DENODO PLATFORM 5.0 INSTALLATION GUIDE

Update Feb 19th, 2015
NOTE

This document is confidential and is the property of Denodo Technologies.

No part of this document may be reproduced in any form by any means without prior written authorization from Denodo Technologies.
CONTENTS

PREFACE .................................................................................................................... I

SCOPE I

WHO SHOULD USE THIS DOCUMENT .................................................................... I

SUMMARY OF CONTENTS ................................................................................................. I

1 INTRODUCTION ........................................................................................................ 1

2 INSTALLATION ........................................................................................................ 3

2.1 HARDWARE REQUIREMENTS .............................................................................. 3

2.2 SOFTWARE REQUIREMENTS .............................................................................. 3

2.2.1 Software Requirements for Virtual DataPort .................................................. 4

2.2.2 Software Requirements for ITPilot ................................................................. 4

2.2.3 Software Requirements for Aracne ................................................................. 5

2.2.4 Pre-installation Actions in Windows Vista/Windows Server 2008/Windows 7 .................................................................................................................. 5

2.3 USING THE INSTALLATION WIZARD .............................................................. 6

2.3.1 Components and Configuration of Virtual DataPort .................................... 10

2.3.2 Components and Configuration of ITPilot ................................................. 11

2.3.3 Aracne ........................................................................................................... 14

2.3.4 Scheduler ...................................................................................................... 15

3 POST-INSTALLATION TASKS ................................................................................. 17

3.1 LICENSE INSTALLATION .................................................................................. 17

3.2 ENABLE SSL CONNECTIONS IN THE DENODO PLATFORM SERVERS ......... 17

3.2.1 Obtaining and Installing an SSL Certificate ................................................ 18

3.2.2 Enabling SSL in Denodo Platform Servers .............................................. 18

3.2.3 Enabling SSL in Denodo Platform Tools .................................................... 20

3.2.4 Enabling HTTPS in the Embedded Apache Tomcat .................................. 20

3.2.5 Enabling SSL for External Clients ............................................................. 21

3.3 POST-INSTALLATION TASKS IN VIRTUAL DATAPORT ............................. 22

3.3.1 External CLASSPATH .................................................................................. 22

3.3.2 Check Windows Service Configuration ...................................................... 22

3.3.3 Installing the SAP JCo Connector ............................................................... 22

3.3.4 Installing the Connector for SAP BW and SAP BI (Multidimensional Sources) .............................................................................................................. 25

3.3.5 Installing the Connector for Oracle Essbase .............................................. 26

3.3.6 Installing the JMS Connectors to Create JMS Listeners and Web services with SOAP over JMS .......................................................... 26

3.3.7 Configuring the VCS Clients ...................................................................... 27

3.3.8 Importing the Certificates of Data Sources (SSL Connections) .............. 30

3.3.9 Increasing the Maximum Simultaneous Requests ...................................... 31

3.4 POST-INSTALLATION TASKS IN ITPILOT .................................................... 32

3.4.1 Checking that the Generation Environment Has Been Installed ........... 32

3.4.2 Configuration of Microsoft Internet Explorer ............................................ 32

3.4.3 Enabling Internet Explorer Sequence Generation Toolbar in Microsoft Windows 2003 and Windows 2008 ........................................... 34

3.4.4 Disabling Internet Explorer Enhanced Security Configuration in Microsoft Windows 2008 .......................................................... 34

3.4.5 Manual Configuration of Adobe Acrobat Professional ................................ 35

3.4.6 Automatic Verification Database ............................................................... 35

3.5 POST-INSTALLATION TASKS IN ARACNE AND SCHEDULER ........... 35

3.5.1 Microsoft Internet Explorer Configuration ............................................... 35
3.5.2 Configuring the Exchange Crawler Server ................................................. 37
3.6 DEPLOYING THE WEB ADMINISTRATION TOOL IN AN EXTERNAL WEB CONTAINER ........................................................................................................... 37

4 UPGRADE TO DENODO PLATFORM 5.0 .......................................................... 39

5 UNINSTALLATION ............................................................................................... 40

6 DENODO4E ....................................................................................................... 41

7 DENODO PLATFORM CONTROL CENTER ....................................................... 42

7.1 EXECUTION ..................................................................................................... 42

7.2 CONTROL CENTER HELP ............................................................................ 42

7.2.1 Online Help Configuration ........................................................................ 44

7.3 STARTING PLATFORM SERVERS AND TOOLS .......................................... 45

7.3.1 Configuration of Custom Environments ..................................................... 47

7.4 INSTALLING AN UPDATE OR A HOTFIX .................................................. 49

7.5 OTHER FUNCTIONS ...................................................................................... 49

7.5.1 Virtual Machine Parameters and Web Container Configuration .............. 49

7.5.2 Uninstalling the Platform ......................................................................... 50

8 INSTALLATION AND CONFIGURATION IN SILENT MODE .................. 52

8.1 SILENT INSTALLATION OF THE DENODO PLATFORM ......................... 52

8.2 SILENT INSTALLATION OF UPDATES ...................................................... 53

8.3 CONFIGURATION OF THE JVM PARAMETERS IN SILENT MODE .......... 54

9 APPENDIX ......................................................................................................... 56

9.1 DEFAULT PORTS USED BY THE DENODO PLATFORM MODULES ... 56

REFERENCES ...................................................................................................... 58
LIST OF FIGURES

Figure 1 General architecture of the Denodo Platform ........................................ 2
Figure 2 Administrator credentials request dialog .................................................. 6
Figure 3 Installation method configuration ............................................................ 7
Figure 4 Choosing the internationalization configuration ...................................... 9
Figure 5 Summary screen ....................................................................................... 10
Figure 6 Selection of modules and components .................................................... 11
Figure 7 SAP Java Connector (JCo) test screen .................................................... 24
Figure 8 Configuring the local repository for Denodo 4E ....................................... 41
Figure 9 Denodo Platform Control Center Help screen ......................................... 43
Figure 10 Information on the current license ........................................................... 44
Figure 11 Online help configuration dialog ............................................................ 45
Figure 12 Denodo Platform Control Center ............................................................ 46
Figure 13 Creation of custom environments ........................................................... 48
Figure 14 Creation and editing of custom environments ......................................... 48
Figure 15 Uninstalling the Denodo Platform .......................................................... 51
Figure 16 Processes running warning .................................................................... 51
PREFACE

SCOPE
The Denodo Platform provides business applications with easy access to integrated views of various heterogeneous and distributed, both structured and non-structured, data sources. The Platform includes modules for real-time access and integration of all types of information, including that contained in databases, Web Services, hidden Web sites, and document repositories. This document introduces readers to the installation and configuration of the Platform and the use of the Denodo Platform Control Center tool that, among other uses, provides a single panel from where all the servers and tools comprising the Platform can be started and stopped.

WHO SHOULD USE THIS DOCUMENT
This document is aimed at developers and administrators requiring detailed information on how to install and configure the Denodo Platform and how to use the Denodo Platform Control Center tool.

SUMMARY OF CONTENTS
More specifically, this document:

- Briefly introduces the Denodo Platform.
- Describes the processes required to install each of the Platform modules.
- Describes the use of the Denodo Platform Control Center tool that, among other uses, provides a single panel from where all the servers and tools comprising the Platform can be started and stopped.
1 INTRODUCTION

Modern business processes must make use of a vast array of information systems, developed over several decades.

These systems, in turn, make use of a vast array of data sources, developed for the most part using completely disparate technologies (relational databases, Web services, XML documents, spreadsheets, flat files …) and very dissimilar data models. Some of these data sources are "well-structured" for ease of querying from client applications; more and more are less well structured and lack even rudimentary schema for facilitating client interactions. Some, in fact, are pure text documents, which make no pretense at adhering to a "structure" of any kind.

Development of modern information systems and optimization of business processes require access and utilization of the data stored in this vast array of inherited data repositories.

Another important trend to consider is the expansion of Social Media and Big Data, an obvious example of the widespread dissemination of needed information and the heterogeneity of data formats. This data, typically found through browsing the Web in its many forms, often adds crucial value to corporate information systems.

In addition, clients, suppliers, and competitors often use simple or complex Web interfaces for accessing their information and services.

The Denodo Platform is a global solution for the real-time integration of heterogeneous and dispersed, structured and non-structured data sources. To do so, it combines different integrated modules (see Figure 1):

- The Virtual DataPort module provides real-time integration of any relevant information for the company, whatever its origin, format, and structure level. It enables to create "virtual databases" that are made up of unified views of data from any type of data source, including databases, SOAP, and REST Web Services, XML, flat files, conventional Web sites (through the ITPilot module), LDAP servers, and non-structured information indexes (through the Aracne module).

- The ITPilot module provides an easy way to access and structure the data present in the Web. This process involves building an abstraction from the specific Web source called wrapper that isolates the client applications from the intrinsic characteristics of that site (access protocol, native data structure, etc.). Thus, the data contained on the hidden Web can be queried and integrated just as easily and effectively as the data contained in a conventional database. In addition to that, ITPilot can also extract structured data from PDF, MS Excel and MS Word documents. ITPilot provides a distributed and scalable environment for generating, executing, and verifying the correct operation of wrappers.

- The Aracne module facilitates crawling, indexing, filtering, and querying of non-structured data in a wide range of repositories and formats. Aracne can integrate information contained in repositories such as the Web, file systems, relational databases, or e-mail servers.

- In addition, the Scheduler module is a complementary tool that allows scheduling batch tasks using any of the former modules to automatically
extract and filter data from several sources and export them to the desired format.

![Diagram of Denodo Platform architecture](image)

**Figure 1** General architecture of the Denodo Platform

This document introduces readers to the installation and configuration of the Platform and the use of the Denodo Platform Control Center tool that, among other uses, provides a single panel from where all the servers and tools comprising the Platform can be started and stopped.

See the Virtual DataPort Administration Guide [VDP], the ITPilot User Guide [ITPILOT], the Aracne Administration Guide [ARCN] and the Scheduler Administration Guide [SCH] for information about how to use each module.
2 INSTALLATION

This section focuses on the installation and configuration of the different modules of the Denodo Platform:

- Sections 2.1 and 2.2 describe the hardware and software requirements to install the Denodo Platform.
- Section 2.3 describes how to launch the installation tool.
- Section 3 describes the required post-installation tasks.
- Section 4 describes how to upgrade from previous versions of the Platform.
- Section 5 describes how to uninstall the platform.
- Section 7 describes how to launch and use the Denodo Platform Control Center.

2.1 HARDWARE REQUIREMENTS

The minimum system requirements are a 2 GHz processor (32-bit or 64-bit), 4 GB of RAM, 5 GB available hard disk and Internet access.

You have to take into account that processor performance depends on the clock frequency of the processor, but also on the number of processor cores and the size of the processor cache.

2.2 SOFTWARE REQUIREMENTS

Before installing the Denodo Platform, you need to install the following:

- An operating system with Java support. The Denodo Platform runs on any of the platforms listed below:
    - If you are installing the Denodo Platform on Windows 2008, Vista or 7, see the section 2.2.4 before continuing with the installation process.
    - If you are going to install ITPilot in Windows, check the Post-Installation tasks for ITPilot (section 3.4).
  - Solaris
  - Linux: Ubuntu 7.1, 8.4 [UBU], CentOS 5 and 6 [CENT].
- Denodo provides a Denodo Platform installer for several environments: 32-bit Windows, 64-bit Windows, 32-bit Linux and 64-bit Linux. These installers include the appropriate Java Runtime Environment (JRE).
If you are installing the Denodo Platform in an environment other than these ones, you have to install a Java Runtime Environment version 1.6 or 1.7.

For an updated list of the supported platforms, contact the Denodo Support Team.

Besides these requirements, each module of the Denodo Platform has additional requirements that must be met in order to use some of their components:

- Additional requirements of Virtual DataPort: see section 2.2.1.
- Additional requirements of ITPilot: see section 2.2.2.
- Additional requirements of Aracne: see section 2.2.3.

2.2.1 Software Requirements for Virtual DataPort

This section lists the additional requirements of Virtual DataPort.

2.2.1.1 Microsoft Excel and Access Sources

To extract data from Microsoft Excel and Access files, you need to install the Excel and Access ODBC drivers. To obtain them, do one of the following:

- Install Microsoft Office in the same machine as the Virtual DataPort Server.

2.2.1.2 Cache

Virtual DataPort incorporates a cache module that stores local copies of the data retrieved from the data sources, in a JDBC database. This may reduce the impact of repeated queries hitting the data source and speed up data retrieval, especially with certain type of sources.

Install an external Database Management Systems (DBMSs) if you are going to cache data. Although Virtual DataPort embeds an Apache Derby [DERBY] database that can be used to store the cache data, we do not recommend using this database on a production environment.

The section ‘Cache Module’ of the Administration Guide [VDP] explains how the cache module works and lists the DBMSs that Virtual DataPort can use to store the cached data.

2.2.2 Software Requirements for ITPilot

This section lists the additional requirements of ITPilot:

- The use of the automatic Browser Pool, install Microsoft Internet Explorer 8.x, 9.x, 10.x or 11.x [MSIE].
- To extract data from PDF files using the Adobe PDF engine, ITPilot has to run on Windows with Adobe Acrobat Professional 7, 8, 9 or 10.
To extract data from Microsoft Word or Microsoft Excel files, install OpenOffice 3.0, 3.1, 3.2 or 3.3 [OOF].

The Verification Server uses a relational database to store information about wrappers execution to allow verifying its correct operation. ITPilot provides an embedded Apache Derby [DERBY] database that can be used for this purpose. An external JDBC database management system can also be used. This feature has been tested with MySQL [MYSQL] and Oracle [ORCL].

ITPilot provides a script to create tables for these DB Management Systems (consult the Post-Installation tasks for more details about these scripts).

2.2.3 Software Requirements for Aracne

This section lists the additional requirements of Aracne:

- To use Aracne with IECrawler or ExchangeMailCrawler, Aracne has to run on Windows with Microsoft Internet Explorer 8.x [MSIE].

- Microsoft Exchange Server MAPI Client and Collaboration Data Objects [MAPI] must be installed on the machine in which the Denodo Exchange Crawler Server is going to be installed to use the Denodo Aracne ExchangeMailCrawler component (the crawler for contents saved in MSExchange), if Microsoft Exchange Server and the Denodo Exchange Crawler Server are installed in different machines.

2.2.4 Pre-installation Actions in Windows Vista/Windows Server 2008/Windows 7

The installation must be executed with administrator privileges. If you are not logged in as an administrator and you have User Account Control turned on, administrator credentials will be requested before the installation process begins (see Figure 2).
2.3 USING THE INSTALLATION WIZARD

Select a Denodo Platform installer. Denodo provides several flavors of the installer and the only difference between them is the Java Runtime Environment (JRE) they include:

- `denodo-install-5.0-linux32.zip`: it includes a 32-bit JRE for Linux.
- `denodo-install-5.0-linux64.zip`: it includes a 64-bit JRE for Linux.
- `denodo-install-5.0-win32.zip`: it includes a 32-bit JRE for Windows.
- `denodo-install-5.0-win64.zip`: it includes a 64-bit JRE for Windows.
- `denodo-install-5.0.zip`: it does not include any JRE. Use this installer if the previous installers are unsuited for your environment.

Before launching this installer, you have to set the environment variable `JAVA_HOME` to point to a JRE version 6 or 7. Otherwise, the installer process will fail.

The modules forming part of the Denodo Platform are installed through a graphical installer. The installation wizard will guide you through each of the necessary configuration steps. In addition, you can always use the `Help` button of the wizard if you need assistance during the installation process.

After the introduction screen, the installation process will display the product licensing conditions, which you should read carefully. If you accept the conditions, you will see the screen shown in Figure 3.

Here you can choose between two different installation methods: `Default Installation` (the default option, recommended for most users) and `Custom`.
Installation (recommended for advanced users). By choosing Default Installation, default values will be given to all the configuration parameters needed by the installer. On the other hand, by choosing Custom Installation, you will be able to specify the values of all the required configuration parameters in the next steps of the installation process.

In this screen, the path in which the software is to be installed must also be specified (hereinafter DENODO_HOME). The installer will copy the required files in this directory.

**Warning:** If you are installing the Denodo Platform in Windows Vista, Windows 2008 or Windows 7, the installation path cannot include “junctions” [JUN].

**Warning:** close all the Microsoft Internet Explorer instances before installing the Denodo Platform. Some ITPIlot modules need the browser to be closed in order to be installed correctly.

**IMPORTANT NOTE:** When the Denodo Platform is installed in a path where there is a previous installation, the existing metadata will be overwritten.

Finally, you can provide the location of your license file. This is an optional step, but if you do not provide the license, you will need to proceed as specified in section 3.1 after the installation process is complete.

![Installation method configuration](image)

**Figure 3** Installation method configuration

The next screen displays the different modules that can be installed and the components contained in each module (see Figure 6). To choose a module, simply click on the check box alongside it. Later subsections will briefly describe the configuration parameters for each module, which can be specified during installation if the Custom Installation option was selected.
The next screen will request internationalization configuration for the Denodo Platform (see Figure 4).

Then, the installed components will be initially configured if the Custom Installation option was selected. The subsections 2.3.1, 2.3.2, 2.3.3 and 2.3.4 explain how to configure the Virtual DataPort, ITPilot, Aracne and Scheduler components, respectively.

The default Denodo Platform installation also includes a J2EE Web service container used to execute the web administration tools from some of the modules and to run the Web Services published by the Platform. To configure this module, if the Custom Installation option was selected, the following information can be provided after the initial configuration of the remaining modules has been completed:

- Port number that the Web service container will use for listening to requests.
- Port number used to stop the Web service container.
- Port number for management purposes. The Web service container uses an additional port number for management purposes.

Before the installer begins copying the necessary files (this process may take several minutes), a screen with a summary of the installation configuration will be shown (see Figure 5 for an example). You can save the summary in plain text or HTML by clicking the Save button. At all times a progress bar will inform you of the progress of the copying process, both global and per module. If you interrupt the installation process while the files are still being copied, you will be given the option to delete the selected installation directory.

Finally, the installation process can create shortcuts in the menus of the operating system to launch the Denodo Platform Control Center (see Section 7.1).
Figure 4  Choosing the internationalization configuration
2.3.1 Components and Configuration of Virtual DataPort

The following components may be installed:

- Administration Tool. The graphical administration tool of Virtual DataPort as well as the classes required to develop applications that run queries with Virtual DataPort (see the Virtual DataPort Administration Guide [VDP]).

- Virtual DataPort Server. The Server that stores the metadata of objects such as data sources, views, etc. and holds the embedded Web container.

Once the components have been selected, if the Custom Installation option was selected, the system starts their configuration process. In the case of DataPort, the following configuration parameters can be specified:

- Server port number: port that the DataPort server will use to listen for requests.

- Shutdown port number: used by the DataPort server to listen for shutdown requests.

- Auxiliary port Number: communications between clients and the DataPort server use an additional port number known as “auxiliary” that can be specified at this time.
- ODBC port number: port that the DataPort server will use to listen for ODBC requests.

- In Windows operating systems, it is possible to install the DataPort server as a Windows service, by checking the option “Install Windows service”.

NOTE: If a firewall software is used to control the traffic between the clients and the server, it must be configured to allow communication using these ports.

Figure 6 Selection of modules and components

2.3.2 Components and Configuration of ITPilot

IMPORTANT NOTE: Before installing ITPilot you must close all instances of Microsoft Internet Explorer running.

In the case of ITPilot, the following components may be installed:

- Navigation Sequence Generator. Used to graphically generate automated Web browsing sequences during the creation of wrappers on Web sources. It is installed as a toolbar in the MS Internet Explorer browser. See the ITPilot Generation Environment Guide [GENITP].

- Wrapper Generator Tool. Used to create wrappers on Web sources. See the ITPilot Generation Environment Guide [GENITP].

- Wrapper Server. Used to run the wrappers created with previous components. See the ITPilot User Guide [ITPILOT].
Denodo Platform 5.0

Installation Guide

• Browser Pool. Wrapper execution environment component required to run automated browsing sequences based on MS Internet Explorer, and that can also use the Denodo browser. See the ITPilot User Guide [ITPILOT].

• Verification Server. Used to automatically verify if ITPilot wrappers continue running properly when changes to the Web sources are found. See the ITPilot User Guide [ITPILOT].

• Administration Console. Web administration console for the wrapper execution environment. See the ITPilot User Guide [ITPILOT].

• Wrapper Client Environment. Used to program applications directly using ITPilot wrappers. See ITPilot Developer Guide [DEVITP].

If the Custom Installation option was selected, any components requiring configuration will request it during the installation process. The following subsections describe the configuration of each of the components requiring configuration.

2.3.2.1 Initial Configuration of the Browser Pool

The following options can be configured for the browser pool:

• Server port number: the port that the Browser Pool will use to listen for requests.

• Shutdown port number: port number used to notify the Browser Pool to shutdown.

• Auxiliary port number: auxiliary port used by the Browser Pool for communications with its clients.

• Initial browser port: port used by ITPilot to communicate with the first opened browser. Consecutive ascending port numbers will be used when additional browsers are requested.

• It is possible to specify that the Browser Pool be installed as a service in Windows operating systems, by checking the “Install Windows service” option.

NOTE: If a firewall software is used to control the traffic between the clients and the server, it must be configured to allow communication using these three ports.

The system must also be informed of the path where the external applications required for certain components are installed:

• Path where Acrobat Professional is installed, in case information is going to be extracted from PDF documents using the Adobe Professional software.

• Path where OpenOffice is installed, if information is going to be extracted from MS Word or MS Excel documents.
2.3.2.2 Wrapper Server

The following settings can be configured for the wrapper server (if the Virtual DataPort Server is also being installed, its settings will be shared with the ITPilot Wrapper Server):

- **Server port number**: port that the ITPilot Wrapper server will use to listen for requests.
- **Shutdown port number**: used by the ITPilot Wrapper server to listen for shutdown requests.
- **Auxiliary port number**: communications between clients and the ITPilot Wrapper server use an additional port number known as “auxiliary” that can be specified at this time.
- In Windows operating systems, it is possible to install the ITPilot Wrapper server as a Windows service, by checking the option “Install Windows service”.

**NOTE**: If a firewall software is used to control the traffic between the clients and the server, it must be configured to allow communication using these ports.

If ITPilot is being installed in the same path where Denodo Virtual DataPort had been previously installed, the settings of the Virtual DataPort wrapper server will be overwritten with the new values.

Furthermore, if a browser pool is not being installed in the same path, the connection data to be used by the wrapper server must be configured:

- **Browser pool IP address**.
- **Browser pool listening port**.

In the event of also installing a browser pool, the values indicated for the pool during installation will be used as values for these parameters.

2.3.2.3 Verification Server

The following settings can be configured for the Verification server:

- **Server port number**: port that the ITPilot Verification server will use to listen for requests.
- **Shutdown port number**: used by the ITPilot Verification server to listen for shutdown requests.
- **Auxiliary port**: auxiliary port used by the Verification server for communications with its clients.
- In Windows operating systems, it is possible to install the ITPilot Verification server as a Windows service, by checking the option “Install Windows service”.

Furthermore, if a wrapper server is not being installed in the same installation, the wrapper server connection data must be configured:
• Wrapper server IP address.

• Wrapper server listening port.

In the event of also installing a wrapper server, the values indicated for this server during installation will be used as values for these parameters.

**NOTE:** If a firewall software is used to control the traffic between the clients and the server, it must be configured to allow communication using these ports.

### 2.3.3 Aracne

In the case of Aracne, the following components can be installed (see Aracne Administration Guide [ARCN] for further information):

• **Aracne Server.** This includes the crawling modules (except the IECrawler and the Exchange Crawler).

• **Indexing/Search Engine Server.** This includes the indexing and data search servers. If required, these servers can be installed in a different machine to that of the Aracne server.

• **Exchange Crawler Server.** Microsoft Exchange Server e-mail accounts crawler. It must be installed in Windows machines belonging to the same domain as the machine on which Microsoft Exchange Server is executed. It is recommended to install it in the same machine as the Exchange e-mail server.

• **IECrawler.** Advanced Web crawler based on MS Internet Explorer.

• **Administration console.** Web administration console that allows configuring the crawling, filtering, and data indexing tasks. It also allows for queries on the collected data after running the crawling/indexing tasks.

If the *Custom Installation* option was selected, any components requiring configuration must be configured initially during the installation process. In the case of Aracne, the following will be requested:

#### 2.3.3.1 Aracne Server

The following settings can be configured for the Aracne server:

• **Server port number:** port that the Aracne server will use to listen for requests.

• **Shutdown port number:** port number used to stop the server.

• **Auxiliary port number:** Communications between clients and the Denodo Aracne administration server use an additional port number known as "auxiliary" that can be specified at this time.

• **In Windows operating systems, it is possible to install the Aracne server as a Windows service,** by checking the option "Install Windows service".

**NOTE:** If a firewall software is used to control the traffic between the clients and the server, it must be configured to allow communication using these ports.
2.3.3.2 Indexing/Search Engine Server

The following can be configured for the indexing/search module:

- Server port number: Port numbers at which the indexing server and the search server will be launched, respectively.
- Shutdown port number: Port numbers to be used to stop the indexing server and the search server, respectively.
- Auxiliary port number of the indexing server and the search server, respectively. Communications between clients and the Denodo Aracne indexing and query servers use an additional port number known as “auxiliary” that can be specified at this time.
- It is possible to specify for the Aracne Indexing/Search Engine server to be installed as a service in Windows operating systems, by checking the option “Install Windows service”.

**NOTE:** If a firewall software is used to control the traffic between the clients and the server, it must be configured to allow communication using these ports.

2.3.3.3 Exchange Crawler Server

The following can be configured for the Exchange Crawler module:

- Exchange Crawler Server port number: port that the Exchange Crawler server will use to listen for requests launched.
- It is possible to specify for the Aracne Exchange Crawler server to be installed as a service in Windows operating systems, by checking the option “Install Windows service”.

**NOTE:** If a firewall is used to control the traffic between the clients and the server, it must be configured to allow communication using these ports.

2.3.4 Scheduler

In the case of the Scheduler module, the following components can be installed (see Scheduler Administration Guide [SCH] for more information):

- Server. The server for scheduling and configuring batch tasks.
- Administration Console. Web administration tool that allows configuring, scheduling and executing batch data extraction tasks.

During the installation process, if the *Custom Installation* option was selected, it is needed to configure the following parameters for the server module:

- Server port number: that the Scheduler server will use to listen for requests.
- Shutdown port number: used by the Scheduler server to listen for shutdown requests.
- Auxiliary Port number: communications between clients and the Scheduler server use an additional port number known as “auxiliary” that can be specified at this time.

- In Windows operating systems, it is possible to install the Scheduler server as a Windows service, by checking the option “Install Windows service”.

**NOTE:** If a firewall software is used to control the traffic between the clients and the server, it must be configured to allow communication using these ports.
3 POST-INSTALLATION TASKS

After the installer finishes its execution, there are still several tasks required before the software can be run. First, the Platform license file must be installed if it was not provided to the installation wizard. Section 3.1 describes how this is done. Section 3.2 describes the steps to enable SSL for the communications between the Platform modules. If this is not required, that section can be skipped. Subsequently, Sections 3.3, 3.4, and 3.5 explain the post-installation steps for the Virtual DataPort, ITPilot, and Aracne and Scheduler modules, respectively.

NOTE: some computer security suites in their default configurations can make the Denodo Platform function incorrectly. If after the installation you cannot create new browsers from the ITPilot Wrapper Generation tool, create browser instances at runtime, or use the MSIE ITPilot toolbar, check that your security software is not blocking any Denodo application or any port used by them.

3.1 LICENSE INSTALLATION

If you did not provide a license file during the installation process or you want to use a different license than the one selected during the installation process, do one of the following:

1. Install the license file using the Denodo Platform Control Center (see section 7.2)
2. Or, rename the desired license file as denodo.lic and place it in the <DENODO_HOME>/conf directory.

Without a valid license file, the Denodo Platform applications will not start properly.

3.2 ENABLE SSL CONNECTIONS IN THE DENODO PLATFORM SERVERS

This section explains how to secure with SSL the connections between the Denodo Platform servers, their administration tools and their clients. If you do not need this, jump to the section 3.3. To know how to establish SSL connections with data sources, go to section 3.3.8.

Note that if you enable SSL in the Denodo Platform servers, you also have to do it in their clients.

SSL requires configuring certificate repositories. There are two types of certificate repositories:

- KeyStore
- TrustStore

**KeyStore**

An application that listens to incoming SSL connections needs a public key and a private key in order to allow clients to access the server. In Java, these keys are stored in a repository called **KeyStore**.
TrustStore
During the initialization of an SSL connection, the server sends its SSL certificate to the client. The client must then decide if it trusts this or not. To do this, the client checks if the certificate has been signed by a trusted certification authority (CA). The TrustStore is a repository of the certificates of trusted certification authorities.

Every Java installation comes with a TrustStore that the JRE uses by default (/jre/lib/security cacerts file). If the server’s certificate is not signed by a trusted authority (i.e. one that is not registered in the Java’s TrustStore), you have to store the certificate of the authority, which can be stored in:

1. The cacerts file of the JRE used to launch the Denodo Platform servers and their tools ($DENODO_HOME/jre/lib/security/cacerts file). This is the recommended option because adding the certificate of the authority to this file will make the configuration of the Denodo Platform servers easier.

2. Or, in a new TrustStore.

Oracle’s Java Development Kit (JDK) ships with a utility called keytool that manages the Certificate Repositories.

3.2.1 Obtaining and Installing an SSL Certificate
Before enabling SSL in the Denodo Platform servers and clients, you have to do the following:

1. If you already have a certificate signed by a Certification Authority (CA), create a KeyStore with it, using the keytool application provided by the JDK.

   If the certificate is obtained from a CA that is not trusted by the JRE, import the certificate of the CA into the TrustStore of the JRE before adding it to the server’s KeyStore.
   See [JRE-CERT] for a list of the CA certificates included in the JRE.

2. If you do not have a certificate, you can generate a self-signed one, using keytool.

3. If you are going to use a self-signed certificate, generate a certificate from its public key and add it to the TrustStore of the Denodo Platform JRE ($DENODO_HOME/jre/lib/security/cacerts).

The following sections explain how to configure the Denodo Platform servers and their clients so they use the appropriate KeyStore and TrustStore. Note that many servers of the Denodo Platform act as clients as well. For example, Virtual DataPort is a client of Aracne when retrieving data from Aracne. That is why you have to configure both repositories for each server.

3.2.2 Enabling SSL in Denodo Platform Servers
Follow these steps to secure incoming connections in the Denodo Platform servers with SSL:
1. Open the configuration files of the servers whose connections have to be secured:
   - Aracne server: 
     $DENODO_HOME/conf/arn/ConfigurationParameters.properties
   - Aracne Index server:
     $DENODO_HOME/conf/arn-index/ConfigurationParameters.properties
   - ITPilot Browser Pool:
     $DENODO_HOME/conf/iebrowser/IEBrowserConfiguration.properties
   - ITPilot Verification server:
     $DENODO_HOME/conf/maintenance/ConfigurationParameters.properties
   - Scheduler server:
     $DENODO_HOME/conf/scheduler/ConfigurationParameters.properties
   - Virtual DataPort server:
     $DENODO_HOME/conf/vdp/VDBConfiguration.properties
   - Embedded Apache Tomcat:
     $DENODO_HOME/resources/apache-tomcat/conf/tomcat.properties

2. In all the files opened in the previous step, uncomment the following properties and change their values:
   - com.denodo.security.ssl.enabled=true
   - com.denodo.security.ssl.keyStore=Path to the KeyStore that contains the certificate of the Denodo Platform servers.
   - com.denodo.security.ssl.keyStorePassword=Password of the KeyStore containing the certificate of the Denodo Platform servers.
   - If the certificate is not signed by a trusted authority (i.e. one that is not registered in the JRE's TrustStore), you have to store the certificate of the authority that signed the certificate. If, instead of importing it into the $DENODO_HOME/jre/lib/security/cacerts TrustStore, you have created a new TrustStore, uncomment the property com.denodo.security.ssl.trustStore and set it to the path of the new TrustStore.

All the Denodo servers might need the TrustStore because some of them act both as clients and as servers. E.g., Virtual DataPort server receives connections from its Administration Tool, but also establishes connections with the Aracne server.
3.2.3 Enabling SSL in Denodo Platform Tools

When SSL is enabled in a Denodo Platform server, all their clients have to trust the public key of the server. This includes the administration tools of the Denodo Platform.

If, instead of importing it into the $DENODO_HOME/jre/lib/security/cacerts TrustStore, you have created a new TrustStore, do the following. Otherwise, jump to the next section.

1. Open the following files:
   - IT Pilot Administration Tool: $DENODO_HOME/conf/itp-admin-tool/ITPAdminConfiguration.properties
   - $DENODO_HOME/conf/itpilot-client/ConfigurationParameters.properties
   - Virtual DataPort Administration Tool: $DENODO_HOME/conf/vdp-admin/VDBAdminConfiguration.properties
   - Denodo Monitor Tool: $DENODO_HOME/tools/monitor/denodo-monitor/conf/ConfigurationParameters.properties

2. In the files opened in the previous step, uncomment the following property and change its value:
   - com.denodo.security.ssl.trustStore=path to the new TrustStore.

The scripts of the Denodo Tools do not have a configuration file. To redefine the default TrustStore that they use, you have to define the javax.net.ssl.trustStore Java system property. For example:

- For Windows:
  ```
  SET JAVA_OPTS=
  -Djavax.net.ssl.trustStore=%DENODO_HOME%/jre/lib/security/cacerts
  ```

- For Unix:
  ```
  export JAVA_OPTS=
  -Djavax.net.ssl.trustStore=$DENODO_HOME/jre/lib/security/cacerts
  ```

3.2.4 Enabling HTTPS in the Embedded Apache Tomcat

The Denodo Platform embeds the Apache Tomcat Web container that is used, among other things, to deploy some of the Denodo Platform administration tools.

The communications between clients and the web applications running in the Apache Tomcat embedded in the Denodo Platform can be secured with HTTPS. The applications running in this Tomcat are:

- Aracne Administration Tool
- ITPilot Administration Tool
Post-installation Tasks

3.2.5 Enabling SSL for External Clients

3.2.5.1 JDBC and other Java Clients

To secure the communication between Denodo servers and their JDBC clients or other Java clients, set the Java system property `javax.net.ssl.trustStore` to point to the TrustStore that contains the certificate used by the Denodo servers. For example:

- For Windows:
  ```
  SET JAVA_OPTS=-Djavax.net.ssl.trustStore=%DENODO_HOME%/jre/lib/security/cacerts
  ```

- For Unix:
  ```
  export JAVA_OPTS=-Djavax.net.ssl.trustStore=$DENODO_HOME/jre/lib/security/cacerts
  ```
Some applications allow you to set this property without setting an environment variable:

- JConsole:

  jconsole
  -J-Djava.net.ssl.trustStore=${DENODO_HOME}/jre/lib/security/cacerts

  In JConsole, when SSL is enabled, enter the URL of the Denodo server with the format `<host>:<port>` instead of `<service>:<...`.

### 3.2.5.2 ODBC Clients

The sections “Access through ODBC” and “Access through an Ado.Net Data Provider” of the Virtual DataPort Developer Guide explain how to enable the SSL communications in ODBC and Ado.Net clients respectively.

### 3.3 POST-INSTALLATION TASKS IN VIRTUAL DATAPORT

The following subsections describe the post-installation tasks for Virtual DataPort.

#### 3.3.1 External CLASSPATH

It may be necessary to add new classes or libraries to the Denodo Virtual DataPort execution classpath.

For example, you need to provide the JDBC drivers of the databases not included within the Denodo Platform.

In this case, you have to copy their jar files to the folder `$DENODO_HOME/extensions/thirdparty/lib` directory.

The section “JDBC Drivers” of the Virtual DataPort Administration Guide [VDP] lists the JDBC drivers that are included within the Denodo Platform.

#### 3.3.2 Check Windows Service Configuration

When Virtual DataPort is installed as a Windows service, the service will be executed by default using the predefined `local system` user account. This may cause problems in some environments when using the Virtual DataPort Administration Tool. Therefore, it is strongly recommended to run the Virtual DataPort windows service using another user account.

#### 3.3.3 Installing the SAP JCo Connector

In order to retrieve data from SAP ERP (BAPI data sources) or from SAP BW / SAP BI (with multidimensional data sources with the BAPI adapter), you have to install the SAP Java Connector 3.0 (SAP JCo).

SAP provides different JCo libraries for several platforms. The JCo library you install must work with the CPU manufacturer and architecture on which Denodo is installed:

- If Virtual DataPort runs on a 32-bit O.S, download the 32-bit connector.
Denodo Platform 5.0 Installation Guide

Post-installation Tasks

- If Virtual DataPort runs on a 32-bit JVM, on a 64-bit O.S, download the 32-bit connector.
- If Virtual DataPort runs on a 64-bit JVM, download the 64-bit connector.

The following subsections explain:

- How to install SAP JCo on Windows: see section 3.3.3.1.
- How to install SAP JCo on Linux: see section 3.3.3.2.

3.3.3.1 Installing SAP JCo on Windows

Follow these steps:

1. Obtain the appropriate driver for your architecture. It can be downloaded from http://service.sap.com/connectors, section SAP Java Connector > Tools & Services.
2. Uncompress the downloaded package in a temporary directory.
3. Create the directory sap-erp-connector inside
   %DENODO_HOME%/extensions/thirdparty/lib/
4. Copy the following files to the directory
   %DENODO_HOME%/extensions/thirdparty/lib/sap-erp-connector/
   o sapjco3.jar
   o sapjco3.dll
5. According to the SAP note #684106 (https://service.sap.com/sap/support/notes/684106), if Virtual DataPort and SAP ERP are not running in the same host, you need to install the latest Visual Studio 2005 runtime libraries:
   b. Scroll to section “Affected Software”.
   c. Click the link “Microsoft Visual C++ 2005 Service Pack 1 Redistributable Package” (KB973544)
   d. Download one of the following platform-specific files depending on your scenario:
      i. Virtual DataPort running on a 32-bit O.S: vcredist_x86.exe
      ii. Virtual DataPort running on a 32-bit JVM, on a 64-bit O.S: vcredist_x86.exe
      iii. Virtual DataPort running on a 64-bit JVM: vcredist_x64.exe
      iv. Itanium system: vcredist_IA64.exe
e. Execute the downloaded file and follow the installation instructions.

6. To test that the JCo connector is properly installed, execute the following command from a console:
   
   ```
   cd %DENODO_HOME%
   cd extensions\thirdparty\lib\sap-erp-connector
   ..\..\..\..\jre\bin\java -jar sapjco3.jar
   ```

   Note that you are launching the Java Runtime Environment installed along with the Denodo Platform. If you are going to launch the Denodo Platform with another JRE, you have to execute this command with that JRE to make sure that at runtime, the SAP JCo connector will work.

   The library is properly installed if this command runs without error and you see something like Figure 7.

   ![Figure 7 SAP Java Connector (JCo) test screen](image)

3.3.3.2 Installing SAP JCo on Linux

Follow these steps:
1. Obtain the appropriate driver for your architecture. It can be downloaded from [http://service.sap.com/connectors](http://service.sap.com/connectors), section SAP Java Connector > Tools & Services.

2. Uncompress the downloaded package in a temporary directory.

3. Create the directory `sap-erp-connector` inside `$DENODO_HOME/extensions/thirdparty/lib`

4. Copy the following files to the directory `sap-erp-connector`

   o `sapjco3.jar`
   o `libsapjco3.so`

5. To test that the JCo connector is working, execute the following command:

   ```
   cd $DENODO_HOME
   cd extensions/thirdparty/lib/sap-erp-connector
   java -jar ..\..\..\..\..\jre\bin\java -jar sapjco3.jar -stdout
   ```

   Note that you are launching the Java Runtime Environment installed along with the Denodo Platform. If you are going to launch the Denodo Platform with another JRE, you have to execute this command with that JRE to make sure that at runtime, the SAP JCo connector will work.

   If the library is properly installed, this command runs without error and provides information about the installed JCo libraries.

### 3.3.4 Installing the Connector for SAP BW and SAP BI (Multidimensional Sources)

In order to retrieve data from SAP BW or SAP BI, you have to install its connector.

To do this, follow these steps:

1. Enable XMLA access in SAP. You can follow the instructions in the following link to know how to configure XMLA: [http://help.sap.com/javadocs/NW04/current/bi/docs/connectors/xmla_howto.html](http://help.sap.com/javadocs/NW04/current/bi/docs/connectors/xmla_howto.html).

   **Note:** You must increase the Web service client time out of SAP BW as explained here: [http://wiki.scn.sap.com/wiki/display/JSTSG/%28WSR%29Problems-P01](http://wiki.scn.sap.com/wiki/display/JSTSG/%28WSR%29Problems-P01)

2. Obtain the SAP Business Intelligence Java Software Development Kit (BI Java SDK) distributed with SAP NetWeaver. The SDK can be downloaded from here: [http://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/30232e2e-44-74ad74a4-c482e2203093](http://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/30232e2e-44-74ad74a4-c482e2203093)

3. Uncompress the package and copy all its jars, including the jars in its subdirectories, to `$DENODO_HOME/extensions/thirdparty/lib/sap-bw-connector/`

3.3.5 Installing the Connector for Oracle Essbase

In order to retrieve data from Oracle Essbase, you have to install its connector. If you have Oracle Essbase version 9, copy the file `ess_japi.jar` to the directory `$DENODO_HOME/lib/extensions/essbase-drivers/9`. If you have Oracle Essbase version 11, copy the file `ess_japi.jar` to the directory `$DENODO_HOME/lib/extensions/essbase-drivers/11`.

3.3.6 Installing the JMS Connectors to Create JMS Listeners and Web services with SOAP over JMS

To connect to a JMS server, you have to install its connector. JMS connectors are a set of jars that have to be copied into the Denodo Platform. You have to do this in order to do the following:

- Create JMS Listeners: see section ‘JMS Listeners’ of the Administration Guide [VDP]. If you plan to do this, copy the client jars of the JMS server to the directory `$DENODO_HOME/extensions/thirdparty/lib`.

- Or, create Web services with SOAP over JMS: see section ‘JMS Listeners’ of the Administration Guide [VDP]. If you plan to publish Web services with the option SOAP over JMS enabled and deploy them in the embedded Web container of the Denodo Platform, copy the client jars of the JMS server to the directory `$DENODO_HOME/resources/apache-tomcat/common/lib`. To deploy Denodo Web services with the option SOAP over JMS enabled, in an external application server, copy the JMS client jars into the `/WEB-INF/lib` directory of the generated war file, before deploying it.

3.3.6.1 JMS Client Jars

This section lists the client jars of the most popular JMS servers. These jars are provided by the vendor.

**Apache ActiveMQ 5.6.0**

The client jars are located in `$ACTIVE_MQ_HOME/lib`:

- `activemq-core.jar`
- `geronimo-j2ee-management.jar`
- `geronimo-jms_1.1_spec.jar`
- `slf4j.jar`

**IBM WebSphere MQ 7.0**

The client jars are located in `$WEBSPHERE_MQ_HOME/java/lib`:

- `com.ibm.mq.jar`
- `com.ibm.mqjms.jar`
• com.ibm.mq.jmqi.jar
• dhbcore.jar
• jta.jar
• fscontext.jar
• providerutil.jar

**OpenJMS 0.7.7**
The client jars are located in `$OPEN_JMS_HOME/lib`:

• openjms.jar
• openjms-common.jar
• openjms-net.jar
• commons-codec.jar
• commons-logging.jar
• concurrent.jar
• jms.jar
• jndi.jar
• spice-jndikit.jar

**Progress SonicMQ 8.0**
The client jars are located in `$SONIC_MQ/Sonic/MQ8.0/lib`:

• mfcontext.jar
• sonic_Client.jar
• sonic_Crypto.jar
• sonic_XA.jar
• sonic_XMessage.jar

### 3.3.7 Configuring the VCS Clients

Virtual DataPort can use a Version Control System (VCS) to store the metadata of the Virtual DataPort server (data sources, views, etc.). This allows users to do the main tasks involved in version control from the Administration Tool: check out / update and check in / commit of databases and their elements.

The supported Version Control Systems are:
- **Subversion.** The section 3.3.7.1 describes how to configure your Subversion client to work with the Denodo Platform.

- **Microsoft Team Foundation Server (TFS).** The section 3.3.7.3 describes how to configure the Denodo Platform to work with TFS.

The following subsections explain the post-installation tasks that you have to perform to work with VCS. To learn how to use the VCS support of Virtual DataPort, read the section “Version Control Systems Integration” of the Virtual DataPort Administration Guide [VDP].

3.3.7.1 Configure a Subversion Client to Use the VCS Integration

To use Subversion to store the metadata of Virtual DataPort, perform these tasks:

1. Set up a Subversion server. The supported version is 1.7.

2. In the host where the Virtual DataPort server will run, do the following:
   a. Install a Subversion client. The version of the client has to be 1.7. The recommended client is Apache Subversion 1.7.x [SVN].
   b. Add to the PATH environment variable the directory where the svn executable is located. For Apache Subversion, that is the bin directory.
   c. Make sure that the “global ignores” list of the Subversion client does not include any of the following patterns:
      - *.a, *.o, (in general, *.<character>)
      - *.vql
      - *.properties
      - *.dependencies

   With Apache Subversion, you have to change the value of the property global-ignores in this configuration file:
   - On Windows: %APPDATA%\Subversion\config
   - On Linux: ~/.subversion/config or /etc/subversion/config

   The default configuration of Subversion clients includes several file and directory name patterns that are ignored by Subversion operations. For example, by default Apache Subversion 1.7.5, ignores the files that match any of the following patterns:
   

   Note that *.o and *.a are included in the list. This is problematic because of the way Virtual DataPort maps folders to their physical location in the file system when exporting to repository or performing VCS operations.

   For example, a folder named a will be physically located at <some>/<path>/folder.a. As all *.a files and directories are ignored by
Subversion by default, VCS operations involving such Virtual DataPort folders will fail.

Make sure that the **global-ignores** list does not include any of the mentioned patterns, as they correspond to the types of files involved in VCS operations.

**Note:** some Subversion servers such as “CollabNet Subversion Edge” cannot handle files whose name contains characters reserved by the file system like \, /, :, *, ?, etc. Therefore, we strongly recommend not using any of these characters in the name of the database or any of its elements if this database will be stored in a Subversion server.

### 3.3.7.2 Configure the Denodo Platform to Work with Subversion

To use Subversion to store the metadata of Virtual DataPort, follow these steps:

1. If the host is Linux, launch the Virtual DataPort server and an Administration Tool, and enable the “Uniqueness detection”. To do this, follow the steps described in the section “Enabling Uniqueness Detection” of the Virtual DataPort Administration Guide. This feature is enabled by default on Windows.

2. If the host is Linux and Virtual DataPort will connect to Subversion using the http or https protocol, enable the “LS Optimization”. To do this, follow the steps described in the section “Activating the LS Optimization” of the Virtual DataPort Administration Guide. This optimization is enabled by default on Windows.

### 3.3.7.3 Configure the Denodo Platform to Work with Microsoft TFS

To use Microsoft Team Foundation Server (TFS) to store the metadata of Virtual DataPort, you have to set up a Microsoft Team Foundation Server (TFS). The supported versions are 2010 or higher.

The Denodo Platform includes the necessary libraries to connect to a TFS server.

The TFS administrator is in charge of creating and managing the collections that will contain team projects with Virtual DataPort metadata and their different development branches, if any. We recommend having at least one branch in each project with Virtual DataPort metadata (as described by this [MSDN article](https://docs.microsoft.com/en-us/previous-version/msdn/hh412538(v=vs.90))).

The recommended repository structure will be like this (the nodes in italics are managed by the TFS administrators, the others are managed by Virtual DataPort):

- **TFS Servers**
  - DefaultCollection
  - Collection 1
  - Collection 2
  - ...
  - VDP Collection 1
    - Team Project 1
• Team Project 2
• ...
• Team Project n
  • Branch-1
  • Branch-2
  • ...
  • Branch-n
  • Main
    o databases
      ▪ my database 1
      ▪ my database 2
      ▪ ...
      ▪ my database n
    ...
    o extensions
    o environments
    o maps

Each TFS collection is backed up by a different database, so it must be managed separately from the others.

Each TFS project can contain several Virtual DataPort databases for each development branch, which will share environments and global elements, so we recommended that each TFS project contain Virtual DataPort metadata related to only one application.

### 3.3.8 Importing the Certificates of Data Sources (SSL Connections)

When Virtual DataPort establishes an SSL connection with a data source, the data source presents a certificate. Virtual DataPort relies on the Java Cryptography Architecture (JCA) to check if the certificate is valid. JCA accepts certificates signed by known Certificate Authorities (CA) (see the list of known CAs here [JRE-CERT]). However, if the certificate used by the server is signed by an authority not present in this list, you have to import this certificate into the list of trusted certificates (called TrustStore).

To import a certificate into the TrustStore of the Java Runtime Environment (JRE), execute the following commands:

```bash
cd <DENODO HOME>
```
cd jre/bin
keytool -importcert -alias <name of the certificate> -file <new certificate>.crt -keystore ../lib/security/cacerts

This command will prompt you for the password of the TrustStore, which by default is "changeit" (without the quotes).

Explanation of the parameters:

- **alias**: this parameter is mandatory. The certificate will be stored in the TrustStore identified by this alias. If the TrustStore already contains a certificate with this alias, use another alias.

- **keystore**: path to the TrustStore where the certificate will be stored.
  
  
  ../lib/security/cacerts is the path of the TrustStore of the JRE included in the Denodo Platform. If you have uncommented the property com.denodo.security.ssl.trustStore of the file <DENODO HOME>/conf/vdp/VDBConfiguration.properties, the value of this parameter has to be the value of this property, instead of "../lib/security/cacerts". That is because, if this property is uncommented, Virtual DataPort will use the TrustStore set in this property of the VDBConfiguration.properties file, instead of the JRE one.

If you are going to launch Virtual DataPort with a JRE not included in the Denodo Platform and the property com.denodo.security.ssl.trustStore is commented, the value of this parameter has to be the path to the cacerts file of this other JRE, which is located in the directory lib/security of the JRE.

To check that the certificate has been imported correctly, execute this command:
keytool -list -v -alias <name of the certificate> -keystore ../lib/security/cacerts

After adding a certificate, you have to restart the Virtual DataPort server in order for the changes to take effect.

The keytool documentation [KEYTOOL] provides much more details about the parameters of this tool.

### 3.3.9 Increasing the Maximum Simultaneous Requests

The Denodo Platform embeds an Apache Tomcat that Virtual DataPort uses to deploy SOAP and REST Web services. If you expect these Web services to receive a high number of concurrent requests, consider increasing the maximum number of threads that Tomcat will create to attend requests. To do this, follow these steps:

1. Open the file
   
   <DENODO_HOME>/resources/apache-tomcat/conf/server.xml.template.
   
2. Look for the attribute maxThreads and replace its default value (150) with a higher one. For example, 300.
   
   There are two occurrences of the maxThreads attribute, one for the non-SSL Connector and another one for the SSL Connector. You can have a different value for each one.
3. From the Administration Tool, do the following:

   a. Open the wizard “Concurrent Requests” on the menu “Administration > Server configuration”.

   b. Make sure that if “Limit concurrent requests” is “On”, the value of “Max concurrent requests” is greater than or equal to the maxThreads attribute. Otherwise, Tomcat will process the requests, but Virtual DataPort will not be able to attend them.

   If you have enabled SSL on Tomcat (explained in the section 3.2.4 Enabling HTTPS in the Embedded Apache Tomcat), “Max concurrent requests” has to be greater than or equal to the sum of the maxThreads attribute of the non-SSL connector and the SSL connector.

Each incoming request requires a thread for the duration of that request. If Tomcat receives more simultaneous requests than the number of available request processing threads, Tomcat creates additional threads up to maxThreads. If Tomcat still receives more simultaneous requests, they are stacked up, up to the value of the acceptCount attribute of the Connector element. Any further simultaneous requests will receive "connection refused" errors.

3.4 POST-INSTALLATION TASKS IN ITPilot

Once ITPilot has been installed, certain configuration tasks must be carried out before it can work properly.

3.4.1 Checking that the Generation Environment Has Been Installed

The module ITPilot Generation Environment installs a toolbar in the Internet Explorer browser that is used to aid in the wrapper generation process (recording browsing sequences and generating data extraction programs).

To check that the toolbar has been installed correctly, do the following:

1. Launch Microsoft Internet Explorer.

2. The navigation sequences generator toolbar should be visible on the browser.
   If it is not, activate it by clicking Sequence Generator in the View – Toolbars menu.

See the section 3.4.3 to enable the toolbar in Microsoft Windows 2003 and Windows 2008.

3.4.2 Configuration of Microsoft Internet Explorer

To run automated browser sequences, among other options, ITPilot will use a Browser pool that will spawn browsers based on Microsoft Internet Explorer or Denodo Browser.

An initial aspect to bear in mind is that the pool browsers will use the configuration established for Microsoft Internet Explorer browsers in the system where the pool is running.
It is recommended to consider the proxy, security, and cookie options, as the browsers will behave according to this configuration.

3.4.2.1 Configuring Microsoft Internet Explorer when the Browser Pool is executed as a Windows Service

In the event of starting the ITPilot browser pool as a Windows service, the service will be executed by default using the predefined *local system* user account. Configuring Microsoft Internet Explorer for this user is not easy (see below). Therefore, it is recommended to run the browser pool windows service using a different user account. The recommended procedure is the following:

1. Create a new user account.
2. Configure the Windows service to use that account from the service properties dialog.
3. Log in to the system using the new credentials.
4. Start Microsoft Internet Explorer.
5. Perform the necessary configuration changes.
6. After logging out, the system will be ready to use the ITPilot browser pool as a Windows service.

If the service is configured to run under the *local system* account, then the recommended procedure to configure Microsoft Internet Explorer on that account is the following:

2. Start the "Interactive Services Detection" Windows service.
3. Use the PsExec utility, included in the PsTools suite, to open a Microsoft Internet Explorer instance on the local system account. To do that, execute the following from a command line (adapting the path of the Microsoft Internet Explorer executable to that of your system):

   4. `PsExec.exe -s -i 0 'C:\Program Files\Internet Explorer\iexplore.exe`

   5. Perform the necessary configuration changes.
6. After closing Microsoft Internet Explorer the system will be ready to use the ITPilot browser pool as a Windows service on the *local system* account.

3.4.2.2 Configuring the Microsoft Internet Explorer 8 and Later

By default, Internet Explorer 8 and 9 spawn each tab as a separate process of the operating system. For this browser to allow ITPilot to record new NSEQL sequences it must be configured to open all tabs in the same process.

- In 32-bit Windows operating systems follow these steps:
1. Execute RegEdit.

2. Browse to HKEY_LOCAL_MACHINE > SOFTWARE > Microsoft > Internet Explorer > Main.

3. Add a DWORD value called TabProcGrowth with value of 0.

- In 64-bit Windows operating systems follow these steps:
  1. Execute the 64-bit version of RegEdit (this is the version included in the PATH by default).
  2. Browse to HKEY_LOCAL_MACHINE > SOFTWARE > Microsoft > Internet Explorer > Main.
  3. Add a DWORD value called TabProcGrowth with value of 0.
  5. Add a DWORD value called TabProcGrowth with value of 0.

3.4.3 Enabling Internet Explorer Sequence Generation Toolbar in Microsoft Windows 2003 and Windows 2008

To show the Internet Explorer toolbars in a Microsoft Windows Server 2003 or Windows Server 2008 operating system, it is necessary to enable third-party browser extensions.

To manually enable third-party browser extensions:

1. Close all instances of Internet Explorer, click Start, point to Settings, and then click on Control Panel.

2. Double-click Internet Options.

3. Click on the Advanced tab.

4. Under Browsing, select the "Enable third-party browser extensions (requires restart)" check box.

5. Restart Internet Explorer.

The previous steps are equivalent to set value "Enable Browser Extensions"="yes" in the registry key "HKCU\Software\Microsoft\Internet Explorer\Main".

3.4.4 Disabling Internet Explorer Enhanced Security Configuration in Microsoft Windows 2008

In Microsoft Windows Server 2008, the Internet Explorer Enhanced Security Configuration feature interferes with the correct functioning of the Sequence Generation Toolbar, and must be disabled.

The steps to disable this feature are as follows:
1. Log on to the computer with a user account that is a member of the local Administrators group.

2. Click Start, go to Administrative Tools, and then click Server Manager.

3. If the User Account Control dialog box appears, confirm that the action it displays is what you want, and then click Continue.

4. Under Security Summary, click Configure IE ESC.

5. Set IE ESC to Off for the appropriate user type (Administrators, Users).

6. Click OK.

7. Restart Internet Explorer.

3.4.5 Manual Configuration of Adobe Acrobat Professional

During the installation process, you can enter the path to the plugins directory of Adobe Acrobat Professional.

If you do not, you can configure it later, by doing one of these actions:

- By setting it in the ITPilot Administration Tool or in the Browser Pool tabs (see [ITPILOT] for more information).

- Or, in the Document Conversion tab of the ITPilot Wrapper Generation Tool’s preferences (see [GENITP] for more information)

3.4.6 Automatic Verification Database

The Verification server uses a relational database to store information about wrappers execution to allow detecting when the sources changes and the wrappers stop working properly. ITPilot provides an embedded Apache Derby [DERBY] database that can be used for this purpose. If the embedded database is going to be used, no action is required in this section.

An external JDBC database management system can also be used. In the current ITPilot version, the supported databases are MySQL [MYSQL] and Oracle [ORCL].

ITPilot provides a script to create the table for these Database Management Systems in the path DENODO_HOME/scripts/itpilot/sql. If an external database is going to be used for this purpose, it is needed to install the database and run on it the corresponding tables creation script.

3.5 POST-INSTALLATION TASKS IN ARACNE AND SCHEDULER

Once Aracne and/or Scheduler have been installed, certain configuration tasks must be carried out before it is ready to run.

3.5.1 Microsoft Internet Explorer Configuration

An aspect to bear in mind when using the IECrawler module of Scheduler and Aracne is that its browsers will use the configuration settings of Microsoft Internet Explorer (MSIE):
• It is recommended to set the MSIE home page to a blank page ("about:blank") to prevent each new browser launched by IECrawler from connecting to the home page before running a task.

• For HTTPS site crawling to work properly, the MSIE preferences must be established to prevent pop-up windows from informing users of the change to secure/insecure mode and to accept digital certificates (if not, the browsers may be blocked). These preferences are in MSIE in the 'Internet Options’ option of the Tools menu, selecting the ‘Advanced Options’ tab and then the “Security” option group.

• To connect to an FTP server using IECrawler, the Active or Passive connection mode of the MSIE FTP client must be configured. The correct value will depend on the firewall software configuration in the client and in the FTP server. The preferences to establish the MSIE FTP client connection mode are in the 'Internet Options' option of the Tools menu, selecting the ‘Advanced Options’ tab and then the “Browse” option group. The MSIE connects in Active mode if the “Enable the list of folders for FTP sites” option is selected, even if the “Use passive FTP” option is selected. If the “Enable the list of folders for FTP sites” option is not selected and the “Use passive FTP” option is selected, the MSIE connects in Passive mode.

• Due to the fact that the browsers behave in line with the configuration settings of MSIE, the other MSIE security and cookie options should also be taken into account.

In the event of executing Aracne or Scheduler as a Windows service, by default it will be executed using the predefined user local system account.

Configuring Microsoft Internet Explorer for this user is not easy (see below). Therefore, it is recommended to execute the Aracne and Scheduler services using a different user account. The recommended procedure is the following:

1. Create a new user account.
2. Configure the Windows service to use that account from the service properties dialog.
3. Log in to the system using the new credentials.
4. Start Microsoft Internet Explorer.
5. Perform the necessary configuration changes.
6. After logging out, the system will be ready to execute Aracne server as a Windows service.

If the service is configured to run under the local system account, then the recommended procedure to configure Microsoft Internet Explorer on that account is the following:

1. Download the Microsoft utility suite PsTools (http://technet.microsoft.com/en-us/sysinternals/bb896649) and unzip it.
2. Start the "Interactive Services Detection" Windows service.
3. Use the PsExec utility, included in the PsTools suite, to open a Microsoft Internet Explorer instance on the local system account. To do that, execute the following from a command line (adapting the path of the Microsoft Internet Explorer executable to that of your system):

```
PsExec.exe -s -i 0
'C:\Program Files\Internet Explorer\iexplore.exe'
```

4. Perform the necessary configuration changes.

5. After closing Microsoft Internet Explorer, the system will be ready to use the ITPilot browser pool as a Windows service on the local system account.

### 3.5.2 Configuring the Exchange Crawler Server

In the case of Exchange Crawler server running as a Windows service, it will work correctly only if the service is run by a Microsoft Exchange Server Administrator user account.

### 3.6 Deploying the Web Administration Tool in an External Web Container

The Denodo Platform provides a war file that contains the Web Administration Tool thus it can be deployed in an external Web container.

The file is located at `<DENODO_HOME>\webapps\admintool\denodo-webadmintool-5.0.0.war`.

This application can be used to manage ITPilot, Aracne and Scheduler servers as well.

After deploying the application in the Web container, there are still several configuration task required before the application can be run:

- Edit the `web.xml` file located in the application WEB-INF directory
- Locate the "users.location.userList" context-param and update its value with the absolute path to the file, located in the WEB-INF/classes directory.
  
  **Example:** `<WEBAPP_DIR>/WEB-INF/classes/users-configuration.xml`

- Locate the "servers.location.itpilot" context-param and update its value with the absolute path to the file.
  
  **Example:** `<WEBAPP_DIR>/WEB-INF/classes/itpilot-servers.xml`

- Do the same with the values of the context-params "servers.location.aracne" and "servers.location.scheduler".

- Save the changes and restart the Web application.

By following these steps, you will end with one Web Administration Tool for the three modules: ITPilot, Aracne and Scheduler.

There is also a way to generate a Web Administration Tool for each module, which is directly deployable in the Web Container, requiring no manual actions (except
deploying it). Thus, if you want to manage the three servers, you will end up with three different .war files.

- To generate the war file for Aracne: execute the script located at <DENODO_HOME>/setup/arn/rewrite_arn_webapp.{bat|sh}

- To generate the war file for ITPilot: execute the script located at <DENODO_HOME>/setup/itpilot/rewrite_itp_webapp.{bat|sh}

- To generate the war file for Scheduler: execute the script located at <DENODO_HOME>/setup/scheduler/rewrite_sched_webapp.{bat|sh}

These scripts modify the same file located at <DENODO_HOME>/webapps/admintool/denodo-webadmin-5.0.0.war, so it must be deployed in the Web Container after executing each of the scripts (obviously, with a different name for each module).
4 UPGRADE TO DENODO PLATFORM 5.0

To upgrade to Denodo Platform 5.0 from a previous version follow the steps described in the Denodo Platform Migration Guide [DENMIGR].
5 UNINSTALLATION

In the uninstallation process, you can uninstall the entire Denodo Platform or just some of its modules.

Before starting the uninstallation process, close all the Denodo servers and tools and all the Microsoft Internet Explorer instances. Otherwise, the folder in which the software was installed will not be deleted.

From the Denodo Platform Control Center (see section 7) and click Uninstall to launch the uninstallation wizard.
6 DENODO4E

The Denodo Platform includes Denodo4E: an Eclipse plug-in that provides tools for creating, debugging and deploying Denodo extensions.

The Denodo4E plug-in has been tested with Eclipse Indigo (3.7).

To install the plug-in, follow these steps:

1. Click Install New Software on the menu Help.

2. Click the button Add.

3. Click Local... to open the “Add Repository” dialog (see Figure 8). In this dialog, select the file <DENODO_HOME>/tools/denodo4e/updateSite and click Ok. Then introduce a name for the new update site. For example "Denodo", and click OK.

4. Select Denodo4E (located under the Denodo Technologies category)

5. Select the check box Contact all update sites during install to find required software.

6. Click Next. Once Eclipse finishes calculating dependencies, click Finish to begin the installation process.

7. Once all the required features are installed, restart Eclipse.
7 DENODO PLATFORM CONTROL CENTER

The Denodo Platform Control Center allows starting and stopping all Denodo Platform servers and tools, as well as a group of additional functions. This section describes its use.

7.1 EXECUTION

To launch the Denodo Platform Control Center do the following:

- Use the shortcuts created on the desktop during the installation and/or the application menus of your Operating System
- Or, execute the script $DENODO_HOME/bin/denodo_platform

NOTE 1: In Windows Vista, Windows 7 and Windows Server 2008, we recommend deactivating the User Account Control before executing Denodo Platform Control Center. If you do not want to do this, you should keep in mind the following issues:

- If you have installed the Denodo Platform under a folder requiring administrator privileges (e.g. the 'Program Files' folder), you may need to explicitly use the “Run as Administrator” option in the contextual menu for the Control Center executable, even if you are already logged in as an administrator user. If you start a “standalone” Internet Explorer (i.e. it is not executed from the ITPilot Wrapper Generation Tool) you also may need to explicitly use the “Run as Administrator” option, to properly use the ITPilot Sequence Generator Toolbar.

- If you are using the PDF, Excel or Word conversion features then the Internet Explorer protected mode must be turned off (this can be done from the Security page of the Internet Options dialog of Internet Explorer).

NOTE 2: When you launch any Denodo server, an icon will be displayed in the notification area of your operating system. This icon will not be displayed when using a 64-bits Java Virtual Machine to launch the server.

7.2 CONTROL CENTER HELP

The first time you launch the Denodo Platform Control Center you will see the Help screen (see Figure 9). If you clear the Show on startup check box, the first thing you will see the next time you launch the Control Center will be the “Custom environments” screen (see section 7.3.1 for details). You can always return to the Help screen by clicking the “Help” button on the right side of the Control Center window.
Figure 9 Denodo Platform Control Center Help screen

This screen is divided in three sections:

- **Documentation & Tutorials**
  
  Denodo Platform documentation: here you can access the Denodo Platform on-line documentation. If you click the link, the Denodo website containing all the user manuals and guides for the Denodo Platform will be opened in your web browser.
  
  - Denodo Platform tutorials: by clicking this link, the Denodo Platform tutorials website will be opened in your web browser.
  
  - Denodo Platform online help: by clicking this link, the online help configuration dialog will appear (see section 7.2.1 for details).

- **License**
  
  - License information: clicking here will open a dialog box displaying your license conditions (see Figure 10).
  
  - Install license: clicking here will open a file chooser where you can select your license file.

- **Useful Links**
Denodo Technologies: this link opens the Denodo Technologies corporate site (www.denodo.com) in your browser.


---

**Figure 10** Information on the current license

### 7.2.1 Online Help Configuration

The location of the help contents can be configured in the online help configuration dialog (see Figure 11). By default, the ITPilot Wrapper Generation Tool and the VDP Administration Tool will show the help located in the local web container when requested. Alternatively, a remote help container can be configured. In this case, the help web applications included with the Denodo Platform should be deployed so they become available at <remoteContainerURI>/help/vdp/platform/5.0/vdpmintool and/or <remoteContainerURI>/help/itp/platform/5.0/itpgentool. The corresponding war files can be found at $DENODO_HOME/webapps/help.
7.3 STARTING PLATFORM SERVERS AND TOOLS

The buttons on the left side of the Control Center (see Figure 12) show the dialog of each module:

- **vdp** opens the dialog of the *Virtual DataPort* module
- **itp** opens the dialog of the *ITPilot* module
- **arn** opens the dialog of the *Aracne* module
- **sched** opens the dialog of the *Scheduler* module

Each one of these dialogs allows you to start/stop the servers and graphical tools of the module and display their status.
See the Virtual DataPort Administration Guide [VDP], the ITPilot User Guide [ITPILOT], the Aracne Administration Guide [ARCN] and the Scheduler Administration Guide [SCH] for a description of the servers and tools comprising the Virtual DataPort, ITPilot, Aracne and Scheduler modules, respectively.

Press the button to start a server. The status will change to “Starting” until the server has started, when it will display the message “Started” together with the server stop (■) button. In addition, if the button of each one of the panels is clicked the port number where the server is listening for requests will be shown.

The big ■ button in the lower-left part of the screen allows stopping all the currently launched servers and web applications of the Denodo Platform.

To start the Virtual DataPort administration tool or the ITPilot wrapper generation tool, press the button in their panel. To close these tools, use the ‘Exit’ option in their ‘File’ menu (they cannot be directly closed from the Control Center).

To start the ITPilot web administration tool, the Aracne web administration tool or the Scheduler web administration tool, press the button in their panel. The tools are deployed in the Web service container embedded with the Denodo Platform. To access any of the tools, click the button once they are started: the appropriate website will be opened in your web browser. You can also use your web browser with, respectively, the following URLs (we assume the default 9090 port and that the Denodo Platform is installed in localhost):

http://localhost:9090/webadmin/denodo-itpilot-admin/
http://localhost:9090/webadmin/denodo-aracne-admin/
The servers/tools of the Denodo Platform also can be launched with a script.
The list below refers to the documents that explain how to launch the servers/tools of each module, with a script:

- **Virtual DataPort**: “Launching the Virtual DataPort Server” and “Launching the Virtual DataPort Administration Tool” sections of the Virtual DataPort Administration Guide [VDP].

- **ITPilot**: “Execution” section of the ITPilot User Guide [ITPILOT].

- **Aracne**: “Installation and Execution” section of the Aracne Administration Guide [ARCN]

- **Scheduler**: “Installation and Execution” section of the Scheduler Administration Guide [SCH].

### 7.3.1 Configuration of Custom Environments

The Denodo Platform Control Center allows to define “Custom environments”, that are a collection of programs that can be managed as a group, instead of individually. The environments can contain servers and/or administration tools at the same time so that all elements of the Denodo Platform required in a certain environment can be started and stopped at once. You can access the Custom environments screen by clicking the “custom” button (see Figure 13).

There is a default custom environment included with the Denodo Platform: Data Integration Environment. This environment cannot be edited or deleted, but if you select it and then click the **i** button, you will be able to see its configuration. The Data Integration Environment includes the Virtual DataPort Server and Administration Tool and the ITPilot Browser Pool.

To create a new environment, press the **+** button to access the environment editing screen, where you can select the name of the environment, the JAVA virtual machine to be used by this environment to execute the programs, and the modules to be started in that environment. You can use the **edit** button to modify an existing environment, and the **x** button to delete it.
Once an environment has been defined, it can be started or stopped by selecting it from the list of environments and pressing the run or the stop button, respectively.

Figure 13  Creation of custom environments

Figure 14  Creation and editing of custom environments
7.4 Installing an Update or a Hotfix

The Control Center displays the version the latest installed update or hotfix, next to the name of each product.

The updates or hotfixes are released in a zip file, which includes the following:

- A RELEASENOTES file for each module that will be updated. These files describe the bug fixes and enhancements.
- A jar file that contains the update.

To install a new update or a hotfix, follow these steps:

1. Close all the Denodo programs and tools; stop all the Denodo servers.
2. Close all the instances of Microsoft Internet Explorer.
3. Decompress the zip file of the update.
4. Read the RELEASE NOTES file of each product. Pay special attention to the “Post-installation actions” of each product.
5. Open the Control Center.
6. Click Update to open the “Platform Updates” dialog. This dialog lists the updates that have been installed (the last update of the list is the current one).
7. Click Install and select the jar file of the update. The Control Center will display a dialog with a progress bar.

The section 8.2 explains how to install an update or a hotfix in a host without graphical support.

7.5 Other Functions

This section describes other functions of the Control Center.

7.5.1 Virtual Machine Parameters and Web Container Configuration

Click Configure to open the “Denodo Platform Control Center configuration”. In this dialog, you can do the following:

- Select the Java Virtual Machine (JVM) used to run the Denodo Platform servers.
- The JVM parameters used to launch each Denodo server.
- The ports that the Denodo embedded Web container listens for incoming connections.

To select the Java Virtual Machine (JVM) used to run the Denodo Platform servers, click Configure and then, Edit. In this dialog, you can add a JVM present on your
system that you want the Denodo Platform servers to run with. The Denodo servers will run with the JVM that is selected on the list.

Although it is possible to change the JVM, we recommend running the Denodo servers with the “Denodo (Internal JVM)”, unless you have used the denodo-install-5.0.zip installer, which does not include a Java Runtime Environment (JRE).

To configure the JVM options of each Denodo server, click **Configure** and then, **JVM options**. This dialog has two tabs:

- **Memory options.** In this tab, you can change the JVM parameters of each Denodo server. The section 8.3 explains how to change these parameters in a host without graphical support.

- **RMI host.** In this tab, you can configure the RMI host name for each server of the Denodo Platform. You have to change this if the host where the Denodo server runs provides several network interfaces. In this case, you have to specify a network interface (as an IP address or a host name) that is visible to clients that have to connect to this server.

In both tabs, next to each field there is a button () that can be used to go back to the default value. The Ok button will save the configuration changes, which will be applied in the next startup of the affected programs.

To configure the ports that the Denodo embedded Web container listens for incoming connections, click **Configure** and enter the desired ports in **HTTP port number**, **Shutdown port number** and **Auxiliary port number**.

You can also configure these ports using the Virtual DataPort Administration Tool (see section “Configuring the Server” of the Virtual DataPort Administration Guide [VDP] for information about how to do this).

### 7.5.2 Uninstalling the Platform

The “Uninstall” button opens a dialog box to uninstall all or part of the Platform modules (see Figure 15).

Before continuing with the uninstallation, stop all Denodo Platform servers and tools (including the Control Center) and close all active instances of Microsoft Internet Explorer. You will be warned if any Denodo server or tool is detected after clicking the Uninstall button, but you will be allowed to continue the uninstallation process (see Figure 16).
Figure 15  Uninstalling the Denodo Platform

Figure 16  Processes running warning
8 INSTALLATION AND CONFIGURATION IN SILENT MODE

The following sections explain how to perform several tasks with the Denodo Platform in a “headless” host (i.e. a host without graphical support):

2. How to perform a silent installation of a Denodo Platform update: see section 8.2.
3. How to configure the JVM parameters: see section 8.3.

8.1 SILENT INSTALLATION OF THE DENODO PLATFORM

In an installation in silent mode, the installation tool does not display any graphical user interface and the process is performed without any user intervention. This can be useful if you want to install the Denodo Platform in a “headless” machine (i.e. without graphical support).

To perform a silent installation, follow these steps:

1. Select a Denodo Platform installer. Denodo provides several flavors of the installer and the only difference between them is the Java Runtime Environment (JRE) they include:
   a. denodo-install-5.0-linux32.zip: it includes a 32-bit JRE for Linux.
   b. denodo-install-5.0-linux64.zip: it includes a 64-bit JRE for Linux.
   c. denodo-install-5.0-win32.zip: it includes a 32-bit JRE for Windows.
   d. denodo-install-5.0-win64.zip: it includes a 64-bit JRE for Windows.
   e. denodo-install-5.0.zip: it does not include any JRE. Use this installer if the previous installers are unsuited for your environment.

2. If you have selected an installer with JRE, go to the next step. Otherwise, set the environment variable JAVA_HOME to point to a JRE version 6 or 7.

3. Perform a temporary installation in a machine with graphical support. The result of performing this installation will be an “autoinstallation” file, which will be used later to perform the silent installation.

   To do this temporary installation, execute the following command from a shell:
   
   cd denodo-install-5.0
   java -DDENODO_JRE_HOME=./jre -DDENODO_AUTOINSTALL_FILE=ENABLED -jar denodo-install-5.0.dat
   
   Remove the parameter -DDENODO_JRE_HOME=./jre if you are using the installer without JRE.

   The variable DENODO_AUTOINSTALL_FILE instructs the installation tool to
store all the settings needed to perform a silent installation. At the end, the installation tool will create a denodo-platform-autoinstall.xml file in the $DENODO_HOME directory with the settings you have selected: target directory, selected modules, settings of each product, etc. You may need to change some of them. E.g., if you want to perform the silent installation in a different directory, change the value of the property INSTALL_PATH.

4. Copy the following files to the machine where you want to perform the silent installation:
   a. The Denodo Platform installer that suits your environment.
   b. The denodo-platform-autoinstall.xml file obtained in the previous step.

5. In the machine where you want to perform the silent installation, do the following:
   a. Decompress the Denodo Platform installer.
   b. Go to the directory where the installer has been decompressed.
   c. If this host is running Linux and you have selected an installer with JRE, execute this:
      chmod +x jre/bin/java
   d. Execute the following:
      java -DDENODO_JRE_HOME=./jre -jar denodo-install-5.0.dat <path-to-xml-file>/denodo-platform-autoinstall.xml
      This command will install the Denodo Platform with the options stored in the denodo-platform-autoinstall.xml file and will also configure the new installation to include and use the JRE embedded in the Denodo Platform’s installer.
      If you do not want to use this JRE or your installer does not include it, omit the -DDENODO_JRE_HOME=./jre argument from the last step.

The following section explains how to do silent installations of the Denodo Platform updates.

8.2 SILENT INSTALLATION OF UPDATES

In a silent installation of an update, the update does not display any user interface and the process is performed without any user intervention. This is useful if you want to install the update using a script or in a “headless” machine (without graphical support).

To do this, unzip the provided zip file and then, from a shell, execute the following command:
java -jar denodo-v50-update<yyyyMMdd>.jar <DENODO_HOME> -c
This installs the update in the <DENODO_HOME> directory.
8.3 CONFIGURATION OF THE JVM PARAMETERS IN SILENT MODE

In a host with graphical support, you can use the Denodo Control Center to change the parameters of the Java Virtual Machine (JVM) used to launch each Denodo server (see section 7.5.1). However, this is not possible in a “headless” environment (i.e. without graphical support).

In this section, you will learn to change these parameters in a host without graphical support. To do this, follow these steps:

1. In the first column of the table below, look for the products whose JVM parameters you want to change.

2. Edit their “Configuration file”

3. In the property “Property to Modify”, set the JVM parameters. Note that the characters “:” has to be escaped with the character “\”. E.g. “XX:\MaxPermSize\=128m”

<table>
<thead>
<tr>
<th>Product</th>
<th>Configuration File</th>
<th>Property to Modify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual DataPort server</td>
<td>&lt;DENODO_HOME&gt;/conf/vdp/VD_BConfiguration.properties</td>
<td>java.env.DENODO_OPTS_START</td>
</tr>
<tr>
<td>ITPilot Wrapper server</td>
<td>&lt;DENODO_HOME&gt;/conf/IEWrapper/IEBrowserConfiguration.properties</td>
<td>java.env.DENODO_OPTS_START</td>
</tr>
<tr>
<td>ITPilot Browser Pool</td>
<td>&lt;DENODO_HOME&gt;/conf/iebrowser/IEBrowserConfiguration.properties</td>
<td>java.env.DENODO_OPTS_START</td>
</tr>
<tr>
<td>ITPilot Verification Server</td>
<td>&lt;DENODO_HOME&gt;/conf/maintenance/ConfigurationParameters.properties</td>
<td>java.env.DENODO_OPTS_START</td>
</tr>
<tr>
<td>Scheduler Server</td>
<td>&lt;DENODO_HOME&gt;/conf/scheduler/ConfigurationParameters.properties</td>
<td>java.env.DENODO_OPTS_START</td>
</tr>
<tr>
<td>Aracne Server</td>
<td>&lt;DENODO_HOME&gt;/conf/arn/ConfigurationParameters.properties</td>
<td>java.env.DENODO_OPTS_START</td>
</tr>
<tr>
<td>Aracne Index Aracne Search Engine server</td>
<td>&lt;DENODO_HOME&gt;/conf/arn-index/ConfigurationParameters.properties</td>
<td>java.env.DENODO_OPTS_START</td>
</tr>
<tr>
<td>Web Container</td>
<td>&lt;DENODO_HOME&gt;/resources/apache-tomcat/conf/tomcat.properties</td>
<td>java.env.DENODO_OPTS_START</td>
</tr>
</tbody>
</table>

4. After saving the changes in the modified files, execute the script <DENODO_HOME>/bin/regenerateFiles
The next time you launch any of the Denodo servers, they will run with the new parameters.
# APPENDIX

## 9.1 DEFAULT PORTS USED BY THE DENODO PLATFORM MODULES

The table below lists the TCP/IP port numbers that each module of the Denodo Platform listens for incoming connections.

If these modules are behind a firewall, you must open the appropriate ports in the firewall. If the Denodo Platform runs on Windows, remember to open the ports in Windows Firewall as well.

These are the default port numbers. They can be changed during the installation process or later, in the administration tool of each module.

<table>
<thead>
<tr>
<th>Server</th>
<th>Default Port</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Virtual DataPort and ITPilot Wrapper Server</strong></td>
<td></td>
</tr>
<tr>
<td>Server port (Virtual DataPort administration tool and JDBC port)</td>
<td>9999</td>
</tr>
<tr>
<td>ODBC port</td>
<td>9996</td>
</tr>
<tr>
<td>Auxiliary port</td>
<td>9997</td>
</tr>
<tr>
<td>Shutdown port</td>
<td>9998</td>
</tr>
<tr>
<td><strong>ITPilot Browser Pool</strong></td>
<td></td>
</tr>
<tr>
<td>Server port</td>
<td>6001</td>
</tr>
<tr>
<td>Shutdown port</td>
<td>6002</td>
</tr>
<tr>
<td>Auxiliary port</td>
<td>6003</td>
</tr>
<tr>
<td>Initial Microsoft Internet Explorer browser port</td>
<td>6100.</td>
</tr>
<tr>
<td>The Browser Pool listens to this port to communicate with the first opened browser. The consecutive ascending port numbers will be used when additional browsers are requested.</td>
<td></td>
</tr>
<tr>
<td>For example, if the size of the browser pool is set to 30, the Browser Pool will listen in the port range from 6001 to 6030.</td>
<td></td>
</tr>
<tr>
<td><strong>ITPilot Verification Server</strong></td>
<td></td>
</tr>
<tr>
<td>Server port</td>
<td>7001</td>
</tr>
<tr>
<td>Shutdown port</td>
<td>7002</td>
</tr>
<tr>
<td>Service</td>
<td>Port Numbers</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td><strong>Auxiliary port</strong></td>
<td>7003</td>
</tr>
<tr>
<td><strong>ITPilot PDF Conversion Server</strong></td>
<td></td>
</tr>
<tr>
<td>Server port</td>
<td>8448</td>
</tr>
<tr>
<td><strong>Aracne Server</strong></td>
<td></td>
</tr>
<tr>
<td>Server port</td>
<td>11000</td>
</tr>
<tr>
<td>Auxiliary port</td>
<td>10998</td>
</tr>
<tr>
<td>Shutdown port</td>
<td>10999</td>
</tr>
<tr>
<td><strong>Aracne Exchange Crawler Server</strong></td>
<td></td>
</tr>
<tr>
<td>Server port</td>
<td>6000</td>
</tr>
<tr>
<td><strong>Aracne Index / Search Engine Server</strong></td>
<td></td>
</tr>
<tr>
<td>Server port</td>
<td>9000</td>
</tr>
<tr>
<td>Auxiliary port</td>
<td>8998</td>
</tr>
<tr>
<td>Shutdown port</td>
<td>8999</td>
</tr>
<tr>
<td><strong>Scheduler Server</strong></td>
<td></td>
</tr>
<tr>
<td>Server port</td>
<td>8000</td>
</tr>
<tr>
<td>Auxiliary port</td>
<td>7998</td>
</tr>
<tr>
<td>Shutdown port</td>
<td>7999</td>
</tr>
<tr>
<td><strong>Denodo Platform Web container</strong></td>
<td></td>
</tr>
<tr>
<td>(Scheduler, ITPilot and Aracne Administration Tools and Virtual DataPort Web services)</td>
<td></td>
</tr>
<tr>
<td>Web container port</td>
<td>9090</td>
</tr>
<tr>
<td>Auxiliary port</td>
<td>9098</td>
</tr>
<tr>
<td>Shutdown port</td>
<td>9099</td>
</tr>
</tbody>
</table>

*Table 1* Default TCP/IP port numbers opened by the Denodo Platform modules
REFERENCES

[DENMIGR] Denodo Platform 5.0 Migration Guide