



# Configuring alerts in Denodo

Revision 20240325

## NOTE

This document is confidential and proprietary of **Denodo Technologies**. No part of this document may be reproduced in any form by any means without prior written authorization of **Denodo Technologies**.

Copyright © 2024  
Denodo Technologies Proprietary and Confidential

## CONTENTS

<b>1 INTRODUCTION.....</b>	<b>3</b>
<b>2 JMX NOTIFICATIONS.....</b>	<b>4</b>
<b>2.1 WHEN TO USE.....</b>	<b>4</b>
<b>2.2 DESCRIPTION.....</b>	<b>4</b>
<b>2.3 STEPS TO CONFIGURE.....</b>	<b>4</b>
<b>3 LOG4J ALERTS.....</b>	<b>11</b>
<b>3.1 WHEN TO USE.....</b>	<b>11</b>
<b>3.2 DESCRIPTION.....</b>	<b>11</b>
<b>3.3 STEPS TO CONFIGURE.....</b>	<b>11</b>
<b>4 DENODO SCHEDULER HANDLERS.....</b>	<b>13</b>
<b>4.1 WHEN TO USE.....</b>	<b>13</b>
<b>4.2 DESCRIPTION.....</b>	<b>13</b>

## 1 INTRODUCTION

---

This document explains how to send notifications by mail regarding events occurring in Denodo Virtual DataPort.

In some situations we need to be alerted if a specific event takes place in Denodo. In order to achieve that, we can use different techniques detailed in this article: log4j appenders, JMX alerts and Denodo Scheduler Handlers.

## 2 JMX NOTIFICATIONS

---

### 2.1 WHEN TO USE

To register alerts regarding **Memory/CPU usage** information or any other information available in the **MBeans** exposed through JMX by the Denodo Platform.

### 2.2 DESCRIPTION

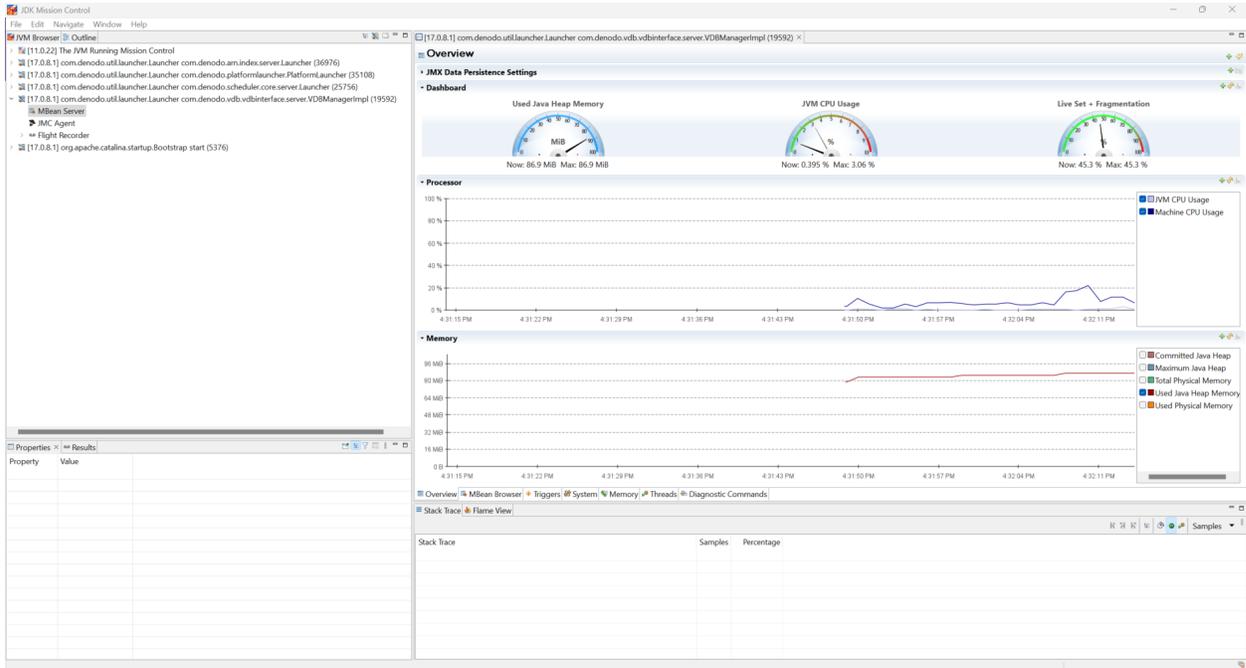
[Java Mission Control](#) (JMC) is an Oracle tool for managing, monitoring, profiling, and troubleshooting Java applications. We will see how to use this tool in order to monitor the Denodo Platform MBeans and to generate alerts based on their status.

### 2.3 STEPS TO CONFIGURE

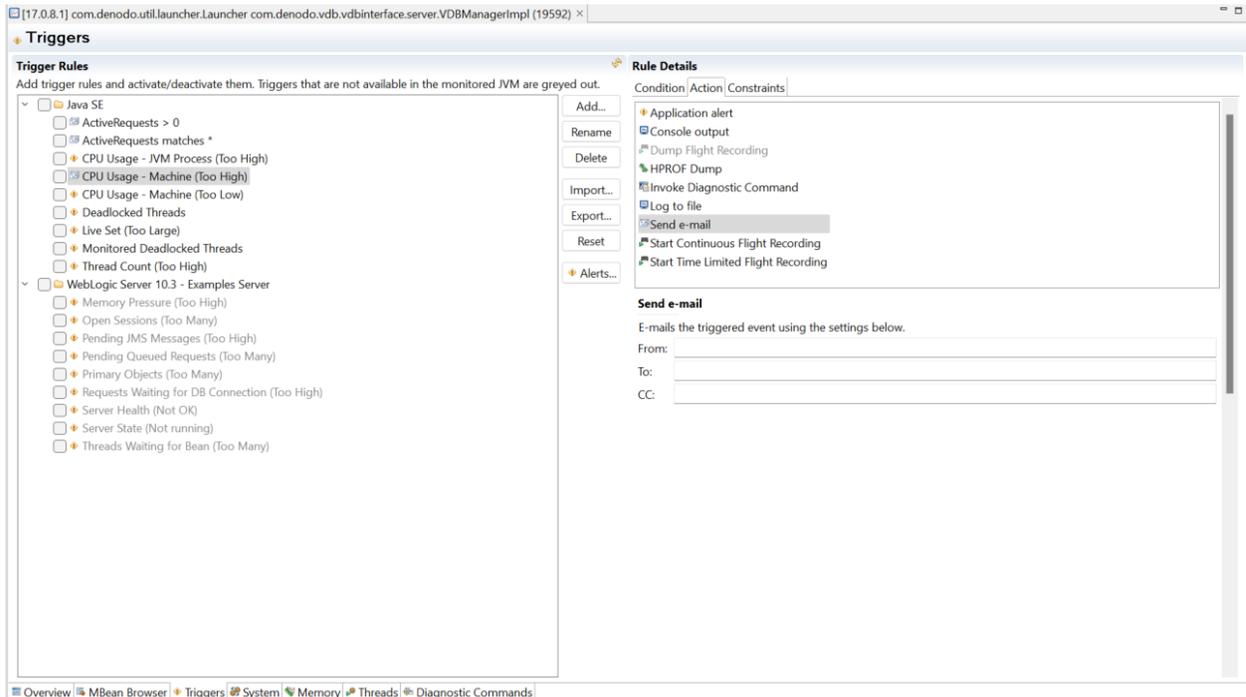
JMC is included as part of the Java distribution, if a JDK is already installed JMC can be executed (otherwise the Java Development Kit needs to be installed). Once the tool is running, perform the following steps:

1. Execute `jmc.exe` located at `<JDK_PATH>/bin`.
2. The "JVM Browser" shows the processes running locally. When the VDP Server is running in a remote machine you can connect with a remote JVM using the option "File" > "Connect" > "Create a new connection".
3. From the JVM Browser search for the VDP Server process. You can identify the process by the process ID (searching previously for this number) or identifying the element which command line starts as following:

```
com.denodo.util.launcher.Launcher  
com.denodo.vdb.vdbinterface.server.VDBManagerImpl...
```

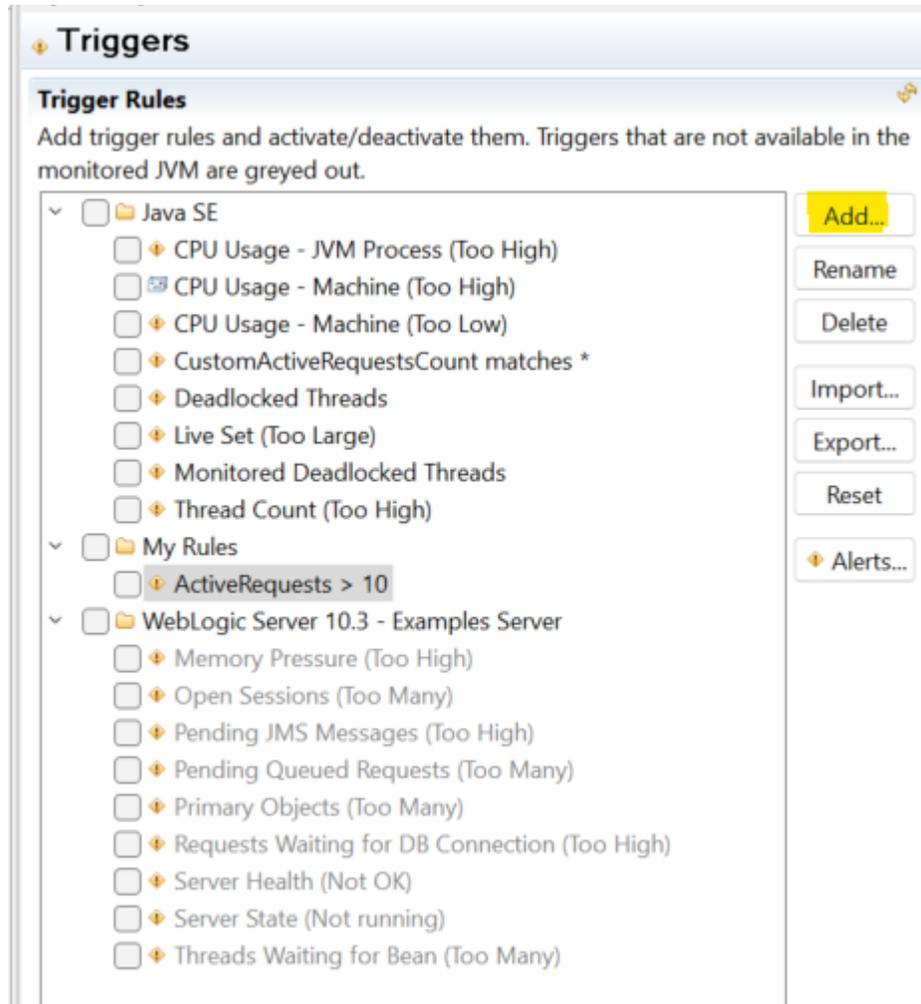


4. By clicking on the JVM Browser element and double clicking on “MBean Server” JMC will connect to the VDP Server.
5. On the MBeanServer wizard a **Triggers** tab is available at the bottom. This will open a wizard where trigger rules can be configured (activated and deactivated) for several events like CPU usage or Deadlocked threads. New trigger rules can also be created based on the exposed MBeans.

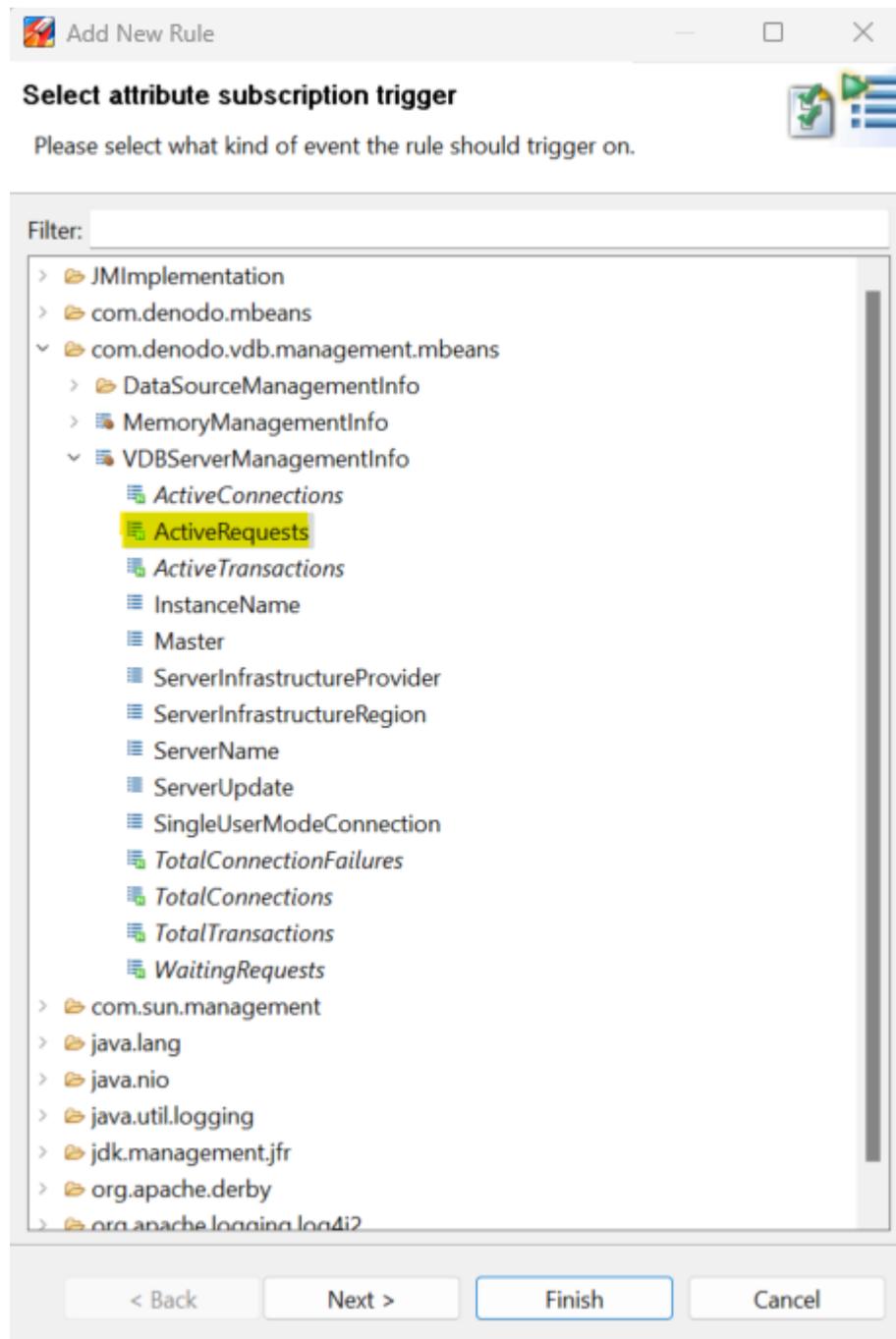


As an example we are going to create a new Trigger that sends an email every time that the number of active requests exceeds a limit number. The steps will be the following:

- a. On the Triggers tab click on “Add...” button.



- b. Search for the MBean containing the information and click on “Finish”. In the KB article [“Monitoring Denodo servers with JMX”](#) more information about the available MBeans can be found.



- c. A new wizard called “**Rule Details**” will show up. In that wizard the values for triggering the condition can be set. In this example the “current value” will be the limit that, when reached, the email will be sent.

**Rule Details**

Condition | Action | Constraints

---

**Description**

MBean Path:

Attribute Name:

Current Value:

Max trigger value:

Sustained period:

Limit period:

Trigger when condition is met.

Trigger when recovering from condition

- Once the condition has been defined, we need to make sure that JMC will evaluate the values properly so the condition can be checked. This means that we need to “cast” the attribute name indicating how it can be measured. To do that, go to the MBeans tab and search for the appropriate bean and the attribute to be measured. In this case, the bean is “**VDBServerManagementInfo**” and the attribute “**ActiveRequests**”.

**MBean Browser**

MBean Tree + MBean Features

Filter:

- > JMIImplementation
- > com.denodo.mbeans
- > com.denodo.vdb.management.mbeans
  - > DataSourceManagementInfo
  - MemoryManagementInfo
  - VDBServerCacheManagementInfo
  - VDBServerManagementInfo**
  - VDBServerRequestsManagementInfo
- > com.sun.management
- > java.lang
  - > GarbageCollector
  - > MemoryManager
  - > MemoryPool
  - ClassLoading
  - Compilation
  - Memory
  - OperatingSystem
  - Runtime
  - Threading
- > java.nio
- > java.util.logging
- > jdk.management.jfr
- > org.apache.derby
- > org.apache.logging.log4j2

Name	Value	Update Interval
ActiveConnections	5	Default
<b>ActiveRequests</b>	0	Default
ActiveTransactions	0	Default
InstanceName	<String>	Default
Master	true	Default
ServerInfrastructureProvider	<String>	Default
ServerInfrastructureRegion	<String>	Default
ServerName	//localhost:9997/VQLServer	Default
ServerUpdate	9.0	Default
SingleUserModeConnection		Default
TotalConnectionFailures	0	Default
TotalConnections	8	Default
TotalTransactions	0	Default
WaitingRequests	0	Default

Once located, right click on “**ActiveRequests**” to indicate to JMC how it needs to manage the retrieved value. In this case, the value will be interpreted as a number, in units of 1.

Name	Value	Update Interval
ActiveConnections	5	Default
ActiveRe		Default
ActiveTr		Default
Instance		Default
Master		Default
ServerIn		Default
ServerIn		Default
ServerN	QLServer	Default
ServerU		Default
SingleUs		Default
TotalCor		Default
TotalCor		Default
TotalTrar		Default
WaitingI		Default

- When the trigger is configured by clicking on the “Action” tab the “Send e-mail” action can be configured.

**Rule Details**

Condition | Action | Constraints

- Application alert
- Console output
- Dump Flight Recording
- HPROF Dump
- Invoke Diagnostic Command
- Log to file
- Send e-mail**
- Start Continuous Flight Recording
- Start Time Limited Flight Recording

**Send e-mail**

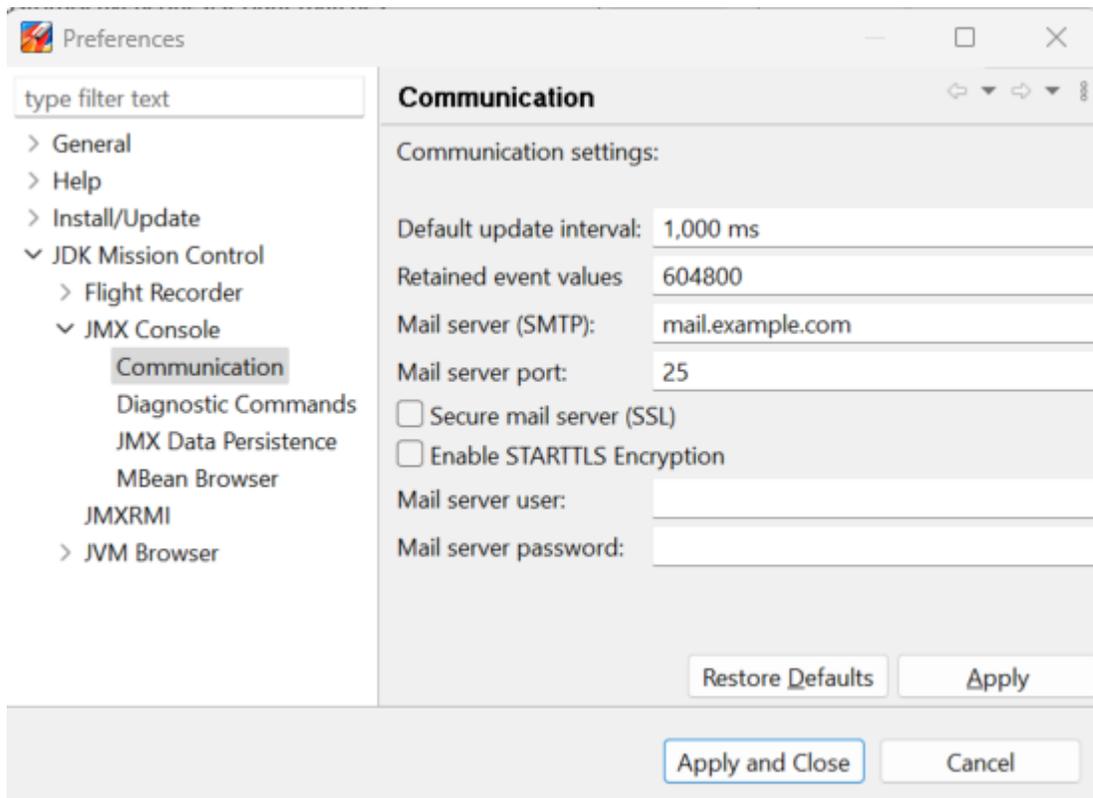
E-mails the triggered event using the settings below.

From:

To:

CC:

8. In addition, the SMTP server has to be configured in order to be able to send the email. For that go to **Window > Preferences > Java Mission Control > JMX Console > Communication** and set up the SMTP Server configuration.



The screenshot shows the 'Preferences' dialog box with the 'Communication' tab selected. The left sidebar shows the navigation tree with 'JMX Console > Communication' highlighted. The main area contains the following settings:

- Communication settings:
- Default update interval: 1,000 ms
- Retained event values: 604800
- Mail server (SMTP): mail.example.com
- Mail server port: 25
- Secure mail server (SSL)
- Enable STARTTLS Encryption
- Mail server user:
- Mail server password:

Buttons at the bottom: Restore Defaults, Apply, Apply and Close, Cancel.

## 3 LOG4J ALERTS

---

### 3.1 WHEN TO USE

To get an e-mail alert of an event logged in the Denodo Virtual DataPort logs.

### 3.2 DESCRIPTION

Virtual DataPort uses the logging library **Apache Log4j** to log the activity of the Virtual DataPort server and its Administration Tool. Log4j allows defining the log level (TRACE, DEBUG, INFO, WARN, ERROR or FATAL) for different categories. The configuration of Log4j for the Virtual DataPort server is controlled by the file:

```
<DENODO_HOME>/conf/vdp/log4j2.xml
```

For the administration tool by the file:

```
<DENODO_HOME>/conf/vdp-admin/log4j2.xml.
```

In these files you can find three main components: Appenders, PatternLayouts and Loggers.

A logger defines a log level for a category and they can be assigned to appenders previously configured. For instance:

```
<RollingFile name="VDBOUT" fileName="../../../logs/vdp/vdp-cache.log">
  <PatternLayout pattern="[%t] %d{yyyy-MM-dd'T'HH:mm:ss.SSS}"/>
</RollingFile>

<Logger name="com.denodo.vdb" level="error">
  <AppenderRef ref="VDBOUT" />
</Logger>
```

This configuration indicates that the category "com.denodo.vdb" on an error event will write the output of the log to the file configured in the VDBOUT appender following the defined pattern layout.

The **SMTPAppender** is an appender which sends an e-mail when a specific logging event occurs, typically on errors or fatal errors. Therefore, instead of writing the output to a file it is possible to configure a destination email address.

### 3.3 STEPS TO CONFIGURE

1. Configure the appender. The appender contains the information related to the SMTP server used for sending the email.

```
<SMTP name="EMAILALERT"
  subject="VDP Log Alert"
  to="toUser@mydomain.com"
  from="fromUser@mydomain.com"
  smtpHost="mail.mydomain.com"
  smtpPort="25"
  smtpProtocol="smtp"
  smtpUsername="mysmtpusername">
```

```
smtpPassword="mysmtpassword"  
smtpDebug="true"  
bufferSize="512">  
</SMTP>
```

2. Configure the loggers. Once the appender is configured, any logger with a log level for a category can be configured indicating the email appender as output:

```
<Logger name="com.denodo.vdp" level="FATAL">  
  <AppenderRef ref="EMAILALERT" />  
</Logger>
```

## 4 DENODO SCHEDULER HANDLERS

### 4.1 WHEN TO USE

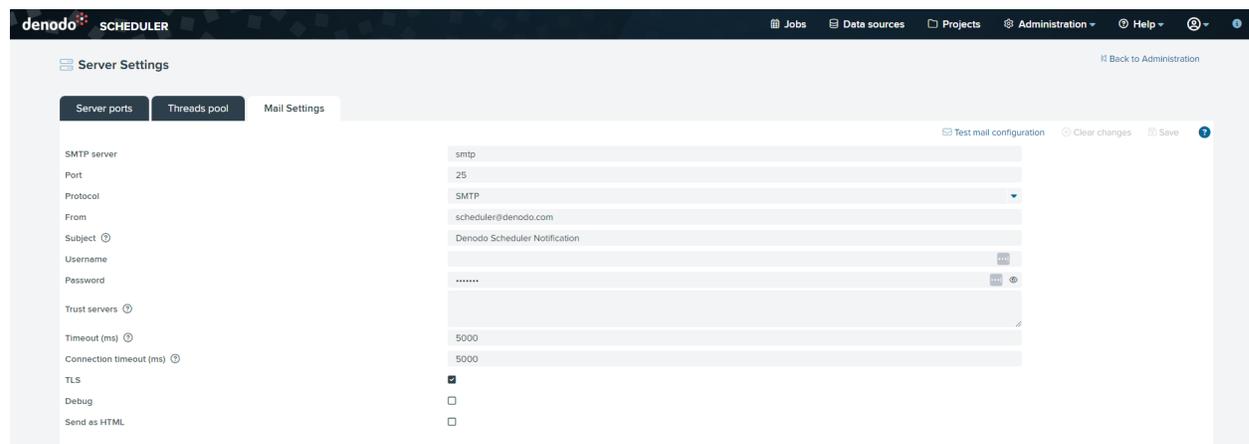
To send an email alert about the failure of a particular Job in Denodo Scheduler.

### 4.2 DESCRIPTION

When configuring Jobs in Scheduler it is possible to configure handlers. A handler is an action that is executed as the last step before completing the execution of a job (if the conditions are met). Denodo Scheduler includes as a built-in handler the “Mail” handler. This handler allows to send an email with the result report on the job execution. It requires a list of destination email addresses to be specified and allows several conditions to be set (determining in which cases the mail has to be sent).

## Steps to configure

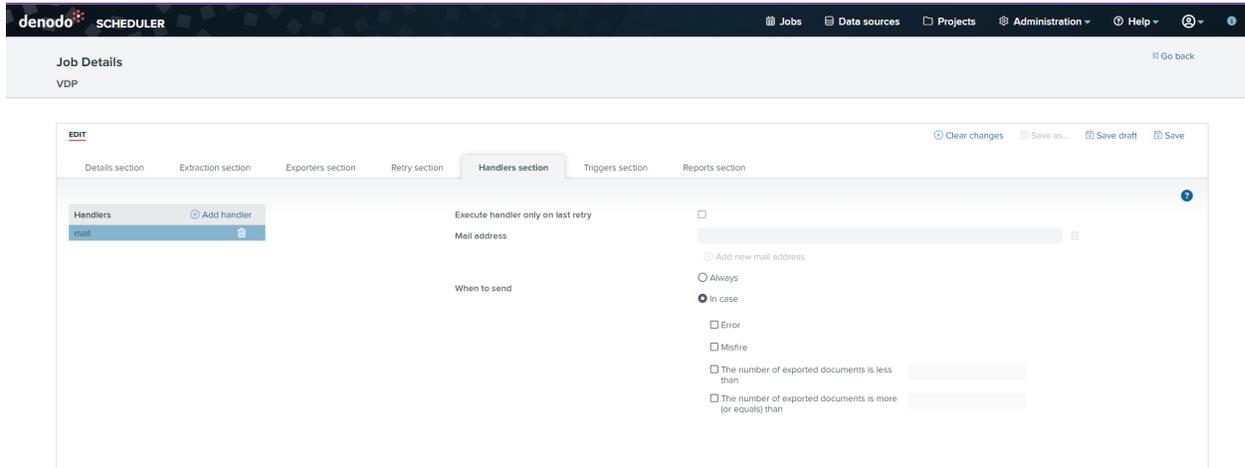
In order to be able to send the email Scheduler requires a SMTP server to be configured. This configuration has to be set in “Administration” > “Server” > “Mail Settings”.



The screenshot shows the Denodo Scheduler Administration interface. The top navigation bar includes "denodo SCHEDULER" and menu items for "Jobs", "Data sources", "Projects", "Administration", and "Help". The main content area is titled "Server Settings" and has three tabs: "Server ports", "Threads pool", and "Mail Settings". The "Mail Settings" tab is active, showing a form for configuring the SMTP server. The form includes fields for "SMTP server" (smtp), "Port" (25), "Protocol" (SMTP), "From" (scheduler@denodo.com), "Subject" (Denodo Scheduler Notification), "Username", "Password", "Trust servers", "Timeout (ms)" (5000), "Connection timeout (ms)" (5000), "TLS" (checked), "Debug" (unchecked), and "Send as HTML" (unchecked). There are also buttons for "Test mail configuration", "Clear changes", and "Save".

Once the mail server is set you have to follow this steps:

1. Select the Job (or create a new one).
2. Click on the “[Handlers section](#)” of the Job configuration.
3. Select “Add Handler” > “mail”.
4. Add the target mail addresses and the configuration “When to send”.
5. Save the Job.



The screenshot shows the Denodo Scheduler interface for configuring a job. The top navigation bar includes 'denodo SCHEDULER' and menu items for 'Jobs', 'Data sources', 'Projects', 'Administration', 'Help', and a user profile. The main header shows 'Job Details' for 'VDP' with a 'Go back' link. The interface is divided into several sections: 'Details section', 'Extraction section', 'Exporters section', 'Retry section', 'Handlers section' (which is active), 'Triggers section', and 'Reports section'. In the 'Handlers section', there is a list of handlers with 'mail' selected. To the right, configuration options are shown: 'Execute handler only on last retry' (unchecked), 'Mail address' (with a text input field and an 'Add new mail address' button), and 'When to send' (with radio buttons for 'Always' and 'In case' (selected), and checkboxes for 'Error', 'Misfire', 'The number of exported documents is less than', and 'The number of exported documents is more (or equals) than').

Next time the Job is executed, before finishing the execution, Scheduler will check if the “When to send” conditions are met and, if it is the case, it will send the Job report to the configured target emails.