



# Denodo Monitor as a Windows service

Revision 20211018

## 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 © 2022  
Denodo Technologies Proprietary and Confidential

## Goal

This document describes how to register the Denodo Monitor as a Windows Service.

## Content

To register the [Denodo Monitor](#) as a service you will need a configuration file containing the information about the environment variables, configuration parameters, etc.

To make the installation of the service easier, a small script can be created to run the commands that are necessary to register the service. Create this script file under `<DENODO_MONITOR_HOME>/bin` and the configuration file under `<DENODO_MONITOR_HOME>/conf`. The script makes the assumption that the configuration file is located under this folder and is called "service.conf". We are going to name the script "denodomonitor\_service.bat".

The configuration slightly differs for Denodo 8 and its prior versions. We have differentiated the configuration in the following two sections: one for Denodo 8 and other for Denodo 7 & 6.

### Denodo 8 Configuration

Use the following template as a starting point for the script file (denodomonitor\_service.bat). Modify the value of DENODO\_HOME variable with the installation folder of the Denodo Platform using backward slashes (\) and make sure to check the MONITOR\_HOME variable is set correctly:

```
echo off

REM -----
REM  Installation Script for running the Denodo Monitor as a Service
REM  in Windows
REM  -----

REM # DENODO_HOME must be set.
SET DENODO_HOME=<DENODO_HOME>
SET TEMP=%DENODO_HOME%\temp\monitor-service
SET TMP=%DENODO_HOME%\temp\monitor-service
SET MONITOR_HOME=%DENODO_HOME%\tools\monitor\denodo-monitor
SET DENODO_JRE11_OPTIONS=--add-opens java.base/java.lang=ALL-UNNAMED --add-
opens java.base/java.lang.invoke=ALL-UNNAMED

if "%DENODO_HOME%"==" " goto exit

SET DENODO_JRE_HOME=%DENODO_HOME%/jre
SET JAVA_BIN=%DENODO_JRE_HOME%\bin\java.exe

if exist "%JAVA_BIN%" goto configuredjavabin
SET JAVA_BIN=%DENODO_JRE_HOME%\bin\java.exe

if exist "%JAVA_BIN%" goto configuredjavabin
SET JAVA_BIN=%DENODO_JRE_HOME%\bin\java.exe
```

```
if exist "%JAVA_BIN%" goto configuredjavabin
SET JAVA_BIN=%JAVA_HOME%\jre\bin\java.exe

if exist "%JAVA_BIN%" goto configuredjavabin
SET JAVA_BIN=%JAVA_HOME%\bin\java.exe

:configuredjavabin

if exist "%JAVA_BIN%" (
  if "%1"=="start" goto :doStart
  if "%1"=="stop" goto :doStop
  if "%1"=="install" goto :doInstall
  if "%1"=="remove" goto :doRemove
  if "%1"=="state" goto :doInfo

goto :usage
)

echo "Unable to execute %0: Environment variable JAVA_HOME must be set"

goto :end

:doStart
"%JAVA_BIN%" -DDENODO_APP="Denodo Monitor" %DENODO_JRE11_OPTIONS% -jar
"%DENODO_HOME%/lib/service-wrapper/wrapper.jar" -t "%MONITOR_HOME
%/conf/service.conf"
goto :end

:doStop
"%JAVA_BIN%" %DENODO_JRE11_OPTIONS% -jar "%DENODO_HOME%/lib/service-
wrapper/wrapper.jar" -p "%MONITOR_HOME%/conf/service.conf"
goto :end

:doInstall
"%JAVA_BIN%" %DENODO_JRE11_OPTIONS% -jar "%DENODO_HOME%/lib/service-
wrapper/wrapper.jar" -i "%MONITOR_HOME%/conf/service.conf"
goto :end

:doRemove
"%JAVA_BIN%" %DENODO_JRE11_OPTIONS% -jar "%DENODO_HOME%/lib/service-
wrapper/wrapper.jar" -r "%MONITOR_HOME%/conf/service.conf"
goto :end

:doInfo
"%JAVA_BIN%" %DENODO_JRE11_OPTIONS% -jar "%DENODO_HOME%/lib/service-
wrapper/wrapper.jar" -q "%MONITOR_HOME%/conf/service.conf"
```

```
goto :end

:exit

  echo "Unable to execute %0: Environment variables JAVA_HOME and DENODO_HOME
must be set (absolute path)"
  goto :end

:usage
  echo "Usage: %0 [start|stop|install|remove|state]"

:end
```

Use the following template for the configuration file (service.conf) replacing <DENODO\_HOME> with the appropriate value.

```
wrapper.java.command=<DENODO_HOME>/jre/bin/java.exe
wrapper.working.dir=<DENODO_HOME>/tools/monitor/denodo-monitor

wrapper.java.app.mainclass=com.denodo.util.launcher.Launcher
wrapper.java.classpath.1=<DENODO_HOME>/lib/denodo-commons-launcher-util.jar

wrapper.java.additional.1=-DconfFile=ConfigurationParameters.properties
wrapper.java.additional.2=-DverboseMode=false
wrapper.java.additional.3=-DWINDOWS_SERVICE=true
wrapper.java.additional.4=-
Djava.system.class.loader=com.denodo.util.launcher.DenodoClassLoader
wrapper.java.additional.5=-Djava.locale.providers=COMPAT,SPI

wrapper.java.initmemory=3

wrapper.app.parameter.1=com.denodo.monitor.cli.DiagnosticTool
wrapper.app.parameter.2=-lib
wrapper.app.parameter.3=<DENODO_HOME>\\tools\\monitor\\denodo-monitor\\lib
wrapper.app.parameter.4=-conf
wrapper.app.parameter.5=<DENODO_HOME>\\tools\\monitor\\denodo-monitor\\conf
wrapper.app.parameter.6=-arg
wrapper.app.parameter.7=-d
wrapper.app.parameter.8=-arg
wrapper.app.parameter.9=<DENODO_HOME>\\tools\\monitor\\denodo-monitor\\logs
wrapper.app.parameter.10=-arg
wrapper.app.parameter.11=%1
wrapper.app.parameter.12=-arg
wrapper.app.parameter.13=%2
wrapper.app.parameter.14=-arg
wrapper.app.parameter.15=%3
```

```
wrapper.logfile=<DENODO_HOME>\\tools\\monitor\\denodo-monitor\\logs
wrapper.logfile.loglevel=ERROR
wrapper.logfile.format=LPTM
wrapper.logfile.maxsize=10m
wrapper.logfile.maxfiles=10
wrapper.syslog.loglevel=ERROR

wrapper.ntservice.name=Denodo Monitor
wrapper.ntservice.displayname=Denodo Monitor 8.0
wrapper.ntservice.description=Denodo Monitor - Denodo Platform 8.0
wrapper.ntservice.dependency.1=
wrapper.ntservice.starttype=DEMAND_START
wrapper.ntservice.interactive=false

#wrapper.affinity=
wrapper.java.maxmemory=1024
```

## Denodo 7 and 6 Configuration

Use the following template as a starting point for the script file (denodomonitor\_service.bat). Modify the value of DENODO\_HOME variable with the installation folder of the Denodo Platform using backward slashes (\) and make sure to check the MONITOR\_HOME variable is set correctly:

```
echo off

REM -----
REM  Installation Script for running the Denodo Monitor as a Service
REM  in Windows
REM  -----

REM # DENODO_HOME must be set.
SET DENODO_HOME=<DENODO_HOME>
SET MONITOR_HOME=%DENODO_HOME%\tools\monitor\denodo-monitor

if "%DENODO_HOME%"==" " goto exit

SET DENODO_JRE_HOME=%DENODO_HOME%\jre
SET JAVA_BIN=%DENODO_JRE_HOME%\bin\java.exe

if exist "%JAVA_BIN%" goto configuredjavabin
SET JAVA_BIN=%DENODO_JRE_HOME%\bin\java.exe

if exist "%JAVA_BIN%" goto configuredjavabin
SET JAVA_BIN=%DENODO_JRE_HOME%\bin\java.exe

if exist "%JAVA_BIN%" goto configuredjavabin
SET JAVA_BIN=%JAVA_HOME%\jre\bin\java.exe
```

```
if exist "%JAVA_BIN%" goto configuredjavabin
SET JAVA_BIN=%JAVA_HOME%\bin\java.exe

:configuredjavabin

if exist "%JAVA_BIN%" (
  if "%1"=="start" goto :doStart
  if "%1"=="stop" goto :doStop
  if "%1"=="install" goto :doInstall
  if "%1"=="remove" goto :doRemove
  if "%1"=="state" goto :doInfo

goto :usage
)

echo "Unable to execute %0: Environment variable JAVA_HOME must be set"

goto :end

:doStart
"%JAVA_BIN%" -DDENODO_APP="Denodo Monitor" -jar "%DENODO_HOME%/lib/service-
wrapper/wrapper.jar" -t "%MONITOR_HOME%/conf/service.conf"
goto :end

:doStop
"%JAVA_BIN%" -jar "%DENODO_HOME%/lib/service-wrapper/wrapper.jar" -p
"%MONITOR_HOME%/conf/service.conf"
goto :end

:doInstall
"%JAVA_BIN%" -jar "%DENODO_HOME%/lib/service-wrapper/wrapper.jar" -i
"%MONITOR_HOME%/conf/service.conf"
goto :end

:doRemove
"%JAVA_BIN%" -jar "%DENODO_HOME%/lib/service-wrapper/wrapper.jar" -r
"%MONITOR_HOME%/conf/service.conf"
goto :end

:doInfo
"%JAVA_BIN%" -jar "%DENODO_HOME%/lib/service-wrapper/wrapper.jar" -q
"%MONITOR_HOME%/conf/service.conf"
goto :end

:exit
```

```
echo "Unable to execute %0: Environment variables JAVA_HOME and DENODO_HOME
must be set (absolute path)"
goto :end

:usage
echo "Usage: %0 [start|stop|install|remove|state]"

:end
```

Use the following template for the configuration file (service.conf) replacing <DENODO\_HOME> with the appropriate value.

```
wrapper.java.command=<DENODO_HOME>/jre/bin/java.exe
wrapper.working.dir=<DENODO_HOME>/tools/monitor/denodo-monitor

wrapper.java.app.mainclass=com.denodo.util.launcher.Launcher
wrapper.java.classpath.1=<DENODO_HOME>/lib/denodo-commons-launcher-util.jar

wrapper.java.additional.1=-DconfFile=ConfigurationParameters.properties
wrapper.java.additional.2=-DverboseMode=false
wrapper.java.additional.3=-
Djavax.xml.xpath.XPathFactory\:http\://java.sun.com/jaxp/xpath/dom=org.apach
e.xpath.jaxp.XPathFactoryImpl
wrapper.java.additional.4=-
Djavax.xml.transform.TransformerFactory=org.apache.xalan.processor.Transform
erFactoryImpl
wrapper.java.additional.5=-
Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryI
mpl
wrapper.java.additional.6=-Ddenodo.rmi.server.hostname.default
wrapper.java.additional.7=-DWINDOWS_SERVICE=true

wrapper.java.initmemory=3

wrapper.app.parameter.1=com.denodo.monitor.DiagnosticTool
wrapper.app.parameter.2=--lib
wrapper.app.parameter.3=<DENODO_HOME>\\tools\\monitor\\denodo-monitor\\lib
wrapper.app.parameter.4=--conf
wrapper.app.parameter.5=<DENODO_HOME>\\tools\\monitor\\denodo-monitor\\conf
wrapper.app.parameter.6=--arg
wrapper.app.parameter.7=-d
wrapper.app.parameter.8=--arg
wrapper.app.parameter.9=<DENODO_HOME>\\tools\\monitor\\denodo-monitor\\logs
wrapper.app.parameter.10=--arg
wrapper.app.parameter.11=%1
wrapper.app.parameter.12=--arg
wrapper.app.parameter.13=%2
wrapper.app.parameter.14=--arg
wrapper.app.parameter.15=%3
```

```
wrapper.logfile=<DENODO_HOME>\\tools\\monitor\\denodo-monitor\\logs
wrapper.logfile.loglevel=ERROR
wrapper.logfile.format=LPTM
wrapper.logfile.maxsize=10m
wrapper.logfile.maxfiles=10
wrapper.syslog.loglevel=ERROR

wrapper.ntservice.name=Denodo Monitor
wrapper.ntservice.displayname=Denodo Monitor 7.0
wrapper.ntservice.description=Denodo Monitor - Denodo Platform 7.0
wrapper.ntservice.dependency.1=
wrapper.ntservice.starttype=DEMAND_START
wrapper.ntservice.interactive=false

#wrapper.affinity=
wrapper.java.maxmemory=1024
```

Any other configuration parameters different from the default configuration must also be modified in this file.

After preparing you can execute the script to register the Denodo Monitor as a Windows service (**note:** execute this command with Administrator privileges):

```
denodomonitor_service.bat install
```

If the service is correctly deployed the following message is shown: "Service Denodo Monitor installed".

To delete the Windows service the "remove" option can be used. To start and stop the service from the command line you can use the "start" and "stop" and the "state" option to check the status of the service.

## References

[Configuring and starting the Denodo Monitor](#)  
[Launching the Denodo Monitor with Solution Manager](#)