRB5_NT_PRINCIPAL /crypto All
B. On the Monarch Swarm server:

1. Stop all of the related Monarch Swarm services.

2. Copy the file `tomcat.keytab` generated in Step A2 and place it in `C:/Users/tomcat`.

3. Modify application properties by opening the file `application-prod.yml`, which is typically located in `C:/Program Files/Datawatch Monarch Swarm`, and then add the following structure to the `application.security` section:

   ```yaml
   authentication:
     provider: ldap
     users-provisioned: true
     default-password: password
   ldap:
     enabled: true
     sso:
       enabled: true
       service-principal: HTTP/<full computer name of Monarch Swarm server>/@<domain name>
       key-tab-location: C:/Users/tomcat/tomcat.keytab
       request-regex: ^/api/.*ldap_sso
   query:
     attribute-mapping:
       login: userPrincipalName
       first-name: givenname
       last-name: sn
       common-name: cn
       email: mail
       phone-numer: telephonenumber
       groups: memberOf
   custom-attributes:
     - displayName
     - distinguishedName
     - name
     - objectCategory
     - objectClass
     - primaryGroupId
     - sAMAccountName
     - SAMAccountType
     - servicePrincipalName
   active-directory: true
   domain: <DOMAIN NAME>
   server: <full computer name of domain controller server>/
   manage-dn: user_name@<domain>
   manage-password: <password>
   search-base: DC=<dc>, DC=<dc>
   search-filter: "(| (userPrincipalName={0}) (sAMAccountName={0}))"
   ```
user-roles:
- <role>
admin-users:
- <list of users with admin privileges in Monarch Swarm>

The table below describes, in detail, the parameters that must be added to the configuration file to configure SSO.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>application.security.authentication.provider</td>
<td>Use ldap for SSO</td>
</tr>
<tr>
<td>application.security.authentication.users-provisioned</td>
<td>Enables (true) or disables (false) explicit provisioning. If explicit provisionning is disabled, the system creates Monarch Swarm users automatically. When set to true, users must be created manually</td>
</tr>
<tr>
<td>application.security.authentication.default-password</td>
<td>The default password for new users created through LDAP and added multiple users</td>
</tr>
<tr>
<td>application.security.authentication.sso.enabled</td>
<td>true to enable SSO</td>
</tr>
<tr>
<td>application.security.authentication.sso.service-principal</td>
<td>Full computer name of the Monarch Swarm server</td>
</tr>
<tr>
<td>application.security.authentication.sso.key-tab-location</td>
<td>Path to keytab file</td>
</tr>
<tr>
<td>application.security.authentication.sso.request-regex</td>
<td>Setting for SSO</td>
</tr>
<tr>
<td>application.security.authentication.sso.query attribute-mapping:</td>
<td>Settings used to add users by LDAP query</td>
</tr>
<tr>
<td>login: userPrincipalName</td>
<td></td>
</tr>
<tr>
<td>first-name: givenname</td>
<td></td>
</tr>
<tr>
<td>last-name: sn</td>
<td></td>
</tr>
<tr>
<td>common-name: cn</td>
<td></td>
</tr>
<tr>
<td>email: mail</td>
<td></td>
</tr>
<tr>
<td>phone-number: telephonenumber</td>
<td></td>
</tr>
<tr>
<td>groups: memberOf</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.sso.query custom-attributes:</td>
<td></td>
</tr>
<tr>
<td>- displayName</td>
<td></td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>--------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>- distinguishedName</td>
<td></td>
</tr>
<tr>
<td>- name</td>
<td></td>
</tr>
<tr>
<td>- objectCategory</td>
<td></td>
</tr>
<tr>
<td>- objectClass</td>
<td></td>
</tr>
<tr>
<td>- primaryGroupID</td>
<td></td>
</tr>
<tr>
<td>- sAMAccountName</td>
<td></td>
</tr>
<tr>
<td>- sAMAccountType</td>
<td></td>
</tr>
<tr>
<td>- servicePrincipalName</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.ldap.active-directory</td>
<td>true when AD is used</td>
</tr>
<tr>
<td>application.security.authentication.ldap.domain</td>
<td>Domain name</td>
</tr>
<tr>
<td>application.security.authentication.ldap.server</td>
<td>Full computer name of LDAP server</td>
</tr>
<tr>
<td>application.security.authentication.ldap.manage-dn</td>
<td>User name and password to use to connect to LDAP server when sso.enabled=false and ldap-active-directory=false</td>
</tr>
<tr>
<td>application.security.authentication.ldap.manage-password</td>
<td>If sso.enabled=true and ldap-active-directory=true, these parameters may be omitted from the config file</td>
</tr>
<tr>
<td></td>
<td>These credentials are also used to add multiple users to Monarch Swarm using LDAP query</td>
</tr>
<tr>
<td>application.security.authentication.ldap.search-base</td>
<td>Domain name (e.g., DC=test,DC=local if domain is test.local)</td>
</tr>
<tr>
<td>application.security.authentication.ldap.search-filter</td>
<td>LDAP configuration</td>
</tr>
<tr>
<td>application.security.authentication.ldap.user-roles</td>
<td>User role for SSO users</td>
</tr>
<tr>
<td>application.security.authentication.ldap.admin-users</td>
<td>List of users automatically created with admin privileges in Monarch Swarm (of users-provisioned: false). When this list is provided, there is no need to login as an administrator and create the first LDAP user.</td>
</tr>
</tbody>
</table>

For example, using the domain and Monarch Swarm server names specified above and assuming that `WIN2K8R2DC.test.local` is the computer name of the domain controller, the following strings should be added:

```yaml
authentication:
  provider: ldap
  users-provisioned: true
```
default-password: password
ldap:
  sso:
    enabled: true
    service-principal: HTTP/WIN-SWARMSERVER.test.local/@TEST.LOCAL
  key-tab-location: C:/Users/tomcat/tomcat.keytab
  request-regex: ^/api/.+/ldap_sso
query:
  attribute-mapping:
    login: userPrincipalName
    first-name: givenname
    last-name: sn
    common-name: cn
    email: mail
    phone-number: telephonenumber
  groups: memberOf
  custom-attributes:
    - displayName
    - distinguishedName
    - name
    - objectCategory
    - objectClass
    - primaryGroupID
    - sAMAccountName
    - SAMAccountType
    - servicePrincipalName
active-directory: true
domain: TEST.LOCAL
server: ldap://WIN2K8R2DC.test.local/
search-base: DC=test,DC=local
search-filter: "(| (userPrincipalName={0}) (sAMAccountName={0}))"
user-roles:
  - 3
admin-users:
  - new_user1@test.local
  - new_user2@test.local

Note that the item "user-roles: - 3" (this item MUST span two lines) indicates that
the created user is assigned the role ANALYST. You must indicate a role for users in the
cfg file as you cannot do so in the application. You can change the user role by
changing the number indicated to any of the following:

- 1 – ADMIN
2 – CONSUMER
3 – ANALYST
4 – CURATOR
5 – DESKTOP
6 – ADVANCED
7 – DATA SCIENTIST

You can also assign the user multiple roles as in the following:

```yaml
  user-roles:
    - 3
    - 4
```

This code assigns the user the roles ANALYST and CURATOR. Note that, in this case, this snippet must span 3 lines.

More information on user roles may be found [here](#).

In this example, two users (i.e., new_user1 and new_user2) are assigned ADMIN roles.

4. Restart the application using the AD user `tomcat`.

C. On the client machine:

1. Log in using the AD username and password of a user who is a member of the domain.

2. If necessary, configure your browser to accept and trust domain sites (i.e., add to local intranet and trusted sites; Google Chrome does not require this).

3. Launch the Monarch Swarm login page using the form `https://<full name of Monarch Swarm server>:8443` (in the present example, we would enter `https://win-swarmserver.test.local:8443` into the address bar of our browser) and then click the SSO Sign In button.
4. Enter the username and password of your user and then click the **Log in** button.

Once logged in, the Monarch Swarm dashboard displays:
A look at the user’s profile should display as follows:

![User Profile Image]

Future sessions using the same client machine will no longer require the user to log in with his/her credentials.
Configuration File Settings

Datawatch Monarch Swarm

Monarch Swarm’s configuration file, named `application-prod.yml`, is typically created and stored in C:\Program Files\Datawatch Monarch Swarm. A snippet of the contents of this file is shown below.

```yaml
spring:
data:
cassandra:
enabled: true
dataSource:
  url: jdbc:postgresql://localhost:1000/newserver
  username: newserver
  password: newserver
http:
  multipart:
    maxSize: 2000MB
    maxRequestSize: 2000MB
server:
  port: 8080
ssl:
enabled: false
  key-store: certificate\localhost.jar
  key-store-password: password
  key-password: password
tomcat:
  accessLog:
    enabled: true
directory: c:/logs
    accept-count: 100 # Maximum queue length for incoming connection requests when all possible request processing threads are in use.
    buffer-size: 512 # Buffer output such that it is only flushed periodically.
    pattern: common # Format pattern for access logs (can be combined).
    prefix: access_log # Log file name prefix.
    rename-on-rotate: false # Make inclusion of the date stamp in the file name until rotate time.
    request-attributes-enabled: false # Set request attributes for IP address, Hostname, protocol and port used for the request.
    rotate: true # Enable access log rotation.
    suffix: .log # Log file name suffix.
application:
  server:
    internet-address: http://10.0.0.100:8080
http:
  cache: # Used by the CachingHttpHeadersFilter
timeToLiveInDays: 31
data-engine:
  store:
    design-mode-limit: 400
  suggestion:
    prepare-cron: 0 0 1 * * # config for schedule
```
Upon installation of the Monarch Swarm application, your configuration file will only include settings for writing to the Cassandra database, the connection to the Monarch Swarm database, the server port, and the SSL certificate (if this option is selected during installation).

The following table describes, in detail, the parameters that may be added to this configuration file.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SPRING</strong></td>
<td></td>
</tr>
<tr>
<td>spring.data.cassandra.enabled</td>
<td>Accepts the values <strong>true</strong> or <strong>false</strong></td>
</tr>
<tr>
<td></td>
<td>Enables (when true) or disables (when false) exports to the Library</td>
</tr>
<tr>
<td></td>
<td>and exports of pinned data</td>
</tr>
<tr>
<td>spring.datasource.url</td>
<td>Describes the connection to the Postgres database for the Datawatch Monarch</td>
</tr>
<tr>
<td>spring.datasource.url.jdbc</td>
<td>Swarm service</td>
</tr>
<tr>
<td>spring.datasource.url.username</td>
<td></td>
</tr>
<tr>
<td>spring.datasource.url.password</td>
<td></td>
</tr>
<tr>
<td>spring.http.multipart.maxFileSize</td>
<td>Describes the maximum size of files that may be uploaded to the application</td>
</tr>
<tr>
<td>spring.http.multipart.maxRequestSize</td>
<td></td>
</tr>
<tr>
<td><strong>SERVER</strong></td>
<td></td>
</tr>
<tr>
<td>server.port</td>
<td>Port on which the application is running</td>
</tr>
<tr>
<td>server.port.ssl.enabled</td>
<td><strong>true</strong> if HTTPS is enabled</td>
</tr>
<tr>
<td>server.port.ssl.key-store</td>
<td>Describe parameters for the SSL certificate</td>
</tr>
<tr>
<td>server.port.ssl.key-store-password</td>
<td></td>
</tr>
<tr>
<td>server.port.ssl.key-store-password-password</td>
<td></td>
</tr>
<tr>
<td>server.tomcat.accesslog.enabled</td>
<td>These items describe settings for Tomcat logs:</td>
</tr>
<tr>
<td>server.tomcat.accesslog.directory</td>
<td>• <strong>buffered</strong> - Buffer output so that it is flushed periodically</td>
</tr>
<tr>
<td>server.tomcat.accesslog.accept-count</td>
<td>• <strong>pattern</strong> - Format pattern for access logs</td>
</tr>
<tr>
<td>server.tomcat.accesslog.buffered</td>
<td>• <strong>prefix</strong> - Log filename prefix</td>
</tr>
<tr>
<td>server.tomcat.accesslog.pattern</td>
<td>• <strong>rename-on-rotate</strong> – Defer inclusion of the date stamp in the filename</td>
</tr>
<tr>
<td>server.tomcat.accesslog.prefix</td>
<td>until rotate time</td>
</tr>
<tr>
<td>server.tomcat.accesslog.rename-on-rotate</td>
<td>• <strong>request-attributes-enabled</strong> – Set request attributes for IP</td>
</tr>
<tr>
<td>server.tomcat.accesslog.request-attributes-enabled</td>
<td>address, hostname, protocol, and port used for the request</td>
</tr>
<tr>
<td>server.tomcat.accesslog.rotate</td>
<td>• <strong>rotate</strong> – enable access log rotation</td>
</tr>
<tr>
<td>server.tomcat.accesslog.suffix</td>
<td>• <strong>suffix</strong> – Log filename suffix</td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>application.server.internet-address</td>
<td>Describes the redirect URL for login to Salesforce, Google Analytics, and Google Adwords (should be identical to the URL specified for ClientId and ClientSecret for these connections).</td>
</tr>
<tr>
<td>application.http.cache.timeToLiveInDays</td>
<td>Describes the amount of time in days that may elapse before a data source’s cache times out</td>
</tr>
<tr>
<td>application.data-engine.store.design-mode-limit</td>
<td>Describes the row limit to be used for data sources in Design Mode; the default value is 10K</td>
</tr>
<tr>
<td>application.data-engine.suggestion.prepare-cron</td>
<td>Describes settings for jobs that calculate suggestions based on data type and content</td>
</tr>
<tr>
<td>application.data-engine-api.url</td>
<td>URL for internal communication with Datawatch Monarch Swarm Data Engine service (http://&lt;machine name&gt;:8081)</td>
</tr>
<tr>
<td>application.dsl.source-cleaner-cron</td>
<td></td>
</tr>
<tr>
<td>application.dsl.source-expiration-in-hours</td>
<td>Describes settings for jobs that delete temporary objects</td>
</tr>
<tr>
<td>application.dsl.temporary-item-cleaner-cron</td>
<td></td>
</tr>
<tr>
<td>application.dsl.temporary-item-expiration-in-hours</td>
<td></td>
</tr>
<tr>
<td>application.license.filepath</td>
<td>Describes the path to the Monarch Swarm license</td>
</tr>
<tr>
<td>application.io.connection.disabled</td>
<td>Specifies which connection types to disable (hide) in Monarch Swarm for all users</td>
</tr>
<tr>
<td>application.io.appDataFolder</td>
<td>Describes the path to the application’s internal storage (i.e., File Library)</td>
</tr>
<tr>
<td>application.io.writer.watson.http-client.timeout</td>
<td>300 - Describes the time in seconds that may elapse before connections to IBM Watson Analytics time out</td>
</tr>
<tr>
<td>application.io.writer.cognos.http-client.timeout</td>
<td>600 - Describes the time in seconds that may elapse before connections to IBM Cognos Analytics time out</td>
</tr>
<tr>
<td>application.security.authentication.xauth.token</td>
<td>1,800 - Describes how many seconds should elapse before a user times out</td>
</tr>
<tr>
<td>application.security.authentication.xauth.token ValidityInSeconds</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.sso.enabled</td>
<td>Settings for LDAP SSO</td>
</tr>
<tr>
<td>application.security.authentication.sso.ad-domain</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.sso.ad-server</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.sso.service-principal</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.sso.key-tab-location</td>
<td></td>
</tr>
<tr>
<td>PARAMETER</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>application.security.authentication.sso.ldap.search-base</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.sso.ldap.search-filter</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.sso.request-regex</td>
<td></td>
</tr>
<tr>
<td>application.security.authentication.sso.user-roles</td>
<td></td>
</tr>
<tr>
<td>application.schedules.monitoring.intervalInMinutes</td>
<td>Number of minutes that must elapse before the next monitoring operation should be executed in a monitoring schedule</td>
</tr>
</tbody>
</table>

Datawatch Monarch Swarm Data Engine

Settings for the Monarch Swarm Data Engine service are specified in C:\Program Files\Datawatch Monarch Swarm\dp\application-prod.yml.

A snippet of the config file is provided below:
spring:
data:
cassandra:
  enabled: true
datasource:
  url: jdbc:postgresql://localhost:5433/dataengine
logging:
  file: logs/app.log
logback:
  loglevel: INFO
server:
  port: 8881
  ssl:
    enabled: false
application:
data-engine:
  suggestion:
    rank-threshold: 0.2 # default ranking thresholds, min value
  store:
    export-data-async-timeout-in-sec: 3600
    design-mode-limit: 10000
    globalRowLimit: 1000000
    column-limit: 100
    distinct-value-limit: 250
io:
  writer:
    watson:
      http-client:
        timeout: 300
    cognos:
      http-client:
        timeout: 600
  reader:
    preview:
      limit: 1000
The following table describes, in detail, the parameters that may be added to this configuration file.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPRING</td>
<td></td>
</tr>
<tr>
<td>spring.data.cassandra.enabled</td>
<td>Accepts the values true or false. Enables (when true) or disables (when false) exports to the Library and exports of pinned data.</td>
</tr>
<tr>
<td>spring.datasource.url</td>
<td>Describes the connection to the Postgres database for the Datawatch Monarch Swarm service.</td>
</tr>
<tr>
<td>LOGGING</td>
<td></td>
</tr>
<tr>
<td>logging.file</td>
<td>full path to Data Engine service log file.</td>
</tr>
<tr>
<td>LOGBACK</td>
<td></td>
</tr>
<tr>
<td>logback.loglevel</td>
<td>Logging level of the Data Engine service log file.</td>
</tr>
<tr>
<td>SERVER</td>
<td></td>
</tr>
<tr>
<td>server.port</td>
<td>8081 – port on which the Data Engine is running.</td>
</tr>
<tr>
<td>server.port.ssl.enabled</td>
<td>true if HTTPS is used.</td>
</tr>
<tr>
<td>server.port.ssl.key-store</td>
<td></td>
</tr>
<tr>
<td>server.port.ssl.key-store-password</td>
<td></td>
</tr>
<tr>
<td>server.port.ssl.key-password</td>
<td></td>
</tr>
<tr>
<td>APPLICATION</td>
<td></td>
</tr>
<tr>
<td>application.data-engine.suggestion.rank-threshold</td>
<td>Describes settings for suggestions based on data type and content; shows minimum rank for retrieving and sorting suggestions.</td>
</tr>
<tr>
<td>application.data-engine.store.design-mode-limit</td>
<td>Describes the row limit to be used for data sources in Design Mode; the default value is 100K.</td>
</tr>
<tr>
<td>application.data-engine.store.globalRowLimit</td>
<td>Row limit applied when the Design Mode limit is disabled.</td>
</tr>
<tr>
<td>application.data-engine.store.column-limit</td>
<td>100 - column limit after Pivot and Transpose.</td>
</tr>
<tr>
<td>application.data-engine.store.distinct-value-limit</td>
<td>250 - number of displayed distinct values limit.</td>
</tr>
</tbody>
</table>
### PARAMETER | DESCRIPTION
--- | ---
application.server.internet-address | redirect url for login to Watson, Salesforse, Google Analytics (should be as redirect url is specified for ClientId and ClientSecret for these connections).

**application.security.authentication.xauth.secret** | security token, should be equal to all other security tokens in all other application config files

**application.core-api.url** | address for internal communication with Datawatch Monarch Swarm service

### READER

**reader.preview.limit** | 1,000 – row limit for previewing data sources

---

**Datawatch Monarch Swarm ML and Spark**

The config file for the Datawatch Monarch Swarm ML and Spark services is typically installed in C:\Program Files\Datawatch Monarch Swarm\ml\social-application.yml.

```yaml
server:
  jwt:
    secret: secret as in all other configs
  ssl:
    enabled: false
    keystore.password: secret
    keystore.path: ..\\certificate\localhost.jks
spark:
  app:
    name: Dw Swarm Decision Engine
    schedule: 0 0 0 1/1 * *
    scheduling:
      enabled: true
    driver:
      memory: 4g
database:
  cassandra:
    connection:
      attempts:
        amount: 0
        wait:
          time:
            seconds: 30
logging:
  loglevel: INFO
  logfilepath: ${TEMP}/MonarchSwarm/logs/ml-app.log
```
This table describes, in detail, the parameters that may be added to the configuration file.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SERVER</strong></td>
<td></td>
</tr>
<tr>
<td>server.jwt.secret</td>
<td>A security token that should be identical to all other security token indicated in all other application config files (e.g., application.security.authentication.xauth.secret in other config files)</td>
</tr>
<tr>
<td>server.port.ssl.enabled</td>
<td><strong>true</strong> if HTTPS is used.</td>
</tr>
<tr>
<td>server.port.ssl.keystore-password</td>
<td>Describes parameters for the SSL certificate.</td>
</tr>
<tr>
<td>server.port.ssl.keystore.path</td>
<td></td>
</tr>
<tr>
<td><strong>SPARK</strong></td>
<td></td>
</tr>
<tr>
<td>spark.app.name</td>
<td>Spark application name</td>
</tr>
<tr>
<td>spark.app.schedule</td>
<td>Cron for Spark job for calculating suggestions (e.g., 0 0/20 * 1/1 * ? *)</td>
</tr>
<tr>
<td>spark.app.scheduleing.enabled</td>
<td><strong>False</strong> if only a single run is applied</td>
</tr>
<tr>
<td><strong>DATABASE</strong></td>
<td></td>
</tr>
<tr>
<td>database.cassandra.connection.attempts.amount</td>
<td>Number of attempts to connect to the Cassandra database</td>
</tr>
<tr>
<td>database.cassandra.connection.wait.time.seconds</td>
<td>Number of second for each attempt to connect to the Cassandra database</td>
</tr>
<tr>
<td><strong>LOGGING</strong></td>
<td></td>
</tr>
<tr>
<td>logging.loglevel</td>
<td>Logging level</td>
</tr>
<tr>
<td>logging.logfilepath</td>
<td>Full path to the ML and Spark services log file (e.g., C:\Windows\Temp\MonarchSwarm\Logs\ml-app.log)</td>
</tr>
<tr>
<td><strong>NEXT-CHANGE</strong></td>
<td></td>
</tr>
<tr>
<td>next-change.min-changes</td>
<td>Minimum number of actions in sequence to generate suggestions</td>
</tr>
</tbody>
</table>
Upgrading Monarch Swarm

Upgrading the Monarch Swarm Application

To gain access to Monarch Swarm’s newer features, you must upgrade the application. Note that cached data sets are reset when upgrades are performed. Statistics are not reset.

To upgrade your Monarch Swarm installation, simply run the installer of the newer version of the application. While no uninstallation of the previous version is necessary, the Cassandra, Monarch Swarm, Monarch Swarm Data Engine, Monarch Swarm ML, and Monarch Swarm Spark services must be stopped prior to the upgrade.

Steps:
1. Double-click on the application installer to run it.
2. Read the license agreement that displays in the next screen and, if you agree with its terms, select the button for I accept the agreement. Click Next when you are finished.
The Database Configuration screen displays.

3. In the fields provided, enter the **PostgreSQL administrator login** and **password** that were provided when the older version of Monarch Swarm was installed.

4. Click **Next** when you are finished.

   The *Select Components* screen displays.

5. If you wish to make changes to the components you had previously installed, you can do so now. Otherwise, simply click **Next**.
6. Click **Install**.

   The installation begins.
7. After installation, the following screen displays.

![Monarch Swarm Setup Screen]

8. Click **Finish**.

   The Monarch Swarm login page displays.
Upgrading Java Runtime Environment

The Java Runtime Environment (JRE) version in the server in which Monarch Swarm has been installed can be upgraded. To do so, download the new JRE version from the official Oracle site, uninstall the old JRE version, and then install the new one. Proceed with running the installer or upgrading Monarch Swarm.

Alternatively, the steps below describe how to upgrade Java Runtime Environment if the application is not going to be upgraded or reinstalled.

**Steps:**

1. Stop all of the related Monarch Swarm services.
2. Go to Control Panel\System and Security\System and select Advanced system settings.
3. In the System Properties dialog that displays, click Environment Variables.
4. Under System variables, change the Java path in Path to the correct value.

   For example, if the current path reads "C:\Program Files\Java\jre1.8.0_161\bin" and your JRE was upgraded to jre1.8.0_181, change the path to "C:\Program Files\Java\jre1.8.0_181\bin" and then click OK on the Edit System Variable dialog to close it.

5. Check that the new path is applied.
6. Click **OK** on the *Environment Variables* and *System Properties* dialogs to close them.

7. In `C:\Program Files\PostgreSQL\9.6\pgAdmin 4\bin`, double-click on the pgAdmin4 application to launch it.

   The pgAdmin4 user interface displays.

![pgAdmin4 interface](image)

8. Connect to the **PostgreSQL 9.6** server using the default user "**postgres**" and password "**postgres**."

9. Locate the database **newserver_dataengine > Schemas > public**.

10. Open the Query Tool by right-clicking on the schema **public** and then selecting **Query Tool** from the options that display.

![Query Tool interface](image)
11. Specify the following script:

   SET pljava.libjvm_location TO 'path';

   ALTER DATABASE newserver_dataengine SET pljava.libjvm_location FROM CURRENT;

   where path is the path to the jvm.dll file of the upgraded JRE (e.g., C:\Program Files\Java\jre1.8.0_181\bin\server\jvm.dll),

12. Execute this script by clicking the **Execute/Refresh (F5)** icon.

13. Close pgAdmin4 and restart the Monarch Swarm service.

**NOTE**

If your Monarch Swarm application was installed using the HTTPS protocol and Java Runtime Environment was upgraded, you must generate a new security certificate. Click [here](#) for instructions on how to obtain this certificate and configure client machines.
Troubleshooting Monarch Swarm

The items in this section describe troubleshooting measures you may need to take when working with Monarch Swarm.

Uninstalling the Application

Monarch Swarm may be uninstalled prior to upgrading it.

Steps:

1. In the folder **Control Panel > Programs > Uninstall a program**, locate the **Monarch Swarm** application, right-click on it, and then select **Uninstall** from the menu that displays.

2. Click **Yes** on the message box that displays.
   
   Monarch Swarm is uninstalled.

3. Uninstall the Datawatch JDBC drivers and PostgreSQL.
   
   Note that other applications besides Monarch Swarm may require PostgreSQL to run correctly. Consult your system administrator before uninstalling the program. Information on how to completely remove PostgreSQL from your server may be found online.

4. Restart your computer.
   
   A newer version of Monarch Swarm may now be installed.
Starting the Service

In some instances, Monarch Swarm may not be accessed from any browser because the service has not been started. This may occur when the operating system of the machine in which the application is installed is updated or restarted.

If the Monarch Swarm application does not start automatically, you may need to start it manually.

Steps:


2. In Services window that displays, locate the Datawatch Monarch Swarm item and then click Start in the left-hand pane of the Services list.

3. Repeat Step 2 for the Monarch Swarm Data Engine, Monarch Swarm ML, and Monarch Swarm Spark services.

NOTE

You can also select the item, right-click on your mouse, and then select Start from the options that display.

The Monarch Swarm services may be started in any order.
4. Close the Services window.

You may need to wait a few minutes to allow these services to restart completely.

The Monarch Swarm login page can now be loaded as usual.

---

### Starting the Cassandra Service

There may be instances when the Cassandra service fails to start automatically (for example, after an OS restart). This may occur because of a corrupted commitlogs file. The following steps describe how to start the Cassandra service when the issue is a corrupted commitlogs file.

**Steps:**

1. Open the Cassandra log file. This file is usually located in C:\Program Files\Datawatch Monarch Swarm\cassandra\logs\system.log.

2. Check the name of the corrupted commitlog file with the latest error.

   In the example below, the corrupted commitlog file is C:\\Program Files\\Datawatch Monarch Swarm\\cassandra\\data\\commitlog\\CommitLog-6-1509613026245.log.
3. Delete this file from the file system indicated.
4. Start the Cassandra service.

### Backing up the Monarch Swarm PostgreSQL Database

Backing up your Monarch Swarm databases is necessary to ensure that all of your data, workspace definitions, security settings, and the like are preserved when PostgreSQL or the application itself is upgraded or reinstalled.

When Monarch Swarm is installed, PostgreSQL 9.6 and its management tool, pgAdmin4, are concurrently installed. This tool can be used to back up and restore your database. The PostgreSQL database includes information on, among others, users, data source/workspace/connection metadata, and permissions.

**NOTE**

The binary path to the directory containing the PostgreSQL utility programs (e.g., pg_dump, pg_restore, etc.) must be specified in **File > Preferences > Paths > Binary paths** before you can back up a database. Contact your system administrator if you need help doing so. More information on this setting may be found [here](#).

**Steps:**

1. In **C:\Program Files\PostgreSQL\9.6\pgAdmin 4\bin**, double-click on the pgAdmin4 application to launch it.
The pgAdmin4 user interface displays.

2. Expand the **Servers** node and then double-click **PostgreSQL 9.6**.

   A login dialog displays.

3. Enter the password for the user **postgres** and then click **OK**.

   You are connected to the PostgreSQL server, and several items display below it.
4. Expand the **Databases** node and then select the Monarch Swarm database. Right-click on it and then select **Backup** from the options that display.
The Backup dialog displays.

5. Provide the information required in the General tab of this dialog and then click on the Dump options tab.

   **NOTE**

   Save the information you provide in the General tab of the Backup dialog. You will need to provide this information when restoring the database at a later time.

6. Move the switches in the Sections box to specify which portions of the database should be backed up.
7. Move the switches in the **Type of Objects** box to specify which database components to back up.

8. Move the switches in the **Do not save** box to specify which database components to exclude from the backup.

9. Move the switches in the rest of the field boxes provided in the tab to specify other settings for your backup.

10. Click **Backup** at the bottom of the dialog when you are finished. Otherwise, click **Cancel** to abort the backup operation.

A popup dialog similar to that shown below displays to indicate successful or unsuccessful backup.

```
Backing up an object on the server 'PostgreSQL 9.6 (localhost:5432) from database 'newserver'...
Thu Jan 19 2017 16:27:24 GMT+0800 (China)
3.85 seconds
Click here for details.
Successfully completed.
```

Details of the backup operation, whether successful or not, may be viewed in the **Process Watcher**, which is launched by clicking on the **Click here for details** link provided in the popup.

By default, your backup will be located in `C:\Program Files\PostgreSQL\9.6\pgAdmin 4\bin`.

More information on how to backup databases in PostgreSQL may be found [here](#).

---

**Restoring the Monarch Swarm PostgreSQL Database**

When PostgreSQL or Monarch Swarm is upgraded or reinstalled, you may need to restore a previously backed-up Monarch Swarm database to continue working with your data.

The pgAdmin4 tool is also used to restore Monarch Swarm databases.

**Steps:**

1. Run the **pgAdmin4** tool and log into the **PostgreSQL 9.6** server.
2. Create a new Monarch Swarm database. If this database already exists, select it.
3. Right-click on this database and, from the options that display, select **Restore**.
The Restore dialog displays.

4. Provide the information required by each of the fields in the General tab of this dialog. This information should match the details you specified in the General tab of the Backup dialog when you backed up your database.

5. Click the Restore options tab.

6. Use the switches provided in each of the field boxes in this tab to specify how your database should be restored.

7. Click Restore at the bottom of the dialog when you are finished. Otherwise, click Cancel to abort the restore operation.

A popup window displays to indicate successful or unsuccessful restoration.

Details of the restore operation, whether successful or not, may be viewed in the Process Watcher, which is launched by clicking on the Click here for details link provided in the popup.

More information on how to restore databases in PostgreSQL may be found here.

**Back up the Cassandra Database**

The following objects are created as separate tables in the Cassandra database under the keyspace newserver.

- Tables exported to the Monarch Swarm library from the application
- Tables exported from Data Prep Studio to Monarch Swarm
- Workspaces with pinned tables saved to Monarch Swarm
The steps below describe how to save snapshots of these tables for back up.

Steps:

1. Set the JAVA_HOME path by going to Control Panel\System and Security\System and then selecting Advanced system settings.
2. In the System Properties dialog that displays, click Environment Variables.
3. In the Environment Variables dialog, click New... under System variables, enter JAVA_HOME as the variable name and then enter the path to your Java application in the Variable value field. Click OK when you are done and then OK in the System Properties dialog.
4. Launch cmd.exe.
5. Go to the bin directory of Cassandra.
   Example: C:\Program Files\Datawatch Monarch Swarm\cassandra\bin
6. Run the following command to create the snapshot for one keyspace:
   nodetool -h host -p port snapshot keyspace_name
   Example: nodetool -h localhost -p 7199 snapshot newserver.

**NOTE**

Monarch Swarm creates two keyspaces in the Cassandra database: newserver and datawatch. Both keyspaces should be backed up.

The files created are saved in the snapshot directory, usually in C:\{(Monarch Swarm Home)\cassandra\data\data\{keyspace name}\{table name}\snapshots

Folder example:
C:\Program Files\Datawatch Monarch Swarm\cassandra\data\data\newserver\data_scheme-3486829030af11e7a545dd2ebe24b125\snapshots\1493962486390

**BACKING UP THE CASSANDRA DATABASE FOR MONARCH SWARM VERSIONS 2.0 AND EARLIER**

The steps described above may not produce the desired results if you are working with Monarch Swarm versions prior to version 2.1 or upgraded and older version to 2.1 and the installation path for Cassandra contains spaces. This issue originates from the default Cassandra configuration file.

In this case, the following steps should be done before snapshot creation:
Steps:

1. Open Services and stop the Cassandra service.

2. Open the cassandra directory and update Line 380 of the default config for Cassandra database (usually in C:\Program Files\Datawatch Monarch Swarm\cassandra\conf\cassandra-env.ps1) as follows:

   Default:

   $env:JVM_OPTS = "$env:JVM_OPTS -XX:CompileCommandFile=$env:CASSANDRA_CONF\hotspot_compiler"

   Update to:

   $env:JVM_OPTS = "$env:JVM_OPTS -XX:CompileCommandFile=""$env:CASSANDRA_CONF\hotspot_compiler""

3. Open command prompt with Administrator rights, go to the bin directory of Cassandra (usually in C:\Program Files\Datawatch Monarch Swarm\cassandra\bin) and then run the following command:

   cassandra -install

4. Restart the Cassandra service.

Restoring the Cassandra Database

You can restore your Cassandra database to repair corrupted tables or create a completely new node. Note that the steps outlined in this section assume that you have properly backed up your Cassandra database and that all of the necessary snapshots are saved in the snapshot directory.

To restore the Cassandra database in the case of corrupted data, the following steps are taken.

Steps:

1. Open cmd.exe.

2. Go to C:\Program Files\Datawatch Monarch Swarm\cassandra\bin.

3. Run nodetool drain.

4. Shut down the Cassandra and Swarm services.

5. Delete all of the files in the Monarch Swarm commitlog directory (in C:\Program Files\Datawatch Monarch Swarm\cassandra\data\commitlog).

6. Delete all of the *.db files in C:\Program Files\Datawatch Monarch Swarm\cassandra\data\data\{keyspace name}\{table name}.

   Example: C:\Program Files\Datawatch Monarch Swarm\cassandra\data\data\newserver\ds_10015-e36e6f90349011e7858ecdb96aa2c5e0
Note that Step 6 must be repeated for all corrupted tables.

7. Go to the snapshot directory and copy all of the contents (per table) to the matching data_directories from which you deleted DB files in the previous step.
   Step 7 must be repeated for all tables to be restored.

8. Restart the **Cassandra** and **Monarch Swarm** services.


10. Restart the Monarch Swarm service.

Follow the steps below to restore create a completely new node using saved Cassandra table snapshots.

**Steps:**

1. Set the Python path by going to Control Panel > System and Security > System > Advanced Settings > Environment Variables.
2. Add C:\Program Files\Datawatch Monarch Swarm to the PATH variable.
3. Go to C:\Program Files\Datawatch Monarch Swarm\cassandra\bin.
4. Copy all of the schema.cql files generated from backup to the bin directory.
5. Recreate the schema. To do so, type `cqlsh -f schema.cql` into the command prompt and press **Enter** on your keyboard.
6. Copy the backup files to the corresponding directories. Note that this step must be performed for all directories to be restored.
7. In command prompt, run nodetool refresh newserver <tablename>.
8. Restart the Monarch Swarm service.

**Product Support**

Product support for Monarch Swarm may be obtained [here](#).